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

1 Introduction

The goal of this project is to create an application which allows the user to navigate in a 3d environment. The intention is that the application will be used as part of an interior design process, where shoppers can set up the environment, add the décor and experience the room fully decorated, before actually paying for the furniture and setting it up. The navigation-application will be developed with a focus on a pleasant user experience and intuitive navigation. As such we will be attempting several non-traditional interaction methods and through focus groups evaluate which methods provide the user with the most intuitive experience. During the early brainstorming sessions the idea was to make an application that would help real-estate agents showcase the houses that they were selling. Looking into this however showed that there was no need for this since what this application does would mostly take away from the need for a real estate agent and would likely result in a target group unwilling to cooperate. Instead we chose to focus on shoppers at IKEA. The reason for this being that it's a discount furniture store where people often go to furnish a whole room or apartment/house. In addition, IKEA has already made a similar application for interior design which shows that there is a market for this.

1.1 Initial Problem Statement

How can we improve user experience in an interior design app 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. [Examstutor.com, 2015] 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" [Prensky, 2001] 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 [Prensky, 2001]. 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.[Prensky, 2009]. 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[1] 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:

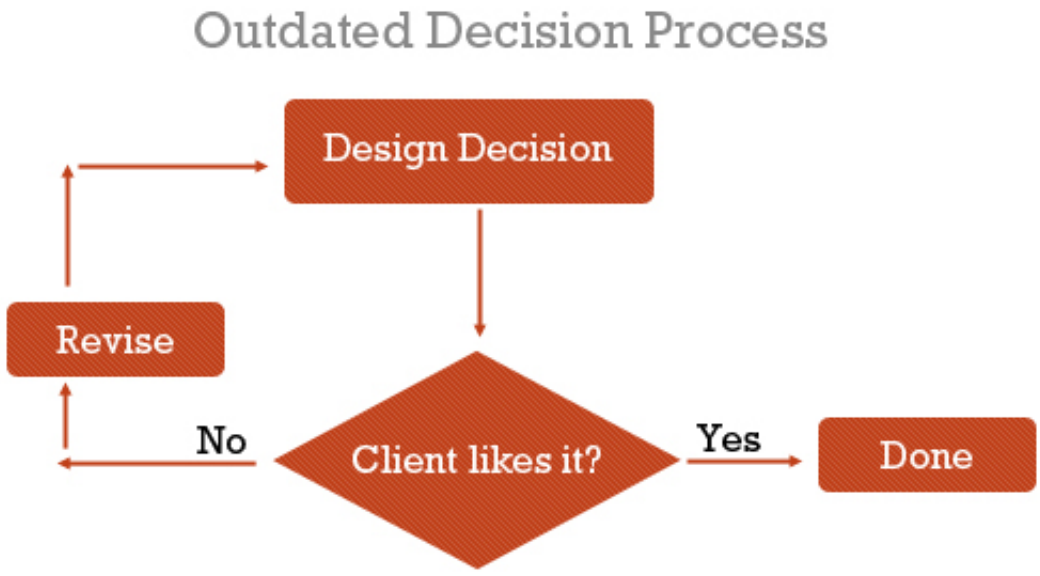

Figure 2.1: old decision process, Jacob Gube 2010

[1] 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:

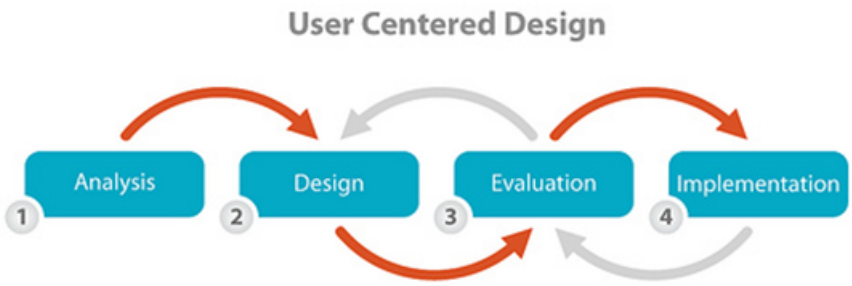

Figure 2.2: a chart of how user-centered design could function, Usabilla 2014

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

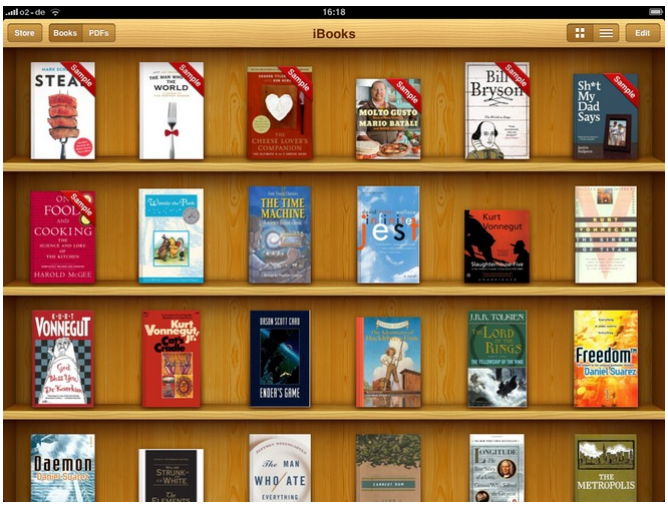

Figure 2.3: Apple iBooks for comparison

might need a source reference

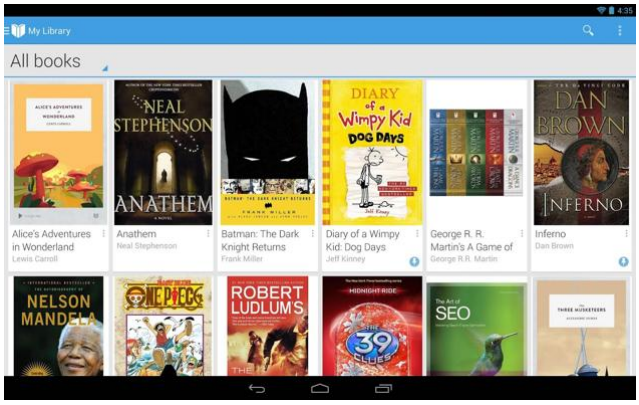

Figure 2.4: Google Play Books for comparison

might need a source reference

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.

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*[Spool, 2005] 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.

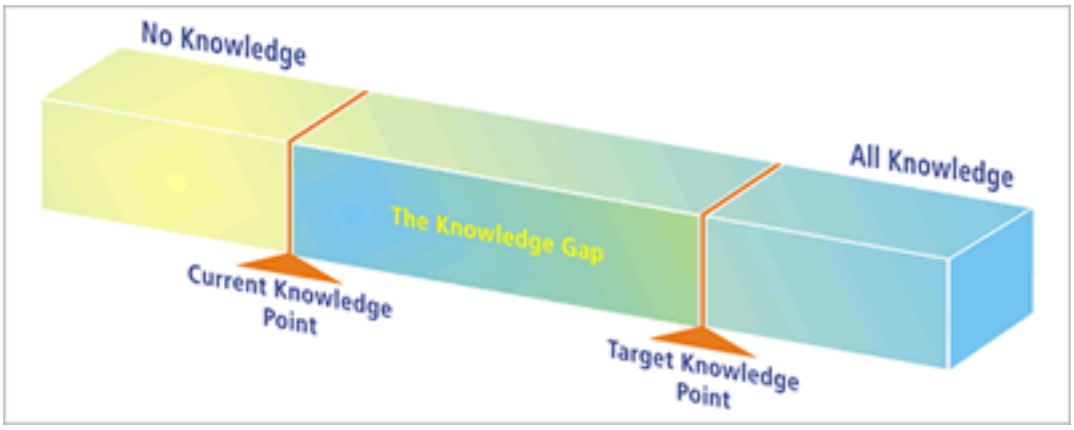

Figure 2.5: the knowledge space viewed as a continuous curve going from *no knowledge* to *all knowledge* [Spool, 2005]

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.

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," [Robinson, 2013] 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[Robinson, 2013]:

  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 [Preece et al., 2015] 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". [Norman, 1988] 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.

[Norman, 1988]

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 aspect to consider. Some users might have limited mobility or problems with manual dexterity - it will cause higher error rates with the interaction [Preece et al., 2015]. 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 10. 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.

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

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.

also not sure if this part is necessary here

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 [Greenfield,] . 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 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.

for some reason cite doesn't work

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 [Developers,] that can measure orientations on X, Y and Z axis, allowing the apps to calculate placement of the device in the 3D environment.

reference error again

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.

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.

