CHALMERS


# A Natural User Interface and Touchless Interaction Approach on Web Browsing

Master of Science Thesis in the Master Degree Programme, Interaction Design & Technologies

GEORGIOS LAFKAS

Report No: 2013:143
ISSN: 1651-4769

Department of Applied I.T.
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden, 2013

# Abstract

The aim of this Master's Thesis is to explore the principles of Natural Interaction and suggest an implementation of a web browsing application that uses them.

The research that takes place is examining the ingredients of Natural User Interfaces and Touchless Interaction with a focus on Gestures and Voice. How can those be combined to control an interface? What are the most important things to mind when designing Natural Interfaces and Interactions? These are questions that this report attempts to answer.

The final result of the thesis is a prototype web browser application that works only with gestures and voice. The prototype simulates browsing on a number of created web pages. It is implemented using the Microsoft Kinect for XBox sensor and Kinect SDK v1.7 with all the included tools and interaction controls. The browser is undergoing a usability test and the participants are observed and later interviewed to describe their experience.

Based on that, the conclusions drawn are that designing an application that uses a Natural Interface needs to provide short and comfortable Gesture and Speech Vocabularies and Feedback mechanisms that inform the user at every point of the interaction.

Furthermore, it is concluded that the design of the content for such an application should be different than the traditional design of a webpage for optimal use while responding to implicit input (context-awareness) adds to the naturalness of the interaction. Finally, the setting for using such an application is recommended to be part of a home entertainment system (e.g. smart TV) or public space installation rather than a desktop computer.

Keywords: human-computer interaction, natural user interfaces, touchless interaction, gestures, speech recognition, touchless web browsing.

# Table of Contents

# Introduction

With the arrival and gained popularity of hardware that supports *Motion Sensing* (Nintendo Wii, Microsoft Kinect, Sony Playstation Move, Leap Motion) the past few years, there is growing interest in the different ways this technology can be used in.

Although the first applications of it is for entertainment purposes (video games) there is now a larger number of scenarios where motion sensing can be employed. These scenarios include the implementation of software in a variety of fields, such as design-related (painting, 3D model design), simulations (for learning or training purposes) and file browsers (e.g. image galleries).

Motion sensing and its derivatives, like gesture recognition, present an alternative way of communicating with software that falls under the umbrella of *Touchless Interaction.* It is the kind of interaction where the user does not get in touch with any sort of input tool and thus receives no form of tactile feedback. Apart from motion sensing, there is another aspect of that sort of interaction, that of *Speech Recognition*. Voice is another channel of communication that can prove useful in the effort to create an experience that will feel natural and intuitive.

*Naturality* and *intuitiveness* are terms very important when speaking about interaction and interfaces, especially modern ones that have evolved much over time. Part of this evolution is taking advantage of the aforementioned technologies. This however must be done in a way which, apart from natural and intuitive, is also usable and efficient. A way that users can familiarize themselves with as fast as possible but that also minds the content and purpose of the application. That constitutes a big step towards bringing *Natural User Interfaces* (NUI) closer to becoming popular and essentially creating an experience that could in specific conditions replace the usage of the traditional input devices, like the keyboard and the mouse.

## Problem

The acceptance of Natural User Interfaces in combination with the anticipated application of Touchless Interaction ideas is going to create challenges in the way user and application behavior is designed. Frequently used types of software are going to face most of these challenges since they are aimed towards a larger number of people of very different backgrounds and computer usage skills. Usable, easy to learn and perform solutions are needed for this new kind of interface to be accepted and have a chance to be used in a sensible manner.

## Goal and aim

The aim of this report is to investigate principles of natural interaction and provide conclusions for how they could be applied to a Natural User Interface. For the purposes of making a more practically-oriented study, a prototype web browsing application is created and tested that provides useful findings to solidify and further the research. That application fits the description of a frequently used piece of software that is open to modification according to the research.

## Research question

Essentially, this report is an answer to the question of what design changes, additions or subtractions commonly used applications would need to go through in order to be well suited for touchless interaction.

## Structure

The outline of this report is the following: First a more in depth reference to the background of the research question is given. Then a theoretical overview regarding important concepts of the research is presented, like gestures, semiotics, speech and natural interaction. There is an overview of the methodology used to conduct this research later, while afterwards the reader can find out how the implementation of the project was carried out, specifically the prototype. The results of the prototype creation and test are shown and an analysis of those is performed, in terms of functionality. Later comes a discussion where the prototype is motivated and evaluated together with the process. Future development is also discussed. Finally the conclusions drawn from the study are presented, where the main question posed here is answered in detail.

## Involved Parties

The work presented in this report is carried out in collaboration with *Humblebee*, a digital agency located in Gothenburg and operating in the area of digital production and Web development with an interest in technical innovation. Their interest lies, among other things, in the usage of modern motion technology for tasks that have not been used with those means in the past and the subsequent user experience. The company acted as a speaking partner throughout the whole process. They were offering feedback on the concept during its evolution in regards to functionality and interaction. The company's offices was a working space in the beginning of the project for the brainstorming sessions and the literature review.

# Background

In this section the reader is introduced to topics that are related to the subject and presented with some notable work in the area.

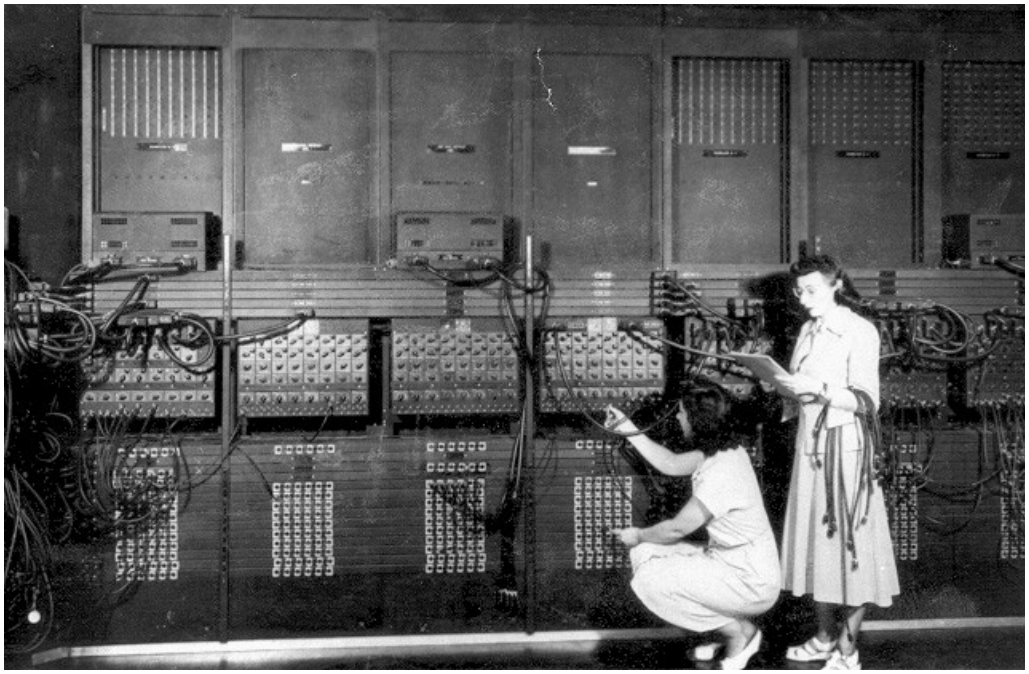
## User Interfaces

A *User Interface* (abbr. UI) is the framework of communication between the application and the user and consists of all the elements that allow the user to issue commands and requests to the application. The application in turn, makes use of the interface to organize the content it contains, prompt the user for some action and provide feedback to the user as to how commands and requests are progressing and what their result is. According to the type of UI, different elements might be used to achieve the above or even omit them.

User interfaces evolved as part of the broader computing evolution, when advances in hardware and software allowed it. Those advances include the more powerful CPU chips, graphic acceleration capabilities, direct input devices (mouse, keyboard) and the appearance of multi-purpose *operating systems* (OS) that would make computers fit for a wider audience (Curtis, 2011). Below there is a short reference to the several types of interfaces that were present at different times during the aforementioned evolution.

## Computer Interfaces (CI)

The first computers were not interactive (Curtis, 2011). Their users had to use *punched cards*, pieces of paper with holes in them, which were placed in special cartridges to perform some logical operation. That was what those computers received as input. The result of the logical operation, the output, came usually in print. The whole interaction could last from hours to days, thus it was not real time.
An example of an interface on those first days of the digital computers is that of the famous ENIAC, the first digital computer. ENIAC (Electronic Numerical Integrator and Computer) occupied more than $63m^2$ of space and weighed nearly 25 tons. Interaction with that giant device took place by manipulating the hardware itself, since that computer did not have the ability to run stored programs; the user had to program it every time. The hardware included cables, wires, vacuum tubes and switches. The user, a programmer, had to rewire and change the position of a number of switches in order to give the necessary commands. The output devices were a printer and punch cards (Moore School of Electrical Engineering, *1946*).



*Two programmers at work with Eniac (image from Penn Libraries)*

## Command Line Interfaces (CLI)

A *Command Line Interface* accepts written text as input for the application running on the computer. There is a "dictionary" with available text commands that the application can recognize and respond to. The user types in the command and presses a key, usually enter, to issue the command. The application then responds by executing something or presenting a result, according to its nature and type of command.

```
Microsoft(R) Windows DOS
(C)Copyright Microsoft Corp 1990-2001.

C:\>mem


    655360 bytes total conventional memory
    655360 bytes available to MS-DOS
    578352 largest executable program size

   4194304 bytes total EMS memory
   4194304 bytes free EMS memory

  19922944 bytes total contiguous extended memory
         0 bytes available contiguous extended memory
  15580160 bytes available XMS memory
           MS-DOS resident in High Memory Area

C:\>
```

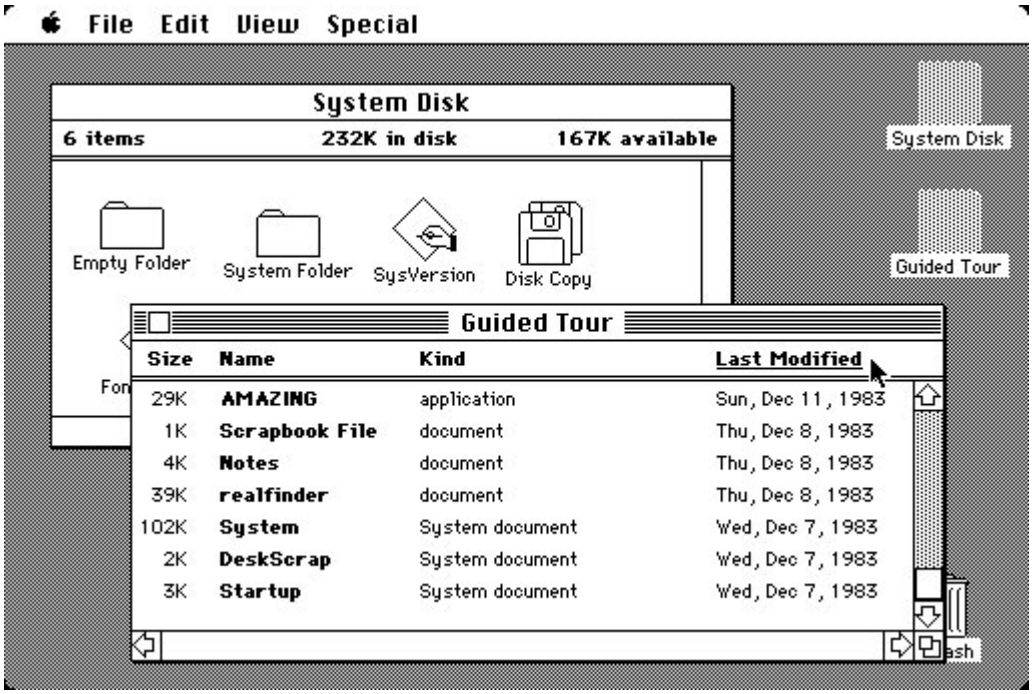
*An instance of a CLI (www.subsowesparc.org)*

The above interaction was very popular at the time of MS DOS (Microsoft Disk Operating System) in the 80's and early 90's, before interfaces with visual elements became popular. Today there are many

pieces of software which either work solely on CLI or have a CLI as well as a graphical interface. Most of them are specialized tools for a task. Users of operating systems like Linux make frequent use of command line interfaces for a variety of tasks as well.

## Graphical User Interfaces (GUI)

*Graphical User Interfaces* are the most popular ones and constitute the primary way of interacting with an application currently.
A GUI makes use of visual elements to provoke behavior from the user, respond to that behavior and provide information. Amongst those elements there are icons, buttons, text fields, various types of list boxes, menus and menu items and more. There are also container elements, which organize content and lay it out in a way that is easy to follow, such as panels, group boxes, tabs and other. Another type of elements is that of output ones, which includes labels, text blocks, message boxes, lists views to name a few.
Graphical interfaces were conceived in the 70's but became popular in the mid-80's with their first appearance on mass market computers (Apple Macintosh). Most modern graphical interfaces can also be described by using the acronym *WIMP*: Windows, Icons, Menus, Pointing device. It should be noted that the invention of the mouse was of great significance to the development of graphical interfaces.



*The GUI as it first appeared on Apple Macintosh (www.history-computer.com)*

The interaction possibilities using WIMP interfaces revolve mainly around using the mouse. Thus actions like pointing (hovering), selecting (clicking), activating (double-clicking), drag and drop, scrolling, right clicking are common in applications and very frequently performed.

## Touch User Interfaces (TUI)

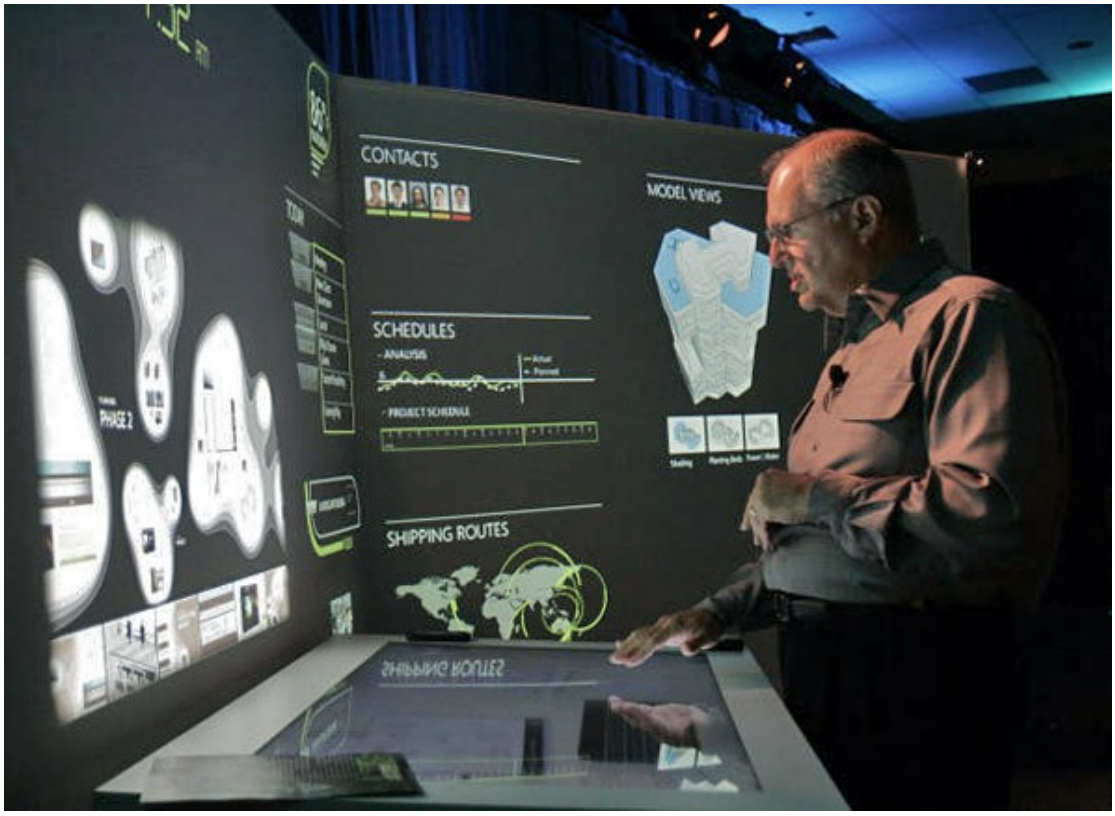
Although Touch Screen technology dates back as far as the 1960s it is only the past few years, since the appearance of touch screen mobile phones, that it has become widely used.
From that point onwards, several different kinds of devices use touch screens: table-like surfaces, tablets, wall displays, information kiosks and more. Since most of these devices have particularities both in terms of what content they hold and in terms of their screen size, it is subsequent that an

investigation into what interface they would mostly benefit from would be carried out.
The results of that investigation, which evolves over time together with the devices themselves, are *Touch User Interfaces*. A TUI has as a main purpose to facilitate the usage of a touch screen by supporting a model of interaction based on the fingers and the hand. In other words, the pointing device moves from being the mouse to being the finger. Moreover, since the size of most of the aforementioned devices is small, it is necessary for a TUI to be taking good advantage of the screen's real estate. That includes designing for smaller screen sizes and carefully picking what content is relevant to show at each point.
Touch interfaces support actions like tapping, double tapping, swiping, tap and hold, multi-touching and depending on the specifications of a particular device, more interactions like tilting and rotating. All of the above are based on direct manipulation of objects. For that reason the term *Post-WIMP* has been invented, which describes interfaces that support behavior beyond the point-and-click. Touch interfaces however still make use of elements and ideas behind graphical interfaces, for instance visual elements, but enhance their capabilities for use with modern devices.

*Tablets are one type of TUI (articles.latimes.com)*

## Natural User Interfaces (NUI)

The *Natural User Interface* is based on design of interactions that feel natural to the human. In that sense, the user and the application communicate in a way that resembles human communication. The aim of such a design model is to make the interface behave in a way that feels *real* and not *iconic* (Curtis, 2011). NUI is a relatively broad term since a touch interface can also be a type of NUI.
Perhaps the most notable feature of a NUI is the support of *gestures*. The interface is able to recognize a movement performed by the user and map it to an action. Another important aspect of natural interfaces is the ability to understand *speech*. Like with gestures, a word, phrase or dictation can be identified and invoke a response from the application.
Apart from the above, there is another important aspect of a natural user interface. *Context-awareness* refers to the ability of the interface to adapt to different environmental conditions or to the subtext of communication received by the user. A simple example of the first case is adjusting the screen brightness depending on the light conditions of a place, while an instance of the second is to turn off the screen when the user turns his head away from it thus removing focus from the application.




*A user is interacting with a NUI using hand movements (Microsoft Research Future Desktop Office, www.hothardware.com)*

# Motion Sensing Technology

There is a variety of technical means that can be used to sense a person's body or partial body movement in space or local muscle movement such as facial expressions. In this section the reader is given a short background of these technologies, for the reader to have a spherical understanding of the world of motion tracking. Getting into too much technical detail on how exactly the artifacts of these technologies operate is beyond the scope of this report, so the reader is advised to refer to the respective literature should he or she be interested in finding out more.

## Mechanical Sensing

Mechanical sensing involves a piece of hardware that the user can interact with and which conceptually acts as the connection between two ends: the performer of the physical move and the recipient of the move, which is the physical or digital environment (Welch and Foxlin, 2002).
Typically, a sensor of that type is designed with one or two rigid mechanical pieces that are connected to other types of sensor mechanisms, such as potentiometers or shaft encoders. As the performer engages the device in some form of motion, the sensor mechanisms move accordingly to support that motion. A typical example of that use case is the *FaroArm* by Faro Technologies. It is a mechanical arm used for physical product measurements that a user can rotate and extend. The arm senses the movement of the person guiding it and moves along with it.
A much simpler case of a mechanical sensor is that of a *switch* or a *key*. For instance, the keys of a keyboard are mechanical pieces that can recognize when the performer presses them and create a representation of that move in the digital environment of the computer, relevant to the context of the application used.

## Inertial Sensing

This technology makes use of *accelerometers* and *gyroscopes* to determine the position of a moving object. It became popular in areas where navigation is of great importance, such as those of marine and aviation, under the name of *Inertial Navigation Systems*.
The standard usage in the areas mentioned above would have a small number of gyroscopes and accelerometers, usually three of each, mounted on a frame. The acceleration vector is calculated by the accelerometer sensors while the frame rotates according to the navigation co-ordinates matrix formed by the gyroscopes. The technique used is called *dead reckoning*.
In their article *Motion Tracking: No Silver Bullet, but a Respectable Arsenal* Welch and Foxlin clarify that such a technology is not a very good fit for tracking human motion, they make a note however on the extensive expected use of sensors like the ones mentioned above which actually happens with mobile phones.

