Interaction Concept and Prototype to optimise the Virtual Product Development Process in the Automotive Industry

Friedrich Bachinger

MASTERARBEIT

eingereicht am
Fachhochschul-Masterstudiengang

Interactive Media

in Hagenberg

im September 2019

Declaration

I hereby declare and confirm that this thesis is entirely the result of my own original work. Where other sources of information have been used, they have been indicated as such and properly acknowledged. I further declare that this or similar work has not been submitted for credit elsewhere.

Hagenberg, September 23, 2019

Friedrich Bachinger

Contents

Abstract

The thesis deals with the development of an intuitive interaction concept in virtual reality, in which a variety of complex operations needs to be integrated. More precisely, a prototype has to be created that can be used in the product development process of the automotive industry. A user-oriented approach is adopted for achieving this goal. This method enables determining the needs of the various stakeholders involved in digital product development.

The developed interactive system should operate with virtual hands as well as with motion controllers. Stakeholders define different user groups in the system, each equipped with various features. Users are allowed to quickly and intuitively access tools required for analysis tests. They can modify geometries, styles and entire packages on the vehicle. Besides, options are offered for handling animations, assuming seating positions, aligning characters and selecting different environment maps. An additional operator is obsolete as the user controls all actions in the virtual environment.

The user has several ways to interact with the user interface, depending on the work to be done and their preferences. The system is designed in such a way that operations can be carried out entirely one-handed when using motion controllers. Furthermore, the system allows a seamless transition to other interaction techniques. Users can choose between trackpad and laser pointer via motion controllers, as well as virtual hands.

A total of two experiments were carried out, including a small number of participants. Different methods were used to achieve quantitative and qualitative results regarding usability and user experience. These studies highlight the iterative development process of the interaction concept, where weak points were identified and eliminated.

Kurzfassung

Diese Thesis befasst sich mit der Erstellung eines intuitiven Interaktionskonzeptes in Virtual Reality, bei dem es eine Vielzahl von komplexen Operationen zu integrieren gilt. Genauer gesagt, soll ein Prototype erstellt werden, welcher nach Fertigstellung im Produkt Entwicklungs Prozess in der Automobilindustrie eingesetzt werden kann. Um dieses Ziel zu erreichen wird ein benutzerorientierter Ansatz angewendet. Mit diesem ist es möglich, die Bedürfnisse der unterschiedlichen Stakeholder zu bestimmen, welche in der digitalen Produktentwicklung involviert sind.

Das entwickelte interaktive System kann sowohl mit virtuellen Händen, als auch mit Motion Controllern operieren. Stakeholder definieren dabei unterschiedliche Benutzergruppen im System, die mit verschiedenen Features ausgestattet sind. Benutzer können dabei beispielsweise schnell und intuitiv auf Werkzeuge zugreifen, welche bei Analysetests benötigt werden. Das interaktive System bietet dem Benutzer diverse Möglichkeiten an, Geometrien, Styles und ganze Pakete an einem Fahrzeug zu ändern. Zudem werden Optionen angeboten, Animationen auszulösen, Sitzpositionen einzunehmen, Charakter auszurichten und unterschiedliche Maps auszuwählen. Ein zusätzlicher Operator ist nicht notwendig, da alle Operationen von dem Benutzer selbst in der virtuellen Umgebung gesteuert werden können.

Der Benutzer hat mehrere Möglichkeiten mit dem User Interface zu interagieren. Diese sind abhängig von der zu erledigenden Arbeit und den eigenen Präferenzen. Das System ist dabei so gestaltet, dass die Bedienung bei dem Einsatz mit Motion Controllern komplett einhändig bedient werden kann. Des Weiteren erlaubt das System einen nahtlosen Übergang zu einer anderen Interaktionstechnik. Hierbei steht dem Benutzer die Eingabe über das Touchpad, Laser Pointer und über virtuelle Hände zur Verfügung.

Insgesamt wurden zwei Experimente mit einer kleinen Anzahl an Teilnehmern durchgeführt. Dabei wurden unterschiedliche Methoden angewendet, um quantitative und qualitative Ergebnisse hinsichtlich Usability und User Experience zu erzielen. Diese Studien verdeutlichen den iterativen Entwicklungsprozess des Interaktionskonzeptes, bei dem Schwachstellen gefunden und eliminiert werden konnten.

Acknowledgement

First of all, I would like to thank my supervisor, Dr. Matthias Oberhauser, who was a fantastic support in writing this thesis. He was always interested in discussing new insights during the development of the project. Furthermore, he was able to challenge me with his knowledge. His expertise and passion in the field of human-computer interaction and user studies were a great motivation in all aspects.

Secondly, I would like to thank my second superior, Mag. Volker Christian, who established the contact with BMW and trusted my abilities regarding the project. He was always willing to listen and gave constructive feedback. He was able to fulfil all my requests with great satisfaction.

Thirdly, I would like to thank the entire department EI.25 of *BMW Group Research and Innovation Center* (FIZ) in Munich who have been very helpful. Open questions regarding interactions in virtual reality, visual aspects of the design and the layout, as well as technical problems with the treatment of blueprints I always was able able to find an interested and helping person. Almost the entire department could participate in the experiments and provided valuable input.

In the final words, I would like to thank my family, who supported me during the work and the whole education. This thesis would not be written without them.

Chapter 1

Introduction

Virtual reality (VR) is being considered as the next big thing in Industry 4.0. It is possible to visualise the most realistic environments in a simulated virtual world thanks to highly productive game engines and high-performance graphics cards. Car manufacturers come up with appealing demo reels where new use cases of VR are shown with engineers and designers of tomorrow[1]. Looking from a new perspective opens up new, unexpected opportunities to increase the impact of a companies product development.

However, most of the interactions with an immersive head-mounted display (HMD) are hidden from the viewer. This phenomenon can be caused by an unintuitive user interface that is hard to control. Another possibility could be the circumstance that operations have to be controlled outside of the virtual environment by an additional associate. Wouldn't it be great to perform all tasks by oneself inside of the created world as a product developer?

1.1 Motivation

Today's software applications allow the visualisation of three-dimensional content directly when assembling geometry for VR. Especially in VR production design reviews, the user is interested in interacting with various tools to perform different analysis tests directly on the vehicle [57]. However, not every application does not satisfy all expectations of the user. As soon as a specific feature is needed, the company depends on the 3rd party developers who maintain and update an application. In the worst case, a highly needed feature will not find its way into an application in the future.

This circumstance leads companies to develop software and features themselves to be able to use target-oriented and customised solutions. Game engine development is essential to enable this flexibility [57]. A suitable user interface has to be created that allows the user to interact within the virtual world. In the best case, the user should be able to perform all tasks without additional help.

The user interface allows direct communication between a human and a computer. The way we experience an interactive system and how effectively we can achieve our goals depends strongly on quality and intuitiveness [18][p. 277]. Thus, the user interface plays a very important role. Different input devices allow new possibilities for interaction.

[1] https://www.youtube.com/watch?v=9cW48GLx1nM

These opportunities mean extra challenges to master when it comes to high-quality interactive systems.

Users who work in VR may have different ideas about how interactions could take place. Beginners and advanced users act individually in situations. Maybe some interaction techniques have already been learned or are incorporated intuitively. Thereby two things should not be mixed: innovative and intuitive interaction techniques. Innovation[2] involves introducing something new. Whereas intuitiveness is based on a presumption[3]. In this work, it is not a top priority to develop something entirely new, but rather to combine already known and established interaction techniques. The user should be able to perform complex tasks and operations as intuitively as possible. Besides, it is essential to create a uniform system that satisfies all the needs of the various stakeholders.

1.2 Research Questions

In this thesis, an intuitive interaction concept in VR for the product development process had to be created. An analysis of three existing VR applications in the field of automotive customisations was carried out. Weak points, as well as positive trends, were identified. Based on this knowledge, an interaction design was established that included all requirements from the analysis. The realised interaction design had to be evaluated and improved during the course of the interactive process. The following research questions arose during the development process to find the optimal interaction concept:

RQ1: How should functions and operations be integrated into an interactive system which allows the user to interact with motion controllers and virtual hands?

RQ2: How should the menu structure be integrated into an interactive system so that various users with different prior knowledge can intuitively use it?

RQ3: How can complex switching operations of a vehicle be easily performed in the product development process?

RQ4: How should an interaction concept be designed to use different tools and workflows as efficiently as possible within the product development process?

1.3 Contributions

The main contribution of this thesis lies in the development of an interactive system for immersive VR. In particular, the system should contribute to the productivity of various stakeholders in an application used in the automotive development process.

Another contribution of this thesis are the results of the continuous evaluation of the interactive system. Similar projects can benefit from this data. Furthermore, the requirements of different stakeholders are analysed, formalised and specified. These insights are valuable for future projects.

A further contribution is to help software engineers who are involved in the optimisation of the interactive system. This thesis describes how the system can be integrated

[2] https://dictionary.cambridge.org/dictionary/english/innovative
[3] https://dictionary.cambridge.org/dictionary/english/intuitive

into an existing project at BMW. All relevant menu components are broken down into their essential parts and described in detail to develop additional features and integrate them into the system.

1.4 Structure of Thesis

Chapter 1 describes the motivation of this thesis and the expected contributions. The further structure of this thesis is explained in the following paragraphs.

Chapter 2 covers the state of the art, which provides an overview of the product development process in the automotive industry and the role of VR. The relevant aspects of usability, user interface and interaction design are discussed.

In Chapter 3, the current BMW visualisation process is presented, and the general system requirements are specified. The stakeholders and the associated user stories are defined in this Section. The conclusion is offered by three user interface reviews, which are related to the subject matter of automotive applications.

Chapter 4 shows the realisation of the interactive system based on the conducted analysis. The necessary tools which are used for the prototype are explained. The planning process, the construction, as well as the description of the interactive system is part of the thesis.

In Chapter 5, two user studies are discussed which deal with interaction techniques and concepts for immersive VR. Both user studies were conducted in the BMW's premises as part of laboratory experiments. The methodology, data collection and the evaluation of the developed system can be found in this Section. Two user studies are conducted to receive feedback on the designed system and show potential for further optimisation.

Chapter 6 contains the results of the user studies evaluation, where the research questions are answered. The Chapter ends with the identified and known hard- and software limitations.

Chapter 7 provides a summary of the thesis and recommendations for future studies are discussed to continue this research. This Chapter gives an outlook on further improvements and possibilities of the designed system.

Chapter 2

Background

This chapter covers the state of the art about the product development process in the automotive industry and virtual reality (VR). The introduction is followed by a definition of usability and different usability evaluation methods. The user interface is discussed with a focus on graphical menus and interface metaphors. The interaction design forms the conclusion. It begins with a definition, which is then followed by the user-centred design approach. The interaction part discusses an extract of different techniques and patterns in VR and closes with principles of interaction design.

2.1 State of the Art

The Section state of the art contains all necessary information on the current status of the design review process in the automotive industry. A short excursus follow the introduction into VR that leads to the conclusion of this Section.

2.1.1 Product Development Process

Developing a vehicle is associated with enormous costs and is estimated between $ 1.000.000.000 and six times that amount [57]. Optimising the research effort and the development process is mandatory to keep the development costs low. In the automotive industry mockups were used extensively during the development. Mockups are full-scale prototypes lacking most features of a functional vehicle. With the rise of Computer-Aided-Engineering (CAD) more and more of these mockups were replaced with digital ones, computer simulations or VR applications at the end of the last century [57].

Once the concept phase is properly completed, the next step is series-development in which graphic design sketches and computer graphics renderings are transferred into three-dimensional (3D) space. Based on this data, 3D models can be created and experienced. 3D visualisations provide the production team and various decision-makers a better understanding of the current state of the development. This process is defined as virtual prototyping and is an important part of the development cycle [34, pp. 10–17].

Virtual technologies are used within the development process to increase efficiency and to ensure precise implementation [61]. Immersive VR helps to simplify cooperation between the collaborating departments and is no longer tied to a model in the virtual environment (VE). Thus, the design process gains speed. New questions arise with

Figure 2.1: BMW opts to incorporate HTC Vive VR headsets and mixed reality into the design review process. Source: [54].

regards to product design during reviews with VR [60]. These VR experiences help the stakeholders to better understand the spatial relationships between the product components [4]. They facilitate collaboration between different teams in various locations. These tasks are summarised under the term *Design Review Process* in VR [34, p. 17]. Figure 2.1 shows how BMW uses VR in the production of manufacturing vehicles. A simplified mockup is used to create a basic sensation of haptic and sense of space. The rest of the interior and exterior of the vehicle is presented fully virtual through VR-glasses.

CAD has already reduced many prototypes and mockups and in the future VR will make some more of them redundant. Nevertheless, the production of mockups and prototypes will be inevitable for some use cases.

2.1.2 Virtual Reality

In this thesis, the term virtual reality is used to describe computer-generated digital environments. These environments can be experienced, and interactions can be performed with objects in the virtual world. A recent definition of VR is available on the Merriam-Webster website [53]:

> "...artificial environment which is experienced through sensory stimuli (such as sights and sounds) provided by a computer and in which one's actions partially determine what happens in the environment".

This work focuses on the interface between the user and an interactive system. VR design concentrates on the transfer of knowledge about how a world works in VR, how objects can be controlled and monitored, as well as what relationships exist between the elements and the user. VR experiences require a special understanding of the user and systems development. The best experience can be expected when both elements work perfectly together [18, p. 10].

The first stereoscopic images were taken in the 1800s before Sir Charles Wheatstone's photography [18, p. 10]. An important achievement was made in 1968 by Ivan Sutherland [40], who wrote a paper of a 3D head-mounted display (HMD). He describes the development of a head-mounted stereoscopic display called *Sword of Damocles*. The HMD could use a mechanical or an ultrasonic tracker and was able to visualise computer-generated wireframe models [38, p. 27].

After a short-lived hype, driven by consumer electronics companies like Nintendo or Sega in the early 1990s, the first decade of the 21st century is known under the name *VR Winter*. Not much attention was paid to VR technology [18, p. 27]. VR was mainly used for niche products in science and industry. However, the consumer market was not affected.

VR became increasingly popular, after a much-noticed Kickstarter campaign[1] from *Oculus* that introduced an inexpensive HMD and the use of even more cheap VR glasses based on smartphone technology. Different sensors such as stereo cameras, six degree-of-freedom (DOF) motion controllers or gloves made alternative input forms possible. Some of them had quickly evolved and established themselves as standard input devices [38, p. 31]. There would never have been a VR revival in the consumer electronics market without the *Oculus Rift* [42].

2.2 Usability

The definition of usability and the usability of complex information are discussed in this Section. Insights about various tests to quantify usability will be presented. The final Subsection shows usability heuristics, which can be applied to a wide range of interactive systems.

2.2.1 Definition

The *International Organization for Standardization* (ISO) defines the ergonomics of human-system interactions in *ISO 9241-11:2018* [37]. According to the ISO definition, a system is used by a specified set of users to achieve specified goals of effectiveness, efficiency and satisfaction in a specified usage context [15, p. 71]:

> "Usability is most often defined as the ease of use and acceptability of a system for a particular class of users carrying out specific tasks in a specific environment."

Design is not the only aspect of whether a product has good usability. It is considered to be a part of the user experience and refers to making a product more accessible and usable. It is composed by combining functions and the context of the user. There are three major outcomes regarding the above mentioned *ISO 9241-11*. The objective of the ISO standard is to make it as easy as possible for the user to become familiar and competent with the user interface. The users should be able to achieve their objective easily. The functionality of a user interface should be easy to remember to make it accessible for subsequent visits [46].

[1] https://www.kickstarter.com/projects/1523379957/oculus-rift-step-into-the-game

Table 2.1: Comparison of Usability Evaluation Techniques [15, p. 72]

	Inspection Methods			*Test Methods*		
	Heuristic Evaluation	Cognitive Walkthrough	Action Analysis	Think Aloud	Field Observation	Questionnaires
Applicably in Phase	all	all	design	design	final testing	all
Required Time	low	medium	high	high	medium	low
Needed Users	none	none	none	3+	20+	30+
Required Evaluators	3+	3+	1-2	1	1+	1
Required Evaluators	low	low	low	high	medium	low
Required Equipment	medium	high	high	medium	high	low
Intrusive	no	no	no	yes	yes	no

2.2.2 Usability of Complex Information Systems

Complex usability can be divided into two dimensions: technical complexity and contextual complexity. The technical complexity includes fundamental interaction characteristics or complex system architectures that contain databases or other devices. *Metaphors*, which are described in the upcoming Section 2.3.2, can be used to hide the technical complexity from the user. The contextual complexity deals with the available actions and role distributions that an interactive system provides to the user to fulfil the needs in specific situations [2, pp. 283–284].

2.2.3 Usability Evaluation Techniques

Table 2.1 provides an overview of possible usability evaluation techniques, split up two groups. On the left side, inspection methods are defined, and on the other side, test methods are listed. The first group contains the heuristic evaluation, cognitive walkthrough and action analysis. In this group, usability problems are identified, and the interface design is improved by testing already established standards. The second group includes thinking aloud, field observation and questionnaires. Valuable information can be obtained directly from the end user with these test methods. Some of these methods can be applied at all development stages, while others can only be used at certain ones. Problem areas can be identified in the user interface [15, pp. 72–73].

- *Heuristic Evaluation:* There are a variety of usability heuristics with different focuses. The usability heuristics provided by Jakob Nielsen [51] is applied for the analysis of the interactive systems. Even if it was developed in 1994, it can be used for games, web applications or VR applications through continuous development. These heuristics contain ten general principles for interaction design. However, they are called heuristics because they are rather broad rules of thumb than specific usability guidelines [32, sec. 15.2.1]. A revised version of these heuristics can be found in Table 2.2. Noteworthy in this context is the heuristics developed by the

Youtube UX Research Team [30]. This heuristic focuses on realistic interactions for VR systems without an integrated menu system. Another Heuristic evaluation to be mentioned comes from Sutcliffe Alistair and Gault Brian [39]. The focus lies on realistic interactions as well and is not suitable for the analysis of this thesis.

- *Cognitive Walkthrough:* The detailed design of a prototype is examined depending on logical steps of interactions in a cognitive walkthrough. This method is a paper-based technique and was developed in 1990 by Lewis Clayton [25]. The test involves an analysis of cognitive tasks which a test person has to complete. The walkthrough is based on well-established theory, and less on trial and error, as heuristic evaluation methods do [3, p. 73].
- *Action Analysis:* It is more important what the participants do than what they say in this type of test. The steps to complete a task are closely monitored. The action analysis is also known as keystroke analysis. There are two methods for performing a test. Method one breaks down the individual actions performed and measures the amount of time taken. The second method is called back-of-the-envelope analysis and is less detailed but faster. The focus of this kind of walkthrough lies on physical, cognitive and perceptual stress [15, p. 73].
- *Think Aloud:* The participants are asked to think aloud to get an insight into why a decision was made while performing experiments. Participants are supposed to verbalise everything that goes through their minds when solving the tasks. The data obtained from this inspection method is transcribed. The result is a protocol called *Think Aloud* [16].
- *Field Observation:* This type of monitoring is simple to apply. The persons to be tested are visited at their workplace. Notes are taken on how the participants work with a product [15, p. 74].
- *Questionnaires:* This type of evaluation can be carried out over distance. There is no need for face-to-face monitoring. The use of questionnaires is suitable for larger amounts of participants in an experiment where the basic understanding of a product is to be analysed. The creation of questionnaires is considered difficult and time-consuming. It is possible to use standard questionnaires that have proven themselves in many cases [3, pp. 146–147].

2.3 User Interface

Good interface design has already been firmly integrated into systems in desktop computers and mobile applications. It should be easy and intuitive to use to ensure that everyone can work efficiently with the same system [5, p. 4]. A interface design approach is mentioned by Jason Jerald [18, p. 277]:

> "Whether a VR interface attempts to be realistic or not, it should be intuitive. An intuitive interface is an interface that can be quickly understood, accurately predicted, and easily used. Intuitiveness is in the mind of the user, but the designer can help form this intuitiveness by conveying through the world and interface itself concepts that support the creation of a mental model."

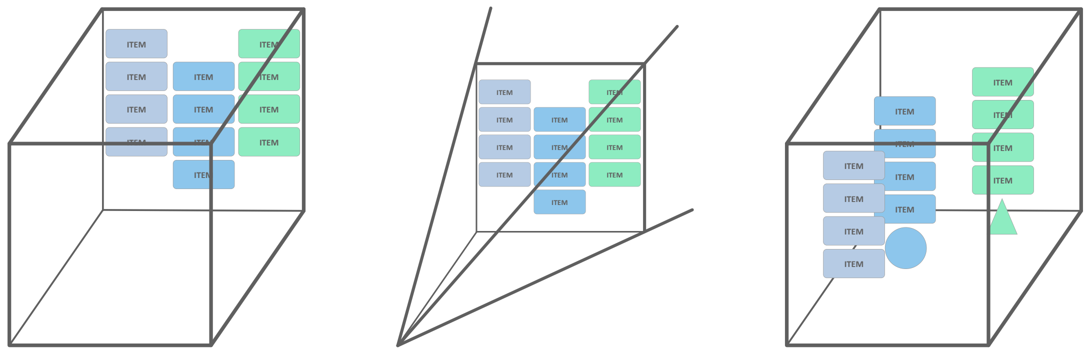
Table 2.2: 10 usability heuristics for user interface design by Jakob Nielsen [51].

Name	Description
Visibility of system status	The system should always keep users informed about what is going on, through appropriate feedback within a reasonable time.
Match between system and the real world	The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.
User control and freedom	Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.
Consistency and standards	Users should not have to wonder whether different words, situations, or actions mean the same thing.
Error prevention	Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.
Recognition rather than recall	Minimise the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for the use of the system should be visible or easily retrievable whenever appropriate.
Flexibility and efficiency of use	Accelerators — unseen by the novice user — may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.
Aesthetic and minimalist design	Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.
Help users recognise, diagnose, and recover from errors	Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.
Help and documentation	Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

This Section gives an overview of the different graphical menus used in VEs. Moreover, the menus are categorised by display location. This is followed by the interface metaphors concept, which provides development and user relief for complex systems. The conclusion of this Section offers principles and standards that have already established for use in VR.

2.3.1 Graphical Menus

A menu provides a collection of items containing options and commands where the user can select an item interactively. Menus provide the user with a reduced selection through a structural arrangement. This helps the user to navigate more easily through the menu and make careful selections [17]. Menus can be arranged differently in the VE. Figure 2.2 illustrates three different ways to display menu systems in VEs [20]. The menu system of *World-Fixed* has a fixed place in the virtual space. The *View-Fixed* menu is attached to the HMD and displays the menu in the user's field of view. The *Object-Fixed* menus



(a) (b) (c)

Figure 2.2: Different arrangements of graphical menus in the virtual environment. *World-Fixed* (a), *View-Fixed* (b), *Object-Fixed*. Adapted from [20].

are attached to virtual objects such as a handheld controller.

Most of these 3D interaction techniques have their origin in two-dimensional (2D) solutions. This started with the so-called *Window, Symbol, Menu, Pointing Device* (WIMP) elements and is assigned to the *desktop metaphor*, which is discussed in Section 2.3.2. Since the 1990s, all developments that do not take place based on mouse interactions are assigned to the *Post-WIMP* [33, p. 464]. In this era 2D interaction metaphors were transferred to 3D VEs [11, p. 55].

Graphical menus in 2D have proven themselves successfully in 3D environments. Many developers use this type of interaction in their applications since this technique is already known to a large number of users [5, p. 386]. Graphical menus can be divided into three categories:

- *Adapted 2D Menus:* Adapted 2D menus work just like in desktop applications. These include pull-down menus, pop-up menus or floating menus. They are often used when more complex functions are required in a system [5, p. 387]. Menus can be structured in different ways. The structure helps the user to learn the application faster and to organise the existing interactions. A hierarchical menu should be considered. Such menus must be carefully designed if an application has complex functions. Otherwise, users will become confused and in the worst case, quit the application [17]. The best choice is to integrate hierarchical menus, in 2D environments as well as in 3D VEs, when many commands exist in a complex architecture [20]. A special variant of *Adapted 2D Menus* is the pie menu, which is often used in combination with gaze technologies. This type of menu provides a hierarchical structure in a circular space-saving arrangement. The huge advantage, when eye gaze technology is used, is that no controllers are necessary to navigate through the menu [5, p. 387].
- *Three-Dimensional Widget:* 3D widgets contain a geometry assigned to functions and behaviours. Such 3D widgets can be considered as interaction elements between traditional 2D and immersive 3D user interfaces [10]. The term *Degrees Of Freedom* (DOF) is required to provide an accurate description of the orientation and location of 3D widgets. This DOF characterises independent dimensions in

which interactions are possible. Objects can be moved in three directions in space and rotated in three directions. The result is in a total of six degrees of freedom [33, p. 245].

- *1-DOF Menus:* Selecting a menu item is a one-dimensional (1D) operation. Due to this consideration, 1-DOF menus were created. In contrast to the *Adapted 2D Menu*, 1-DOF menu has only one hierarchy level and is 3D. Very often 1D menus are placed around a wrist or a controller known as the *Ring Menu* [28]. In a circular menu, hand rotations serve to reach all menu items. A button can be used to execute the desired operation for an actual selection. Handheld widgets can be used as well, allowing the hand positions to be used as a selection instead of rotation. The distance between the hands allows selecting different menu items [5, pp. 390–391].

2.3.2 Interface Metaphors

User-understood concepts and frameworks are often used to master a user interface's complexity in terms of actions, tasks and goals. These frameworks are called interface metaphors [8, p. 441]. Many interface design books recommend using such frameworks. Interface metaphors are central components of *Conceptual Models*. They define the behaviour and properties [32]. There are a variety of metaphor classifications. Ideally, the same metaphor should be reused within an application, which can be quite difficult to implement in practice. Different operating concepts are needed, especially for complex applications. They must be obvious and visible to the user [18, p. 301]. Interface Metaphors provide concepts, criteria and a common vocabulary for a better exchange of information. The argumentation on why a decision was made is facilitated [23, p. 451].

Figure 2.3 visualises the interaction of the used user interface components. The *Conceptual Model* reflects the designers point of view. The opposite is the users mental model, which deals with the tasks and goals of the user. The connection, the *System Image*, corresponds to the physical structure implemented in the user interface. This model should create a bridge between the designer and the user [23, pp. 447–448]. In software development, the difference between the *Conceptual Model* and the user mental model can be huge. Due to the complexity of the system, the user might not recognise causal relationships between action and reaction. Therefore, the represented model should be as simple as possible to allow the user to become easier familiar with an application [9, p. 29]:

> "The closer the represented model comes to the user's mental model, the easier he will find the program to use and to understand."

A central task of user-models is to create reality as simple as possible for the user. This makes it easier for the operator to understand a complex system without knowing how the system works in the background. Furthermore, a user model does not necessarily have to be true but should be efficient in the application [9, pp. 29–30].

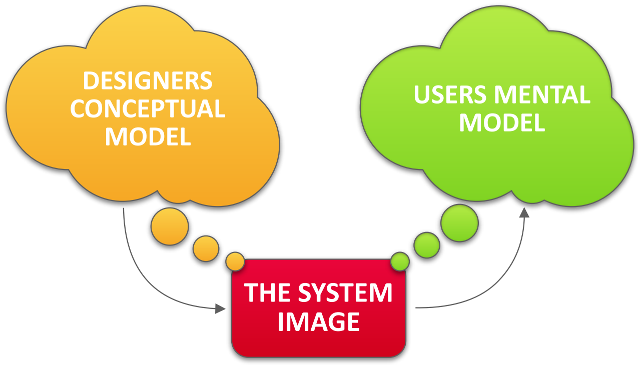


Figure 2.3: Model of the user interface. Adapted from [31].

2.4 Interaction Design

This Section provides an overview of the interaction design process. A definition of this topic is given in the beginning. Concept models are described in more detail to clarify the interactions between the positions of the various stakeholders. Consequently, there is an insight into the user-centered design approach, where the user moves to the centre of attention. This is followed by human-centred interaction aspects where different techniques and patterns are discussed. Interaction design principles are explained in more detail at the end of this Section.

2.4.1 Definition

Interaction design refers to interactions between an application and a user. The goal is to provide an interaction concept to enable users to achieve goals most effectively. A definition is available from the *Interaction Design Foundation* [45]:

> "...it is the design of the interaction between users and products. Most often when people talk about interaction design, the products tend to be software products like apps or websites. The goal of interaction design is to create products that enable the user to achieve their objective(s) in the best way possible".

2.4.2 User-Centered Design

The user-centered design approach focuses on the user to enable high usability and user experience. John D. Gould and Clayton Lewis [12] describe principles that lead to successful interaction design: The attention should be focused on the users and their tasks already at the design phase of the product. The end-users know best what they need and can provide the designer with all the necessary information to find an individual and tailor-made design solution. The purpose is to understand the requirements for creating a design that all needs are fulfilled. It is crucial to weight the conflicting demands carefully. An important decision is how much freedom is given to the user. Proper balancing is crucial to not overload a system unnecessarily [32, sec. 9.2].

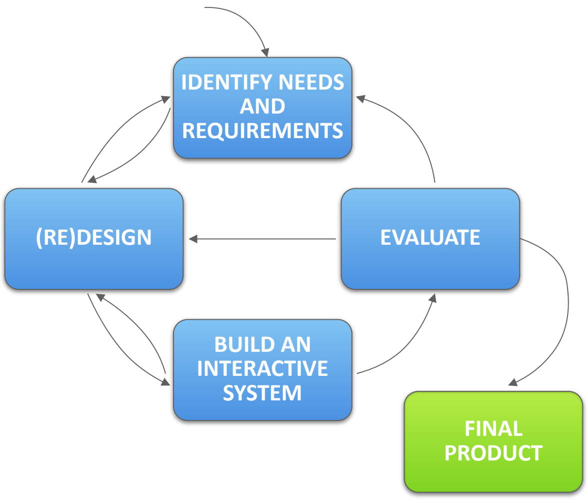


Figure 2.4: A simple interaction design life-cycle model. Adopted from [32, sec. 9.2.5].

Empirical measurements should accompany the development process. The reactions of the users are taken into account as soon as possible during the first design prototypes. User feedback can be incorporated directly into the development process instead of just working on a technical solution as a result. The gained insights can contribute to making alternative design decisions and monitoring the progress of a prototype [32, sec. 9.2.4].

In the iterative design process, errors in the design can be found during testing. After these errors have been corrected, further tests are performed by users. The changes and effects are monitored. This process should be conducted as often as necessary to ensure user satisfaction [32, sec. 9.2]. Once end-users are involved in the development process, they feel more connected to a product. Consequently, they are likely to support the system and work with it in the future [32, sec. 9.2.1].

Figure 2.4 shows a simple interaction design life-cycle model. This model illustrates how the individual building blocks fit together in the user-centric approach. The final product is the result of the requirements, the design alternatives, the developed prototypes and the evaluation.

2.4.3 Conceptual Models

The development of a *Conceptual Model* [19] begins after an analysis of all functional user requirements are available. The goal is to create metaphors and to specify analogies. It contains concepts on how to manage objects and change their attributes. The operations that can be performed within a system should be listed. This procedure clarifies the relationships between the concepts. Objects and actions are enumerated and checked for shared use. In the subsequent design process of the user interaction design, concepts can be accessed that are similar in execution. A hierarchical arrangement leads to a better overview and understanding of complex systems, which is discussed in Section 2.3.1. The last task in this process is to create the mapping between the concepts and the task area of the system [19, p. 25]. The analysis includes the declaration of used concepts, which are accessible to the user. If these declarations are to be made, one rule must be followed [19, p. 28]:

> "If it isn't in the conceptual model, the system should not require users to be aware of it."

It should be noted that the complexity increases exponentially with each added concept. An important principle is that the *Conceptual Model* is as simple as possible and contains all the requirements for a system. Therefore, it should be strictly observed whether concepts are essential [19, p. 28].

A unique lexicon created for the *Conceptual Model* provides all relevant names for objects, actions and attributes for the entire involved team. A common vocabulary is accessed throughout the iterative development process. Misunderstandings are eliminated with consistent usage. Once a *Conceptual Model* has been developed, the use cases should be created. These are scenarios that describe the user requirements for the system. The approach of how the user wants to perform actions should be in focus. Use cases are a central and reliable supporter to create a user-oriented interaction concept. The user interface, layout decisions, as well as the interactions to be developed are based on this *Conceptual Model*. At this level, previously defined use cases can be rewritten and adapted. Errors in the model that become visible during the test should be corrected immediately. The *Conceptual Model* forms a central interface between all persons involved and should always be kept up to date [19, p. 28].

