

---

## PROJECT 2

*Human-Computer Interaction*

---

Aalborg University

Ida Taglioni, Janis Ludvigs Berzins-Berzitis, Jonas Litvinas  
Malte Juel Breddam, Morten Porsing, Laurynas Lubys

This is where the date is supposed to be  
CHANGE THIS BEFORE HANDIN

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Initial Problem Statement . . . . .	2
<b>2</b>	<b>Analysis</b>	<b>3</b>
2.1	Target Group . . . . .	3
2.1.1	Psychographics . . . . .	3
2.1.2	Target group conclusion . . . . .	4
2.2	User Experience Strategy . . . . .	5
2.2.1	Introduction to user experience . . . . .	6
2.2.2	Intuitiveness is familiarity . . . . .	8
2.2.3	Designing intuitively . . . . .	10
2.2.4	Usability . . . . .	11
2.3	Interaction Methods in 3d Environment . . . . .	14
2.3.1	Traditional interaction . . . . .	14
2.3.2	Sensors . . . . .	15
2.3.3	Conclusion . . . . .	17
2.4	Graphical Design . . . . .	17
2.5	State of the Art . . . . .	23
2.5.1	Ikea Kitchen Planner . . . . .	23
2.5.2	HomeDesign3D . . . . .	24
2.5.3	AutoDesk's HomeStyler . . . . .	25
2.5.4	Planner5D . . . . .	25
2.5.5	Conclusion . . . . .	26
2.6	Design Requirements . . . . .	26

2.7	Design Requirements . . . . .	26
2.7.1	Immediacy and Simplicity . . . . .	28
2.7.2	Graphical element sizes and placement . . . . .	28
2.8	Conclusion . . . . .	28
2.8.1	Final problem statement . . . . .	28
<b>3</b>	<b>Design</b>	<b>29</b>
3.0.2	<for mobiles> . . . . .	29
<b>4</b>	<b>Implementation</b>	<b>31</b>
4.0.3	Implementation of 3D testing area . . . . .	31
4.0.4	Unity Implementation . . . . .	33
4.0.5	ButtonScript . . . . .	33
4.0.6	Buttons Scene . . . . .	33
4.0.7	Gyroscope Camera . . . . .	35
4.0.8	Joystick Scene . . . . .	36

# 1 | Introduction

As the technology becomes more advanced, our expectations of the technology increase as well. Microprocessors have become increasingly faster and smaller and according to Moore's law, the improvements will continue. We're now at the point where you can actually fit a computer which 10 years ago would be considered state of the art into your pocket and make it affordable for the general population. As the size of the hardware decreases, so does the physical limitation of the devices. There is now room for specialized sensors in the average smartphone, things like accelerometers, magnetometers and gyroscope along with support for multitouch, all of which allow the user to interact with their device in new and non-traditional ways. Most applications however do not employ these sensors and as such a lot of users are not even aware of the possibilities they provide. By using these non-traditional interaction methods one can strive to improve the user's experience by challenging them to use their devices in new and exciting ways. When the users have their preconceptions challenged a state of flow is established and even tedious tasks may seem fun. This report will explore some of these non-traditional interaction methods, specifically when it comes to navigation in a 3d environment. These methods will be explored by creating an environment using Autodesk Maya, Adobe Photoshop and Unity3d where the users will have a chance to try out various means of navigating through the three dimensional world, in an attempt to determine the most efficient and most engaging way of controls.

## 1.1 INITIAL PROBLEM STATEMENT

How can we improve user experience when navigating a 3d space using non-traditional mobile sensors?

# **2 | Analysis**

## **2.1 TARGET GROUP**

Profound understanding of the target group helps in the process of creating a useful concept. It will be an answer to the user's specific needs and wishes. Dividing the target group into different group categorizations - segments, gives a possibility to understand the chosen subjects deeper and leads to a better concept design. Created prototype will be specifically designed for a segmented and generalized group of people of this project. Connecting with target group is essential since the product will be used by actual users. In most cases it is not enough to generalize a group of people from surface observations or presumed stereotypes, since sometimes people do not act as they speak and their actual needs can vary a lot.

However since this project is targeting people who might need to use non-traditional interaction in 3D environment on their mobile devices it can be used in many fields as video games, simulations, localization - simulative approach of viewing architectural objects, city and landscape areas as in "Google maps" etc, controlling something that requires remote usage as robots, planes etc. So target group can also be very wide in all perspectives. it means that target group is not generalizable and it does not require deep analysis.

There are many ways to segment a target audience. Probably the most popular is Geographic or Demographic segmentation. However it is most relevant to analyse psychographics as geographics and demographics would give too big range in age, geographical position etc. this kind of data would not be conclusive.

### **2.1.1 PSYCHOGRAPHICS**

Psychographic generalization segments target group according social class, lifestyle and personality characteristics. [?] It is important and relevant to understand the customer's needs,

their habits and personality since it can partially answer how the app's concept can be developed.

### DIGITAL KNOWLEDGE

Marc Prensky in his "Digital Natives, Digital Immigrants" [?] article categorizes his understanding of target group into two segments, when it comes to understanding or learning with digital technology. He categorizes them as "digital natives" and "digital immigrants". There are few more categorizations that people use, such as "Born digital" or "Digital Settlers", so it is common to separate people into "digital knowledge" groups. Immigrant: Is the one who was born and grew up before the technological revolution, for example a 65 years old man who did not have all the computers and digital tools or equipment that people do now. This person only adopted the technology at a certain age or point in their life when it was needed. Digital native is the one who grew up in the technological era, where he had access for example to the Internet, computers and probably experienced one or more ways of learning in a digital environment [?]. However, Prensky notes that time will make everyone a "digital native", as everyone will be born in a world full of advanced technology, so old generalization terminology will not be suiting in the future. He quotes Albert Einstein - "*The problems that exist in the world today cannot be solved by the level of thinking that created them.*" Prensky later introduces "Digital wisdom" that is a more general term but fitting in this era.[?]. Since target group is so wide in demographic and geographic aspects only grasp would be physiographic aspects but generalized to profession and lifestyle. Person's profession and lifestyle can show if specific person might use non-traditional interaction when it comes to navigating in 3D environment. For example video gamer, would most likely use one or more non-traditional approaches while playing mobile video game. Fx even simple game on a smartphone requires multitouch, or other sensors to control it. Therefore person who has a need or wants to do that, needs to have or already have digital wisdom. Then the focus is not how to create such interaction but rather how to make it efficient, effective, easily learnable etc. These aspects will be covered later in User Experience section.

#### 2.1.2 TARGET GROUP CONCLUSION

Conclusion can be made that most of the target group will be with digital-wisdom.

## 2.2 USER EXPERIENCE STRATEGY

This project is aiming for developing the intuitive user experience. To have a focus on intuitiveness means that we must be able to understand what that implies. The dictionary defines intuitive as:

- perceiving directly by intuition without rational thought, as a person or the mind.

they define the concept of intuitiveness as human perception by intuition, what then is intuition? again the dictionary will provide a relatively easy answer:

- intuition

1. The act or faculty of perceiving, or apprehending by means of the senses or of the mind; cognition; understanding.
2. immediate or intuitive recognition or appreciation, as of moral, psychological, or aesthetic qualities; insight; intuition; discernment: an artist of rare perception.
3. the result or product of perceiving, as distinguished from the act of perceiving; percept.
4. Psychology. a single unified awareness derived from sensory processes while a stimulus is present.

from this definition it is clear that the concept of intuitiveness is a human concept, more specifically a human subconscious concept. In an article from 1994 Jef Raskin[Raskin, 1994] talks about how intuitiveness comes from familiarity, while the article is quite old the observations that he makes does support the idea that intuitiveness is directly linked with the targeted users. In the article Raskin talks about an experiment that he performed, where he asks a test participant to perform a certain task with a mouse, back in 1994 the mouse was still not a tool that was commonplace and as such the test subject had no familiarity with how to work with a mouse, and required help. Raskin showed the participant how to move the mouse in the correct manner, and instantly the participant knew how it worked and didn't require any more help, because as Raskin notes: "*The directional mapping of the mouse was "intuitive" because in this regard it operated just like joysticks (to say nothing of pencils) with which she [The test participant] was familiar*" [Raskin, 1994] this observation strongly supports the idea of intuition

as familiarity. With this in mind the goal of this section becomes clear: first this section will give a brief overview of the topic of user experience. Next the section will try to define what the intuitive user experience is, and lastly how does the gained knowledge translate to being used as guidelines for making an intuitive app for a mobile device.

### 2.2.1 INTRODUCTION TO USER EXPERIENCE

A study of user experience<sup>1</sup> is a study of how a user feels when interacting with a system. The field encompasses a whole range of different and seemingly unrelated topics. The most known part of UX is probably the concept of usability which will be discussed later in the section, other things make up UX, such as: Design, Accessibility, System performance, Ergonomics, human factors and more concepts[Gube, 2010]. The term user experience was originally coined by Dr. Donald Norman, who was the first to describe the importance of user-centered design. User-centered design is a design concept that lets the users dictate(to a certain degree) what the system should contain and what form it should take. Before user-centered design the general design process looked like:

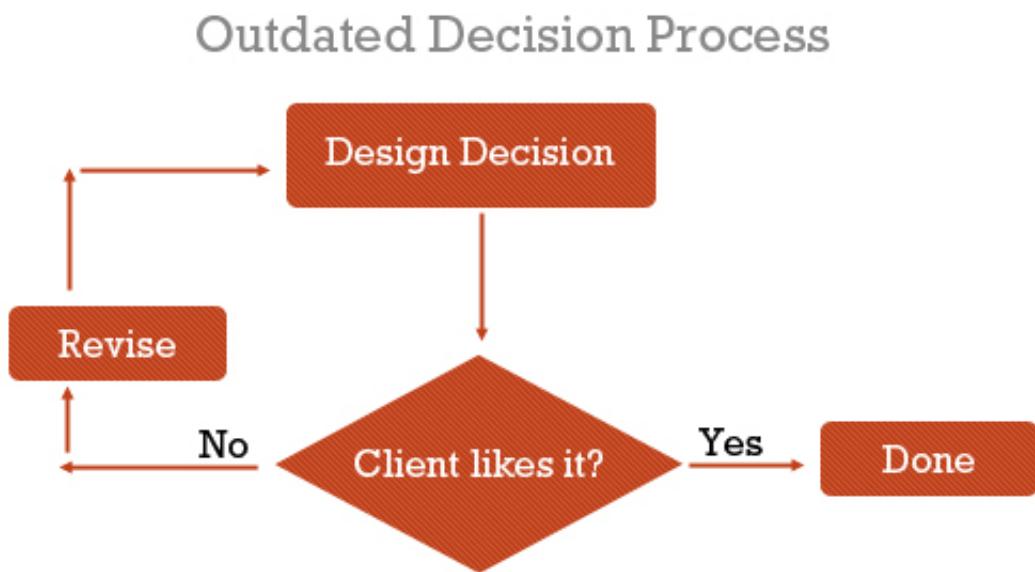


Figure 2.1: old decision process, Jacob Gube 2010

<sup>1</sup>hereafter referred to as UX

nowhere in the design process was the users a factor, the design was simply made according to how the designers as well as the client felt it should be. making the same kind of chart for a user-centered approach would look:

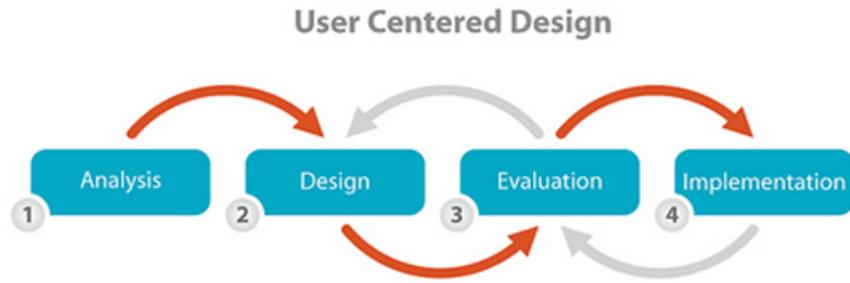


Figure 2.2: a chart of how user-centered design could function, Usabilla 2014

as this chart shows, user-centered design can be an iterative process. the grey arrow represents the user feedback, which shows that the users should be involved in the evaluation of a design.