## Acoustic Sensing

The foundation of acoustic systems is the transmission and sensing of sound waves (Welch and Foxlin, 2002). They generally function by calculating the amount of time which a short ultrasonic pulse is traveling for.
This particular technology is mostly used to detect movement in an area and trigger an event (e.g. various alarm systems).

## Magnetic Sensing

Systems of that category make use of measurements from the local magnetic field at the sensor, using either magnetometers or electric current that is induced in an electromagnetic coil when a changing magnetic field passes through it.
Magnetic sensing can be applied to human-tracking. That approach has its advantages, as noted by Welch and Foxlin (2002): the size of the wearable component can be small; magnetic fields do not have line-of-sight problems since they pass through the body; finally, with one single *source* unit multiple sensor units can be tracked. There are, however, drawbacks most notable of which the distortion and strength of the magnetic field.

## Optical Sensing

Optical sensors work with reflected or emitted light. An optical sensor needs a light source, such as an object that reflects ambient light (e.g. surfaces) or devices that can themselves project light that they generate (e.g. LED, light bulbs).
An optical sensor can be analog or digital. An advantage of optical sensors, especially 2D digital ones, is the fact that they can depict an image of the scene with much information. There are also a number of filters that can be used, like infrared, to cut out wavelengths that are not needed. This gives flexibility to the systems that employ optical sensors on how to recognize the human body.
What is considered to be a drawback for these systems is the fact that optical sensors require a clear *line of sight* towards the object that must be recognized. Occlusion of the light source or the object being tracked can cause malfunctions to the system.

## Radio and Microwave Sensing

This technology has not been used for the purposes of identifying human movement. Navigation systems, radars and several kinds of airport landing aids use this technology to a good extent. In the process of making things smaller and cheaper it is possible that radio and microwave could be

employed for motion tracking.

# Current motion tracking solutions

At this point the reader is presented with a number of commercial products from the motion tracking field, mostly in terms of how they perform the sensing of movement. It is easy to notice that many of these artifacts were originally created as controllers for video games on the respective gaming console of the company that released them.

## Nintendo Wii

Nintendo Wii is a gaming console released in 2006. It includes a wireless controller, *Wii Remote*, which can recognize motion and rotation. Supported gestures or movements are swinging, swiping, thrusting and turning the device.
Wii Remote or *Wiimote* as it is known possesses an infrared (IR) camera that communicates with a *sensor bar* that is placed underneath the TV. The sensor bar is in fact a number of IR LEDs that are powered by the Wii. This communication allows the Wii to calculate the position of the Wiimote relative to the television screen, thus making it possible to point at things on the screen (Hejn and Rosenkvist, 2008).
Wiimote has three accelerometers built-in, that way being able to understand movements along three axes (Wisniowski, 2006). Finally, after an upgrade to the device the new controller *Wiimote Plus* can identify exact orientation using a gyroscope.

*A player is swinging a Wiimote while playing a game on the Wii console (www.gamevicio.com)*

## Playstation Move

In 2010 Sony released a motion controller for the Playstation 3 console, named Playstation Move. This controller has a ball made of rubber material on top that lights up when the controller is in use. To track the player holding it, it works together with Playstation Eye, basically a digital camera for the console. The Eye identifies the X, Y and Z coordinates of the light ball in 3D space, which is the camera

field. That way the system knows where a player is. To recognize motion, Move has a three-axis gyroscope which gives information about angle and orientation as well as a three-axis accelerometer for the acceleration of the device when a user is performing some thrusting move with it (Miller, 2010).

*Demonstration of PlayStation Move (www.joystiq.com)*

## Kinect

*Kinect* is a device developed by Microsoft and released initially for the Xbox 360 console in 2010. It is the first type of game controller where the player does not touch anything to give input to the system, but the system itself recognizes the player's pose, gestures and position. In 2012 a new version of the device was released for use with Windows (*Kinect for Windows*).
The Kinect device uses a 3D depth sensor, an RGB camera and a multi-array microphone to receive input from the user. (Hoiem, 2011).
The depth sensor consists of an infrared projector and a monochrome CMOS sensor. In order for the Kinect to understand what exists in the scene and at which distance it needs a *depth image* of the scene. The IR projector emits infrared light on the scene. This light bounces off the surfaces of the objects in the scene and creates a pattern. The brighter the light that bounces off a surface, the closer that surface is to the Kinect. The darker it is, the further away it is from the device (Stark, 2012). Using software that runs in the device, the Kinect can also predict what parts of the light pattern mentioned above belong to a person and therefore track the person's skeleton.
The Kinect tracks 20 joints on a skeleton. That information, the *skeleton data*, is made available to the application which can then compare the position of one or more joints to estimate the pose of the person tracked and whether or not a gesture is being performed. As a result the application can respond to different gestures.


*The Kinect sensor, with the IR emitter (far left), CMOS camera (far right) and normal RGB camera (middle) (www.icranconference.com)*

## Leap Motion

The *Leap* is a small USB device developed by *Leap Motion* and first released for a limited number of developers in 2012.
It is a 12.7mm height x 80mm width peripheral that can track hand and finger movements within a 3D space of 8 cubic feet just above it. The Leap features very high precision up to 1/100$^{th}$ of a millimeter, which makes it the most accurate motion sensing device to date.
At the time of writing this report the market release of the device has not yet been done although many testing units have been sent out to developers across the world. There is a great number of demos and prototypes on the Web that suggest how it will be used, from playing games using one or two hands and the fingers to manipulating the operating system and other applications.
Since it is a new proprietary technology there is not much information on how exactly it works. However it is rather certain that it employs infrared LEDs and CCD cameras, according to unofficial sources and on-line forums.

*3D character control with a Leap connected to a laptop (www.engagdet.com)*

## Omek Interactive

Omek Interactive (www.omekinteractive.com) is a company that works with gesture recognition and motion tracking solutions.
That company does not release its own hardware but rather develops the algorithms for recognizing

22 joints on the hands and provides a framework for developers of different platforms (Unity 3D, Windows Presentation Foundation, Adobe Flash, C#) to work with. A couple of products are *Omek Grasp* and *Omek Beckon*, for close-range hand tracking and long-range body tracking respectively. Moreover the company provides an automated gesture creation tool *GAT* (Gesture Authoring Tool) for development of user defined gestures.


*Omek Virtual Bookshelf interface, with an augmented hand as a cursor (www.omekinteractive.com)*

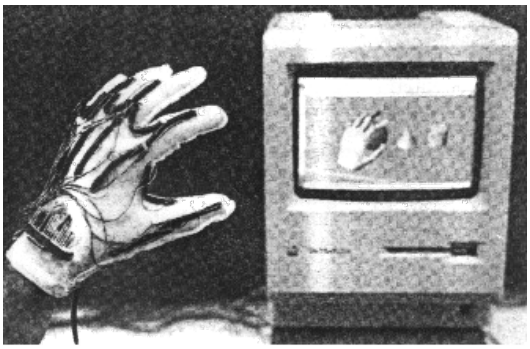
# Related Work

Work on the Touchless Interaction field has been extensive but mainly at a research level. Experimentation with manipulating objects in the digital world dates back to the 70's but the past few years, with the appearance of technologies like the ones presented in the previous section, it has grown. There is a big number of research, student and even commercial projects that employ Natural User Interface and Touchless Interaction practices to accomplish tasks such as navigating and browsing a file system or the web.

The most notable early work on the area is the "*Put that There*" project by Chris Schmandt carried out in the MIT Media Lab in 1979. For this project Schmandt used a combination of pointing and voice commands to create and move around colorful shapes on a big screen. The setup for this experiment took place inside a media room that placed the user seated on a chair opposite a wall-sized display. The user had a cube-shaped sensor connected to smaller cubes placed on the fingertips for pointing to specific places on the screen. The sensor system was based on measurements made of a nutating magnetic field made by one of the cubes that acted as the transmitter. The speech system used the DP-100 Connected Speech Recognition System by NEC which allowed up to five words per sentence without pause. The vocabulary of the speech device had a maximum of 120 words by default. In essence the user would point at the screen where a crosshair appeared. Then he would give voice commands, such as "*Create a yellow circle there*", *"Move that there", "Move that west of that"* and similar. The result would be the appearance of the spoken colored shape at the pointed location. Richard A. Bolt argues that this type of interaction where data is indexed spatially with such commands as the above is natural because the user is present in a real space (Bolt, 1980).

*The predecessor of NUIs, pointing and speaking to interact in "Put that There" (shot from "Put that there demonstration video)*

A slightly different approach on touchless interaction came in 1987 with the appearance of *DataGlove (*Zimmerman, et al., 1987*)*. DataGlove was a glove with sensors attached, that the user would wear on his hand to interact with virtual objects on the computer screen. It used a variety of sensors, like goniometers and ultrasonic transducers to handle position and orientation. The computer had ultrasonic receivers mounted around the monitor and ran specialized software. DataGlove could recognize two types of gestures: object manipulations such as picking up, rotating, throwing and commands like "draw a line", "produce a sound", "set a color". The gestures were also categorized as static (signs) or dynamic (hand motions). An interesting distinction from mouse-based interfaces is that while in those the selection of an object and operation on it is broken into two parts, a DataGlove-oriented interface allowed both at once by gesturing over an object. Zimmerman et al (1987) note that the hand is a natural way to manipulate real objects and according to them the digital world should be shaped to resemble the physical world and not the other way around.



Taken from [ZL87].

*Manipulating objects in the digital world using DataGlove (www.netzspannung.org)*

A more current approach on interacting with an interface without touch is presented by Chattopadhyay and Bolchini (2013). In their work, they propose the use of ultra-high-resolution Wall-

size Displays to create an environment where users are seated and interact with them without touching a device, in a collaborative manner. The scenario for such a setup could be a brainstorming or other similar session where more than one person contribute to the process. The questions raised regard the proper way to give feedback to the user as well as providing the right affordances to avoid confusion when executing a gesture. In the particular case the users had to perform *selecting*, *moving* and *de-selecting* tasks on files and folders. The writers conclude that there is a gap between the user's and the system's mental model because of the lack of haptic feedback and the distance from the screen. That gap can be bridged using proper visual feedback and affordance techniques, such as changing the opacity of the cursor when the user selects an item. These concepts will be discussed later in this report.

Browsing the web in a touchless manner is the purpose behind *DepthJS*, a project carried out in the MIT Media Lab in 2010 by researchers Aaron Zinman, Doug Fritz, Greg Elliott and Roy Shilkrot. DepthJS is an open source javascript library that works with the Kinect sensor to allow users to navigate the Web using their hand. The system supports navigating between tabs, visiting links, scrolling, going back and forth visited pages, zooming in and out and panning. The library, after installed, allows the user to assume control of the selected browser. The user sits at a comfortable distance so as to be seen by the sensor and be able to see the web content. Then he forms gestures in the air to control the browser like panning, gripping and swiping sideways or up and down.

Finally, Microsoft has shown a great interest in the area of natural interfaces. Microsoft Research has dedicated a significant part of its research efforts in bringing Natural User Interfaces closer to the consumer. This has manifested into the *Human Interface Guidelines,* a report and guide on how to create such interfaces using the Kinect sensor and the newest, at the time of this writing, version of the Kinect development kit. The Human Interface Guidelines are in fact the foundation for the prototype application created for the purposes of this research, as will be presented in the Results chapter and discussed in the Discussion.

The above is only a part of the work done in the area. Discussion of other pieces of similar work will take place in the Process section of this report.

# Theory

A Natural User Interface allows the user to perform gestures and use voice commands. These two elements of interaction have their own particularities. At this point a theoretical analysis of what constitutes a gesture and its associated properties takes place. Speech is also examined. Both the above are explored in the context of interacting with a machine without neglecting to refer to the connotations that interaction might be carrying outside the digital world.

## Gestures

There is no single commonly agreed on definition for what a gesture is. Although it is a very old form of communication among humans, there is a difficulty defining it in absolute terms. The reason for it is that gestures are essentially movements of the body but there has to be a distinction between communicative gestures and other movements of the human body (Donovan and Brereton, 2005). To classify movement Donovan, in his work *Movement in gesture interfaces*, presents what researchers have found to be the four basic dichotomies of gesture. These are shown on the image below (adapted from the aforementioned research paper):

| | | |
|---|---|---|
| **Act**<br>Movements that effect a material action in the world | ← → | **Symbol**<br>Movements that serve a communicative purpose |
| **Transparent**<br>Movements whose meaning is self evident | ← → | **Opaque**<br>Movements where meaning is not self-evident |
| **Centrifugal**<br>Intentionally directed towards another | ← → | **Centripetal**<br>Not intentionally directed towards another |
| **Autonomous**<br>Movements that exist independently of other modes of communication | ← → | **Partial**<br>Movements that rely for part of their meaning on another modality |

Dichotomies of gesture (Movement in gesture interfaces)

From the above, Donovan states that *Act-Symbol* is the dichotomy most frequently called to separate a gesture from another type of movement. There is much more debate on the subject mainly based on whether a gesture has a clear communicative intent or whether manipulative movements, i.e. involving objects, can also be categorized as gestures.

In spite of the challenges in defining what a gesture is, four taxonomies of such movements have been established: *iconic*, *metaphoric*, *beat* and *deictic* (McNeil, Gesture: A Psycholinguistic Approach*).*

*Iconic* gestures are the ones where the performer makes a move that imitates what the performer is

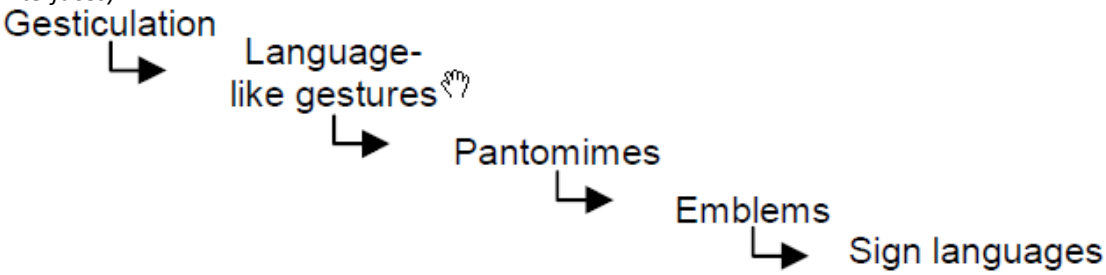
saying at the same time. For example, a person describing a time when he or she was rock climbing at the same time making the rock climbing motion is performing an iconic gesture.

A *metaphoric* gesture is also related to the content of a speech but it does not refer to specific physical actions or items. It is rather a description of a general concept. An instance of that is a person speaking about something abstract, such as the age groups of "old" and "young" and extending the palm of one hand for each group.
*Beats* are gestures that accompany speech but are performed in a rhythmic manner that matches that of the speech. They are usually fast movements of the hand sideways or up and down. In that case the form of the gesture doesn't have significance in terms of content but it is a tool that emphasizes some other part of the communication, like structure (for instance reaching the end of a story).

Finally, *deictic* gestures are the ones where the person is pointing at something, a physical object, location or something either not present physically either abstract, like an idea.

A gesture fulfills some role. This can either be a *semiotic* role, i.e. transfer information, an *ergotic* role that has to do with the ability to manipulate objects or an *epistemic* role; to learn through getting in physical touch with the environment (Gope, 2011).

Gestures are used throughout are communication in everyday life. The most common form of gesturing is *gesticulation*. Gesticulation is according to McNeil, motion that carries a meaning related to the content of accompanying speech. Apart from everyday use of gestures however there are such movements that are used for other specific purposes. Those movements can be arranged in what is known as the *Kendon Continuum*, shown on the following image (adapted from *Movement in gesture interfaces)*.




The Kendon Continuum (Movement in gesture interfaces)

Language-like gestures or "Speech-framed gestures" as McNeil identifies them are not only accompanying but part of the spoken sentence (for example, "*The car was going [gesture of a moving object going from left to right]")*. Pantomimes are independent of speech. Emblems refer to postures that have to be formed in a specific way (e.g. the peace sign). Sign languages are languages built solely on hand movements that allow people with hearing disabilities to communicate.

There is a vast amount of information and research on what gestures are but for the purposes of this report which focuses on gestures in the context of interacting with a natural interface, the following two descriptions are adopted:

*"A gesture is a movement of one's body that conveys meaning to oneself or to a partner in communication.*" (Hummels and Stappers, 1998).

*"Gestures are expressive, meaningful body motions with the intent to convey information or interact with the environment." (*Gope, 2011*).*

So in essence, for the rest of this report a gesture is considered a movement performed by someone with the intent to express a meaning to another or interact with the environment.

## Gesture Recognition

Gesture recognition is the ability of a system to understand and interpret a partial or whole body movement performed by a human. The area of gesture recognition includes different ways on how to achieve that. In the Related Work of the Background section there was a brief description of several systems that have used gesture recognition for controlling and interacting with objects in the digital world.

In general, there are two common approaches to understand gesture in terms of interaction between man and machine: the Data Glove approach and the Vision Based approach (Ibraheem and Kahn, 2012).

The first approach was presented earlier in this report, showing DataGlove as an example.

The Vision Based approach relies on capturing images using a camera. An example of a vision based system would be one that uses the Kinect sensor. There are two subdivisions of in the category of vision based gesture recognition systems: appearance based and 3D model based (Ibraheem and Kahn, 2012).

Appearance based systems work it two stages: the first stage is the training one, where a database of hand signs is formed and mathematical parameters related to them called *eigenvectors* are calculated. The second stage is the testing stage, during which those parameters are compared to the ones extracted from video or camera input (Garg, Aggarwal and Sofat, 2009).

On the other hand, 3D model based systems work using a 3D kinematic hand model with a number of degrees of freedom (DOF). Because of that freedom of movement the model can have various poses. 2D images of those poses are generated and compared to the input image taken from the system in real-time (Ibraheem and Kahn, 2012).

Another distinction between gesture recognition systems relies on whether or not there is a need for a marker element for the system to track a gesture (Kavanagh, 2012).

Marker-based systems require some motion sensing device or other kind of track-enabling element, such as a small surface with a pattern, to be attached to the moving parts. This helps the recognition software track the movement. DataGlove is essentially a marker-based device.
Markerless systems operate by tracking motion into space. There is no need for extra items to be worn or attached to the user. The Leap device presented in the previous chapter is an example of a markerless system.
The advantage of marker-based systems is that they tend to be more accurate, albeit can prove cumbersome. Markerless systems are “walk in and use” and preferred for the purposes of interacting with a NUI, although they can suffer from precision issues (Kavanagh, 2012*)*.

When speaking about natural interfaces, gesture recognition is an indispensable part of the system. Regardless of whether the interaction includes touch or is touchless, the interface must be able to realize how the user is behaving and respond to that behavior. The total number of gestures the user can perform is called the *gesture set* or *gesture vocabulary* for that interface.

# Speech

Speech is the main way humans communicate. This communication can include exchanging information, thoughts, ideas or feelings.

Speech comprises of a series of coordinated complex movements from such parts of the human body as the neck, chest and abdomen (National Institute on Deafness and Other Communication Disorders,

2010). Those movements modify the basic tone of a human's voice into a sound that can be interpreted by others. This interpretation relies on a set of rules that define, among other things, what each sound means. The set of rules is the *language* the speaker is using (American Speech-Language-Hearing Association, What Is Language? What Is Speech?, accessed 12th July 2013).

When examining speech there are three important aspects that differentiate one speaker from another: articulation, voice and fluency.

Articulation refers to the way a human pronounces a word or a sound. In the field of articulatory phonetics it refers to how several speech organs (like the tongue) are configured while speaking.
Voice is the sound generated when air flows from the lungs and the vocal cords come close together. Voice is unique for every human, even though there can be similar sounding voices. From the above definition it is evident that not only humans have a voice, but animals as well.
Fluency is the pace one speaks with. Usually a person is considered fluent when speaking in a fast manner.

The manner of speech and the language are constantly changing due to cultural, environmental and social reasons.

