Lily Gebremariam

Using Tilt-Gesture of Mobile Phone as a Game Controller: User Experience Evaluation

Thesis

Helsinki Metropolia University of Applied Sciences

Bachelor of Engineering

Media EngineeringThesis

23 October 2017

Abstract

Author	Lily Gebremariam
Title	Using Tilt-Gesture of Mobile Phone as a Game Controller: User Experience Evaluation
Number of Pages	40 pages + 2 appendices
Date	23 October 2017
Degree	Bachelor of Engineering
Degree Programme	Media Engineering
Specialisation option	Human – computer interaction
Instructor	Merja Bauters, Researching Lecturer

The purpose of this final year project was to execute an experiment on using tilt-gesture of mobile phone as a game controller on large screen in a public setting. Overall user experience regarding interaction technique includes the following: intuitiveness, social acceptance and feasibility were examined from the experiment.

To implement the experiment a collaborative real-time “ping pong” prototype game application was developed and used as a case study. The game uses tilt-gesture of mobile phone as a remote controller to let a pair of players play the game on a big screen.

Two experiments were conducted with Metropolia and Laurea Universities of Applied Sciences (UAS) students. Qualitative research methods such as interviews, survey questionnaire and observation were implemented to evaluate the user experiences. The data recorded from the experiment was used to analyses the user experience on three levels; “experience”, “an experience” and “co- experience.

The results showed that turning large LED screens in public use into interactive game applications enables users to play a collaborative game without buying a game console. This was experienced as fun and intuitive. The results also emphasized that understanding the experience and social behavior of users in a public setting is a crucial aspect in designing gesture based- interactive applications.

Feedback from the users helped to identify the improvement issues in the game as well as important features of the game UI. Advantages and drawbacks of the game implementation technique were collected, which could be used for future study.

Keywords	Tilt-gesture, controller, user experiment, interaction technique, public display, ping-pong

Contents

List of Acronyms

BBG - Browser-Based Game

UI - User Interface

HTML - Hypertext Mark-up Language

PC - Personal Computer

TCP-Transmission Control Protocol

API - Application Programming Interface

W3C- World Wide Web Consortium

DOM- Document Object Model

XML- Extensible Mark-up Language

1 Introduction

Recent studies have found making a phone call is the sixth most common use of mobile phone. In today's world, mobile phones are essential tools in daily life. To keep up with the high demands of the smart phone market, mobile phone manufacturers are equipping their products with many extra gadgets to give their costumers pleasant experiences. Mobile phones are tightly attached with humans, for instance, they are used to control other devices, such as television set or games. These phones come with a built-in accelerometer, gyroscopes, compasses intended to capture motion and orientation data.
Moreover, most browsers support HTML5 DeviceOrientation and DeviceMotion events. These events provide important information about the orientation of the device, motion, and acceleration of the device. Motion events bring rotation data, that is how much the device tilts front to back, side to side and the direction in which the device is tilting. [1.]

There are several applications, which use device orientation, such as game application, mapping, and navigation system. Obviously, the advantage of device orientation is enabling the device to act as a controller in game applications. For example, a character can be made to jump or move just by tilting the device in different axis. [1.]

In a web- based game applications, the arrow keys on our computers or the touch pads are used for interaction. What if the phone itself could be used as a game controller? Is using mobile device as a controller easy, fun, intuitive and socially acceptable in front of a public screen? The idea of using mobile phone as a controller is not a new concept. There are different browser game applications designed in synchronization with computer. However, interacting with these kind of games and quizzes on large screens is less in common. Little work has been done for examining the user experience of playing in public places using mobiles as controllers.

In this thesis work, the main goal is to plan and execute an experiment on the game prototype which uses tilt-gesture (device orientation) of mobile phone as a game controller. I studied how mobile device orientation can be used as input in creating such game application and I also developed and tested the prototype.

Prototype is a browser-based game application designed withHTML5 Canvas, Node.js and socket.io. It is a collaborative real-time “ping pong” game application, which uses mobile orientation as a controller to let a pair of players play the game on a PC, tablets, or a big screen. Based on the user experience approaches, I have conducted user experience experimentation and analyzed the experiments. The aim was to find out if using the tilting motion of a device is easy, fun, socially acceptable, the drawbacks and the feasibility of using device orientation in developing a browser game based on this approach and the overall user experience.

2 Previous research, gesture recognition and interaction techniques

Long before the emergence of using mobile devices as a controller, Wii-gaming console had a great influence in the current built-in accelerometers-based gestural controls in the mobile devices. [2.]

The Japanese’s company Nintendo developed the Wii remote in 2006. The company has achieved extraordinary revenue and attention in the game console market since its launch. The “Wii mote” is a game controller which works wirelessly by sensing translational acceleration in the three coordinate axes and rotational orientation. The controller can do this by using the accelerometers and the light sensors that are built in it. The bold feature of the remote is its motion sensing ability, which allows the user to interact with the game characters on screen through gesture recognition. [2.]

After the invention of “Wii-remote”, using the same advantage of accelerometer has increased in mobile phones applications. Different game applications have come up with using mobile devices as a controller rather than having a separate device for controlling hardware. These games are in most cases 2D games, which do not have a high game control demand on the device. Almost all the games designed in this way use the client–server system architecture, using the mobile device as a client and running the server application on the computer.

Most games are still aimed at being used in synchronization with PCs. However, there have been games developed that use large public screen instead of PC. Vajk and Coulton (2008) mention the concept of playing games on large displays in public places using each player's mobile phone as a controller for such games. They came up with a 2D car racing game where the movements are controlled by device’s accelerometer. Nokia 5500 mobile phone was used as a controller. It was connected to a large game screens thorough Bluetooth technology. The game uses transmission control protocol (TCP) protocol for the communication between the client component and the server. The phones had multipurpose in the game, not only does it act as a controller, but it also gives information to the players on the screen – such as the score. The user can interact with the game on the screen by motion and swipe gesture as an input using the built-in accelerometer. Moreover, the user can use a virtual key pad. The input methods were evaluated in terms of usability by controlling the developed 2D space shooter. [2.]

The technique of interaction between mobile phones and different interfaces varies greatly. The mobile devices' built-in features such as sensors, high quality cameras, accelerometers, pressure sensors, enable this interaction to happen with PCs and big screen displays. For The big LCD screens in public places, which do not have a potential interaction initiative, these interaction techniques are promising.

Ballagas, Borchers, Rohs and Heridan (2006) mentioned two interaction techniques based on the data input method. A direct technique is to use the mobile phone as a controller in making a direct interaction with the screen interface by selecting the favoured input. Indirect technique creates interaction using a mediator; the mobile device lets the user control the screen interface by using the mouse cursor for selecting the input. Most of these interaction techniques are implemented by using computer vision techniques or motion sensors in the devices. [3.]

Another interaction technique, namely gesture, can also be used. Due to the easy use and the built-in motion sensors in most smartphones, the technique of using motion sensor as gesture-based interfaces has become increasingly common. The gesture enables the user to fulfil a certain type of task or experience. For example, some Map applications enable users to locate a certain place by making a pointing gesture. [4; 5.]

There are three common and seamless gesture controls with sensor–enabled mobile phones and large displays: orientation- aware (tilt) gesture, throwing gesture, and motion gesture. Orientation aware interaction involves the tilt gesture of a mobile device as a remote control to perform a certain task. The method is convenient for applications with continuous interactions. The interaction involves the up, down, left, and right movement of the phone in moving some characters position (such as in games), to highlight drop-down menu (widgets) or navigating some areas in a maps application. The four tilt movements could be either discrete or continuous. [4; 6.] (See figure 1)

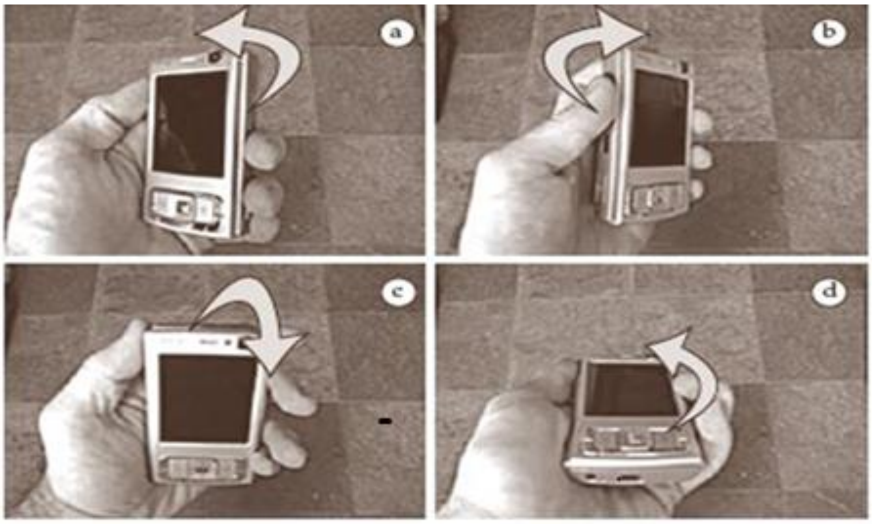


Figure 1. Tilting gestures for walking in the environment. Reprinted from Valente, Luis, Souza, Clarisse Sieckenius de, & Feijó, Bruno (2009) [7].

The throwing interaction is an intuitive technique to share personal media data such as photos, video, recorded music with the sync display. It simply involves throwing gesture of the mobile device to transfer media document to the large screen. In most of the applications designed based on this approach, the mobile phone and the display are communicating over Wi-Fi or Bluetooth. [8.]

One use case scenario of throwing interaction is the “MobiToss” application, designed by Oulu media lab. The application uses a throw gesture control to share multimedia art using a mobile phone. The users record a video or take a photo with the mobile phone and using the throw gesture of the mobile phone to transfer the item onto a large display (see figure 2). The transferred photo or video can be instantly seen on the screen. Moreover, the user can manipulate the item by tilting the phone in different angles. The application was designed to be used by any phone with a built-in accelerometer sensor, the phone and the display are connected via Bluetooth. [8.]

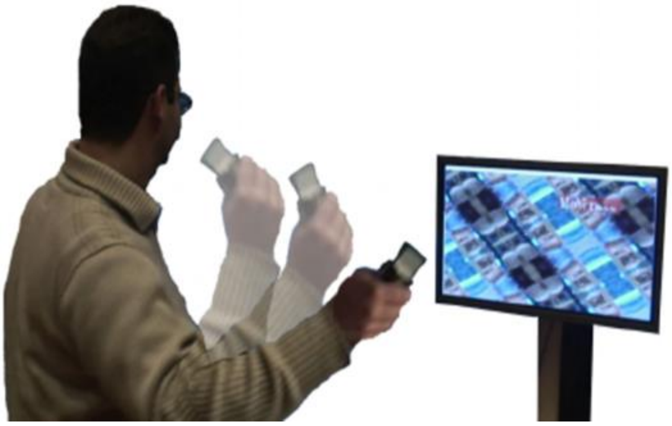


Figure 2. Throwing a video to the large screen. Reprinted from Scheible, Jürgen (2008) [8].