User Experience and usability is often confused since a large portion of the guidelines for proper usability also applies to giving a good user experience. What sets the user experience apart from usability is that UX deals with the feeling of usage and usability deals with the effectiveness of usage An example of which could be the iBooks app for iPad.

source  
for the  
example



Figure 2.3: Apple iBooks for comparison

might  
need a  
source  
reference

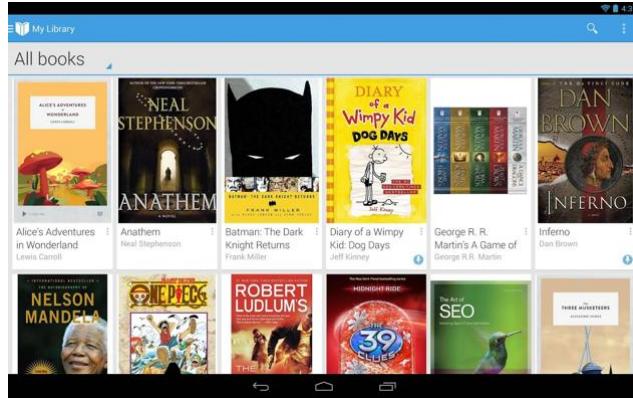


Figure 2.4: Google Play Books for comparison

An application for reading and browsing E-books. The layout is simple, it provides an overview of the owned books with a visual representation of the covers which is common for such apps and as such, do not set itself apart from the state of the art when it comes to usability, however the user experience is greatly improved simply by changing the background to resemble a bookshelf, it makes the experience of logging onto iBooks resemble the experience of going into a book store or library a lot more, this approach relates to the concept of intuition as familiarity, which will be discussed in the next subsection.

might  
need a  
source  
reference

## 2.2.2 INTUITIVENESS IS FAMILIARITY

as explained in the previous section 2.2.1 user centered design is a main pillar of user experience, this is even more true when talking about intuition as a design concept. As Jared M. Spool mentions in his 2005 article *People Intuit, not Interfaces*[?] the article mentions that it is the users that define whether or not an interface is intuitive as the interface itself is nothing more than a collection of code. What this shows is that for an interface to be intuitive, a comprehensible knowledge about the targeted users' previous experience with similar interaction, is not only useful but absolutely crucial. the article introduces the concept of a knowledge space, which is the arbitrary space that holds all the knowledge that pertains to a given interface.

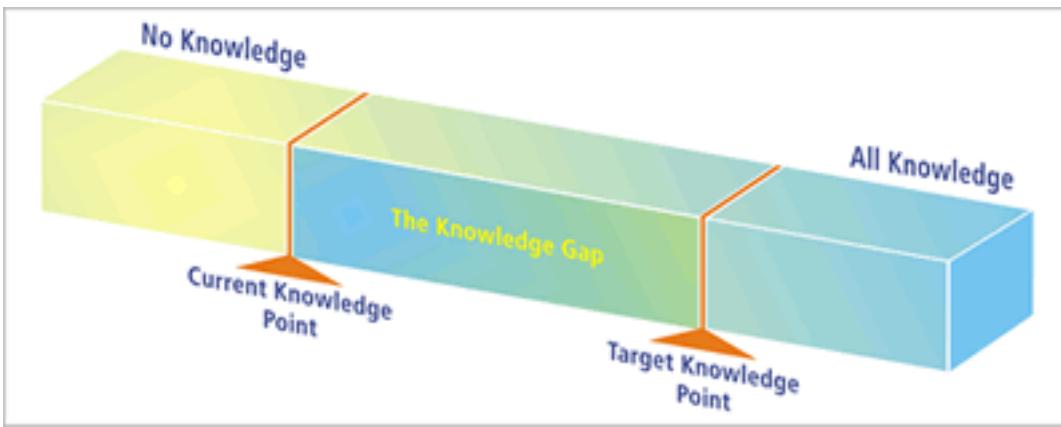


Figure 2.5: the knowledge space viewed as a continuous curve going from *no knowledge* to *all knowledge* [?]

As seen in figure: 2.5 there are two points of interest in the knowledge space that is the *Current Knowledge Point* and the *Target Knowledge Point* a brief explanation of the two points:

### **Current Knowledge Point**

This is the expected user knowledge which can be defined by a multitude of ways i.e. user interviews, analysis of similar apps etc. Figuring out what the current knowledge point is will enable the app to fill the knowledge gap without having to guide the user through every tiny detail.

### **Target Knowledge Point**

this is the amount of knowledge a user needs, to be able to use the app/programme as intended.

### **The Knowledge Gab**

the knowledge gab is all the knowledge that the app/programme will have to provide to the user, this is usually done with a series of tutorials.

He puts forth two conditions which he determines are the two conditions needed before users will classify an interface as being intuitive, these are:

- Both the current knowledge point and the target knowledge point are identical. When the user walks up to the design, they know everything they need to operate it and complete their objective.
- The current knowledge point and the target knowledge point are separate, but the user is completely unaware the design is helping them bridge the gap. The user is being trained, but in a way that seems natural.

Of these two conditions the latter one will probably be of more use to the project as the navigation with the gyroscope will not be a control scheme that the user necessarily have used before. Since the end product is going to introduce an uncommon way of interacting it will be important to know which kind of interaction will feel most familiar for the user, that is where the iterative process will enable extensive testing of different interaction models, to determine the correct approach for our users.

#### **FLOW THROUGH THE KNOWLEDGE GAP**

For the user to experience a state of flow the goal and the progress towards it for the user must be clear and well defined as this will be one of the most driving factors when trying to reach this state. The goal should be not too difficult nor too easy, but perfect for the user's skill. Just enough to provide the perfect balance between users skill and the challenges given.

The real experience begins with when one uses all the concentration on task or challenge at hand making user more and more focused until user "steps out" of his reality. This is when state of flow is being reached. During this state the user may only feel restricted to the rules of the activity instead of the ones in reality. This can also be called a loss of self awareness. During this state the user may lose sense of time or even one's bodily needs if state of flow is very strong.

To maintain the state of flow receiving immediate feedbacks step by step on how the user is doing is to be established in order to adjust the challenges and skill level necessary to achieve upcoming goals.

#### **2.2.3 DESIGNING INTUITIVELY**

The topics discussed in the previous section helps define what the app has to be able to do, but besides these ideas, besides these topics the project will look at the following two structures that can help create a pleasant UX:

- Empowering Users to Complete Tasks Faster

*"When a user has a good experience, one of the first things they say that they liked about it is that it was fast. Since users "equate fast with easy," [?] the app that this project will develop does not contain a wide range of features but is a relatively specialized app, while this diminishes the urgency of the app being fast, it should not be neglected. Robinson points to 6 ways of empowering the users effectiveness[?]:*

**1. Make the app work faster**

this is a straight forward engineering problem, as better/less code results in a faster interface.

**2. Simplify your users' work flow**

this means cutting down on the amount of screens that the app employs.

**3. Make sure your navigation is intuitive**

As talked about earlier intuition is related to familiarity and familiarity coupled with the umbrella structure mentioned above should be able to provide an intuitive navigation within the app.

**4. Reduce the amount of text**

in relation to the second point, if an app has a lot of text it will slow down the work flow of the user, at least in the beginning.

**5. Examine your graphics**

Robinson points to graphics as being an important part of how a user perceives an app, she urges to keep the graphics: "*clean and not distracting*"

**6. Buttons**

when making any kind of button make sure that the user never questions whether or not it is a button, further Robinson also encourages to give the buttons one word labels such as "send", "buy", "find" etc. of course the words should represent the action that the button performs.

these points together with the intuitiveness discussion above should enable the app to provide an intuitive user experience.

## **2.2.4 USABILITY**

According to John Wiley's "Interaction design", there are usability goals that worth considering when developing interactive usable system [?] Covered usability goals are:

- effective to use - it is how system does what it suppose to do;
- efficient to use - how does the system supports user while he interacts with it;
- safe to use ;

- good utility - provides with variety of functionality that user might need;
- easy to learn;
- easy to remember how to use.

Wiley notes that not all of the goals are needed for most of the systems. Specific system should have specific goals. Probably most important and relevant for this application design is effectiveness and efficiency. To help user to navigate 3D environment easy system has to be with accurate response which saves time and does wanted task quickly. Fx. if user wants to view 3D object from certain angle, it is easy and fast to navigate to the point where the view angle is desired. For the beginners of such interaction with 3D environment, "easy to learn" and "easy to remember" goals would be most relevant. Don Norman in his "The Design of Everyday Things" states that there is principle of "affordance". [?] Principle is being explained as fx. Cup is affordable to being picked up; door is affordable of being opened because of it's handle; button affords being pressed etc. So if the user is beginner, principle of "affordance" is extremely useful. Norman notes that there are two "affordances" real and perceived. Real is the one in real world as in examples mentioned above, and perceived is imitation of real. Fx. in this design, for beginners users especially, navigation buttons needs to look very similar to "real" navigation buttons. Fx. if user is familiar with any common video gaming console buttons, they have to be represented in a interactive software system with similarity to real one.

Don Norman also covers other principles of usability:

- Visibility - how visible is the button on the screen ?
- Feedback - what kind of feedback does the system gives to the user ?
- Constrains - visual constraints as fx graying out some menu which is not used;
- Mapping - layout of menus, buttons etc.;
- Consistency - layout's, colors, typography etc consistency through the system;
- Affordance - real and perceived affordance, covered above.

[?]

Like with usability goals not all of the principles of usability needs to be in one system. However, in this case, design of the program can include all mentioned principles. Fx, if the

3D navigation system is controlled by the buttons on the screen, they need to be visible, give some sort of feedback when pressed, mapped out with consistency rules and visualized with "affordance" principle.

## MOBILE USABILITY

Since the design of this system is meant to be on a mobile platform, there are certain aspects to consider. Some users might have limited mobility or problems with manual dexterity - it will cause higher error rates with the interaction [?]. Even small aspects like person's bigger fingers can cause troubles while using mobile phone and especially while interaction needs precision. This problem is mentioned in many articles, reports, researches as fx. in "The Generalized Perceived Input Point Model and How to Double Touch Accuracy by Extracting Fingerprint" where they introduce possible solution to this problem. Since the focus of this project is not solving such problems, deep analysis will not be done. However, some considerations might be done, as fx. using Norman's principle of feedback. Buttons pressed on screen can change color or size, mobile device could use some sensors to indicate that action was achieved, as vibration, blinking flash etc. It is especially useful since most smart-phones using on-screen touch buttons and physical sensation of touching the button is not existing. In that case, user who did not register if he successfully completed wanted action, mobile device would help and give feedback using its sensors. Feedback is also useful to people who do not put lots of attention to their interaction accuracy, fx. pressing somewhere around button area and not on it. Feedback would help user to know if action was successful. In general, it is important to consider minorities and helping user effectively understand their efficiency of actions. Since this non-traditional interaction method can be used by many people as was discovered in analysis of target group, product needs to be optimized to at least fit majority of the users.

### 2.2.5 CONCLUSION

This section has aimed at defining the intuitive user experience which has been defined to be an fulfillment of one of the two conditions mentioned on page ???. The section has also showed that intuitiveness depends on both the users previous knowledge as well as the design of the interface. Considering Norman's and Wiley's mentioned usability goals and principles it is possible to create an useful application. Usability goals of effectiveness and efficiency, will be achieved by using all covered principles. Principles will be used in creation of visual elements

and coding. Having goals as easy to learn and remember, will help user with digital knowledge to narrow knowledge gap while using app first time. Solution to that is to create interactive navigation using affordance principle and visual familiarity.

## 2.3 INTERACTION METHODS IN 3D ENVIRONMENT

There are different ways of conveying 3d environment within the smart device spectrum (in this case - smartphones and tablets). It is important that the aim of the project is established before moving onto conclusions on which approach to take. Since the project revolves around orientation in 3-dimensional environment, irrelevant approaches will be eliminated right away.

A very important part when developing the application to fit a pleasurable user experience is to make sure that it works as flawless as possible, and there are no misinterpretations when using the product. For the product that is going to be developed, the “traditional” interaction methods do not cover the functionalities that are needed to cover our initial concept needs. To assure that the alternative interaction is integrated in a convenient manner, knowledge about different sensors and possible combination of two or more to make more intelligent outcomes should be established.

It is also important to note that the proportions will differ from reality because of the device size. This project is aimed at visual representation of a rather realistic environment, and that will be taken into account during the implementation.