## Speech recognition

Systems that can accept commands given orally are called Speech Recognition systems.
In practice, speech recognition has been employed prior to the appearance of Natural User Interfaces in other environments. The most frequent use is for automated telephone answering. A typical example is a person calling a bus company to book tickets and being presented with a virtual speaker that guides the person by asking questions related to the desired destination and time. The caller has to pronounce the name of the destination, date and time and approve the final booking.
Such a system can be classified depending on three different aspects of its functionality. Therefore there is a categorization in terms of speech utterance, speaker model and type of vocabulary (Vimala and Radha, 2012).

***Speech Utterance***
Utterance refers to the vocalization of a word or a phrase that means something to the computer. Recognizers in that category can belong to one of the following divisions:

Isolated Words
These recognizers require that there is no audio signal before or after speaking a word or a phrase (both sides of the sampling window). Sometimes they have "Listen/Not listen" states (Cook, 2000). In other words, the system requires a pause between spoken words or phrases.

Connected Words
The difference with the previous category is that here separate utterances can be performed together with no pause.

Continuous Speech
Recognizers of this category allow the user to speak in a natural fluent manner. Systems like that are dictation systems.

Spontaneous speech
Spontaneous recognizers allow natural non-rehearsed speech. That means they have tolerance for particularities in individual speech, like mispronunciations and stutters.

***Speaker Model***
Every person is a different speaker because of the unique features of a person's voice. Here there are two categories of recognizers.

Speaker dependent
A system in that category is designed for one specific speaker. It performs accurately for that speaker but not the same for others.

Speaker independent
As the name implies, those recognizers work similarly for virtually anyone. They are not tailored to one individual but can understand patterns of a large group of people.

***Type of Vocabulary***
This classification refers to the breadth of the vocabulary, i.e. the number of words the recognizer can understand. Vocabularies can be

- Small, consisting of tens of words
- Medium, hundreds of words,
- Large, thousands of words,
- Very large, tens of thousands of words

Another category of systems are those that can recognize a spoken word that doesn't exist in the vocabulary. Those are called out-of-vocabulary (OOV) systems. OOV is often used to refer to how often a non-existent word is falsely matched to a word from the vocabulary (OOV rate).

It is easy to understand that the complexity of a speech recognizer depends on its features and capabilities. A speaker-independent continuous speech system with a very large vocabulary offers great flexibility and has a wide variety of possible usage but developing it to be accurate is much harder than an isolated work speaker-dependent one with a few words in its vocabulary, which performs one very specific task.

# Semiotics

Semiotics is the study of signs. Ferdinand de Saussure (1857 – 1913) refers to it as *semiology* and defines it as *a science which studies the role of signs as part of social life* (Chandler, 2013).
Charles Sanders Peirce (1839 – 1914) uses a triadic model to study signs. The three parts comprising it are *representamen, object, interpretant* (Ferreira, Barr and Noble, 2005). Representamen, also known as "sign", "representation" or "ground" (Atkin, 2006), is the signifying element. In other words, it is the actual depiction of the intended meaning. Object is the intended meaning of the sign, what the sign stands for. The interpretant is, in the simplest form, the understanding that one forms of the object, in other words how one interprets the sign.
Signs, according to Peirce, are classified into three fundamental divisions: the icon, the index and the symbol.
When the representamen is a depiction of the object or resembles it in some way, then the representamen is an *icon*. An example of this is the printer icon found in many applications. The sign (printer image) is representing the actual printer.
*Indexes* are signs that exist because there is some sort of cause-effect relationship between the representamen and the object, or the representamen has a direct physical relation to the object. The most common example of that is the photograph: it is a detailed representation of something that looks just like that in reality (as opposed to icons that are depictions, e.g. through painting).
Finally, *symbols* are signs that mean something by convention. There is an agreement that, e.g. the "nuclear hazard" image, means there is danger of nuclear radiation and the person must know that agreement to be able to interpret that sign.

*An icon, an index and a symbol*

The usefulness of Semiotics when talking about Natural User Interfaces lies on the fact that it can be a reference for gesture design. One of the branches of Semiotics is *Semantics*, which examines the relation between the sign and the thing the sign refers to. Moreover, *Pragmatics* examines the relation between the sign and the user of the sign. That can lead to interesting conclusions, especially considering the diversity of background the users of a NUI might be coming from and what the Semantics of a sign are according to their background. Although there is an ongoing study and debate on how much the field of Semiotics can actually offer Interaction Design (Andersen, 2001), (O'Neill, Theory and Data: The Problems of Using Semiotic Theory in HCI Research), minding for multi-cultural user groups and their understanding of signs is a necessary approach when designing interactions and interfaces.

# Natural Interaction concepts

In this section concepts related to natural human behavior when interacting with others or the environment are explored. More specifically, the terms intuition and intuitive are defined while communication channels and multimodality are also explained. Finally, there's a reference to the very important aspect of context of interaction and communication.

## Intuition and intuitive behavior

One of the primary goals in the field of Interaction Design is to create digital products that are easy for a person to use as quickly and comfortably as possible. Very often the term "intuitive" is used to describe interfaces and interactions where the user could immediately act in a manner that felt natural and the interface responded to that action as the user believed and expected it to do. Intuition, by definition, is a feeling based on instinct about whether a situation, decision or person is right or wrong (Driscoll, 2012).

According to Gerd Gigerenzer intuition is "unconscious intelligence" while Malcom Gladwell speaks of the "adaptive unconscious or rapid cognition skills" that are at work when a person makes a quick decision to face a problem or a situation. Caroline Myss refers to it as "an ability present in everyone because it is a survival skill".

It is obvious from the above that intuitive behavior employs cognition to a very small degree if any. In other words, the *cognitive load* when performing an intuitive action is very low because of the absence of an intense thinking process.

In computer interfaces and human-computer interaction there is, according to Raskin (1994), a more accurate term to describe intuitiveness. He uses the word "familiarity" and goes on to explain that an interface that resembles something the user already knows is intuitive. A more technical definition suited for gestural interaction comes from Stern, Wachs and Edan (2006) who state that "*intuitiveness is a cognitive association between a command or intent and its physical gestural expression*".

Intuitiveness is a term that attracts a lot of attention by HCI specialists and designers because of its key importance in Interaction Design. Having taken all the above in consideration this research gives its own definition. For the purposes of the exploration of Natural User Interfaces and Touchless

Interaction, an *intuitive* interaction is one that "can quickly be mapped to an expected result, includes no cognitive load and can be performed without effort by the user". This is further discussed in the Discussion chapter under the usability testing.

## Channels of communication and multimodality

In human communication there are different ways to send a signal that conveys information. Verbal communication appears as the primary way to do so, however it is widely accepted that there are other means that either accompany speech (gesticulation) or even substitute it completely. These means are usually body-language related, like gesturing and body stance but can also be related to speech itself. Those latter ones are called *paralinguistic* elements, examples of which are tonality and pitch of the voice, fluency and loudness among others.
The reason why the above is important is that it allows the study of all the different ways a person can interact through. All these ways, or mediums, are often called *communication channels*. Below there is a list of the main communications channels (Hewett et al., 2011):

- Sound – refers to oral communication, from language to other vocalizations such as sighs, grunts and noises that can carry some meaning like clapping, tapping etc.
- Vision – anything that can be seen by one person, including facial expressions, body language, gestures, twitches and such.
- Physical contact – interaction that includes touch, like handshakes, hugs, strokes among others.
- Smell – signals humans receive from others through *olfaction* usually at a subconscious level, for instance pheromones.

It is important to note that some signals sent through the above channels are deliberate while others are sent unconsciously, i.e. the sender is not aware that they are communicating that specific notion at that time.
Since there are different mediums to be used when sending a signal, it is natural that those will many times be combined for more effective communication. Interactions that use more than one communication channel (also called *modality)* are called *multimodal* interactions. For example, a teacher that is speaking and pointing at a board when delivering a lecture is engaging in multimodal behavior. Similarly, systems that the user can interact with using more than one way to give input are called multimodal systems (Jaimes and Sebe, 2007).

## Context

When humans interact there is usually a topic upon which their interaction is based. There is, in other words, an underlying foundation which the actual interaction is built on using the aforementioned elements as building blocks. That foundation is largely responsible for what will be communicated (content) and in what way (modality).

Apart from that, humans have the tendency to be influenced by anything that can be perceived by their senses and that is not a part of the interaction itself (Valli, 2005). This influence is transferred into the interaction which is also affected.

The above elements, the foundation of the communication as well as the environment that encloses it (literally or metaphorically) constitute the *context* of the interaction. It is very important to know the context in which a behavior takes place because very often it explains things about the behavior itself.

When speaking about interfaces and interactive systems, sometimes there is mention of the term *context awareness*. This refers to the ability of the system to interpret behavior and actions by the user (Valli, 2005) that are related to the task at hand but are not explicitly given as commands. Other times the system can adapt to the surroundings of the interaction (e.g. brightening screen in low-light

conditions) and thus provide an experience that feels more comfortable for the user thus enhancing usability.

# Ergonomics

When using a computer there is a number of rules regarding the suggested setup in terms of physical comfort. Some of these rules are (UCLA Ergonomics, 2012):

- Sit back against the backrest.
- Have knees at hip-level or lower and feet supported.
- The elbows should have a slightly open angle, $100^{o}$ – $110^{o}$. Wrists should be in a straight position.
- Keep the mouse and keyboard within reach.
- Have the monitor at arm's length distance and 2" to 3" above seated eye-level.

The list includes other rules, such as floating hands over the keyboard when typing, limiting repetitive motion, taking breaks and more. The purpose of those is to prevent the user from acquiring some chronic or temporary health problem, most often related to their body pose, muscles, tendons or even eyesight.

These concerns are related to the *ergonomics* of the system. The International Ergonomics Association provides the following definition: "*Ergonomics (or human factors) is the scientific discipline concerned with the understanding of the interactions among humans and other elements of a system, and the profession that applies theoretical principles, data and methods to design in order to optimize human well-being and overall system performance"* (International Ergonomics Association, 2011).

According to Tannen (2011) ergonomics are now more important than ever for a variety of reasons, one of which is the growing usage of touch screens and direct-manipulation interfaces. It is safe to say that the same is true for Natural User Interfaces that rely on gestures and voice: their usage, albeit not extended for the time, also demands some rules or guidelines that will make for a more comfortable and efficient experience.

Tannen also mentions the value of studying *anthropometrics*: measuring the human body size and proportions. This can give the designer a better understanding of how systems can be designed for convenient usage by different users who, naturally, possess anatomical differences (e.g. height, physical condition, age etc). Pheasant and Haslegrave (2005) state that focus should be given on Reach, Clearance, Posture and Strength. *Reach* refers to the extending of the arms while *clearance* focuses on keeping things at a reasonable distance. *Posture* is the deviation from a natural, comfortable position and can be investigated even at limb-level. *Strength*, finally, is about the physical effort the user needs to put into using a system and its resistance (mainly about tangible systems).

Human-centered design commands that human factors are an essential part of designing interactions and that can be proven by the research that has gone into creating usable ergonomically systems.

# Methodology

This chapter presents the steps of the process that was followed when investigating Natural User Interfaces and Touchless Interaction. Those steps are for the most part common in every design process. In this case, particularities of the concept and the technology behind it created slight variations that will be mentioned in the next chapter.

## Overview

The process started with an ideation phase centered on using Motion Sensing technology to see whether it is possible to enhance the user experience when using some desktop application. After that, the effort was on narrowing down the possibilities in order to define a concept. When the idea was conceived a literature study followed in order to relate this idea to research and at the same time receive some inspiration that would make it more specific. The study was succeeded by a benchmarking and related work research. Afterwards came the stage of setting the requirements and specifications of the application to be developed. This was followed by a low-fi prototyping phase. The results of that phase were used to refine the concept. With this new development the prototyping process began. Finally, the hi-fi prototype was tested by users.
A theoretical background of the Ideation, Requirements, Prototyping and Usability Testing phases of the process can be found below.

## Ideation

Dorta, Perez and Lesage (2008) refer to ideation as “the initial idea generation at the onset of conception of a design solution”. They go on to mention that it is a process where the designers express their inner images and conversing with themselves.

Visser (2006) mentions that the expression and conversation need to be constant for the designer to make decisions while he mentions sketching and prototyping as examples of conversation.
Ideation is a part of a three-stage process in every design project, according to Tim Brown (2008). In his interpretation, which looks at ideation within a broader context, this specific part includes “generating, developing and testing ideas that may lead to solutions”.

There is a variety of methods that help generate ideas. Robert Cooper and Scott Edget (2008) present them in a study to determine which are the most popular ones and which the most effective. Some of them are customer visit teams, customer brainstorming (Voice of Customer methods), soliciting the external scientific/technical community, scanning small businesses and business startups (Open innovation methods), peripheral vision, idea capture internally. Regardless of the method chosen, Cooper and Edget conclude that effective ideation is a vital part in the process of launching a product and as such effectiveness in executing ideation methods should be considered a best practice.

## Requirements

A software development process starts with eliciting requirements – determining exactly what the software will do (Saiedian and Dale, 2000). By extracting the requirements users get to know exactly what they want and developers can implement a system that matches them.

This phase usually comes after an idea has been conceived and a foundation to work on has been set. Naturally, the requirements themselves might influence the vision and cause alterations to the initial concept.

Requirements Engineering (RE) can include several activities. A typical process starts with Scoping, which is a general statement of intent and setting of boundaries for the research. Fact Gathering is performed using interviews, questionnaires, observation, text analysis, use cases and scenarios among others in order to acquire a more narrow view on what is needed. After that the system is decomposed in the Analysis phase in terms of its purpose, involved objects and other important specifications. Then comes the Modeling stage, which borrows the results from the analysis and creates notations which make the system's functions and properties clearer. Validation is the activity in which the users have to agree on whether the result of the process thus far reflects their wishes. Afterwards, in the Trade-off Analysis some requirements that are not functional or cannot be met by a specification are found. Finally, the last step of the process is the Negotiation where users, who often might have conflicting requirements, negotiate and discuss (Sutcliffe and Alistair, 2013).

It is easily understood that this part of the development process is the most crucial one in terms of a system's functionality.

## Prototyping

Beaudouin-Lafon and Mackay (2003) define a prototype as "a concrete representation of part or all of an interactive system" while they underline that a prototype is a tangible artifact and not a description that is left open to interpretation. Moggridge (2007) uses the term representation as well and adds that it is made before a final solution exists. Lim and Stolterman (2008) in their definition highlight the importance of prototypes as learning tools by stating "Prototypes are the means by which designers organically and evolutionarily learn, discover, generate and refine designs."

The above and many other definitions of what a prototype is all converge in that it constitutes a first more or less functional representation of a future product. This representation has some significance in the design process and serves a purpose. According to Houde and Hill (1997) prototypes address the *role* of a product, its *look and feel* and its *technical implementation*. The role refers to what part it plays in the user's life and why it is useful. Look and feel is all the characteristics that have to do with how it appears and how it feels like to use it. Finally the implementation includes technical information on how it works.

Floyd (1984) on the other hand focused on software engineering aspects of prototyping. He mentions that its functionality may be implemented or simulated but in any case it should support authentic, non-trivial tasks. He also distinguishes three purposes of a prototype: exploration, experimentation and evolution. The first refers to gathering information about what the software should do and its scope, in essence requirements. The second refers to evaluating how well a proposed solution matches the requirements. Finally the third is about making a system that can continuously adjust to changing requirements.

Overall the purpose of a prototype is to present the idea for a solution early on, emulate its functionality and work as a communications medium among the participants in the development process, users and creators.

There are several types of prototypes. The most notable distinction is that of *off-line prototypes* (also known as *paper prototypes*) and *on-line prototypes* or *software prototypes* (Beaudouin-Lafon and Mackay, 2003).

Off-line prototypes are low-fidelity, which means they are usually not complete in terms of functionality and their look and feel is not indicative of the final result. They are good however in trying out different designs, ideas and discovering weaknesses early on. On-line prototypes are digital and relatively high-fidelity. They are interactive and can include much of the software's functionality.

There is much debate on the choice of prototype to build, however it is generally agreed that paper

prototypes are cheap and quick to make, which makes for a good starting point.

## User test

User-centered design is an approach to user interface design and development that involves users throughout the application design and development process (Venkataramesh and Veerabhadraiah, 2013). One of the final parts of this process is the evaluation of the proposed solution, from which important conclusions can be drawn that might lead to a redesign or partial modifications. Since users are involved in every part of the process, it is natural that they are involved in this as well having in fact a primary role.

When speaking about usability testing there are two big categories: *summative* tests and *formative* tests. Summative tests evaluate the usability of the complete product, usually as a prelaunch-check or an analysis of a product that one would like to improve (Bowles and Box, 2010). Formative testing on the other hand, is conducted on an unfinished system and aims to help *form* the design, i.e. identify weaknesses and improve them.

In Usability Testing the user is given a real task to perform. The design is then assessed, often using metrics. The most important ones are those related to *user task analysis*. Namely, those are Learnability, Intuitiveness, Preciseness, Fault Tolerance and Memorability (Venkataramesh and Veerabhadraiah, 2013).

Learnability is a metric for how easy it is for the user to learn how to perform a task. Intuitiveness evaluates how obvious the task is to accomplish. Preciseness refers to the rate and frequency of errors in the task. The capacity of a design to recover from those errors is called Fault Tolerance. Finally, Memorability is related to how easy it is to repeat a task. The tasks should be able to reveal big problems, in other words it should be actions that are performed frequently, can have serious consequences if done improperly and the development team believes are of critical importance (Redish, 2005).

Regarding the number of users to test, Jakob Nielsen (2000) argues that after five user tests 85% of usability issues will have been discovered, which is enough to start a new iteration and correct those problems. As for who those users should be, representatives of the application's user base and target group is the best approach (Bowles and Box, 2010*).*

# Execution

In this section the reader can be informed about the implementation of the project using the methods mentioned in the previous chapter.

## Ideation

The ideation phase of the project started from being more general and abstract to getting to more specific topics of Human-Computer Interaction and User Experience.

Since a stakeholder in this project was a digital agency with a focus on new technologies and innovation, there were talks about employing motion sensing technology to explore areas of interaction outside the ordinary.

At first a meeting was held where the area of interest was briefly examined. Technologies like the Kinect and the Leap were brought on to the table and their potential discussed. At this point the requirement was to stimulate exploratory thinking in order to come up with a concept that uses those technologies in areas other than gaming. Areas like education and training were mentioned, especially within the context of aiding children.

Next came a brainstorming meeting in which a representative of the agency and the responsible for the project engaged in exchanging ideas for a concept. It was at that point, when the area of Human-Computer Interaction and Motion technology was discussed. There was specific mention of "enhancing the user experience by employing the body as an input mechanism" and how something like that could be achieved. An experimental project, in which a user is capturing a screenshot from a video using only a hand motion and that image is stored on a mobile phone, was the inspiration that started this discussion.

With that in mind, a conversation with the academic supervisor took place that revealed the specific areas of applications that can be explored. With those areas in mind the final idea was conceived. A spectrum containing all the possible applications was mentally constructed. The spectrum categorized all applications depending on how easy a transition would be to touchless interaction from their current use with keyboard and mouse. On one end of the spectrum there are those that are not ideal candidates for the transition (e.g. word processors) whereas on the other gaming applications where placed as perfect candidates. The decision was to work in the middle area with applications that pose as good candidates but might include challenges or perhaps a change on the setting they are used.

After being discussed with all the involved parties, the next phase of the project started.

## Concept Definition

At this point, the result from the Ideation phase was used to put the concept in specific terms.

The browser was chosen as the application to work with. This happened for two reasons: a) it belongs to that middle part of the spectrum mentioned above and b) it is perhaps the most frequently used application for every computer user. That means it is widely used enough to warrant an effort at reviewing its interaction paradigm.

The method for this was on-line research on terms related to gestural interaction and interfaces of commonly used applications. In general, the duration of this stage was short because of the previous work on finding a proper area to work in.

# Literature Study

In order to acquire a better understanding of the field and get a picture of what had happened before a review of related literature took place.

The way to accumulate the necessary bibliography was mainly by on-line research. From the sum of scientific results, articles, papers and books that were returned a number of those was picked. From that number some were rejected for various reasons, from being too technical to not being much related to the subject or being vague. A good number was selected for reviewing.

During the reviewing process, the sources were ranked with significance. The ones closer to bringing a better understanding of the field fast were prioritized. That included articles on Natural User Interfaces and gestures as well as past work on the area of Touchless Interaction. The whole time of reviewing, notes and remarks were made for each source that included its main theme, content and conclusion as well as a reference to whether it is a good addition to the documentation of the project.

The sources that talked about similar projects and work as well as those that deconstructed what a NUI is and which the challenges of gestural interfaces are were of significant importance at this phase.