Throw gesture technique, used in "MobiToss" is convenient for the user in many ways. First, the user is using his/her personal phone, which cuts the need of buying extra proprietary application and gadget. Second, using the phone the user can receive a confidential content such as web links and music video. Third, MobiToss application allow users to participate anonymously in a public group by creating content and sending it wirelessly. [8.]

Some web applications use a motion gesture as an input to perform a certain task. For example, user can shake the mobile device to clear or flip a web page. These applications control the device's acceleration and use it for signal processing.

Different gesture control is possible with accelerometer and sensor built in devices. The data gathered from the sensor is the first crucial event in gesture recognition using mobile devices. This comprises setting up an event listener for the accelerometer, analysing the deviations during events, and taking an action based on the analysis. [8; 9.] In the next chapter, I will discuss these processes in detail.

2.1 Device Orientation

Accelerometer is a device intended to detect acceleration, the rate of change in velocity, but in smartphones, the device detects the change in the orientation and rotation of the device. [10.] Most smartphones have a built-in accelerometer used to calculate the tilt motion and the acceleration by sensing movement and vibration. First, an accelerometer first of all determines the orientation of the device about the earth's gravity when the device is at rest. Secondly, an accelerometer detects the movement of the device with respect to the initial point as a reference. It measures the relative motion of a physical thing relative to gravity in one dimension. By measuring the downward force of gravity, it can help to determine how the device is angled. Mobile devices have three accelerometers situated in x, y, z-axis respectively and each measures acceleration along its dimension. [11.] Using the vertical gravitational acceleration as a reference, it is possible to calculate the tilt angle by a simple trigonometry. Many devices also determine the portrait and landscape screen orientation modes in a way by calculating the tilt angle in this approach. [2.]

The same types of accelerometers employed in the Nintendo Wii remote controller are used in mobile devices. These include a mini cantilever beam connected to a seismic mass sealed in a gas pocket. Whenever the device moves, the beam bends under accelerative forces, and these are measured and reported. [2; 11.]

The device orientation application programming interface (API) is a set of functions that will allow an application to access data from an OS or other applications and services. World Wide Web Consortium (W3C) specify a DeviceOrientaion API as a set of document object model (DOM) events (API) that gives data about the orientation of a device for hypertext mark-up language (HTML) and extensible mark-up language (XML) documents (web pages). The data provided by the API comes from the device's gyroscope, the accelerometer, and the compass depending on the devices' usage. [12; 13.]

Device orientation events are things that happen when a user manipulates a web page or an application. For example, if we are loading a page or clicking a key pad, changing a device orientation in our case, is an event. While using JavaScript in the web application, the language is enabled to react with these events in the browser. [12; 13.]

In relation to the earth’s gravitational force; web enabled devices can determine their orientation. JavaScript events handle this information, there are two JavaScript events in handling this data namely “DeviceMotionEvent” and “DeviceOrientationEvent” provided by Device Orientaion API.

When an accelerometer detects a change in the device’s orientation, the measured data is fired by DeviceOrientaion Event. With this data coming from the event, we can interact in response to the orientation change of the device in the browser. This event provides the physical positioning of the device running the application. The deviceorientation event is initiated each time. A new information comes from an orientation sensor about the current orientation of the device in reference to the Earth coordinate frame [12; 13.]. The change in the acceleration is listened by the other event,” DeviceMotionEvent”, this event is most commonly found in the mobile phone device. In this case, the sensors in the device can measure the acceleration. The measured acceleration is fired by DeviceMotionEvent, it listens and process this information. These Events provide information about the acceleration for the device's position and orientation for the application. This event data can be handled in application code and react whenever there is a change in the speed of the devices or orientation.To let an application code to react with the events as they happen, one need to create a JavaScript Functions handler, and this is done in the browser. People can register function for every specific event as a handler in that case to listen the deviceorientaion event and get the orientation change, the user has to create function handler like” function handleOrientation (event)”, this function give an updated orientation data whenever a new event happens. [1; 13.]

2.2 Processing the orientation events

The orientation event has values given by the browser. These values are based on three types of data namely alpha, beta and gamma. When using these events, the user must consider the coordinate frames of the earth and the device. Hence before discussing how the events are processed, it is appropriate to know what these orientation data are and how it is measured. The centre of the device is used as a fixed coordinate axis and it is always measured in reference with the standard orientation of the screen, which is the portrait. [11; 13.]

- The x axis is in the plane positive toward the east and negative in the west. The axis is always perpendicular to the y axis.
- The Y axis is in the plane positive toward the north (top of the screen) and negative in the south. The axis is always perpendicular to the x axis.

- The Z axis is perpendicular to the horizontal plane, it is positive away from the centre of the earth and negative towards the centre of the earth.

Figure 3 below helps to visualize the planes clearly.

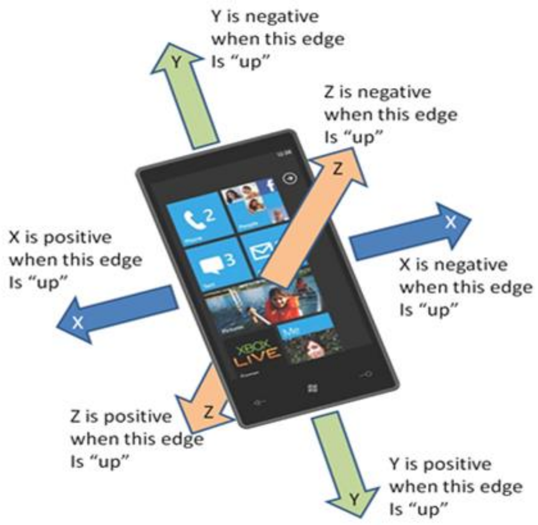


Figure 3. Rotational axis of mobile device. Reprinted from Sai Charan Raj (2015) [15].

The rotation angles created when the device rotates about the earths coordinate frame are represented as; "alpha", "beta" and "gamma". "Alpha" measures the rotation angle in degree along the axis perpendicular to the screen. "Beta" indicates the rotation angle created when the device is tilted front-to- back. Whereas "gamma" measures the angle the device is tilted left-to –right (see figure 4). [13; 14.]

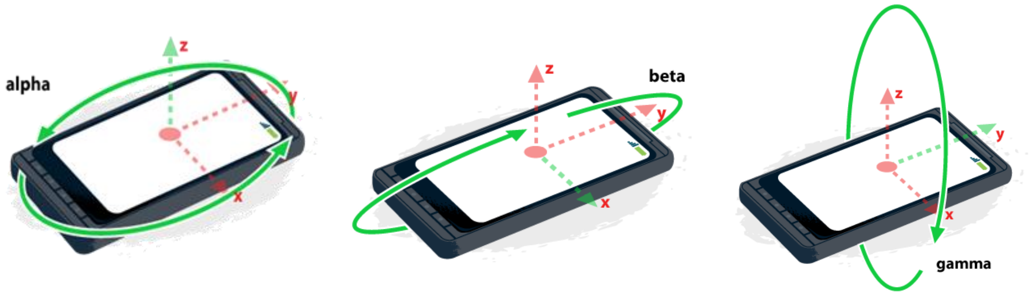


Figure 4. Rotational Angles. Reprinted from dphenderson (2016) [14].

The alpha value changes, when the device rotates around the Z-axis. At the position where the screen of the device faces the Earth's North Pole value, the angle is 0°and as the device tilts to the left, the value will increase. The beta value changes, when the device rotates around the X-axis. The angle measures 0° at the position where both the bottom and top of the phone are at equal distance from the Earth's surface. As the device tilted to the front the angle increases towards 180° and -180° as it tilts backward. The gamma value changes, when the device rotates around the Y-axis. At the position where the right and left sides of the device are at equal distance from the earth, the angle is 0°. As the device tilts to the right, the value will increase towards 90°, in the contrary when it is tilted towards the left it starts to decrease towards negative 90°. [13; 14; 16.]

2.3 Orientation values

Now it makes sense to define the DeviceOrientationEvent, the event contains the values alpha, beta and gamma: As I stated earlier by attaching an event handler, it is possible to listen for these events and respond to any changes. When the event handler is attached, it will receive one row of type DeviceOrientationEvent, which contains four values of the event:

- DeviceOrientationEvent.absolute
- DeviceOrientationEvent.alpha
- DeviceOrientationEvent.beta
- DeviceOrientationEvent.gamma

The value reported for each axis indicates the amount of rotation (the angles stated above) around a given axis in reference to a standard coordinate frame. The DeviceOrientationEvent.alpha value represents the motion of the device around the z axis, represented in degrees with values ranging from 0 to 360.

The amount of rotation value is stated as an angle for each axis in reference to the coordinate frame.

- The DeviceOrientationEvent.beta value represents the motion of the device around the x axis, represented in degrees with values ranging from -180 to 180. This represents a front to back motion of the device.
- The DeviceOrientationEvent.gamma value represents the motion of the device around the y axis, represented in degrees with values ranging from -90 to 90. This represents a left to right motion of the device. [13.]

A JavaScript function called handleOrientation () listens the orientation and motion events, hence the OrientationEvent has to bind with the event handler and a function is created in the following way;

```
Function handleOrientation (event) {
     var absolute = event.absolute;
     var alpha    = event.alpha;
     var beta     = event.beta;
     var gamma    = event.gamma;
```

When the event handler is appealed, it will receive one row of type DeviceOrientation-Event, which contains four elements mentioned above. The function gets an updated orientation data periodically whenever there is a change.

Moreover, the whole function needs to be attached to the “window” object, which is a global JavaScript objects that allow the JavaScript to "talk to" the browser. The code below is created to attach the function to the “window “object;

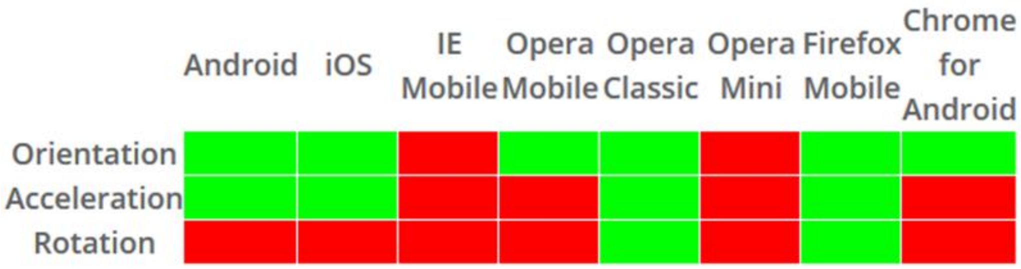
```
window.addEventListener("deviceorientation",
this.handleOrientation, true);
```

In the same way we can append the function to the “window “object as;

```
window.addEventListener("devicemotion",handleMotion,
true);
```

Appending the above line of code as a script Inside the HTML page of the browser application, enables the application to listen to the Deviceorientation event and get the orientation change of the device as it tilts in every direction in real-time. [1; 14.]

The W3C device orientation API provides marvelous information of the specification, which is not stable and susceptible to any change. The API has several variabilities in deferent browsers and operating systems. For example, the absolute property is not defined in mobile Safari. Apart from Internet Explorer and Opera Mini, the device orientation event is supported by most of the mobile browsers. Likewise, these browsers have difficulties in supporting deviceorientation event with windows 7 operating systems. This data is not accessed by native Android browser. A third-party browser such as Firefox is needed. [13; 17.] As a result, one needs to check the browsers the compatibility for usage. Below, figure 5 guides as a compatibility checker for different browsers across the events.