### 2.3.1 TRADITIONAL INTERACTION

As technology evolves, new ways of interacting with computational devices are constantly built. With that, people's needs also change, the people adapt to new ways of communication with technology [?]. The transition from the classical buttons on a cellphone to a touchscreen has made new ways of interaction possible - the delimitation of physical buttons made it available to have any customized graphical interfaces on the screen possible. This made life easier for casual tasks - like zooming a photo using two fingers as multitouch input (which allows registration of multiple points of contact simultaneously), which is much more intuitive than the classical button alternative. Soon enough non-traditional sensors started finding their place in smartphones - the implementation of these sensors in the smartphone allowed new forms of interaction, such as video calling, flashlight and screen orientation, followed by more

Double-check  
this part  
please,  
because  
not sure  
if it's  
right.

also not  
sure if  
this part  
is nec-  
essary  
here

for some  
rea-  
son cite  
doesn't  
work

interesting unusual uses - application developers started making instrument tuners (GStrings), barcode scanners (Barcode Scanner), radiation detectors (GammaPix), pulse detectors (Instant Heart Rate), light intensity meters (Light Meter) and countless other applications that use the sensors to their favour in a non-traditional manner. However, we will only focus on sensors that support our problem area, which points to the ones that can work with 3d environment - the relevant ones for this project are the gyroscope, accelerometer and magnetometer. In section 2.3.2 possible sensors and combinations of them that are considered to be used to achieve a functioning prototype will be discussed.

### 2.3.2 SENSORS

As it has been established so far there are three main sensors that can support orientation in 3D environment possible. These sensors are accelerometer, gyroscope and magnetometer. In the following section it will be determined whether these sensors are in fact useful to this project's case and if so - to what extent.

#### **GYROSCOPE, ACCELEROMETER AND MAGNETOMETER**

Gyroscope, Accelerometer and Magnetometer are the three sensors that most of the newer smartphones have [?] that can measure orientations on X, Y and Z axis, allowing the apps to calculate placement of the device in the 3D environment.

Gyroscope can be used to help determine orientation using gravity. Since it detects rotation in 3-dimensional space, it can be used in favor of this project to convey the rotation that is in a way similar to the one that is natural - eyesight. Gyroscope, in comparison to magnetometer and accelerometer, is the physically largest and most expensive sensor, so the possible limitations in the smart devices in-built Gyroscopes have to be considered.

reference  
error  
again



Figure 2.6: iPhone game using a gyroscope sensor

Gyroscope cannot detect the direction the phone is facing. Luckily, the accelerometer can support it to give the impression of the environment representation in 3-dimensional space - this helps to stabilize the view angle to represent real world by giving the position perpendicular to the Earth's surface. In other words, it would eliminate wrong position starting point - moving forward horizontally in reality while in the virtual environment it goes upside down could cause cluster. Accelerometer, along with other sensors is commonly used in the augmented reality concepts (Yelp Monocle, Google Ingress, SpecTrek etc).

The last sensor that is able to collect 3-dimensional data is the magnetometer. It is typically used to measure the absolute position of 3-dimensional orientation in terms of Geographical placement on earth, which is irrelevant for this project. Additionally, magnetic interference can disturb its flow, which may lead to inaccurate results. In future development of the project, an online feature with more precise positioning in relation to other users could be considered. Alternatively, the magnetometer can be used to support the gyroscope and accelerometer with positioning, if they lack precision axis-wise. If there are no problems with accelerometer and gyroscope axis measurement i.e. blind spots that these two sensors can not recognize - implementation of the magnetometer is not necessary.

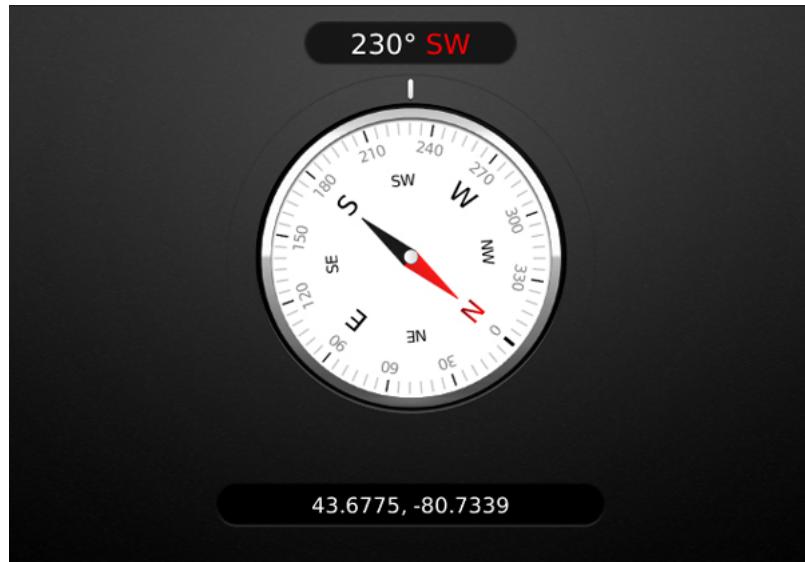


Figure 2.7: a simple compass app that establishes the magnetometer sensor

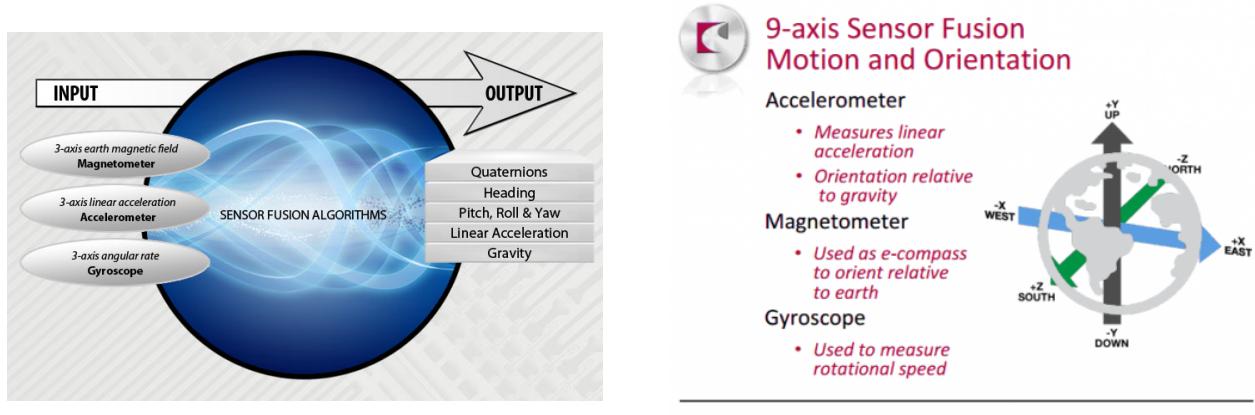


Figure 2.8: uses of sensors

### 2.3.3 CONCLUSION

The sensors discussed in earlier sections will help us work towards establishing that the perception of 3d environment is ensured to be conveyed in the best manner. After gaining more knowledge it can be seen that there is a variety of options how the use of sensors can be enabled in different application contexts. For the projects specific case - a fusion of two sensors will be used - gyroscope and accelerometer, as these are the mobile sensors that support the 3d

environment emulation the best for our case.

## 2.4 GRAPHICAL DESIGN

In graphical design, like any other design, there are many options to be considered. Colours, fonts, balance and many more factors should be chosen carefully. Most importantly, how to mix these elements together without making a mess. This will reflect greatly on how a design is being perceived. [Ciotti, 2013]

Two of the things people often complain about in applications is a confusing interface design and poor navigation. [Neil, 2014] This can be prevented by using design patterns. These have been created to avoid a messy and intolerable app. First, navigation. There are two ways to make navigating through an app easier. Persistent and transient. Persistent navigation is list menus and tab menus or menu structures. Transient navigation has to be revealed through a tab action or the likes.[Neil, 2014] Consider if the user needs to see the menu at all times and if not, an off-canvas solution like a side-bar could be preferred.[Neil, 2014] This way, the app is able to hold many informations without being confusing or plastered. Too much text in one page or a simple form taking up several pages will make the app confusing. A sign in for example should only be one page. A way to not get an overlapping look is by using vertical labels instead of horizontal. [Neil, 2014] Or you could have the horizontal labels where the text disappears as soon as the user starts typing, but you risk that the user forgets what they should fill in.[Neil, 2014] Some apps, like Instagram, shows the "sign in" and "sign up" option all the way through the tutorial. This also insures that the user do not have to go through a whole tutorial if they do not need it.

Keeping these patterns in mind there are still many things to consider. First of all, remember the size of the screen that is being designed for. Avoid using big scaled photos and put too much information on one page. This will make it look cluttered and make it less intuitive. [Sardo, 2009] In short, make everything as clean and simple as possible.

### COLORS

Colours are not just colours when designing a brand, an app or a website. Colours are perceived in various ways and is a big part of how the design is coming across to the user. [Ciotti, 2013]

It is important to remember that when choosing the colour palette for a design, that how we

perceive colour is very different. Also, colours can change according to what it is next to. Yellow might look different next to grey than it will next to purple for instance. [Hampton-Smith, 2014]

When it comes to colour psychology the truth is, it is too dependent on personal experience. There is no one right answer to which color falls into what mood. [Ciotti, 2013] There is, however, many studies conducted on this matter. One study shows that 90% of people make snap judgement based on colour alone. [Ciotti, 2013] Another study shows that an intend of purchasing is linked with how a brand is perceived i.e. what kind of "personality" does the brand have? [Ciotti, 2013]



Figure 2.9: Overall image of how colours are generally perceived. [Ciotti, 2013]

But all in all, the concept of the app is key. Almost every study shows that it is greatly more important to choose a colour that shows the personality of your product than picking a stereotype colour. [Ciotti, 2013] This app is directed towards interior design and therefore it is fitting to give the app a inspiring and creative personality. According to fig. 2.12 the color purple is the inspirational and creative color. This is very feminine but mixing it with a neutral colour like grey might just take it down a notch.

Colour preferences differ between genders as well. A study shows that women prefer soft colours and tints while men prefer bright colours and shades. [Ciotti, 2013] Since the target group of this app is not gender specific it is important to make the app gender-neutral i.e. not too soft and feminine but at the same time not too bright.

So how does one find the best way to coordinate different colours? Research indicates that the isolation effect is very useful.

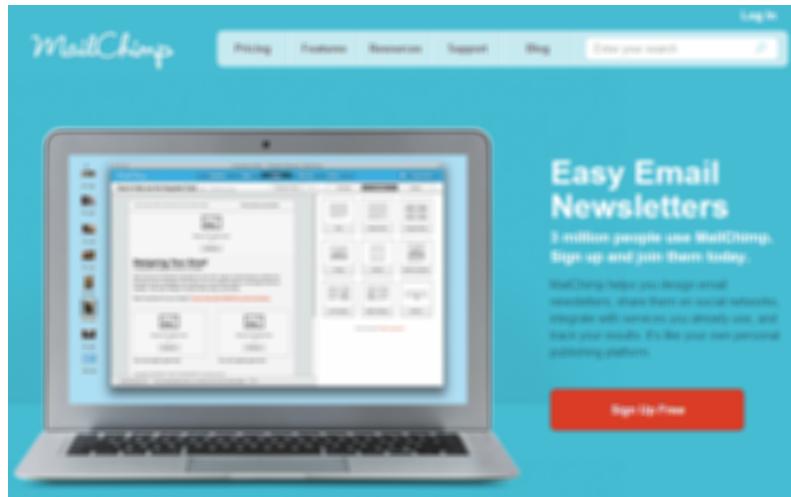


Figure 2.10: "The sign-up button stands out because it's like a red "island" in a sea of blue." [Ciotti, 2013]

Using the isolation effect will help the user have a more efficient experience because the most important feature e.g. a "sign up" button, stands out. [Ciotti, 2013] (See fig. 2.13) Research suggests that a colour scheme that consists of analogues colors and combine it with a accent complimentary color or a tertiary color is preferred among users. [Ciotti, 2013]

When designing your layout it is, once again, key to keep everything simple and streamlined. Follow the general rules, left-to-right and top-to-bottom. Make sure the most important feature is in the top left corner where the user will look first.[Sardo, 2009] Be careful, yet not boring, when choosing a colour scheme or font type. In general, keep the graphics clean and simple. No muss, no fuss.