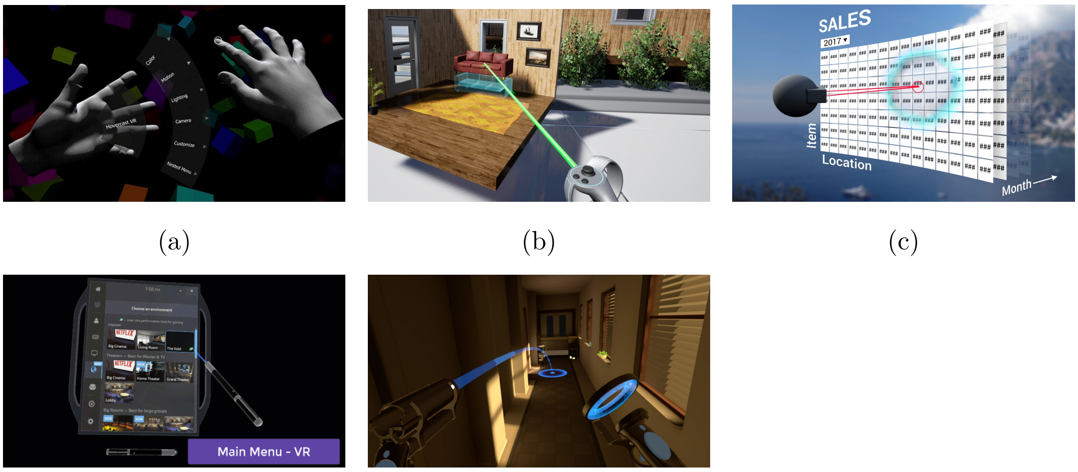
The goal is to generate a simple *Conceptual Model* where all functions are easy and intuitive to use. Care should be taken when extending the model, as complex overlays can occur very quickly, and an excellent initial concept cannot be achieved. Applications usually contain several ways to execute a function. The user has to learn all versions at least once to use the most suitable one. It should be remembered that once an interaction has been learned, users want to use it permanently. Ideally, such interactions can also be found in other applications. Careful consideration must be given to whether the same operations with different choices have a positive impact on the user experience compared to a user interface with constraints [32, sec. 2.3].

2.4.4 Interaction Techniques

There are two basic interaction techniques in VR: Realistic interactions and unrealistic interactions. Realistic interactions lead to a high degree of interaction fidelity. The aim is to implement the interactions in VR as realistically as possible like they take place in the real world [5, sec. 7.3.1]. This interaction technique is mainly used in training simulations, therapy and human-factors evaluations. In these cases, the interactions must be realistic. Otherwise, negative effects can be caused. Based on the natural and intuitive interaction technique, this kind needs only a short learning period [18, p. 289].

In contrast to the realistic interactions, the unrealistic ones are at the other end of the interaction fidelity such as when a key is pressed to trigger an action. This kind of interaction is not necessarily a disadvantage. The usage of this type allows a faster execution, fatigue symptoms decrease and can increase the enjoyment [18, pp. 289–290].

There are magical interaction techniques between these two previous techniques. These techniques extend the physical movements of the user with improved abilities



(a) (b) (c) (d) (e)

Figure 2.5: Different interaction patterns which can be applied in interaction metaphors. *Hand Selection Pattern* (a), *Pointing Pattern* (b), *Gaze Pattern* (c), *Widget Panels Pattern* (d), *Teleport Pattern* (e). Sources: [41, 44, 49, 50, 58].

and are separated from realism [5, sec. 7.3.1]. Very often, such interaction techniques use interface metaphors to create mental models for prototypes. The magic interaction techniques try to improve the user experience by reducing the fidelity of interaction and bypassing the limitations of the real world [18, p. 290].

2.4.5 Interaction Patterns

There are many different interaction patterns that can be applied in interaction metaphors. In this Section, only the patterns relevant to the thesis are listed. A collection of interaction patterns for virtual and augmented reality is available at the website from *Threesixty Reality*[2]. A definition of interaction patterns is as follows [18, sec. 28]:

> "An interaction pattern is a generalised high-level interaction concept that can be used over and over again across different applications to achieve common user goals."

Figure 2.5 shows five different interaction patterns, which are used in the virtual system. All these interaction techniques are discussed next:

- *Hand Selection Pattern:* This pattern selects or manipulates objects with touch interaction and belongs to the realistic interaction technique [18, p. 325] and can be seen in Fig. 2.5 (a). The interactions are typically based on virtual hands that, for example, press buttons. These kinds are particularly intuitive because interacting directly with objects in the virtual world is possible. However, this entails physiological limitations associated with the range of the arm. For objects that are not tangible, the user must first move to the location. In many cases, this is inconvenient and increases the complexity of the user interface [5, pp. 264–266].

[2] https://threesixtyreality.co.uk/interaction-design-patterns-library/virtual-reality

- *Pointing Pattern:* This pattern represents a very popular and basic pattern, especially for the selection of objects [18, p. 327]. A screenshot is visible in Fig. 2.5 (b). This pattern belongs to the magical interaction techniques. A ray is emitted in a defined direction from a certain point, starting from the user. As soon as an intersection between the ray and a selectable object is found, an event is triggered. Usually, the ray is emitted from a motion controller. This method allows the user to select or manipulate objects even from a distance easily. The pointing pattern belongs to the category *Vector-Based Pointing Technique* [5, p. 273].
- *Gaze Pattern:* The targeting with the *Gaze Pattern* in Fig. 2.5 (c) allows the user to use the focal point of the gaze when an HMD is used. This type of selecting is not possible in the real world, so it belongs to the magical interaction techniques. It is not essential where the user is looking with the eyes. The centre of the visual field of view is used as a cross-hair. A ray, starting from the cross-hair, is emitted in the direction of the field of view. As soon as an intersection between an interactive element and the ray is established, the user receives visual feedback to allow further interactions [48]. This works almost like with the *Pointing Pattern*, but in this case, no additional input devices are required. This pattern works with eye-tracking as well. The input technique allows better control of the cross-hair that adapts to the viewer's eye direction.
- *Widgets and Panels Pattern:* This pattern is one of the most widely used VR interaction techniques for indirect control and is considered as unrealistic and can be seen in Fig. 2.5 (d). It follows the typical 2D interaction pattern on desktop widgets or panel/window metaphors, which is described in Section 2.3.1. A widget is a geometric user interface element. Such widgets are placed in so-called panels, which serve as containers for multiple widgets. The correct placement must be ensured so that the widgets are accessible at any time and are not obstructed by other 3D geometry. Such panels can be connected to the hand controller or can float in the virtual world. The use of these elements is recommended for complex tasks where direct interactions with objects are difficult [18, p. 345].
- *Teleport Pattern:* The *Teleport Pattern* in Fig. 2.5 (e) enables the user to take a predefined position, which can be difficult to reach. The user sees a location marker as a signifier in the virtual space. The avatar is teleported to the new position after triggering the action utilising a ray cast and a pointing interaction [59]. This type of interaction techniques belongs to the magical ones.

2.4.6 Principles of Interaction Design

As soon as users want to interact with an interactive system, they have to figure out how the system works. They have to explore the system and find possible actions and operations to interact within the virtual world [18, p. 278]:

> "Discoverability is especially important for fully immersive VR because the user is blind and deaf to the real world, cut off from those real-world humans who want to help."

Essential tools help to facilitate the exploration for the user. There are many principles in this domain, and in the following, the *Principles of Norman's Interaction Design* [32, sec. 1.6.3] are listed:

- *Visibility:* All action options should be visible and clearly assigned to all existing controls. The more visible interaction options are available, the more likely it is that the user will know what to do next. If these functions are not visible to the user, finding them becomes more difficult, and the actual interaction is delayed [32, sec. 1.6.3].
- *Feedback:* In real life, we receive continuous feedback in all our activities. In the virtual world, we need feedback as well on the activities performed or the moment when an action goal is reached [32, sec. 1.6.3]. Feedback communicates the current state of a task being performed and helps to continue with further actions [18, p. 281]. Feedback can take the form of audio, tactile, verbal, visual or a combination of both. These include sensory substitution, ghosting, highlighting, audio cues, passive haptics and rumbling [18, pp. 304–305].
- *Affordance:* Affordance means which actions are possible at a certain moment and how the user can trigger an interaction. The affordance describes the relationship between the possibilities the user has and the properties of an object [18, pp. 278–279]. As soon as one of the users recognises the affordability of an object, it is easier to perform an interaction. There are two types of affordability: Perceived and real. Physical objects have a good price-performance ratio, for example, when the user grabs an object. This kind of interaction has not to be learned. However, screen-based interactions are virtual and must be learned [32, sec. 1.6.2].
- *Constraints:* The restrictions apply to limitations of actions and behaviour. Such restrictions may be logical, semantic or cultural to facilitate interaction [18, p. 280]. The user should only be offered interaction possibilities that are active and available. For example, inactive elements should be greyed out [32, sec. 1.6.2]. The degrees of freedom play an important role in VR. Interactions can be simplified, and accuracy, precision, as well as user efficiency, increased if the restrictions are correctly applied. [5, p. 479].
- *Consistency:* Consistency deals with how an interface responds to similar tasks that occur because using identical operations and with the same elements. A consistent interface is one that follows the rules: for example, the user presses a particular key for a particular action over and over. An inconsistent interface tempts the user to make more mistakes when executing actions. It becomes increasingly difficult for the user to remember individual actions [32, sec. 1.6.2].
- *Mappings:* Mappings describe the relationship between the object to be controlled and the effect created during or after an interaction. Only when the user recognises a connection between the interactive element and the following event, a clear and understandable mapping can be made. Mappings in VR are particularly important. Often a controller has a natural mapping for one technique, but a bad mapping for another [18, p. 282].

Chapter 3

Analysis

The Chapter provides information about the BMW product development process, which describes the visualisation process, the general requirements on the interactive system and the stakeholders, including the user stories and use cases. Subsequently, an analysis of three VR user interfaces is performed. The first one is an internal solution of BMW, and the remaining two are free VR applications that have a focus on customisation of vehicles.

3.1 Visualisation Process at BMW

This Section deals with the current visualisation process of BMW. Subsequently, the requirements for the interactive system are defined in the form of user stories. Afterwards, the stakeholders are identified who will use the user interface. A use case diagram explains the relationship of stakeholders between each other and to the system.

3.1.1 Visualisation Process

The visualisation process is subordinated to the BMW development process and is called *Idea To Offer* (ITO). It defines how a new vehicle is developed from the initial idea to the end product. A concrete offer can be made to the customer in the end [47].

The design-technology convergence (DTC) is part of this process. The design model is transferred into a technical model during the DTC. The objective is to create an optimal composition of form and function. All factors that influence the design, the surface construction, the functionality and the impact of the various components are taken into account [55]. The design requirements have to be combined with the technical requirements without exceeding the time or cost involved.

This process often leads to trade-offs and essential decisions that highly influence the course of the vehicle project. Good decisions must be based on data and information. One of these sources is visualisation. The vehicle is continuously visualised during the DTC. All members of the project team, (designers, engineers, management) can see the possible visual outcome of the current state of development with this visualisation. A dedicated team is responsible for visualising the most up-to-date CAD data and providing it via different output devices (VR glasses, projectors, LED walls). The Unreal Engine is an important software package for this process that is in use at the time of writing.

3.1.2 General Requirements

The requirements analysis is an important process in the initial phase. The tasks, the users and the context are analysed, and economic framework conditions are derived. After completion of this process, the elaborated requirements in the form of user stories are incorporated into the development process [33, p. 50].

Experienced and unfamiliar users concerning VR uses the interactive system. Users without an affinity for gaming are involved as well in the use of the system. This circumstance requires the creation of various prerequisites to make it easier for inexperienced users to get started.

Like any application, this one has a particular focus, and a suitable solution has to be found. The menu must be optimised for the design review process and has to be compatible with the already existing 3D models, environments and functionality. The interactive components of the controller, as well as the motion controller, should be visible. All those elements may have the original form or be represented in an abstract shape. A direct view of the controller is not always guaranteed, as the posture must adapt to the elements to be examined. An important task is to change elements inside of a list blindly. Various tools should be selected via the user interface to support the design review process. The menu should be completely controllable one-handed since different tools can be operated with both hands.

Other motion controllers need to be integrated at a reasonable cost later. The menu folder architecture has a complex structure. The composition of the menu must be adaptable and flexible with considerable effort. This integration allows additional features to be added quickly. A suitable user interface should facilitate working with the interactive system. In project meetings, where people are face-to-face present, increased communication means working within a virtual world can take place in a noisy environment. This aspect must be taken into account when developing the input and feedback options of the user interface.

3.1.3 Stakeholder, User Stories and Use Cases

A *User-Centered Design* approach, as mentioned in Section 2.4.2, was chosen to gather the requirements for the development, based on use cases. User models are designed for the development to reflect on the users and their interactions. In this way, the user behaviour, the thinking process, the aspiration and the motivation reason can be captured. Complex structures and relationships can be mapped more effectively using these models [9, pp. 76–77]. In the further analysis process, those models are used for the development of *Conceptual Models*, which are discussed in Section 2.4.3. The *Unified Modelling Language* (UML) [35] is applied for modelling and visualizing the use cases. Figure 3.1 illustrates the users and their respective use. The different stakeholders are specified in the following paragraphs. An overview of the use-cases is given in Table 3.1.

As a *Product manager*, I am interested in the visualisation of the current overall state of the vehicle. It is essential for me to customise the packages of a car. In rare cases, I would like to change styles and geometry variants of a vehicle. I will use the system only for a short period of several minutes. Since VR is not integrated into my everyday work, I need a straightforward and intuitive system (Use Case 1.2). I want to avoid complicated interaction techniques (Use Case 1.1) or the use of analysis tools as

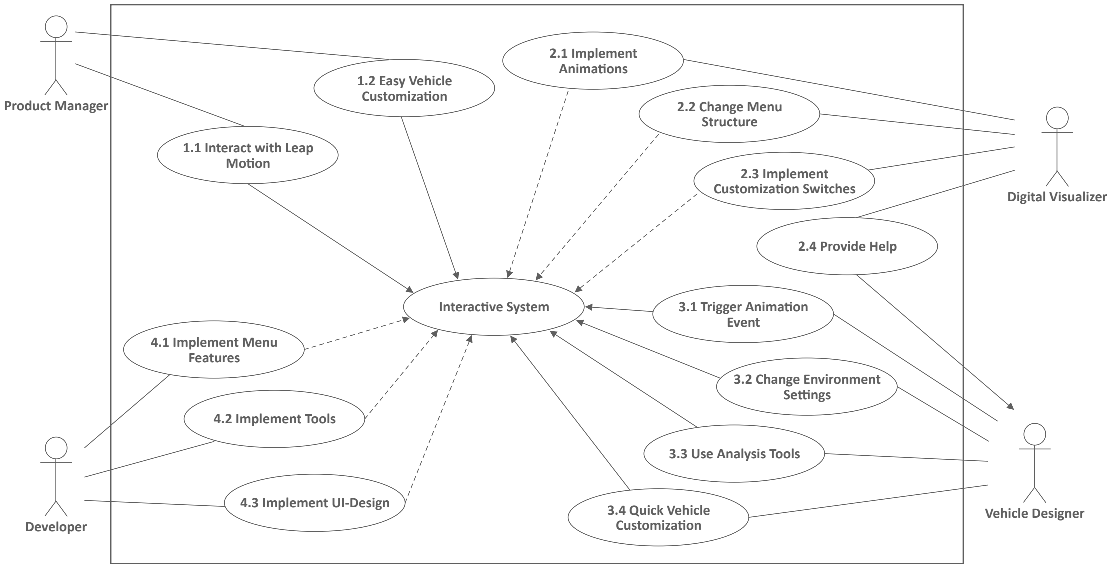


Figure 3.1: The UML Use Case Diagram for the interactive system.

much as possible because I am not familiar with the system and its menu structure.

As a *Digital Visualizer*, I am responsible for the visualisation of the virtual vehicle and for keeping the model up-to-date. I add animations (Use Case 2.1) and operations (Use Case 2.3) to the system that allows changes in the vehicle during the design review process. Since the parts of a vehicle change almost every week, the user interface should maintain itself after a basic installation. I want to customise the menu structure (Use Case 2.2) with less effort. Currently, I support different stakeholders by calling up the desired operations and functions. With the newly developed user interface, I am the first contact person for occurred interaction problems (Use Case 2.4).

As a *Vehicle Designer*, I am responsible for the appearance of the vehicle. I need various analysis tools (Use Case 3.3) to investigate surfaces and component integrations. I spend up to two hours in the virtual environment during an inspection. That is why I need an intuitive system that allows me to make quick changes to the vehicle (Use Case 3.4) and the environment (Use Case 3.2). I want to examine different design attitudes of car components. This analysis requires me to trigger animation events (Use Case 3.1) and operations to change the geometry of the vehicle (Use Case 2.3). Ideally, all functions and operations are controllable via motion controllers.

As a *Developer*, I am in charge of implementing menu features (Use Case 4.1), integrating tools (Use Case 4.2) and updating the user interface design (Use Case 4.3) of the system. A modular and generative architecture is the key to keep the time required for system updates as low as possible. The system maintenance adapts automatically to the needs of different stakeholders and requirements of the current vehicle status.

Table 3.1: Overview of identified use cases of the design and review process at BMW.

No.	Name	Goal/Functional Requirement
Product Manager Use Case		
1.1	Interaction with *Leap Motion*	The product manager should be able to use the Leap Motion device to have access to an adapted natural lightweight input system.
1.2	Easy Vehicle Customization	The product manager should be able to make easy changes to the geometry, styles and packages of the car.
Digital Visualizer		
2.1	Implement Animations	The digital visualizer should be able to implement animations, which are automatically assigned to the menu.
2.2	Change Menu Structure	The digital visualizer should be able to manage the menu structure comfortable with less effort.
2.3	Provide Help	The digital visualizer should be able to support the vehicle designer when interaction issues occur.
2.4	Implement Customization Switches	The digital visualizer should be able to implement custom switches, which are automatically assigned to the menu.
Vehicle Designer		
3.1	Trigger Animations	The vehicle designer should be able to trigger animations of the car and the environment.
3.2	Change Environment Settings	The vehicle designer should be able to make custom changes to the VE.
3.3	Use Analysis Tools	The vehicle designer should be able to use and select analysis tools and various workflows comfortably and quickly.
3.4	Quick Vehicle Customization	The vehicle designer should be able to make quick customisations on the car.
Developer		
4.1	Implement Menu Features	The developer should be able to implement the specific menu features, which are required in the design and review process.
4.2	Implement Tools	The developer should be able to implement new tools with a minimum of effort.
4.3	Implement UI-Design	The developer should be able to implement the UI-design to make the interactive system as intuitive as possible.

3.2 Review of Automotive Virtual Reality User Interfaces

Three applications were used for the analysis of automotive VR user interfaces. The first application is in the development stage and currently in use at BMW. It offers a solution for the design and review process when user interactions are needed. The other two applications are freely available on different platforms (Steam[1], VivePort[2]) for PC systems. These applications demonstrate the possibilities of VR for adapting and customisation of vehicles. The particular emphasis of the application is not considered. Only the user interactions are in focus.

The applications are subjected to an analysis evaluation using *Nielson's 10 usability*

[1] https://store.steampowered.com/
[2] https://www.viveport.com

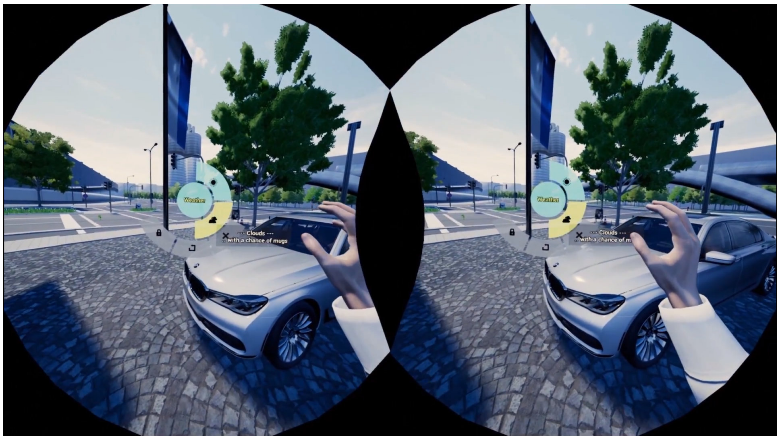


Figure 3.2: Screenshot of the application which is currently used at BMW.

heuristics, which is discussed in Section 2.2.3. A short description is given at the beginning of each application review, and the important insights are summarised.

3.2.1 Existing BMW Review Application

The current VR system combines the *Gaze Pattern*, *Hand Selection Pattern* and gesture-based control *Widget and Panels Pattern*. Those interaction patterns are described in Section 2.5. The user interface is used entirely without a motion controller. A screenshot can be seen in Figure 3.2[3]. The *Leap Motion* device, which is described in Section 4.1.4, is used to recognise gestures. Once the inner palms are visible, this process automatically pops up a menu of flat 3D-widgets on one hand, based on which palm became first visible. Touch gestures are used to make a selection. Either a radial menu appears where various settings can be conducted, or a tool for the analysis process can be selected. The direction and angle of the HMD are used to make a selection on the radial menu.

Evaluation with Usability Heuristics

In this Section, the BMW Review Application Tool is analysed using the usability heuristics by Jakob Nielsen, who are described in Table 2.2 (p. 9). The two applied interaction techniques are split into a quick-menu, which is attached on a virtual hand and a radial menu, which has options for advanced settings.

1. *Visibility of system status*

   Both: No audio signal is detected as feedback in the system.

   Radial menu

   - The user receives feedback as soon as a tool or operation is selected with gazing in the form of a radial progress bar animation and a tooltip text beside. Selecting by gazing lets a menu item appear in the highlight of the radial menu.

[3] https://www.youtube.com/watch?v=9cW48GLx1nM

- The radial menu is always placed at the same distance in the user's field of view. Only a placement adjustment follows when the user turns sideways out of the field of view.
- The radial menu has inactive elements which are greyed out.

Hands menu: The activated elements are identifiable by another visual representation of the menu element when the manual menu is open.

2. *Match between system and the real world*

   Both: The provided tooltips speak the language of the users.

   Radial menu: This kind of menu uses an HMD Gazing technology as an input that allows menu items to be selected.

   Hands menu: The user performs actions with a hand selection pattern that allows hands to be represented realistically. 3D widgets are attached to the side of the hand. A touch movement on a menu item fulfils an action. Some tools are working with touch gestures.

3. *User control and freedom*

   Both: The user is able to close the menu entirely or take a step back in the hierarchy at any time. The application contains no undo and redo functionality.

   Radial menu: The radial menu automatically adjusts itself horizontally to the height of the HMD. This position stays stationary in the same place until the menu is taken out of the visible region by head movement. This action triggers an operation, where the radial menu is centred again in the field of view.

   Hands menu: The menu uses no stabilisers and is directly attached to the hand. The orientation of the menu items adapts to the user's viewing direction.

4. *Consistency and standards*

   Both: The developers use well-known interface metaphors for VR.

   Radial: The menus in the radial menu contain only icons. These icons can be well interpreted and remain constant across the hierarchy levels. Tooltips facilitate the accurate identification of menu items.

   Hands menu

   - Each virtual hand has four different menu elements attached. On the right side, there are tools, and on the left side, setting changes can be done. Only one menu can be opened at a time. However, it is unclear when exactly a menu will be opened and when it will disappear.
   - In the beginning, only the virtual hands are available. In order to open the radial menu, the user has to open the virtual hand menu first. Inside is a menu element contained that opens the radial menu.
   - In a short interval, actions are called in a loop performing a slow touch movement on an interactive element. This triggering behaviour is only present on this menu elements.

5. *Error prevention*

   Radial menu: The radial menu has inactive components containing a tooltip with information about why they are disabled.

Hands menu

- Short cuts are provided to the user in the form of 3D widgets, which are attached to the hand. These facilitate the selection of tools with the hand menu.
- Sometimes interactions trigger an action of two different menu elements at the same time. Furthermore, it was noticed that as soon as an action is deactivated, in some situations, it is immediately reactivated by mistake. There are no protection mechanisms provided.

6. *Recognition rather than recall*

   Radial menu: The operation of the radial menu is very challenging. Since the menu works very sensitively and quickly with gazing input, it is difficult to memorise the individual menu elements and their position in the hierarchy. A more extended learning period is necessary to get used to the system and to perform operations.

   Hand menu

   - The hand interactions with tools are managed intuitively after a low learning effort. Once these techniques have been learned, they should be retrievable shortly.
   - There is no visible logic for the arrangement of the menu items.

7. *Flexibility and efficiency of use*

   Radial menu: The menu structure can be extended as required in the hierarchy. Up to four levels are available in the application for settings. The findings revealed that no distinction was made between beginners and experts by the developers. The menu navigation inside of the radial menu is constant. Expectations are met when selecting menu items.

   Hand menu: The extensibility and flexibility are very limited. In this case, four elements can be attached to each hand.

8. *Aesthetic and minimalist design*

   Both: The menus have a minimalist design approach and are different in their presentation form.

   Radial menu: The menu requires much space in the virtual environment. The font is easy to read. The colour of the icons, timer and text is black. No emphasis is placed on graphics and text.

9. *Help users recognize, diagnose, and recover from errors*

   Both: The user does not receive an error message in case of action-related errors.

10. *Help and documentation*

    Both

    - The font is not legible on a light background (looking at the sky).
    - It can happen that the feedback text appears behind the menu after an action is performed, and the text is not readable.

Conclusion

The *BMW Review Application* has implemented intuitive interaction techniques. The gazing interaction technique is directed at users who already have VR experience and have a been briefed about the use of the menu. The second type of interaction, which allows interaction of virtual hands and objects in the VE, can be used for VR beginners as well. The user is placed outside of the vehicle at the beginning. The environment, as well as the vehicle, are visualised in real dimensions. Altogether, several features are provided to the user. Customisation of vehicle components is not included in the version to be tested. The next two Lists discuss the radial menu and virtual hand menu.

Radial Menu

- Working with the radial menu is a pure disaster. The gazing technology faces many challenges during the tests that emerge, especially when applied inside the vehicle. The menu elements are very large dimensioned. Therefore, this technology is only conditionally suitable in the driver's seat position. The problem is that in many situations, the menu is hidden by interior components of the vehicle. As a result, menu elements like the back button cannot be triggered. Convenient operations outside the vehicle with the gaze are possible because of the size of the elements.
- It is challenging to keep a clear overview of the menu and its structure, even if only a few elements are visible in the radial menu due to their large dimension. The menu must be closed for an evaluation process on the vehicle. Otherwise, the vehicle is hidden behind the user interface. Consequently, fast adjustments to the vehicle would not be possible in the design review process.
- The user must continually pay attention to the viewing direction as soon as the radial menu is opened. The automatic selection, which is timer-based, often triggers unwanted actions. This selection cannot be combined with high efficiency and is especially tricky for beginners, as they are dependent on the tooltip text.
- The tooltips elements are very large and take up plenty of space on the user interface. In some situations, the tooltips cannot be read when targeting the item in the radial menu. The reason for that is because the tooltip text is placed next to the menu item and is truncated. A head rotation causes a loss of focus on the menu item, and the tooltip is therefore no longer present. Besides, the tooltips contain information on why they are inactive. This information is permanently displayed even though the user has no way to fix the problem. These messages lengthen the size of the provided tooltips.
- Once an item is selected, the user must rotate the head to remove the view from the menu. Otherwise, the underlying menu element will be triggered. This circumstance is exhausting and burdensome with increasing duration, as the user has to pay additional attention to trigger something correctly. A wrong behaviour of the systems is that status messages appear behind an opened radial menu and show an extensive scaling of the text so that a head movement must be carried out to be able to read the text.
- The progress bar provides information on how long a menu item can be viewed until it triggers an action. In some cases, the progress bar is not visible due to conflicts with the user interface materials. This situation leads to unwanted, performed actions.

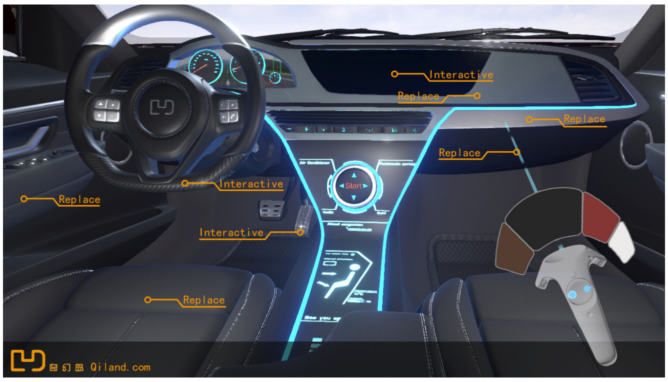


Figure 3.3: Screenshot of the application Automotive Customization Experience.

- The gazing technology offers a challenging introduction for VR beginners. This interactive system can only be operated with increased learning effort when used with complex menu structures. The gazing technology is very tiring for the eyes when used for a long time. Moreover, a head movement must be carried out for all actions to be performed. In the case of visual impairments, body posture must be changed to be able to re-establish visual contact.

Virtual Hand Menu

- The interactions using virtual hands are designed intuitively. After a little experimentation period, the menu options become visible to the user. The interaction by touching menu items is straightforward. The use of a realistic hand representation allows a fast immersion into the virtual environment.
- The general problem was that during testing, the selection of a menu element is challenging because the attached menu has no stabilisers. The result is a jittery menu. However, the alignment of the menu elements on the virtual hand works very well. The distances are well chosen that an element can be selected easily.
- An issue concerning ergonomic is that the hand must be rotated very far outside to open the virtual hand menu. When this user interface has been opened for an extended time, it does not seem comfortable and is not well designed ergonomically.

3.2.2 Automotive Customization Experience

The application *Automotive Customization Experience*[4] is specialized in changing the characteristics of a vehicle. Figure 3.3[5] shows a screenshot inside the car reflecting the interior. The user sits in the driver's seat position at the start. The main tool is a selection tool based on the *Pointing Pattern*. An object interaction is implemented as well using the *Hand Selection Pattern*. Both patterns are described in Section 2.5. The menu is connected to the motion controller and is assigned to the category *Object fixed* which is mentioned in Section 2.3.1. No second motion controller is used in this application.

[4] http://www.qiland.com
[5] https://www.viveport.com/apps/8216bfcc-db08-41c2-8d61-5d4756ad8d86

A ray is always visible, originating from the controller. As soon as an intersection of the ray with an interactive component of the vehicle occurs, the menu gets visible, and the menu structure automatically adapts to two object changing possibilities. Firstly, there is a trigger component that activates an event and plays, for example, a door animation. Secondly, different interior styles can be adapted to the users' needs. The menu system has only one level in the hierarchy. Thus only one property of an element can be changed.

Evaluation with Usability Heuristics

As in the previous Section, the application *Automotive Customization Experience* is analysed with the usability heuristics by Jakob Nielsen in Section 2.2.3:

1. *Visibility of system status*

   The user receives no system status in text form. The changes are immediately visible to the user and should serve as status recognition. The user does not need an additional menu status due to the simple hierarchy structure of the menu system.
2. *Match between system and the real world*

   The user interface reminds the user of a colour palette held with one hand when styles are selected. However, due to the one-handed operating concept, colour selection with a second controller is not possible. Changes to a style are made by selection via the trackpad.
3. *User control and freedom*

   The application contains no undo or redo functionality. This possibility would not be relevant in this small range of functions. The user does not need an exit option due to the hierarchy system of the menu.
4. *Consistency and standards*

   The developers use well-known interface metaphors for VR.
5. *Error prevention*

   The user does not receive any error messages for actions that are not performed.
6. *Recognition rather than recall*

   A ray from the motion controller is always activated and visible. As soon as an interaction with an interactive component occurs, a menu appears above the controller. Once a 3D widget appears, an event can be executed by clicking the trigger. The user automatically recognises the interaction technique when using the application.
7. *Flexibility and efficiency of use*

   The analysis demonstrates that the developers did not distinguish between beginners and experts. Interactions possibilities remain constant.
8. *Aesthetic and minimalist design*

   The developers are not using text on the interactive menu. The 3D widgets are displayed in the form of icons which provide information about actions.
9. *Help users recognize, diagnose, and recover from errors*

   No error messages appeared during testing.

10. *Help and documentation*
    In addition to the vehicle in the front field of view, a dialogue box with an overview of the functionalities is permanently available. The overall help in the form of instructions for interactions is displayed in an Asian foreign language.

Conclusion

The application *Automotive Customisation Experience* has integrated an intuitive operating concept that is particularly suitable for VR beginners with interests in the field of cars. The system uses two well-known interaction metaphors that are used in many VEs. The positioning of the avatar corresponds to the actual seating position in the vehicle. The view of the interior is only minimally restricted and corresponds to real space factors. Overall, only a few features are implemented. The application can be seen as a preview of a final version.

A positive aspect is the permanently active laser ray, which signals interaction possibilities right from the start (for beginners). Whenever an interaction possibility is available, the menu immediately adapts to the object attributes. An outline surrounds the selected interior elements. It is possible to identify the changing object in this way. As soon as an animation of an interior element is available, a floating symbol, which is pulsating in size, signals another interaction possibilities. A click on this object triggers the animation. The motion controller is wrapped in a transparent material that does not restrict the view of the interior. There is only one motion controller available. Therefore the interaction system works entirely one-handed, which avoids complicated interactions techniques.