	Android	iOS	IE Mobile	Opera Mobile	Opera Classic	Opera Mini	Firefox Mobile	Chrome for Android
Orientation	🟩	🟩	🟥	🟩	🟩	🟥	🟩	🟩
Acceleration	🟩	🟩	🟥	🟥	🟩	🟥	🟩	🟥
Rotation	🟥	🟥	🟥	🟥	🟩	🟥	🟩	🟥

Figure 5. Browser compatiablity. Reprinted from Ruadhán O'Donoghue (2013) [17.]

2.4 Application areas of using device orientation

Thanks to this device orientation API, people can move a character by tilting the device in the direction they want their character to move. Game development is doubtless the most interesting area in which motion and orientation data are useful: for example, one might think of using this data to control the direction of characters, vehicles, or balls, and make them jump.

Beside games, motion and orientation data can be used in open web Applications to implement gestures, such as the shaking gesture, or combined with other data, such as geolocation data, to improve mapping web applications. [4; 9.]

Someone, for example, might use device orientation to rotate the view of a map as the device rotates. In a web and video game applications, device’s orientation controls the tilting of the device in the direction of the character. In these games, the application interface must continuously detect the level of the phone. However, the continuous detection of the phone will be disturbed when the user makes accidental interface actions. Moreover, the user’s wrist does not rotate the phone to make a perfect tilt along the x and y- axes, determining the continuous level of the phone is problematic. During such scenarios, Simple tilt detection does not work well for, a more powerful solution is required. [9.]

Accelerometer-based gesture recognition is a crucial area highlighted in human-computer interaction. Some web applications use a motion gesture as an input to perform a certain task. For example, one can shake the mobile device to clear or flip a web page. These applications control the device's acceleration and use it for signal processing. Inertial sensors built-in the mobile devices paved the way for interaction technique based "gestures" or movement of the users holding the device. The measurements coming from the accelerometer and gyroscope in the mobile devices are used to enable gesture recognition system. [18.]

In navigation and mapping applications, the device's orientation events are needed to align the orientation with the real-time location of the subject holding the device and up-to-date the map as the user moves around. With GeoLocation it is possible to turn these events into a navigation system. [18.]

Map based and augmented reality applications in mobile browser use Geolocation API, to determine location in terms of position and coordinate objects. [19.] The tilt- gesture control is mostly used in several game and map navigation applications.

3 Case study

Ping pong is a basic "tennis like" game designed for collaborating two players at once. It is one of the oldest video games appeared in the industry. Today it is one of the significant games in terms of generating high revenue in the game industry. [20.] The game includes two paddles and one ball. As in the tennis game, the rule is to throw the ball using the paddles and to conquest the opponent (see figure 6). The player scores a point whenever the opponent misses the ball. Two players could use two paddles, or one player can play with the computer as an opponent.

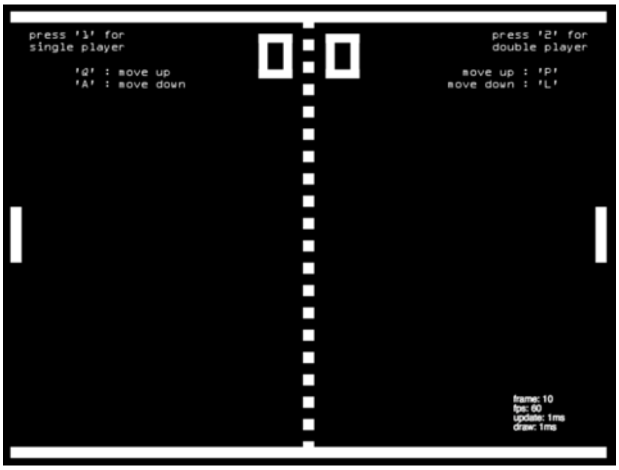


Figure 6. Classic Ping pong game on computer UI [20.]

Any game which can be played on web browser over the internet is called a browser–based game. A browse by query (BBG) is written in JavaScript, hypertext pre-processor (PHP) and HTML5 programming languages. It can also use additional plug-ins and languages as a backend programming such as Java and web graphics library (WebGL) to allow the devices acceleration. [21.]

There are many browser-based games available on the market. Browser games are intended to be played on different devices with different operating systems such as: mobiles, tablets, and computers. Because of this, the games are designed to be less demanding graphically and have a small system requirement. Most of the games are classic 2D games, however, it is increasingly common to see those classic games reinvented as a browser game even in 3D using HTML5 and WebGL. For example, Pac-Man is a long-known arcade classic game, which become available as a browser game. Run escape is a popular massive multiuser online role-playing game (MMORPG) browser game written in Java.

3.1 The prototype – browser based ping pong game overview

The game employs HTML5 canvas, DeviceOrientation Node.js and web sockets. The game is realized with a client-server architecture, the front end of the game is made using HTML5 canvas and JavaScript on the client side. Using basic principles of physics and box2D, HTML5 canvas allows animating the objects. In JavaScript, the <canvas> element added in HTML5 is used to create graphics via scripting. For example, it can be used to make graphs, photo compositions, create animations, or even do real-time video processing or rendering. The <canvas> element is used by WebGL to draw hardware-accelerated 3D graphics on web pages. [21.] The whole logic of the application except the communication between the phone and the host is made using HTML5 Canvas. The paddles drawing, the board features and the ball interaction mechanisms are made with the help of canvas.

Node.js is used as a back-end server, to serve the game application. Socket.io is used to create the web socket connection between the browser on the host screen and the phone. Node.js will creates a unique id that is linked with a socket connection to the browser game. Once the connection is created as a “socket”, a “route” is made through the server and all commands from the phone are sent to the browser game and back to the host through the socket.

Socket.io is a powerful tool for creating real-time applications with bidirectional communication between the server side and client side. A socket is a single connection. [22.] In the prototype game, a socket is created whenever a device is connected to the server. Node.js and socket.io are used to create this instance as needed.

The rule of the game is the same as the classic computer ping pong game, except the controller is a mobile device instead of a computer key arrow or touch pad on the usual browser ping pong game. It is a duo game with two opponents, and each player can use his /her mobile phone device as a paddle. The game interface could be on a computer or a large screen. By tilting the device in the direction of the movement of the paddle, the players can hit the ball to the opponent's direction. The two paddles are placed one on each side of the playing field (left and right) opposite to each other. Figure 7 below shows the lay out of the game.

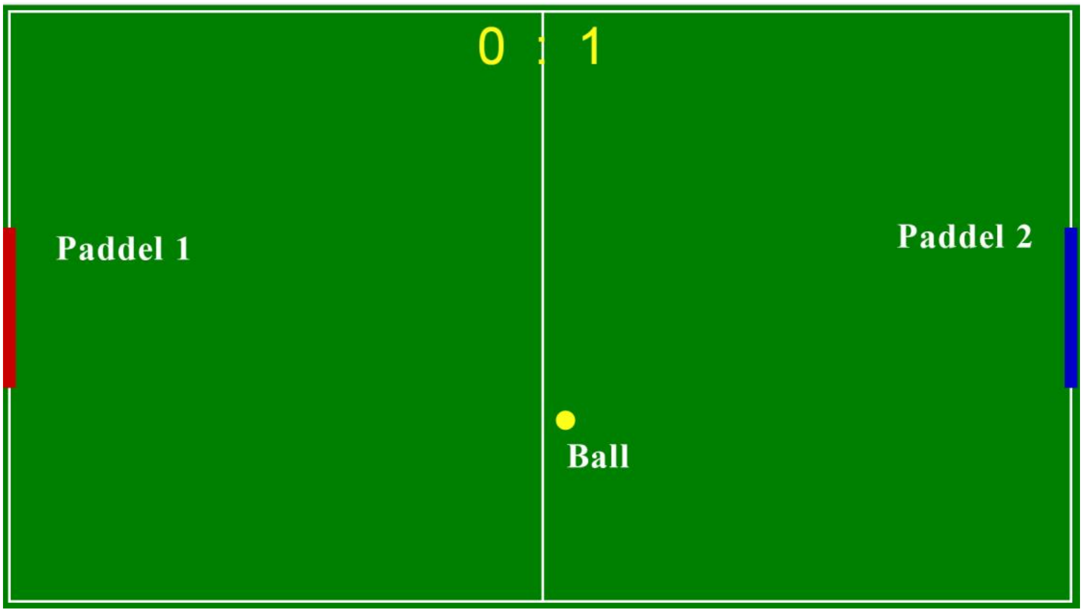


Figure 7. Screen shots of Playground UI.

In this prototype, the paddles are designed in the vertical axis, assuming the device in the standard, portrait orientation. The players can move their corresponding paddle either up or down. There is a ball, which will be moving constantly with in the playing field. If the ball collides with the top edge of the field, the bottom edge of the field, or one of the paddles then it will bounce back. But if the ball collides with either the right or left side of the field then point is scored on the player who lets the ball pass to collide with his side of the field's edge. The points scored by each player are recorded and the first player to score 5 points will be the winner. Figure 8 below shows the flow of the game.

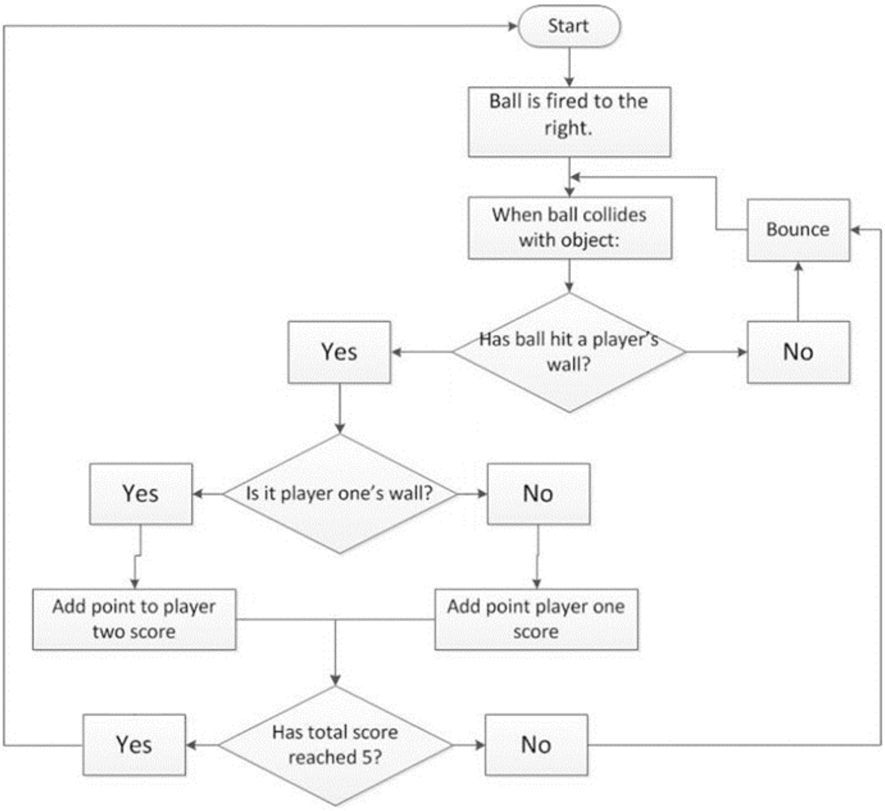

Start
Ball is fired to the right.
When ball collides with object:
Bounce
Has ball hit a player's wall?
Yes
No
Is it player one's wall?
Yes
No
Add point to player two score
Add point player one score
Has total score reached 5?
Yes
No

Figure 8. Game flow chart.