## Benchmarking

After the literature review, a rough first idea of what the proposed interface for a browsing application should look like existed. However that was not specific enough, nor had it the appropriate depth that comes from seeing a similar piece of work in action.
The benchmarking process included first and foremost locating projects in the area of natural interaction and gestural interfaces. The Kinect community is rather broad and active, providing with a good source of material.

Many projects were examined, from natural interfaces to using a traditional Windows environment using only the hands and from finger tracking to gesture recognition. Most of these projects were student ones, ranging from bachelor thesis work to PhD level. Areas included HCI, computer vision and artificial intelligence.
Professional projects were also put to evaluation, albeit a limited number of them exists. Personal projects or community ones are a vast majority. A number of them were actually tried live to get a better feel of the interaction using gestures. Those are mostly anonymous test projects found at hubs like KinectHacks (www.kinecthacks.com) and related to gesture recognition.
The Kinect Development Toolkit also offers a number of demo applications on Natural Interaction. They proved to be a big help in some areas and helped clear the field.

Since this was a benchmarking stage, there real purpose behind it was to evaluate what is good about the solutions explored and how that could be of assistance to the purpose of designing a new Natural User Interface for browsing the web. In the Results section some more specific details about that will be given.

For the process, a Kinect sensor was used to capture all motion and gestures. Moreover Microsoft Visual Studio 2010 and 2012 were necessary in order to execute the applications, because most of them are written in C# and use Windows Forms or other related frameworks. Together with the Kinect sensor, OpenNI and Microsoft drivers were installed. They both provide the capability of using the Kinect with Windows and running applications.

# Requirements

This process started with analyzing the specifications of a web browser. Simply put, defining what a browser does. Actions from opening web pages to bookmarking were all noted down. Their value in

the browsing experience was put into a hierarchy (see Results chapter) from most important in terms of functionality to least and most frequently used to rarely ever.

For the analysis, Firefox was chosen but then compared to the other popular browsers like Opera, Chrome and Internet Explorer. Their functionality for the purposes of this research did not differ. Their interface and menu commands were examined. The results of this examination were necessary, as they formed the input of the next stage.

## Low-fidelity Prototype

Using the requirement analysis from the previous phase, a paper prototype was made of the target browser application.

The prototype was drawn by hand on a notebook and consisted of a series of sketches of the browser's interface and a suggested gesture set for using it. The most important elements of it were highlighted with color coding for the idea behind it to be communicated better.

That prototype was shown and discussed with the academic and agency supervisors. Feedback was given on the use of gestures as well as the proposed interactions. That feedback provoked more thought on the concept which lead to some adjustments in the initial idea.

## Concept Refinement

At this point, a decision was necessary to be made that would designate the focus of the outcome. The two routes that could be followed were a) focusing on the technology aspect of the application and working on making it work with the current web and b) focusing on the Interaction Design aspect of it and suggesting a new implementation of the web suited for gestural interaction.
This dilemma was discussed with the academic supervisor. The second option was favored for two reasons. Firstly, it was deemed impractical to work on creating software that would run in the real Web since that would require a relatively longer period of learning the development tools and programming. That would leave little time at the end for the actual testing of the conceived model. This would likely lead to compromises that can alternate the intended interactions. The second reason is based on the argument that an application like the one developed for this project requires its own content that would be designed specifically for touchless interaction. That means the web content should be optimized for use with gestures, in terms of screen elements, layout, font-sizes and navigation options.

## Hi-fidelity Prototype

After all the theoretical and conceptual work had been completed, the actual implementation of the final prototype took place.
The tools chosen were an XBox Kinect sensor with Microsoft Drivers, the Kinect SDK v1.7 and Microsoft Visual Studio 2012. The programming language was C# and the framework was Windows Presentation Foundation (WPF) which uses XAML, a markup language for constructing interfaces within WPF. The reason for the aforementioned combination of tools is that they are optimized to work together, at least at the time of this writing. They offer a relatively easy way to integrate with the Kinect sensor. In fact, most of Microsoft's Kinect Developer Toolkit's demos are implemented with a similar approach. The prototype makes use of the Kinect Interactions and Kinect Interface Controls which were introduced with the latest version of the software development kit and which provide built-in support for specific gestures and graphical elements. In the Results section of the report those gestures and elements are presented.

During the prototyping period, the results from the previous phases of the design process were applied. Some of them exactly as when conceived and some altered because of new knowledge that

came to light until reaching this point or some practical limitations related to the implementation. Overall though, the prototype reflects the research done.

## Usability Test

The final part of the process was the usability test. A number of five user tests were held. Participants were all males, aged 25, 26(x2), 27 and 35 years old and all experienced with the usage of computers. Four of them were users of smartphones with touch screens. Two were students of the Interaction Design and Technologies program, two were professionals in software engineering and development and one of the participants was a professional in digital project management.

The environment was setup to simulate a real browsing experience with gestural interaction as much as possible. First, a 24-inch widescreen was used that was raised and tilted upwards for better visual contact. The Kinect sensor was placed under the screen and at a distance from it, on the floor a few pieces of paper acted as placeholders for the user to stand on.

The User Test itself was composed as a text document that was given to the participants to first read through and then execute. The test instructed the users to navigate to webpages, read articles, visit bookmarked pages and perform searches. It can be found in the appendix at the end of this report.

Before the test begun, the designer gave a brief introduction to the work done and the context in which it took place. More specifically, it was explained to the users that this application is part of an effort to investigate whether browsing is something that could be done using gestures and voice in the broader area of using the human body or parts of it as an input and control mechanism.

During the test, the designer stood in the room and watched the interaction at the same time taking notes of things considered significant like the flow of the interaction, points where the user seemed confused or did not perform as expected as well as those that seemed to flow naturally. The user test was not recorded for two reasons.After the test was finished, the participant was interviewed by the designer. Questions based on observation and general were made. In the beginning the user was left to speak about his first impression and provide comments openly with no guidance. Only after the point where no more comments were made spontaneously were questions asked. Those included their thoughts on the design of the application, the combination of gestures and voice and how intuitive it felt, the thoughts on the layout of the pages presented and their opinion on how a system like that could be used. The answers were written down and maintained for drawing conclusions. Those will be presented in the next chapter. Firstly, the observation, notes and interview which followed were believed to be sufficient. Secondly, since the number of tests was small and took place within 3 days it was easy to compare behaviors without the need to review them one by one. The value of recording is however recognized and should be applied in case of more than one iterations of this test.

The room was silent for the extra reason that voice recognition was used that could easily misinterpret any sound.

After the test was finished, the participant was interviewed by the designer. Questions based on observation and general were made. In the beginning the user was left to speak about his first impression and provide comments openly with no guidance. Only after the point where no more comments were made spontaneously were questions asked. Those included their thoughts on the design of the application, the combination of gestures and voice and how intuitive it felt, the thoughts on the layout of the pages presented and their opinion on how a system like that could be used. The answers were written down and maintained for drawing conclusions. Those will be presented in the next chapter.

# Results

This section presents the results from the Benchmarking, Requirements, Low and High fidelity Prototype and Usability Test phases of the project. An analysis of the results takes place at the same time.

## Benchmarking

During this phase of the research a variety of projects was reviewed, some of which tried and tested. Most of the projects were related to gesture recognition, since at the time writing custom or using third-party gesture recognition software was considered as an option for the final prototype. In the end that did not happen, since the Kinect SDK version 1.7 that was used has its own built-in gestures, but that will be presented in the Hi-Fidelity Prototype section of this chapter.
The purpose of this phase was a) to see other current related work on the field, the process followed and the result produced for inspiration and learning purposes and b) to evaluate what is practical and efficient for a Natural User Interface by seeing it in action.

### Operating System Control with Gestures

A few of the systems reviewed were created to control a computer's Operating System, mostly Windows 7, by using gestures. The projects listed below were the ones related to that task:

- KinEmote
- Winect
- Kinect Mouse Cursor
- Windows Cursor Depth Camera Control System
- Kinect SDK Natural User Interface

The control mostly relied on taking control of the mouse pointer with the hand instead of using the mouse device. Then a gesture was mapped to each action that a mouse can perform, like left click, double click, right click, scrolling and zooming in or out. An example of such a system is *Windows Cursor Depth Camera Control System* and can be found online at http://www.youtube.com/watch?v=IgTuLWv-WD4. Here the user can have their hand automatically detected when raised in front of the screen. To simply move the mouse pointer the user's hand must be closed to a fist. For a single click the user raises the pointer finger, while for double clicking both the pointer and the middle finger must be raised. Right click is achieved with raising the thumb and little finger. In order to zoom in pinching and pulling back the hands is required. To go to the previous page when in a browser the user swipes left.

Another similar but more advanced work can be found at http://www.youtube.com/watch?v=sBXf_HcEVlw&list=PL3BF32A1CA61EED5E. This project, entitled *Kinect SDK Natural User Interface*, is proposing a model for a NUI using gestures to manipulate windows. Gestures include full and partial arm movements. The user stands in front of multiple screens placed in a grid formation to form one bigger screen. Then he can manipulate windows, minimize and maximize them and move them around. Raising a hand up is scrolling up, while extending the hand forward grabs a window and the user can then relocate it and release it by pulling the hand back. By bringing one hand in front of the other just in front of the user's abdominal area the windows appear in 3D space one behind the other for the user to switch through them. By raising both arms to shoulder level and then quickly lowering them to the user's sides the windows minimize.

## Javascript Framework

*Zigfu* is a Javascript framework that allows the development of Kinect applications for the Web. That means it provides developers with an API that gives them access to the information the Kinect sensor registers, like skeleton data (joints) and gestures such as swiping or pushing.

The reason Zigfu is mentioned here is that it was tried as a prototyping solution. It proved to be hard to manipulate a cursor using that framework, because it was not as responsive as it should be for interacting with an interface. Specifically it was difficult to achieve a balance between the speed of a hand movement and the responsiveness of the cursor. Despite the calibration issue it is a promising new technology.

## Benchmarking Evaluation

The aforementioned work together with other similar reviewed projects from the Kinect on-line community succeeded in their dual aim: on one hand they provided a good idea of what work is being done with Motion Sensing Technology and on the other they raised some important questions and issues regarding Natural Interaction.

The most important issues have to do with the naturalness and difficulty of performing a gesture. By watching the demonstrations for the two projects referenced above, as well as others, and comparing them to the reviewed literature and guidelines it becomes obvious that interactions were designed with a focus on technological capacity than user comfort. For example, both those projects although impressive features in terms of programming, do not give out a feeling of comfort and intuitiveness. Instead they seem rather arbitrary. Controlling the mouse pointer with the hand is not ideal, since the hand is not so accurate when pointing somewhere and, by trial, even the slightest finger movement causes at least a little hand shake which can ruin an interaction. The “Minority Report” type of interaction on the other hand (made famous by the 2002 film which featured Gesture Interfaces and Touchless Interaction) totally neglects the ergonomics of human-computer interaction. Some of the movements performed, such as the sudden lowering of both hands, can even cause muscle or tendon damage especially for groups of people with a history of arm or shoulder injuries.

Overall, the Benchmarking phase revealed the major problems that exist today with Natural User Interfaces. The subjectivity when it comes to the word “natural”, since what feels easy and comfortable for one person might not for another. The difficulty in using gestures for interfaces that were not designed for that kind of use and as noted above, the lack of ergonomic support for those proposed interactions.

# Requirements

At this point the specifications of the target application are defined. Since this application is a browser and an analysis of what current browsers can do has been carried out, a list with all the necessary attributes is assembled. This list constitutes the *Functional Requirements* of the application.

Apart from the functionality specifications an application has other important responsibilities that create more demands in terms of requirements. Specifically for the case of a gesture controlled browser, those are the Gesture and Speech Vocabularies, concerns about the UI design that supports Touchless Interaction and feedback mechanisms to give the sense of comfort and control to the user.

To accommodate the above two types of lists were created. One with all the functional requirements, separated into three parts depending on the importance of the task they realize. The second, regarding all the other technical requirements that were mentioned above.

## Functional Requirements

Tables 1, 2 and 3 describe the functional requirements for this browser.

| MAIN FUNCTIONS | |
|---|---|
| Name | Description |
| ENTER URL | Enter a web address (www.example.com) |
| GO TO URL | Open the page designated by the entered URL |
| NAVIGATION BACK | Go back in browsing history, to the previously opened web page |
| NAVIGATION FORWARD | Go forward in browsing history, to the page visited before going back |
| CLICKING ON BUTTONS, LINKS ETC. | Interact with page elements |
| STOP LOADING PAGE | Interrupt a request to open a page |
| REFRESH PAGE | Reload an opened page |

**Table 1: Functionality that has to be supported by the prototype**

Table 1 includes all the primary actions a user can perform with a browser. Entering a web address in the address bar, visiting that address and navigating back and forth in the browsing history are all essential and frequently used functions. Interacting with control elements like pressing buttons and links are also in this category. Finally there are the actions of stopping a page from loading and refreshing a page, which might not be used that often in modern browsers because of fast internet speeds but are considered basic.

| SECONDARY FUNCTIONS | |
|---|---|
| Name | Description |
| SELECT/HIGHLIGHT TEXT | Same as click and drag the mouse pointer over a word or piece of text |
| COPY/PASTE | Copy the selected element(s) to the clipboard and paste somewhere else |
| SEARCH | Look for a word or phrase in the document or perform a web search for it |
| ZOOM-IN, ZOOM-OUT | Magnify/Demagnify content |
| VIEW CONTEXT MENU | Bring up pop-up menu with extra actions on the document (same as right click) |
| BOOKMARK | Add a page to the list of bookmarks or view the list |

**Table 2: Functionality that is not primary but widely used in real browsing**

The secondary functions refer to actions that are performed frequently by users but are not core browser functionality. Selecting text and performing copy and paste commands are useful actions that fit in that category. Searching could refer to either searching the Web or searching for a word in the current document. It is important to note that there must be a distinction between the two just like there is one in traditional ways of browsing. More on this will be discussed in the Discussion section. Viewing the context menu is usually used when trying to access settings of the application or to perform some action quickly without visiting the menus (shortcut). For that reason it can be considered a secondary function. Bookmarking and viewing the bookmarks is an important part of the browsing experience since very often users prefer to store links to websites they visit often and open them from the bookmarks menu.

| NOVEL EXPERIENCE FUNCTIONS | |
|---|---|
| Name | Description |
| 3D VIEW | Use depth to stack pages and shuffle through them, take advantage of 3D space |
| MOVEMENT RESPONSIVENESS | Change angle when the user moves so the view remains same |
| IMPLICIT COMMANDS | Settings change according to contextual input, e.g. increase font size when user is far away, brightness if it's too dark or light, volume if in noisy or very quiet environment. The system must detect that the user has trouble with the current ones. |

**Table 3: Innovative functionality that is not currently used in browsing**

This last table presents some original concepts that can enhance the user experience in terms of interacting naturally with an application. These functions rely heavily on usage of sensing technology and aim to make the browser, in this case, a context-aware system that responds to input that is not direct. *3D View* refers to using a third dimension to layout content. Specifically it could be used to organize the open pages in layers and switch through them using an appropriate gesture. The 3D space is an area under investigation as to how it could be employed for applications not related to simulations or games. More on this will follow in the Discussion. *Movement Responsiveness* is the ability of the application to understand the relative position of the user and adjust the angle so that the content remains visible no matter where the user is located, within a fixed range. An ability like that would work closely with 3D View because "turning" the screen content towards the user would require the depth dimension for proper display. *Implicit Commands* is a set of settings that can be adjusted automatically by the system as a response to environmental conditions or indirect user actions. For example in low-lit conditions the screen can become brighter or when the user is leaning too much to read the content the system can increase the font size to make reading easier.

# Technical Requirements

The second type of requirements is related to making the Interaction with the application more smooth, comfortable and effective. That has to do with making input an intuitive process by using proper gestures and speech. Since there is lack of tactile feedback the need to give the user the impression of control with other means is evident. Thus, the requirements for Gesture Vocabulary, Speech Vocabulary, User Interface Design and Feedback.

### *Gesture Vocabulary*

Based on the research done through reviewing the respective literature, a list of qualities for the gesture vocabulary was composed. The vocabulary should be:

- Short
- Task-oriented, not "one style fits all"
- Ergonomic, should not include complex poses or gestures that place too much stress on the hands or other part of the body
- Minding the meaning of signs, paying attention to the semiotic meaning of a gesture and the pragmatic (practical) and hedonic (enjoyment of performing it) value of it.

A short gesture vocabulary makes it easier to remember all the primary commands and does not confuse the user. Associating gestures with specific unique tasks helps mapping input to an action and

avoiding multiple performances of a movement until the right one is found. Performing a hand movement should be easy for people with normal motor skills and no physical problems related to them. Complex hand postures should be avoided both because of difficulty to execute and for avoiding risk of causing damage to muscles or tendons, since it is obvious that a gesture will be repeated at least a few times throughout the whole interaction. Finally it is important to create a feeling of familiarity when performing a gesture, which the user will enjoy. At the same time the gesture should be generic enough not to be associated with a special meaning of some culture. That last requirement depends also on the target group of the application and whether localization is under consideration.

### *Speech Vocabulary*

The Speech Vocabulary refers to the chosen words that will act as voice commands. The requirements for this input mechanism bear many resemblances to that of gestural input. Specifically, the vocabulary should be:

- Short
- Clear
- Easy to pronounce
- Substitute for actions that cannot be easily performed with gestures

The vocabulary should preferably consist of a small number of words for the same reasons as the gestures. That is something slightly ambiguous since an application might include a big number of functions that need to be implemented. For the purposes of a browser application a number of roughly ten words is a good balance. That number is enough to cover all the frequently used functions a user performs with a browser.

Note that a) the vocabulary only refers to the words that act as commands and b) those commands are browser related. An on-line application that runs in the browser might need more or less commands, but in any case it is a separate application that has its own speech set. In general, the total number of words that can be recognized might be hundreds or even thousands, but the vast number of words in that case would be for dictating purposes or similar.

The language should be imperative so it is clear that the user can get in the habit of using the appropriate tone for a command. The words chosen should be easy to pronounce to avoid multiple repetitions until the speech recognition system recognizes the command.

Finally, perhaps the most important requirement is that words should be preferred where gestures are difficult to apply. That removes the need for broadening the gesture set and creating complex gestures. Expanding the speech vocabulary is also preferable than expanding the gesture one since gestures usually require some graphical element to interact with which can clutter the user interface.

The option of supporting both gesture and voice for an action came up during the usability test interviews. Although there are ideas on how it can be achieved, in this first iteration it has not been explored further.

### *User Interface*

The design of a UI for an application that works with gesture and voice varies in certain aspects from the traditional GUI. The main reason for that is the fact that manipulating objects on screen with a hand requires that they have relatively large size. The cursor should be larger than the mouse pointer to create a good metaphor for the hand and provide a sense of good control over the input mechanism. That means the interface benefits from minimalistic designs with few and big control elements (buttons etc). In general the UI must have the following characteristics:

- Clear and spacious
- Focusing on indexicality and icons

- Follow Fitt's law
- Understand intent, accidents and be lenient

The first characteristic has been described. The focus on indexes and icons means the avoidance of symbols since those create a bigger cognitive load (the user must learn what their meaning is). Indexes and icons are more self-explanatory. Fitt's law is used to mathematically calculate the time needed to point at something. Its application in terms of interface design states that moving a pointer a short distance to a large target is faster than moving it a large distance to a smaller target (Pavlus, 2013). That means elements should be relatively close together and occupy real estate enough to be easy to point at. Lastly, Natural Interaction relies a lot on the capability of an interface to realize when the user is interacting with it on purpose and when the behavior is unintentional. In short, the interface must be able to differentiate partial gestures from full ones and speech that is not directed at it. In case of false-positives, behavior mistakenly perceived as input, the interface should be flexible to return to the prior state without much effort and in little time.