Some negative aspects were encountered in the application: The entire interaction system is limited in terms of customisation. Only geometry groups (packages) can be edited in style when changing the material. Further adjustments are not possible, such as geometry changes. An efficient solution for displaying list elements in 3D space has been developed. The result is quite intuitive, but the user does not see how many elements are available in a list. The menu offers only one hierarchy level, and the elements themselves are arranged *1-DOF* style as described in Section 2.3.1. This selection technique is straightforward. A direct selection of colour is not possible, because only the next or previous element can be selected.

The automatic selection of interactive elements seems to make sense at the beginning. Operating the system for an extended period of time makes it more difficult to view the interior. The user should be careful not to select any components with the motion controller accidentally. The interaction techniques described in combination with the one-level menu would not be sufficient for productive use in the design and review process at BMW. Additional features seem to be difficult to integrate and extend because of the hierarchy structure.

3.2.3 Relay Cars

The application is intended for users who want to experience different vehicle models within a VR. The user is located in a showroom with different vehicles. The screenshot

Figure 3.4: Screenshot of the application Relay Cars.

in Figure 3.4[6] shows the integrated menu system. The description of the application is available on the developer page [43]:

> "*RelayCars®* is the Virtual Reality automotive experience that puts you in the driver's seat of hundreds of vehicles. You have the power to choose your vehicle category, year, make, and model from our extensive VR library. Explore our vehicles inside and out in full 360° stereoscopic 3D. Our detailed showroom experiences feature vehicles ranging from exotics to hybrids, to sedans and full-size trucks."

The use of both motion controllers enables the user to travel to different locations with the *Teleport Metaphor*. At each position, a different car is presented. In the immediate surroundings of every vehicle is an interactive console that allows user interaction using the *Grasping Metaphor* which follows the principles of the *Hand Selection pattern*. Those patterns are described in Section 2.4.5. The menu system belongs the category *World-Fixed* is mentioned in Section 2.3.1. The controller representation is visible and has a similar design to the original HTC Vive motion controller.

The user interface contains virtual buttons for customisation. Five buttons are available in the current version of the time of writing. The menu structure is designed to allow a maximum of two hierarchy levels. The user takes an interactive sphere from the available colour palette and throws it directly onto the car to change the colour. This principle also applies to the selection of another car or vehicle model. Users who wish to place themselves inside the vehicle must perform the same step using a different interactive menu item. The user is automatically ported into the driver's position when activated via a control key assignment. As long as the button is pressed, the user remains in the vehicle and can view the interior at a close distance.

[6] https://store.steampowered.com/app/994610/RelayCars

Evaluation with Usability Heuristics

This evaluation is the last in this chapter. Once again, the *10 Usability Heuristics for user interface design by Jakob Nielsen* is used to evaluate the application regarding usability and can be found in Section 2.2.3:

1. *Visibility of system status*
    - The system status is not signalised. The user starts in the middle of the room and has no help at all, only spotlights that illuminate the control panel signal a possible possibility of interaction. For the first time, the user must experiment and move to the control panel to obtain additional information on how to perform further interactions.
    - The new interactive components adapt to the selected menu item after pressing a button. However, it requires an interpretation of the available interaction possibilities. Users have to read the help text to know which level of the menu they are currently in.
    - The user has a highlight effect and haptic feedback when interacting with a button, which signals the possibility of interaction. All other elements miss the haptic feedback feature.
2. *Match between system and the real world*

    The control panel signals interaction possibilities via buttons, like we often find in real life, for example, when we look at a remote control. The texts of interactive buttons such as buttons and 3D widgets are in a simple language. The circular widgets for selecting the models contain a preview image of the object to be selected. The corresponding material is applied directly to the 3D widget when choosing the colour.
3. *User control and freedom*

    The application contains no undo or redo functionality. This possibility would not be relevant in this small range of functions. The user is presented with an additional button element for an exit option as soon as he leaves the first hierarchy level.
4. *Consistency and standards*

    The developers use well-known interface metaphors for VR.
5. *Error prevention*

    The user does not receive an error message if an action is not performed. If a thrown ball does not hit the vehicle, it disappears into the ground and returns to its original location in the menu. No message is displayed.
6. *Recognition rather than recall*

    The instructions for executing an interaction are displayed after pressing a button on the control panel. The interaction concept for vehicle changes is always based on the same interaction concept. An interactive element is thrown onto the car and triggers customisation.
7. *Flexibility and efficiency of use*

    Results of the analysis have shown that the application does not distinguish between beginners and experts. The action possibilities remain constant.

8. *Aesthetic and minimalist design*
   The text length of text information is limited due to size restrictions.
9. *Help users recognize, diagnose, and recover from errors*
   No error messages appeared during testing.
10. *Help and documentation*
    The help text and instruction for interactions are in a simple language and do not use technical terms. The information describes the relevant aspects in a compact form. A help button allows the display of a dialogue box describing all interaction possibilities with the motion controller.

Conclusion

The application *Relay Cars* has an intuitively understandable interaction concept, and the target group are VR beginners with interests in the field of cars. Well-known interaction metaphors are used to facilitate the interactions from the very start in this system. The avatar is located outside the vehicle at the beginning. Overall, the application has only a small number of features. The application itself is specialised in the visualisation and customisation of different vehicles. An adjustment of the interior is not possible.

The interactive system is wholly developed on an interaction concept based on 3D widgets. Virtual keys are pressed on a control panel with the two motion controllers, allowing for further different vehicle settings. A straightforward user interface offers the user interaction options, and a customized help text is available at all times. The user receives feedback in the form of an outline around the selected object, and the controller starts vibrating when a controller-button intersection occurs. The customisation is based on a familiar form of interaction in which an interactive element is thrown onto the car. Therefore, the user is able to change the colour and the model of a selected car on a control panel.

The system has some disadvantages for the use in the VR design and review process in the automotive industry. The user can only adjust the exterior of the vehicle and needs to be close to the control panel for making adjustments. The control panel is located far away from the vehicle. Throwing requires an enormous amount of effort to ensure that the thrown element reaches the vehicle. The current version of the control panel itself is well designed and allows two hierarchy levels. The control panel can quickly become confusing if further adjustments and features are required. The avatar must be teleported to vehicles located in the VE to make customisations to the car. The user only receives a 360° stereoscopic image to view the interior. Therefore customisation of the interior is not possible. The stereoscopic image is scaled too high, resulting in a too oversized interior. Besides, various loading processes between vehicle changes blur the user experience.

The concept of customising the vehicle outside the vehicle is undoubtedly interesting if only the exterior of the vehicle has to be evaluated. However, it is essential for the design and review process that the user has access to the interior as well. The last point to be mentioned is that the fast adaptation of elements is difficult because the control panel must always be within reach of the arm.

Chapter 4

Implementation

This chapter describes the tools used in the practical part of this thesis. It provides information about the game engine, the used 3D modelling software and the VR toolset. Subsequently, the planning process is followed by a description of the concept model and the interaction concept. The final part deals with the interactive system, which contains the menu structure, folder structure and the input systems. Besides, this Section focuses on the design approach, the integration of the system into a project and the different feedback methods of the interactive system.

4.1 Tools

The first part provides information about the selected game engine and the used 3D modelling software. Afterwards, details about the used VR set are given and conclude with the finger tracking sensor system.

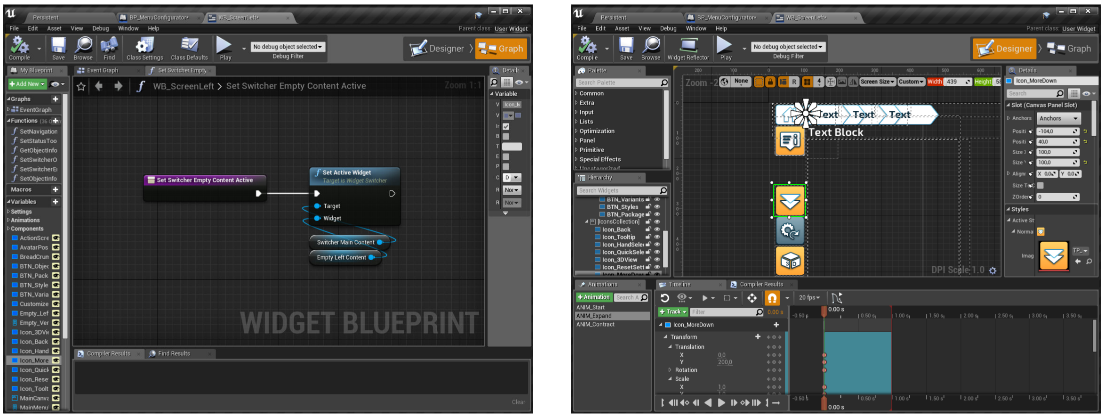
4.1.1 Unreal Engine

The *Unreal Engine*[1] was released by *Epic Games*[2] in 1998. It offers a solution for creating games, simulation, training and industrial applications. The system can be enhanced with additional plug-ins to achieve specific requirements. The *Unreal Engine* allows creating high-quality, photorealistic and immersive augmented reality and VR experiences for architecture, product design and manufacturing.

Concept of Blueprints

The game engine offers two possibilities for adding logic to a project. The adding can be done with the traditional programming language C++ or with a visual scripting language. *Blueprints* are a node-based system where box-like objects are linked with variables, functions and flow elements similar to a flow diagram. A screenshot is shown in Figure 4.1(a). It enables non-programmers such as artists and designers to apply changes quickly without consulting a developer. Overall, the visual processing offers a

[1] https://www.unrealengine.com
[2] https://www.epicgames.com



(a) (b)

Figure 4.1: The concept of Blueprints in Unreal Engine and the graphical user interface of Unreal Motion Graphic. *Screenshot of Blueprints* (a), *Screenshot of Unreal Motion Graphics* (b).

clear representation of functionalities and variables [24, sec. 1.6.5]. A specific feature of the *Blueprint* concept is the easy way to migrate an existing project to another one. This feature is helpful when working simultaneously on different projects which need the same functionalities. A negative aspect concerns clarity. It can quickly become confusing if a project increases in functionality.

Unreal Motion Graphics

Unreal Motion Graphics (UMG) is a proprietary solution for the *Unreal Engine* to design graphical 2D user interfaces. A graphical editor is used to create hierarchical menus, user interfaces, head-up displays in appearance and their functionalities. Graphical elements called *Widgets* which can keep a fixed position on the screen (overlay) or 3D space [29, sec. 6.1.2]. A screenshot of the UMG can be seen in Figure 4.1(b).

4.1.2 Autodesk Maya

The software *Maya*[3] from *Autodesk*[4], is a 3D computer graphics software and is used for the creation of applications, games, movies and visual effects. It offers functions for 3D animation, modelling, simulation and has plugins for rendering.

4.1.3 HTC Vive / HTC Vive Pro

At the time of writing, *HTC Vive*[5] had established its position in the consumer VR market for high-end equipment. This product is a complete VR system which requires

[3] https://www.autodesk.com/products/maya/overview
[4] https://www.autodesk.com
[5] https://www.vive.com/us/product/vive-virtual-reality-system

Figure 4.2: The input options of the HTC Vive motion controller.

no additional hardware. One set consists of an HMD, two motion controllers and two tracking stations called *Lighthouses*.

The first model of the *HTC Vive* has a resolution of 1080×1200 pixels per eye since its market launch in 2016, which corresponds to a total resolution of 2160×1200 pixels for both eyes. Its newest model, the *HTC Vive Pro*, was launched in 2018 and has a total resolution of 2880 x 1660 pixels for both eyes (1440×1600 pixels per eye). The field of view (110°) is the same for both models. Hence, the new model offers a sharper image and a reduced screen-door-effect.

The *Lighthouse* tracker, has two different versions. They differ in simultaneously usable computers with the same trackers and from the 360° room-scale body tracking. The latest version offers a bigger interactive area with a maximal dimension of $7m \times 7m$ and allows unlimited use of trackers. The older model has an interactive area of $3.5m \times 3.5m$ and allows maximal two trackers. Body tracking allows the user to move freely within a specific room size. These movements are transferred into virtual space.

The trackers allow interactions via motion controllers. The motion controllers must have a direct visual connection with at least one of the trackers to be recognized by the VR system. The controller has several individual inputs. The trackpad can be operated by touching and pressing. The trigger on the rear and two grip buttons on the side are used as inputs as well. These elements are shown in Figure 4.2.

4.1.4 Leap Motion

The *Leap Motion*[6] technology makes it possible to track and visualize the movement of hands and fingers in the VE. Gestures and physical interactions open up further interaction possibilities. The *Leap Motion* is installed on the head-mounted display and works with a field of view of 135 degrees. The interactions must take place within the action radius of 60cm. The hands are automatically tracked and can be used for interactions and manipulations within this range.

[6] https://www.leapmotion.com

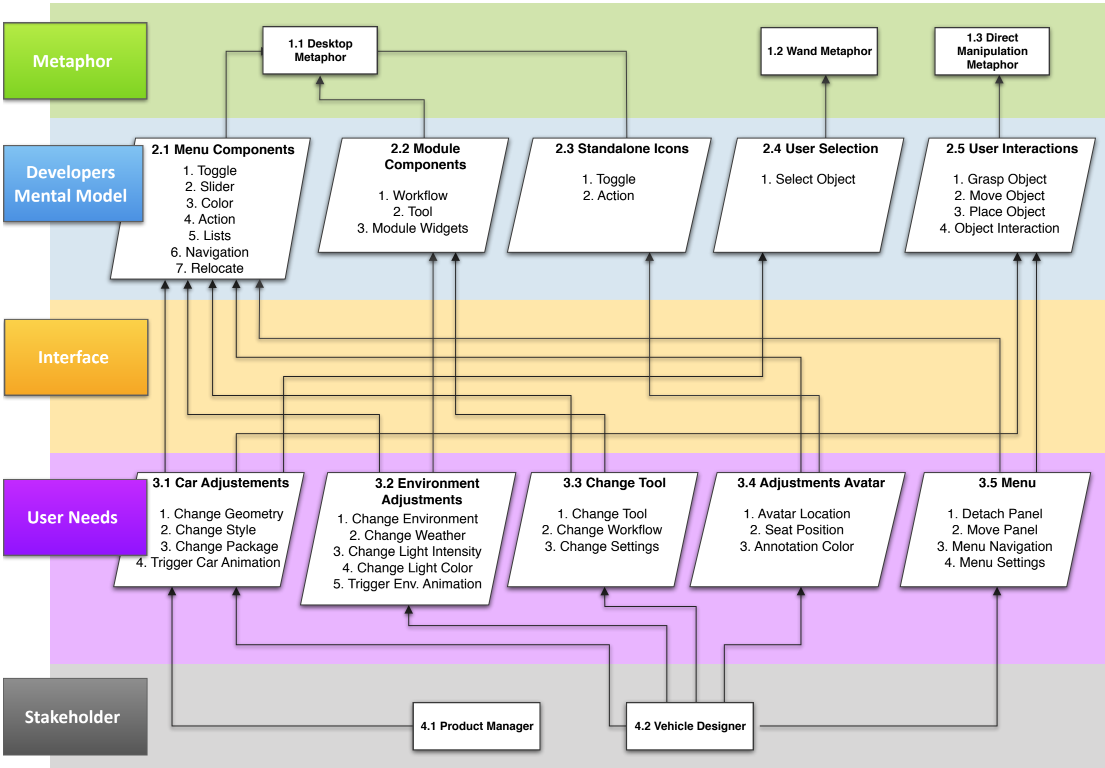


Figure 4.3: The conceptual model of the interactive system.

4.2 Planning Process

This Section describes the structure and operation of the system using *Conceptual Models*, which are discussed in Section 2.4.3. The flowchart in Figure 4.3 provides an overview of the *Conceptual Model*, which is discussed in more detail. Metaphors and concepts are listed which are available to the user. The relationships between the objects and actions are displayed afterwards. The conclusion deals with the mapping between the concepts and the performed tasks within the system.

Metaphors

The *Desktop Metaphor* describes interactions with an input device and a 2D graphical user interface. The users know this metaphor from working with computers most probably. The menu navigation is well known because similar mechanisms are used to navigate through a menu or to select elements. This operation is usually done with a mouse on desktop devices. In VR, a pointing device is used, which emits a ray from the motion controller, as described in Section 2.4.5.

The *Wall Metaphor* refers to a type of selection where objects are chosen reminding of magical interactions with a wand like *Harry Potter*[7]. This way of selecting objects is used in many applications to perform an action. Similarly, the context of a menu can be

[7] https://www.pottermore.com/explore-the-story/harry-potter

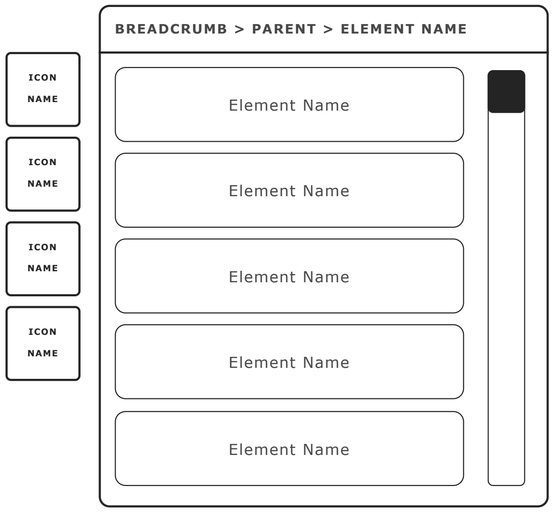


Figure 4.4: Abstract representation of the user interface.

adapted to the object context. This procedure allows an efficient and direct selection of an object without the need to search for the object in an item management list. The *Direct Manipulation Metaphor* is a form of interaction in VR that is often perceived as a natural and realistic type of interaction.

Interaction Concept

The defined concepts are decomposed in detail in this Section. Subdivisions show all interaction elements of the menu available to the user. Table 4.1 lists the executable actions with the corresponding attributes and the interaction options. An abstract design of the main user interface is available in Figure 4.4. A full expanded abstract user interface can be found in Appendix C.1 (p. 81).

- *Menu Components* describes all interactive components visible and available to the user within the main menu (elements like Toggle, Colour and for Navigation). Those elements are available in Appendix C.4 (p. 86).
- *Quick Selection* is a developed concept that allows the user to select tools grouped by workflows quickly separated from the main menu. The second user interface is called *Module* and offers menu items for fast selection. This user interface is always visible to the user and can be customized. According to the stakeholder, up to four Modules are visible. An abstract representation of a *Module* can be found in Appendix C.1 (p. 81).
- The *Stand-alone Icons* are 3D-widgets, which are indirectly connected to the user menu, or they can be found floating at a specific position in the VE (door animation icon, seat position icon).
- *Smart Selection* contains a concept to give the user a context-based menu when selecting an object with the laser pointer in the VE. Further modifications can be made afterwards to a geometry quickly.
- *Object Interactions* deal with features that result from a detached menu. A detached

menu from the main user interface can be positioned anywhere in the VE to make changes on the vehicle in the future quickly. The user interface allows detaching geometry from the vehicle as well.

The user interface is divided into two areas to handle the complexity. The main menu relates to changes in the vehicle, environment or menu adjustments. The second user interface (Quick Selection) deals with the analysis process that takes place in virtual product development. The visual result can be seen in Figure 4.10. Beginners and Product Managers don't face the second user interface. This separation allows beginners or similar user groups not to get overwhelmed with an oversupply of features of a system that does not meet their needs. The complexity of the shareholders can be adjusted with this user-defined setting.

Relationships and Mapping

All concepts used in the interactive system have the goal to meet user expectations. Therefore it is up to the user which interaction concept is chosen at a given moment. Different components of a vehicle are changed in geometry, material properties or the overall package. These physical components of a vehicle have and equivalent in the list and can be found in the menu. Objects such as a light source or animation trigger elements are also included in menu items if it is necessary to change properties in the environment. Object selections and interactions can be mapped to the actual component in the VE.

4.3 Architecture

This Section deals with the process of choosing the menu structure. An overview of the directory structure of the project follows afterwards. The next part deals with the input methods of the system: a virtual hand menu and an input via motion controllers. The menu elements used in the user interface are explained how the feedback system is integrated using which methods. Finally, the description is given on how the developed system can be implemented in an existing project.

4.3.1 Menu Structure

This Section examines the menu structure of the interactive system. As mentioned in Section 3.1.2, it is a flexible system that adapts to the needs of a specific stakeholder. Figure 4.5 shows the primary hierarchy levels of a stakeholder with maximum functionality.

The analysis of several interactive systems in Chapter 3 has shown that the use of a menu on a fixed object close to the body has proven successful. The menu can be a constant companion independent of the location. Fixed menus in VE are not suitable, because different positions inside and outside the vehicle have to be taken. For example, the functional scope is limited if only tools with virtual hands are used for interaction. That is why the use of controllers is indispensable. These controllers can be equipped with object-fixed user interfaces, which provide a stable display of a menu.

Up to five main menu elements are available for the stakeholders, in which different menu components are available. A complete list of those elements is presented in Table

Table 4.1: Decomposed interaction concept of the interactive system.

No.	Object/Interaction	No.	Action	Attributes
2.1 Menu Components				
2.1.1	Toggle	3.5.4	Grid	On / Off
		3.4.1	Seat Positions	Function Call
		3.2.5	Environment Animation	Call Function
2.1.2	Slider	3.3.3	Light Intensity	Decimal Value
2.1.3	Colour	3.3.4	Light Colour	RGB Value
		3.4.3	Annotation Colour	RGB Value
2.1.4	Action (Menu)	3.1.4	Trigger Car Animation	Function Call
		3.2.5	Trigger Env. Animation	Function Call
		3.3.1	Change Tool	Function Call
		3.3.1	Change Workflow	Function Call
2.1.5	List-View	3.1.1	Change Geometry	Array[Geometry]
		3.1.2	Change Style	Array[Styles]
		3.1.3	Change Package	Array[Package]
		3.2.1	Change Environment	Array[Environment]
		3.2.2	Change Weather	Array[Weather]
			Manipulate Screenshot	Array[Screenshot]
2.1.6	Navigation	3.5.3	Choose Menu-Item	Function Call
2.1.7	Relocate	3.4.1	Avatar Location	Function Call
2.2 Quick Selection				
2.2.1	Workflow	3.3.1	Change Workflow	Function Call
2.2.2	Tool	3.3.2	Change Tool	Function Call
2.2.3	Module Widgets	3.3.3	Change Settings	Change (2.1) Comp.
2.3 Stand-alone Icons				
2.3.1	Toggle	3.4.1	Seat Positions	Function Call
2.3.2	Action (Icon)	3.2.5	Environment Animation	Trigger Event
2.4 Smart Selection				
2.4.1	Select Object	3.1.1	Change Geometry	Array[Geometry]
		3.1.2	Change Style	Array[Style]
		3.1.3	Change Package	Array[Package]
2.5 Object Interactions				
2.5.1	Grasp Object	3.5.1	Detach Panel	Menu Interaction
2.5.2	Move Object	3.5.2	Move Panel	Menu Interaction
2.5.3	Place Object	3.5.2	Connect Panel with Grid	Menu Interaction
2.5.4	Object Interaction	3.1.1	Change Geometry	Button Interaction
		3.1.2	Change Style	Button Interaction

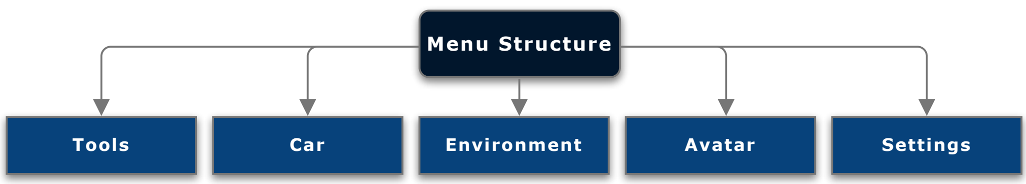


Figure 4.5: The menu structure in the main level, which provides a stakeholder and full functionality.

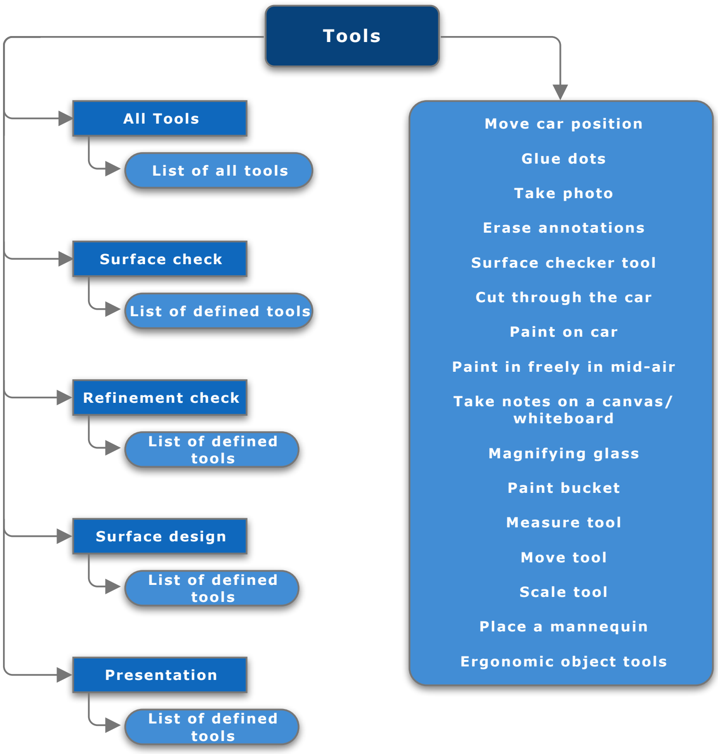


Figure 4.6: An overview of the tools hierarchy with all components.

4.1. The arrangement of these elements is based on their relevance to the product development process. The order is examined iteratively and adapted to the user's needs.

- The first menu item contains *Tools*, where workflows group tools for the car analysis. The tools can be called in sequence inside of a workflow when this feature is required. The whole list of all available analytics tools can be found in Figure 4.6. At the time of writing six tools were available for the testing purposes. Appendix C.2 (p. 82) provides a detailed overview of the contents of the remaining submenu items.
- The *Car* menu item provides menu elements that allow changes in the appearance of the car (geometry, styles or packages).
- The *Environment* menu includes operations that can modify the environment and adjust specific parameters.
- The *Avatar* contains actions which can be applied to the avatar. This includes repositioning, rotations and takes a specific seat position. In case a network with multiplayer functionality is desired, the *Avatar* should be in future customisable with a name and a colour code.
- Different *Settings* of the interactive system are located in the last menu item.

The interactive menu consists of two fixed menu components on the left motion controller. The right one has only *Modules*, to decrease complexity. As soon as the menu is opened

Blueprints	Configurator	MenuConfigurator, VRStateManagement
	Controller	Console, RotateMesh, ViveController, ViveMenuController, CameraController, PlayerController, gameInstance, VRPawn, ObjectHider, ObjectChanger, TransitionManager, TagManager
	Interactive Elements	Grid, SpawnedPhysicsActor, Button, Triggers
	Menu	UIRotateIcons, UIModuleLeap, UIModules, UIScreens, UIWorldPreview, Widgets, UserWidgets
Common	Enums	ActionType, AvatarPosition, Button, ElementType, Hand, ModuleActionType, PreviewChangeType, SeatPosition, ToastAction, Tools, TouchpadDirections, UserRoles, WidgetAction, Workflows
	Functions	UserInterface
	Interfaces	Interactions, LeapMotion, Menu, Module
	Structs	Animation, Contrller, Menu, Modules, ObjectInformation, Theme, toast, Tools, Touchpad
Misc	Effects	Disolve, LaserBeam, Qutline, Snow
	Feedback	Haptic, Sound
	Materials	Objects, ParticleSystems, Text
	Meshes	Button, Controller, Interactive elements, Leap Motion, User Interface
	Textures	Materials
	Theme	Textures, Fonts

Figure 4.7: The folder structure of the interactive system.

and activated, the user operates in the main panel. Is the *Smart Selection* invoked, and an object is selected, the main menu changes context-based. In this case, the user has a limited choice of functions for changing the object. A seamless transition is created between these two components without the user having to worry about navigation. The navigation options are identical in both user interfaces.

4.3.2 Folder Structure

The Figure 4.7 shows the folder structure of the main directories in the first column of the (VR/) folder. It contains a separation of *Blueprints* (VR/Blueprints/), the commonly needed files (VR/Common/) and the miscellaneous files (VR/Misc/). In the *Blueprints* are further subfolders, which can be seen in the second column.

An essential settings directory (Configurator/) is placed inside the /*Blueprints/*) folder and contains the *MenuConfigurator* which is responsible for the dynamic creation of the menu structure. In the same directory is the *VRStateMangement*, which holds the user actions and all necessary variables of the menu state. In this area, the stakeholder or the menu structure can be changed. A tutorial is available in Appendix D.2 (p. 88). Next, all controllers which are required by the interactive system are included in the (Controller/). The subfolder (Interactive Elements/) contains objects such as 3D widgets for door animations or changing seat position. The last folder (Menu/) contains all widget *Blueprints* that can be edited with the UMG-editor.

The subfolder (Common/) contains all relevant structs (Structs/), enumerations (Enums/) and interfaces (Interfaces/) which are accessed by the *Blueprints*. These are shown in the second column. The folder (Functions/) includes a *Blueprint* library which contains global methods. A detailed overview of the enumerations and structs can be found in Appendix C.3 (p. 84).

The directory (Misc/) contains different content types, which can be seen in the second column. The (Effects) folder comprises particle systems and materials. A folder named (Feedback/) includes audio and envelopes for haptic of the motion controllers. All materials of the interactive system are contained in the folder (Materials). The 3D objects can be found in the folder (Meshes/). The textures have their folder in (Textures/). The last folder (Themes/) contains all graphics contents and fonts used in the UMG-editor.

4.3.3 System Integration

The interactive system is designed in such a way that the player pawn must be selected as the main pawn in the *Unreal Engine* settings. The entire folder, which is described in Section 4.3.2 must be included in a BMW related project. Afterwards, the user has access to the menu via the motion controller or the *Leap Motion*.

In the *Blueprint MenuConfigurator* all customization options of the user interface are centralized. The initial stakeholder type for the current product status must be selected manually in the *Blueprint* before starting the application. The stakeholder type is located in the *VRStateManagement*. Currently, four different stakeholders are defined (Beginner, Product Manager, Vehicle Designer, Max Features). The menu can be modified and updated during runtime. The shortcuts are listed in Appendix D.1 (p. 88). All maps (environments) need an additional post-process material for the outline effect. In the array section of the *Post Process Volume* the material *PP_MI_Outliner* should be added. The last step is to add two collision channels to the settings in the collision tab. The names to be added are *Interactable* and *ObjectChanger*.

A menu construction kit is available for customization, in which *Blueprint* functions with clearly defined method names can be selected. Besides, no further settings are necessary for the system to start. Appendix D (p. 88) contains help guides to the topics: *Change Stakeholder*, *Add New Workflow*, *Add New Tool*, *Add a New Component* and *Extend Menu Component Functionality*.

4.4 Interactive System

This Section provides an overview of the design approach and the visual connection of the menu items. The interactive system allows input via virtual hands or motion controllers. How to deal with the complexity of the system is a dedicated part to illustrate the importance of separation and order. Several feedback systems are integrated to support the users in their work. They have a functional description supported with screenshots in the following paragraphs. System integration into existing projects concludes this Section.

4.4.1 Design Approach

Wireframe-layouts were created during the elaboration of the interaction design to equip the interactive system with features. The first tryout how the user interface could be structured can be seen in Fig. 4.8 (a). In the beginning, a simple graphical menu was functionally implemented with the UMG to test the existing functions in VR, which is displayed in Fig. 4.8 (b). Different design mockups had been created to test design ideas with different backgrounds, which is visualized in Fig. 4.8 (c). The final design is available in Fig. 4.8 (d).

During the development of the user interface, the *Principles of Gestalt Psychology* [22] were considered. Lisa Graham [13] provides information in the article *Gestalt Theory in Interactive Media* for this special topic. The focus of this article refers to the five laws of Gestalt:

- The first law deals with the fundamental *Relationship* between the figure and the background. The legibility is reinforced with an applied contrast.
- The law of *Proximity* describes elements that are close together and form a group. Careful use can ensure a well-arranged visual appearance.
- The law of *Closure* refers to the ability of people to try to fill gaps in familiar shapes naturally. Especially when something is missing in design, attention is paid to the existing elements and the missing parts are replaced. In this way, lines, circles or other patterns can complement a shape.
- The law of *Similarity* describes elements having the same shape, size, colour, proximity and direction as a group. The same applies when a greater distance is available.
- The last law *Continuation* deals with the relationship between lines and shapes. An eye follows a line, curve, or part of a shape, independently of an intersection. It always follows the simplest path. The user's attraction can be guided by static or moving lines in interactive media disciplines.