To start a new game, a web link to the server deploying the game is used. To run the game a device is needed which will be used as a host. The host will be used to display all the necessary graphics and instructions needed to play the game. The host can be any device that has a web browser. When user connects to the server an option is presented whether the device connected is going to be used as a host or a player. If connected as a host, then that device is responsible for running the game. Now the host is only capable of starting / restarting a game, ending a game, and displaying all the necessary graphics.

Adding a player

By using the web address or QR code a new player can join the game using a mobile device that is capable of producing device orientation data. When a new device is connected as a player it is indicated to the player by displaying the colour that corresponds to the colour of the paddle that the player will be controlling. The players' input data is real time and it is connected to the server. The server takes the input data and send it back to the game in the browser.

Rendering the graphics

The first thing is to define the canvas on which all the necessary graphics are drawn and animated. The canvas will have a 2d context and the size is equal to the full browser size and will update on resize. Then all the necessary shapes are rendered on this canvas. Figure 9 below shows the rendering of the shapes on the canvas.

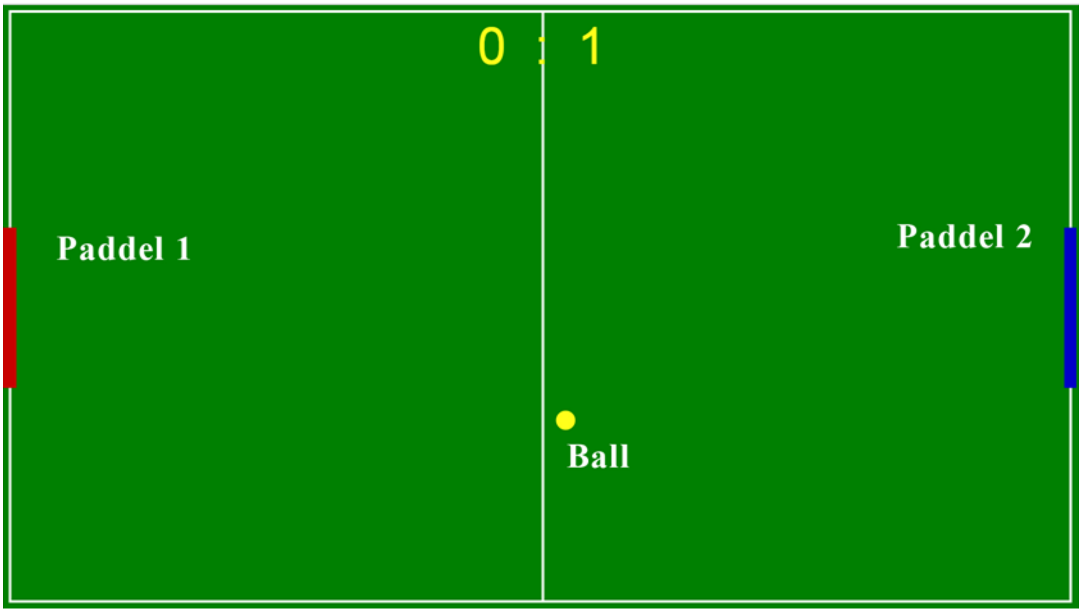


Figure 9. Rendering of the shapes on the canvas

To add the animation, the canvas is rendered every 15miliseconds using the 'setintrval();' function.

User interaction with the game

Now that there is a game logic to work on, the next step is to find out all the possible inputs and outputs of the game. To start with there should be two users interacting with the game, and each user will have one paddle to control. The movement of the paddle is constrained only in one direction, which is either top to bottom. In this case, the paddle's orientations are chosen to be left and right therefore the axis of control is going to be y-axis. From the previous chapter, it is found out that a mobile phone devise's orientation information can be read at any time using a web API. The goal is to make the paddle change its position according to the data sent from the mobile phone device. At this point, there are a few challenges that can easily be seen. First how the data is sent from mobile phone to the devise that is going to process the game is explained followed by explanation on how this data is used by the game.

The games can be played using the links or QR code below:
For playing the game using “right to Left” tilt the phone about the y-axis as show below in figure 10 URL: https://lilyponglr.herokuapp.com/

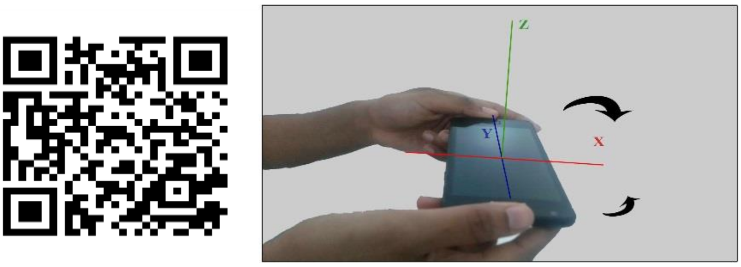


Figure 10. QR code and Rotation of Device in Y-axis.

For playing the game using “Up and Down” tilt the phone about the X-axis as show below in figure 11. URL: https://lilytiltpong.herokuapp.com/

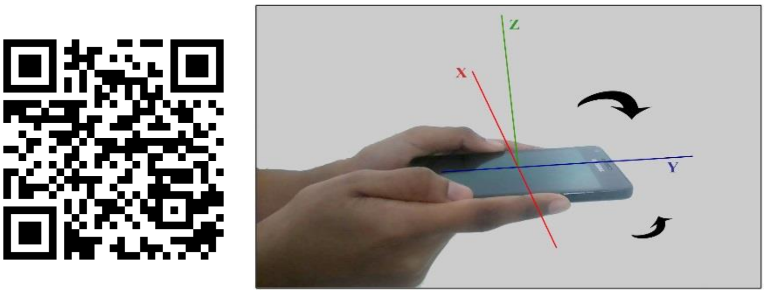


Figure 11. QR code and Rotation of Device in X-axis.

It should be noted that the game features might not load properly on some browsers as internet explorer or due to lack of orientation sensor in the phone.

3.2 Ping pong game implementation

In this sub chapter of the study, the implementation of ping pong game is described. First, a brief description of the logic behind the ping pong game and the implementation processes is presented. Followed by a presentation on how any mobile phones orientation data can be used as an input for controlling this game is presented. Finally, a detailed description of how the whole game is implemented is discussed.

To have a consistent terminology throughout the next sections, the following terms will be defined in accordance with this chapter's context.
Host – the device that the game is going to run on
Player – the mobile phone device that is used to control the game
Sever – the device that is going to administer the information flow between the player and **Host**.
Paddle – part of the game that is going to be controlled by the user
Game – Ping game that is chosen for the study

The implementation of the game begins by creating an express node.js server that will serve the application. All the programming occurs on the client side, in the web browser. The action performed by the mobile browser and the PC (large screen) are independent of each other. The two only communicate with the server, in this project the server communicating the two through "app.js" server file. When the user is opening the browser on the mobile the page running is the "mobile.html". When the browser on the PC is opened, the page that is running is the "host.html" page. However, I do not create these pages separately, though they communicate through the server. I create one "index. Html" file contains both files on the same page. This will help not to have two URL addresses for the mobile and the large screen.

A new game instance is created for every browser opened on the display screen and each mobile phone connected. Each instance has a unique id and name. In the server "App.js" file an array stores the instance with an id for the PC connection, and a unique string that will link the PC to the mobile. A unique socket is created whenever any device is connected to the server. Node.js and socket.io automatically create this socket. Whenever a user visits the game browser on the mobile or the browser on the PC we must notify the “App.js”.

All the programming handling the mobile phone is in one HTML file “mobile.html”. It is completely independent of the programming happing for the PC or large screen in another HTML file, “Index.html”. Communication occurs only through the server (app.js). While conducting the experiment on the case study, the players run the “mobile.html” page in their phone browser. The index page was running all the time on the large screen browser intended for carrying out the experiment.

All the necessary game codes will run on the Host. In this case, all the codes are written in JavaScript, therefore need only a browser to run JavaScript. Having the latest version of the web browser will help a lot, as the smoothness of the game depends on how the browser handles the java script. The game utilizes HTML5 canvas element to run all 2d graphics and animations needed. Using the mobile phone sensor (that senses orientation) and the information it provides to control the paddles on the ping pong game according to the movement of the phone. Any mobile phone that has a sensor (to detect orientation) and which can be accessed is appropriate. Therefore, the JavaScript which will run on the mobile device will first register the mobile to the game and access the data from the device orientations sensor, and send it to the host PC via the server.

- Setting up the server
- Creating a new instance over the socket
- Connection is added to an existing instance
- Pull the mobile device orientation data
- Use the data to update the ball in the HTML5 Canvas
- The connection will be handled accordingly

The implementation of the game involves the steps I listed above. I have not included the code needed to implement the game application, instead I have attached all the folders used in creating the game at on GitHub with the link;
https://github.com/strewb/ping-pong.

Server implementation

A simple server is set up which can handle data transmission back and forth from the Host to the mobile device via the server. For these reason, I have used a node as a server with modules soket.io and express. The main purpose of the server is to first it will create a channel for communication and then manages the channel. The first thing the server does is to identify the user if it is going to be a host or a player. If the connected user is a host, then the server will send the game together with the orientation data received form the player (if there are any player connected). If the connected user is a player, the sever will request orientation data from the player to send it to the host (see figure 12).

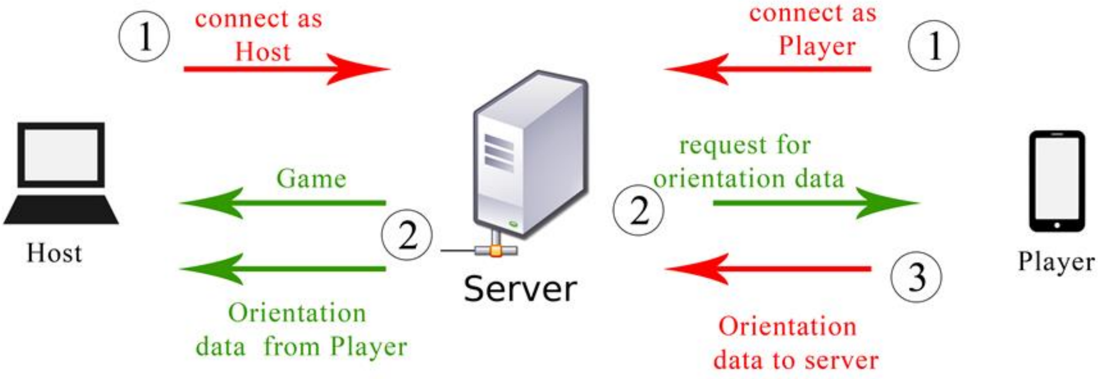

1
connect as Host
Host
Game
2
Orientation data from Player
Server
connect as Player
1
request for orientation data
2
3
Orientation data to server
Player

Figure 12. Server Client Communication

The server can only handle two users at a time, for instance, two different opponents can play on one game session. If extra players try to engage in the game, the server updates the latest data send by the new browser and override the old one.