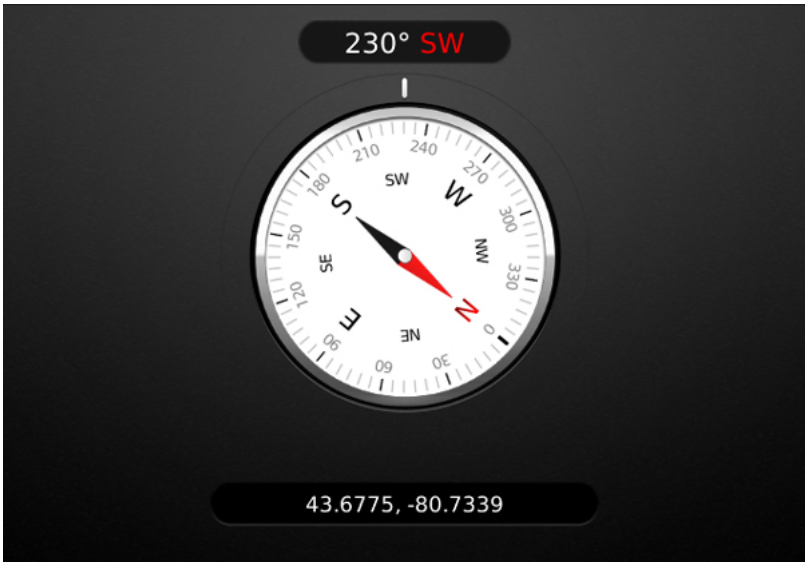

Figure 2.7: a simple compass app that establishes the magnetometer sensor

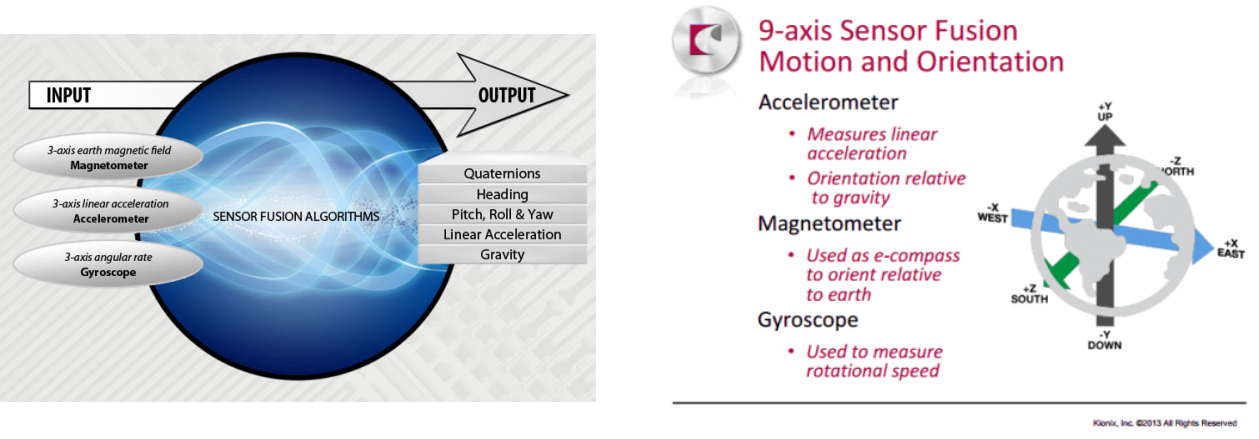

Figure 2.8: uses of sensors

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 complaint about in applications is a confusing interface design and poor navigation. [Neil, 2014] This can be prevented by using design patterns. These has been created to avoid a messy and untolerable 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 over lapping 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 to 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 percieved. [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 to soft and feminine but at the same time not to 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 succesfully 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.

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 is 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. 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 therefor 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 AutoDesk's 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. 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

fix reference to reference to Idas section on the isolation effect

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.

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.

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

The goal for the prototype is to define which way to control the 3D virtual environment is the most familiar one for the user and also the most intuitive one. As mentioned in User Experience chapter (**??**), familiarity is very related to intuitiveness.

For the project to answer the question of minimizing the knowledge gap through the concept of familiarity, several design requirements need to be established:

- Uphold one of the two conditions from the knowledge gap (section **??**)
- Make UI elements stand out through the use of isolation (section **??**)
- Make use of gyroscope in terms of reflecting real life movement
- Stick to one color scheme
- Limit amount of information given to the user

In the following section, these requirements will be covered.