4.4.2 Input System

Two input systems are in use in the developed application: Virtual hands and motion controllers. The selection of the respective input follows automatically. The virtual hands are used if the tracking system detects no motion controllers and the *Leap Motion* sensor is active. As soon as a motion controller is enabled and detected, the virtual hand gets replaced. The interactive system is designed in a way to allow virtual hands to interact with the controller menu, which can be seen in Fig. 4.9 (c). In this example, virtual hands are interacting with the motion controller menu to choose a package of the car. The inputs of the virtual hands and the motion controller are explained in the following two Sections.

- *Virtual Hands:* Virtual hands allow an easy and natural way to interact with the environment. In the same way, when operating a tablet or mobile phone, an action is executed by touching a menu item or a geometry in the VE. An example where the exterior colour of the car is changed can be seen in Fig. 4.9 (a). Selected geometries can be equipped with functionality. A 3D widget menu appears if the user touches such a geometry. This widget menu contains 3D buttons, which can be pressed to trigger an action, as shown in Fig. 4.9 (b). At the index fingertip is a spherical collider, which is responsible for the recognition of interactive elements. As soon as a defined distance to an interactive element is exceeded, the possibility of interaction with 3D widgets gets active. Fig. 4.9 (d) contains the expanded menu, where the user is changing the exterior colour with the laser pointer.

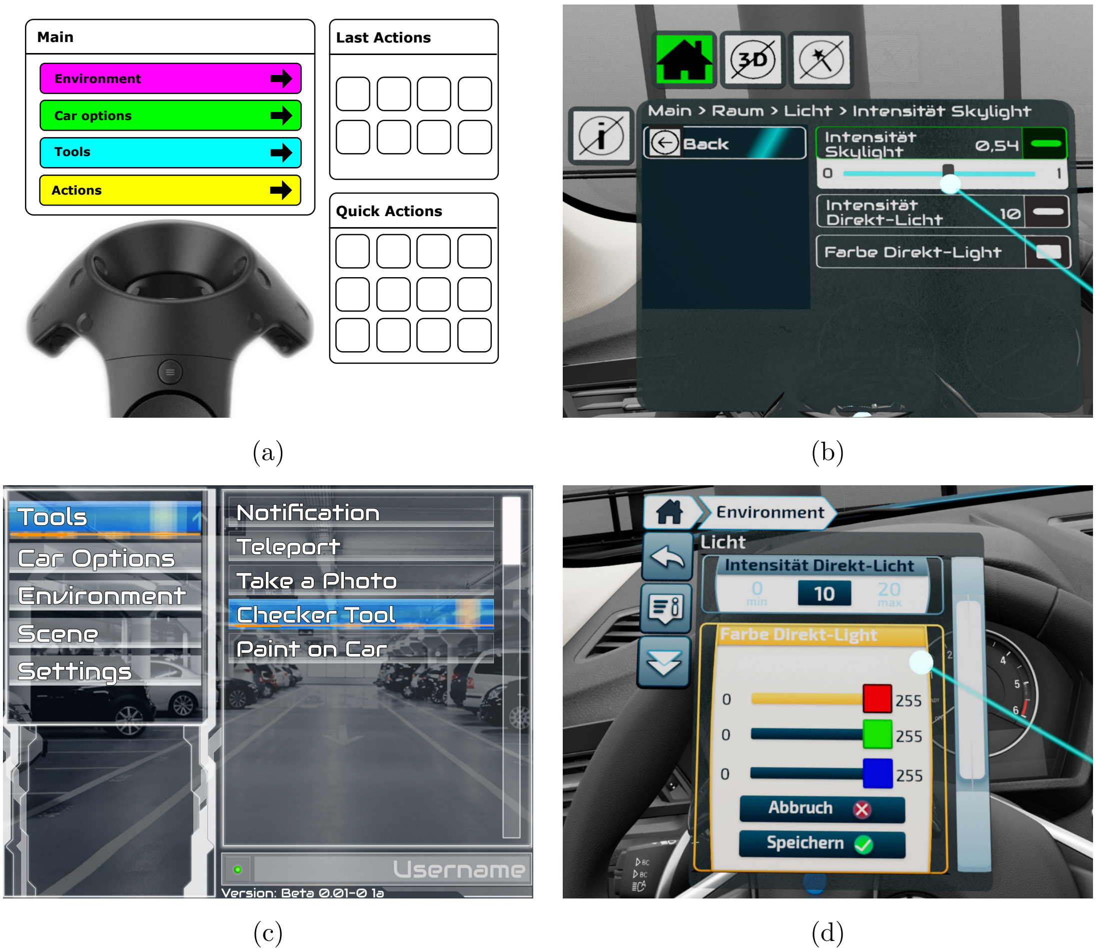


Figure 4.8: From the beginning of a design concept to the final user interface design. *Wireframe Prototype* (a), *Functional Design of the first Prototype* (b), *Prototype Mockup* (c), *Final Design of the Interactive System* (d).

- *Motion Controller:* The motion controller provides the user with two possibilities to navigate in the menu: Trackpad and laser pointer interactions. The user interface is designed in such a way that it can be operated entirely one-handed.

  The trigger confirms changes inside of a menu component like a slider component when using the trackpad solution. The user has the choice of whether to press or touch the trackpad. Furthermore, the action possibilities are shown at any time. There are directional arrows with top/right/bottom/left positioned above the trackpad. These arrows are invisible when no action is available.

  Figure in 4.2 shows the input technique using a laser pointer. The laser ray appears automatically as soon as an interaction possibility with an element is detected. This concept affects the user interface as well as the geometry of the vehicle. The trigger button can be touched or pressed for executing an action. The grab buttons are used for grabbing and picking up 3D objects. These interaction buttons are available in Figure 4.2.

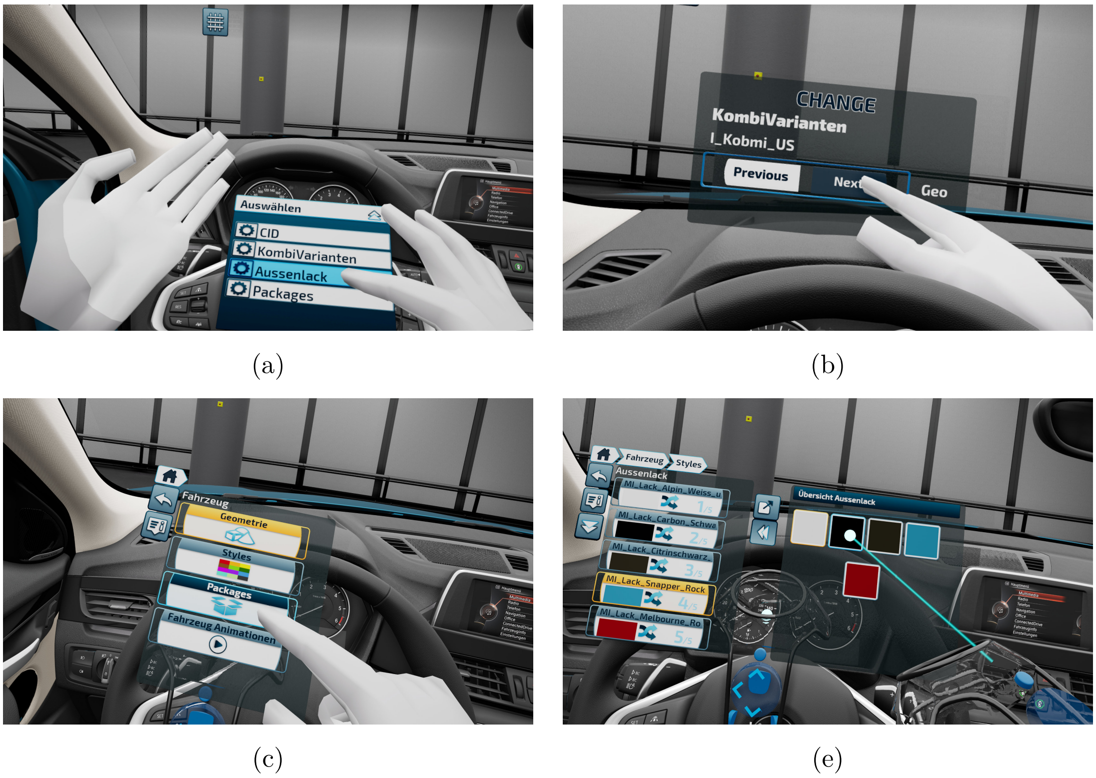
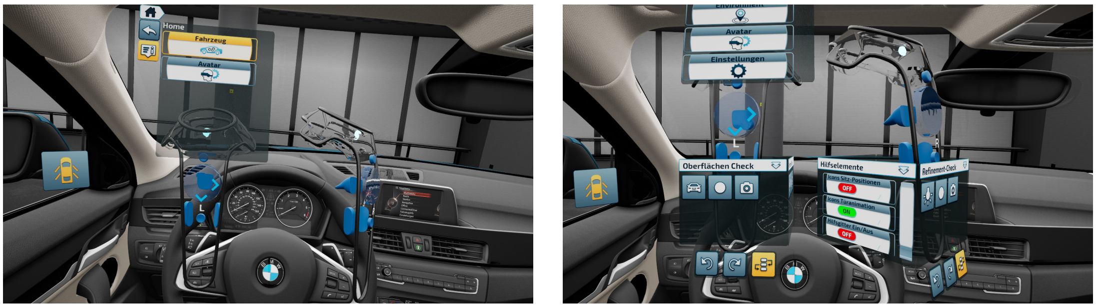


Figure 4.9: Different input methods with virtual hands and motion controllers. *Leap Motion Interaction with Hand Menu* (a), *Leap Motion Interaction with 3D Widget Menu* (b), *Leap Motion and Motion Controller with Hand Menu* (c), *Motion Controller Menu Interaction* (d).

4.4.3 Feedback System

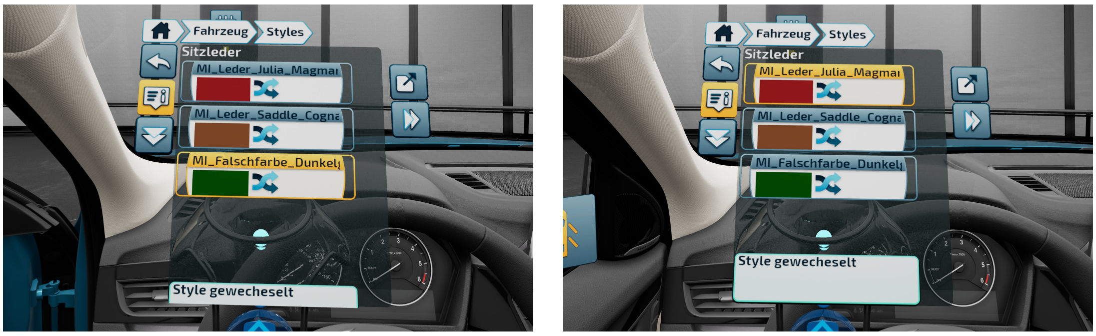
This Section describes the feedback methods used in the interactive system. Users have an optimal overview of their status and interaction options through the integration of these methods:

- *Toast System:* The user interface is equipped with a toast system to monitor system changes visually. Information on state changes appear in the menu panel at the bottom, which can be seen in Fig. 4.11 (a). A translation of the toast is performed starting below the bottom edge. The translation ends when the entire message-box appears, as shown in Fig. 4.11 (b). After a predefined time, the object disappears from the menu panel.
- *Haptic Feedback:* Different haptic levels of feedback are integrated, differing in pulsation intervals and vibration strength. Frequent recurring changes will result in only a small amount of vibration, such as a hover action through a menu item. The strength has been adjusted to be noticeable, but not intrusive. Unlike an object selection, it reproduces a high vibration in the form of a double push wave to direct the view of the user to the motion controller. An overview of available haptic feedback is in Table 4.2.



(a) (b)

Figure 4.10: Different available features for the stakeholders. *Menu system for product designer* (a), *Extended menu system for vehicle designer* (b).



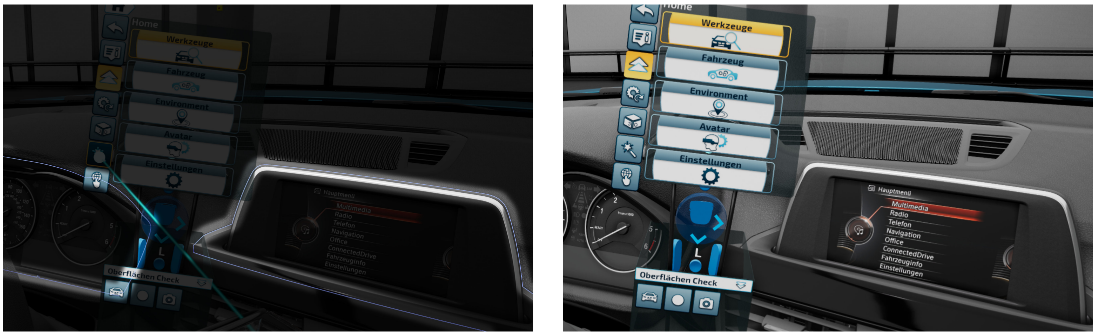
(a) (b)

Figure 4.11: The Toast System displays messages a state of the environment or the car changes. *Toast System Beginning* (a), *Toast System Full Presentation* (b).

Table 4.2: The available haptic feedbacks in the interactive system are presented.

Name	Duration	Character	Occurrence
Action Feedback	short	strong	one time
Hand Attention	long	aggressive	2 pulse waves
Hover Feedback	short	light,	one time
Select Feedback	between short and long	between strong and light	one time

- *Visual Feedback:* Additional signifiers are presented to the user after a symbol state next to the menu elements has changed. These indicators notify the user about new possibilities for interaction. For example, a 3D object is displayed for a predefined duration when selecting 3D views in the menu. As soon as the *Smart-Selection* is activated, all interactive 3D objects receive an outline. This behaviour indicates a possibility to interact with these objects, as shown in Fig. 4.12 (a). The outline disappears after a predefined time, and the initial condition is restored, which can be seen in Fig. 4.12 (b). If the symbol with the hand icon for direct manipulation is selected, an additional asset is displayed to the user on the controller. This asset



(a) (b)

Figure 4.12: Provide additional feedback to the user when a object is selected. *Visual Feedback Activated Look* (a), *Visual Feedback Normal Look* (b).

(a) (b)

Figure 4.13: The user has additional interaction possibilities when the Hand Selection is active. *Interaction Expansion with Enabled Plasma Ball* (a), *Interaction Expansion Active Plasma Ball* (b).

is described in the next paragraph.

- *Interaction expansion with controller:* As soon as the direct manipulation via an icon next to the menu is active, the controller is extended with a specific asset. The purpose is to remind the user of an active functionality. Before interacting with the motion controller, the user only sees an inactive plasma ball which can be seen in Fig. 4.13 (a). The plasma ball becomes immediately active as the user hits an area with interaction possibilities. A screenshot is available in Fig. 4.13 (b).
- *Tooltip:* Tooltips fulfil two purposes in this interactive system. First, they provide text information for purely icon-based elements. Second, menu items that may have an extended caption are displayed in their entirety. With this solution, the menu remains clear despite long menu entries. A screenshot is available in Fig. 4.14 (a).
- *Screen Transitions:* Screen transitions are used if environmental, or avatar properties are changed like changing the seat position. This effect should prevent the use of unwanted follow-up phenomena, such as motion sickness.

(a) (b)

Figure 4.14: Help elements like tooltips and breadcrumb to provide a better overview of the current state. *Tooltip* (a), *Breadcrumb* (b).

- *Audio:* Audio feedback is not used when working with motion controllers. Several people can be involved in VR design and review meetings. The number of audio sources should be limited for communicational reasons between these persons. Besides, motion controllers already have haptic feedback. Using *Leap Motion* with virtual hands misses haptic cues. Therefore sound was implemented to provide adequate feedback.
- *Bread Crumb System:* The bread crumb system consists of multiple parts. The visual appearance contains more or fewer components depending on the hierarchy level in which a menu element is located. The home icon is displayed on the left part of the system. The name of the currently opened menu level is always in the upper right. Between those components are the remaining names of the levels. The interactive elements are variable in size and adapted to the contained text to keep the bread crumb systems size as compact as possible. The complete breadcrumb is shown in Fig. 4.14 (b).

Chapter 5

Evaluation

A user-oriented empirically evaluation was carried out to improve the prototype in the course of the iterative development process. Different tasks had to be performed during the experiments to obtain quantitative data. Usability tests, questionnaires and interviews were intended to get qualitative and quantitative measurement results. An overview of the collected data is given in this Chapter. The final prototype is examined by a heuristic evaluation and two experiments.

5.1 Methodology

Usability tests are used to measure the overall impression of an interactive system and have their origin in experimental psychology. In these tests, the participant has to solve clearly defined tasks under certain conditions. Weak points can be identified and eliminated with this method [33, p. 161]. This type of test helps to evaluate and improve early variants of designs in the development process [9, p. 95].

Usability tests are very time consuming with a large number of participants. The best results can be achieved when five participants run as many small tests that are needed in the iterative optimisation process [52]. Usability can only be verified to a specific context in which it is applied. The effectiveness, efficiency and satisfaction of the participants are measured psychometrically [6]. In addition to the usability test, a user experience questionnaire can be conducted to obtain more qualitative data to initiate optimisations. Questions are asked regarding attractiveness, perspicuity, efficiency, dependability, stimulation and novelty [36, p. 385].

The state of a prototype is tested in a controlled laboratory experiment. The results help to eliminate found weak points and to improve the overall system. This type of evaluation involves a group of end-users testing a prototype. Complex systems may be separated into parts, which are individually tested. These parts can be grouped into functional sections, pictographs, operating elements and comprehensibility of system messages [33, pp. 160–161].

5.2 Data Collection

Various methods were used to obtain qualitative and quantitative data during the entire evaluation process. The participants' tasks were evaluated in two laboratory experiments. An audio recording was carried out during the experiments. Afterwards, a transcription of the audio material was conducted as a *Think Aloud* protocol, mentioned in Section 2.2.3. Questionnaires to measure the usability and the user experience were presented to the participant after the tasks were completed. Further open questions were clarified in a final interview. These research methods are explained in more detail in the following Subsections.

5.2.1 Laboratory Experiment

The experiments took place in the premises of BMW at Munich, where the design review process is usually carried out. The participants were seated on a platform with a dimension of 2×2 meters. A set consisting of *HTC Vive Pro* with a *Leap Motion* was installed. Those tools are described in Section 4.1. The testers did not experience any restrictions with the operating range of the hands.

There are two fundamentally different experiment designs to compare design variants. First, in a *Between Subjects* experiment the participants are randomly distributed over several variants of a design. Second, the *Within Subjects* experiment have participants who compare and test only two different variant designs [33, p. 165]. *Within Subject* was selected for this experiment since different variant designs should be compared.

5.2.2 System Usability Scale (SUS)

The *System Usability Scale* (SUS) is a widely used standardised questionnaire to evaluate the usability of an interactive system. It was developed to determine a single measurement result of perceived subjective usability. The SUS offers a questionnaire which is divided into ten Sections and contains positive as well as negative questions [27].

The SUS was selected for the thesis because it has several advantages over other evaluation methods. It was developed in the nineties, and its standard form is still the most widely used test method for usability. It is a powerful tool which will be used in the foreseeable future with high probability [27]. The usage of the SUS in the forthcoming is an advantage if new controllers and input devices have to be compared with each other.

The participant evaluates the SUS by rating a question with a number between (1) and (5). The questionnaire is numbered consecutively. One point of the entered value is subtracted from the calculation for even numbers. The result of the odd numbers is subtracted from (5). The obtained values of the ten questions are added to a sum and multiplied by the number (2.5). If participants are not able to give a rating, they have to rate a question with the average value of (3). Figure 5.1 shows how well a system performs in a test. The grading is in a range between A - F, where F is the worst possible grade. This process is the standardised calculation method of the SUS [27].

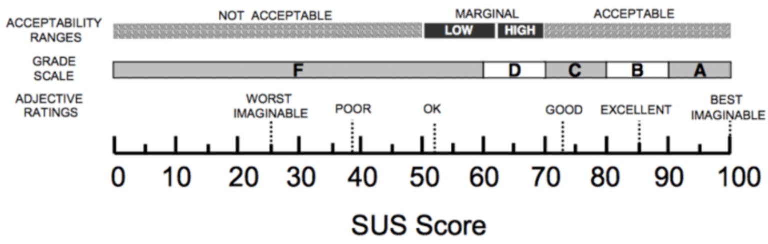


Figure 5.1: The SUS score system for interpreting the result. Source: [7].

Table 5.1: Benchmark intervals for the UEQ scale.

	Att.	Eff.	Per.	Dep.	Sti.	Nov.
Excellent	≥1.72	≥1.64	≥1.82	≥1.6	≥1.50	≥1.34
Good	≥1.50 <1.72	≥1.31 <1.64	≥1.37 <1.82	≥1.40 <1.60	≥1.31 <1.50	≥0.96 <1.34
Above Average	≥1.09 <1.50	≥0.84 <1.31	≥0.90 <1.37	≥1.06 <1.40	≥1.00 <1.31	≥0.63 <0.94
Below Average	≥0.65 <1.09	≥0.50 <0.84	≥0.53 <0.90	≥0.70 <1.06	≥0.52 <1.00	≥0.24 <0.63
Bad	<0.65	<0.50	<0.53	<0.70	<0.52	<0,24

5.2.3 User Experience Questionnaire (UEQ)

The *User Experience Questionnaire* (UEQ) is a fast and reliable questionnaire to assess questions regarding the user experience of an interactive system. It is possible to continuously improve the user experience of a system by comparing design variants. Comparisons can be made with direct competitors on the market, also known as benchmarks. Decisive for the usage of this questionnaire in the lab experiment is to determine whether the product offers a sufficient user experience. The questionnaire contains six scales with 26 items [36, pp. 384–385].

On the UEQ development team's website[1], the current questionnaire (version four) is downloadable in more than 20 languages. A handbook is accessible for the questionnaire evaluation, which is prepared for download as PDF. An Excel[2] template is available for the analysis to facilitate the evaluation. The UEQ provides a benchmark consisting of a data set of 18433 persons and a total of 401 studies. Business software, web pages, webshops and social networks were evaluated. Conclusions can be derived from the relative quality by comparing the results with the product to be evaluated. An overview is given in Table 5.1 to compare the results of the laboratory experiments.

The questionnaire for the UEQ can be found in Appendix E.6 (p. 100). Assumptions can be made where concrete improvement is needed with the evaluation of the six user experience qualities (attractiveness, perspicuity, efficiency, dependability, stimulation, novelty) [36, pp. 390–391].

[1] https://www.ueq-online.org
[2] https://products.office.com/en/excel

5.2.4 After Scenario Questionnaire (ASQ)

The *After Scenario Questionnaire* (ASQ) has a specific focus on the satisfaction of the user. The analysis of *Ease of Task Completion, Satisfaction with Completion Time* and *Satisfaction with Support Information* allows the evaluation of three dimensions of effectiveness, efficiency and satisfaction. The ASQ is performed during the laboratory experiment after the participant has completed a scenario. In Appendix E.4 (p. 98) the questionnaire is available which is used for the experiments. After each scenario, the participant is confronted with the same three questions. The attendee is asked to rate these on a seven-level *Likert Scale* [1] ranging from strongly agree (1) to strongly disagree (7) [26]. The questions are listed below:

1. Overall, I am satisfied with the ease of completing the tasks in this scenario.
2. Overall, I am satisfied with the amount of time it took to complete the tasks in this scenario.
3. Overall, I am satisfied with the support information (online help, messages, documentation) when completing the tasks.

A calculation can be performed after the participant has completed all scenarios and answered the ASQ questions. The average of the mean values from the three questions is applied. This value can be used to derive a statement about each scenario. The average value of all scenarios is used to make a statement for the entire system.

5.2.5 Semi-Structured Interview

A semi-structured interview with the participants was conducted at the end of an experiment in the first lab experiment. This type of interview combines structured and unstructured interviews and includes closed and open questions [32, sec. 7.4.3]. The answer is left to the respondent for open questions, and they cannot be answered with yes or no. In closed questions, the respondent has no choice but to answer with yes/no, or the questions are forced choices [21].

5.2.6 Think Aloud Protocol

This method is used to capture the thoughts of the user during the experiment. This method is described in Section 2.2.3. The qualitative data obtained from the interview, as well from the *Think Aloud* protocols, are encoded and categorised. The categories are available in Table 5.2.

5.3 Heuristic Evaluation of the Interactive System

The final prototype is compared with the usability heuristics by Jacob Nielson. The heuristics details are available in Section 2.2.3.

1. *Visibility of system status*

   In *Feedback Systems*, which can be found in Section 4.4.3, eight feedback systems are described, that are integrated into the interactive system. Whenever a menu element was selected, care was taken to ensure immediate feedback for the user.

Table 5.2: Experiment transcription categories and corresponding encoding.

Category	Name	Encode	Name
C1	Navigation	E1	Menu Elements
C2	Interaction	E2	Trackpad Sensitivity
C3	Design	E3	Ergonomics of Controller
C4	Ergonomics	E4	Menu Structure
C5	Feedback	E5	Interaction Feedback
C6	Experience	E6	System Behaviour
C7	Missing Feature	E7	Blind Navigation
		E8	Execution Time
		E9	Signifiers
		E10	Interactive System
		E11	Gamification
		E12	Icons
		E13	Breadcrumb
		E14	Interaction with Environment and Vehicle
		E15	Interaction Technique
		E16	Text
		E17	Cognitive Aspects

The different feedbacks occur in the form of visual, haptic and auditory to support the user during work.

2. *Match between system and the real world*

   Three well-known interface metaphors are integrated into the interactive system, which can be used at any time according to the stakeholder type. First, virtual hands allow the user a natural way to interact with 2D user interfaces and with 3D objects. This interaction concept is familiar to most users who work with mobile phones or tablets. Second, when using the system with motion controllers, the user can perform actions with a laser pointer. The third possibility is to operate the user interface via the trackpad. The menu content is adapted to the language of the stakeholders and can be interpreted efficiently due to their daily use and familiarity.

3. *User control and freedom*

   - The navigation of the trackpad is designed in such a way that it can be operated entirely one-handed. Each visible menu element has a fixed position in the menu hierarchy and can be selected in sequence.
   - *Virtual hands:* The virtual hands have a fixed menu attached to the hand that automatically adjusts to the user's viewing direction. The selection of menu elements is two-dimensional and thus restricted in the degrees of freedom. With this reduction, the user is in a position to perform efficient and target-oriented interactions.
   - *Laser pointer:* Working with the laser pointer is a 3D action, as it is attached to the motion controller. The selection of menu elements on the user interface is a 2D interaction, which makes the operation simple, like working with virtual hands.

4. *Consistency and standards*
    - The user has the option to select an interaction type and can use it consistently throughout the interactive system. A seamless transition to another interaction type is always possible, depending on the use case. Besides, the user is able to reset the settings previously changed via the user interface. An *Undo* and *Redo* function was not implemented due time restrictions.
    - *Trackpad:* The navigation with the trackpad always uses the same interaction concept. In the menu hierarchy, the left and right touch fields of the trackpad lead to back or forward actions regarding level hierarchy. When the user wants to select the previous or next menu item, the upper or lower part of the trackpad has to be pressed. If the value of a menu element like a slider should be manipulated, the user has the same navigation options as in the menu hierarchy.
    - *Laser pointer:* A consistent interaction concept was ensured during operating with the laser pointer, and it is straight forward. As soon as the user hovers over an active element with the pointer, an action can be triggered via the trigger button. Either the home-icon or the back-button can be selected, to reach a level in the hierarchy towards the root level.
5. *Error prevention*
    - The user interface has inactive components, which show up for stakeholders, which are familiar with the interactive system.
    - *Trackpad:* Navigating with the trackpad has a cool-down optimisation integrated, which eliminates accidental fast double pressings.
    - *Laser pointer:* An interceptor is implemented when using the Laser pointer with 3d widgets or the workflow panel, which disables the main panel user interface, as actions occur. Thus no unintended interactions can be performed.
6. *Flexibility and efficiency of use*
    - Stakeholders can be defined, which handle different complexities of the interactive system. Only the functions which correspond to a stakeholder are visible to the user.
    - *Laser pointer:* The main menu structure, workflows and the tools can be extended as required and can take up to five levels in the hierarchy.
    - *Virtual hands:* This menu has only an option box, where menu elements are placed. This kind of implementation allows a simplified interaction, especially for stakeholders, which are not familiar with the system.
7. *Aesthetic and minimalist design*

    The icons and text pieces act as a unit in menu components. The text content is clipped if the text length is longer. This adaptation ensures that the user interface remains minimalistic and straightforward, as well the recognition of elements is guaranteed. The motion controllers have an abstract surface of wire which makes all interactive buttons visible to the user.

8. *Help users recognize, diagnose, and recover from errors*
   Toast messages provide immediate feedback as soon as an action is performed in the menu system. They have three different output types. Positive events are displayed in green colour. In case objects are not available in a scene or cannot be found by the menu system, a yellow toast message is displayed as a warning. A red error message is displayed when a critical error occurs. For example, this information appears, when the menu system is not installed correctly or manipulated falsely.
9. *Recognition rather than recall*
   - Different signifiers are implemented, which indicate interaction possibilities of features or functionalities.
   - *Virtual hands:* The virtual hand menu opens automatically when the palm is visible. Inexperienced users do not have to worry about activating the menu system.
   - *Laser pointer:* A laser ray is automatically activated and visible as soon as interactivity is possible by targeting an object with the motion controllers.
   - *Trackpad:* The motion controllers show the interaction possibilities of the trackpad dependent of the menu position or menu item.
10. Help and documentation
    Documentation is accessible for visual designers, who are responsible for the maintenance of a project. Help is only available in the form of tooltips when the laser pointer technique is used. There is no need for a help system for the trackpad, as the menu items provide all necessary information.

5.4 Experiment 1 — Interaction Techniques Evaluation

The idea behind the first experiment is to evaluate the product state by comparing two prototypes in different stages. Prospective users are involved in the development process to establish a high level of effectiveness and intuitiveness of the prototype. This experiment is intended to make an essential contribution to identifying weak points.

The optimal menu structure should be determined based on the requirements of the participants. An overall impression of the different stages has to be gathered. The interaction via a laser pointer is compared with the input via the trackpad to check its usability in a professional working environment. It is intended to conduct an as-is investigation of the current state to get closer to the final product. In this Section, the following hypotheses are evaluated and discussed:

Hypothesis 1: The Completion Time of prototype A is higher than the prototype B.

Hypothesis 2: The participant can solve a given task within one minute.

Hypothesis 3: The Error Task Rate of prototype A is higher than the prototype B.

Hypothesis 4: Prototype A has reduced usability compared to prototype B.

5.4.1 Methodology

The participants were asked to perform different tasks during the experiment. The time and error rate were measured for each participant to collect quantitative data. They

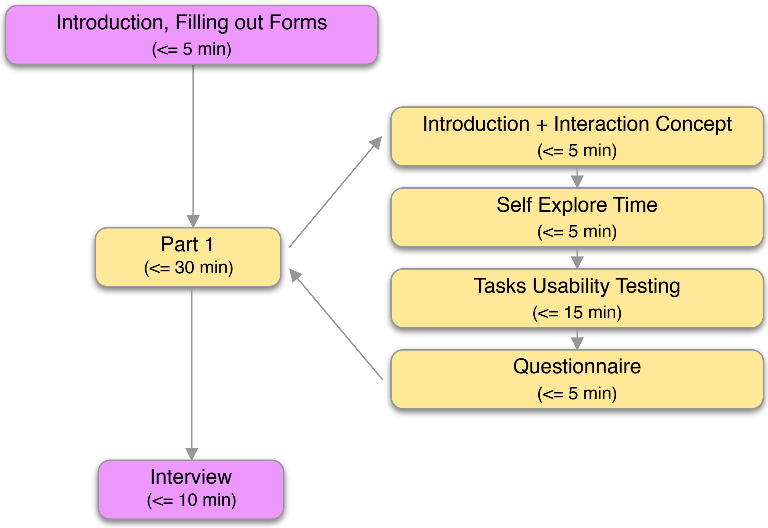


Figure 5.2: Overview of the first lab experiment procedure.

had to think aloud during the experiment. A usability test was performed after the completion of the tasks to collect qualitative data. The lab experiment concluded with an interview. The experiment was conducted in the German language.

Procedure

Figure 5.2 provides an overview of the whole procedure. The duration of a test was designed to be completed in less than one hour. First, all participants received a general overview before the experiment started. The introduction ended with filling out a consent and a demographics form. The action block took around 15 min.