The mobile phone will read orientation data and sends it to the sever using

```
  window.addEventListener('deviceorientation', function(event) {
      var alpha = event.alpha; // direction
      var beta = event.beta; // tilt front-back
      var gamma = event.gamma; // tilt left-right
      var data = { gameName: 'adam', playerName: playerName, alpha: alpha, beta: beta, gamma: gamma };
      socket.emit('orientation data', data);
   });
```

Then the server receives the orientation data and sends it to the game

```
   socket.on('orientation data', function(data) {
    var game = getOrCreateGame(data.gameName);
    if (game.hostSocket) game.hostSocket.emit('orientation data',
data);
  });
```

Then the game will receive orientation data and use it.

```
  socket.on('orientation data', function(data) {
      var player = players.filter(function(p) { return p.name ===
data.playerName; })[0];
      if  (!player)  return  console.log('Player  not  found',
data.playerName);
  ...//orientation data is processed here});
```

Using mobile orientation data

To look at how the mobile orientation data that is provided by the mobile device can be used as an input for controlling the movement of the paddles, first how the paddles are drawn is discussed below then a presentation on how this data can be used is presented. To draw the paddle on the canvas, ‘ctx.fillrect(x,y,w,h);’ is used. That is the width (w) and height (h)of the paddle, which are going to be fixed values, and the starting point from which these values are going to be measured (x and y) are needed. Here the values of x and y are dependent on the size of the canvas. Furthermore, the value of x depends on the position of the paddle (either left or right), and the value of y will be controlled by the user. The maximum and minimum values of y are 0 and the height of the canvas respectively. Therefor using a variable (variable named pos in this case) which can have a value from 0 to 1, it is possible to make y change according to this variable. That is, if pos is 0 for example then multiplying it with the height of the canvas the value for y can be found. Below is a code taken from the game that shows how it is implemented.

```
function setupPlayer(options) {
var player = {
          name: options.name,
          placement: options.placement,
          color: options.color,
          pos: 0.5, w: 10, h: 270 };
player.getPos = function(gs) {
      var pos; if (player.placement === 'left')   pos = { x:
      -player.offset, y: gs.y + gs.h * player.pos};
      if (player.placement === 'right')  pos = { x: gs.vp.w +
      player.offset, y: gs.y + gs.h * player.pos }
      if (pos.x + player.w > gs.x + gs.w) pos.x = gs.w -
      player.w + gs.x;
      if (pos.y + player.h > gs.y + gs.h) pos.y = gs.h -
      player.h + gs.y;
 return pos;
  };
  return player;
}
```

Then changing the mobile orientation data, which is represented with angel, to a value within 0 to 1 it is possible to represent the pos value of the paddle. To illustrate this a code from the game is given below.

```
var tilt = data.beta;      // orientation data from mobile
if (tilt < -45) tilt = -45;
if (tilt > 45) tilt = 45;
tilt += 45;
    // to simplify, only values from-45 to 45 are used and
    converted to 0 to 90 to avoid negative values.
player.pos = tilt / 90;
```

3.3 User experience experiment methods

User experience is a broad concept related to different definitions from ordinary usability factor to the feel and experience of the actual user interacting with the device or a system. As the aim of this paper is concerned with the experience that the user holds and interacts with, it is feasible to consider more than the usability features of the application. The author in [22.] characterized three ways to describe the experience between the interaction of the users with a product as; “**experience”, “an experience”** and **“co- experience”**.

“Experience” is how a user constantly assesses his or her goals relative to the products. “An experience” is characterized by an interaction with the product outlined with a certain character in user’s memory and a sense of completion, resulting a behavioural change during the interaction. “Co-experience” is defined as the experiences of the user, which could be influenced by the presence of others during the interaction. [23.] All these three experiences are significantly conferred in the context of the user experience evaluation in my study.

The methods for conducting an experiment requires users to perform practical tasks, in which the device or the application is designed to support in realistic situations. Importantly, by conducting user-based methods of the experiment it is feasible to conquer the limitations of expert-based simulation evaluations that can never quite replicate the exact nature of user behaviours and are too narrow to find out about what users experience [24.]

There are Different types of user-based methods. These include: scenario-based tasks, interviews, focus groups, and think-aloud sessions. Each of these methods could be used to contemplate different factors of user experience. [24.]

Research Goal and description of selected methods

The goal is to find out and analyse the user experience and the feasibility of using mobile device orientation in developing a browser-based game application.
The aim is to carry out a user experiences on:

- What the users feel when using mobile device as a controller (orientation-aware interaction technique)
- Is it easy and fun using a mobile tilt gesture?
- Is it socially acceptable to use this kind of gesture in a public place on a big screen?
- To find out the limitations occurring because of the game implementation approach.
- What is the user experience in using different interaction techniques, when using the tilt gesture versus the touch gesture on the mobile screen (keypad arrow) as the game controller input?

In this chapter, I discussed the methods used in the user experience experimentation. Methodology is a way to systematically solve the found problem. It consists of the various steps that are executed to investigate solutions for the problem at hand.

In the experiment, I want to capture data that will help me in understanding the different experiences the design provides for the users. The goal is to identify key factors that may affect the outcome of an activity and then observe the activities, inputs, limitations, resources, and output experience.

Hence, for analysing the user experience, I conducted an experiment, observation and a questioner followed by semi-structured interview. As the goal is to determine the cause and effect relationship between using the mobile phone tilt gesture in the game and the achieved satisfaction of the user using the technique. The selected experiment methods allow finding information on the three levels of experiences, namely, “experience”, “an experience” and “co-experience”.

Through the experiment, I wanted to see the user’s satisfaction and intuitiveness of using smartphone orientation as a controller, and the social aspect of using the different tilt gestures of the device in a public area.

The focus of the experiment was on the technique and the approach the game is implemented, not the game itself or its features as the game is only implemented as a prototype to find out the user experience.

Luckily while working on this thesis work I come across a computer ping pong game, “PongNub” on the web. The game uses touch-input gesture as an interaction method unlike the game implemented for the case study. It is a single player game were the computer act as an opponent, the player can use the “up and down” arrows on the mobile interface to move the paddles.

As The main purpose being to carry experiment using the game implemented for the case study, I let the users to try out “PongNub”, so that they can compare the two interaction techniques. I do not own this game, it is used for experiment purpose. This was done just to analyse which interaction method is intuitive, the comparison experiment was carried out to improve precision. The game can be played from the site: https://www.pubnub.com/developers/demos/pongnub/

Variable selection

Two different variables are defined in the experiment: independent variables and dependent variables. The Independent variables are the ones controlled in the experiment

[25.]. In this experiment, the two gesture controls are the independent variable: the device orientation as a controller and the touch-input gesture as a controller.

The dependent variables are the outputs gained from the independent variables [25.], in this experiment case the user experience and the limitations of the game implementation approach are the dependent variables.

Selection of users

The experiment was carried out in lobby area of Metropolia UAS Technology faculty. The first aim of the experiment was to conduct a session with anyone who is willing to try out the game and participate in the experiment. However, for analysis purpose and among the participants who were willing to fill out the questioners, six pairs of users were selected. The aim was also to find out how the crowd who will gather around during the experiment will feel and experience about the game.

Survey

A survey was chosen in order to investigate subjects' opinions of the smartphone used as a game controller. The survey was performed in the context of the experiment. Only the subjects who participate in the experiment could take part and fill in the survey (investigated all experience levels, see appendix 1 for the survey and how the questions relate to different levels of experience).

It was emphasized in [26.] that a questionnaire should be used when analyzing the user experience with games. Through the literature review in it was revealed that the questionnaire-based methods are common in the studies of user experience in games. For instance, in both the Game Experience Questionnaire and the Game Engagement Questionnaire form were used for evaluating the effect of mouse vs keyboard gamepad control in first person shooter ames (FPS). Similarly, the study in [26.] applied the Engagement Sampling Questionnaire to assess the player's desire to continue playing. In addition to the questionnaires, another evaluation tool was the “GameFlow model of player enjoyment” in video games proposed by P. Sweetser. This model was developed based on games user experience literature and Csikszentmihalyi’s concept of flow. It breaks player enjoyment into several parts such as “concentration, challenge, player skills, control, clear goals, feedback, and social interaction [26; 27.].”

All the participants filled in a questionnaire at the end of each game session. The questions are based on the research goal questions. The survey questionnaire on user experience was developed following the seven-staged survey research process from [28.] The questionnaire was tested through several iterations. Through the questionnaire testing question wording was improved, interpretation difficulties were removed, a format of some questions was changed to facilitate data analysis and interpretation. Analysis of the questionnaire (see Appendix 1-2), will be executed later in this chapter according to the three levels of experiences: experience, an experience, and co-experience.

The questionnaire consisted of six open and fifteen closed questions as shown in the appendix 1. Each of the participants filled out the questionnaire at the end of each game sessions. Basic information like gender and study field were not included in the questionnaire at the beginning, however, the answers were added manually after the interview. The first two questions enquired the type of phone and browser the participants will use as an input device to the game; the questions were designed as both quantities have a direct effect on the quality of game as well as the game's functionality. Three questions; 10-13 were constructed to determine how does the experience of the user in playing the game will be affected by the presence of others, measuring the "co-experience" level of the user. Questions 6-9 were created to gather the "experience" of the user when they interact with the game. Questions 14-18 were "an experience" related questions.

Interviews and observation

The author in [29] mentioned using only questionnaire with a predetermined set of questions, limits the user experience measure, especially the social aspect affecting the experience. The authors suggest conducting a structured interview afterwards in which subjective communication can be executed.

Hence, as a qualitative measure, each participant was individually interviewed after the game session has ended and they had filled out the questioner. Allowing the participants to narrate their experience of using the two different techniques of interaction. This interview included questions on participants' experience when using the chosen gesture in the public place and how they felt when playing in public space (co-experience). The procedure is to first let the users try out the game using games in both experiments. I took notes during the interview for purpose of analysis. The interviews were an average of 8 minutes long, with the shortest session lasting about 5 minutes.

As the experiment is conducted on a public area, it was possible to observe the participants playing the game and those who were watching the players. The observations were written down and used in analyses. The participants watching the gamers were not involved in the questioner survey or the controlled experiment.

The observation method was critical for analysing the game implementation method and limitations. As the participants, may or may not have knowledge on the game implementation approach, observation is the only data gathering technique about the limitations and future work. The method is also helpful to see the “experience” of the user in meeting the goal of the product (Game).

I had an observation moment with each pair of subjects for the whole experiment session, which is 10 minutes. To fill out the questioner took 5-6 minutes. In some cases, where the subjects were not grasping the game or how to use the device tilting orientation, the experiment was repeated.

At the end during the interview session, all the necessary experiences related questions were asked verbally. In addition to the written comments on the questionnaire (see question 21 in the appendix 1.), the interview questions were semi-structured which let the participants give suggestions and comments orally which led to further discussion. The interview question aim was to gain the emotional response and relationship they had with the game or any suggestion which is “**an experience**” of the user.

4 Process, data gathering and analysis

The experiment was carried out for finding out the intuitiveness of using mobile device as a game controller with different groups of people, and two different interaction techniques in a public area. The experiment consists of participants who have and have not played a device based controller game before the experiment. The participants consist of both female and male students.