***Feedback***

This part of the interface is one of the most crucial ones. A feedback mechanism that provides information to the user regarding impact his behavior has on the system can fill the gap that the lack of haptic feedback is creating. Regarding feedback, the following are important:

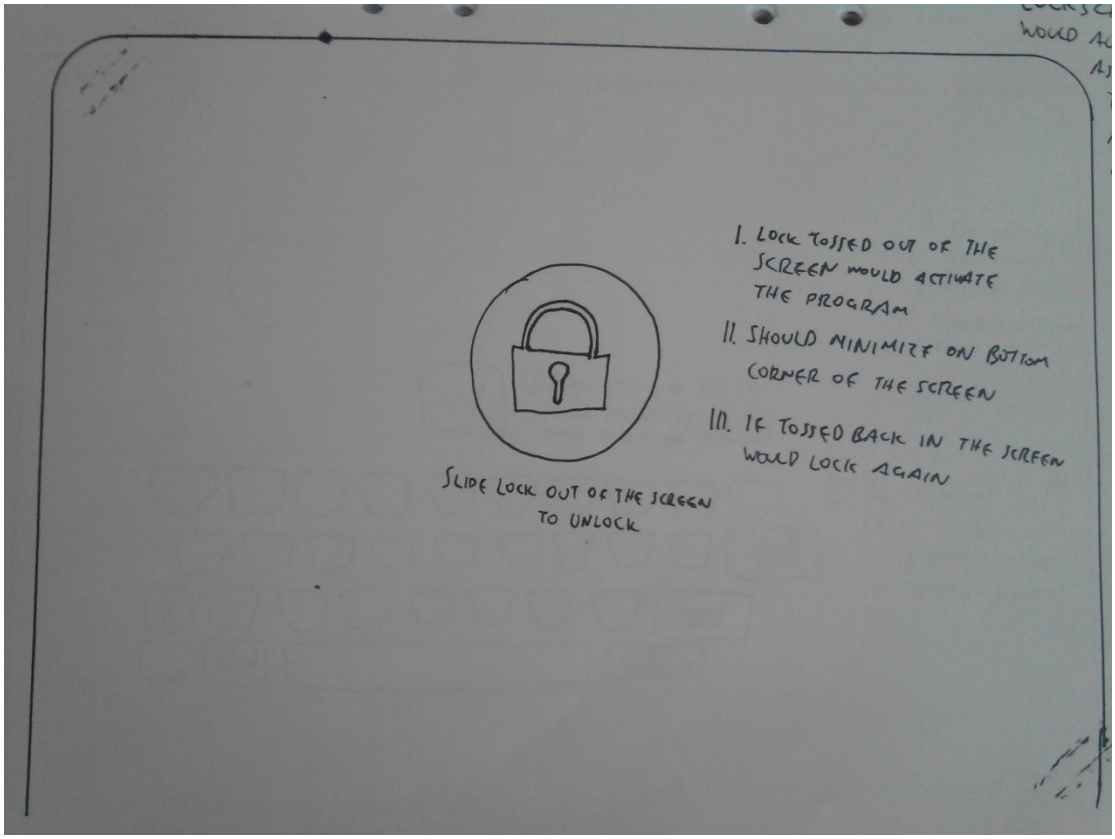
- Combine visual with acoustic
- Proper pointing cue
- Let the user know when he is not within the active interaction zone
- Real-time responsiveness
- Smoothen the flow of interaction

By sending out both visual and audio signals while the interaction is running the user receives input for the two primary communication channels. That forces his attention to be focused on the response the interface is giving to his commands. The combination is good to have especially with those interactions that would otherwise rely on touch, e.g. pressing a button. The pointer or cursor used should be designed to be easily mapped to the hand of the user, thus making the connection between digital and physical tighter. Moreover, the interface should remind the user that all interaction happens within a specific area and alert him when he leaves it. It is important that everything happens synchronously, i.e. immediately after the user has performed an action or at the same time. Last but not least, the flow of interaction should be aided by the feedback mechanism. Giving the user tips when in perceived uncertainty is, for example, a way to achieve that. It is also another implicit input mode, a fact that highlights the importance of that mode to Natural Interaction.

# Low-fidelity Prototype

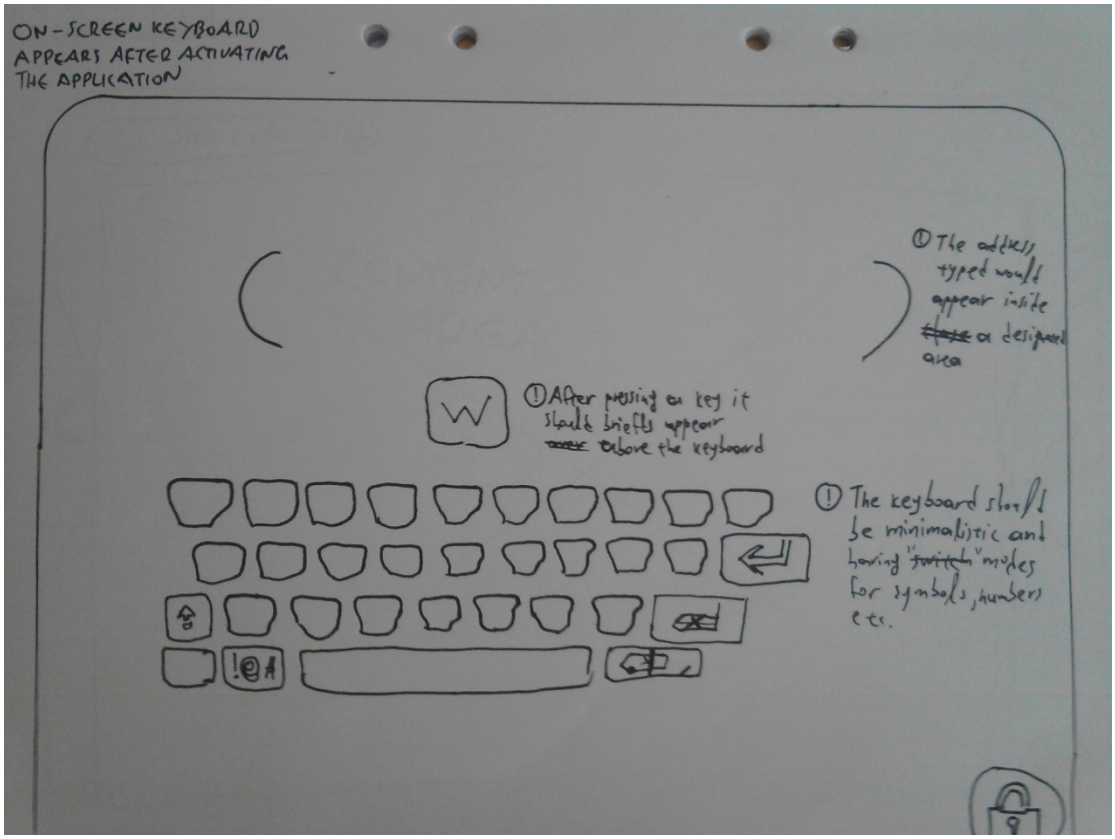
The result of this phase was a paper prototype. More specifically, it was sketches of the conceived application together with annotations that explain the functionality, placement and behavior of the control elements used. Following are some images from this prototyping phase.

This image shows the application home screen in the paper prototype. A lock signifies that the program must be unlocked to be interacted with, very much like the unlocking of the screens in smartphones where the user must slide the lock to one side. In this case the lock is supposed to be grabbed and tossed to the right side of the application. That would unlock the application which could then be used.



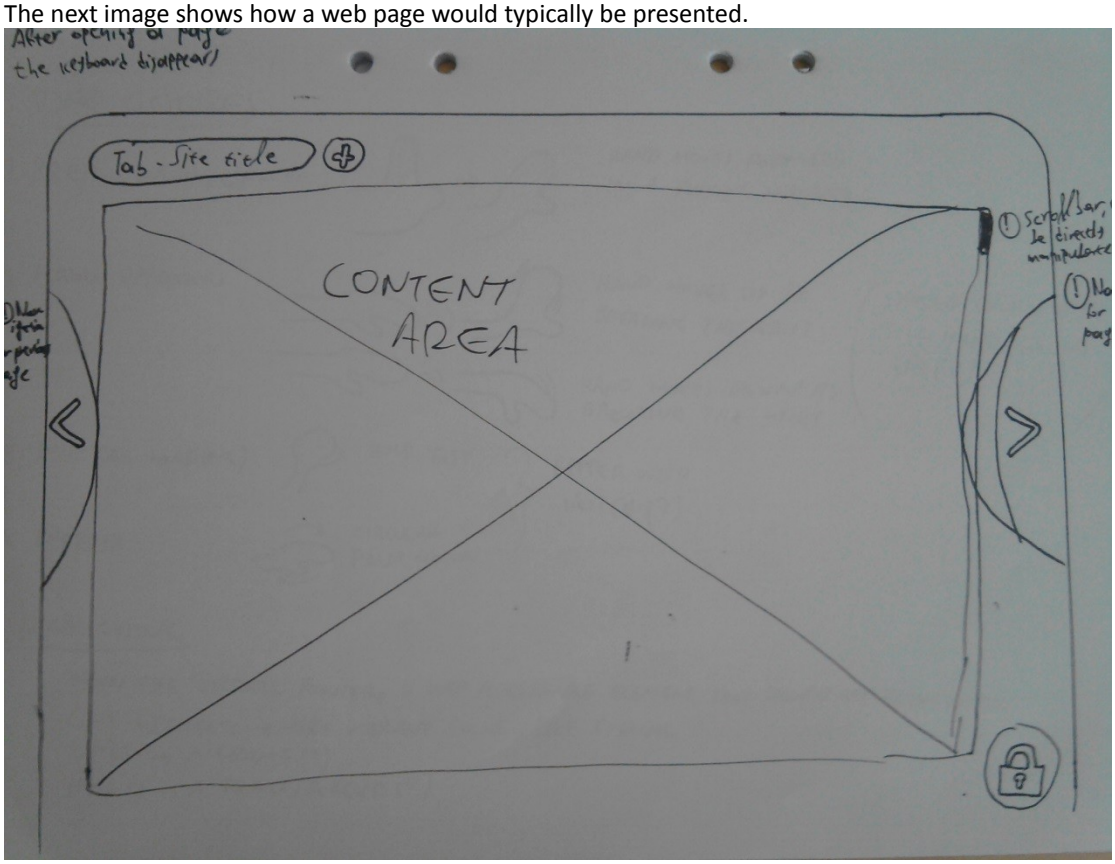

**Paper prototype: Application first screen**

Once unlocked, the application presents an on-screen keyboard for the user to type in the address of a webpage. A big text field in the middle of the screen is the place where the text will appear. After pressing a key the respective letter would appear above the keyboard as feedback that it has been pressed. This keyboard works much like the on-screen keyboards found on smartphones. The following image shows that scenario.




**Paper prototype: Typing in a web address**

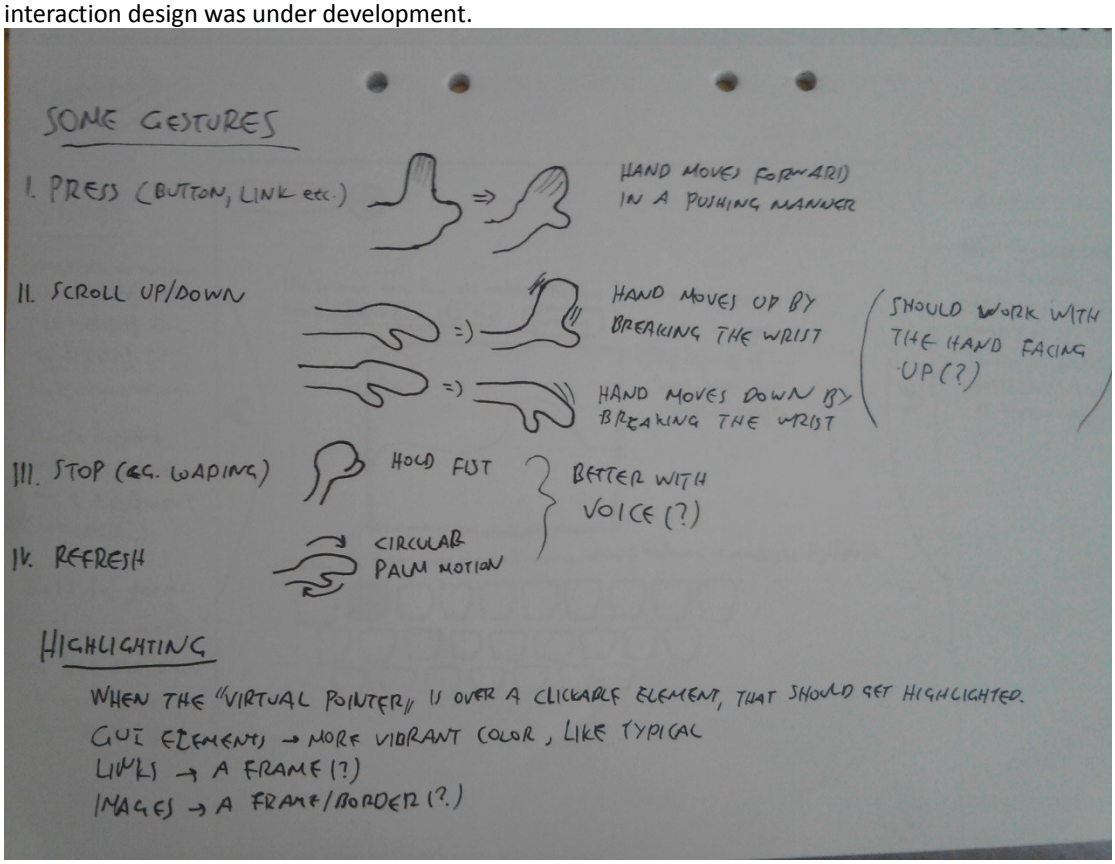
The next image shows how a web page would typically be presented.




**Paper prototype: Webpage open in the browser**

There is a tab on top which shows which page is open currently. The plus sign would take the user to another screen like the first one. In the middle the area is occupied by the content, i.e. the page. On the left and right sides there are arrows that can be pressed to go back or forth in the navigation history. The lock icon can be grabbed and tossed back in the screen to lock the application again. Locking the application would mean that the user cannot have access to the content and any gestures would not be interpreted unless the application is first unlocked.

At the same time with the application design the first design for the Gesture Vocabulary took place. Below the basic interactions with the applications are shown. These are “Push” to press, “Swipe” up or down to scroll, “Hold fist” to stop an action (like loading a page) and “Rotate” to refresh the page or repeat an action. In this image some design questions are also posed, like whether substituting the stop/reload gestures with voice. This phase was an early prototyping phase which means the interaction design was under development.



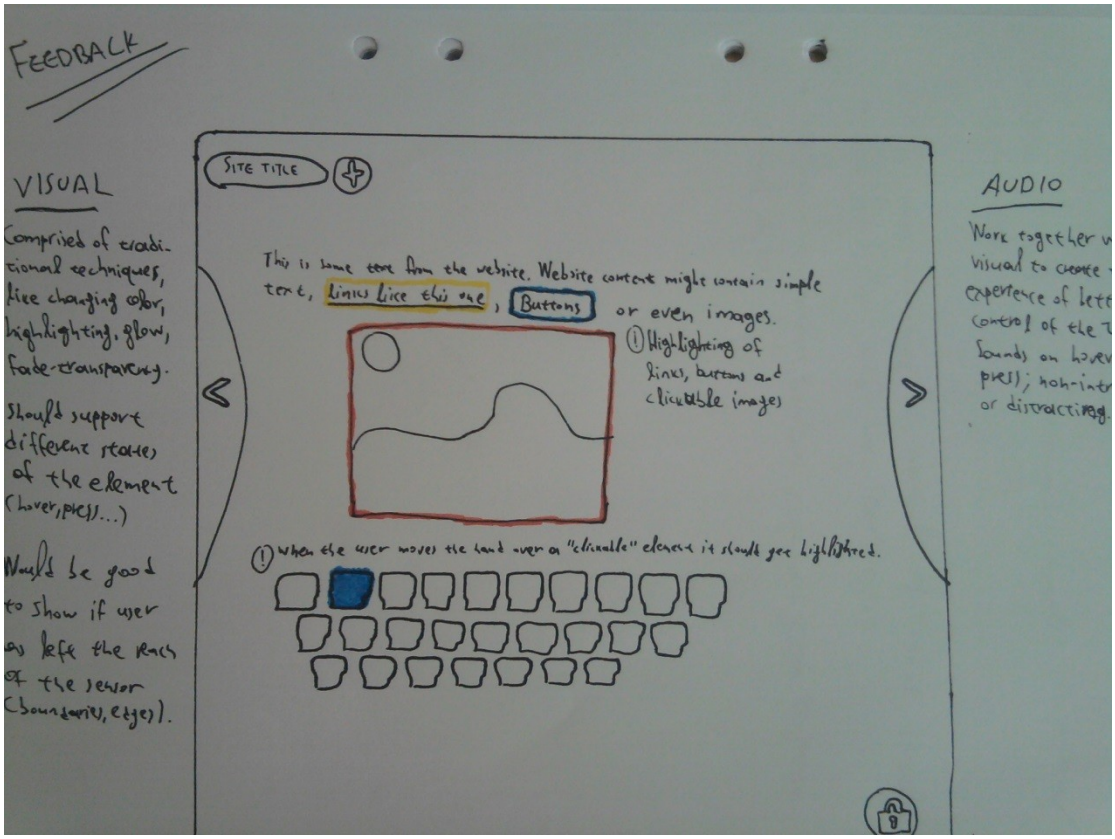

**Paper prototype: Early gesture set for the application**

## Feedback

Regarding the feedback mechanism, the sketch that follows presents the visual feedback provided to the user for any action. There are also annotations for the audio feedback.

Links should be highlighted when the user hovers over them using appropriate coloring such as red, yellow and blue. Those colors are often used in graphical interfaces and web pages too. Buttons and images that are clickable follow the same pattern.

Audio feedback should be provided in the case of the above actions too. Those would be in the form of short unobtrusive sounds.




**Paper prototype: Feedback**

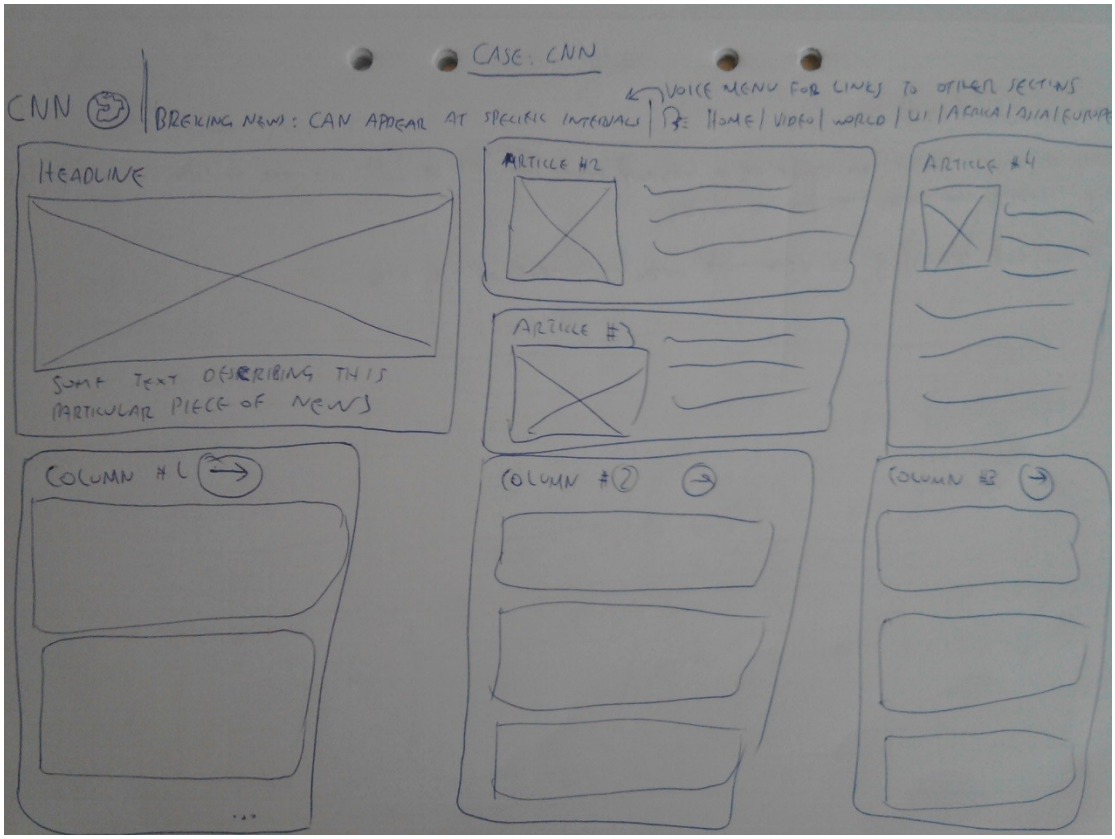
## Web pages

Except for prototyping the application a few prototypes were created for the content of the application as well. The content of a browser is the webpages that it loads and displays. This thesis, as stated earlier, is suggesting that web content ought to be redesigned for the browsing experience to be experienced fully. For that reason a couple of use cases were examined. One is a content-heavy website and the other a Google search results page.

