Project Information

Site:	Monash Moodle	Printed by:	Jin Chun Tan
Unit:	ENG1003 - Engineering mobile apps - S1 2021	Date:	Saturday, 17 December 2022, 5:51 PM
Book:	Project Information		

Table of contents

1. Summary

Pristine Cabs is a Taxi company that requires an app for their business. The company has decided to expand their business and require an application to centralise taxis' bookings, which is the preliminary expansion requirement. As the business will grow, the company will want more features added to the app.

Upon advertisement, Monash University found a number of students interested in developing the app. The university has come up with the idea of making teams of students to let all of them have the opportunity to develop a product. At the end of 7-8 weeks, all the teams will showcase the app. The client will select a team and their app to work further on this ongoing project.

The interface of the application needs to be user friendly and have information available right on the spot. While designing the user interface, keep in mind that the client has indicated that they do not wish to see tabs in the user interface.

The client has some initial idea of how the app should function, and has described the features that they need:

- The user should be able to plan for a trip going to multiple locations and schedule the booking of a taxi
- The user should be able to book a taxi
- The user should be able to change the taxi type for a scheduled booking
- The user should be able to cancel the booking
- The user should be able to view all their bookings
- The user should be able to view the estimated fare and distance for a booking

Your manager (the demonstrator) has already organised for the client interview to take place and has provided you with the narration of the findings and process [link]. You should ensure you check the FAQ for this assignment as well. Any additional information provided there is considered part of the formal specifications for this project.

The project manager has provided some technical information for the project which is attached on the following pages.

2. COPYRIGHT WARNING

Commonwealth of Australia Copyright Act 1968

Warning

© **2021** Monash University. All Rights Reserved.

This material has been reproduced and communicated to you by or on behalf of Monash University in accordance with section 113P of the Copyright Act 1968 (the Act).

The material in this communication may be subject to copyright under the Act. Any further reproduction or communication of this material by you may be the subject of copyright protection under the Act.

3. Technical Information

This section will outline the technical information required to access various Web Service APIs, data and services to be used in the project.

You are provided with two libraries that you should use:

- services.js that you can load as a call in the HEAD element to use for the API calls to the various data APIs.
- taxiData.js that you should include (download and add to) in your app in the js folder. This provides you with the initial data for the taxis available to you in the app for testing.

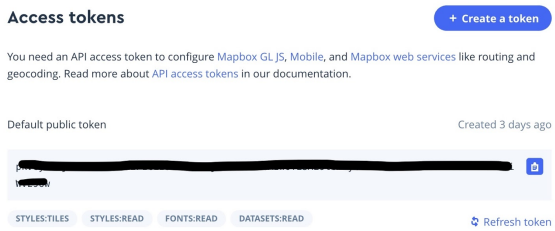
You should ensure that you load these library in the head element of your HTML pages that need them.

3.1. MapBox API

You will need to use the MapBox API to display the Map - you are not allowed to use any additional API or web services other then the ones indicated. If you're unsure, make sure you ask on the edStem forums before using them.

Make sure that your team signs up for a MapBox account by:

1. Visit the signup page [link]
2. Sign up for a new account. You will need a generic Gmail account created for your team prior to sign up.
3. Scroll down to "Access tokens" and note the "default public token" or leave this page open for now. That is your API key.



You can refer to the MapBox walkthrough [link].

You will need to include the MapBox JS and CSS file in your **head** element of the HTML page that needs to show a map.

```
<script src='https://api.tiles.mapbox.com/mapbox-gl-js/v1.1.1/mapbox-gl.js'></script>
<link href='https://api.tiles.mapbox.com/mapbox-gl-js/v1.1.1/mapbox-gl.css' rel='stylesheet'>
```

You will also need to set the size of the mapbox div by assigning it a **class** attribute in HTML along with the width and height in css.

For example:

(in css up in the head of the file, assign 'mapbox' class to the div element)

```
<style>
.mapbox {

   height: 500px,
   width: 50vw
}
</style>
```

vw represents view width (of the screen) and **vh** represents view height. You can also specify pixels (px) instead if you want a fixed size regardless of screen size.

3.2. Geocoding API

You will need to use the OpenCage API for Geocoding [link]. This requires you to sign up for an API key from their website.

You will need to use the query with the **jsonp** parameter.

Forward geocoding

https://api.opencagedata.com/geocode/v1/json?q=LAT+LNG&key=KEY&jsonp=FUNCTION

Reverse geocoding

https://api.opencagedata.com/geocode/v1/json?q=PLACENAME&key=KEY&jsonp=FUNCTION

You should read their documentation carefully.

3.3. Haversine formula for distance on earth

You will need to implement the haversine formula as a function to calculate distance between two points using coordinates.