- The experiment was conducted in Metropolia University of Applied Sciences (UAS) lobby area on 27th March 2017. All participants were allowed to play the game using their own smart phones.

- 4×4 LED screen was used as the game interface (playground)
- The client browse application was running on Windows 10 OS laptop
- Chrome cast was used to cast the browser game interface from the laptop to the LED screen
- Three different interaction techniques were tested
- Smart phones devices are used as a players' paddle to control the ball on the playground
- Two QR codes were generated for the URL address of the different games.

Three pairs of participants were recruited. Four of the participants are students from Metropolia University of Applied Sciences. The other one pair was from Laurea UAS, business administration students (seeTable 1).

- Three game sessions were experimented
- Three pairs of students were participated in the game
- All the players use their own mobile phone (phones with different models)

Table 1. pairs of participants information .

List of Paris	Participants Occupation
Pair A	IT (participant 1) and civil engineering (participant 2)
Pair B	Both IT students (participant 3 and 4)
Pair C	Both business administration student (participant 5 and 6)

Two Experiments were conducted in using the mobile phone device orientation as a game controller. All the participants participated in both experiments consecutively. The aim was to let participants play the game using their mobile device tilt orientation as a controller. As I mentioned in earlier chapter, two slightly different games were implemented for the purpose of the experiment. The participants used the QR code scanner to access the paddle controls from the browser on their mobile phone (see the QR code in chapter 2) or type the URL address of the game on their browsers. Once the game application is opened in the browser the interface has two options “Host” and “Player”. Either letting the user to use the phone as a playground (Host) or as a paddle controller, for this experiment purpose all the participants were told to use their phone as a controller (player).

Experiment 1:
The participants played the game, which uses only the “up” and “down” tilt motion of the device to control the paddles (the game was designed to use the beta value of the device orientation). The mobile screen of the participants must be in the portrait orientation to play the game.

Experiment 2:
The participants use only the “left” and “right” tilt motion of the device to control the paddles in the game (the game was designed to uses the gamma value of the device orientation). The mobile screen of the participants must be in the landscape orientation to play the game.

After the participants participate in the two experiments they were let to try out the “PongNub” game Application as a pair on the web (See Figure 13).

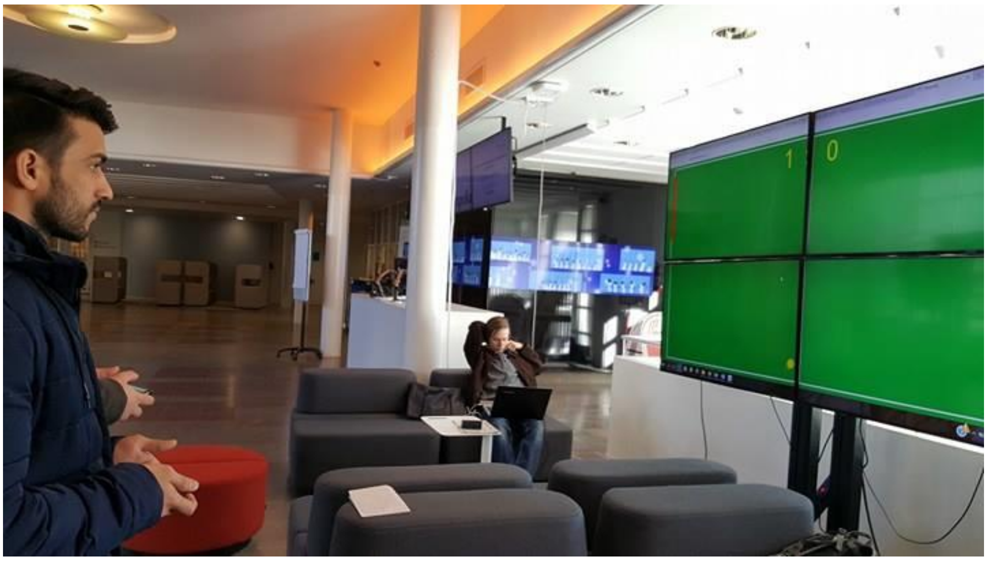


Figure 13. Pair of participants playing the game on LED screen during experiment 2.

4.1 Analysis and results

The **“experience”** of the participants in achieving the goal of the game, in terms of scoring a point and conquering the partner was greatly dependent on several factors such as the speed of the paddle, knowledge on how the game works, and the type of phone being used.

The use of different smart phones by the participants led to different outcomes in the functionality of the games in the experiment, one participant was unable to play the game using his Samsung J1 mobile phone. The participant was able to access the game interface as a "player" on the browser. However, the phone was not synchronized with the game playground on the big screen, and was not able to move the paddles.

Two participants ("**Pair C**") did not notice that they were using a browser-based application. I also noticed that by the time they asked me to elaborate question 3 on the questioner, the participants were less affine, not concerned about the technology underlying. Their attention was on the game. They did not mention a functionality issues related to the realization. They were appreciating "the quick movement of the paddle". The rest of the participants realized they were using browser-based application and web-based remote controls at a first glance.

I have observed, the ease of learning how the game works was fairly easy for all the participants. However, the users experienced a behavioural change and get a bit annoyed when their mobile screens go off during the game. This led one opponent to score more while the other player was not able to move his/her paddle due to the screen.

The observation method showed the orientation difference affects the experience of the user. Participants who were in favour of using the "up and down" tilt motion of their devices were quick in moving the paddle, similarly participants who were in favour of the other orientation," right to left" were able to score more points and move the paddle quickly. Two participants who chose two different orientation techniques as their preferences were playing as a pair in one game. This has affected the competitive nature of the game and the speed of the paddle movement, score of one participant over another. The result from the questioners also showed most of the participants preferred the idea of using the "up and down" tilt orientation of their mobile devices as a controller.

All the game sessions in experiment two remarkably exhibited a social aspect of the game at the public place, "**Co-experience"**. The first game session was held between two participants who had not known each other before the experiment. I observed during their first match, the competitive mode of the players was less, and the players were more focused on the game logic rather than winning. These same participants repeat the second game using "up and down" orientation of the device, this time I observed they had a good understanding with the game.

In second pair of participants in the experiment were friends, and they come with their fellow class-mates as a crowd. This session exhibited a real-life taunt technique with one player attracting the other opponents' attention all game long, allowing his ally to take over half the paddle movement was almost undetected. Thus, the game ended with a competitive mode and influence from the crowd. I noticed both participants were not conscious about their surrounding and only focused on the game.

The third pair of participants were not students at the institutes. These participants had a background in business, one participant was reluctant at first to try out the game, but she made up her mind to try out the game. The result from the observation method showed, it was difficult for these players to play the game for a longer session in the public area, they were not in the competitive mode during their first trial. Since the game was also a new experience for them they were in the training phase. However, they were tempted to try for second trial, were more focused on opponents conquering one another. Competition was apparent, session found the winning player in a very advantageous position with the losing side covertly blaming each other for the loss. The game setting environment was not fully comfortable for them as they were distracted by the people who passed-by the game area and giving comments.

Questionnaire

The survey questioner consists of question addressing the three levels of experience. Through Questions 6-9 "**experience**"; objective data about the learnability, functionality, adaptability of the interaction technique was collected.

As indicated in chapter 2, the type of browser used to run the application should support motion event API and device orientation. Considering this fact, type of browser being used by all the participants, name and model of the mobile phones were collected from the questioner and summarized as shown in table 3 below. Moreover, to identify the cause of the error in synchronization of the game with the phone, data about the type of sensor the phones have, was investigated and included in the table. The data collected from the first questions in the questioner indicate from all the participants, 83.3% of them were using Android phones, the rest 16.7% used iPhone to play the game during the experiment.

Table 2. Type of phone, browser, and built-in orientation sensor

Mobile phones used in the experiment	**Browsers used on the phone**	**Orientation sensor**
Samsung galaxy s3(2)	chrome	Yes
One plus3	Internet explorer	Yes
Samsung galaxy s6	Samsung browser (Default browser)	Yes
IPhone 7 plus	Safari	Yes
HTC	chrome	Yes
Sony Xperia	Chrome	Yes
Samsung J1	chrome	No

As figure 14 below Illustrates, four out of the six participants choose the “right to left” Tilt gesture as their preference to handle the phone for controlling the paddle while the other two prefer the game with “up and down” tilt option. The result clearly shows the importance of creating two games with different tilt gesture orientation was on point. Four of the participant (66.7 %) who preferred “right to left” tilt elaborate in the interview, even in doing other tasks such as recording videos using their mobile phone cameras they prefer holding the phone with two hands and the landscape orientation. For the rest, it is innate and convenient to use their devices in landscape orientation for this application.

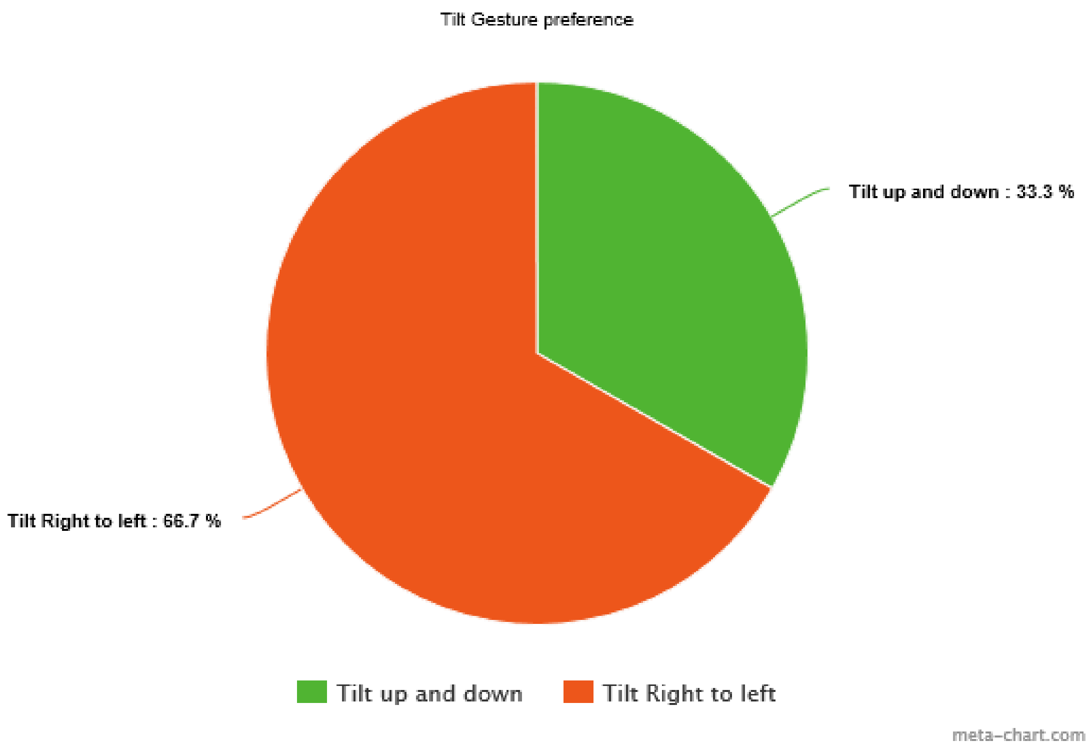


Figure 14. Rating of tilt gesture preference

Out of all the participants, only one participant mentioned he has never played a ping pong game. How the game proceeds, so he was trained how the game works before he started to play with the partner. This had a great impact on how quick the participant was able to successfully collaborate with the partner. He lost the game with 3 scores and was able to succeed more while playing the game for the second time.