## 2.5 STATE OF THE ART

def: State of the art *State of the art is the level of knowledge and development achieved in a technique, science, etc, esp at present*

This section will be an analysis of a number of applications that all focus on designing a room or a set of rooms. The previous sections have provided the necessary framework for doing the analysis. specifically the analysis will cover:

- the familiarity the of the different aspects of the apps.  
what parts of the app have been seen in other apps or in real life.

- the knowledge space for the app  
here the analysis will try to determine if the app successfully bridges the knowledge gap talked about in ux section ref: fig. 2.5.
- The graphical design: Color, overall layout. Farver, fonts og sådan overall lauout
- the different interaction methods that is used.

finally the end of this section will sum up the trends noted and will give a overview of what aspects the different applications have lacked behind with and what aspects this project could aim to improve.

### 2.5.1 IKEA KITCHEN PLANNER

In this application customers can create accurate measurements of their own kitchen and place the furniture from IKEAs catalogue. It is possible to do different wall measurements, add wallpapers to walls, apply different ceiling and floor covers, add windows and doors. Users can view the layout from top-down view and later see how furnitured layouts look in 3D perspective.

IKEA's web application for designing kitchens is used mainly in actual IKEA stores. This could indicate that users need help using this application. It is used as a tool with focus on efficiency and not so much an app you would use at home for interior design. Most of the people that were asked in the initial interview also confirmed that they do not use this application. Some of the participants are familiar with the app and have tried the it but do not use it.

When you first enter the app there are no immediate help or tutorial. There is, however, two places in the app where it is possible to get help yourself. Very reasonable layout going from left to right and top to bottom. There is a lot of easy-recognizable buttons which helps the user experience along. It is very slow though and highly affects the usability.

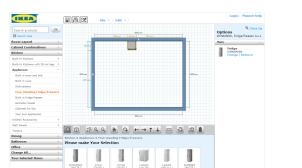


Figure 2.11: IKEAs kitchen planner webapplication in 2D mode.

The app is grey and clinical but again, it is a practical tool for IKEA customers to visualize a kitchen. The application offers plenty of useful features and can be used to give a grasp of how people's homes would look like prior to buying the actual items, however, it is rarely used by IKEA's customers. The problem could be that the application is hard to use, leading to long time spans used to build the desired kitchen design. A solution to this possible problem could be to create an application that is more intuitive and takes less time to achieve the user's needs.

### 2.5.2 HOMEDESIGN3D

This mobile application is made for interior design. It has most of the basic features; building rooms, placing furniture, windows and doors. The user can then switch to a 3D view. Here there are two settings to choose from - a joystick where you use both thumbs to move around the house, or arrows where you can view the room by moving your finger around and use the arrows to move from room to room. From the 3D view you can paint the walls and change flooring.

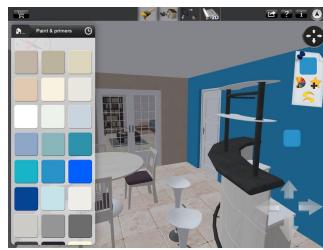


Figure 2.12: HomeDesign3D application. Painting walls.

When you first enter the app you are faced with the option of buying the full version. This is actually the start of a long tutorial. It is very hard to notice though, since the points at the bottom that indicates that you can swipe is overrun by a commercial. It is very unlikely that the user will find this help from the beginning and therefore will properly "go back" to the menu. The tutorial is easy to understand but very long. It has both pictures and text. It looks messy because of the background and the hand drawn hand that shows how to do the different thing. It is not very consistent in matters of graphical design. You can find this app later on in the menu. The next step is the design. There is apparently no start menu or the likes.

The icons are easy recognizable.

The layout looks very cluttered. There is the big commercial at the bottom and the menu bar in the top. The mix of the cool black bars and buttons with the beige background does not

go very well together. The fact that the background is textured as a wall as well does not help the graphical aspect of this app.

### 2.5.3 AUTODESKS HOMESTYLER

This app is autodesk's attempt at making an interior design app. the app does not provide a user guide from when you open up the app, this is opposed to the idea about bridging the knowledge gap with tutorials. The app does however provide a guide for users, once they are actually designing their room, but if the user is not able to get to this point then they are stuck. The overall look of the app is reminiscent of the flat design pattern as seen in Windows 8, this makes the app feel very modern. it also makes the app look very exclusive. However during the main activity of the app the design does not exactly match the look when browsing the catalogue, this leads to the app lacking consistency.



Figure 2.13: Homestylers lack of consistency. Right: The design of the catalogue. Left: The design of the actual room planning feature.

The app's button design also seem to be focused on bigger screens than a smartphone. The big button at the top, labeled "Redesign" is a good example of both the isolation effect mentioned in ??and also the button follows the principle of having the buttons be labeled with a single word ?? . the interaction in the app is primarily clicks, with some multi touch functions for the more advanced functions such as resizing, moving, rotating furniture, these functions are explained the first time the user is using them and then never again.

fix reference to reference to Idas section on the isolation effect

### 2.5.4 PLANNER5D

Planner5D is an application made both for web and mobile. The web is however better executed than the mobile version. This app allows you to build rooms and place furniture, windows etc.



Figure 2.14: Planner5D, placing furniture.

Start screen gives nice tutorials but are primarily composed of text. The knowledge gap is very tiny, close to nothing. The menu bars are well divided into four sections. The menu on the left is very apple like in its graphical feedback. Besides this there is a toolbox which is nicely divided by category, a bar at the top for social/profile aspects and lastly a menu that appears when you have placed a piece of furniture in the room; this is only in icons and no text hovers over this which makes it unclear what the different buttons do since the icons are not very familiar. A lot of focus on user friendliness.

The app uses mainly point and click with drag.

The graphic design is very minimalistic and modern. The different features has been nicely placed which puts the focus on your own design.

Furniture interaction does not match the real life movements of the user.

### **2.5.5 CONCLUSION**

## **2.6 DESIGN REQUIREMENTS**

- familiarity
- non-traditional interaction explained without text
- uphold one of the two conditions from the knowledge gap bit.
- make UI elements obvious(make use of isolation)
- collapsible umbrella style UI
- make use of accelerometer/gyroscope
- stick to one color scheme
- limit amount of information given to the user

## **2.7 DESIGN REQUIREMENTS**

There are two main focus areas: familiarity and knowledge-gap. This chapter will discuss these two aspects and establish specific design requirements. The goal for the prototype is to

define which way to control the 3D virtual environment is the most familiar one for the user. As mentioned in User Experience chapter (??), familiarity is very related to intuitiveness. To minimize the knowledge gap for the users with digital knowledge, two conditions from the knowledge gap (section ??) will be upheld through the concept of familiarity. The users are expected to understand the controls and use them with little amount of practice. Based on that, several design requirements need to be established. In general navigation has to give positive user experience, user has to be in the “flow” state.

## **TECHNICAL REQUIREMENTS**

### 1. Graphical User Interface

#### (a) Controllers

- i. The controllers have to be familiar to the user, the buttons need to stand out by using perceived affordance, isolation, visibility, mapping and consistency principles [ref 1 -Usability][ref 2 - Graphical design]

#### (b) Software engineering

- i. Controls has to be responsive - effective [ref 3 - UX Conclusion]
- ii. Controls has to give effective feedback [ref 4 - UX Conclusion]

### 2. Software engineering

#### (a) Controls has to be responsive - effective [ref 3 - UX Conclusion]

#### (b) Controls has to give effective feedback [ref 4 - UX Conclusion]

## **CONCEPT REQUIREMENTS**

### 1. Navigation

#### (a) Navigation has to be familiar for the user

- i. The navigation should either relate to controls the users have used before, or relate directly to the real world.

#### (b) Make use of the gyroscope in context of familiarity;

- (c) Consider graphical element placement and sizes as the navigation is for mobile platforms;
- (d) 3D navigation test level;
  - i. Navigational level has to help to reveal each control schemes efficiency and effectiveness [Ref - Usability] The test level has to challenge the user to a certain degree in order to be able to compare the different designs. The level should must also make it clear to the user what the objective is and how they should achieve the goal in order to measure the effectiveness of the controls. The colour scheme of the level should help put the focus on the objective ?? rather than the surroundings.

## **CONCLUSION**

These design requirements will help build the necessary prototypes that aim to minimize the knowledge gap using an interface to move in 3d virtual environment on a mobile platform.

## **2.8 CONCLUSION**

### **2.8.1 FINAL PROBLEM STATEMENT**

*How does non-traditional interaction using the concept of familiarity help minimize the knowledge gap for users with digital wisdom, for navigating in a 3D environment?*

# 3 | Design

## 3.0.2 INTRODUCTION

In this chapter, the design of the prototype and its graphical decisions will be discussed and defined, with the requirements set in the analysis, which were further defined in the design requirements section (??). When designing, different aspects will be taken into consideration for user experience - usability goals ?? and principles ??, mobile usability ??, establishing intuitivity through familiarity, as well as the technical side on which the user experience also depends - fluent implementation of the graphical user interface and non-traditional sensors, while also ensuring that the system performance is smooth. In further design iterations users would be involved in the design process ??, as user centered design is one of the elements to designing a good user experience ??.

## 3.0.3 CONCEPT

The goal of the prototype is to help in answering which control scheme may be the best solution for the most familiar and effective navigation in 3d environment. Designing a prototype that creates a familiar and effective navigation that is not traditional can go in various directions, but the concept was narrowed down to using just the sensors that could help establish the design requirements initiated earlier (??), those being the multi-touch and the gyroscope. To be able to design a prototype with all the necessary aspects, control schemes and a base test level for navigation in 3D virtual environment have to be designed with the established design requirements ?? . The prototype has to implement two features to navigate in 3D virtual environment - the moving of the camera and the camera rotation. It was chosen to make extremities of the control schemes to establish familiarity in distinct ways. The familiarity concept will be put into effect as the control schemes for controlling the camera have to be

linked with something that the user might be familiar with already.

### **3.0.4 INTERFACE DESIGN**

The currently existing sensors in mobiles enable the creation of unusual ways of controlling a 3d environment. With the goal in mind of achieving familiarity through the non-traditional sensors, two different approaches were established. The knowledge and familiarity to the currently existing products on the market (??), and the familiarity of movement representative to the one of movement in real life. For both approaches the current and the target knowledge ??, points for the user are expected to be separate. The design needs to be established in a way to help the users get through the knowledge gap intuitively, when they are involved in the designed task. It was chosen to make extremities of the control schemes to establish familiarity in distinct ways. It was important to keep the same movement speed settings for all the control schemes for later evaluations. That meant that in the perfect scenario, the user should be able to get from one position to another with the same amount of time spent for that task on each of the control schemes.

### **3.0.5 CONTROL SCHEMES**

Three distinct control schemes were designed for the navigation. Buttons-only, Joysticks-only, and one that includes buttons for moving back and forth, with a gyroscope for moving the camera. The fact that newer smart devices are capable of multi-touch input has been used as an advantage to enable both the movement and the rotation to be controlled simultaneously. That creates the possibility for the user to walk and turn around at the same time.

#### **GYROSCOPIC CONTROLS**

Establishing the gyroscopic sensor creates a possibility to interact in an unusual, but familiar way. It enables the prototype to be built around what is most familiar to the real life in movement - actual moving around in reality to navigate. Controlling the camera with a built-in gyroscope in the tablet is familiar with a natural way for a person to look around - by turning the direction user wishes to face. The movement forward and backwards was implemented with on-screen buttons as these were familiar to the target group through the usual human-computer interaction, since most of the controllers for movement are represented as such (e.g. arrows on

the computer keyboard, music player, cell phone). Such control scheme, when supported by the flow theory ?? has the potential to be the most engaging one.

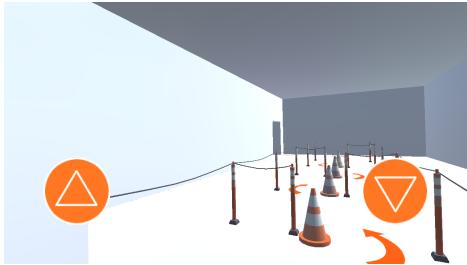


Figure 3.1: Initial sketch of gyroscope 1

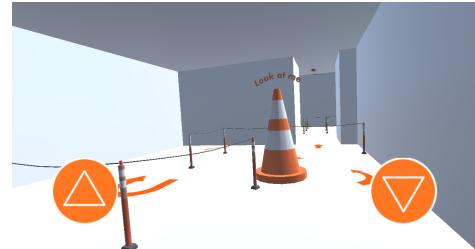


Figure 3.2: Initial sketch of gyroscope 2

## JOYSTICK

In this prototype the user has to navigate using two joysticks - one for movement and one for camera movement/rotation. This should be easy to learn for the users that have experienced using a joystick before, and the target group is expected to have some knowledge as of how a joystick is supposed to work, because of their popularity in arcade and electronic games, where joysticks are placed on game console's remotes like Sony Playstation series or Microsoft Xbox. Additionally, even for users with no previous joystick controller experience, that should not be a problem, as the control scheme is supposed to borrow the same concept as moving a computer mouse on the screen in the direction that is same as the device movement itself - both use 2d directional movement.