The experiment started with a practical introduction to the interaction possibilities with the motion controllers. After the introductory, the participants were offered the opportunity to explore the user interface by their own within a short period. This activity was followed by the execution of eight tasks, which can be found in Appendix F.1 (p. 102). The participant received specific and detailed assignments which had to be solved within the interactive system. The order of the tasks to be executed was randomised for all participants. The tasks were written down in advance and read to the participant to provide every participant with the same instructions. This process ensured consistency of the wording during the information transfer. The *SUS* questionnaire formed the end of the experiment, which is mentioned in Section 5.2.2.

Participants

Five male participants took part in the experiment. The age of two participants was between 30 and 39. The age of another two was between 40 and 49, and the last participant was older than 50. The attendees were subject matter experts *SMEs* [9, p. 54], which were specialists in the design review process at BMW. Due to legal restrictions, only BMW employees could be part of the experiments. The sample is not representative.

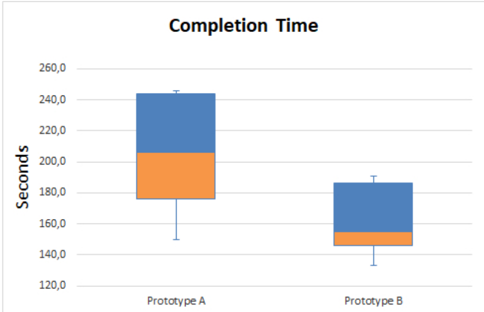


Figure 5.3: Result of the experiment 1 for Completion Time.

The deciding factor which participants can be included in this experiment was that the prototype had not been tested in VR. All participants were right-handed and had a normal or corrected-to-normal vision. Three of them wore glasses, and two participants had contact lenses. None of the attendees reported eye disorders like colour weaknesses. All of the five participants had already VR experiences and were familiar with different interaction techniques due to their work. Four participants claimed to be experts; one of them considers himself advanced in the field of VR interactions. All testers signed an informed consent form and filled out a demographic profile.

Data Analysis

A repeated-measurement analysis of different stages of the interactive system with two within-subjects factors was conducted: The interactive system in the early stage and a revised version based on the results of the first part of this experiment. A *Wilcoxon Signed Ranks Test* [56] (WSRT) was used for the comparisons of the variants. Therefore, the independent variable was the variant state of the interactive system. The dependent variables were the *Task Success Rate*, the *Completion Time*, and the *Error Rate*.

5.4.2 Quantitative Result

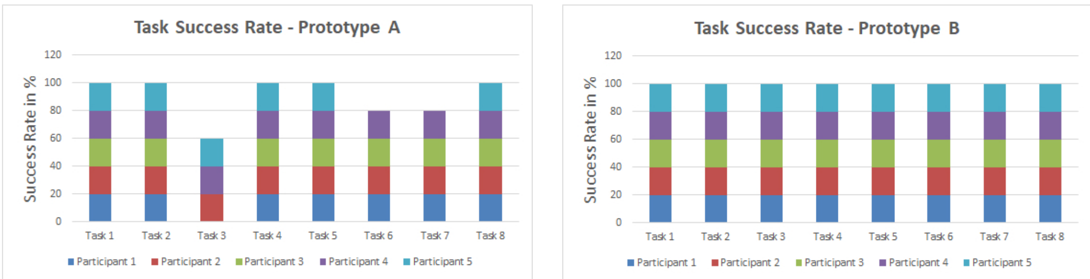
The evaluation of the interactive system included three different quantitative metrics. The effectiveness in terms of *Task Success Rate* was measured. This measurement determines how many tasks are performed correctly. The second measurement, the *Completion Time*, refers to the efficiency which is measured in minutes and seconds. The *Error Task Rate* is the last metric to be measured. Attention was paid to how often wrong navigation within the menu was performed.

Completion Time

Prototype A and prototype B were compared with the WSRT in terms of completion time. The results are shown in Table 5.3 with a visual representation in Figure 5.3. For the prototype A, the average total time of 3.477 minutes was measured to solve all tasks. Prototype B achieved an average of 2.727 overall time. The investigation shows that the measurement results are not significant ($p = 0.223$). This result is due to a small

Table 5.3: Analysis of the Completion Time between prototype A and B.

Wilcoxon Signed-Rank Test for Paired Samples ($\alpha = 0.05$, $n = 5$, $df = 1$)									
Comparison	mean	SD	min	max	med	n	z	p	w
A	3.477	1.183	2.27	5.15	3.417	5	-1.219	0.223	3
B	2.727	0.301	2.43	3.08	2.583	5			
Ranks	n	mean	sum						
negative	3^a	4	12						
positive	2^b	1.5	3						
Result	a. completion time prototype $B <$ completion time prototype A								
	b. completion time prototype $B >$ completion time prototype A								



(a) (b)

Figure 5.4: The comparison result of the Task Success rate. Task Success Rate Prototype A (a), Task Success Rate Prototype B (b).

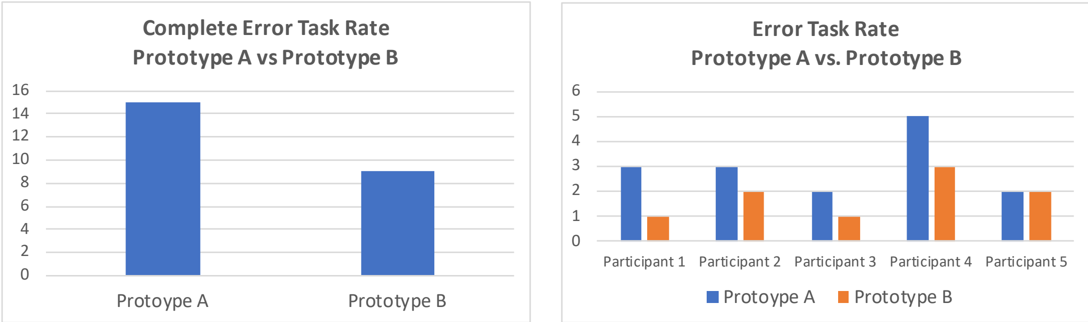
number of samples from ($n = 5$). It is challenging to find significant differences between the two samples when the quantity number is low.

The WSRT result is the sum of (12) negative ranks and the sum of (3) positive ranks. The evaluation revealed that the prototype B has a faster completion time than the variant prototype A. The results indicate that the first hypothesis can be confirmed.

Task Success Rate

The measurement of the *Task Success Rate* provides information on task completion within a defined time. This experiment considers only whether a task was executed correctly without detours or not. The time factor serves as the input data for calculating the success rate of the task. A task had to be solved within a minute.

A binary classification was used for the evaluation, differentiating between *Completed* and *Not Completed*. In Figure 5.4 the individual tasks of the two prototypes are compared. The success rate achieved with the prototype B was 92.5%. Prototype A exceeded the defined time mark of one minute three times. The success rate for prototype B is 100%. All tasks could be performed in a defined time. The *Task Success Rate* shows that the participants of prototype A could not complete all assignments within one minute. The second hypothesis can be confirmed as a result indicates, because all tasks could be solved under one minute with prototype B.



(a) (b)

Figure 5.5: The result of the complete Error Task Rate and the Error Task Rate for each participant. Complete Error Task Rate (a), Error Task Rate Participants (b).

Error Task Rate

The *Error Task Rate* describes the relationship between a task to be performed, and errors made to achieve a goal. In this experiment, an error was detected when a menu item was searched at a wrong position or a false input was made.

Prototype A shows a higher *Error Task Rate* when comparing the two prototypes. 15 errors were made during the tasks by the participants in total. Prototype B had an error rate of 9. A graphical result is shown in Figure 5.5.

Hypothesis four can be answered with this result: Prototype B has a lower *Error Rate* than prototype A. The third hypothesis indicates to be confirmed, as prototype A has a higher *Error Task Rate* rate than prototype B.

5.4.3 Qualitative Result

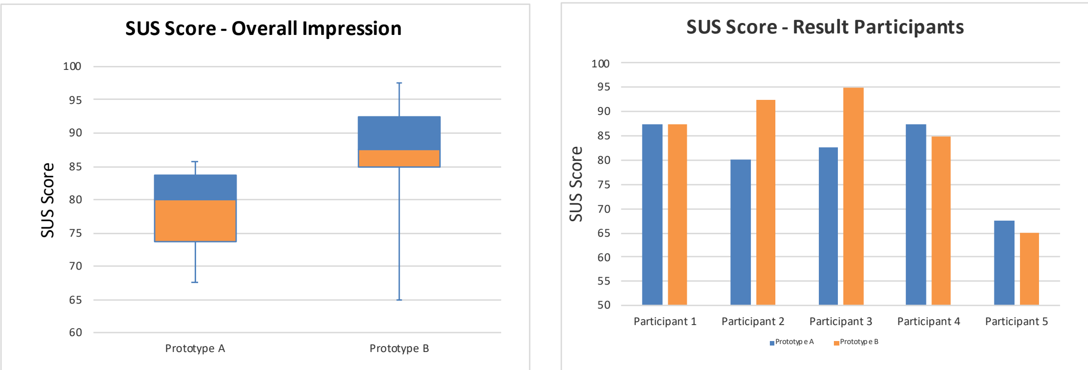
Qualitative data were collected using usability tests, think aloud protocols and interviews. In the following Sections, the SUS of both prototypes is evaluated and compared with each other. The conclusion is formed with statements of the participants during the experiments, which are listed in summary. Identified problems of prototype A are pointed out, and improvements of prototype B are discussed.

System Usability Scale

In the comparison between both prototypes, prototype B had achieved a higher SUS score. The mean value for prototype B was 82.5 (SD: 8.216), which is followed by prototype A with a mean value of 81 (SD: 10.607). Both prototypes received a SUS grade B, which is classified as *Excellent* and achieved an *Acceptable* in the acceptance range. The grades were determined via the *SUS Score System* which can be found in Section 5.1.
Figure 5.6 shows a visual representation of the SUS result and a breakdown of each participant. The results indicate that the first hypothesis can be confirmed, as prototype B has achieved a higher score than prototype A.

Table 5.4: The SUS Grade of the compared prototypes.

Prototype	SUS score mean	SUS Grade
A	80.5	B - Excellent
B	85	B - Excellent



(a) (b)

Figure 5.6: The SUS score result of the overall Impression the impression of the participants. SUS score overall Impression (a), SUS score impression participants (b).

Interview and Think Aloud

The data were subdivided into thematic categories. This division was made in the context of optimising the interactive system. The category names of the division are navigation, interaction, design, ergonomics, feedback, experience, missing features and errors. A detailed segmentation is shown in Table 5.2. The content of the statements was encoded to obtain a better classification of the issues.

The complete summary of qualitative data can be found in Appendix G.1 (p. 104). The following list contains only the most relevant qualitative findings obtained from the whole first laboratory experiment:

1. *Navigation*
   - Some issues were identified when using buttons inside of menu elements like in the colour component. A logical arrangement of the concerning elements could improve the navigation (E1).
   - The menu structure in prototype A caused problems compared to prototype B when searching for specific menu items. A revision and simplification of the root menu structure improved the navigation within the interactive system (E4).
   - A positive statement from some participants was that they could navigate blindly between a geometry or style using the trackpad. They understood the multiple usages of different interaction techniques. They could choose to regard their needs (E7).
2. *Interaction*

- A critical issue dealt with interactive menu components inside of the menu system. In some situations, the participants had problems to leave/confirm a menu element. Since all buttons were assigned to a function, only the trigger was available for executing a *Confirmation*, a *Going Back* and an *Exit* possibility. Some participants persons did not think that the trigger would perform an action at this point. The integration of signifiers in prototype B improved this circumstance (E1).
- The interaction of the trackpad caused problems for some participants. Such an interaction and navigation possibility has not yet been tested or tried by some attendees. Unintentional and unwanted interactions occurred during the tests due to the sensitivity of the trackpad, which increased the execution time. It turned out that an acclimatisation phase was required for this type of interaction. Five minutes to explore the interactive system were not sufficient to deal with this type of interaction. The prototype B could achieve an improvement in sensitivity compared to the first variant according to the participants' statements (E7).
- The participants reported working with the laser pointer is simple, intuitive and suitable for beginners. Experienced users would probably additionally use the trackpad interaction for faster execution time and navigate more comfortably (E8).
- The overall impression was that using the laser pointer interaction technique is foolproof. The*Smart-Selection* was very intuitive, and working with the laser pointer is convenient (E10).

3. *Design*
   - The participants had problems with some menu items understanding the arrangement in prototype A. The layout was analysed and improved (E1).
   - In a direct comparison of the two prototypes, prototype B was graphically convincing with the new design. The final design was only integrated into prototype B (E10).
   - Some suggestions were made regarding the font enhancement and the icon symbolism. The user interface was improved graphically, and the icons were easier to understand. The reduction of more icons and less text was accepted as decisive (E12).
4. *Ergonomics*

   The participants had made several statements regarding the ergonomics of the controller. The use of all interactive elements of the controller led to uncomfortable hand and finger postures. Besides, these postures could lead to unwanted interactions. Mention was given about the long stroke of the trigger. This circumstance can be perceived as strenuous if it is used frequently. The laser beam angle has to be optimised to improve the hand posture (E3).
5. *Feedback*
   - In prototype A, the participants had trouble interpreting a correctly performed action caused by the lack of appropriate feedback. As a result, different feedback systems were integrated to give the user a good indication of the

current status. The use of flashing colours on the buttons of the motion controller was well received, but the signal was too soft in early testings. Not all participants saw that flashing or could derive information (E5).

- Some participants mentioned that the outline signifiers helped to guide the interaction process when a panel is detached and should be moved. The execution time was improved when perceiving, such as helpers (E9).
- The participants stated that the interactive system felt good. The grabbing feature became better and intuitive as soon as the concept of grabbing in VR was understood. The laser pointer seemed to work more efficiently than the trackpad. It could be used one-handed for operations and was more precise (E10).
- The feedback was positive about integrating more icons and reducing the amount of text, as this makes the menu system more user-friendly. Seeing no text on icons was especially problematic at the beginning for new users. A text was subsequently added to the root level, and the tooltips were activated by default (E12).

6. *Experience*
   - The testers reported that they were faster in performing interactions or at least did not notice any difference when comparing both prototypes. Some participants said they were faster with the laser pointer (E8).
   - It was hard for some participants to deal with the user interface and all the integrated features during the explore phase. There was a desire to integrate various stakeholders with different interests to restrict complexity. One participant stated that this user interface is perfect for beginners when only a few features are selectable (E10).
   - The trackpad interaction technique became more comfortable during prolonged use due to the more pleasant hand posture during actions was a statement given by an attendant (E15).
7. *Missing Feature*
   - In prototype A, the participants missed a bread crumb system to be informed about their current state (E1).
   - Some improvements were made to the trackpad sensitivity to prevent double event triggering in prototype B (E2).
   - The placement of the panel was optimised and enhanced with features that allow faster changing of settings. The spawning behaviour of detached menus has been improved to work more conveniently with those elements (E6).
   - A request was made to integrate playful content to enhance the user experience. Further features were developed and tested in the second experiment (E11).

5.4.4 Discussion

This experiment presents the current product state of the interactive system while comparing two prototypes. In the conclusion of this experiment, the results of the first experiment are compared with the hypotheses. The *Interaction Technique Experiment* shows that the latest prototype B has a better outcome than prototype A. The first hypothesis indicates to be confirmed with this result. All results are not significant, and the experiment is not representative.

The *Hypothesis 1* deals with the *Completion Time* that has to be spent on the respective prototype. On average, the participants had to spend less time on prototype B to complete the tasks. This result shows that it is possible to work more efficiently with the newer variant (prototype B) right from the start.

The *Hypothesis 2* indicates to be confirmed as well. All tasks were be performed within one minute with prototype B. A timeout of the maximum duration would not stand for intuitiveness and would trigger negative side effects for the user, which could complicate the work with the interactive system.

Prototype B could achieve a lower *Error Task Rate* than prototype A. The *Hypothesis 3* indicates to be confirmed, as the result shows. The analysis of prototype A outcome could lead to optimisation. The menu structure and its hierarchy were adapted to make it easier to find elements within the menu.

In the SUS scale comparison between both prototypes, prototype B could achieve a higher score. The *Hypothesis 4* indicates to be confirmed. This evaluation provides a subjective assessment of the users of the interactive system. The improvements could contribute positively to the overall usability result.

Additional Notes

The time factor triggered the desire of some participants to complete the exercises as quickly as possible. This fact meant that the content of some menus was not read correctly. Sometimes the first matching menu item was used. Furthermore, regarding the time, it was observed that once an unwanted action was triggered, a desperate attempt was made to get back to a good starting position as quickly as possible. This circumstance led to further errors. The time factor is only conditionally suitable for measuring interaction techniques, especially in connection with an intuitive menu.

5.5 Experiment 2 — Intuitive Interaction Concept Evaluation

The second experiment had the objective to evaluate the interaction concept and the current product status, which was in the alpha phase. All available interaction possibilities within the interactive system had to be tested, which included input via laser pointer, trackpad, motion controllers and direct hand input using *Leap Motion*.

Use case scenarios were applied to evaluate everyday interactions of the different stakeholders, to validate the intuitiveness of the interaction concept. A scenario consists of partial tasks which the participant has to perform. In this experiment, the following hypotheses should be answered:

Hypothesis 5: The satisfaction in performing scenario tasks show an ASQ average and a score between *1* and *2*.

Hypothesis 6: Each scale of the UEQ result should be in a range between *Excellent* and *Good* with the benchmark comparison.

Hypothesis 7: The result of the SUS score should be within an *Acceptable* range and the value of the SUS grade should be at least *B*.

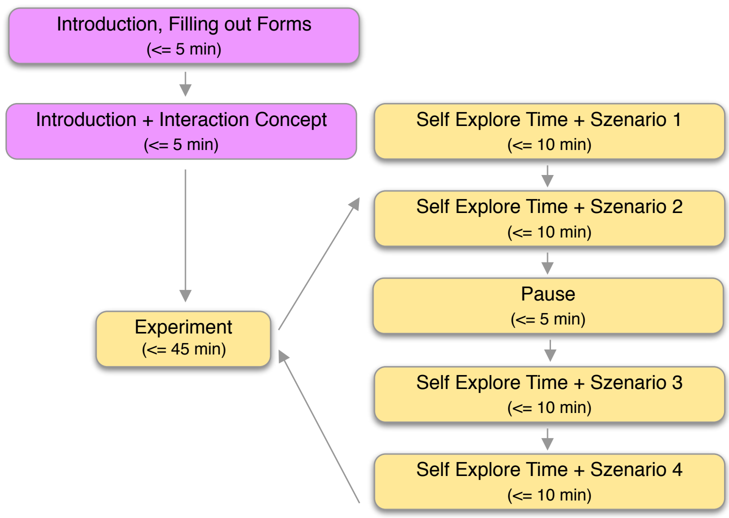


Figure 5.7: Overview of the second lab experiment procedure.

5.5.1 Methodology

The participants had to solve several tasks within scenarios. After each scenario, they had to answer three questions regarding satisfaction. The participant was encouraged to speak out loud when performing the tasks. A usability and a user experience questionnaire were conducted in the end.

Procedure

Figure 5.7 provides an overview of the whole procedure. The length of an experiment was intended to be finished in less than one hour for each participant. The attendees had to sign a consent form and fill out a demographics questionnaire. Once the actual experiment started, an overall outline was first given to all participants.

The cognitive load increased during the experiments, as more features were added in each scenario. The participants received more opportunities to fulfil their tasks, going along with increased complexity. The aim was to find out whether the needs of the various stakeholders could be satisfied. The respective tasks were specified in Appendix F.2 (p. 103).

In each scenario, the participant had up to eight minutes to explore the menu. The tasks were executed in sequence due to the increasing complexity and were performed equally for all participants. A task of a scenario was presented in a comprehensive form to the participant. Once a task was executed correctly, the next one followed until all tasks of a scenario were completed. The participants were encouraged to think aloud while performing the tasks. The attendees were asked three questions about the satisfaction, as described in Section 5.2.4 at the end of a scenario. After all the scenarios had been completed, two questionnaires were presented to the participant.

Participants

One woman and four men took part in this experiment. Two subjects were between 40 and 49, and the other three testers were between 30 and 39. The participants were subject matter experts *SME*s [9, p. 54], specialists in the field of digital visualisation in virtual product development. Only BMW employees were able to participate in this experiment as it took place at BMW's premises as well. The samples are not representative. The selection for participation in this experiment was limited to the fact that there was no VR experience with the interactive system among the participants. All participants were right-handed, three of them wore glasses, and two testers wore contact lenses. None of the attendees reported achromatic colour blindness. All participants stated to be experts with VR interactions. Two attendees mentioned owning a VR device. All subjects signed a consent form and completed a demographic questionnaire.

Data Analysis

Two questionnaires were handed out at the end of each experiment to assess the usability and the user experience of the final developed system. At the end of a scenario, the ASQ was applied to measure the user's satisfaction for performed tasks. The achieved SUS score for this experiment (prototype C) was compared with the winner of the first experiment, prototype B. The aim was to get an impression of how the system evolved with extended functionality. The UEQ results were benchmarked against a dataset to get an assessment of the current system in terms of user experience.

5.5.2 Qualitative Result

Qualitative data were collected through usability and user experience tests. An ASQ after each scenario and a think aloud protocol were conducted. At the end of the experiment, each participant was given a SUS and a UEQ questionnaire to evaluate the latest status of the prototype.

After Scenario Questionnaire

This questionnaire was used to evaluate the difficulty of tasks, the time spent on the tasks and the information provided on the satisfaction of each participant. No comparison was made with any other prototype. The evaluation of this test should contribute to the overall assessment of usability and user experience. The participant had to answer questions after each scenario. The rating scale has seven levels from (1) to (7), with (7) being the worst rating.

The average value of all participants in a scenario was determined in this evaluation. The difficulty of the tasks on this scale was in the range between 1.4 and 2.6, with a mean of the value of 2. The participants rated the time they needed to cope with a scenario between 1.4 and 2.2. This result provides a mean of 1.75. The supporting information available to cope with a scenario, such as messages and tooltips, was rated between 1.4 and 2.2. The evaluation was based on a score of 1.4. A graphical overview about each scenario is given in Figure 5.8. The overall result of the ASQ score of all scenarios amounts to an average value of 1.933. Hypothesis five indicates to be accepted with this test since the ASQ score is determined between (1) and (3).

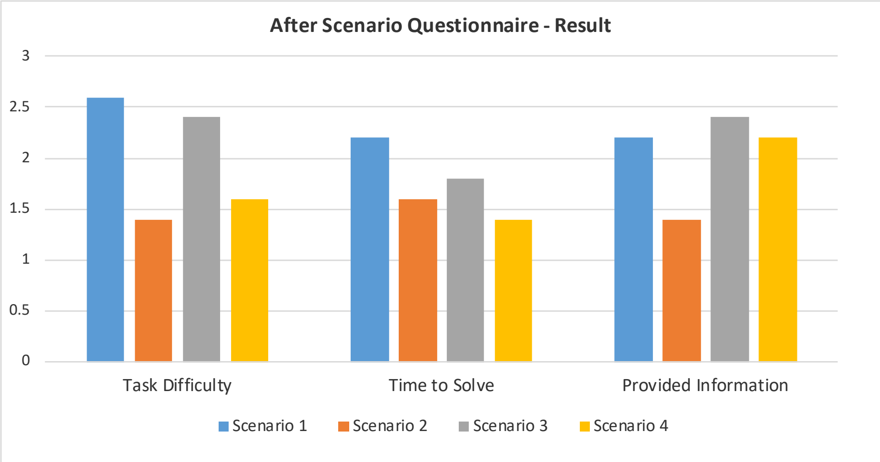


Figure 5.8: The result of the After Scenario Questionnaire.

User Experience Questionnaire

This questionnaire contributes to the main focus of this experiment to validate the user experience of the current system. Overall, there were no inconsistencies in the evaluation of the data. The participants could answer all items seriously. This result was measured with the simple heuristic method, which is included in the template.

The benchmark indicates the quality of a product by comparing the target with other products from a dataset. The chart in Figure 5.10 shows a graphical result of the prototype for the different scales. The exact results are displayed in Table 5.5 and compared with the values in Table 5.1. A single item's result can be seen in Figure 5.9.

The attractiveness, dependability, stimulation and novelty could reach an *Excellent* grade. The perspicuity score was *Good*, and the efficiency was *Above Average*. The benchmark does not provide a total score of all scales due to the different measurement ranges.

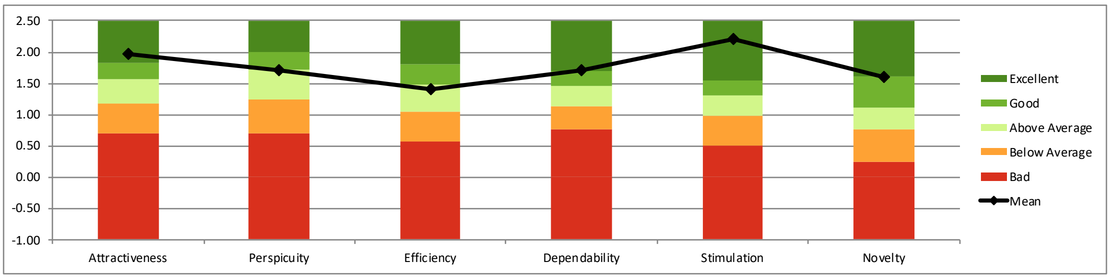


Figure 5.10: The result of the benchmark with the 6 scales.

The developer's handbook of the UEQ does not explicitly describe how many participants are required to complete a questionnaire with adequate results. Between 20–30, participants should attend an experiment for reliable results. The user experience analysis

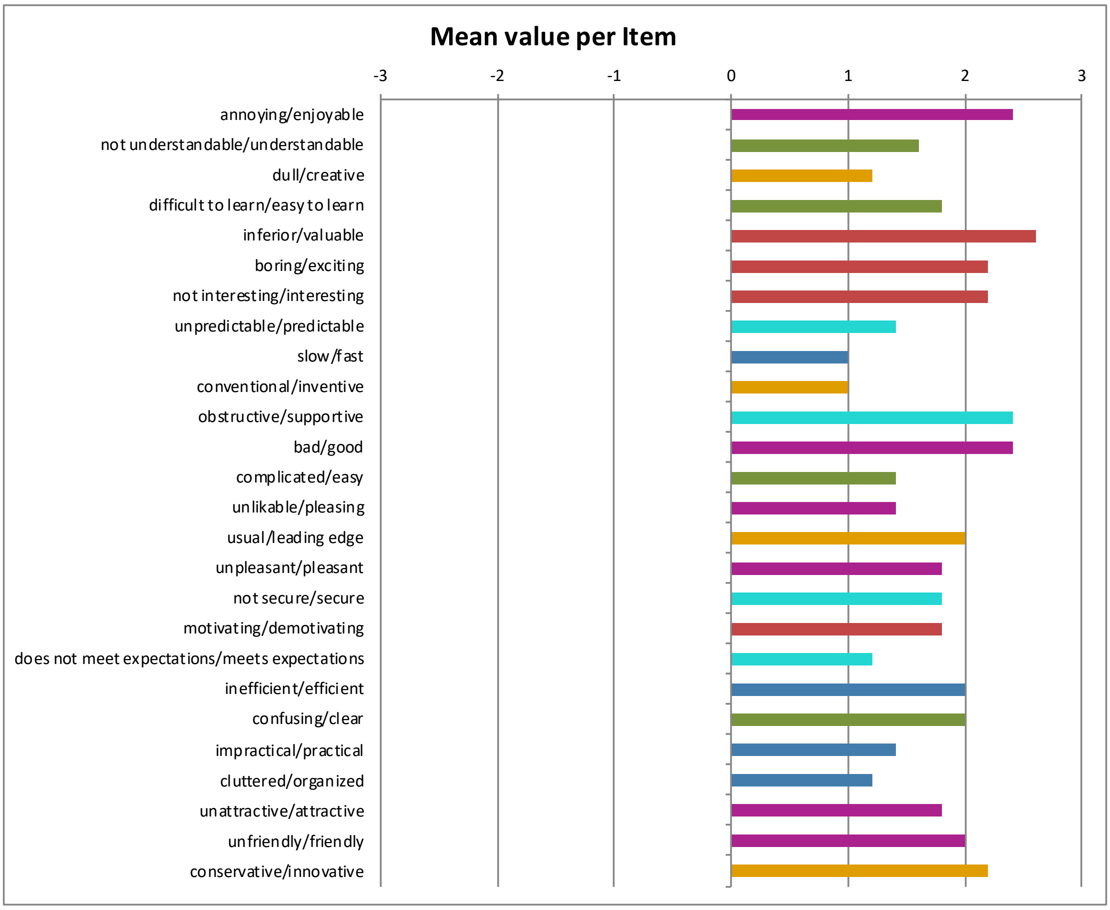


Figure 5.9: The result of the mean values of the UEQ.

Table 5.5: Experiment 2 - Results benchmark.

Scale	Mean	SD	Comparison to benchmark	Interpretation
Attractiveness	1.97	0.14	Excellent	In the range of the 10% best results
Perspicuity	1.70	0.45	Good	25% of results better. 50% of results worse
Efficiency	1.40	0.38	Above Average	25% of results better. 50% of results worse
Dependability	1.70	0.33	Excellent	In the range of the 10% best results
Stimulation	2.20	0.54	Excellent	In the range of the 10% best results
Novelty	1.60	0.52	Excellent	10% of results better 75% of results worse

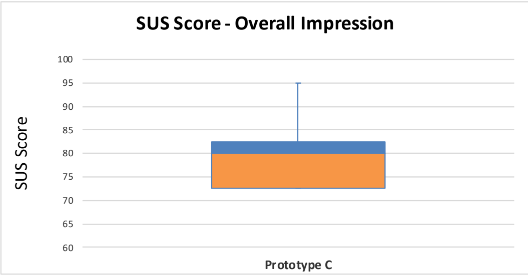
does not provide reliable data, as only five test persons participated. Hypothesis six indicates to be not accepted since the scale of efficiency matched a ranking of *Above the Average*. All other scales reached at least a *Good* or *Excellent*.

System Usability Scale

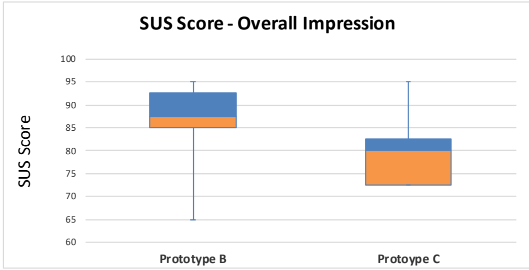
The SUS score achieved a mean value of 80.5 (SD: 9.25). The result is shown in Table 5.6. The prototype C receives a SUS grade of *B* as a result, which is classified as an

Table 5.6: Experiment 2 - Result of SUS score.

Prototype	SUS score mean	SUS grade
B	85	B - Excellent
C	80.5	B - Excellent



(a)



(b)

Figure 5.11: The overall impression SUS score result of the participants and comparison of prototype B versus C. SUS score result (a), SUS score comparison prototype B versus C (b).

adjectives rating with an *Excellent*. A graphical representation of the overall result of the general impression is shown in Figure 5.11.

Figure 5.11 (b) contains the winner of the first experiment (prototype B), which is compared with the current product state, prototype C. The usability has decreased with an expanded range of features. The result is available in Table 5.6. The examined prototype receives a comparable SUS score as the score of the first prototype B. The hypothesis seven indicates to be accepted since a SUS grade of *B* could be obtained. Therefore the prototype is in the *Acceptable* range.

Think Aloud

The think aloud protocols were evaluated on the same principles as in the first experiment by splitting the data into categories, as shown in Table 5.2 (p. 52). The statements of the participants were encoded. A summary of the think aloud protocol can be found in Appendix G.2 (p. 108).

1. *Navigation*
   A problem was mentioned when using virtual hands with the menu, that it is not obvious how to return to the root level from a higher one. An additional back button could enhance usability (E1).
2. *Interaction*
   - Some statements were made regarding the interaction of menu elements when using virtual hands. The 3D menu should have a minimum distance to the touched geometry for better mapping (E10).
   - Two participants indicated preferring to click rather than to touch when working with motion controllers to avoid deliberately triggering an action.

Touching could represent a high barrier in the first-time contact with VR and motion controllers (E2).

- Sometimes interactions with virtual hands proved to be difficult, as the hand animations did not function properly. Inaccurate actions were performed, which were perceived as laborious. Comments have been collected where virtual hands work very well with the 3D menu. A suggestion for improvement was made regarding the display angle of the menu using virtual hands. The position of the menu should be improved as well (E15).