The website chosen to represent pages with a significant volume of content was the news website CNN. A choice like that enables the evaluation of the concepts of Natural User Interaction and Touchless Interaction in a setting where efficiency, practicality and a serious tone must be preserved.

The design for this page is based on the current version of the website. It is adapted to what is considered, based on the findings of this research, good design for gestural interaction. This adaptation was inspired by the CNN mobile application, not in terms of design so much but in terms of the logic behind it. The most important or trending pieces of news are shown while the many links to different side events of one big event are narrowed down to a selected few. The color scheme is preserved while the layout is following a similar approach as the “metro” layout. That is very convenient when working with the new Kinect SDK and provides enough space and size for interacting through a bigger cursor, like the ones preferred for touchless interaction.

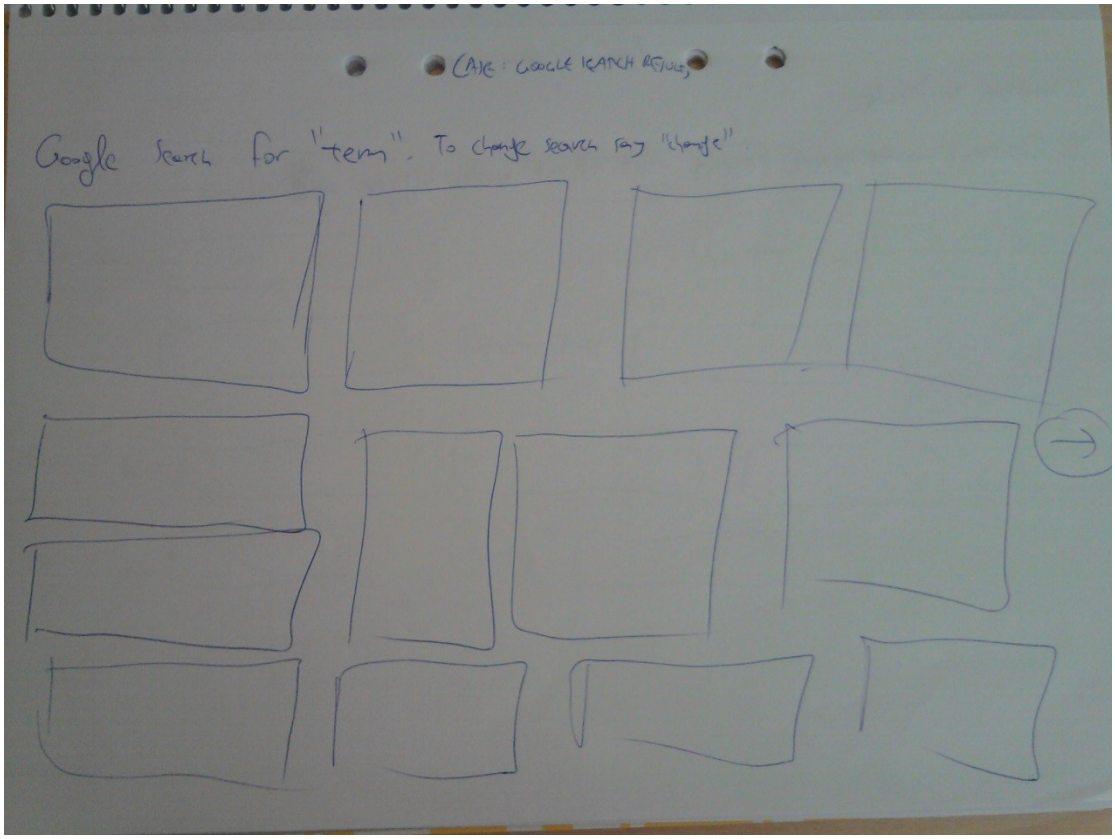
At the top there is a navigation menu which works with voice. The user speaks the content of the label and that part of the website is visited. Inside the page there are big square areas that are clickable. Each of them is a piece of news and opens the respective page with the article. Those areas are laid out in a way that corresponds to their importance in the news hierarchy and their relevance to a subject. Size and spatial grouping address those two issues.




**Paper prototype: Sketch of CNN front page**

Finally, the other use case was a Google search results' page. The reason for this choice is that searching is something that happens regularly when browsing. Hence it has specific important to see how searches could be performed using a form of input other than the traditional. Again, efficiency and practicality are of the essence.

The design shown on the next image is a grid with all the results of a search displayed as big square buttons that can be pressed. Once more, the size of the square is related to the relevance of the result to the search term. The arrow on the left means the user can press it and scroll to see more results.




**Paper prototype: Google search results page sketch**

## Usage

The sketches were used in discussions between the supervisors and the designer. They helped identify some weaknesses which were changed in the final version. The most notable change was the removal of the on-screen keyboard and its substitution with Voice Recognition. The next section describes the final prototype.

# Hi-fidelity Prototype

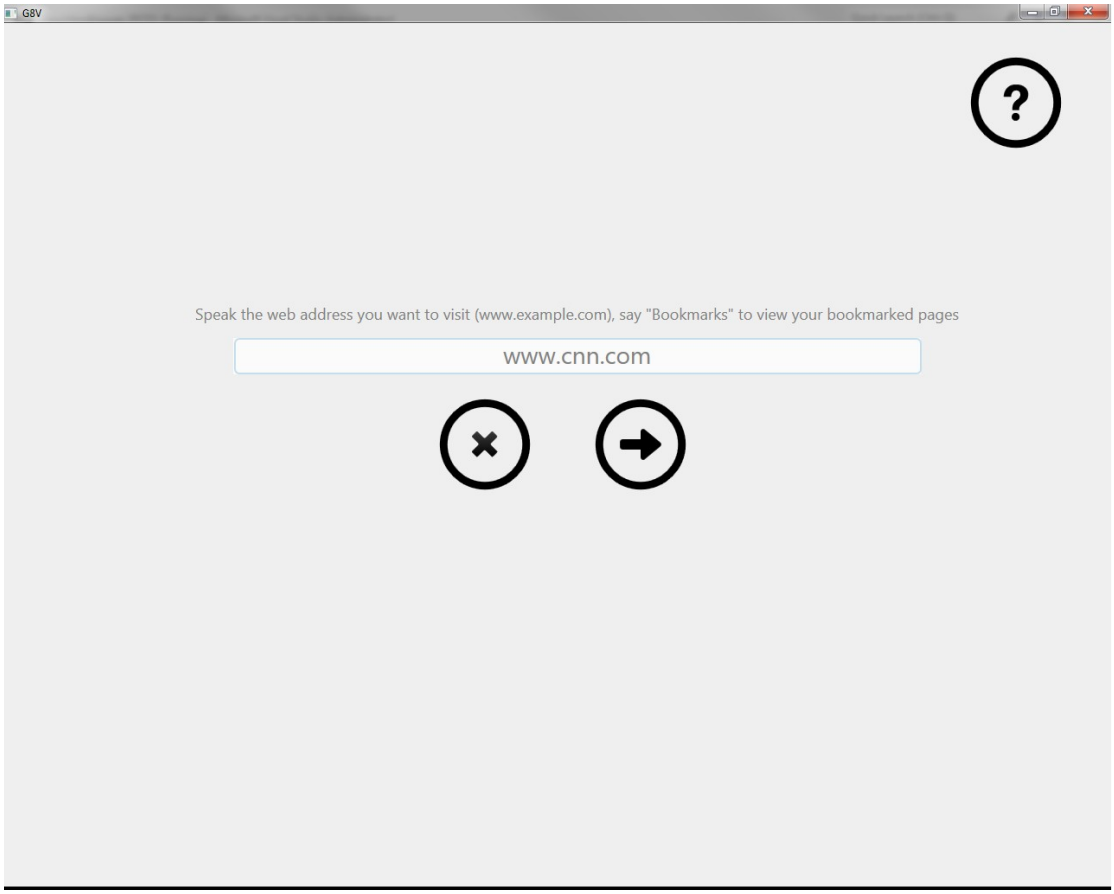
The practical aim of this thesis was to create a prototype that would showcase the ideas conceived and the research carried out. This prototype would also serve as a testing tool from which conclusions for further improvement would be drawn.

## Final Design

The final prototype has some notable differences from the paper prototype. This happened for two reasons. Firstly, after the paper prototype was created a demo on-screen keyboard was implemented prior to the final prototype, to test how typing on air feels like. The response and general experience did not encourage the implementation of a keyboard for the final version. It seemed hard to type fast and the rate of errors was rather high. At that time, on-line speech dictation tools were discovered which gave the idea of using solely voice for URL input.
After a number of tests with Speech Recognition it was obvious that this was a more promising way to go.

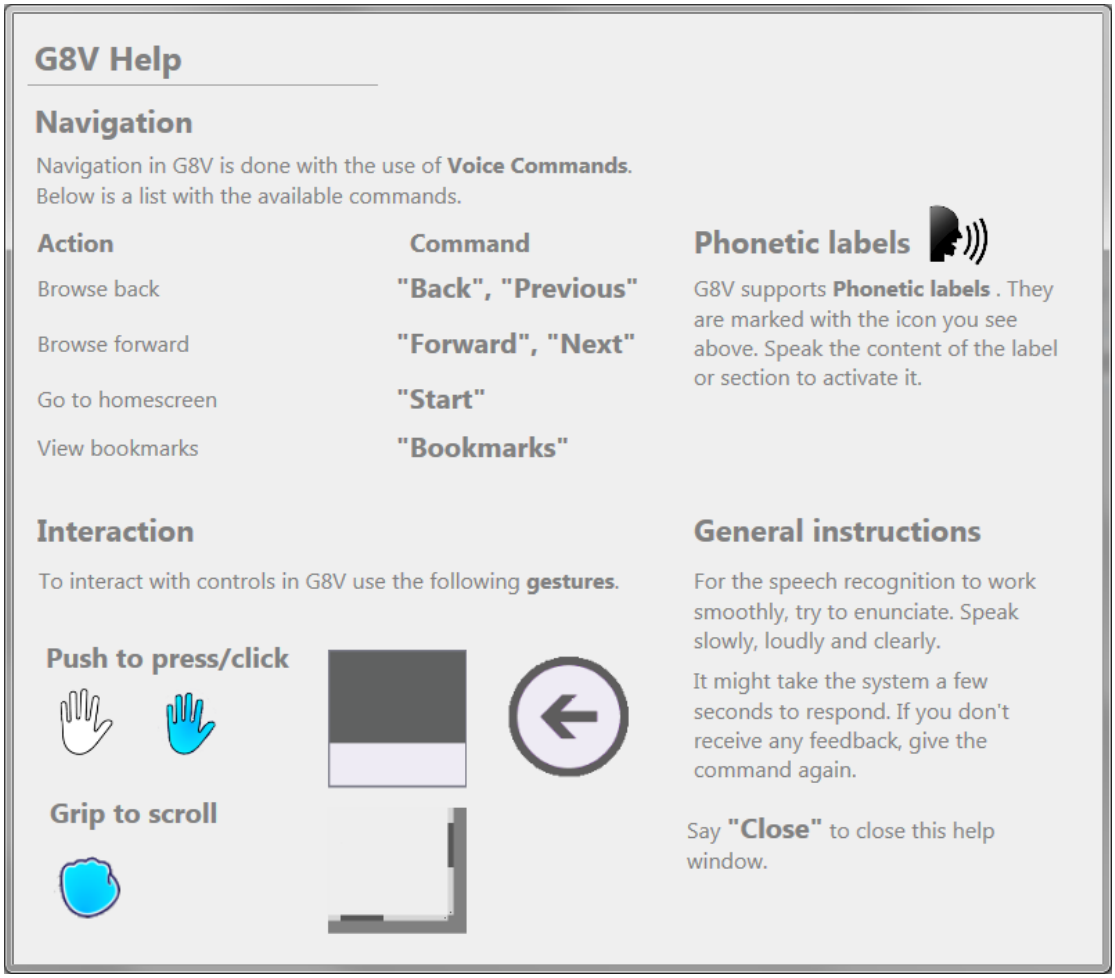
Secondly, some features of the paper prototype were not implemented for practical reasons. For example the lock/unlock feature of the application although useful was not deemed as a high priority feature and was left as part of future improvement. Other than that, most of the other features were

implemented perhaps with slight modifications. The final version is largely influenced by the research done by Microsoft Research on Natural Interaction and presented in *Kinect v1.7 Human Interface Guidelines.* All these design decisions will be discussed in the Discussion chapter.
The start screen of the application consists of a “Help” button, a “Clear” button for the text field and a “Go” button for visiting the address or performing a search.



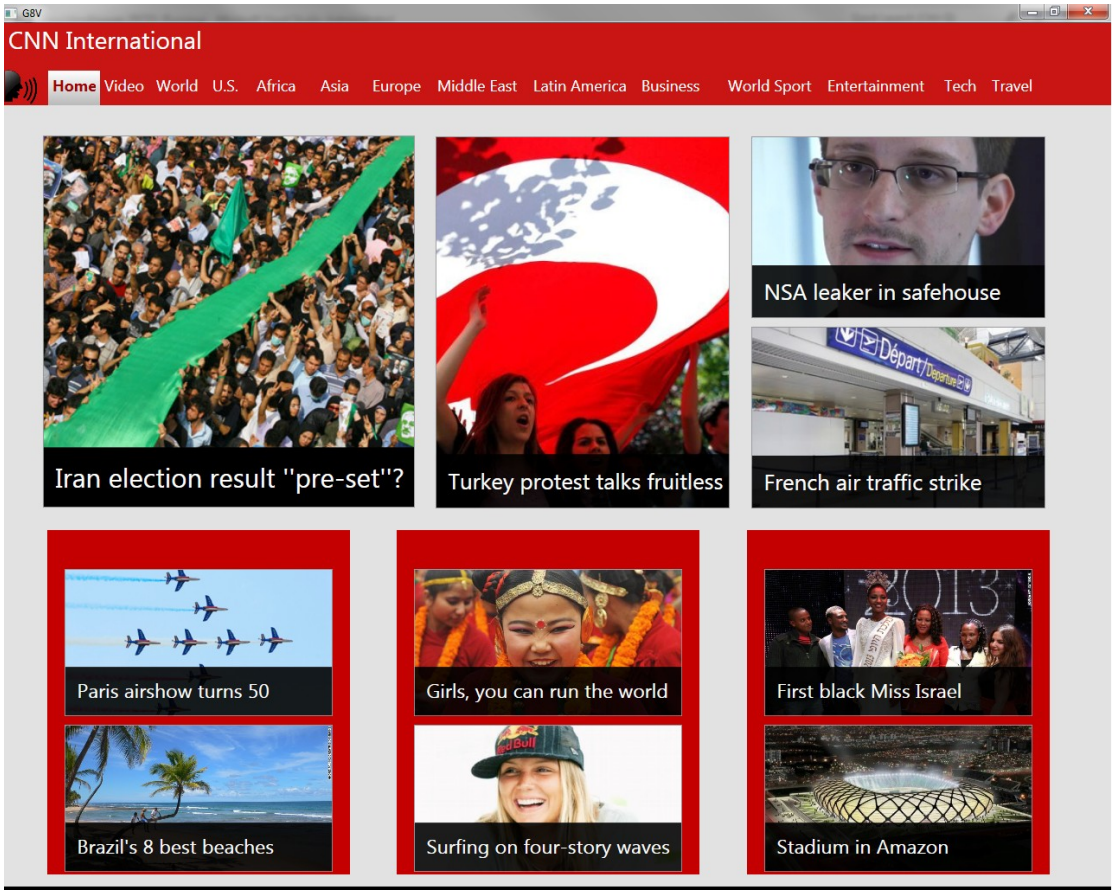

**Final prototype: Application start screen, with a URL recognized after being spoken**

If the user clicks on the help button, the following dialog appears. Here the user receives information on how to use the browser, including the possible interactions with gestures and voice. The “Phonetic Labels” are pieces of text in a webpage that can be given as commands and are used for navigation. Usually they refer to a section of the website, as will be seen next.



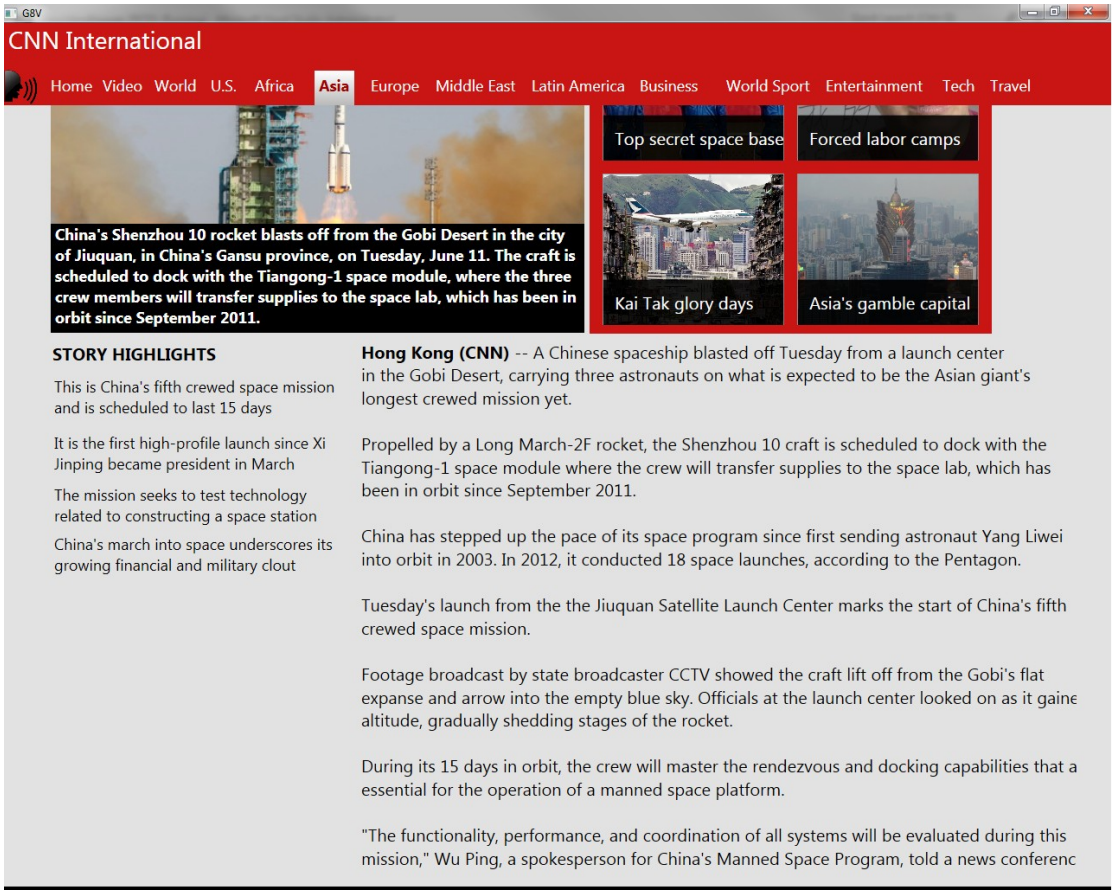

**Final prototype: Application help screen**

The CNN homepage is similar to the paper prototype version. The labels on the top menu are Phonetic, meaning the user can pronounce the words and navigated to the respective section of the website. All gestural interaction takes place with “Push to Press” and “Grip to Scroll”. Those two gestures are supported by the Kinect Development Kit and are ready to use as soon as it is integrated into a project. The elements to interact with are the big square buttons which are links to articles and the page itself, when the content is long enough to have scrollbars. The user simply grips the page and scrolls up or down, left or right to show the hidden content.