The same participant also indicated it was difficult to engage with the game at first (see Question 6 in appendix 1), while the rest mentioned it was easy for them to play the game at first glance. Some participants had an experience with using an interactive gesture based application in large public screen, they have mentioned in the interview using interactive screens at the mall which uses touch input as an interaction method.

The movement of the paddle was easy for 50% of the participants, and the rest half indicate “more or less” as an answer to the questioner (see question 5, In appendix 1 “Have you ever used an interactive Gesture based application in large public screen”). This helped the participant to understand how to play the game and the type of phone being used.

Data gathered from questions 10, 11, 12, 14 on the questioner clearly shows the level of fun and intuitiveness of gesture based interaction method has a great value over other interaction methods. Only one (16.6 %) of the participants chose both “wireless gesture based control” and “touch based” controller as an interaction method, 83.3% chose wireless gesture as their preference. The participant mentioned in the interview: “the type of interaction method depends on the functionality of the application, playing the game using the touch input I was able to control the speed of the paddle more easily rather than the tilt motion but also the tilt motion gesture feels more innate in playing the game in real life”. Moreover, comparing the data from the two experiments, using the tilt motion of the device over the touch input method used by the “PongNub” game in the second experiment showed a positive feedback from all over the participants.

“An experience” of the participants were majorly collected from questions 14-18 on the survey questioner. Few had given a short comments and feedback at the end of the questioner. These responses reflect the participants’ emotional response with the interaction, behavioural changes, comments, and personal feeling.

Below are some participant comments: “Playing a tennis game with this approach would be more intuitive and fun, if a physical hand gesture to throw the paddles in air could be incorporated instead of tilting the mobile device”. (Participant 4, IT student)

“I would say using a mobile device as a controller is a very good approach especially in public places or social gatherings. Since almost everybody has a smart phone, it seems natural to use it as a controller. It would also be cheap as we only need a screen to implement it at any place.” (Participant 2, civil engineering student)

“In the middle of playing the game, the mobile phone screen shut down and the game is interrupted and it’s a bit annoying.” (Participant 3, IT student)

Only one out of the six participants had used an interactive gesture based application on large public screens, for all the rest it was a new experience. However, they were familiar with playing an interactive gesture based games on computers and TV screen using game consoles as PS, HTC vive, XBOX, Joysticks, Nintendo Wii. Their overall feedback to the idea of playing the game on public place in a big screen was positive one and new experience. All have mentioned they would like to try other applications with the same interaction technique, which could be useful in their daily life. Except one participant who preferred to use both gesture and touch based controller, all the participants chose to use a “wireless gesture based control” as their first preference to interact with the applications in a public screen. Moreover, all the participants agreed upon using mobile device rather than buying a console is more convenient and cheap.

The social context and how the participants feel in playing the game in a public setting were addressed through Questions 10-13 of the questioner **experience”.** Out of the three pairs of participants, four mentioned it is fun to use mobile phone tilt gesture as a controller in public place, two says it is normal. The results show though it is intuitive and fun to use mobile phone as an interaction means, using it one the public large screens resembles normal to 33.3 % of the participants.

Interview

Through the interview method subjective data about personal feeling, comment and opinion, emotional response “**an experience**” about the interaction were collected. Many of the participants have stated that although they have had previous experiences with motion sensing input devices as a game consoles, using their mobile phone as a game controller was a new experience.

I directly quoted the additional suggestions and verbal comments the participants gave during the interview for improvement issues. ”If HTC controller (virtual reality headset) could be attached in the game it would be more fun, but I think these kinds of controllers are expensive”. (Participant 1, IT student)

“While playing the game using the “right” and” Left” tilt motion of the device I had to use two hands to hold the phone and control the paddles, and I don’t like to use two hands for controller, I prefer the first game where I could use only one hand.” (Participant 1, IT student)

Participants also give a positive feedback about the advantage of playing the game the using the mobile phone as a playground (host), as well as a controller or even hosting the game on PCs. (participant 1 and 2)

“To play the ping pong game using the touch arrow of the device was less innate compared to moving the paddle using the tilt motion of the device; it feels like I am moving the paddles in real life, but if a motion gesture could be added it would have been more fun." (Participant 4, IT student)

One user mentioned that she does not like to load a QR code. "It is better to use the mobile phone instantly in the same Wii controller is used." (Participant 5, business administration student)

When the participants were asked their perception on, if this kind of game is good for social events, two replied: "Maybe" in their answers. The rest believed it is a good game for social events. Two participants mentioned that using the touch-input gesture on their mobile screen as "PUBNUB" application does not change the way the device is held or used. For the most part, the use does not conflict with the existing perceptions of mobile phone usage in social settings.

All the participants were somehow pleased with the interface of the game. However, they have left an improvement and suggested issues during the interview session such as:

- Improving the speed of the paddle;
- If the player could determine the colour of the ball and paddle instead of the application;
- Sound and some spark features could be included as the ball collides with the paddle and point is scored;
- To include a touch button on the playground screen (the host screen) to initiate the game.

One participant has mentioned in the interview about the collaboration aspects of the game. The participants' feedback was supportive of having more than one person to move the paddle and play the game on the screen: "If the game enables the players' to have competition with a separate screen and we could be able to compare outcome in real-time"; will make the game more competitive and inspiring to play as a group. (Participant 1, IT student)

4.2 Discussion

All the three data gathering methods give a high amount of data. The participants' objective perspective about learnability, adaptability (ease) of the interaction technique, the current **experience** they have with the game was collected through the survey questions and observation methods.

One participant’s perception with the functionality of the game was less satisfactory in relation with the type of mobile device used. His first attempt to the game was hindered due to the sensor built-in the mobile device. Unless the participant is more affinity to the technology behind, the cause resembles more to the functionality issue of the game.

The specification details about the type of sensors built-in inside all the mobile phones used in the experiment has been researched. Most of the android phones except Samsung galaxy J1, have around ten sensor types including “orientation sensor”, which was necessary for the game application to use the phone as a controller. As specified in [30.] Samsung galaxy J1, lacks orientation sensor and have only proximity and accelerometer sensors which were not enough for the phone to interpret the movement of the paddle in synchronization with the orientation of the phone on the big screen. This has shown the powerfulness of built –in sensors such as accelerometer and orientation sensors in mobile devices in detecting device orientation as discussed in the theoretical background section of the thesis.

Not only the sensor but also the version of the device determines and affect the speed of the paddle. The movement of the paddle was quick with latest mobile phones. The communication between the phone’s orientation values through the server to the Host was quick with latest mobile phones, which resulted in quick movement of the paddles. Improving the speed of the paddle was also a suggested idea from some participants. This has affected the **experience** of the users’ in scoring a point due to the type of phone being used by them or the opponent. The issues above reflected that the game could be played well with latest mobile phones and only those having orientation sensor.

The ease of learning how the game works at first was easy **experience** for all the participants. However, the idea of prompting the game in a real setting without a guide was vague. A method to start the game and see the playground should be incorporated in Host Interface. During the experiment, I initiated the game for the players from the computer were the game was casted by pressing the “start “button on the Host screen. To start the game for himself, participant was interested in having a touch input on the Host screen. In addition, a description for the game could also be incorporated on the screen.

In this manner, users can interact with the game without guide and training from extra person.

Most of the improvement and additional suggestion on the features of the game were collected from the interview and comment section on the questioner. The participants' **an experience** with the game and interaction technique inspires additional features to be incorporated in the game.

The participants get high level of satisfaction from the interaction technique, they expressed their experience were more similar in playing ping pong in real world when using the tilt gesture of the device. However, they were interested in having more visual effects on the interface, which could increase their satisfaction and making the game more appealing. The list of suggestions includes sound effect for scoring, choosing colour of the ball and manipulating the speed of the paddle. Since the aim of the study was on the creating the game with a least functionality for user experience evaluation, the game features were not highlighted.

Implementing two games based on two different orientations in the case study was beneficial in determining how users prefer to hold their phone. Most of the users prefer the paddle to be controlled by tilting the device in "right" to "left" orientation. Using two hands to hold the phone for the purpose of the application was convenient one. For few holding the device using one hand is an innate way of playing ping pong game in a real world.

As the game is a browser based application, the host (playground) browser was always ON during the entire experiment on a personal computer and casted to the LED screen. However, this was only done for the experiment purpose and casting method is less viable as a permanent means to display the application on the LED screen, as the casting could be interrupted, and someone must always be on the computer to monitor and cast the game whenever the monitor mode is sleep. Implementing the game using another technology, so that the game is hosted on large digital display without someone opening the browser on the large screen, is advised. The issue of mobile screen goes to sleep mode during the game was annoying for some participants.

The game interaction technique gives the participants an opportunity to play without installing a native mobile application unlike most other games. However, the data connectivity issue was a concern for one participant, he preferred a native application, which could be download and work offline. Though it is a convenient approach to most applications, for applications using remote control such as the game in the case study, the argument is invalid. A means of wireless connection is a must to exchange data between the mobile device and the computer or LED screen. However, considering the remote connection of the smartphone with the public screen through a web browser was an advantage, the fact that there was no need to an install a native application on the device was less time consuming and saved the storage memory of the device. In addition, the participants are interested in fastest and easiest possible way to open the game on their mobile devices, which seems favourable for a native application in enabling the user without typing the URL address of the application on the browser. The alternative solution to solve the issue was to generate QR code.

"Co- experience "of the participants were mainly gathered through the observation. Determining the social aspect, **Co experience** of the user with the game greatly depend on the environment where the experiment was held. The game was merely fun and initiative one for those participants who were friends. For some, the participation was dependent to other people in the environment as the game involves tilting hand gesture which could draw others attention. They felt the competition mood was also subjective for others judgement due to the setting. Moreover, placing the game in a public setting was an issue for some participants as need to wait for a partner to join the game.

Even though gesture and orientation based mobile interfaces provide an intuitive interaction technique, the interfaces design require to consider the users behaviour towards disruptiveness or embarrassment in a certain location. For social events or in an environment where players know each other well the game resembles to be fun. The output of the experiment detected social acceptability as a critical consideration. As the experiment was conducted in a school compound, the collected data was not enough to conclude about the experience on social behaviour in a more public place. My suggestion for future work is to conduct social experiments in more public areas such as malls and big social events. This will give a more concise result.

5 Conclusion

The goal of the thesis was to conduct an experiment on using tilt-gesture of a mobile phone as a game controller on a large screen and study the users' experience with the interaction method. The background study about device orientation helped to implement the game successfully. The game proved that the built-in features of smart phones such as sensors and accelerometer in combination with a good server and internet communication can turn a mobile phone into a wireless gesture-based controller.

The survey and interview methods were the most powerful tool in collecting the user experience at the "experience "and "an experience" level whereas the observation method was crucial in getting the user experience in social context (co-experience). The result of the two experiments conducted during the thesis work indicated that to use a mobile phone as a control device was more favourable than using wired game consoles and key arrows on a computer.