The prototype needed to have two features to navigate in 3D virtual environment - the moving of the character and the camera movement/rotation. The familiarity concept was put into effect as the control schemes for controlling the environment had to be linked with something that the user might be familiar with already. It was chosen to make extremities of the control schemes to establish familiarity in distinct ways.

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 and intuitivity through the non-traditional sensors, two different approaches to familiarity were established. The knowledge and familiarity to the currently existing products on the market (State of the Art, **??**), 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.

2.7.1 Control Schemes

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 an in-built gyroscope in the tablet is familiar with a natural way for a person to look around - by turning the direction user wishes to move. The movement forward and backwards was implemented with on-screen buttons as these were familiar to the target group through the usual daily tasks, since most of the controllers for movement are represented as such (e.g. arrows on the computer keyboard, music player, cell phone).

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 to 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.

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.

2.7.2 Immediacy and Simplicity

To communicate information in a simpler, and faster to perceive manner, the designs will be represented by concepts that are already familiar to the user. This means, that to communicate information to the user, graphical elements will 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. As discussed in sections (?, ?, ?), this will further shape the familiarity and intuitivity for the application.

(need ref from the upcoming usability section)

references

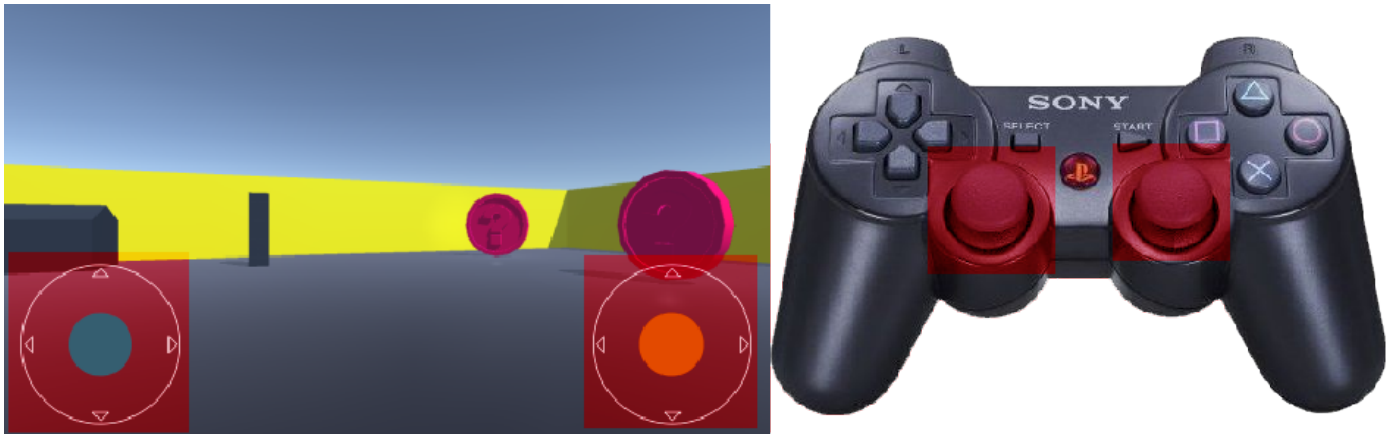

Figure 2.11: Comparison of initial design sketch of on-screen joystick and the Sony Playstation controller

2.7.3 Graphical element sizes and placement

To further emphasize on the user experience, the button size should be set accordingly - to ensure that the users would not have difficulties by unintentionally tapping the wrong section 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.

(ref to UX tetraplegia stuff)

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.

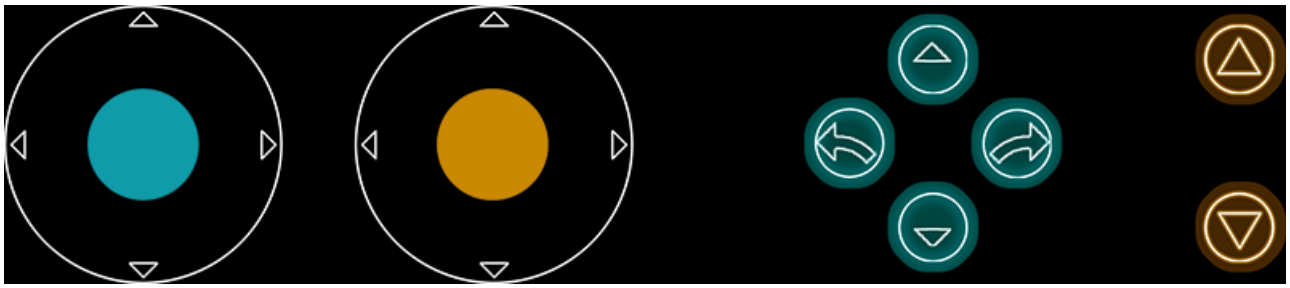

Figure 2.12: Initial sketch. Color differences to distinguish controllers that hold different functions

2.7.4 Isolation

To further emphasize on the initial design requirements, the concept of isolation (**??**) should be used when designing the controllers - this should be 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.