3. *Design*
   - The option-component in the module selection could be larger to facilitate the interaction process while working with virtual hands. An improvement could be accomplished when adding a greater distance to the first element to avoid unintentionally navigating to the element below (E3).
   - A 3D button requires better visual feedback when touching the element with virtual hands. Currently, the operator must be very close to the element to see what is happening (E5).
   - Several participants indicated that the icon does not function as expected with the tool selection element (Workflow selection). The rotation should be optimised at this point. Recommendations for the style of icons were made for the workflow tools icon and the rotation icon, for a better understanding (E12).
4. *Ergonomics*
   - The adjustment of the angle from the emitted laser beam (motion controller) was received as very positive. After a short training period, the ergonomic selection was considered to be better. This option has an offset angle of -55 degrees. A statement was made that the controllers must always be kept very high without the ergonomic setting to be able to operate them efficiently (E3).
   - The menu could take a better position when using the menu with the virtual hands for improved handling and a better remaining in the users' field of view (E10).
5. *Feedback*
   - A problem was mentioned that an event should only happen when a click sound is perceptible. Currently, an action is triggered at a defined threshold (E1).
   - The audio feedback of the virtual hands was executed incorrectly during button interaction. The audio signal should not be heard until an action is performed when releasing a button (E5).
   - A participant had mentioned that they do not know which objects are intractable when using virtual hands. Otherwise, trying to find out if an interaction is possible will involve a loss of time (E14).

6. *Experience*
   - The feedback of the experience was very positive. Statements have been made that the desired interactions in the system work as expected. The selection of

different interaction techniques is very helpful and allows the user to select the most appropriate one. The motion controllers and the menu harmonise well as a unit (E6).

- The combination with virtual hands and the use of a motion controller was quite well received because interactions considered to be very controllable. It is possible to use the finger for pointing, which is considered as a natural form of communication. As soon as more interactions have to be made inside of the vehicle, the use of two motion controllers is preferred by two participants (E6).
- However, there were also weaknesses identified regarding the experience. One participant faced a serious problem with the steering wheel. Since this person sits very close to the front in a real vehicle, no ergonomic hand position could be adopted without changing the sitting position. Afterwards, the desired interactions could be carried out better, but only with a compromised solution. For some users, scenario three was very overwhelming because of the many features that the menu offered. These participants indicated that a reduced menu would be preferable. Selecting the tools in the lower part of the controller does not necessarily require four additional displays with rotation (E10).

7. *Missing Feature*
   - Two wishes were expressed regarding the menu items. In the first scenario, the participants could use their virtual hands to display a 3D object menu above a vehicle geometry. Two times it was requested that this menu should also be closable (E1).
   - A participant preferred to manually control the trigger when changing a variant, style or package for better control over the event. As soon as several elements are contained in a list, the user could switch more efficient between two desired settings (E6).
   - A request was once expressed that the ergonomic pose should be set as default when looking at the emitted laser beam of a motion controller. Firstly, the laser beam could be noticed more quickly when exploring the controller, and secondly, unnecessary additional adjustment processes could be avoided (E3).
   - The hovering of an element worked very well. A wish was expressed that objects should be selectable. Especially in situations where a group of people would be talking about a specific geometry, it would help all participants to focus on a highlighted geometry. New persons who join a meeting would have an obvious starting point immediately (E5).
   - Some statements were made about the system behaviour. Two participants missed the option to select a step size when adjusting the position of the avatar. It could be a tedious process if the avatar was too far away from a target location. An option to rotate the avatar would also be helpful if the tracking system does not place the character correctly in the scene. Another point was mentioned where it would be desirable to switch the laser beam on and off. It was mentioned that the 3D widgets, which represent trigger points in the VE (door rotation, seating position), should have a bigger collider, allowing them to be easily aimed at a greater distance (E6).

8. *Error*
    - Errors with the 3D button interactions were determined as well with virtual hands. During slow interactions and deep pressure gestures, a button may be pressed two times. An optimisation in this respect is desirable, covering different pressure properties to increase reliability (E5).
    - The right menu on the controller currently showed no tooltips (E5).

5.5.3 Discussion

This experiment presents the analysis of the last prototype in the alpha phase. The focus was on the evaluation of the product state according to the interaction concept, user experience and usability. Questionnaires collected the data to achieve subjective results. Altogether there are three hypotheses which have to be answered in this experiment.

The *Fifth Hypothesis* deals with satisfaction in performing a scenario. A score between (1) and (2) could be achieved. The hypothesis indicates to be accepted. The focus was on an equivalent degree of tasks difficulty for the various stakeholders. Stable satisfaction of the participants could be observed with increasing complexity. The comparison of the first task and the second task shows that the satisfaction with an interaction with a motion controller and a virtual hand is better than with two virtual hands. This result reflects the statements transcribed during the think aloud witch is available in Appendix G.2 (p. 102).

The *Sixth Hypothesis* indicates to be not confirmed with a UEQ result between *Excellent* and *Above Average* through a benchmark comparison. Only the efficiency narrowly missed the mark *Good*. The remaining scales could prove either a good or excellent in the evaluation. Figure 5.10 (p. 65) shows the result of the UEQ. The evaluated result is not significant due to a small number of participants. However, it is quite obvious that the current product status was positively accepted. At this point, optimisations have to be done by eliminating found errors.

The SUS scale of the evaluated prototype C could achieve a score within the *Acceptable* range with a grade a *B*. The *Seventh Hypothesis* indicates to be accepted with this result. This outcome was compared with the result of the first experiment. The winner (prototype B) of the first experiment could achieve a higher SUS score than prototype C. This effect is probably because of the increased functionality and the resulting higher complexity.

Additional Notes

The stakeholder *Beginners* and *Product Managers* were able to handle the menu. No special skills were required to work efficiently with the menu. There was always a risk that when using two virtual hands, the recognition of the hands did not always work smoothly, or the hands influenced each other. As the complexity increased, at least an explanation was needed which features the interactive system offers. Some terms like *Smart Selection, Auto Selection*, *Quick Selection* and *Auto Workflow* needed a short explanation. Once a feature is unknown, it is difficult to spot it, and it is likely used coincidentally.

Chapter 6

Discussion

This part of the thesis offers the final discussion about the developed interactive system and the gained knowledge in developing an institutive interaction concept for VR. The research questions asked in the introduction are answered. In the conclusion, game engine limitations, encountered problems during the iterative development are discussed.

6.1 Discussion

This thesis deals with the interaction design of complex operations and tasks in VR and focuses on developing an application for the product development process in the automotive industry. At the beginning of the writing in Section 1.2, the following research questions were formulated:

RQ1: How should functions and operations be integrated into an interactive system which allows the user to interact with motion controllers and virtual hands?

RQ2: How should the menu structure be integrated into an interactive system so that various users with different prior knowledge can intuitively use it?

RQ3: How can complex switching operations of a vehicle be easily performed in the product development process?

RQ4: How should an interaction concept be designed to use different tools and workflows as efficiently as possible within the product development process?

An analysis was carried out to determine the requirements of the prototype based on user stories to answer these questions. In the iterative development process, experiments were conducted to examine the prototype state, identify weak points and add enhancements. This procedure was supported by usability, user experience and post-tasks questionnaires, as well as interviews and think aloud protocols.

Research Question 1

The time factor influenced the focus on a minimalistic approach to accomplish the final prototype in the given time. The consistency of an interaction concept contributes substantially to the development of an intuitive interactive system. The decision was made in the analysis process to reuse menu elements as often as possible and to integrate a modular and dynamic architecture. The system to be developed is based on a well-known interface metaphor, the *WIMP Metaphor*, which is discussed in Section 2.3.1. This metaphor allows input via a pointing device in the VE. This principle has been applied and further developed by using the same interaction technique whenever a laser pointer is active or when virtual hands are needed. This kind of integration enables users to decide independently which type of interaction is best suited for them in a specific scenario.

Input via the index finger works great as long as the lighting conditions and the environmental requirements for the *Leap Motion* are met. It was recognised at an early stage that acoustic feedback was essential in addition to the graphic response to perform adequate interactions with virtual hands. The more feedback users have, the more easily actions can be executed. Some problems occurred when interactions were performed too quickly. It was not being perceived as a gesture of the system. This effect is probably one of the reasons why the developers of the *Leap Motion* mentioned to avoid using touchscreen-style interactions[1].

In the second laboratory experiment, the interaction types with virtual hands, with two motion controllers, and the mixed form with one virtual hand and one motion controller were analysed. It turned out that the participants could master the interactions well only with virtual hands. They were surprised by how smoothly and efficiently the system worked. However, some bugs were identified that made interaction complicated, such as insufficient interaction areas, incorrectly placed audio signal feedback and poor feedback behaviour when pressing a 3D button. Compared to the interaction with one motion controller and one hand, the response was even better. This mixture of both techniques was suitable for beginners as well, as the feedback from the participants revealed. The main advantage is that the menu of the user interface can be stabilised with the controller, and interactions can be facilitated. The evaluation of the ASQ in Section 5.8 showed the average satisfaction with a virtual controller, and a virtual hand is better in all three scales. This outcome is consistent with the statements and the actions of the participants. The best solution for users who already have VR experience is to usage two motion controllers to speed up the design and review process. This interaction method allows the user to make changes without having the menu constantly in front of their eyes. The user can access the desired menu item quickly by tapping or clicking gestures. Besides, the user has control at all times over which type of interaction best suits a particular scenario.

Research Question 2

The analysis of the general requirements showed that it might be useful to include stakeholders. In this manner, it is possible to organise different groupings of abilities and tasks better. Only those menu elements are presented to the user which are defined in their role. Inactive elements like teasers and upcoming features are only available for advanced user roles to avoid confusion among beginners.

Initial estimates were made of how often certain menu groups would be called to make the menu structure understandable. Consequently, the main groups located at the root level had to be defined. These are crucial since the arrangement in the menu

[1] https://developer.leapmotion.com/documentation

elements is limited. A maximum of five elements can be displayed on the main level. Otherwise, the remaining elements will disappear behind a scrollable canvas. All further menu elements should be divided into these five groups. The actual name and order have changed during the iterative development process. The naming of all menu elements corresponds to the language of the digital visualizers, which are responsible for the projects and maintenance.

All in all, care was taken to reduce the text, which should draw the user's focus to symbols. This effect made the user interface appear tidy. The interactive elements were large enough to allow smooth interactions with the laser pointer or virtual hands. The status was always visible to the user. A breadcrumb system and a quick retraction to the starting position could be used if problems occurred during the usage with the user interface.

In the first laboratory experiment with prototype A (Part 1), weak spots were found, where the participants could not locate menu positions. In the iterative development process, the error rate could be reduced, as shown in Figure 5.5 (p. 58). At the same time, the task success-rate visualized in Figure 5.4 (p. 57) showed that all tasks could be solved within one minute. The feedback from the participants indicates an intuitive menu structure, where actions could be performed successfully.

Research Question 3

Intuitiveness is different for each user. This circumstance makes it difficult to use an interaction method over and over in the expectation that all users will get along equally well with it. The experiments showed distinct preferences for interactions among the testers.

Some participants had difficulties with touch interactions when using the trackpad because the reaction is very sensitive, and the gross motor skills mentioned by the users can lead to errors very quickly. Touch interactions with the trackpad always require a learning phase. The input concept, how the trackpad works, was recognised in the self-explore phase. Furthermore, the trackpad allows blind navigation, which had excellent feedback, especially during the analysis of different geometries, styles and packages.

The input via laser pointer was easier to handle for most of the participants. This technique requires a shorter training period than performing tasks with the trackpad. Even inexperienced participants were able to cope with the interaction concept. It was recognised that the hands with the motion controllers took very different positions while observing the different participants. Firstly, this was caused by an obstructive geometry (steering wheel) in front of the user. Secondly, it was related to the output ray angle of the laser pointer. An ergonomic emission angle has improved working with the laser pointer. After a short period with this interaction type, the majority were in favour of it. The type of interaction with the laser pointer enables efficient and intuitive work. The concept of *Smart Selection* allows the user to select a geometry with the laser pointer where the trigger executes an action. The content menu adapts automatically to the available options of the geometry. Another click on the element changes the geometry to the next element in the menu list of the user interface. Therefore, there is no need to look at the user interface when an action has to be carried out quickly with the laser pointer technique.

The 3rd interaction technique offers the use of virtual hands. As soon as these are used, a menu above the object is displayed with a direct touch on a geometry. A straightforward 3D user interface allows changes to the selected geometry. This type of interaction is especially suitable for beginners.

All interaction forms were integrated into the interactive system for the intuitive interaction concept. The user can select the appropriate interaction method as desired. Care has been taken to ensure that the menu navigation is consistent in themselves, to allow a seamless transition to another interaction type at any time.

In both experiments, the SUS result showed a grade of *B*. The UEQ in the second experiment was convincing with an overall very good benchmark result with an average rating of *Excellent*. The achieved result is probably because the type of interaction could be freely chosen to fulfil the tasks.

Research Question 4

First drafts of possible prototypes demonstrated that an additional interface for tools selection seemed to be suitable. This feature allows them to be split the complexity into different areas. Firstly, the handling of changes in the environment, vehicle or menu settings is performed in one area of the interactive system. The other area is for tool selection. The most suitable place for selecting tools was at the wrist. The user interfaces are separated in terms of space but still act as an entire unit. This additional feature with two user interfaces is restricted to stakeholders who perform analysis tests.

The implementation of the main user interface was a difficult and very time-consuming process. As a result, the workflows could only be tested in the second experiment. This research showed with think aloud method that new users were overwhelmed with all the additional options. That is why a limitation was deliberately made for inexperienced stakeholders. A summary of the transcription is available in Appendix G (p. 104).

The tools are grouped into workflows. Currently, due to the lack of tools, there are only two workflows in which up to six different tools are arranged. A workflow contains up to maximal eight tools. A tool can be selected via the icon menu of the user interface using a laser pointer. After activation, the geometry of the motion controller is replaced with the geometry of the chosen analyse tool. The menu closes at the end. The user can activate the auto workflow. This state means that clicking on the menu button automatically selects the next tool in the list. This process runs in a loop. It is possible to leave the mode either by deactivating the auto-workflow or by deactivating the active menu item. As soon as the auto-workflow is deactivated, pressing the menu key opens the menu, and the user can continue working in the usual way. Since the menu is intended to work one-handed according to the analysis 3, the user can select a tool from the menu in parallel to this interaction method.

6.2 Hard- and Software Limitations

This Section discusses limitations concerning hard- and software. The first problem concerns the *Leap Motion*. If the user is inside of the vehicle, the distance from most interactive objects is further away than the action radius of the hand recognition device. As a result, the user's head has to move towards the desired object. Otherwise, interactions become ambiguous and can trigger unwanted events.

From an ergonomic point of view, VIVE motion controllers are cumbersome in the usage during the product development in vehicles. The hand posture for a controller is designed in such a way that the controller is kept horizontal in commonly used applications. Due to lack of space, the controller is held vertically in a vehicle. In this pose, it is more challenging to detect buttons and perform actions on the trackpad. Holding the hands up to keep the view on the menu system is increasingly tiring as well.

There are currently three prefabricated materials for UMG that work with a widget: *Normal*, *Opaque* and *Translucent*. All three materials prove to be not perfect during the tryouts. The *Normal* material does not allow transparency at all, the *Opaque* material allows transparency, but the background is completely black. The *Translucent* material handles transparency correctly, but the render quality of the displayed text or image decreases. The widget being displayed is blurred when moved quickly and needs a short time to recover its visibility and readability. The lines and fonts of the translucent material are thinner than the other materials. Furthermore, texts and images should preferably be displayed on a dark background and in a light colour. Otherwise, the lines will be thinned out, which leads to poor legibility.

UMG cannot accept any post-processing effects. Materials applied to fonts and textures cannot contain effects such as glowing. This effects must already be included directly in the textures, which must be taken into account when creating the content.

Lines should have a minimum pixel density in their width. Otherwise, those elements are hardly noticed at a short distance due to the hardware limitations. This finding appeared with a Vive Pro, regardless of the resolution of the content to be displayed. Another HMD with a higher pixel density was not available for further testing.

Many pre-built widget layouts which could be used in a user interface lead to a performance bottleneck. The first attempt of the prototype led to drastic performance losses and thus to lagging, where the entire menu was created in advance. With the current solution, the menu is dynamically created, and the components are entirely reusable.

6.3 Summary

Creating an intuitive user interface is a complicated process because users have different experiences with VR. Therefore it is indispensable to integrate functioning signifiers that guide the users through the experience. A distinctive feedback system helps users to keep track of their status and to interpret the existing operating options accurately. Nevertheless, an intuitive concept requires minimal information on how interactions can take place. Mainly when appointments are limited in time, and there is no time for a self-exploration phase, there should be a brief introduction about interaction possibilities. A navigation concept was introduced that allows the user to navigate blindly to allow one-handed operations with the trackpad of the motion controller. Due to the sensitivity, inexperienced users with trackpad interactions need an extended learning process to get along with this kind of input. The application with the combination of a virtual hand and a motion controller received excellent reactions. The users could take advantage of both interaction techniques by operating the user interface with a natural input with the fingers and the stable attachment of the motion controller menu. It turned out that too much text is very chaotic and confusing. In contrast, a purely icon-based menu leads to many user questions. The aim was to create a combination that would make the user interface concise and straightforward. Right from the start, the prototype was designed to be a modular, easily customizable and expandable menu system. The project administrator can limit the features according to the stakeholder and adapt it to different scenarios, required for a specific product development process with this gained flexibility.

Chapter 7

Conclusion and Future Work

The needs of the prospective users were determined using a user-centred approach, and a prototype with an intuitive interaction concept was developed in this thesis. In an iterative development process, the status of the prototype was to be evaluated in user studies, problem areas identified and subsequently eliminated. The final prototype could be integrated into an existing project.

Three interactive systems were evaluated regarding usability user interfaces heuristics for the *Analysis* in Section 3. The current software solution of BMW and two automotive VR applications accessible to the public were analysed. These findings contributed to the evaluation of suitable interaction techniques. The requirements of the software system were considered during the planning process, and a conceptual model was created. It was necessary to find out whether the user mental model corresponds to the desired ideas of the developed interactive system with user studies.

The first laboratory experiment was carried out as soon as possible until the initial system could be tested. The aim was to get a first impression of how the interactive system felt and to find fundamental weaknesses. The number of participants of five persons proved to be satisfactory, as two experiments could be carried out. The interview was a very time-consuming process. The most important statements were mentioned with the think aloud method. That is the reason why the interview was no longer used in the second experiment. The SUS and UEQ provided good qualitative results for the evaluation, and a proper assessment could be made. Due the small number of participants, no significant results could be achieved. The qualitative and quantitative results finally answered the research questions.

The *Evaluation* in Section 5 revealed the main contribution of this research, an interaction concept and a prototype for the VR product development in the automotive industry. The prototype could be successfully implemented in an existing project at BMW. An intuitive interaction concept is a combination of many components providing a positive experience within an interactive system. The limitations, which were identified based on the analysis of the product requirements, reduced the selection of suitable interaction techniques. The applied interaction patterns in Section 2.4.5 (hand selection pattern, pointing pattern, widgets and panels pattern) result in a good symbiosis, as the experiments in the applied user studies showed.

The developed prototype is an intuitive tool for the analysis process for immersive VR product development in the automotive industry. The seamless integration of the interactive system into existing projects at BMW should contribute to an intuitive interaction concept and facilitate the required time for analyses in VR. Simultaneously, an additional operator can be removed as all operations and tasks can be performed directly by the user in the VE. Furthermore, stakeholders define the range of features available to the user according to the use case. A reduction of the features allows either experienced or inexperienced users to use the interactive system intuitively and efficiently. All these adjustments can be made through a flexible and modular system.

The prototype provides a modular solution for an interactive system that can be equipped with additional features. The region of the user interface where tools and workflows can be selected directly requires further investigations. The findings from the second project and the newly implemented improvements should be reviewed. In the user studies, a small group of subject matter experts were selected. Further experiments should be directed at different stakeholders, and the number of participants should be increased to achieve significant results.

The results of the user studies showed that the input with virtual hands and the user interface received very positive feedback. Further research would be useful in this area according to a problem that arose at the end of the prototype development: How to deal with necessary, multiple function assignments of geometries? In an example with a door handle in the interior, a door animation could be triggered, but the geometry could be changed as well.

The interaction concept was designed in such a way to be able to work within the vehicle in the best possible and intuitive way. No tests were conducted when the user is outside of a vehicle in VR. It would be interesting to observe how users interact with the user interface outside of a vehicle. Since more freedom of movement is available, the user interface could dynamically adapt to this new situation and whether it has a positive effect. For example, the menu context could be automatically adapted to the new environment and the users' needs. Perhaps the separation of the interior and the exterior regarding the vehicle could provide more clarity and increased efficiency.

One aspect to consider in the future would be the integration of speech recognition. As soon as the workplace is no longer tied to a single location and a quiet working environment is available, the use of speech recognition makes sense. This interface would be the most natural form of input as long as an intelligent recognition system can be used. Like a dictation system, recordings can be stored to a specific geometry. This data can be saved as a document and further processed in text form if required. It would be interesting to use an avatar that executes different instructions. In real life, this would come close to a visit to a garage where a mechanic is hired to solve problems. Consideration must be given to the fact that many unique terms and names have to be recognised. A proper solution with extended artificial intelligence has to be developed. In the article from He Zhiyi et al. [14] a solution for speech recognition-based interaction approach for virtual maintenance simulation is described. It presents an abstract mechanism for instructive language and a framework for speech recognition. The processes are comparable to product development, where also different tools for the analysis and operations for geometry changes have to be provided.

Appendix A

DVD Content

Format: DVD+R, Double Layer, 8.5GB

A.1 Master Thesis

Directory: /Thesis/

Friedrich-Bachinger-2019.pdf	Master Thesis.
Transcription.pdf	Transcriptions of interviews and think-aloud protocols

A.2 Project

Directory: /Project/

Media/	Screenshots, movie clips.
Source/	Source files of the Unreal Engine project.
Bin/	Executable prototype.

Appendix B

Abbreviations

- *1D* - One-Dimensional
- *2D* - Two-Dimensional
- *3D* - Three-Dimensional
- *ASQ* - After Scenario Questionnaire
- *BMW* - Bayerische Motoren Werke (Bavarian Motor Works)
- *CAD* - Computer-Aided Design
- *DOF* - Degrees of Freedom
- *DTC* - Design-Technology Convergence
- *HMD* - Head Mounted Display
- *ISO* - International Organization for Standardization
- *ITO* - Idea to Offer
- *PC* - Personal Computer
- *SD* - Standard Deviation
- *SUS* - System Usability Scale
- *UEQ* - User Experience Questionnaire
- *UI* - User Interface
- *UMG* - Unreal Motion Graphics
- *UML* - The Unified Modelling Language
- *VE* - Virtual Environment
- *VR* - Virtual Reality
- *WIMP* - Window, Symbol, Menu, Pointing Device
- *WSRT* - Wilcoxon Signed Ranks Test

Appendix C

Technical Details

C.1 Abstract User Interface

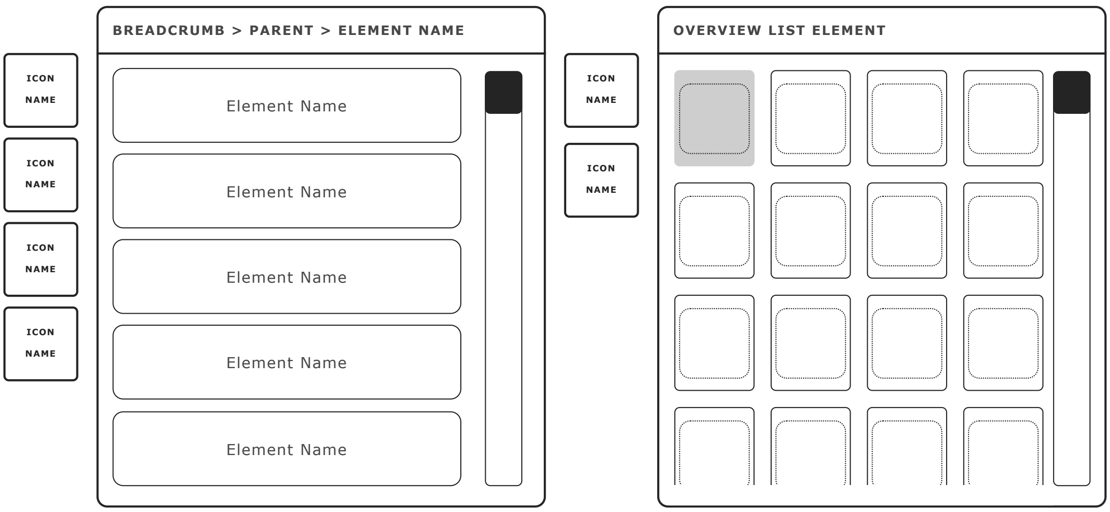


Figure C.1: Abstract representation of the extended user interface.

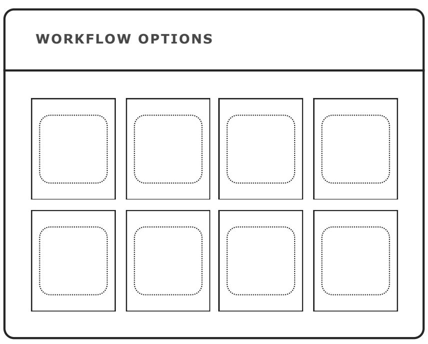


Figure C.2: Abstract representation of the module for quick selections.

C.2 Menu Structure

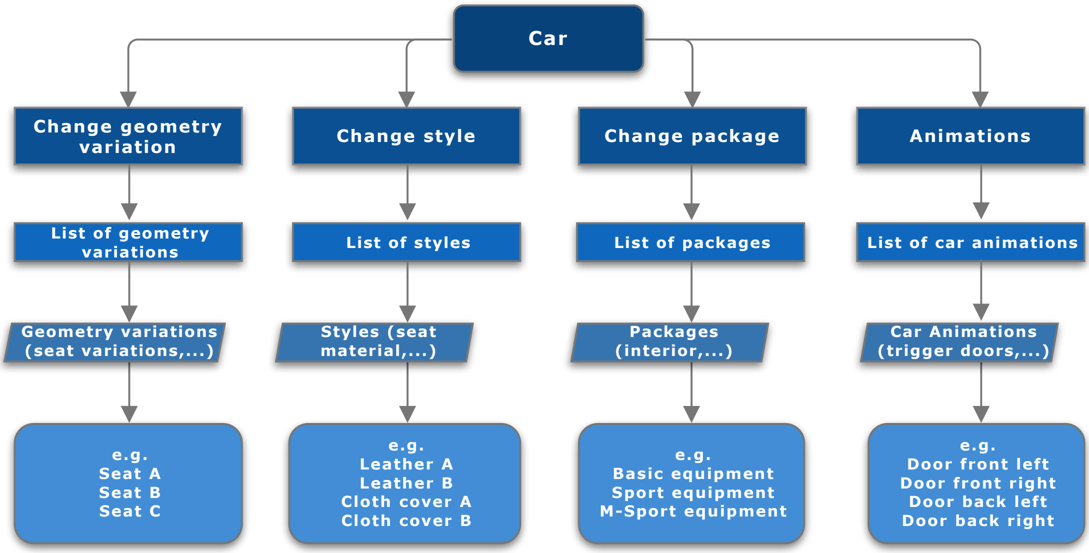


Figure C.3: The menu structure of the car.

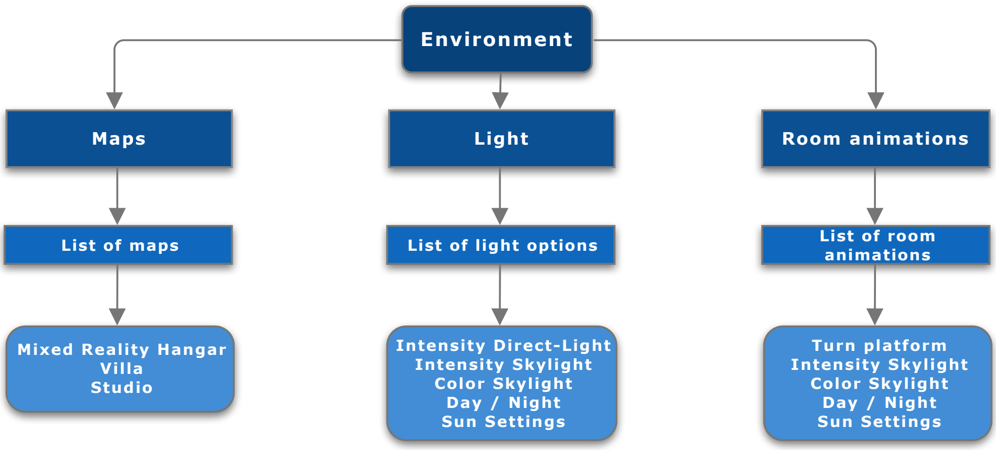


Figure C.4: The menu structure of the environment.

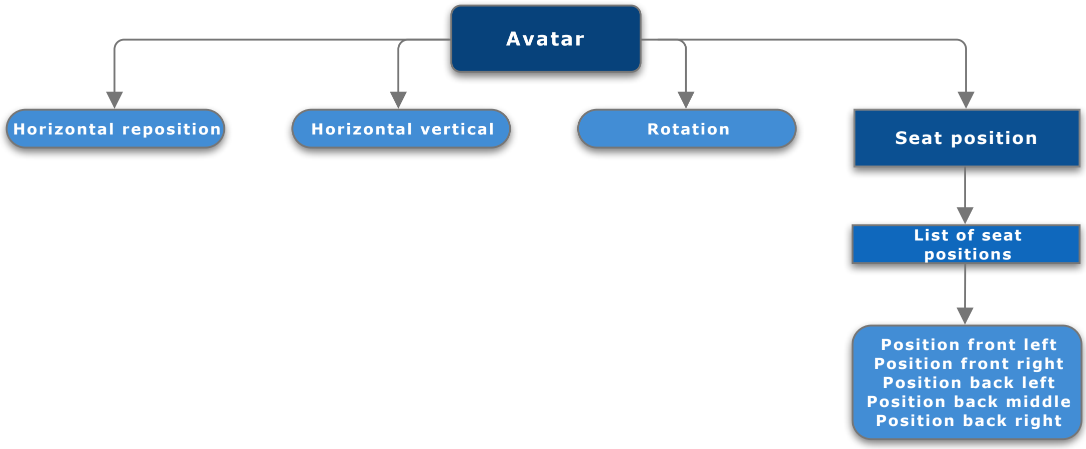


Figure C.5: The menu structure of the avatar.

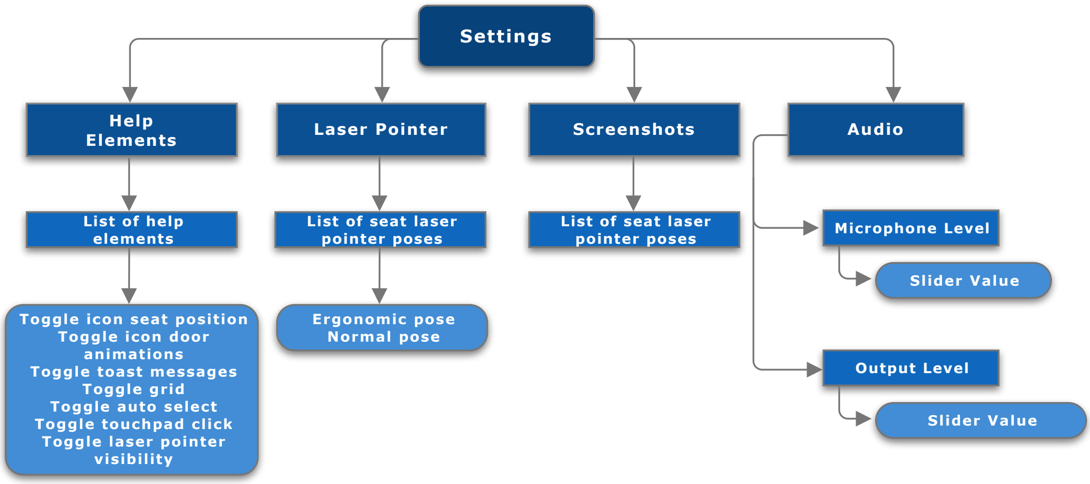


Figure C.6: The menu structure of the settings.