$a = \sin^2(\Delta\varphi/2) + \cos\varphi1 \cdot \cos\varphi2 \cdot \sin^2(\Delta\lambda/2)$

$c = 2 \cdot \text{atan2}(\sqrt{a}, \sqrt{(1-a)})$

$d = R \cdot c$

where φ is latitude, λ is longitude, R is earth's radius

Here's a sample implementation. All angles need to be in radians.

```
const R = 6371; // km
const φ1 = lat1 * Math.PI/180; // φ, λ in radians
const φ2 = lat2 * Math.PI/180;
const Δφ = (lat2-lat1) * Math.PI/180;
const Δλ = (lon2-lon1) * Math.PI/180;

const a = Math.sin(Δφ/2) * Math.sin(Δφ/2) +
Math.cos(φ1) * Math.cos(φ2) *
Math.sin(Δλ/2) * Math.sin(Δλ/2);
const c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1-a));

let d = R * c; // in km
```

You can use this but replace all symbols with proper variables.

Reference: https://www.movable-type.co.uk/scripts/latlong.html

3.4. Fare structure

Basic Taxi (Sedan) Fare Scheme

Flag rate: RM3.00

Distance based fare rate: 10c/115m

Advanced booking: RM2.00

Night levy: 50% surcharge between midnight and 6AM

Additional levy for larger vehicles

SUV: RM5.00

Van: RM10.00

Minibus: RM15.00

3.5. Storing data in Local Storage

We are interested in saving data on the local device, and you should have the following behavior for data in local storage on the user's device.

- User and Travel data should be stored on the device.

4. Getting Started

Keep in mind that the Team Manager (your demonstrator) will mentor you and update both organisations about your progress on the project.

You can follow the steps below to get started.

1. Each member will need to track the project, task progress and information using Trello. Each member has to use Trello individually to monitor, update, discuss and track their tasks for the assignment on the Team Board.

2. Make sure each member has the team git repository set up on their computer.

3. Your team needs to create an account on MapBox (Walkthrough [link]).

4. You should align your planned work to the assessment milestone / phases set out for you.

5. You can create HTML, JS and CSS files as required. Please give them **sensible** names, and organise your JS, CSS and Image files into **scripts**, **styles**, and **images** folders respectively. See the assignment instructions for the overall folder structure.

6. You can test the code by opening the HTML page(s) directly in Google Chrome.

5. Milestones and Phases

This section will cover the work to be completed in the various phases of the assignment.

- Week 5: Project Planning Phase
- Week 6: Requirements Gathering Phase
- Week 7: Design Phase
- Week 8-11: Prototyping and Implementation Phase

You will have time each week set aside to work on the assignment as a team.

Standard late-submission penalties apply for each phase's deliverable. See each sub-section (follows) for details on the deliverables.

5.1. Planning Phase

Writing a Project Management Plan

Your team will be working on the Project Management Plan for Assignment Two. You should've already discussed many of the details required for the PMP during your meeting in Week 5, so this is a formal step to document the information in one location.

What is a Project Management Plan?

The Project Management Plan (PMP) is an internal introduction document that gives all the information necessary to get started with the project. The document is typically used to provide new team members with sufficient information so that they know the processes and plan for the project that they are joining.

An analogy to the PMP will be your degree course handbook which outlines your prerequisites before enrolling to a unit and course map for the next 3 or 4 years in order to graduate.

The document should contain at least these components:

1. Cover page stating the project title and team members name
2. Table of Contents
3. Introduction / Project information
4. Info on personnel
5. Decisions on process
6. Communication management
7. Risk management

The language used should be formal and free from typos, and written from the *third person perspective*. All pages are titled at the header and numbered at the footer (except cover page and table of contents). The font used should be consistent throughout the document. PMP details policies and decisions along with justification as the following sections in order.

We have provided a template which you can import [Download] into Google Docs to start off with. Please make sure to delete the *blue italics instructions* and fill in the details on the coversheet. You are recommended to pick a different cover sheet design.

Your team will need to make sure that you make it a Google Document rather than 'opening it in google docs'. This enables the activity and change history so that you can revert the document to a previous form, and allow us to see that every member has contributed to the document in a fair manner.

Submission

You will need to submit this document as PDF file on Moodle under the '**Week 6**' section to the 'A2 Project Management Plan Submission'. Standard late-submission penalties apply.

Submission link (Under Week 6): https://lms.monash.edu/mod/assign/view.php?id=8198366

This is due **10th of April 2021 (Saturday) at 8PM Local Campus Time.** Standard late-submission penalties apply.

5.2. Requirements Phase

Gathering Requirements

You are provided with a summarised narrative of the requirements gathering session that was conducted. You will need to extract information from the narrative and the project summary for your requirements document.

Writing a requirements document

You should use Google Docs for the document creation.

The document should be set out in the following structure:

- Cover page
- Table of contents
- Introduction / Project Information
  - Project background and summary
- List of features
- For each feature (starting on a new page)
  - User stories and their associated acceptance criteria

Submission

You will need to submit this document as PDF file on Moodle under the 'Week 7' section to the 'A2 Requirements Document Submission'. Standard late-submission penalties apply.

Submission link (Under Week 7): https://lms.monash.edu/mod/assign/view.php?id=8198368

This is due 17th of April 2021 (Saturday) at 8PM Local Campus Time. Standard late-submission penalties apply.