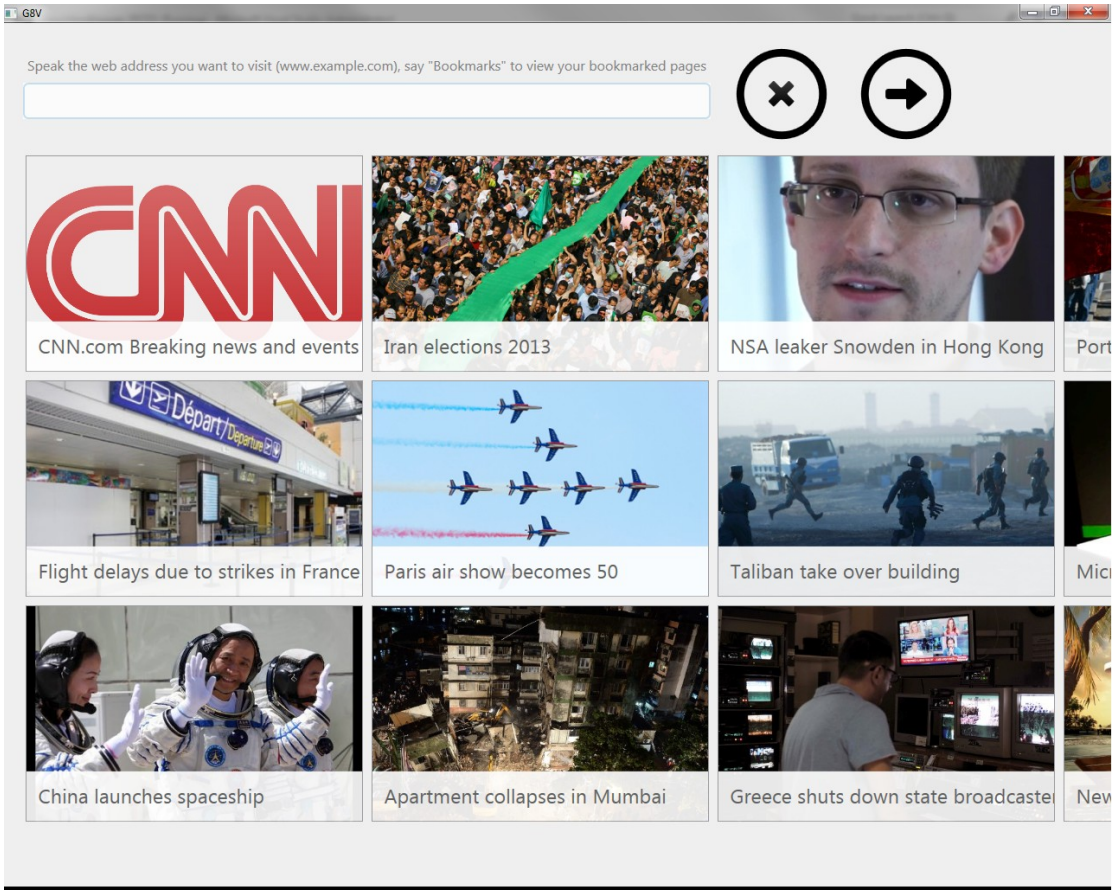

**Final prototype: CNN homepage**

The same design is maintained through all the pages, as seen in this article page.



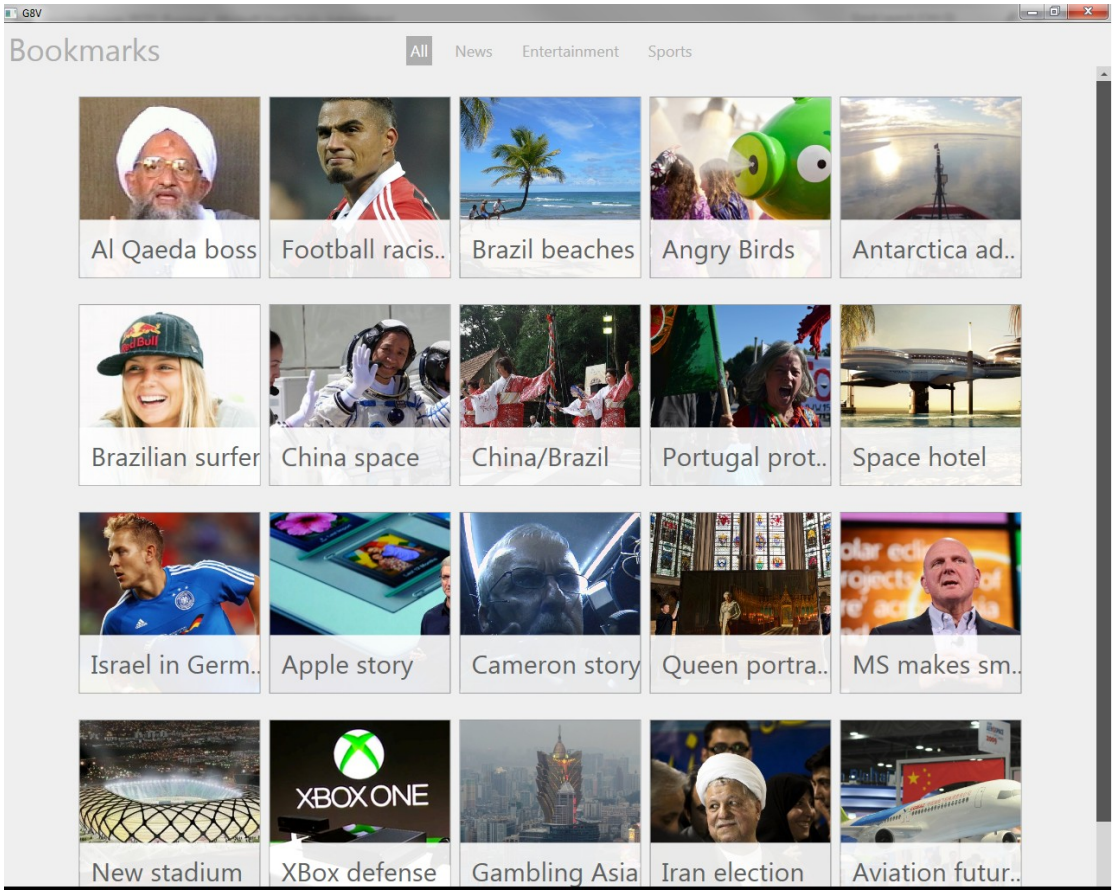

**Final prototype An article page**

When performing a search, the results appear as seen below. This is once again, similar to the paper prototype.




**Final prototype: Search results for the term "cnn"**

The bookmarks are organized in a grid, as seen next.




**Final prototype: Bookmarks page**

Finally, a map is panned by simply performing a grip and pan gesture. The user grips the map but instead of scrolling only along one axis he can scroll along both.


**Final prototype: A page with a pannable map**

## Functionality

The browser, named *G8V*, can perform all the tasks specified in the Main Functions table of the Requirements phase, except for stop and refresh. Refresh can actually be achieved by revisiting the page, there is simply no unique command for that. There is also support for showing the bookmarks.

The difference between entering a page or performing a search has to do with the content of the text field. If the user provides a full URL address the browser will open the page, whereas if the user only speaks a word or phrase a web search will be performed.

### Interaction

Like said before, Interaction with G8V is done with gestures and voice. The gestures were mentioned earlier and are a push one, for the square and circular buttons and a grip one for scrolling pages or panning a map.

As far as speech is concerned, G8V recognizes commands for navigating the browsing history (Back, Forward), going to the start screen of the application (Start) and viewing the bookmarks (Bookmarks). Apart from those commands, the browser can recognize a URL in the form *www.example.com*. All parts of the URL must be pronounced, each ‘w’ separately and the dots. Furthermore, the browser can recognize single words. Lastly, the Phonetic labels which were explained above are also part of the speech vocabulary. They are indicated with an icon of a man speaking.

### Feedback

When the user hovers over a button, the button grows in size and gives the impression that it “pops

up”. At the same time a short sound is reproduced.

When performing a gesture the cursor becomes colored. If the gesture is a grip it changes to a grip shape. That can be seen in the image of the Help section. When a button is pressed a sound is also played, while the same happens when navigating from one page to another.

## Usability Test

The final prototype was put through a small number of test sessions. The primary aim of the tests was to understand whether the implemented ideas of Natural Interaction are appropriate for a browser application. Discovering issues and points to improve is also a very important part of the user test.

## Observations

The observations made during the test were mainly about what the users’ perceived model of the interface was (how they thought it worked) and the problems they faced during the process. Observations were also made about the system and its functionality. The following list sums up the results of the observation.

1. All the users reported a slight confusion on how to begin the interaction. When being presented with the first screen they did not immediately know how to press a button. After moving their hand around for a couple of seconds they realized that it is used as a cursor.
2. Two of the users misinterpreted the “Push to Press” gesture. They tried to tap instead. After a few repetitions they managed to perform the push gesture. The rest of the users also had some problems performing the push gesture in the beginning, usually performing it too fast and losing aim of the button.
3. All the users often kept their hand in the air even when waiting for a page to load or when not interacting with the interface, instead of lowering it to their side.
4. One of the users was crouching a bit forward in order to get a better position for performing a gesture, although this was not necessary.
5. Every user performed well with the “Grip to Scroll” gesture without any noticeable problems.
6. Two of the users had to step back and forth a couple of times until they found the proper place to stand, although it was marked on the floor with pieces of paper.
7. Three of the users made rather frequent use of the “Back” and “Forward” navigation commands.
8. Two of the users did not realize that Phonetic Labels were available on the CNN webpage immediately and tried to use “Push to Press” to go to that section of the website.
9. All the users performed very well when they realized Phonetic Labels were available.
10. Four users were confused with the questions that used the word “application homescreen”. They thought it referred to the “Home” page of the CNN website. They used the command “Home” which is a Phonetic Label instead of “Start” which is a voice command that takes the user to the first screen of the application.
11. One user used the word “Search” to go to the application homescreen, instead of “Start”. The application recognized this word as “Start” and took the user to the right place anyway.
12. False positives were frequent for the Speech Recognition system. Even the sound of a sheet of paper getting wrapped caused a response by the interface at some point.
13. One of the users revisited the Help section a second time.
14. The setup of the sensor and distance from it was not proper at the first user test. The user experienced problems interacting with the interface and had to repeat a gesture multiple times until it got properly recognized. The setup was adjusted after

the end of that test and the following ones had a much smoother flow.
15. Most users tried to use Phonetic Labels in a page (the one with the map) where that functionality did not exist. In other words they did not notice the absence of the speaking man icon.
16. Two of the users used an alternative way of reaching a specific page than what the user test instructed (search with a term instead going through the URL address).

The above observations will be discussed in the Discussion section.

## Interview results

After the end of a test session a short interview with the participants followed. The users were first let alone to share their impressions and thoughts on what happened earlier while they were using the system. Many of their comments matched the observations made while watching them. After that they were asked a few questions more specifically about parts of the interaction and the interface. The list below sums up the points brought up in the interviews.

1. The users reported their confusion as to how to start using the application. One of the users recommended that a short introductory tutorial can be used to solve this problem, e.g. the application asking the user to perform a gesture in order to get started.

2. One of the users reported slight frustration with accidental navigation caused by false positives.
3. All the users stated their satisfaction with using voice to navigate back and forth or navigate to different sections of the CNN website (Phonetic Labels).
4. All the users thought it was easy and enjoyable to use the grip gesture in order to scroll or pan the map.
5. One of the users felt that the interaction space, the area in which the gestures were recognized, was not very clear.
6. The users thought that the presentation of the application was good in terms of layout and organization of elements.
7. One of the users questioned whether browsing the web using touchless interaction can feel natural. Incidentally it was the first user test where the setup was not correct.
8. The users that tried to tap instead of pushing felt it was a more natural gesture in the beginning.
9. One of the users said that a degree of freedom to choose between interacting with a gesture or a voice command would be useful.
10. One of the users said that being able to have the Help dialog available at all times would be helpful for a beginner user.
11. Two of the users agreed that after a few repetitions the interaction felt easier and stated that it was a matter of getting into the habit of it.
12. The users agreed that proper positioning is of great importance to the overall experience.
13. The users, when asked about potential use of an application like that, responded that it could be used for browsing in public spaces, for distance collaboration like phone conferences and at home as a web browser for a TV.
14. One of the users also brought up the need for a “pause” state like the one presented in the paper prototype.

In general the five participants had mostly similar remarks. The next section is dedicated to discussing the results and those observations.

# Discussion

The results and the process followed are hereby connected to the theory. The future developments on Natural Interaction and the current topic are then discussed.

## Result Discussion

The prototype and the results of the user tests are discussed here.

## Prototype

The interactions used for the application are the ones that are recommended and come as part of the Kinect Development Kit. Hummels and Stappers (1998) refer namely to *pushing* and *pulling* as gestures with a manipulative role. Pushing is used in the prototype application, while pulling is essentially a grip gesture combined with a movement of the arm towards a direction.

Pushing is an innate gesture (Human Interface Guidelines, p. 23) since it is a gesture that is usual in real life. However there are problems with that move, the most important one being the fact that there is almost no chance of isolating movement only on the Z-axis. When humans move their arm forward they usually move it a bit to the side or up and down as well (HIG p.100). Despite that, the implementation of this gesture is rather good with Kinect and the problem is minimized. It only occurs when the users perform the gesture at very high speed as shown through the user test. By using bigger control elements (e.g. buttons) which are harder to miss this problem is further addressed.

Gripping is an action that also occurs often in real life too, especially when picking or pulling objects. This gesture works together with Kinect Scroll Viewer, an element of the Kinect SDK for building interfaces with scrollable content. From the user test it was obvious that after the user realized how to perform this gesture, it worked rather smoothly without much strain. HIG state that scrolling is less tiring when the user does not have to reach across the body to perform this gesture (HIG p. 110) and that was kept in mind for the application. Another finding is that it is ergonomically better to scroll horizontally than vertically (HIG p. 110). This was applied in the search results page. In case vertical scrolling is necessary, like in article pages, than the height should be kept relatively low and not spanning the entire screen. Here it should be noted that the Bookmarks page employs vertical scrolling instead of horizontal. This happened for testing and evaluation purposes. For the specific case it did not seem to cause any discomfort or strain since the height of the list was not very long.

The gesture vocabulary consists, therefore, of two gestures. This is a very short vocabulary which leaves some open space for more gestures should it be necessary to add more functionality to the browser that cannot be supported by the current model. The gesture set carries an UI Mindset (HIG p. 28). That means it focuses on efficiency and comfort rather than challenge and fun, which are the characteristics of a Game Mindset.

The final prototype does not have a virtual keyboard. As stated in (HIG p. 116) a virtual keyboard could be useful for brief text entry, but not for extended typing. At the same time writing content through voice is also questioned. The argument for using voice for typing web addresses however is that the relative short length of a URL cannot be considered “content” (HIG, p. 117). The complexity of the URL is another aspect of entering text using voice. Addresses that use abbreviations or shorter versions of a word (e.g. ixdcth, ituniv) must be supported. This is a matter of how the speech dictionary is constructed. Developing it so that it can recognize all possible combinations between separate letters or so that it can recognize parts of a word can solve this problem but the difficulty of this task has not been examined. In any case, the speech recognition engine can understand anything that exists in the

dictionary. If the word “univ” exists, it will be recognized.

It was mentioned earlier in this report that Natural User Interfaces are not only a matter of designing the application but also the content that the application will be working with. In the case of a web browser the content is the webpage. What was mentioned was that current webpages are not optimized for use with gestures and voice. At this point it is appropriate to make a parallelization between the technology researched here and that of mobile devices.
Mobile devices can browse the same Web that a computer “sees” but more often than not they don’t. On the contrary, most high traffic websites offer applications that provide optimized versions of their content for mobile devices such as tablets and smartphones. CNN, for example, has its own application that is designed differently from the one accessed through a browser. Moreover, Responsive Web Design is a way to develop web pages that change their appearance depending on the size of the screen they are seen through. This type of web development is now more important than ever due to the large number of different access mediums that have different screen sizes and resolutions. What the above tells us is that new technologies that suggest different interaction methods, like touch screens, also implicitly demand that the content is adjusted to those new methods. With the same logic in mind, it seems appropriate that if Natural User Interfaces are going to be employed and devices like the Kinect sensor and the Leap will be used for controlling them, then the content for those applications must be designed to suit that new form of interaction. This means, depending on the particularities of each technology, using variations of traditional control elements. In the case of G8V, the buttons used are usually big and square. For the CNN website, the pages have links to other articles and pieces of news but not as many as the normal version of the site has. Those design choices were made consciously to facilitate gestures and voice.

Overall, the prototype showcases what the designer feels is good practice when building a NUI for web browsing. The practical knowledge that exists at this point in time has been evaluated and applied to the degree that it was possible and needed for the desired functionality.

## Usability Test

According to Jakob Nielsen (2000) a small number of users is enough to identify the vast majority of problems with a design. He also mentions that the repetition of usability problems is noticeable test after test within this small user set.

The above statement was verified while performing the user tests for G8V. After the first iteration, the second added one or two more issues to the ones already identified. By the end of the fifth test the list with notes created was almost identical to the four previous ones. The proper step after pinpointing the problems with a design would be to work on those weaknesses and test again. It would be, however, beneficial to add variety to the sample of test users. That would include having both males and females to try out the prototype as well as different age groups. This was a difficult task at the time the tests were taking place because it was during a period when not many people were available.

As far as the actual results are concerned, the proper placement of the user in front of the sensor is of paramount importance, as is the proper positioning of the sensor. This was the very first thing noticed at the user tests and is underlined in HIG (HIG p. 13) where 1.5 – 2 meters are proposed as the optimal distance between the user and the sensor.

Regarding interaction and modality, the interface supports the two main communication channels, speech and gesture. The users seemed to understand the value of combining those two channels to use the interface however at times they seemed to want to use one over the other. Oviatt (1999) states that offering multimodal support does not necessarily mean that users will interact multimodally with the interface; they tend to stick to the channel they feel more comfortable with. That goes in hand with the comment of one of the participants who wondered about the possibility of

letting the user choose which interaction method to perform.

As far as the initial confusion on how to start using the application, this issue falls under the category of what is called *discoverability*. HIG (HIG, p. 42) suggests the usage of quick tutorials for beginners, provision of visual cues or indication of gestures to enhance it. On a related note, Jean and Jung (2011) make an interesting remark on interface exploration by saying that trying out ways to interact can lead to a more entertaining experience and alternative than intended usage of a gesture, with a similar nonetheless result.

One of the important issues raised by testing the prototype was that of erroneous accidental input. Accidental input happened mostly because of false interpretation of a sound by the system. The frequency of false gestural input was much less and took place only when the sensor did not have a clear line of sight to the user, e.g. was blocked by part of the monitor or bad lighting. Silicon Labs on a white paper entitled *Designing Intuitive Gesture Based Human Interface Systems* propose that gestures should be tuned to the requirements of the system and the individual case while the system in turn must learn to neglect inadvertent input. For speech input that is likely harder to achieve with a continuous speech system but should be much easier with an isolated-words one. In that case a trigger-word can be used to activate the speech module after which the user should be able to give vocal commands. This probably removes part of the naturalness together with reducing chances of false positives, but is a matter that requires further testing in any case.

The usability tests brought up significant issues because they showed how unfamiliar with the application users understand its use. That is especially important in more novel experiences like the one presented here, where the user has likely no previous experience with a similar system at all. Some of the users' remarks are believed to be based on this lack of familiarity. This can be related to the same lack of familiarity when a user first starts working with a computer. A proposed interaction might seem difficult but using the mouse for the first time is often somewhat troublesome since there is a lack of calibration that makes the user move the device faster or slower than appropriate. As far as the designer is concerned, a good interaction in terms of usability is after all one that can be learned fast and performed with no significant effort.

# Process Discussion

In the Ideation phase an important clarification was made early on that was based on categorizing desktop applications depending on their purpose. The categories defined were three: professional applications that rely heavily on text (word processors, spreadsheets and similar), game applications with purely entertainment purposes and finally “mid-spectrum” applications. Those last ones are tools that can have multiple roles, from being a tool to perform some complex task (3D modeling, video editing) to others that can have a casual or serious use depending on the user’s intent. In this latter subcategory lies browsing. That fact played a role in choosing this type of application.

This project investigated the key elements of redesigning a browsing application so that it can be used with voice and gestures. The primary focus in the process was *Usability*. Michael Nielsen et al (2004*)* state that Usability comprises of Learnability, Efficiency, Memorability, Errors and Coverage. Learnability is the time and effort needed to reach a specific level of performance. Efficiency refers to the steady performance of expert users. Memorability concerns the ease of usage for the casual users. Errors are the rates for minor and detrimental failures. Finally Coverage is about the actions that can be performed that were discovered against the total number of available actions.

Throughout the process those five principles were constantly revisited either when evaluating related work or while building the prototype. Ideas were discarded or applied depending on whether they violated or not the above principles. In general, the question of whether the application would have legitimate chances of being favored within a certain context over traditional applications was always posed.