C.3 Architecture Structure

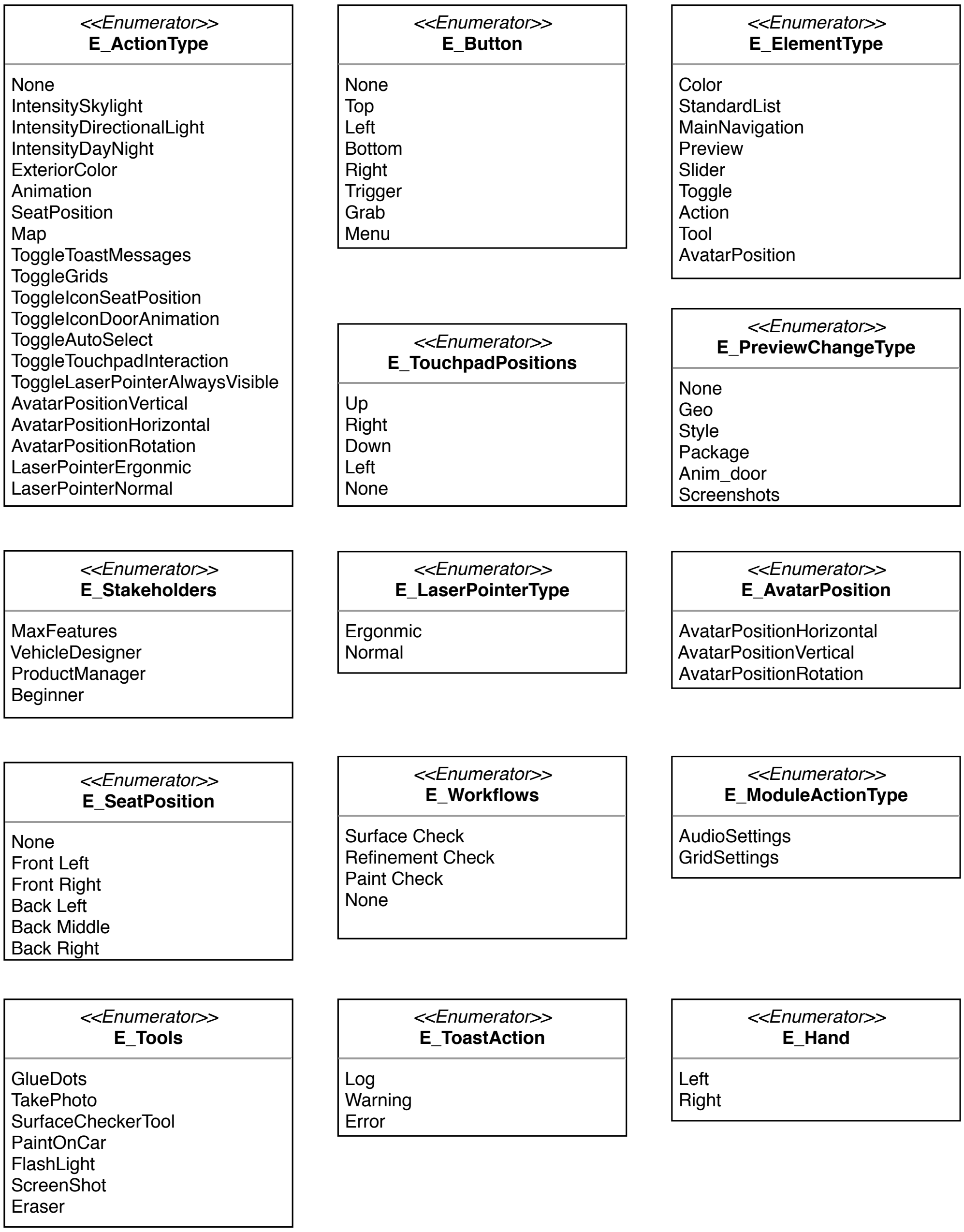


Figure C.7: The architecture of the enumerators.

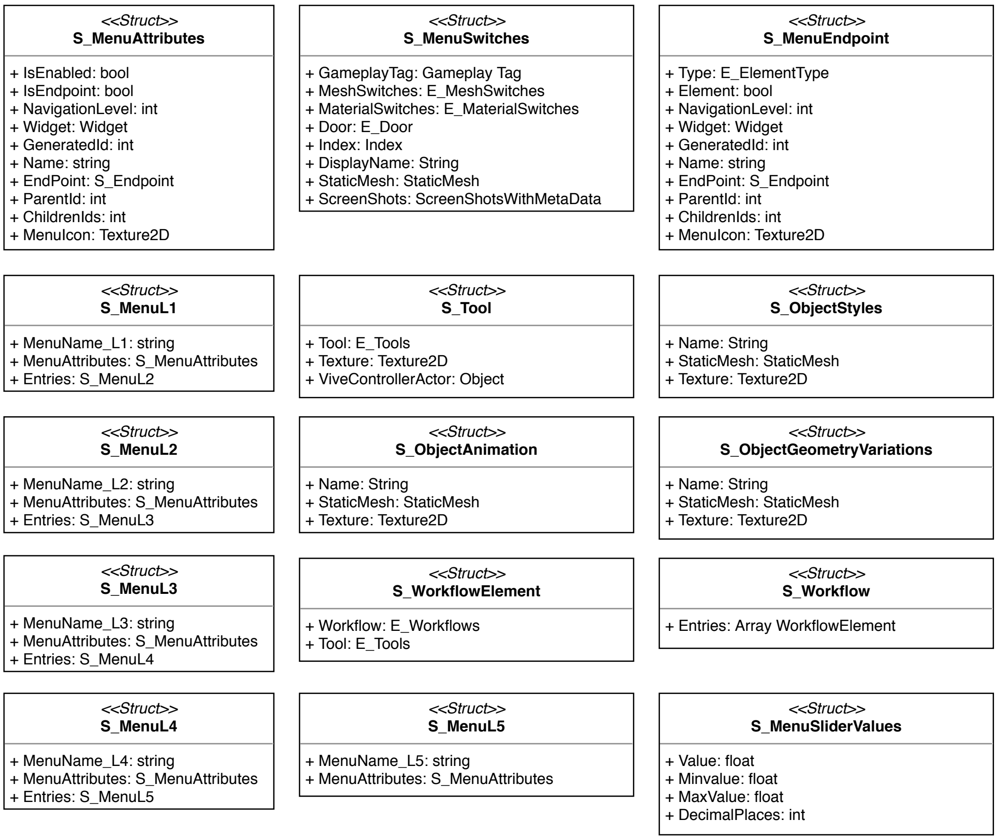
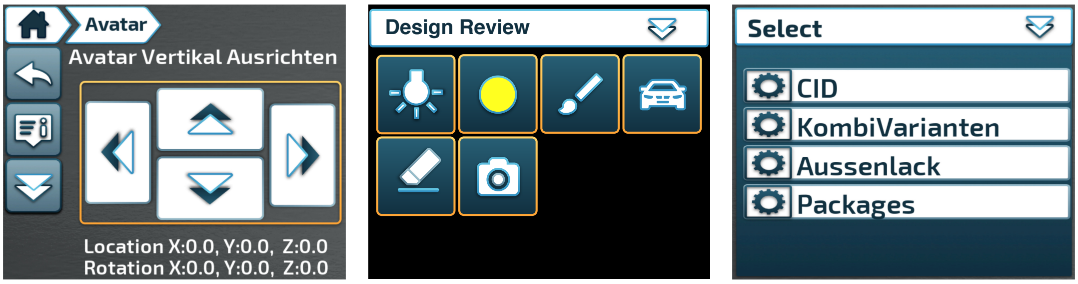


Figure C.8: The architecture of the structs.

C.4 User Interface Components

(a) (b) (c)

Figure C.9: These navigation menu elements and action elements can be found in the main menu. The user has access to different tools which are grouped into workflows. An overview of car customization possibilities is given in the second screenshot. The geometries, styles and packages can be changed, as well as car animations can be selected. The last screenshot shows all door animations available to the user. *Main Menu Workflows* (a), *Car Modifiers* (b), *Door Animation Elements* (c).



(a) (b) (c)

Figure C.10: The user can change the position and the rotating angle of the avatar. This element works with the laser pointer as well with the touchpad. The menu module provides the user quick access to tools which are needed in the analysis process. This user interface works only with the laser pointer because motion controllers are needed anyway for the use of a tool. The last screenshot shows the menu when virtual hands are available. Basically, it is a menu module, but it has additional elements for better interactions when working with virtual hands. *Avatar Position Element* (a), *Menu Module Tools* (b), *Virtual Hand* (c).

(a) (b) (c)

Figure C.11: Different menu elements are displayed in the screenshots where the user can adjustments to specific environment variables or user interface settings. *Avatar Position Element* (a), *Slider Element* (b), *Toggle Element* (c).

Appendix D

Tutorials

D.1 Shortcuts

1. *Key N* - Switch Users
2. *Key U* - Switch Maps

D.2 Change Stakeholder

1. Find the file *BPA_VRStateManagement*, which is located in *Blueprints/Configurator*.
2. Select n the left panel inside *Variables* the *Stakeholder Type*.
3. Change the *Default Value* of the current selected stakeholder in *Details* on the right panel.

D.3 Add New Workflow

1. The first step is to find the file *E_Worflows* in *Common/Enums*. All available workflows are listed at this location. Add a new *Workflow*.
2. Next step is to open the file *BPA_VRStateManagement*, which can be find in *Blueprints/Configurator*.
3. Locate the variable *Workflows* in the left panel *Variables*. Select this variable.
4. On the right panel *Details* a new array can be assigned to the *Default Value Workflows*.
5. Choose the newly created Workflow, and add all required tools. Each new tool needs a *Workflow* and a *Tool*, which are stored in an array as well.

D.4 Add New Tool

1. The first step is to find the file *E_Tools* in *Common/Enums*. All available tools are listed at this location. Add a new *Tool*.

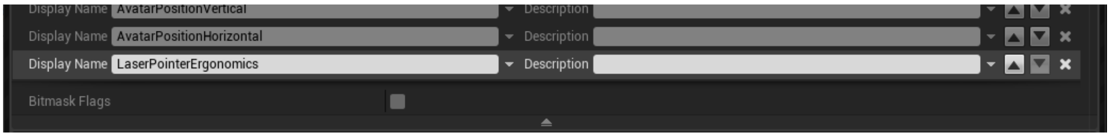


Figure D.1: Extend the enumeration action type and write a new display name.

2. Next step is to open the file *BPA_VRStateManagement*, which can be find in *Blueprints/Configurator*.
3. Locate the variable *Tools* in the left panel *Variables*. Select this variable.
4. On the right panel *Details* a new array can be assigned to the *Default Value Tools*.
5. Add the newly created tool to the list. Every tool consists of the tool enum, textures for the buttons, a texture for the menu and a Blueprint of a *ViveController*.

In case the *ViveController* needs to be stopped manually

6. Find the file *BP_ViveMenuParent* which is located in *Blueprints/Controller/Vive*.
7. Find the function *Stop Active Tool* and add the behaviour to stop the custom *ViveController* Blueprint.

D.5 Add New Menu Component

This tutorial outlines how a new menu item can be integrated into the user interface. The feature to implement allows the user to choose between two different poses of the motion controller. The user can choose between a regular and an ergonomic posture of the laser pointer angle. This tutorial covers these both menu items with the parent menu item. First, the menu item is added to the hierarchy. A range of functions then extends the menu component.

Integration of Menu Item into Hierarchy

1. In figure D.1 the enumeration *E_ActionType* has to be extended with a new position name called *LaserPointerErgonomics* before adding a new component. This enum can be found in *Common/Enums*.
2. Next step is to open the Blueprint *BPA_MenuConfigurator* and search for *Create Menu Item Action Template* and duplicate it.
3. The element has to be renamed afterwards to *Create Menu Item Laser Pointer Ergonomics* and has to be moved into the category *Create Menu Item* by drag and drop.
4. In this step, the individual menu items and the parent menu should be adapted. Changes can be made to the menu item, menu icon, action type and values to be transferred. In this example, fixed values are assigned. An overview is given in D.2. The location of the stored graphical items is in *Misc/Theme/TP_Hollywood/Textures*.

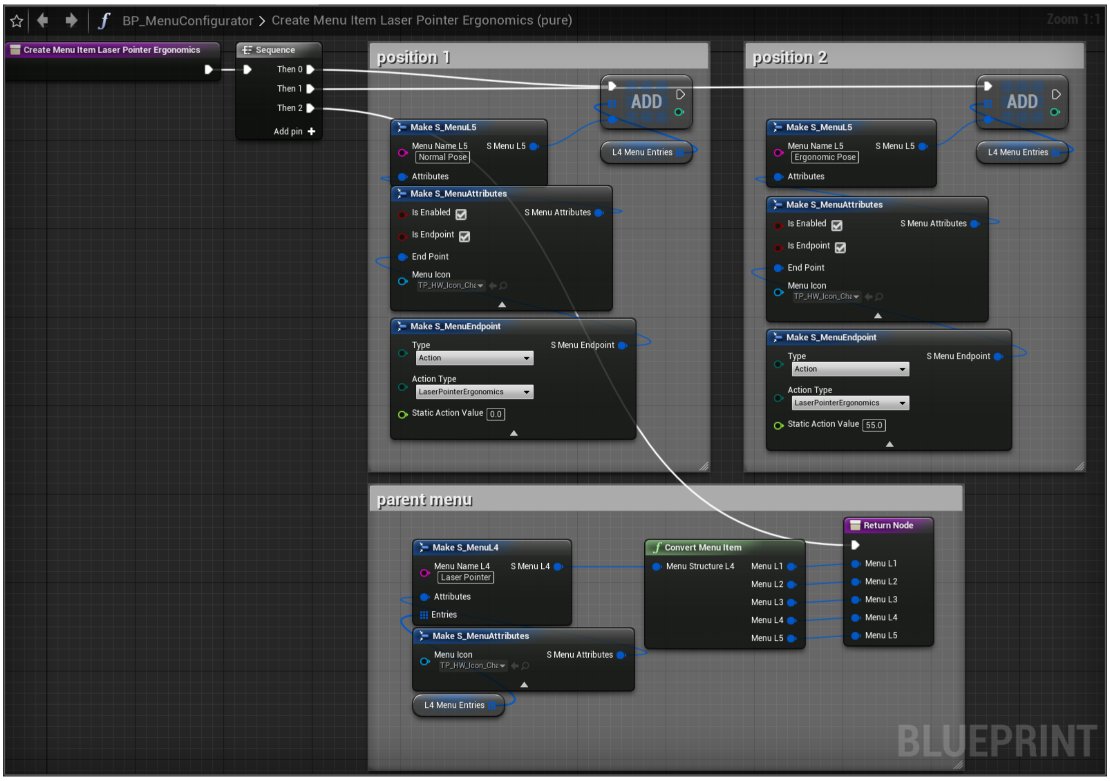


Figure D.2: Create a new function from the template.

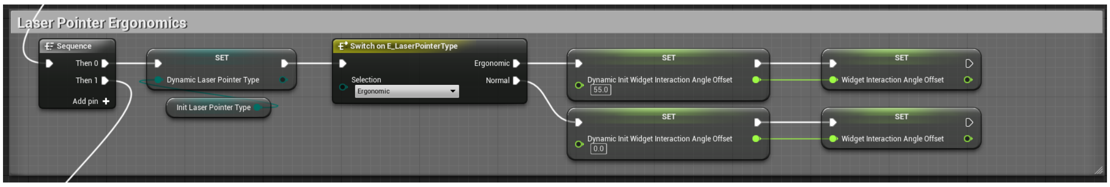


Figure D.3: Init and reseting values for the initialization process has to be set.

5. A float variable must be created at this point, which must be added to the *BPA_VRStateManagement* to define the exit angle of the ray. This variable belongs to the category *Current State* and name it *WidgetInteractionAngleOffset*.
6. Afterwards the value must be assigned during initialization. In the function *Initial Setting* which can be found in *Init* category, the start value must be set. This can be seen in Figure D.3. An additional variable *Dynamic Init WidgetInteractionAngleOffset* stores the value for a later resetting possibility.
7. To reset the menu back to the factory settings, we need to update the *ResetSetting* function. This method can be found in the category *Init* as well. This calls the command to perform the desired manipulation. In this case both *MenuViveControllers* call the function *Update Widget Interaction Angle* and adjust the initial value. This can be seen in Figure D.4.

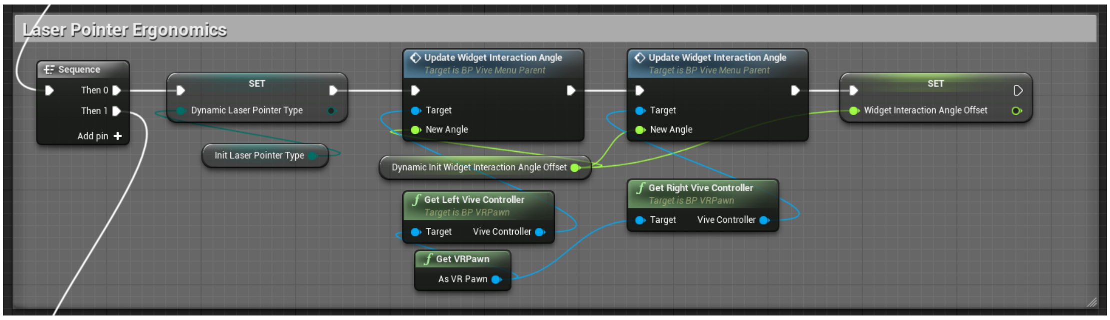


Figure D.4: A function for resetting the menu has to be implemented.

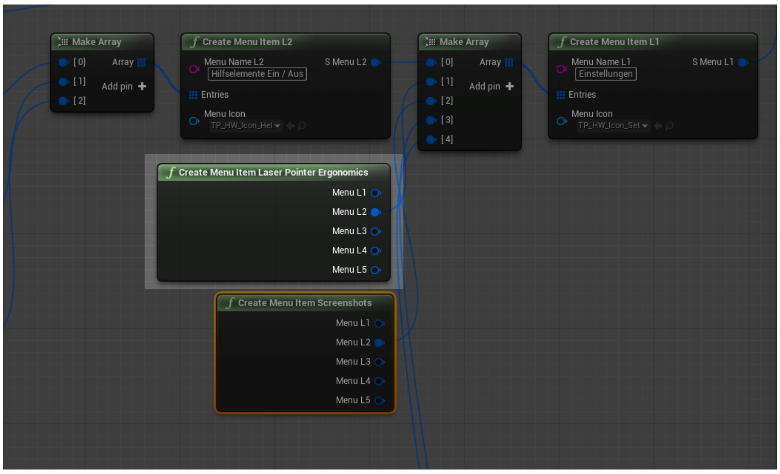


Figure D.5: Integration of the new method and connecting it to a menu array.

8. The function *InitDB* is located in the Blueprint *BPA_MenuConfigurator* which is located in *Blueprints/Configurator*. This method contains the setting options for the individual stakeholder. After a click on *Vehicle Designer* we search for the desired position *Settings* and integrate the created function *Create Menu Item Laser Pointer Ergonomics* with the *Settings* menu. The last step is to make a connection with the array, which can be seen in Figure D.5.

D.6 Extend Menu Component Functionality

1. Find the file *BPA_VRStateManagement*, which is located in *Blueprints/Configurator*. A new method for the desired action has to be created. In this case the function is called *OnToggleLaserPointerErgonomics*. The content of the function can be seen in figure D.6.
2. First, the UserWidget *WB_ActionComponent* is launched, which is located in *Blueprints/Menu/Widgets/Elements*. Then the function *Find Symbol* should be opened. The Symbols can be customized at this place.
3. Next find the function *OnClickElement*. Inside this method a global function

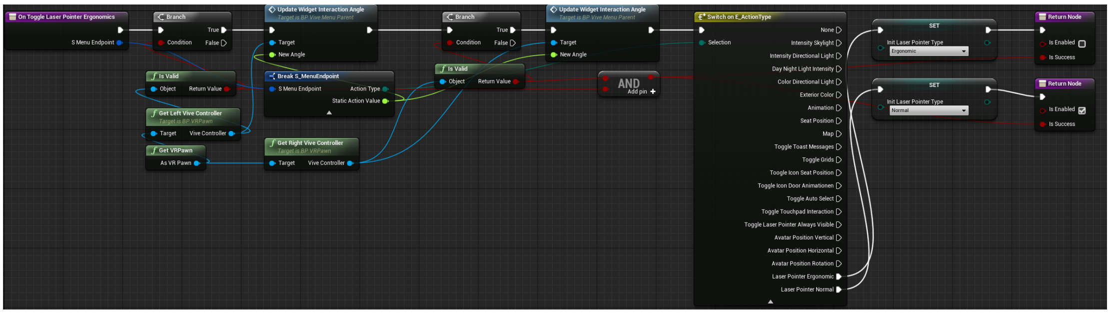


Figure D.6: Add a new action for handling action type endpoint.

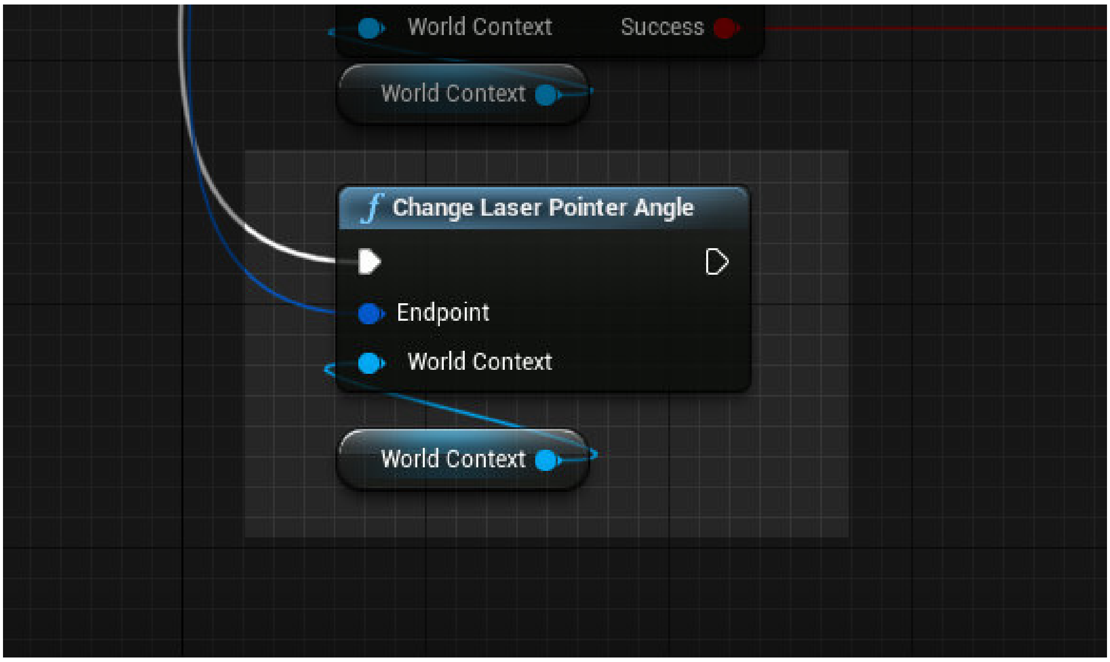


Figure D.7: The menu element has to be extended with some functionality to handle the new endpoint.

OnEndpointAction should be clicked to get all interaction possibilities of the *ActionComponent*. The enum switch has to be extended with the new endpoint of the *BPA_VRStateManagement* Blueprint. This is shown in screenshot D.7.

4. The last step deals with the manipulation process to be carried out. The call is shown in Figure D.8.

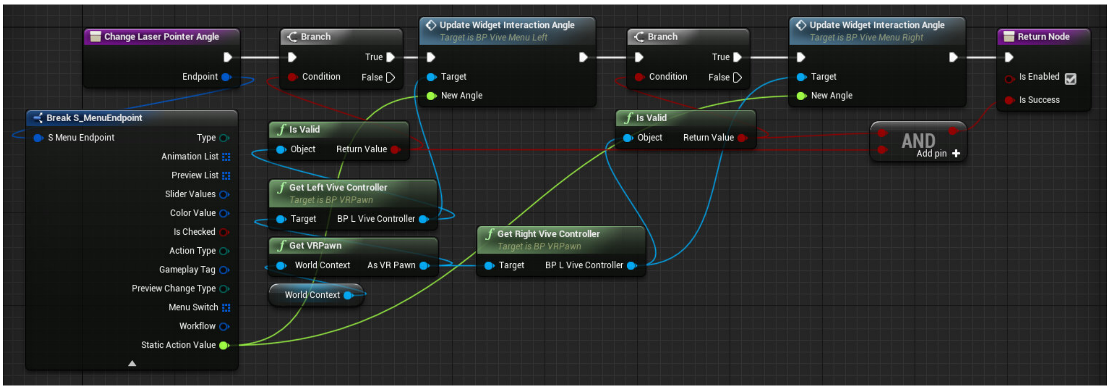


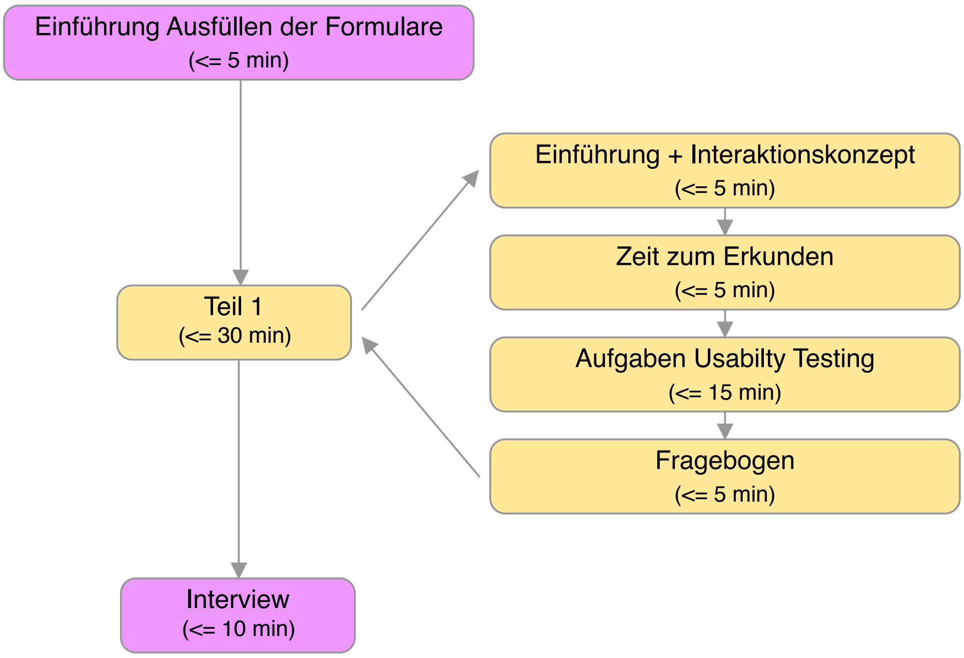
Figure D.8: Overview of the manipulation process to handle the endpoint.

Appendix E

Forms

Allgemeine Informationen

Ziel der Masterarbeit ist, ein intuitives Interaktionskonzept für ein Menüsystem in der Virtual Reality zu entwickeln, mit dem der Benutzer mehrere komplexe Funktionalitäten steuern kann. Verschiedene Benutzergruppen arbeiten mit dem gleichen Menüsystem und sollen in der Lage sein, bestimmte Zustände des Fahrzeugs oder Objekte in VR selbst verändern. Für Tests und Analysen im Design- und Review-Prozess werden spezielle Werkzeuge eingesetzt. Diese Tools und Zusatzfunktionalitäten sollten leicht zugänglich und intuitiv zu bedienen sein, ohne aufwendige Einführungen, Schulungen, Online-Hilfen oder Tutorials.



Zusätzliche Informationen:

Der Tester wird angehalten, während der Interaktion laut mit zu denken (Think Aloud). Bei der Experimentierphase soll die Laser-Pointer Interaktion, sowie auch die Touchpad-Interaktion getestet werden.

Figure E.1: Information handout for lab experiments 1 and 2 (German)

Teilnehmer ID: Verwendung: Datum: **2019 / __ / __**

Einverständniserklärung zum Usability-Test

Bitte lesen und unterschreiben Sie dieses Formular.

Dies ist eine Studie zum Thema Optimierung von Benutzerinteraktionen für Personen, die an dem Virtual-Reality Design-Review-Prozess in der Automobilindustrie involviert sind. Unser Ziel ist es, die Benutzeroberfläche ansprechend, intuitiv und benutzerfreundlich zu gestalten. Ihre Teilnahme wird uns helfen, dieses Ziel zu erreichen.

In diesem Usability-Test:

- Sie werden aufgefordert, bestimmte Aufgaben auf einem Computer auszuführen.
- Sie werden nach den Aufgaben einen Usability Test durchführen.
- Wir werden auch ein Interview mit Ihnen über die von Ihnen durchgeführten Aufgaben führen.

Die Teilnahme an dieser Usability-Studie ist freiwillig. Alle Informationen werden streng vertraulich behandelt. Die Beschreibungen und Ergebnisse können verwendet werden, um den Design- und Überprüfungsprozess in der Virtual Reality zu verbessern. Ihr Name oder eine andere Identifikation wird zu keinem Zeitpunkt verwendet.

Ich bin darüber informiert, dass das Interview im Ton aufgezeichnet wird und bin damit einverstanden.

Wenn Sie nach dem heutigen Tag noch Fragen haben, wenden Sie sich bitte an Friedrich Bachinger unter friedrich.bachinger@gmx.net.

Dies ist ein Test der Benutzerinteraktion und des User Interface. Wir testen nicht Sie. Wir wollen die verwirrenden Faktoren finden, damit wir die Interaktionsmethoden der Benutzer verbessern können. Sie können bei Bedarf Pausen einlegen und Ihre Teilnahme an der Studie jederzeit beenden.

Übersetzt mit www.DeepL.com/Translator

Unterschrift des Probanden Datum

Figure E.2: Usability test consent form in German

Teilnehmer ID:		Verwendung:		Datum:	**2019 / __ / __**

System Usabilty Scale

Instruktionen: Markieren Sie für jede der folgenden Anweisungen ein Kästchen, das Ihre Reaktionen auf das interaktive System am besten beschreibt.

		Stimme gar nicht zu				Stimme voll zu
1.	Ich kann mir sehr gut vorstellen, das System regelmäßig zu nutzen	☐	☐	☐	☐	☐
2.	Ich empfinde das System als unnötig komplex	☐	☐	☐	☐	☐
3.	Ich empfinde das System als einfach zu nutzen	☐	☐	☐	☐	☐
4.	Ich denke, dass ich technischen Support brauchen würde, um das System zu nutzen	☐	☐	☐	☐	☐
5.	Ich finde, dass die verschiedenen Funktionen des Systems gut integriert sind	☐	☐	☐	☐	☐
6.	Ich finde, dass es im System zu viele Inkonsistenzen gibt	☐	☐	☐	☐	☐
7.	Ich kann mir vorstellen, dass die meisten Leute das System schnell zu beherrschen lernen	☐	☐	☐	☐	☐
8	Ich empfinde die Bedienung als sehr umständlich	☐	☐	☐	☐	☐
9	Ich habe mich bei der Nutzung des Systems sehr sicher gefühlt	☐	☐	☐	☐	☐
10	Ich musste eine Menge Dinge lernen, bevor ich mit dem System arbeiten konnte	☐	☐	☐	☐	☐

Bitte geben Sie Kommentare zu diesem System ab:

Dieser Fragebogen ist auf Basis des System Usability Scale (SUS) aufgebaut, der von John Brooke während seiner Arbeit bei Digital Equipment Corporation entwickelt wurde.

Figure E.3: System Usability Scale (SUS) in German

Teilnehmer ID: Verwendung: Datum: **2019 / __ / __**

After Scenario Questionnaire (ASQ)

Instruktionen: Bitte kreuzen Sie immer **eine** Antwort an, auch wenn Sie bei der Einschätzung zu einem Begriffspaar unsicher sind oder finden, dass es nicht so gut zum Produkt passt.