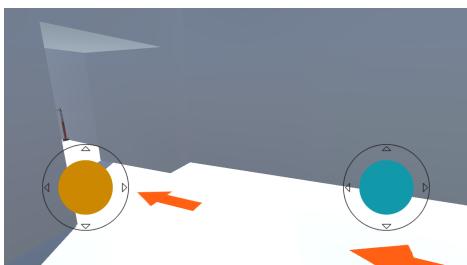


Figure 3.3: Initial sketch of joystick 1

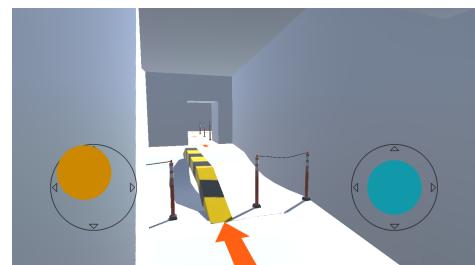


Figure 3.4: Initial sketch of joystick 2

## ON-SCREEN BUTTONS

The way that should be the most familiar with the users through daily use - only buttons as the way to move both camera and the character. In this case camera would be moved only with

arrow keys located on the screen and same for moving around - arrows indicating movement back and forth. This should be familiar with anyone that has used buttons for navigation of any sorts in virtual environment. By implementing familiarity concept to navigation, the problem of minimizing the knowledge gap is tangled so that the user is being trained in a way that seems natural.



Figure 3.5: Initial sketch of buttons 1

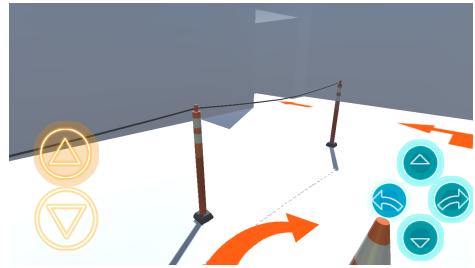


Figure 3.6: Initial sketch of buttons 2

### 3.0.6 ISOLATION

To further emphasize on the initial design requirements, the concept of isolation (??) should be used when designing the controllers - this is done by making them stand out from the level content. This is supposed to help the user understand what parts of the application give feedback upon interaction.

### 3.0.7 IMMEDIACY AND SIMPLICITY

To communicate information in a simpler, and faster to perceive manner, the designs should be represented by concepts that are already familiar to the user. This means, that to communicate information to the user, graphical elements should be represented as symbols that indicate either movement or rotation for buttons, and controls that represent an actual joystick, rather than text. At the same time, this emphasizes the concept of affordance ??, as the buttons represent mechanical buttons used in traditional types of controllers, as well as with the joystick controls. This will further shape the familiarity and intuitivity for the application (??).

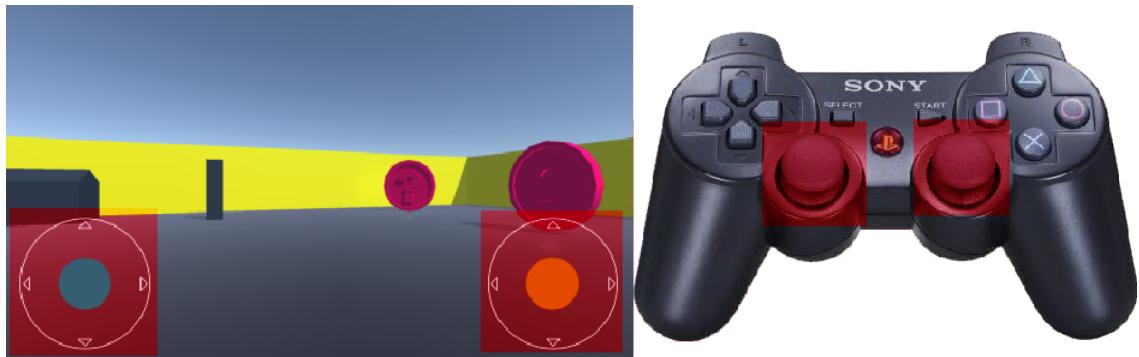


Figure 3.7: Comparison of initial design sketch of on-screen joystick and the Sony Playstation controller

### 3.0.8 GRAPHICAL ELEMENT SIZES AND PLACEMENT

To further emphasize on the user experience, the button size should be set accordingly - to (ref to UX ensure that the users would not have difficulties by unintentionally tapping the wrong section tetraple-gia stuff) of the controls, individual buttons have to be separated from each other and given the sizes for easy accessibility to reduce the “Fat Finger” problem. To enable a bigger view of the environment horizontally, the application will be built to primarily be viewed when holding the device in a landscape mode. Since the device is supposed to be held sideways and by both hands, all of the interaction should be done on the sides of the screen, for ease of control access. To show the difference between movement and rotation controls, they should be given different looks - shape of directional arrows for buttons as well as color indication for both buttons and joystick controls.

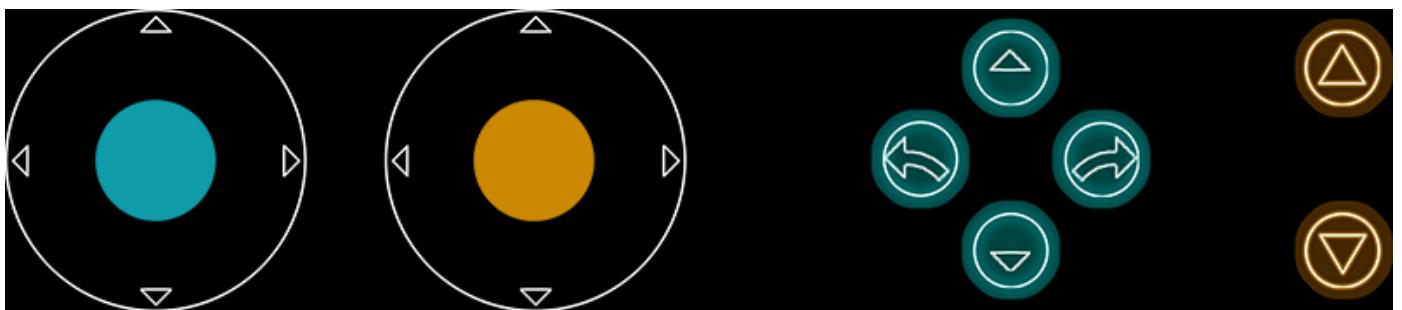


Figure 3.8: Initial sketch. Color differences to distinguish controllers that hold different functions

### 3.0.9 DESIGN OF 3D TESTING AREA

To test different non-traditional control schemes 3D test area was designed. It consists of twisted path that test participants had to walk through as fast as possible ??.

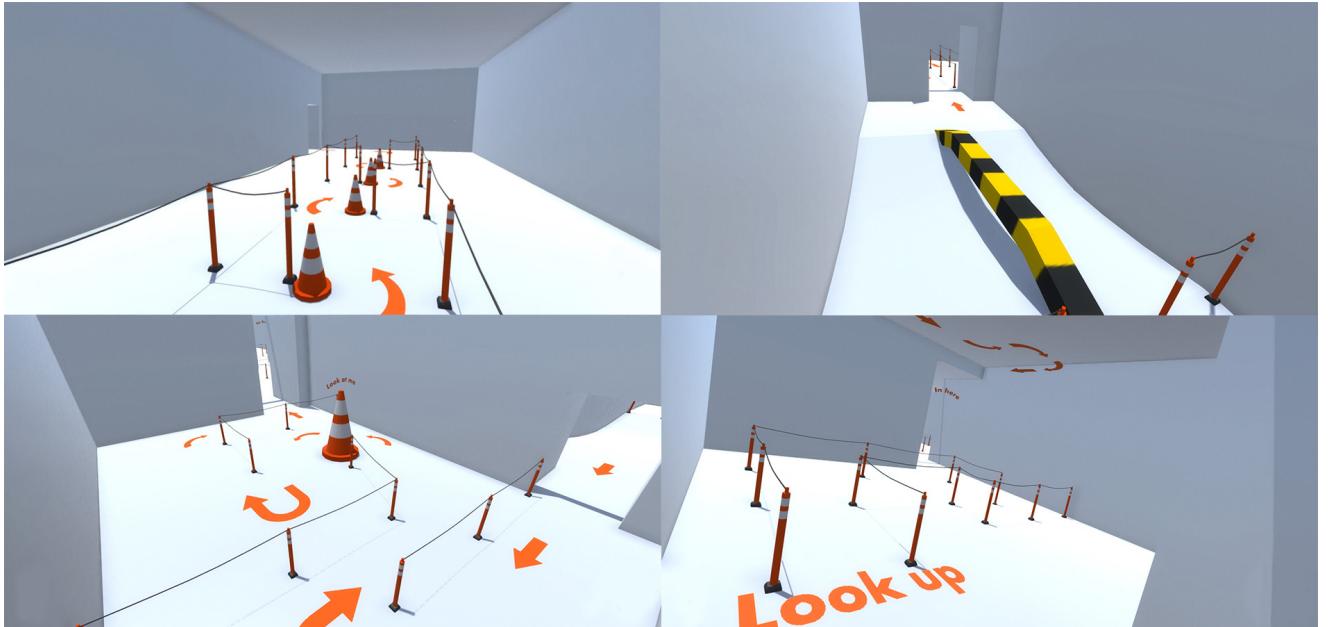


Figure 3.9: Pictures of the testing area

Focus of testing area was only on navigation for different schemes, so effort was placed on path and tasks and not on environment itself. All not important bits of the area were colored white and important as navigational arrows, cones and poles, explanatory text notes colored with sharp contrasting colors as red and orange. The “familiarity” consideration was done - navigational buttons and important bits on the level were kept in similar color, to help user recognize and focus on the purpose. Analysing SOTA’s applications gave understanding what important aspects of navigation is. Firstly walking fluently around obstacles as furniture, doors, narrow paths. First two levels were developed for this to test. Second consideration was that users need to look around placed furniture which was not considered in most SOTA analyzed applications. Small area with huge cone and text “look at me” was placed and arrows in circular path around it. This represents how user would walk around furniture and inspect it by looking - focusing on one point while walking around ??.

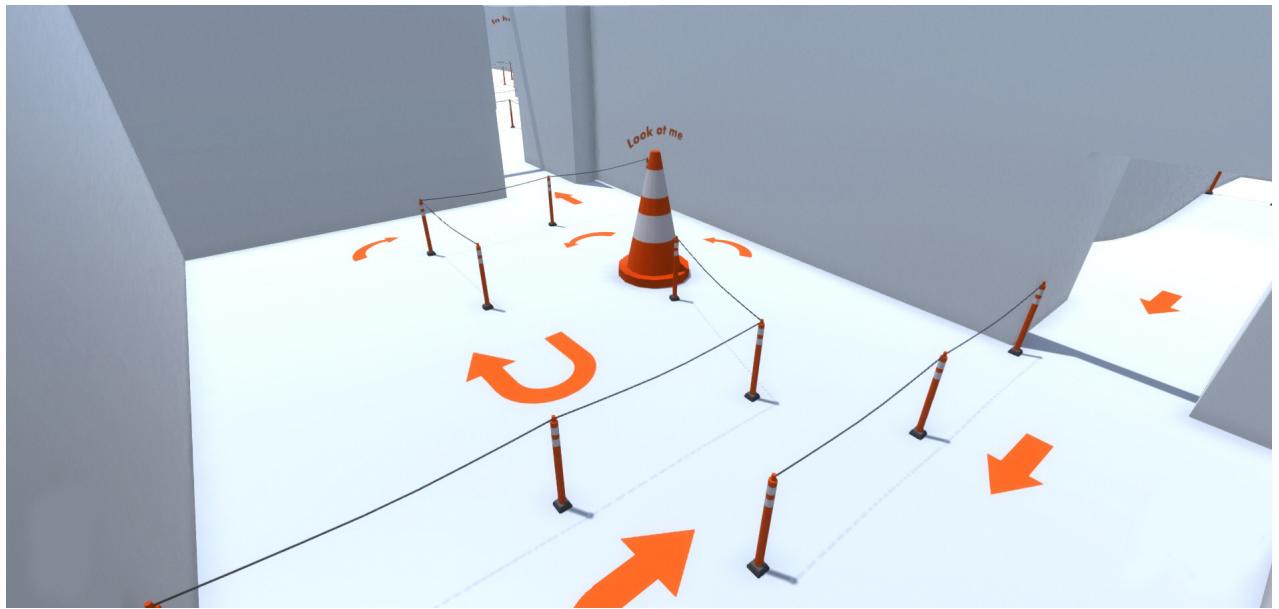


Figure 3.10: Third level of testing area. This level is to test how well user walks around an object.

