Contents

[**Acknowledgements** 1](#_Toc94128895)

[**Glossary** 1](#_Toc94128896)

[**List of Figures** 2](#_Toc94128897)

[**Introduction** 2](#_Toc94128898)

[1.1. Context and motivation 2](#_Toc94128899)

[1.2. Objective 2](#_Toc94128900)

[1.3. Thesis Structure 2](#_Toc94128901)

[**Materials and Methods** 2](#_Toc94128902)

[2.1. Functional requirements 2](#_Toc94128903)

[2.2. System analysis and design 2](#_Toc94128904)

[2.3. Implementation 2](#_Toc94128905)

[**Result and Discussion** 3](#_Toc94128906)

[3.1. Result 3](#_Toc94128907)

[3.2. Discussion 3](#_Toc94128908)

[3.3. User Interface 3](#_Toc94128909)

[**Conclusion and Future developments** 3](#_Toc94128910)

[4.1. Conclusion 3](#_Toc94128911)

[4.2. Development 3](#_Toc94128912)

[**References** 3](#_Toc94128913)

# **Acknowledgements**

# **Glossary**

# **List of Figures**

# **Introduction**

## **Context and motivation**

## **Objective**

## **Thesis Structure**

# **Materials and Methods**

## **Functional requirements**

## **System analysis and design**

* 1. **Use Case Diagram**
     1. **A**
        1. **A**
     2. **B**
        1. **B**
  2. **Sequence Diagram**
     1. **A**
        1. **A**
     2. **B**
        1. **B**
  3. **Class Diagram**
     1. **A**
        1. **A**
     2. **B**
        1. **B**
  4. **System Architecture**
     1. **A**
        1. **A**
     2. **B**
        1. **B**

## **Implementation**

* 1. **Installing environment?? – Break down steps**
     1. **(input every installation here)**
     2. **Node js**
     3. **Websocket**
     4. **Please don’t delete this line to keep the layout**
  2. **Index page**
     1. (**contains information for index page)**
     2. **Identify the need for data**
     3. **Data resource**
     4. **Data process: Edit and convert data**
     5. **Please don’t delete this line to keep the layout**
  3. **Waiting room**
     1. **Break Down Steps**
     2. **Server Side**
        1. Host a local Server with NodeJS and Express
        2. Embedded a map with Google Maps JavaScript API
        3. Geocoding Service
        4. Google Cloud Storage
     3. **Please don’t delete this line to keep the layout**
  4. **Gaming room**
     1. **Server side**
        1. **Host a local server with NodeJS and Websocket**
           1. Before we go further about hosting a local Sever, we would like to briefly define a few important terms and concepts:
* **Node.js:** is an open-source, cross-platform runtime environment that allows developer create server-side and networking applications. Node.js applications are written in Javascript and provides a rich library of various Javascript modules which simplifies the development of web applications using Node.js.
* **Npm (Node Package Manager):** is an online repository for the publishing of open-source Node.js projects and a command line utility for installing packages, managing versions and dependencies.
* **Websocket:** WebSocket is a computer communications protocol, providing full-duplex communication channels over a single TCP connection. The WebSocket protocol was standardized by the IETF as RFC 6455 in 2011, and the WebSocket API in Web IDL is being standardized by the W3C. WebSocket is distinct from HTTP.
* **Server:** waits for client request messages, processes them when they arrive, and responds to the web browser with an HTTP response message (e.g. “HTTP/1.1 200 OK” for success, “HTTP/1.1 401 Unauthorized”, .etc). A response can be many things, but the most common response from and API is JSON (JavaScript Object Notation) file.
* **Client:** is everything in a web application that is displayed or takes place on the end user device. It also sends requests (like GET, POST, DELETE, UPDATE, .etc) to the server.
* **JSON:** JavaScript Object Notation is a standard way of formatting data using syntax from JavaScript.
* **API (Application Programming Interface):** Code that allows a client to interact with a server.

As the definitions shown above, here are the method that we use to host a local server:

* + - 1. **Create a Websocket server**
    1. **Client side**

# **Result and Discussion**

## **Result**

## **Discussion**

## **User Interface**

# **Conclusion and Future developments**

## **Conclusion**

## **Development**

# **References**