A significant part of the process was spent on discovering and assessing similar work. The knowledge that came as a result was important for the progress of the work but there were times when performing the next step looked hard. This has to do with the fact that the field of Natural Interfaces and Natural Interaction is rather new and there are many different perceptions of what those terms mean. Moreover the technology available at the time of implementing the prototype ruled out a few possibilities. For example, Finger Tracking is harder to achieve with Kinect while maintaining high rates of accuracy so implementing a virtual keyboard was rejected. It is likely that the result would be the same in any case, but a device that offers higher precision can at least allow the trial of such a solution.

What would contribute a lot to this project would be a second iteration of prototyping and testing. After consulting the results from the first round of tests, a new set of users would have to perform the same tasks as the previous. This would probably show more accurately how close to achieving a good degree of Usability the suggested interface is.

# Future Work

There are many challenges that Natural User Interfaces and Touchless Interaction must face before this model of interaction can be considered usable for a wide variety of applications.

First of all the skepticism that surrounds gesture interfaces must be addressed. The lack of standardization (Elgan, 2013) causes uncertainty as to how broad the usage of this interaction method can be. Norman and Nielsen (2010) talk about the violation of well-established rules of Interaction Design in gestural touch interfaces but the same doubts are transferred to touchless interfaces too. Kavanagh (2012) states that gestural interaction can be applied anywhere in the digital world but few are the areas of effective utilization. He also says that a system so generic as to cover any area has not yet been developed and claims that if it ever does it will be too complex to be natural.

Establishing models and standards for gestures and voice in terms of interacting with applications is perhaps the first step for future development. In simple terms, any technological improvement will be to no avail if there is no clear specification for the design of gesture and speech sets in NUI applications.

Having taken that out of the way, the next big step is about *context-awareness*. This term was mentioned and discussed earlier but its significance must be underlined. In its simplest form context can be about interpreting surrounding conditions. In its most advanced it can be responding to *emotions*. A system like that interprets the user's emotional state by recognizing signs in the vocal and visual channels, posture and gestures (Voeffray, 2011).

An interesting intersection is that of the concept of context-awareness and speech control. A system that has the ability to understand *when* its user is issuing a voice command to it and not simply speaking to someone else is closer to what humans consider natural behavior, since that is something that happens often in the communication between people. One approach that can offer the solution to this problem is that of constant check of the user's posture. If the user is facing away from the system (monitor) then than can be a good indication that the speech recognition should pause until the user resumes facing it. A more advanced approach would be to perform *eye-tracking* and lock the system when the eyes are detected not to be focused on any part of the screen. At the time of this writing there are companies that work with this technology, such as *Tobii,* to partly control operating systems using gazing. A good implementation of such a concept is very promising in the whole area of Natural User Interfaces and Touchless interaction since it adds another layer of communication, that of using the eyes, to speech and gestures.

The above remarks are general notes on the future work that needs to be done on the area of Natural

Interaction. Specifically about the current topic the future development would refer to refining the prototype and making sure through usability tests that it offers a solid effective user experience. Afterwards, features that have not been implemented in the first iteration like a pause state and opening multiple tabs, would be realized. The aim of the future work would be to bring the standard of this browsing application as close to the standard of typical browsers as possible without neglecting the fact that the input is given in a completely different way.

# Conclusions

This project was driven by the need to understand which the elements of Natural Interaction are and how they could be applied to the design of an application that has widespread and frequent usage. This is important because the design of such an application differs from one with a highly specialized intent and user base. Therefore the effort was placed on coming up with design choices that are suited towards a general-purpose application.

The work done on this project comprises of borrowing or adapting knowledge from a number of different scientific areas. Besides, Interaction Design is a multidisciplinary field that takes into account theories and principles from any practice that can be considered relevant for each project. The conclusions drawn therefore are an interpretation on the designer's part, of the theory and knowledge presented in the previous chapters.

## Gesture Vocabulary Design

As mentioned earlier, a gesture set should be short and consisting of simple to perform task-oriented gestures. The gesture set should work closely with the interface elements, i.e. by seeing an element the user must immediately realize which gesture is associated with manipulating it.

## Speech Vocabulary Design

The vocal commands should support activities that are difficult or too vague to perform with gestures. Indications of where voice is supported should exist and be easily noticeable and interpreted. The sum of commands should be as short as possible so the user does not get the feeling of a large difficult to explore set of actions. The usage of phrases can be a solution to similar-sounding tasks (e.g. “bookmark” could be a command for adding a page to the bookmarks but it sounds very similar to “bookmarks” which would show the bookmarks) but long phrase usage should be limited and single words should be favored instead. Difficult to pronounce words are not a good candidate either, especially if an international user base is the target audience.

## Feedback Mechanisms Design

The way the user receives a response on the results of his actions is perhaps the most crucial part for him to feel in control. Indicators of user action should exist both visually and audibly. Visual indicators should inform and train the user on the interaction with elements by sending cues of progress during his performance. Acoustic feedback should be evident upon completion of an action. Tips and messages can be employed. They have to be unobtrusive (modeless) and be presented when the user is seemingly in doubt about an interaction, e.g. when repeating a failed attempt or taking too long to perform an action.

## Content Design

This area of investigation is perhaps the broader one. Although not in focus of this project, it heavily influences interaction.
On a web browser interface elements can be considered content since each webpage may have its own controls. Those controls should be sizeable and distinct. They should be indicative of how the user must act upon them, in other words provide the right *affordance* for gestural interaction. This principle does not differ from traditional interaction. Moreover, controls that are not in need to be used constantly can be transient and viewed again when the user engages, for instance, by moving a hand.

Content itself should be designed with a focus on clarity rather than richness. What this means is that it is preferable to have less visible content on a screen that is easy to distinguish rather than much content that occupies space. Font-size should be larger than usual and margins should be enough to make the user feel comfortable with using the cursor over the content without high risk of accidental input.

## Context-Awareness capacity

This is another topic that demands in-depth research both technologically and theoretically. This conclusion is about higher-level principles since the whole concept of context-awareness has not been applied, but for a couple of simple cases, to any system of widespread use.

Context refers to implicit user input and environmental conditions. In the first case, signs of user lack of comfort or difficulty should be interpreted and the interface should adjust by simultaneously providing feedback. In the second, employing sensors to detect environmental parameters like light conditions or noise can benefit the interaction and turn the system into a smart trainable one.

## Context of use and environment setting

After all, the actual *intended use* of systems that are based on Natural Interaction principles is what will define *how* they are used. The first question before starting the development of such an interface ought to be what the purpose is behind it. This will probably lead to a series of answers that could standardize a set of interactions similarly to what has happened with using the mouse to click and the keyboard to type.

The application prototyped for this project would be used at home and could be found installed on an appliance like a TV. The concept of Smart Television fits well with the suggested application. A user would be seated at an appropriate distance from the television set and would use the browser without needing to touch any hardware. The relative quietness inside a room offers the ability to effectively use voice as part of the interaction. This setting can provide a smooth and enjoyable browsing experience using Natural Interaction techniques.

Other types of usage have been explored. Public spaces can definitely benefit from applications such as the one under development here with one important consideration: occluding external noise. Perhaps a type of booth where external noise does not reach the system's audio input devices can enable the use of the application in such an environment. Other possible use cases are company workplaces and school classrooms.

# Acknowledgements

The author would like to thank the academic supervisor of this thesis, Thommy Eriksson for the collaboration and useful comments as well as for providing a proper workplace for the project to be carried in. In addition, the author wants to thank the digital agency Humblebee, namely Daniel Solving for helping with the initial startup of the project and especially supervisor Jonas Jonsander for the continuous insight and discussion on this and other interesting topics on the area of future interaction design. Finally, thanks to Achilleas, Azad, Ioannis, Jonas and Peter for their test remarks.

# References

American Speech-Language-Hearing Association. What Is Language? What Is Speech?. Available: http://www.asha.org/public/speech/development/language_speech.htm. Last accessed 12th July 2013.

Andersen, P. (2001). What Semiotics Can and Cannot Do for HCI. Knowledge-Based Systems. 14 (1), p419-424.

Atkin, A. (2006). Peirce's Theory of Signs. Available: http://plato.stanford.edu/entries/peirce-semiotics/#SigEleSig. Last accessed 11th July 2013.

Beaudouin-Lafon, M., Mackay, W. (2003). Prototyping Tools and Techniques. In: Beaudouin-Lafon, M and Mackay, W The human-computer interaction handbook. Hisdale, NJ: L. Erlbaum Associates Inc. p1006.

Bolt, R. (1980). “Put-that-there”: Voice and gesture at the graphics interface. Proceeding SIGGRAPH '80 Proceedings of the 7th annual conference on Computer graphics and interactive techniques. 14 (3), p262-270.

Bowles, C., Box, J. (2010). Undercover User Experience. San Fransisco: New Riders. p99-100.

Brown, T. (2008). Design Thinking. Harvard Business Review, 86(6), p84-92.

Chandler, D. (2013). Semiotics for Beginners. Available: http://users.aber.ac.uk/dgc/Documents/S4B/sem-gloss.html. Last accessed 11th July 2013.

Chattopadhyay, D., Bolchini,D. (2013). Laid-Back, Touchless Collaboration around Wall-size Displays: Visual Feedback and Affordances. POWERWALL: International Workshop on Interactive, Ultra-High-Resolution Displays, part of the SIGCHI Conference on Human Factors in Computing Systems

Cook, S. (2000). Speech Recognition HOWTO. Available: http://tldp.org/HOWTO/Speech-Recognition-HOWTO/index.html. Last accessed 12th July 2013.

Cooper, R., Edget, S. (2008). Ideation for Product Innovation: What are the best methods?. Product Innovation Best Practices Series, Product Development Institute, Stage Gate International. Reference Paper #29, p1-8.

Curtis, B. (2011). Natural User Interface - The Second Revolution in Human/Computer Interaction. Available: http://amddevcentral.com/afds/pages/OLD/sessions.aspx. Last accessed 26th July 2013.

Donovan, J., Brereton, M. (2005). Movements in gesture interfaces. In Larssen, Astrid, Robertson, Toni, Brereton, Margot, Loke, Lian, & Edwards, Jenny (Eds.) Critical Computing 2005 - Between Sense and Sensibility, the Fourth Decennial Aarhus Conference. Proceedings of the Worskshop : Approaches to Movement- Based Interaction, 21 August 2005, Denmark,

Aarhus.

Dorta, T., Perez, E., Lesage, A. (2008). The ideation gap: hybrid tools, design flow and practice. Great Britain: Elsevier Ltd. p3.

Driscoll, T. (2012). Research Paper: What Is Intuition And How Can It Be Used?. Available: http://www.icoachacademy.com/blog/coaching-resources/research-papers/tracy-driscoll-what-is-intuition-and-how-can-it-used/. Last accessed 12th July 2013.

Elgan, M. (2013). Opinion: Gesture-based interfaces are out of control. Available: http://www.digitalartsonline.co.uk/news/interactive-design/gesture-based-interfaces-are-out-of-control/. Last accessed 24th July 2013.

Ferreira, J., Barr, P., Noble, J. (2005). The semiotics of user interface redesign. Proceeding AUIC '05 Proceedings of the Sixth Australasian conference on User interface. 40 (1), p47-53.

Floyd, C. (1984). A Systematic Look at Prototyping. In Budde, ed., Approaches to Prototyping. Springer Verlag, p105-122

Garg, P., Aggarwal, N., Sofat, S. (2009). Vision Based Hand Gesture Recognition. World Academy of Science, Engineering and Technology 25. p972-977

Gope, D. (2011). Hand Gesture Interaction With Human-Computer. Global Journal of Computer Science and Technology. 11 (23).

Hejn, K., Rosenkvist, J. (2008). Headtracking using a Wiimote. Copenhagen: University of Copenhagen. p5.

Hewett, D., Barber, M., Firth, G., Harrison, T. (2011). The Nature of Human Communication. In: The Intensive Interaction Handbook. UK: Sage Publications. p1-6.

Hoiem, D. (2011). How The Kinect Works. Available: http://www.docstoc.com/docs/158488797/How-the-Kinect-Works. Last accessed 9th July 2013.

Houde, S., Hill, C. (1997). What do Prototypes Prototype? In M. Helander, T.K. Landauer and P. Prabhu, eds., Handbook of Human-Computer Interaction. Elsevier Science BV

Hummels, C. & Stappers, P. J. (1998), Meaningful Gestures for Human Computer Interaction: Beyond Hand Postures., in 'FG' , IEEE Computer Society, , pp. 591-596.

Ibraheem, N., Khan, R. (2012). Vision Based Gesture Recognition Using Neural Networks Approaches: A Review. International Journal of human Computer Interaction (IJHCI)). 3 (1).

International Ergonomics Association. (2011). The Discipline of Ergonomics. Available: http://iea.cc/01_what/What%20is%20Ergonomics.html. Last accessed 15th July 2013.

Jaimes, A., Sebe, N. (2007). Multimodal human-computer interaction: A survey. Computer Vision and Image Understanding. 108 (1-2), p116-134.

Jean, D Jung, K. (2011). Providing Freedom of Interaction with Spatial Centered Interfaces. ACM Advances in Computer Entertainment Technology conference. p1-6.

Kavanagh, S. (2012). Facilitating Natural User Interfaces through Freehand Gesture Recognition. CHI'12. p1-6.

Lim, Y., Stolterman, E., and Tenenberg, J. (2008). The anatomy of prototypes: Prototypes as filters, prototypes as manifestations of design ideas. ACM Trans. Comput.-Hum. Interact. 15(2), p1-27

McNeil, D. Gesture: A Psycholinguistic Approach. For Psycholinguistics Section, The Encyclopedia of Language and Linguistics.

Miller, P. (2010). Playstation Move: Everything You Ever Wanted To Know. Available: http://www.engadget.com/2010/03/11/PlayStation-move-everything-you-ever-wanted-to-know/. Last accessed 9th July 2013.

Moggridge, B (2007). Designing Interactions. USA: MIT Press.

Moore School of Electrical Engineering. (1946). A Report On The ENIAC. A Report On The ENIAC. 1 (1), p1-2.

National Institute on Deafness and Other Communication Disorders. (2010). What Is Voice? What Is Speech? What Is Language?. Available: http://www.nidcd.nih.gov/health/voice/pages/whatis_vsl.aspx. Last accessed 12th July 2013.

Nielsen, J. (2000). Why You Only Need To Test With 5 Users. Available: http://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/. Last accessed 19th July 2013.

Nielsen, M., Störring, M., Moeslund, T., Granum, E. (2004). A Procedure for Developing Intuitive and Ergonomic Gesture Interfaces for HCI. In: Camurri, A Volpe, G Gesture-Based Communication in Human-Computer Interaction. Berlin: Springer Berlin Heidelberg. p409-420.

Norman, D., Nielsen, J. (2010). Gestural interfaces: a step backward in usability. Magazine Interactions. 17 (5), p46-49.

O'Neill, S. Theory and Data: The Problems of Using Semiotic Theory in HCI Research. The HCI Group Napier University.

Oviatt, S. (1999). 10 Myths of Multimodal Interaction. Communications of the ACM. 42 (11), p74-81.

Pavlus, J. (2013). Does Gestural Computing Break Fitt's Law?. Available: http://www.technologyreview.com/view/511101/does-gestural-computing-break-fitts-law/. Last accessed 20th March 2013

Pheasant, S., Haslegrave, K. (2005). Bodyspace: Anthropometry, Ergonomics and the Design of Work. 3rd ed. CRC Press.

Raskin, J. (1994) Intuitive Equals Familiar. Communications of the ACM. 37(9), 17.

Redish, J. (2005). Six Steps to Ensure a Successful Usability Test. Available: http://www.uie.com/articles/successful_usability_test/. Last accessed 18th July 2013.

Saiedian, H., Dale, R. (2000). Requirements engineering: making the connection between the software developer and customer. Information and Software Technology. 42 (6), p419-428.

Silicon Labs. Designing Intuitive Gesture-Based Human Interface Systems.

Stark, C. (2012). This is how the Kinect actually works. Available: http://mashable.com/2012/11/29/microsoft-kinect/. Last accessed 9th July 2013.

Stern, H., Wachs, J., Edan, Y. (2006). Human Factors for Design of Hand Gesture Human - Machine Interaction. Systems, Man and Cybernetics. SMC '06. IEEE International Conference. 5 (1), p4052-4056.

Sutcliffe, Alistair G. (2013): Requirements Engineering. In: Soegaard, Mads and Dam, Rikke Friis (eds.). "The Encyclopedia of Human-Computer Interaction, 2nd Ed.". Aarhus, Denmark: The Interaction Design Foundation. Available online at http://www.interaction-design.org/encyclopedia/requirements_engineering.html

Tannen, R (2011). Ergonomics for Interaction Designers, Understanding and Applying Physical Fit in User Interface Research & Design.

UCLA Ergonomics. (2012). Tips for Computer Users. Available: http://ergonomics.ucla.edu/homepage/office-ergonomics/tips-for-computer-users.html. Last accessed 15th July 2013.

Valli, A. (2005). Notes on Natural Interaction.

Venkataramesh, S., Veerabhadraiah, S. (2013). Effective Usability Testing – Knowledge of User Centered Design is a Key Requirement. International Journal of Emerging Technology and Advanced Engineering. 3 (1), p627-635.

Vimala C., Radha V. (2012). A Review on Speech Recognition Challenges and Approaches. World of Computer Science and Information Technology Journal (WCSIT). 2 (1), p1-7.

Visser, W. (2006) The cognitive artifacts of designing. Lawrence Erlbaum Associates, Mahwah

Voeffray, C. (2011). Emotion-sensitive Human-Computer Interaction (HCI): State of the art - Seminar paper. Emotion Recognition. p1-4.

Welch, G., Foxlin, E. (2002). Motion Tracking: No silver bullet, but a respectable arsenal. Computer Graphics and Applications, IEEE. 22 (6), p24-38.

Wisniowski, H. (2006). ANALOG DEVICES AND NINTENDO COLLABORATION DRIVES VIDEO GAME INNOVATION WITH IMEMS MOTION SIGNAL PROCESSING TECHNOLOGY. Available: http://www.analog.com/en/press-release/May_09_2006_ADI_Nintendo_Collaboration/press.html. Last accessed 7th July 2013.

Zimmerman, T., Lanier, J., Blanchard, C., Bryson, S., Harvill, Y. (1987). A hand gesture interface device. · Newsletter ACM SIGCHI Bulletin. 17 (SI), p189-192.

# Appendix

## G8V Test Instructions

1. Open the Help window, by pressing the help button.
2. Read the instructions in the help section. When you are done, close that window.

3. Visit www.cnn.com.
4. Find the article "**Brazil's 8 best beaches**".
5. Answer these questions related to the article:
    a. What are the people in the Ipanema beach photo doing?
    _____
    b. How many miles of coastline does the state of Bahia have?
    _____
    c. What is Porto de Galinhas famous for?
    _____
6. Go to the site's page with European news.
7. Find the article about the **portrait**.
8. Answer these questions related to the article:
    a. Which queen is it talking about?
    _____
    b. Where has the incident taken place?
    _____
    c. Who is the first that was coronated at that place?
    _____
9. Return to the application's homescreen.
10. Perform a search with the term "**cnn**".
11. From the list of results, click on the one that talks about **Apple**.
12. Answer these questions related to the article:
    a. What does the author think Apple will be most known for in the future?
    _____
13. Navigate back to the search results.
14. Find the result that speaks about **Portugal**.
15. Answer these questions related to the article:
    a. What is the name of the Portuguese Minister?
    _____
    b. What should the growth strategy be based on? (hint: below the middle)
    _____
16. Go back to the application's homescreen.
17. Go to your bookmarks.
18. Find the bookmark about the **futuristic hotel**.

19. Answer these questions related to the article:
    a. What is the name of the hotel?

       _____

    b. What island is it going to be built on?

       _____
20. Go to the Asia section of the website.
21. Open the main article.
22. Answer these questions related to the article:
    a. Where did the rocket launch from?

       _____

    b. How long will it stay in orbit?

       _____
23. Go the "Tech" section of the website.
24. Open the main article.
25. Answer these questions related to the article:
    a. What feature provides for a better gameplay?(hint: scroll towards the end)

       _____

    b. How much more expensive is Xbox One than Playstation 4?

       _____
26. Go to the CNN frontpage (Home).
27. Open the main article and view the video.
28. Pause the video after a while.
29. Return to the application homescreen.
30. Go to www.continentmaps.com
31. Write down 3 *neighboring* countries in Africa:

    _____

    _____

    _____