2.7.5 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 <FOR MOBILES>

when designing a mobile app with UX focus, the unique challenges of the mobile platform has to be considered. A brief look at three of the most outstanding challenges:

- Screen size
  as opposed to a traditional computer screen the general mobile platform has a much more limited amount of screen space. This restriction will force the designers to eliminate as many redundancies as possible so as to not clutter the screen with unnecessary information. [Sardo, 2009]

- User input
  user input is according to Giorgio Sardo one of the smartphones weakness. it is mentioned that "Entering text on a mobile phone is hard, and people tend to avoid it if they can"[Sardo, 2009]

- Loading times
  Mobile devices are generally slower than a PC or Mac, both when it comes to processing power and internet speed, assuming they're using a mobile network [Nielsen and Budui, 2015]

Some of the guidelines for optimizing for mobile devices are cutting features, reduce word count and enlarge interface elements to accommodate the "fat finger problem".[Nielsen and Budui, 2015 an example of poor simplification used in the book is IKEA where they simplify the mobile site by only showing a single item when browsing for bedframes.

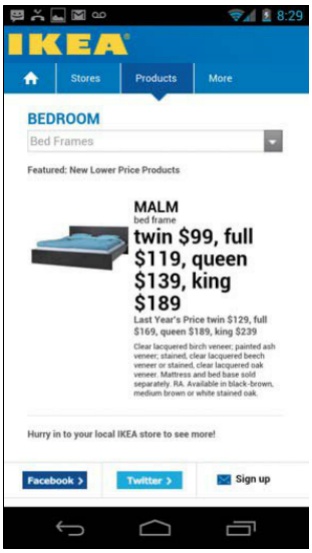

Figure 3.1: the mobile website from IKEA, anno 2013

4 | Implementation

4.0.3 Implementation of 3D testing area

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

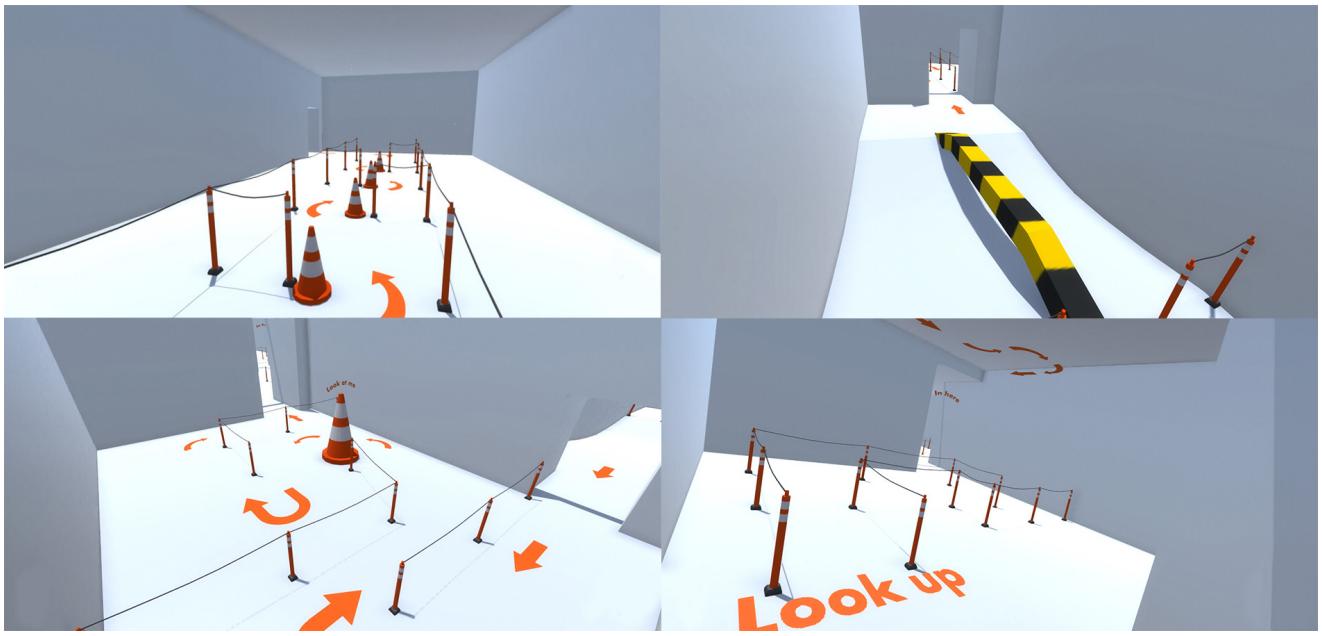

Figure 4.1: Pictures of the testing area

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.

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. Next two levels were developed to test how efficient it is to look up and down while walking in given direction. it represents looking at the lamp, carpet or any other element that is placed above or below user. 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.

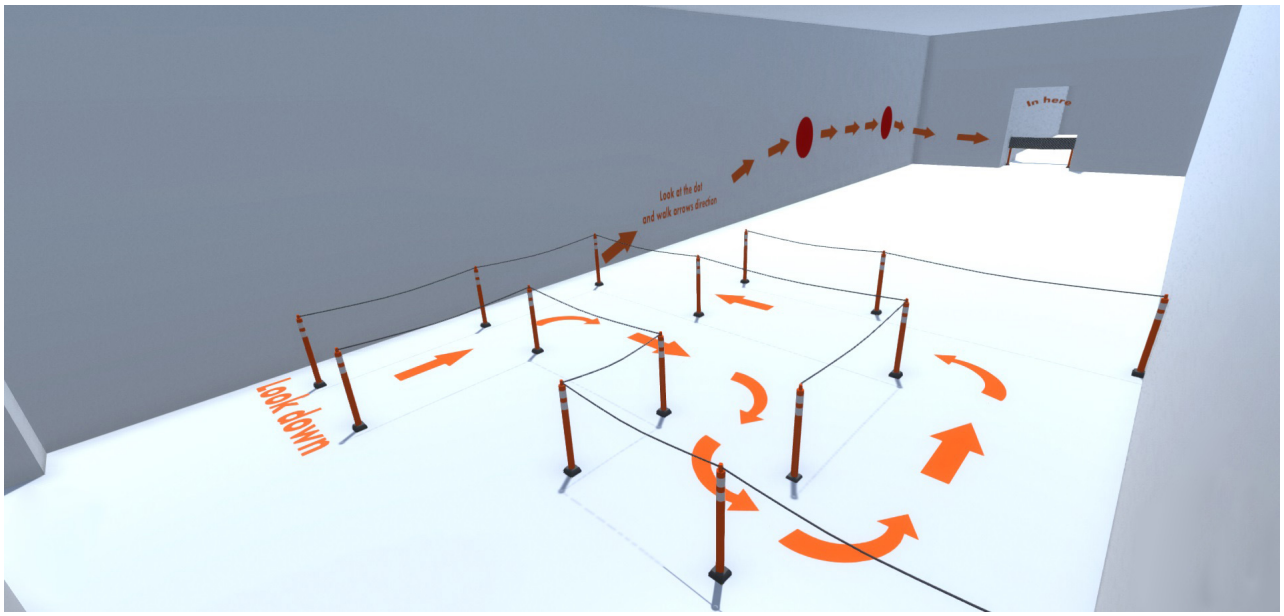

Figure 4.2: Last level of testing area. This level is to test how well user can walk while looking down and sideways

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 bellow shows 3D cone and it’s texture.