Next two levels were developed to test how efficient it is to look up and down while walking in given direction. It helps to test how well user can walk and look down or up at the same time ??.



Figure 3.11: Test chamber to test how well user can walk and look up

The last test area was created to see how user goes straight but looks to one side as the user would walk-by, but focus at some furniture aside ??.

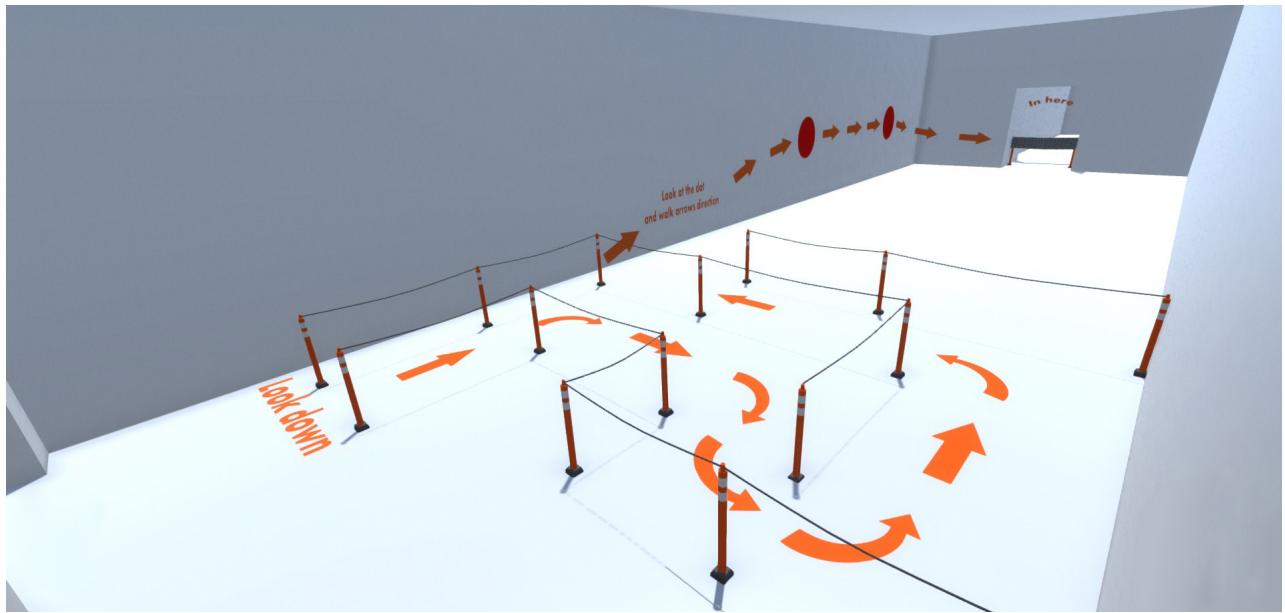


Figure 3.12: Last level of testing area. This level is to test how well user can walk while looking down and sideways

# 4 | Implementation

## 4.0.10 DEVELOPMENT OF 3D ELEMENTS

Actual implementation consisted of creating level areas, 3D prefabs - cones and poles with string, sprites - notes and arrows. 3D elements were created using software "Maya" and textures with "Photoshop". Each level started by simple creation of a square representing a room. Then considerations of what test it will include were done and assets that were created were placed making different challenges. Picture ?? shows 3D cone and it's texture.

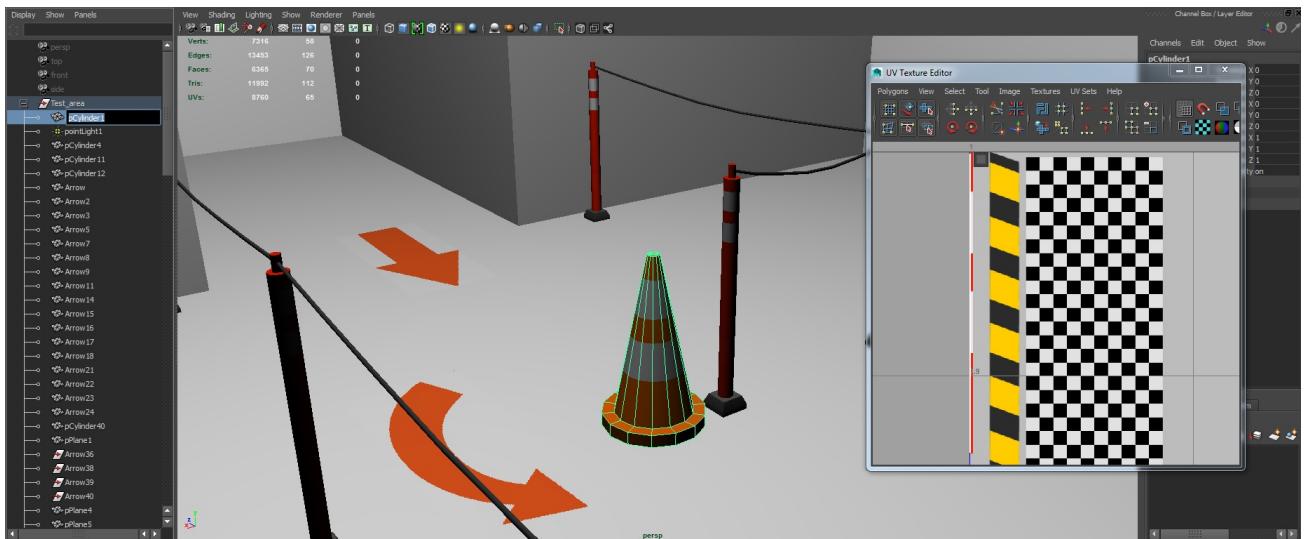


Figure 4.1: Picture of a cone from testing level

Every object that required texturing, also needed its own UV map. UV map is 3 dimensional model representation in 2 dimensions - it helps to see and paint texture using tools as Photoshop. Following picture shows UV map as red lines, that helps to see the boundaries for the texturing in Photoshop ??.

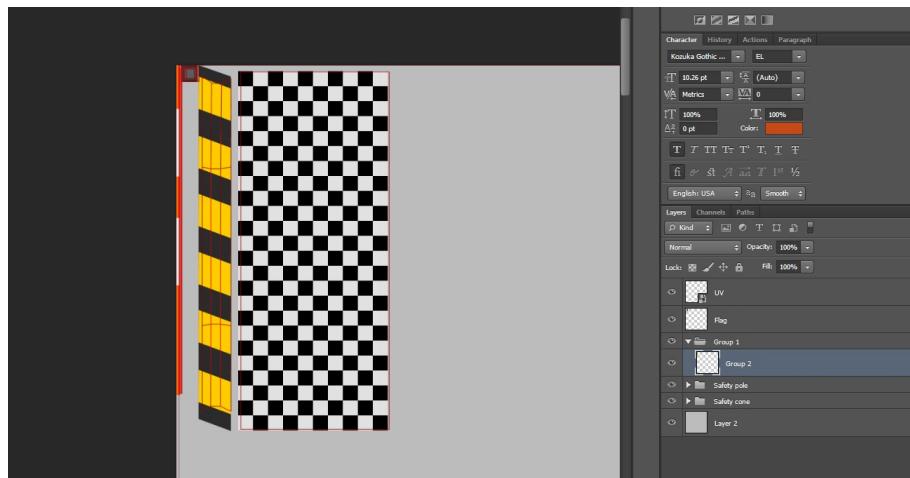


Figure 4.2: picture of a cone from testing level

Text, directional arrows, and other symbols needed to be transparent so it would blend well with most of backgrounds. Such images were exported as png picture format to keep transparency ??.

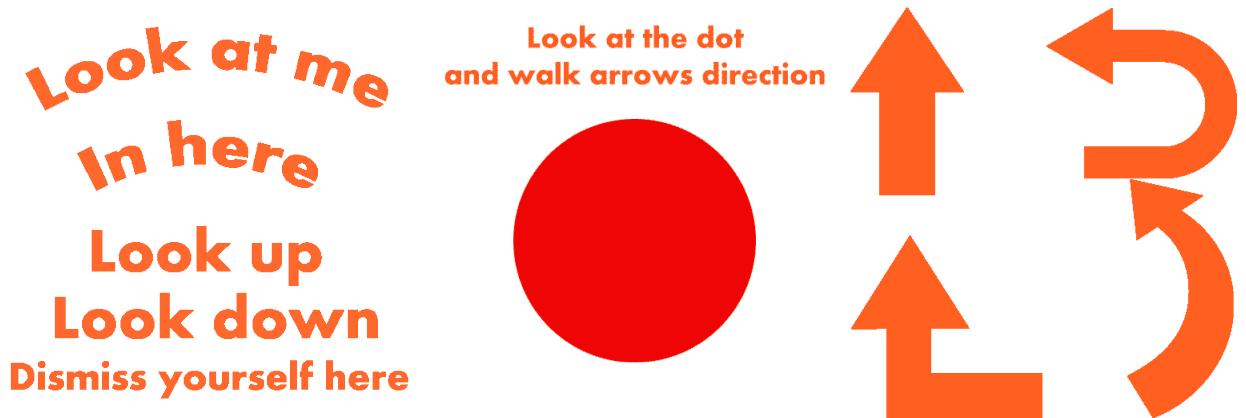


Figure 4.3: Pictures of elements that required transparency for game engine - also called sprites

Since graphics and aesthetics were not focus of this test level, textures were created very minimalistic - without any shadowing, noise, imitation of being old or used etc. It also helped to make test very optimized in hardware performance, since textures size on disk were as little as few kilobytes.

#### 4.0.11 UNITY INTRODUCTION

For this implementation the choice was made to use the freely available Unity3D engine, this was chosen as Unity provides a lot of features that would otherwise mean that this project would not be possible to complete in the amount of time given. This section will provide a quick glance at what Unity is and introduce some common terms used within Unity. Everything that is seen in a Unity game is composed of `GameObjects` these objects are what the scripts described in the following sections will be attached to. A newly created `GameObject` will only contain what is known as a `Transform` the transform is an object that holds the information about where the object is within the 3D space. It also holds the objects rotational information and its scale. the position of the object is expressed as a vector, a vector is a mathematical tool for representing a coordinate in space. As such vectors are the primary tool that is used in unity for expressing things like velocity, position, forces etc.

#### 4.0.12 BUTTONS

The buttons used in all three schemes are all created using unity's UI system. A button in the UI system is just an empty `GameObject` that holds the following components:

- **Rect Transform**

this component is equivalent to the regular `Transform`. However this component does not contain any information about the objects position in 3D space as it is part of the UI it only needs to know its coordinates on a 2D plane. The UI system provides a convenient way of making sure that UI will look the same no matter what screen size is used. It does this by having coordinates relative to a anchor point that is set by the developer.

- **Button**

this is the standard script that unity uses to detect clicks on a button and can call methods on different objects when clicked. This script is the script that has been modified to allow for pressing and holding.

- **Image**

this component is a component that will draw an image on the canvas this image will be the full size of the button

-

### ButtonScript

ButtonScript inherits from the in-built Unity Button object. This is done so that the script can access the method `protected bool IsPressed()`. This script is the script that all buttons in both the buttons and gyro scene uses. It is a simple extension as all it has to do is check if the button is being pressed, and if it is then check which button it is and act accordingly.