Szenario 1

1. Insgesamt bin ich damit zufrieden, wie leicht die Aufgaben in diesem Szenario zu lösen waren.

stimmt voll zu 1 2 3 4 5 6 7 **stimmt nicht zu**

2. Insgesamt bin ich damit zufrieden, wie viel Zeit ich für die Lösung der Aufgaben in diesem Szenario aufwenden musste.

stimmt voll zu 1 2 3 4 5 6 7 **stimmt nicht zu**

3. Insgesamt bin ich mit den unterstützenden Informationen (Nachrichten, Tooltips) bei der Bearbeitung des Szenarios zufrieden.

stimmt voll zu 1 2 3 4 5 6 7 **stimmt nicht zu**

Szenario 2

1. Insgesamt bin ich damit zufrieden, wie leicht die Aufgaben in diesem Szenario zu lösen waren.

stimmt voll zu 1 2 3 4 5 6 7 **stimmt nicht zu**

2. Insgesamt bin ich damit zufrieden, wie viel Zeit ich für die Lösung der Aufgaben in diesem Szenario aufwenden musste.

stimmt voll zu 1 2 3 4 5 6 7 **stimmt nicht zu**

3. Insgesamt bin ich mit den unterstützenden Informationen (Nachrichten, Tooltips) bei der Bearbeitung des Szenarios zufrieden.

stimmt voll zu 1 2 3 4 5 6 7 **stimmt nicht zu**

Szenario 3

1. Insgesamt bin ich damit zufrieden, wie leicht die Aufgaben in diesem Szenario zu lösen waren.

stimmt voll zu 1 2 3 4 5 6 7 **stimmt nicht zu**

2. Insgesamt bin ich damit zufrieden, wie viel Zeit ich für die Lösung der Aufgaben in diesem Szenario aufwenden musste.

stimmt voll zu 1 2 3 4 5 6 7 **stimmt nicht zu**

3. Insgesamt bin ich mit den unterstützenden Informationen (Nachrichten, Tooltips) bei der Bearbeitung des Szenarios zufrieden.

stimmt voll zu 1 2 3 4 5 6 7 **stimmt nicht zu**

Figure E.4: After Scenario Questionnaire (ASQ) in German Part 1

Teilnehmer ID: ____ Verwendung: ____ Datum: **2019 / __ / __**

After Scenario Questionnaire (ASQ)

Szenario 4

1. Insgesamt bin ich damit zufrieden, wie leicht die Aufgaben in diesem Szenario zu lösen waren.

   stimmt voll zu [1] [2] [3] [4] [5] [6] [7] **stimmt nicht zu**

2. Insgesamt bin ich damit zufrieden, wie viel Zeit ich für die Lösung der Aufgaben in diesem Szenario aufwenden musste.

   stimmt voll zu [1] [2] [3] [4] [5] [6] [7] **stimmt nicht zu**

3. Insgesamt bin ich mit den unterstützenden Informationen (Nachrichten, Tooltips) bei der Bearbeitung des Szenarios zufrieden.

   stimmt voll zu [1] [2] [3] [4] [5] [6] [7] **stimmt nicht zu**

Szenario 5

1. Insgesamt bin ich damit zufrieden, wie leicht die Aufgaben in diesem Szenario zu lösen waren.

   stimmt voll zu [1] [2] [3] [4] [5] [6] [7] **stimmt nicht zu**

2. Insgesamt bin ich damit zufrieden, wie viel Zeit ich für die Lösung der Aufgaben in diesem Szenario aufwenden musste.

   stimmt voll zu [1] [2] [3] [4] [5] [6] [7] **stimmt nicht zu**

3. Insgesamt bin ich mit den unterstützenden Informationen (Nachrichten, Tooltips) bei der Bearbeitung des Szenarios zufrieden.

   stimmt voll zu [1] [2] [3] [4] [5] [6] [7] **stimmt nicht zu**

Szenario 6

1. Insgesamt bin ich damit zufrieden, wie leicht die Aufgaben in diesem Szenario zu lösen waren.

   stimmt voll zu [1] [2] [3] [4] [5] [6] [7] **stimmt nicht zu**

2. Insgesamt bin ich damit zufrieden, wie viel Zeit ich für die Lösung der Aufgaben in diesem Szenario aufwenden musste.

   stimmt voll zu [1] [2] [3] [4] [5] [6] [7] **stimmt nicht zu**

3. Insgesamt bin ich mit den unterstützenden Informationen (Nachrichten, Tooltips) bei der Bearbeitung des Szenarios zufrieden.

   stimmt voll zu [1] [2] [3] [4] [5] [6] [7] **stimmt nicht zu**

Figure E.5: After Scenario Questionnaire (ASQ) in German part 2

Teilnehmer ID: Verwendung: Datum: **2019 / __ / __**

User Experience Questionnaire (UEQ)

Instruktionen: Bitte kreuzen Sie immer **eine** Antwort an, auch wenn Sie bei der Einschätzung zu einem Begriffspaar unsicher sind oder finden, dass es nicht so gut zum Produkt passt.

Entscheiden Sie möglichst **spontan**. Es ist wichtig, dass Sie nicht lange über die Begriffe nachdenken, damit Ihre unmittelbare Einschätzung zum Tragen kommt.

Es gibt keine „richtige" oder „falsche" Antwort. Ihre persönliche Meinung zählt!

		1	2	3	4	5	6	7	
1.	unerfreulich	☐	☐	☐	☐	☐	☐	☐	erfreulich
2.	unverständlich	☐	☐	☐	☐	☐	☐	☐	verständlich
3.	kreativ	☐	☐	☐	☐	☐	☐	☐	phantasielos
4.	leicht zu lernen	☐	☐	☐	☐	☐	☐	☐	schwer zu lernen
5.	wertvoll	☐	☐	☐	☐	☐	☐	☐	minderwertig
6.	langweilig	☐	☐	☐	☐	☐	☐	☐	spannend
7.	uninteressant	☐	☐	☐	☐	☐	☐	☐	interessant
8.	unberechenbar	☐	☐	☐	☐	☐	☐	☐	voraussagbar
9.	schnell	☐	☐	☐	☐	☐	☐	☐	langsam
10.	originell	☐	☐	☐	☐	☐	☐	☐	konventionell
11.	behinderend	☐	☐	☐	☐	☐	☐	☐	unterstützend
12.	gut	☐	☐	☐	☐	☐	☐	☐	schlecht
13.	kompliziert	☐	☐	☐	☐	☐	☐	☐	einfach
14.	abstoßend	☐	☐	☐	☐	☐	☐	☐	anziehend
15.	herkömmlich	☐	☐	☐	☐	☐	☐	☐	neuartig
16.	unangenehm	☐	☐	☐	☐	☐	☐	☐	angenehm
17.	sicher	☐	☐	☐	☐	☐	☐	☐	unsicher
18.	aktivierend	☐	☐	☐	☐	☐	☐	☐	einschläferend
19.	erwartungskonform	☐	☐	☐	☐	☐	☐	☐	nicht erwartungskonform
20.	ineffizient	☐	☐	☐	☐	☐	☐	☐	effizient
21.	übersichtlich	☐	☐	☐	☐	☐	☐	☐	verwirrend
22.	unpragmatisch	☐	☐	☐	☐	☐	☐	☐	pragmatisch
23.	aufgeräumnt	☐	☐	☐	☐	☐	☐	☐	überladen
24.	attaktiv	☐	☐	☐	☐	☐	☐	☐	unattraktiv
25.	sympathisch	☐	☐	☐	☐	☐	☐	☐	unsympathisch
26.	konservativ	☐	☐	☐	☐	☐	☐	☐	innovativ

Figure E.6: User Experience Questionnaire (UEQ) in German

Teilnehmer ID: ______ Verwendung: ______ Datum: **2019 / __ / __**

Demographischer Fragebogen

Ihre Antworten auf die folgenden Fragen helfen bei der Forschungsarbeit, die Testergebnisse zu analysieren.

Tätigkeitsfeld: ____________

1. Geschlecht
   ☐ Mann ☐ Frau ☐ Sonstiges
2. Alter
   ☐ < 29 ☐ 30 - 39 ☐ 40 - 49 ☐ > 50
3. Ausbildung
   ☐ <= A bitur ☐ Bachelor ☐ Master ☐ Doktor
4. Sind Sie Links- oder Rechtshänder?
   ☐ Links ☐ Rechts
5. Sind sie Brillenträger?
   ☐ Ja ☐ Nein
6. Tragen Sie Kontaktlinsen?
   ☐ Ja ☐ Nein
7. Ist eine Farbenblindheit bekannt?
   ☐ Ja ☐ Nein
8. Wie vertraut sind Sie mit Virtual-Reality-Interaktionen?
   ☐ Amateur ☐ Fortgeschritten ☐ Experte
9. Besitzen Sie ein Virtual-Reality-Gerät?
   ☐ Nein ☐ Oculus Rift ☐ HTC Vive ☐ Playstation VR
   ☐ Other

Figure E.7: Demographic questionnaire in German

Appendix F

Experiment Tasks

F.1 Experiment 1 — Tasks

Table F.1: Experiment 1 - Tasks

Enumeration	Task
T 1.1	Start from *Root/Home* and change the current *Exterior Color* of the vehicle via touchpad.
T 1.2	Start from *Root/Home* and toggle the *Helping Grid* in front of the steering wheel to a visible/invisible state via touchpad.
T 1.3	Start from *Root/Home* and change the light intensity of the *Direct-Light* via touchpad.
T 1.4	Start from *Root/Home* and change the state of the current *CID* (Central Information Display) with the laser pointer via the menu.
T 1.5	Start from *Root/Home* and change the colour of the *Direct-Light* via laser pointer.
T 1.6	Start from *Exterior Colour* and detach the menu panel of the current selected preview and attach it on the *Helping Grid*.
T 1.7	Start from *Root/Home* and activate the *Smart-Selection* and change the current *Kombivarianten* via laser pointer.
T 1.8	Start from the *Root/Home* and activate the feature *3D-Preview*. Afterwards change the geometry variant of *Bierdose* via laser pointer.

F.2 Experiment 2 — Tasks

Table F.2: table

Experiment 2 - Tasks

Enumeration	Tas
Scenario 1 (Beginner) - Hand interaction	
T 2.1	Different exterior colours can be selected. Use the menu on your virtual hand to change the current style *Außenlack* to *Rot*.
T 2.2	Two different *Instrumenten-Kombis* exist in the current state. Change the current *Instrumenten-Kombi* to *I-Kombi ECE* in front of the steering wheel.
Scenario 2 (Product Manager) - Hand interaction + one motion controller	
T 2.3	There is a maximum of seven-seat positions to choose from, depending on the vehicle. Take a seat in the rear middle row using the menu system.
T 2.4	The current seating position does not fit precisely. Adjust the position of the avatar to your individual needs.
T 2.5	The current map has several car animations. Open the driver's side *Tür Vorne Links*.
Scenario 3 (Vehicle Designer) - Two motion controllers	
T 2.6	There are three ways to change the current *CID*. Find at least two ways and change the variant of the CID.
T 2.7	The upcoming analysis test will require the vehicle to be outdoors. Find a way to change the current map to *Villa*.
T 2.8	There are two ways to select an analysis tool. Find a way to select the tool *Glue Dots*.
Scenario 4 (Vehicle Designer) - Analysis tools	
T 2.9	There are several *Workflows*, which include analysis tools. Find a way to change the current workflow *Oberflächen Check* to *Refinement Check*.
T 2.10	There are two ways to select an analysis tool. Find a way to select the tool *Glue Dots*.
T 2.11	Currently a automatic selection of the next tool inside the workflow is active. Find a way to inactivate this feature. Activate the tooltips if required.

Appendix G

Summary Transcriptions

G.1 Summary — Lab Experiment 1

Interview and Think-Aloud Results — Part 1

Table G.1: Experiment 1 - Part 1 - Navigation

Code	Summary
E1 Menu Elements	- the list feature is nice and gives a good overview
	- color component location of cancel and save button have logical issues
E4 Menu Structure	- the list feature is nice and gives a good overview
	- *Hilfsgitter* menu element was not found in expected location
	- main menu buttons are not uniquely identifiable
	- difficulties to distinguish between *Szene* and *Raum*
E7 Blind Navigation	- trackpad allows blind navigation
	- understanding of both concepts with laser pointer and trackpad
E12 Icons	- exit possibilities assist during navigation

Table G.2: Experiment 1 - Part 1 - Interaction

Code	Summary
E1 Menu Elements	- consequential errors due missed touch action, value changing very slow
	- difficulties with closing slider menu element
	- interactive area of slider component to small
	- difficulties handling trackpad
E2 Trackpad Sensitivity	- trackpad function seems not to work properly
	- trackpad is too sensitive, menu items can be skipped too easy
	- accidental touch action performed
	- after short learning period trackpad operates efficiently
E6 System Behaviour	- selecting menu elements has difficulties with laser ray
	- show/hide laser ray is not efficient
E8 Execution Time	- not looking at user interface means faster execution time
E10 Interactive System	- user selects a suitable interaction technique depending on the task

Table G.3: Experiment 1 - Part 1 - Design

Code	Summary
E1 Menu Elements	- optimize preview layout and make it more understandable - interactive area of slider component to small - better overview of list element
E12 System Behaviour	- icons size was appropriate, difficulties with icon symbolic - text size was appropriate - icons size was appropriate, icon symbolic appropriate - optimize grouping of icons

Table G.4: Experiment 1 - Part 1 - Ergonomics

Code	Summary
E3 Ergonomics of controller	- uncomfortable hand movement from menu-button to the lower edge of the trackpad - difficulties with left hand, good to have another interaction technique beside trackpad - optimize laser pointer ray angle of motion controller - long stroke of trigger can be disturbing when used frequently - using laser pointer can become strenuous over time - using trackpad is more comfortable over time

Table G.5: Experiment 1 - Part 1 - Feedback

Code	Summary
E5 Interaction Feedback	- optic and haptic feedback are appropriate - interaction buttons signifier (haptic pulsating) not efficient - appropriate feedback when changes happen on display
E9 Execution Time	- outline signifier helps guiding the interaction process - missing signifiers after activating feature

Table G.6: Experiment 1 - Part 1 - Experience

Code	Summary
E5 Interaction Feedback	- handle complexity under professional operating conditions
E6 System Behaviour	- good choosing between trackpad and laser pointer, depending on the action to performed
E10 Interactive System	- interactive system feels good - Leap Motion experience failed in previous tryouts, this interactive system is intuitive - grabbing gets intuitive after understanding the process - laser pointer is more efficient and faster - trackpad interactions can be made one-handed but needs more practice - laser pointer interaction is good - navigating with trackpad is very good and fast

Table G.7: Experiment 1 - Part 1 - Missing Feature

Code	Summary
E1 Menu Elements	- value changing of slider very slow
	- integrate breadcrumbs for better overview
E2 Trackpad Sensitivity	- combine touch and pressing with trackpad
E5 Interaction Feedback	- more clicks on object geometry let user toggle through list
E6 System Behaviour	- missing feature of resetting all menu settings
	- optimize spawn location of detached panel
E11 Gamification	- integrate more smaller gadgets
E14 Interaction with Environment and Vehicle	- integrate more interactions with vehicle
	- more interactions with environment
E15 Interaction Technique	- interaction with eyes could be integrated

Table G.8: Experiment 1 - Part 2 - Navigation

Code	Summary
E1 Menu Elements	- automatic modification of *styles* fits well
E4 Menu Structure	- *Hilfsgitter* was easy to find
	- *Direktlicht* menu entry was easy to find
	- navigation is easy
	- understanding problem of *Licht* and *Sun*, combine menu elements

Table G.9: Experiment 1 - Part 2 - Interaction

Code	Summary
E1 Menu Elements	- problems with confirmation, trackpad is very sensitive
	- toggle on/off toggle-element interaction problem
	- make trigger click action more consistent in menu elements
	- touch interaction problems with confirmation
	- using trigger for confirmation and cancel feels not intuitive, it's more a learning process
E2 Trackpad Sensitivity	- handle controller left hand-side clumsy, trackpad sensitivity is ok, laser-pointer is more convenient
	- touch function at start requires familiarisation
	- with practice the user interface becomes simpler to use
	- pressing is better than touching
	- the trackpad function requires training time
	- prefer right hand when it's possible to avoid interactions with left hand
E8 Execution Time	- beginner could prefer laser pointer, experienced users will use touch functionality as well
E10 Interactive System	- with practice the user interface becomes simpler to use
	- laser pointer interaction is foolproof
	- laser pointer for beginners, experienced users additionally use trackpad
	- smart-selection interaction was intuitive
	- working with laser pointer is convenient
	- there is no interaction technique which would be more suitable at present
E12 Icons	- lateral icons could be bigger

Table G.10: Experiment 1 - Part 2 - Design

Code	Summary
E3 Ergonomics of Controller	- icons have to be learned in beforehand
E10 Interactive System	- user interface has graphically improved - graphically well implemented
E12 Icons	- icons symbolic was understandable - reduction to icons is well implemented

Table G.11: Experiment 1 - Part 2 - Ergonomics

Code	Summary
E3 Ergonomics of controller	- laser beam exit angle should be improved - steering wheel is disturbing, user has to change the hand pose

Table G.12: Experiment 1 - Part 2 - Feedback

Code	Summary
E5 Interaction Feedback	- better and more appropriate feedback
E9	- interaction signifiers not prominent enough - tooltips should be enabled by default

Table G.13: Experiment 1 - Part 2 - Experience

Code	Summary
E6 System Behaviour	- looks tidy, neither too much nor too little
E8 Execution Time	- faster execution time - execution time is almost the same - laser pointer has a faster execution time
E10 Interactive System	- laser pointer interaction is easy - system is more understandable - no menu would be best option - perfect user interface for beginners with few features - using trackpad or laser pointer depends on many factors and what you want to do - choosing laser pointer or trackpad depends on how many list elements are available - *Smart-Selection* is a nice feature
E17 Cognitive Aspect	- initially overloaded due to numerous icons - memorizing sub-items of menu entries

Table G.14: Experiment 1 - Part 2 - Missing Feature

Code	Summary
E1 Menu Elements	- equipping slider component with reset functionality
E6 System Behaviour	- tooltips shoud be activted from start
E10 Interactive System	- touch functionality with fingers is be better then with laser pointer interaction - lateral icons should be accessible via trackpad

G.2 Summary — Lab Experiment 2

Interview and Think-Aloud Results

Table G.15: Experiment 2 - Navigation

Code	Summary
E1 Menu Elements	- navigation problem with leaving area in menu for virtual hands

Table G.16: Experiment 2 - Interaction

Code	Summary
E1 Menu Elements	- optimize spawn point of 3D menu above an object - 3D button fires sometimes two times - problems with execution via motion controller - problems with leaving avatar position element
E2 Trackpad Sensitivity	- trackpad click would be better then touching - trackpad click is better then touching - grabbing point of object could be optimized
E10 Interactive System	- optimize spawn point of 3D menu above an object
E15 Interactive Technique	- Leap Motion works inaccurate with virtual hand menu - Leap Motion works very well with 3D menu - showing menu with virutal hands needs angle optimisation - difficulties to fulfill an action with fingers on menu with virtual hands

Table G.17: Experiment 2 - Design

Code	Summary
E1 Menu Elements	- workflow button could be bigger for better interaction in tool menu (virtual hands) - leaving element unintentionally triggers an action
E5 Interaction Feedback	- optimize visual menu button feedback
E12 Icons	- menu icon animation in tool menu don’t works as expected - interactive areas on menu for virtual hands are to small - optimize workflow icon style for menu-tools - optimize rotation icon style for menu-tools

Table G.18: Experiment 2 - Ergonomics

Code	Summary
E3 Controller	- ergonomic pose is better
E10 Interactive System	- optimisation between offset hand and virtual hands menu
E15 Interaction Technique	- no ergonomic hand position because I always have to hold it up high

Table G.19: Experiment 2 - Feedback

Code	Summary
E1 Menu Elements	- trigger should fire an event when click sound occurs
E5 Interaction Feedback	- wrong audio feedback and followed by wrong misplaced action event
E14 Interaction with Env. and Car	- interactive objects need additional signifier when working with Leap Motion

Table G.20: Experiment 2 - Experience

Code	Summary
E6 System Behaviour	- interactions in system work as expected - both controllers act as one unit - good to have more interaction possibilities - Auto Workflow is not intuitive and should be improved
E10 Interactive System	- input problems because of steering wheel - the menu system overwhelms the user with all available features - reduced menu would be better - menu works quite well - menu with four extra displays is very cool
E15 Interaction Technique	- controller menu combined with virtual hand is better than virtual hands - trackpad click is better then a touch - working with two motion controller is the best solution - grabbing object with the right side is confusing

Table G.21: Experiment 2 - Missing Feature

Code	Summary
E1 Menu Elements	- missing close button for menu for virtual hands
E2 Trackpad Sensitivity	- expected the day/night is a toggle element
E3 Ergonomics	- ergonomics should be default
E5 Interaction Feedback	- outline behaviour should be optimized when hovering an object
E6 System Behaviour	- missing stepsize for avatar position - laser pointer on/off would could improve system - bigger collider of 3D widgets
E10 Interactive System	- trackpad should also be available for the tools in module

Table G.22: Experiment 2 - Error

Code	Summary
E5 Interaction Feedback	- missing the tooltips right hand

References

Literature

[1] Gerald Albaum. “The Likert Scale Revisited: An Alternate Version”. *International Journal of Market Research* 39 (Jan. 1997), pp. 331–348 (cit. on p. 51).

[2] Michael Albers and Brian Still. *Usability of Complex Systems: Evaluation of User Interaction*. CRC Press, 2011 (cit. on p. 7).

[3] David Benyon. *Designing Interactive Systems: A Comprehensive Guide to HCI, UX and Interaction Design*. Pearson, 2013 (cit. on p. 8).

[4] Leif P. Berg and Judy M. Vance. “An Industry Case Study: Investigating Early Design Decision Making in Virtual Reality”. *Journal of Computing and Information Science in Engineering* 17.1 (Nov. 2016), pp. 011001-1–011001-7. URL: https://doi.org/10.1115/1.4034267 (cit. on p. 5).

[5] Doug Bowman et al. *3D User Interfaces: Theory and Practice*. Addison-Wesley, 2017 (cit. on pp. 8, 10, 11, 14–17).

[6] John Brooke. “SUS: A Quick and Dirty Usability Scale”. In: *Usability Evaluation in Industry*. Taylor and Francis, London, Nov. 1996, pp. 189–194 (cit. on p. 48).

[7] John Brooke. “SUS: A Retrospective”. *Journal of Usability Studies* 8 (Jan. 2013), pp. 29–40 (cit. on p. 50).

[8] John M. Carroll and John C. Thomas. “Metaphor and the Cognitive Representation of Computing Systems”. *IEEE Transactions on Systems, Man, and Cybernetics* 12.2 (1982), pp. 107–116 (cit. on p. 11).

[9] Alan Cooper, Robert Reimann, and David Cronin. *About Face 3: The Essentials of Interaction Design*. John Wiley & Sons, 2010 (cit. on pp. 11, 19, 48, 55, 64).

[10] Raimund Dachselt and Michael Hinz. “Three-Dimensional Widgets Revisited - Towards Future Standardization”. *IEEE VR 2005 Workshop New Directions in 3D User Interfaces* 16 (Mar. 2005), pp. 89–92 (cit. on p. 10).

[11] Raimund Dachselt and Anett Hübner. “Three-Dimensional Menus: A Survey and Taxonomy”. *Computers & Graphics* 31.1 (2007), pp. 53–65 (cit. on p. 10).

[12] John D. Gould and Clayton Lewis. “Designing for Usability: Key Principles and what Designers Think”. *Communications of the ACM* 28.3 (1985), pp. 300–311 (cit. on p. 12).

[13] Lisa Graham. “Gestalt Theory in Interactive Media Design”. *Journal of Humanities and Social Sciences* 2.1 (2008), pp. 1–12 (cit. on p. 42).

[14] Zhiyi He et al. “A Speech Recognition-Based Interaction Approach Applying to Immersive Virtual Maintenance Simulation”. In: *2017 Second International Conference on Reliability Systems Engineering (ICRSE)*. IEEE. 2017, pp. 1–5 (cit. on p. 78).

[15] Andreas Holzinger. “Usability Engineering Methods For Software Developers”. *Commun. ACM* 48 (Jan. 2005), pp. 71–74 (cit. on pp. 6–8).

[16] Riitta Jääskeläinen. “Think-aloud protocol”. In: *Handbook of Translation Studies.* John Benjamins Publishing Company Amsterdam, The Netherlands, Jan. 2010, pp. 371–373 (cit. on p. 8).

[17] Richard H. Jacoby and Stephen R. Ellis. “Using Virtual Menus in a Virtual Environment”. In: *Visual Data Interpretation.* Vol. 1668. International Society for Optics and Photonics. 1992, pp. 39–49 (cit. on pp. 9, 10).

[18] Jason Jerald. *The VR Book: Human-Centered Design for Virtual Reality.* Morgan & Claypool, 2015 (cit. on pp. 1, 5, 6, 8, 11, 14–17).

[19] Jeff Johnson and Austin Henderson. “Conceptual Models: Begin by Designing What to Design”. *Interactions* 9.1 (2002), pp. 25–32 (cit. on pp. 13, 14).

[20] Namgyu Kim et al. “Multimodal Menu Presentation and Selection in Immersive Virtual Environments”. In: *Proceedings of the IEEE Virtual Reality 2000 Conference.* VR ’00. Washington, DC, USA: IEEE Computer Society, 2000, pp. 281– (cit. on pp. 9, 10).

[21] Geraldine B King. “Open & Closed Questions: The Reference Interview”. *RQ* 12 (Jan. 1972), pp. 157–160 (cit. on p. 51).

[22] Kurt Koffka. *Principles of Gestalt Psychology.* Routledge, 2013 (cit. on p. 42).

[23] Thomas K. Landauer, Prasad V. Prabhu, and Martin G. Helander. *Handbook of Human-Computer Interaction.* Elsevier Science Inc., 1997 (cit. on p. 11).

[24] Johanna Lee. *Learning Unreal Engine Game Development.* Packt Publishing, 2016 (cit. on p. 33).

[25] Clayton Lewis et al. “Testing a Walkthrough Methodology for Theory-Based Design of Walk-Up-End-Use Interfaces”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* ACM. 1990, pp. 235–242 (cit. on p. 8).

[26] James R. Lewis. “Psychometric Evaluation of an After-Scenario Questionnaire for Computer Usability Studies: The ASQ”. *SIGCHI Bulletin* 23.1 (Jan. 1991), pp. 78–81 (cit. on p. 51).

[27] James R. Lewis. “The System Usability Scale: Past, Present, and Future”. *International Journal of Human-Computer Interaction* 34.7 (2018), pp. 577–590 (cit. on p. 49).

[28] Jiandong Liang and Mark Green. “JDCAD: A Highly Interactive 3D Modeling System”. *Computers & Graphics* 18.4 (1994), pp. 499–506 (cit. on p. 11).

[29] Mitch McCaffrey. *Unreal Engine VR Cookbook: Developing Virtual Reality with UE4*. Addison-Wesley Professional, 2017 (cit. on p. 33).

[30] Rabia Murtza, Stephen Monroe, and Robert J. Youmans. "Heuristic Evaluation for Virtual Reality Systems". In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. Vol. 61. 1. SAGE Publications Sage CA: Los Angeles, CA. 2017, pp. 2067–2071 (cit. on p. 8).

[31] D.A. Norman. "Cognitive Engineering". *Cognitive Engineering* (Jan. 1986), pp. 31–61 (cit. on p. 12).

[32] Jenny Preece, Yvonne Rogers, and Helen Sharp. *Interaction Design: Beyond Human-Computer Interaction*. John Wiley & Sons, 2015 (cit. on pp. 7, 11–14, 17, 51).

[33] Bernhard Preim and Raimund Dachselt. *Interaktive Systeme: Band 2: User Interface Engineering, 3D-Interaktion, Natural User Interfaces*. Springer-Verlag, 2015 (cit. on pp. 10, 11, 19, 48, 49).

[34] Martin H. Rademacher. *Virtual Reality in der Produktentwicklung: Instrumentarium zur Bewertung der Einsatzmöglichkeiten am Beispiel der Automobilindustrie*. Springer-Verlag, 2014 (cit. on pp. 4, 5).

[35] James Rumbaugh, Ivar Jacobson, and Grady Booch. *The Unified Modeling Language Reference Manual*. Pearson Higher Education, 2004 (cit. on p. 19).

[36] Martin Schrepp, Andreas Hinderks, and Joerg Thomaschewski. "Applying the User Experience Questionnaire (UEQ) in Different Evaluation Scenarios". In: *International Conference of Design, User Experience, and Usability*. Springer. 2014, pp. 383–392 (cit. on pp. 48, 50).

[37] International Organization For Standardization. *ISO 9241-11 - Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs): Part 11: Guidance on Usability*. Berlin: ISO, 1998 (cit. on p. 6).

[38] Frank Steinicke. *Being Really Virtual*. Springer, 2016 (cit. on p. 6).

[39] Alistair Sutcliffe and Brian Gault. "Heuristic Evaluation of Virtual Reality Applications". *Interacting with computers* 16.4 (2004), pp. 831–849 (cit. on p. 8).

[40] Ivan E. Sutherland. "A Head-Mounted Three Dimensional Display". *FJCC*. AFIPS '68 (Fall, part I) 33 (Jan. 1969), pp. 757–764 (cit. on p. 6).

Online sources

[41] *Advanced VR Framework*. Mar. 2019. URL: https://www.unrealengine.com/marketplace/en-US/slug/advanced-vr-framework (visited on 09/17/2019) (cit. on p. 15).

[42] Jerry Beilinson. *Palmer Luckey and the Virtual Reality Resurrection*. 2014. URL: https://www.popularmechanics.com/technology/gadgets/a12956/palmer-luckey-and-the-virtual-reality-resurrection-16834760 (visited on 01/11/2019) (cit. on p. 6).

[43] Relay Cars. *Relay Cars*. 2018. URL: https://www.relaycars.com (visited on 03/31/2019) (cit. on p. 29).

[44] Joe Connolly. *VR Design Review Nr 2 - Menus*. Dec. 2016. URL: https://blog.sketchbox3d.com/vr-design-review-2-menus-b0d7ddc3078 (visited on 09/19/2019) (cit. on p. 15).

[45] Interaction Design Foundation. *What is Interaction Design?* 2019. URL: https://www.interaction-design.org/literature/article/what-is-interaction-design (visited on 04/01/2019) (cit. on p. 12).

[46] Interaction Design Foundation. *What is Usability?* 2019. URL: https://www.interaction-design.org/literature/topics/usability (visited on 03/28/2019) (cit. on p. 6).

[47] Wolfgang Herrmann. *Digitale Transformation der BMW Group: Wir lösen uns von starren Hierarchien*. 2017. URL: https://www.computerwoche.de/a/wir-loesen-uns-von-starren-hierarchien,3330082,3 (visited on 01/11/2019) (cit. on p. 18).

[48] Benjamin Hillmann. *Budget Cuts im Test: VR-Stealth-Action mit vollem Körpereinsatz*. 2018. URL: https://threesixtyreality.co.uk/interaction-patterns/teleport-specific-location (visited on 04/02/2019) (cit. on p. 16).

[49] Holographic Interfaces. *Multidimensional Data Display and Manipulation in a Spatial Interface*. July 2017. URL: https://medium.com/@HolographicInterfaces/multidimdata-857a7563001a (visited on 09/17/2019) (cit. on p. 15).

[50] Zach Kinster. *Hovercast VR Menu: Power at Your Fingertips*. Feb. 2015. URL: http://blog.leapmotion.com/hovercast-vr-menu-power-fingertips (visited on 09/17/2019) (cit. on p. 15).

[51] Jakob Nielsen. *10 Usability Heuristics for User Interface Design*. 1994. URL: https://www.nngroup.com/articles/ten-usability-heuristics/ (visited on 09/16/2019) (cit. on pp. 7, 9).

[52] Jakob Nielsen. *Why you only need to test with 5 users*. 2000. URL: https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/ (cit. on p. 48).

[53] Merriam-Webster Online. *Merriam-Webster Online Dictionary*. 2019. URL: https://www.merriam-webster.com/dictionary/virtualreality (visited on 03/11/2019) (cit. on p. 5).

[54] BMW press. *BMW opts to incorporate HTC Vive VR headsets and mixed reality into the development of new vehicle models*. 2016. URL: https://www.press.bmwgroup.com/global/article/detail/T0258902EN/bmw-opts-to-incorporate-htc-vive-vr-headsets-and-mixed-reality-into-the-development-of-new-vehicle-models-computer-images-instead-of-laboriously-constructed-draft-models:-greater-flexibility-faster-results-and-lower-costs?language=en (visited on 03/28/2019) (cit. on p. 5).

[55] BMW press. *Designentwicklung bei der BMW Group am Beispiel der neuen BMW 5er Limousine*. 2009. URL: https://www.7-forum.com/news/Designentwicklung-bei-der-BMW-Group-am-B-3088.html (visited on 07/16/2019) (cit. on p. 18).

[56] Boston University School of Public Health. *Wilcoxon Signed Rank Test*. 1994. URL: http://sphweb.bumc.bu.edu/otlt/MPH-Modules/BS/BS704_Nonparametric/BS704_Nonparametric6.html (visited on 09/18/2019) (cit. on p. 56).

[57] Rob Stumpf. *BMW Uses Virtual Reality to Build Prototypes*. 2017. URL: https://www.thedrive.com/sheetmetal/8619/bmw-uses-virtual-reality-to-build-prototypes (visited on 03/28/2019) (cit. on pp. 1, 4).

[58] *Teleport Pattern with Motion Controllers*. June 2019. URL: https://www.gamez.de/reviews/budget-cuts-review (visited on 09/17/2019) (cit. on p. 15).

[59] Threesixty. *Targeting with gaze*. 2018. URL: https://threesixtyreality.co.uk/interaction-patterns/targeting-with-gaze (visited on 04/02/2019) (cit. on p. 16).

[60] Bayerische Motoren Werke. *Car Design: in 7 Schritten zum Auto von morgen*. 2019. URL: https://www.bmw.com/de/design/car-design-in-7-schritten.html (visited on 05/20/2019) (cit. on p. 5).

[61] Bayerische Motoren Werke. *The design process at the BMW Group*. 2012. URL: https://www.carbodydesign.com/2012/03/the-design-process-at-the-bmw-group/ (visited on 05/20/2019) (cit. on p. 4).

Check Final Print Size

— Check final print size! —

width = 100mm
height = 50mm

— Remove this page after printing! —