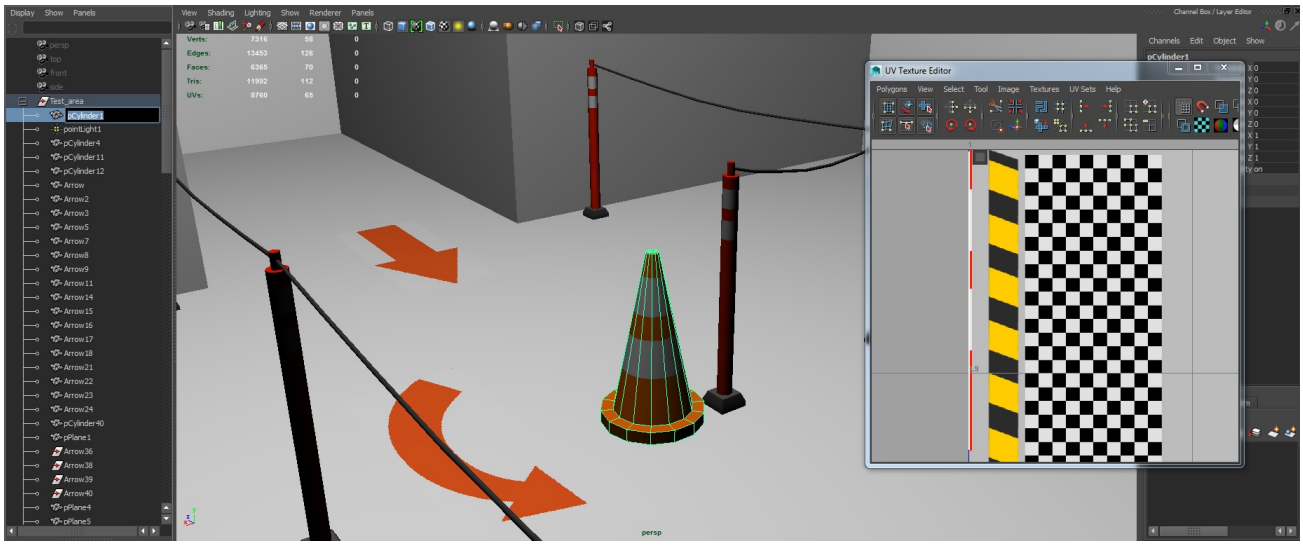

Figure 4.3: picture of a cone from testing level

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. 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.

4.0.4 Unity Implementation

4.0.5 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

4.0.6 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 to anything to its y component the `rotateLeftRight` method will only rotate in its y component an keep the rotation of the x component. This method uses Quaternion.euler 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.

4.0.7 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

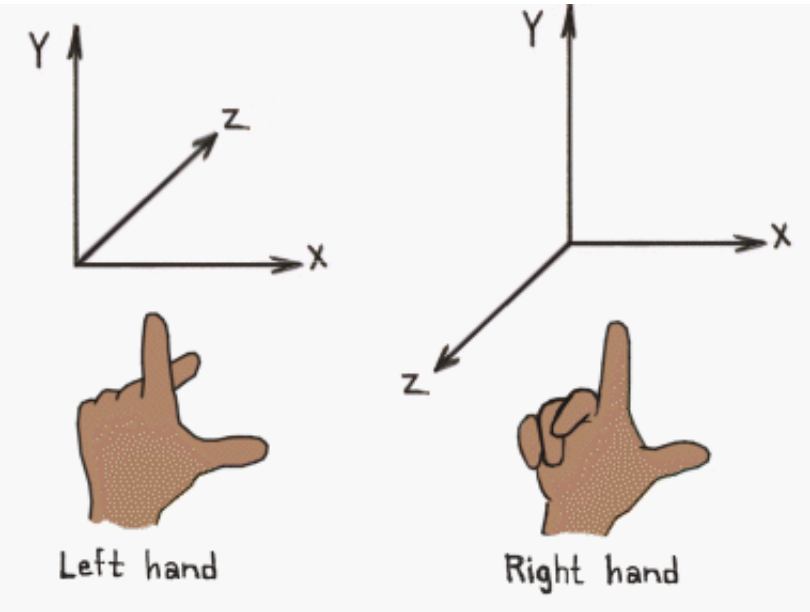

Figure 4.5: right-handed vs left-handed coordinate systems

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.

4.0.8 Joystick Scene

The joystick scene is technically the most difficult as this involves The joystick scene uses some of Unitys 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.

Bibliography

[Bonneville, 2010] Bonneville, D. (2010). Best practices of combining typefaces. http://www.smashingmagazine.com/2010/11/04/best-practices-of-combining-typefaces.

[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.

[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/.

[Hampton-Smith, 2014] Hampton-Smith, S. (2014). How to master colour theory. http://www.creativebloq.com/colour/colour-theory-11121290.

[Mayer, 2010] Mayer, D. (2010). What font should i use?: Five principles for choosing and using typefaces. http://www.smashingmagazine.com/2010/12/14/what-font-should-i-use-five-principles-for-choosing-and-using-typefaces.

[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/.