#### 4.0.13 BUTTONS SCENE

The button scene is the only scene that only uses `CamMovement` and `ButtonScript` and as such is the simplest scene technically. Presented below is the methods contained within `camMovement`:

- `public void move(bool rightButton)`

This method will depending on the boolean given to it move the camera forward or backward. This is done by creating a directional vector formed from the transforms forward vector:

```
Vector3 directionVector = new Vector3(transform.forward.x,
                                      0,
                                      transform.forward.z)
```

The directional vector will always have a y component of 0 as the camera should not be able to move in the y direction. This vector is then multiplied by a integer variable that will take the value of either -1 or 1 depending on the value of `rightButton` by converting the boolean into an integer with the line:

```
int goingForward = rightButton ? -1 : 1;
```

Finally the directional vector is multiplied with a variable for determining the speed of the movement. This directional vector is then set as the velocity of the rigidbody that is attached to the GameObject.

- `public void rotateLeftRight(bool right)`

This method will rotate the camera left or right in a similar manner to `move` but where `move` does not do anything to its y component the `rotateLeftRight` method will only rotate in its y component and keep the rotation of the x component. This method uses `Quaternion.eulerAngles` to convert our euler angles into Quaternions, and `Quaternion.Slerp` to smoothly make the camera rotate. This can be seen in figure 4.4

```
Quaternion target = Quaternion.Euler(transform.rotation.eulerAngles.x ,
                                      transform.rotation.eulerAngles.y + 2 * rotationSpeed * rotateRight,
                                      0);
transform.rotation = Quaternion.Slerp(transform.rotation, target, Time.deltaTime * smooth);
```

Figure 4.4: conversion of quaternions into euler angles and use of the slerp function

- `public void rotateUpDown(bool down)`

this function is more or less the exact same as `rotateLeftRight` but where `rotateLeftRight` makes changes to the y component of the rotation, `rotateUpDown` rotates the x component.

- `public void stopMovement()`

this function is a simple function to make all movement on the camera stop, it will set both the linear and angular velocity of the object to zero.

#### 4.0.14 GYROSCOPE CAMERA

the gyroscopic camera uses two scripts to work, first and most importantly the `GyroController` which is the script responsible for getting the input from the gyroscope and rotating accordingly. Unity can take input from the gyroscope directly, but if an object is rotated by the raw data from the gyroscope there will immediately be a problem as the coordinate system that is used to by gyroscope is a right-handed coordinate system

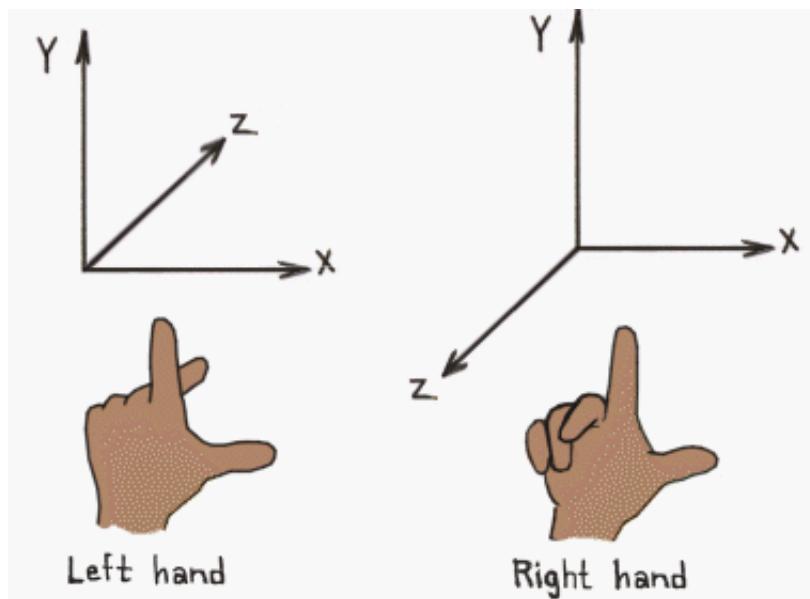


Figure 4.5: right-handed vs left-handed coordinate systems

where unity uses a left hand coordinate system. To have the coordinates given by the gyroscope match the coordinates of Unity the `GyroController` has a method for converting from right-handed to left-handed coordinate systems this method uses Quaternions which is how Unity handles rotation internally explaining quaternions is beyond the scope of this project besides the `GyroController`, this scene also uses the `CamMovement` and `ButtonScript` explained in the previous section, however in this scene only the forward and backward buttons are used, as the gyro is handling rotation of the camera.

#### **4.0.15 JOYSTICK SCENE**

The joystick scene is technically the most difficult as this involves The joystick scene uses some of Unity's standard assets, these include:

##### **MOBILE JOYSTICK**

The mobile joystick script creates two axis for unity's input manager. These two axis are then accessed by the first person controller and use these values to move camera.

##### **THE CHARACTER CONTROLLER**

this is actually a fully fleshed component, along with components like the transform, renderer etc. For this scene it is primarily used to generate a collider for the camera.

##### **FIRST PERSON CONTROLLER**

Unity provides a standard first person controller that comes with a whole range of features. For this scene the FPS controller needs to take input from the joysticks, to do this the FPS controller's `RotateView()` method has been modified to take input from the virtual axis created by the mobile joystick:

```
private void RotateView() {
    Vector2 input = new Vector2(CrossPlatformInputManager.GetAxisRaw("HorizontalLook"),
                                CrossPlatformInputManager.GetAxisRaw("VerticalLook"));
    float camX = m_Camera.transform.localEulerAngles.x;
    if ((camX > 280 && camX <= 360) ||
        (camX >= 0 && camX < 80) ||
        (camX >= 80 && camX < 180 && input.y > 0) ||
        (camX > 180 && camX <= 280 && input.y < 0))
        m_Camera.transform.localEulerAngles += new Vector3(-input.y * m_LookSpeed,
                                                          m_Camera.transform.localEulerAngles.y,
                                                          m_Camera.transform.localEulerAngles.z);
    m_YRotation = input.y;
    transform.localEulerAngles += new Vector3(0, input.x * m_LookSpeed, 0);
}
```

Figure 4.6: the modified RotateView() from the FPS controller

this method first reads the input from the axis named `HorizontalLook` and `VerticalLook` and stores these two values in a vector. This vector is essentially the directional vector we want to add to the cameras rotation. while a solution such as:

```
Vector3 dVector = new Vector3(input.x, input.y, 0);
transform.localEulerAngles += dVector;
```

will make the transform rotate in the direction of the vector, it will also cause the camera angles to get stuck. This problem is known as gimbal locking, which is the situation where two rotational axis is pointing in the same direction. To avoid this problem the script checks to make sure that the camera is not rotated into a gimbal lock and adds the rotation to the cameras local rotation. The script does this by checking if the user is looking down or up at an angle of less than 80 degrees.

# 5 | Evaluation

The evaluation is conducted to conclude whether the problem stated in this report is being solved. The answer will be acquired when the data gained from evaluation is analysed and concluded upon. This section will attempt to evaluate the control schemes and will cover methods and different approaches planned to be used when conducting the evaluation. It will be discussed why usability testing and performance testing methods were chosen for evaluation of the prototype and why it seemed the most efficient and suitable choice.

## 5.1 USABILITY TESTING

This testing method can reveal various errors that could occur when a user interacts with a product. Usability testing is a method that seeks to test five areas. [?]

- Efficiency How quickly can the user complete certain tasks when they know the design. [?]
- Satisfaction How satisfied are the users with the design. [?]
- Errors How many errors do the users make. Is it severe errors and how quickly will they recover. [?]
- Learnability How easy is it for the user to complete certain tasks when they first encounter the design.[?]
- Memorability How well do the user remember the design after some time away from the design. [?]

The benefits from usability testing is that it identifies major usability issues from a few number of participants. [?] According to Jakob Nielsen it is enough to have only five participants as the problems will show clearly based on this alone. [?]

There are several ways to conduct a usability test e.g. focus groups, user testing etc. The following will describe the methods we chose to carry out.

### **5.1.1 PERFORMANCE TESTING**

Performance testing is a method that focusses on, as the name reveals, performance. This can be measured in different ways. For instance, time and numbers of errors made. It allows the facilitators to obtain measures of effectiveness and efficiency. [?]

The benefits of performance testing is that it reveals major problems including problems related to the users skills and expectations. [?] Tell the user how to achieve the goal but not how to do it [] Observe and measure without commenting[]

### **5.1.2 CARD SORTING**

Card sorting is a method for discovering latent structured. [?] It allows the test participant to give critical feedback without having to do it directly to the testers. You will need fifteen participants for a card sorting test [?]. Normally a usability test only need five users but this testing method needs more participants to get a full view of the users preferred structure. This can not be accomplished by five participants. [?] As Jakob Nielsen says [?], this test differs from other usability tests by being a generative method. This means that we do not yet have a design and need to establish user needs first as we tested for the navigation before we designed the actual app.

According to Jakob Nielsen, the classic way to ruin a card sorting test is to give the user familiar command names. This will make the user look for that specific command name instead of acting as they normally would.[?] E.g. Do not say "now you will use a joystick" and thus giving them the information that this should be controlled as a joystick and the usability problems there might be will not be revealed. It will not be certain if the test participant actually came to the conclusion that it should work as a joystick themselves.

Card sorting can be used in various ways. To group words, to name groups of words, to describe a product and more. We used it to make the test participant feel more comfortable

choosing a critical word rather than interviewing them. Here is how we used it together with the performance test.

## 5.2 FIRST TEST - NAVIGATION

The first thing that is tested is the navigation. This needed to be established before making the actual design and to make sure that the app is usable. It is a usability test that will consist of two parts; performance and sort-carding. The test is set in a controlled environment where the participants need to go from A to B, testing the control schemes, rotating the order of what control scheme they start out with.

Before the test starts, the test participants will watch us go through the level so they know where to go. This should eliminate some bias when it comes to getting to know where to go and should help keep the focus on how to get there. Otherwise the first control scheme would be slower every time as they would have to find their way through the level.

We will time them to see which control scheme is the easiest to control. We will also observe how much they struggle and how fast they get from A to B. Also we will estimate how long time it takes them to have a grip on the controls.

After each control scheme we will ask them to do a card sorting. They pick 5 words and afterwards they would be asked why they chose those particular words. This will help us getting them to say something critical about the controls to us.

We will make a scoreboard so the test participants can see how fast they completed the course compared to the other people who tested it. This should add a competitive element and some fun to the test.

### 5.2.1 ANALYSIS OF DATA

The first thing that was done after testing, was color coding the cards. This way we could sort them out into categories. Then we made bar charts to get a visual and clear image of the results from the card sorting. The cards were sort after:

- Intuitive/familiar UX
- Positive feelings
- Negative feelings

- Complaints
- Unusable. These cards were not related to the control schemes.

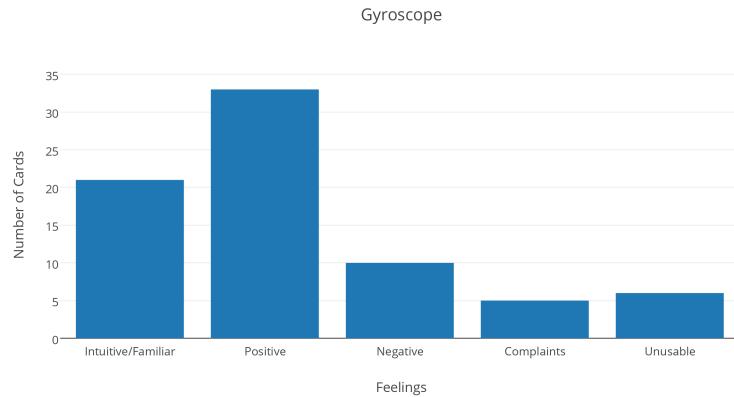


Figure 5.1: Categorisation of the cards for the gyroscope control scheme.

The gyroscope turned out to be very intuitive and quite fun for users. It has a small amount of negative feelings and a few complaints. These were mostly based on the fact that you have to turn all the way around and not being able to sit down while using this and that it moved to slow. The intuitive cards that were selected showed that the users thought it was nice that their movements was met in a virtual world. Also they thought that it was easy to understand once you have practised a bit. The positive feelings towards the gyroscope was quite clear; it was fun, exciting and friendly. Almost all positive cards had something to with the fact that this was new and that the users had not yet experienced this which made it more attractive.