The user experiences indicate that using the tilt gesture of the phone as a controller was a more innate interaction than using a touch input as a controller. The use of mobile devices as a console was a good alternative for the users who are resistant to game consoles to try and enjoy gaming with a tool they are familiar with in their daily lives. Implementing a collaborative application with the interaction technique made the application more enjoyable and fun. However, the social acceptability of using such applications in a public place is disruptive for some users.

Though the main aim of this study was not to develop a game, I was happy about the way the prototype game turned out, and the welcome it received from the participants who played it was heart-warming. Although controlling paddle motion by tilting the device initially seemed simple enough for anyone to understand, during the experiment I realized it was not always the case. I needed to explain how paddle motion works to some of the players. If I ever had to deploy the game somewhere again, I would consider adding a description on the host interface to make the game self-explanatory and the paddle to be controlled only by simply moving the hand vertically up and down.

The idea of implementing the paddles to move in directions was confusing when creating an application with socket.io. Using device orientation as a controller depends on the communication technology whether it is a web socket or real-time. This would be a crucial thing to consider for anyone who would like to implement such games and interactive applications using socket.io or other communication technologies.

References

1 Pete LePage. This End Up: Using Device Orientation [online].April 2011. URL: https://www.html5rocks.com/en/tutorials/device/orientation/. Accesed on 3 March 2017.

2 Vajk T., Coulton P., Bamford W., Edwards R. Using a mobile phone as a ‘Wii-like’ controller for playing games on a large public display[serial online]. International Journal of Computer Games Technology, vol. 2008, p. 1–6, January 2008. URL: https://www.hindawi.com/journals/ijcgt/2008/539078/. Accessed 20 March 2017.

3 Ballagas R, Borchers J, Rohs M, Sheridan J. The Smart Phone: A Ubiquitous Input Device [serial online] 2006; 5(1):70-77. URL: https://hci.rwth-aachen.de/materials/publications/ballagas2005d.pdf. Accessed 20 March 2017.

4 Baldauf, Matthias and Peter Fröhlich. Investigating Serendipitous Smartphone Interaction with Public Displays. In: Joanna Lumsden, editor. Emerging Perspectives on the Design, Use, and Evaluation of Mobile and Handheld Devices. School of Engineering and Applied Science, Aston University, UK; 2015. p. 239-268.

5 Jérôme Van Zaen, Jody Hausmann, Kevin Salvi. Gesture recognition for interest detection in mobile applications [online]. IEEE Pervasive Computing. 29 September 2014. URL: http://ieeexplore.ieee.org/document/6911163/. Accessed 20 March 2017.

6 Raimund Dachselt, Robert Buchholz. Throw and Tilt – Seamless Interaction across Devices Using Mobile Phone Gestures [online]. Department of Simulation and Graphics, Faculty of Computer Science Magdeburg, Germany; 2009. URL: https://imld.de/cnt/uploads/2008-MEIS-Throw-and-Tilt.pdf. Accessed 20 March 2017.

7 Valente, Luis, Souza, Clarisse Sieckenius de, & Feijó, Bruno. Turn off the graphics: designing non-visual interfaces for mobile phone games. Journal of the Brazilian Computer Society 2009; 15(1): 45-58.

8 Scheible, Jürgen. MobiToss: A Novel Gesture Based Interface for Creating and Sharing Mobile Multimedia Art on Large Public Displays 2008 [online] 2008:957-960. URL:http://www.mediateam.oulu.fi/publications/pdf/1131.pdf. Accessed 21 March 2017.

9 Rusk D. Mobile Device 3D Accelerometer-Based Gesture Recognition - Thorn Technologies [online]. Thorn Technologies; 17 July 2013.

URL:https://www.thorntech.com/2013/07/mobile-device-3d-accelerometer-based-gesture-recognition/. Accessed 20 March 2017.

10 Jonathan Reid. HTML5 Programmer's Reference. 1st ed. New York: Apress; 2015.

11 Penny de Byl. Holistic game development with unity: An all-in-one guide to implementing game mechanics, art, design and programming. Massachusetts: Focal Press; 2012. P.423–477.

12 Aurelio De Rosa. An Introduction to the Device Orientation API [online]. Envatotuts+; May 2014.
URL: https://code.tutsplus.com/tutorials/an-introduction-to-the-device-orientation-api--cms-21067. Accessed 3 March 2017.

13 Cthorner. Detecting Device Orientation [online]. Mozilla Developer Network; Sep 29, 2017.
URL: https://developer.mozilla.org/en-US/docs/Web/API/Detecting_device_orientation. Accessed 1 October 2017.

14 Dphenderson,Jeremie,Sheppy,kohei.yoshino. Orientation and Motion Data explained [online]. Mozilla Developer Network; Feb 29, 2016.
URL:https://developer.mozilla.org/en-US/docs/Web/Guide/Events/Orientation_and_motion_data_explained. Accessed 3 March 2017.

15 Ruadhán O'Donoghue. HTML5 for the Mobile Web: Device Orientation Events. [online]Mobiforge; July 10, 2013
URL:https://mobiforge.com/design-development/html5-mobile-web-device- orientation-events. Accessed 3 March 2017.

16 Ahmet Mermerkaya.Using Device Orientation in HTML5 [online]. SitePoint; May 2, 2014.
URL: https://www.sitepoint.com/using-device-orientation-html5/. Accessed 20 March 2017.

17 Ruadhán O'Donoghue. HTML5 for the Mobile Web: Device Orientation Events. [Online]Mobiforge; July 10, 2013
URL:https://mobiforge.com/design-development/ html5-mobile-web-device- orientation-events. Accessed 3 March 2017.

18 Wang X., Tarrío P., Metola E., Bernardos A.M., Casar J.R.Gesture Recognition Using Mobile Phone's Inertial Sensors. In: Omatu S, De Paz Santana J, González S, Molina, Bernardos A, Rodríguez J. editors. Distributed Computing and Artificial Intelligence. Advances in Intelligent and Soft Computing, vol 151. Springer, Berlin, Heidelberg; 2012.

19 Eslambolchilar, P. and Murray-Smith, R. Tilt-based automatic zooming and scaling in mobile devices – A state-space implementation. Mobile Human-Computer Interaction – MobileHCI 2004: 6th International Symposium, Glasgow, UK; September 13 - 16, 2004. 3160:p. 120-131.

20 Pong Game. About Pong [online]. Ponggame.org; 2011-2017.
URL: http://www.ponggame.org. Accessed 20 March 2017.

21 Sheppy. Canvas [online]. Mozilla Developers Network; 24 March 2012.URL: http://mdn.beonex.com/en/HTML/Canvas.html. Accessed 20 March 2017

22 Chopra v. Web Socket Essentials – Building Apps with HTML5 Web Sockets. 1st ed. Birmingham, UK: Packt Publishing; 2015. P.381-392.

23 Jodi Forlizzi, Katja Battarbee. Understanding experience in interactive systems [online].Carnegie Mellon University; 2004.
URL: http://repository.cmu.edu/cgi/viewcontent.cgi?article=1045&context=hcii. Accessed 20 March 2017.

24 Alshehri, Freeman. User Experience of Mobile Devices: A Three Layer Method of Evaluation [online]. Auckland, New Zealand; 8th -10th Dec 2014.
URL:http://aut.researchgateway.ac.nz/bitstream/handle/10292/8029/acis20140_submission_230.pdf?isAllowed=y&sequence=1. Accessed 21 March 2017.

25 Wohlin C., Runeson P., Höst M., Ohlsson M.C., Regnell B., Wesslén A. Experimentation in Software Engineering: An Introduction. Kluwer Academic Publishers, 2012.

26 Schoenau-Fog H., Birke A., Reng L.: Evaluation of continuation desire as an iterative game development method. 16th International Academic MindTrek Conference, Tampere, Finland, October 3-5 2012, p. 241-243.

27 Sweetser P, Wyeth P. GameFlow: A Model for Evaluating Player Enjoyment in Games. ACM Computers in Entertainment 2005, 3(3): 3.

28 Kasunic M. Designing an Effective Survey [online]. Carnegie Mellon Software Engineering Institute; 2005.
URL: www.sei.cmu.edu/pub/documents/05.reports/pdf/05hb004.pdf. Accessed 21 March 2017.

29 Marcus A. Design, User Experience, and Usability. Theory, Methods, Tools and Practice. Berlin: Springer; 2011.

30 Device Specification .Galaxy J1 Samsumg specification [Online].DeviceSpecification; 2017.
URL: https://www.devicespecifications.com/en/model/42ca3243. Accessed 7 March 2017.

User experience evaluation in using mobile phone orientation as a game controller: Questioners

Questions 6-9 measure “experience”, 10-13 measures “Co experience”, and 14-18 measures “an experience” level of the users.

1. Name and model of the smart phone you use ……………………………………….
2. Which browser do you use on your phone? ....
3. Have you ever played browser based game? ………………………………………..
4. Have you ever played a ping pong game? Yes □ No □
5. Have you ever used an interactive Gesture based application in large public screen?

   Yes □ No □
6. Was it easy to engage with the game at first or need training? ………………………...
7. Was it easy to move the paddles up and down and hit the ball?

   Yes □ More or less □ Not at all □
8. Were you able to properly collaborate with your partner

   Yes □ Somehow □ Not at all □
9. Was it easy to play the game?

   Yes □ No □ May be □
10. Was using the mobile phone as a controller in public place

    Embarrassing □ Fun □ Normal □

11. Would you play other games with this approach in public screens

    Yes □ No □ May be □

12. What kind of interaction technique do u prefer to interact with applications in a large screen in a public place?

Wireless gesture based control □ Touch interaction □

Console based game □

13. Do you think this is a good game for social events?

Yes □ No □ May be □

14. Was using a mobile device orientation as controller easier than

Keyboard arrows □ Touch gesture on mobile phone □ Same □

No opinion □

14. What kind of game consoles have you used to play an interactive game on PC screens?

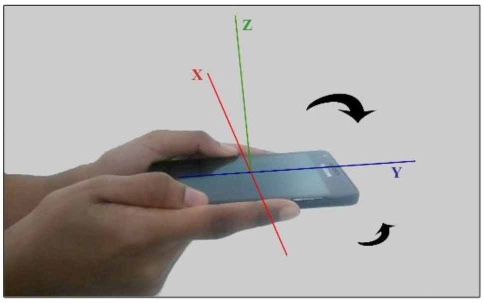
………………………………………..

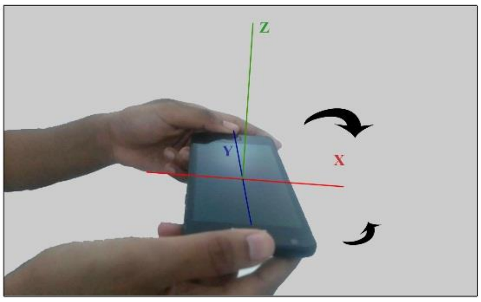
16. Is it cheap and convenient to use mobile phone as a controller rather than buying a console?

Yes □ No □ May be □

17. Which tilt gesture do you prefer, to tilt the phone

Up and down □ Right to left □





18. Any comment or suggestion

……………………………………………………………………