5.3. Design Phase

Designing the classes (data structures)

For drawing the class diagram, you should use the web tool draw.io.

You should begin by:

1. Identify all the data that needs to be captured in your system
2. Categorise / Group the data into concepts
   (i.e. in a Library system, Book is a concept that stores information about a single Book)
3. Design the classes around the concepts.
4. Draw the class diagram using Draw.io.

Designing the User Interface

1. Draw wireframes for Assignment 2.

   You should use wireframes to design the user interface - use draw.io or any other preferred tool.

   We recommend that you do all annotations digitally instead of doing it on the paper so that it's easier to move the text and lines around later.

2. Draw the storyboard for Assignment 2.

   Using the 'final' version of on the computer (scan in the paper copy or use the digital drawing), do up a storyboard for the assignment.

Writing the Design Document

Next, using the output of the two activities above, your team will create the design document.

You should use Google Docs for the document creation.

The document should be set out in the following structure:

- Cover page
- Table of contents
- Introduction / Project Information
  - Project background and summary
- Class Diagram
- Wireframe for each page
- Storyboard for the app

Submission

You will need to submit this document as PDF file on Moodle under the 'Week 8' section to the 'A2 Design Document Submission'. Standard late-submission penalties apply.

Submission link (Under Week 8): https://lms.monash.edu/mod/assign/view.php?id=8198374

This is due 24th of April 2021 (Saturday) at 8PM Local Campus Time. Standard late-submission penalties apply.

5.4. Prototyping and Implementation Phase

We recommend that you create a HTML-MDL Prototype of your assignment instead of just diving straight into coding it up separately. This allows your team to ensure that your HTML looks the same across each page.

Starting the prototyping process

1. Begin by planning out the file structure and naming scheme for your application. If you're unsure, check with your demonstrator for some help at this point.
2. Next, you will need to select a navigation layout that is closest to your needs from the MDL Reference [link]. You may need to modify it slightly to suit your needs.
3. You should have one person prepare a 'base' file that everyone can use as a template.
4. Next, make sure you choose the primary and accent colors in the MDL Customisation Tool [link], download (and link) the CSS file as appropriate.
5. Once the template file is ready, you can commit it to the Team's Git Repository.
6. Your team can now split the work to be done by pages to the rest of the team, and use copies of the template with the correct file names (as agreed earlier).

Prototyping with MDL

You need to use the MDL Components [link], and build up the pages based on the wireframes that your team have finalised.

If you're unsure about positioning elements and getting elements on the page to show up as you'd like, ask a demonstrator for some help.

Make sure that you commit all your work to the Team's Git Repository.

Planning for Implementation

For this phase, your team will need to begin by creating a plan for implementing **the main functionality** of the assignment. You can also do planning for *all the functionality* (recommended) should you wish to do so, but the least you must do is plan the *main functionality* out.

Your team can choose to either model this as one of the following options:

1. High level and low level psuedocode
2. Activity Diagram (Detailed)

This should be contained within a formal document named the Implementation Plan, which consists of:

1. Cover sheet
2. Table of contents
3. Introduction
4. Diagram / Plan with explanation

Submission

You will need to submit this document as PDF file on Moodle under the '**Week 9**' section to the 'A2 Implementation Plan Submission'. Standard late-submission penalties apply.

Submission link (Under Week 9): https://lms.monash.edu/mod/assign/view.php?id=8198375

This is due **1st of May 2021 (Saturday) at 8PM Local Campus Time.** Standard late-submission penalties apply.

Continuing the Implementation Process

You should work as a team in pairs (using pair programming) to continue the implementation for your app, based on the requirements and design document. You may like to refer to the rubric available on Moodle to ensure that you're not doing anything otherwise incorrect. For any clarification of design or requirements, check with your demonstrator.

Remember to keep your JavaScript and CSS code separate from your HTML pages (i.e. have separate files for them). Don't forget to follow the Coding Standards as you work. It's a lot less work to do as you go rather than leave it to the end.

5.5. Handover (Client Presentation)

This assignment includes a oral presentation component worth 4% of your overall unit mark.

The sponsoring organisation has requested for a presentation of the web app for their top-level and middle-level management staff in a 15 minute session. They want to make sure the mobile web app will meet their requirements and function accurately in challenging circumstances. Specifically, they are interested in seeing the interface and functionality of the web app and would like to hear about the design choices that your team have made.

You need to inform the organisation about any specific hardware requirements and limitations the web app might have at this stage and offer your professional opinion for future development of features in line with their needs.

Monash University has scheduled this client presentation to happen during your practical class in Week 12. You should ensure that your presentation is not framed as a university assignment. The purpose of this is not to sell the app. Remember to prepare well in advance of the due date (including any visual aids) and should be well rehearsed as a group and individually.

You will be required to submit slides to Moodle by the end of your practical class in Week 12 for marking.

You are required to have the video turned on in order for us to assess your individual presentation skills. If you are unable to do so please let your demonstrator know the reason at the beginning before you start the presentation.

We will be recording video via the Zoom platform for each presentation for marking and future staff training purposes. This video will not be distributed or shared.