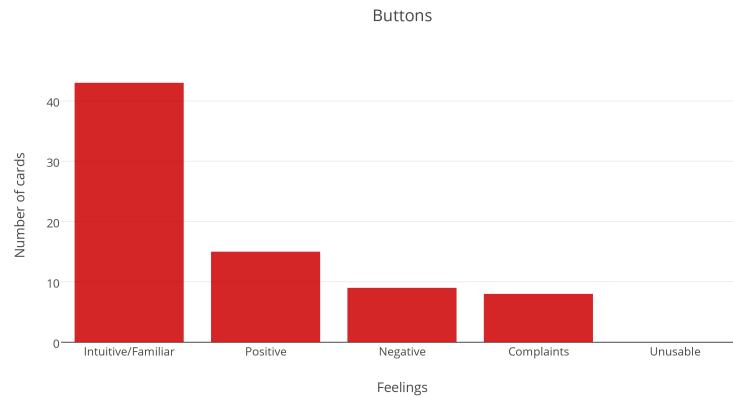


Figure 5.2: Categorisation of the cards for the button control scheme.

The button control scheme has the absolute highest number of intuitive/familiar feelings. This makes sense since it is common for users to use buttons. It has about the same amount of negative feelings and complaints as the gyroscope. These feelings were however quite interesting. 5 out of 9 negative cards said dull or boring. The users found that this was too common and had been seen before. It was clearly the most efficient as the intuitive and positive feelings were all directed towards the speed, how easy it was to use and that it was trustworthy. There is no way we can actually improve on the negative remarks because if the buttons are too last year our hands are tied.

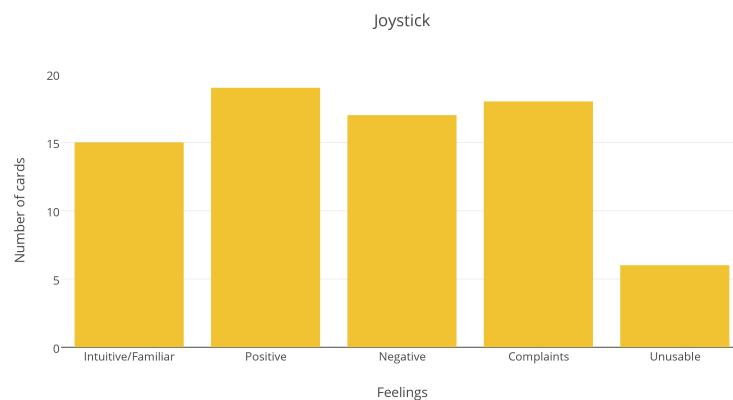


Figure 5.3: Categorisation of the cards for the joystick control scheme.

Lastly, the joystick has a lot of negative feelings and complaints. Even more than it has intuitive/familiar feelings. This clearly shows that the users did not enjoy using this controlling

scheme as much as the others and was quite frustrated with it. This could also be provoked by the way we chose to implement this and the fact that it did not resemble a joystick well enough. The main issues that were pointed out to us in the negative feelings was that it was frustrating, hard to control. The users commented on the camera the most. Also 8/18 complaints were directed towards the cameras movement and that the joystick lacked sensitivity. These negative feelings could maybe have been avoided if the implementation had been better. The positive feelings however was flexible and creative. The users thought it was fun, once you get to know it. The few that did give this control scheme a intuitive/familiar card thought that it was very clear that this was a joystick but still struggled to control it due to the cameras movements.

The joystick might not be as bad as it appears in the bar chart. The main reason for the high number of complaints and negative feelings was that the cameras movement was out of control. This could have been prevented by better implementation. So as far as the joystick goes, we will not know if this is the reason for the feedback until we can fix it and test it out once more.

The time from the performance tests was calculated to find the mean and made graphs to demonstrate this as well.

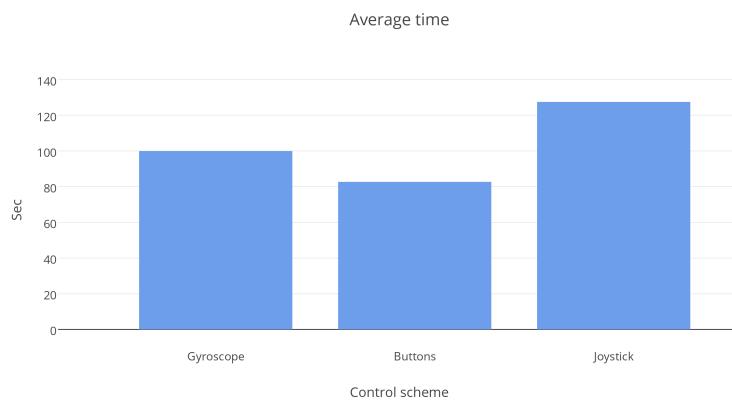


Figure 5.4: The average time it took for the participants to complete the level within the different control schemes.

It is fairly easy to conclude that the buttons were the most efficient and the joystick the most problematic for our users based on the time alone. But is the buttons the most successful after all?

It shows very clearly that the users had most intuitive/familiar comments about the buttons scheme. And the one they had the hardest time coping with was the joystick. Both the performance test and the card sorting test showed that the fastest, most intuitive and most efficient was the buttons scheme. However, their emotional response to this was not as we hoped. The one controlling scheme the users found the most exciting was the gyroscope. They did have some complaints, but these were mostly based on the fact that there was a learning curve and some technical issues that can be fixed.

If we were to compare the gyroscope and the button scheme for final conclusion the users had 58 positive feelings/intuitive about the button scheme and 17 negative feelings/complaints. Whereas the gyroscope had 54 positive feelings/intuitive and 15 negative feelings/complaints.

### **5.3 CONCLUSION OF RESULTS**

In conclusion, we now know that we will not be using the joystick. Whether we choose the gyroscope or the buttons ultimately depends on our focus. Do we want a control scheme that is exiting, new and fun or do we want one that we know for sure will cause the users no usability problems and be the most efficient although they think it is dull? The final problem statement is interested in making an application that minimizes the knowledge gap using familiarity. Also, the user experience section suggest that when people think something is fast, it is good UX. ?? And furthermore, it is concluded that we want an app that is easy to remember and learn how to use. ??

Based on this, we can conclude that the buttons is the optimal choice for our navigation app. Even though the users were more emotionally engaged in the gyroscope, the focus for this app lies within familiarity and not excitement. And even though the negative cards that were chosen in the card sorting suggested that the users found this boring, it still sets the bar high with 43 cards chosen on familiarity and intuition.

# 6 | Discussion

The testing of this implementation, have a few points that should change, specifically four points have been identified that has had a negative impact on the test. these are:

- Test environment
- Tested users can be classified as gamers
- Competition

these first section of the discussion will focus on explaining and solving these problems. Next the discussion will look into the main points that can be used to make the redesign better. the discussion will then look at some of the choice made along the way for which control schemes that were implemented and look at what could be alternatives. Finally the discussion chapter will look into some of the technical difficulties that the Joystick control scheme experienced(more on that later) as well as look into what could have made the test level better suited for testing navigation.

## 6.1 TEST ENVIRONMENT

During the testing of this prototype the environment was not taken into account as it was done in the team's group room, this introduced some problems that should not have been present, for instance: two test participants were brought to test for the team, these were not separated during their individual tests and the testers were not told not to talk during the test. this might have affected their responses as well as their completion time. This could have been fixed by having the participants test the app in one room and do the card sorting in another. If the tests had been carried out in two rooms, one room could have held a one-way mirror where

the testers could have observed the test participants from, and also filmed. this could have allowed for an in-depth analysis of the emotional responses from the participants. Tested users are gamers An issue that was not taken into account before the test was carried out was the fact that many of our test participants were not only digital natives but gamers. This meant that our target group was not well represented. They may have had too many skills than we anticipated which may have affected some outcomes in the card sorting test e.g. that the buttons were boring, that they are used to using a joystick and have a smaller knowledge gap than the digital natives.

## 6.2 ANOMALIES IN THE TEST DATA

When looking at the data gathered from the tests, 2 test subjects stand out as anomalies. While the joystick is generally slower than the other 2 control schemes, there was one test participant who spent nearly twice as long as the second slowest test participant. The deviation from the norm in this test was so large that it single handedly increased the average time by close to 25%. Another anomaly we noticed was on the gyroscope. One of our test participants ran into some technical issues during the test, to which we could attribute his overwhelmingly negative comments. This test participant accounted for 40% of the gyroscopes negative comments. One must note that the first of these anomalies CAN be eliminated since it's part of our quantitative data, while the other can not since it's qualitative. Even though it accounts for 40% of the negative comments relating to this control scheme

## 6.3 COMPETITION

The test was designed as a competition in hopes of motivating the user to do their best ie. as fast as they could. This worked well during the test, but it later showed that the test participants might have been too focused on the competition rather than the navigation. This could also be because they were not instructed properly before the card sorting. However, it did affect the card sorting results as we can see many chose cards like "fun" or "engaging" and reasoned it with the fact that it was fun because it was a competition. We might still have been able to pull this off if we had given more clear instruction to what they should base the cards on.

## **6.4 TEST DATA TO REDESIGN**

In order to test whether the button controls IS actually the most efficient, another test should be done where the test participant get a chance to play around with the controls before being timed. This should eliminate or at least reduce the knowledge gap, allowing the test to focus only on efficiency instead. While the joystick controls received mostly negative feedback there were some positive comments though, particularly on the fact that it allowed the user to move sideways instead of just back and forth. This feature is something that shall be brought into the next iteration.

## **6.5 CONTROL SCHEMES CHOICES**

## **6.6 TECHNICAL DIFFICULTIES - CHOICES AND PREPARATION**

When looking at the results for the joysticks it's obvious that these were the least popular. However, there were a lot of complaints about technical issues with the joystick controls, which means it might be necessary to calibrate it and test it in the second iteration as well.

## **6.7 LEVEL DESIGN - OBSTACLES ARE LOWER THAN THE CAMERA HEIGHT. THIS RESULTED IN COLLISION ISSUES**

# Bibliography

- [Ciotti, 2013] Ciotti, G. (2013). The psychology of colour in marketing and branding. <http://www.helpscout.net/blog/psychology-of-color/>.
- [Developers, ] Developers, A. Motion sensors. <http://www.smashingmagazine.com/2010/11/04/best-practices-of-combining-typefaces>.
- [Examstutor.com, 2015] Examstutor.com (2015). Psychographic segment. [http://www.examstutor.com/business/resources/studyroom/marketing/market\\_analysis/8\\_psychographic\\_segmentation.php](http://www.examstutor.com/business/resources/studyroom/marketing/market_analysis/8_psychographic_segmentation.php).
- [Greenfield, ] Greenfield, S. Modern technology is changing the way our brains work. Daily Mail website. <http://www.dailymail.co.uk/sciencetech/article-565207/Modern-technology-changing-way-brains-work-says-neuroscientist.html>,.
- [Gube, 2010] Gube, J. (2010). What is user experience design? overview, tools and resources. Smashing Magazine. <http://www.smashingmagazine.com/2010/10/05/what-is-user-experience-design-overview-tools-and-resources>.
- [Neil, 2014] Neil, T. (2014). *Mobile Design Pattern Gallery*. O'Reilly Media, Inc., 2nd edition. ISBN:8349857349687.
- [Nielsen and Budui, 2015] Nielsen, J. and Budui, R. (2015). *Mobile Usability*. The Nielsen Norman Group. ISBN:978-0-321-88448-0.
- [Norman, 1988] Norman, D. (1988). *The Design of Everyday Things*. DOUBLEDAY.
- [Preece et al., 2015] Preece, J., Sharp, H., and Rogers, Y. (2015). *Interaction Design: Beyond Human-Computer Interaction, 4th Edition*. Wiley.

[Prensky, 2001] Prensky, M. (2001). Digital immigrants , digital natives. <http://www.marcprensky.com/writing/Prensky%20-%20Digital%20Natives,%20Digital%20Immigrants%20-%20Part1.pdf>.

[Prensky, 2009] Prensky, M. (2009). From digital immigrants and digital natives to digital wisdom.

[Raskin, 1994] Raskin, J. (1994). Intuitive equals familiar. Technical report, Communications of the ACM.

[Robinson, 2013] Robinson, K. (2013). 3 keys to develop a faster, more intuitive user experience. Openview Labs. <http://labs.openviewpartners.com/intuitive-user-experience-design/>.

[Sardo, 2009] Sardo, G. (2009). Building mobile user experiences.

[Spool, 2005] Spool, J. M. (2005). What makes a design seem 'intuitive'? [http://www.uie.com/articles/design\\_intuitive/](http://www.uie.com/articles/design_intuitive/).