**Project Log book Entry**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Student Name:** | Rhokmot Ali |  | **Supervisor:** |  |
| **Student No:** | 1302260 |  | **Date:** | 21/11/17 |

|  |
| --- |
| Tasks done since last meeting |
| 17/10/17 – 5 days: (task) Find out how to get on the university server = <https://mi-linux.wlv.ac.uk/wiki/index.php/Main_Page> and also see if node js works on our server which is does by typing in node –version 1302260@csl-student:~$ node --version  v6.12.3 as you can see the version of it. write some report for the project proposal. (Management/Deliverable) Research done on the mi-Linux site, also the use of terminal servers, form of academic question and research into technologies needed.  21/10/17- 10 days: (task) Report started looking into source for project. (Management) Api Research, libraries, sha and cryptography, front end and back end.  1/11/17- 9 days: Completed report before submission, format in correct manor. (Management) put content page, formatting font size and references page in order.  21/11/17- 17 days: Learning and testing small samples of Node.js, All back-end stuff server side: https://www.youtube.com/watch?v=U8XF6AFGqlc     1. Local server is setup using node js – local server is the local server on the laptop localhost:8080 running a node script using terminal, all is needed is to install node https://nodejs.org/en/download/ - then enter node in terminal 2. I have used Express JS to handle requests from front end – express is a further framework for node js It is minimal and robust and provides a very simple api for developing web applications quickly. It greatly improves the speed of development. Simple library code need to access node var express = require (‘express’). https://www.youtube.com/watch?v=gnsO8-xJ8rs 3. Basic UI is setup to test whether the response getting from back end is fine or not. Doing something simple as a hello world, res.send(‘Hello World)’ https://www.youtube.com/watch?v=gnsO8-xJ8rs   01/12/2017 – 14 days: Complete – professionalism report and slides and through explanations on  16/12/17 – 18 days: Application is running and map and marker working, the api is running through google maps, have not check through mobile yet as the server needs to be worked on. Challenge was getting api key and I found on google maps api web page https://developers.google.com/maps/documentation/javascript/get-api-key, I didn’t know how to access it and how to put in on my html index. page but I found on YouTube with tutorials <https://www.youtube.com/watch?v=Zxf1mnP5zcw>  21/12/17 - Map uses geo location api to track the user’s locations: read many blogs how to access such as https://developers.google.com/maps/documentation/javascript/geolocation, how to user api, learn that I have to use a developer account to get the api keys,  When tracking it uses only one location, challenge was implementing the dynamic location of the user using the website. Found by this link = https://jsfiddle.net/api/post/library/pure/  03/1/18 – I will be using mongodb as the database for my web application, database schemas designed for this web application are location schema, user schema and comment schema.  07/1/18 – Researched on how to develop mobile apps in a professional manner, turned to Ionic framework and read through the details of using this JavaScript on <https://www.tutorialspoint.com/ionic/index.htm> and implemented into my file. Also looked at ionics documentation https://ionicframework.com/docs/  14/1/18 – Have wrote the code for ionic: implemented the rating and comments features for the current location of the user  18/1/18 – Node js is able to make connection with mongodb using mongoose package = https://www.youtube.com/watch?v=93APwwh35PA&t=33s  27/1/18 – Register page is complete, user can register him or herself on the web application and the unique usernames and passwords are stored in user schema of mongodb. I went through many registers pages on websites, the general are login and passwords. There are three important are password and confirmed password.  5/2/18 – For the data security reasons, I learned how to use BCrypt library for hashing the passwords, now the hashed passwords are stored in user schema instead of plain text passwords.  11/2/18 – Implemented the login page and now the register user can login into the web application, also I am using toasters for notification purpose in case the user is entering the wrong password or trying to register the already existing username.  20/2/18 – User can click on any location and give it rating from one to five. The users rating is stored in location schema and as soon as user clicks on any location the info window of the location shows the average rating along with the location address.  28/2/18 – User can click on any location and put comments for it. The comments will be stored in comments schema.  4/3/18 – Implemented a “comments” page, now as soon as user clicks on any location he is given option to view the comments. If user chooses to view the comments he/she is redirected to comments page, where he/she can see the comments given by other users for that location.  4/4/18 – Working on the dissertation and the artefact level 3  20/4/18 – Fixed the format table of contents properly and picked some content sections  27/4/18 – Looked at the Stemming for artefact level 3 - <https://www.codeschool.com/blog/2016/03/25/machine-learning-working-with-stop-words-stemming-and-spam/>  3/5/18 - Reading about and working on captcha on the login page, captcha is used to prevent robots from hackers – <https://www.npmjs.com/package/express-recaptcha>, https://www.youtube.com/watch?v=UzCkSzmEq8E  7/5/18 - Advance search - location search using google API - https://developers.google.com/places/web-service/search |
| Tasks to do before next meeting |
| Task to complete: Map and marker api should run through google map also mobile  Table of contents page with information, ask Jun li about mongodb datebase  Write some literature review, fix contents page |