Assignment 2

Akshatha Hebbar
Akshatha.hebbar01@student.csulb.edu
028017780

Table of Contents

GITHUB

https://github.com/AkshathaHebba/mapty.git

Question 1

Enhance a program that performs CRUD (create, read, update, delete) operations. The user wants to keep the deleted and modified records in the database and to be accessible when needed, as in the banking system or social media. You need to modify this program to accommodate the new requirements. (06)

Mapty Workout Logger

It is a user-friendly JavaScript application which allows the user to monitor, change and manage their workouts. leaflet library is used to identify the user's geo location, using which then opens the map view allowing the user to log their workouts which are **active**.
Types of workouts available:

1. Running 🏃‍♂️

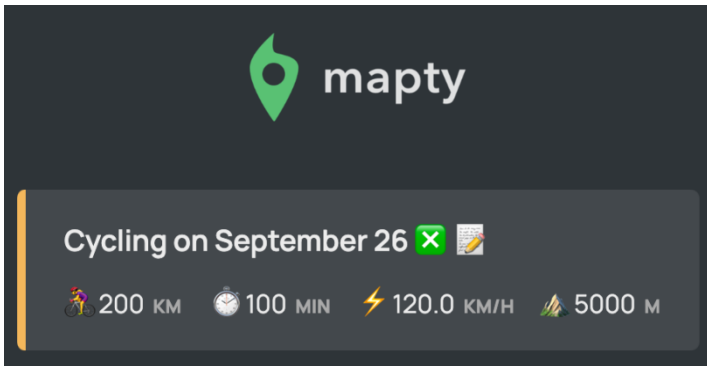
User can log their work out details specific to running like distance (Kms), time taken (Mins) and cadence (SPM – steps per minute). Using this, speed is automatically calculated in min/kms.

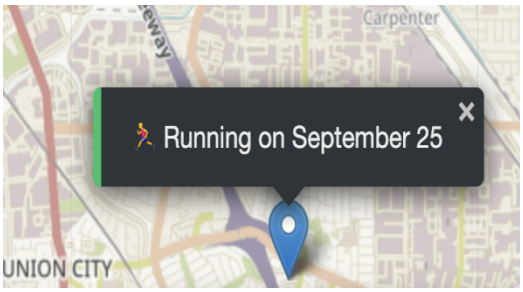
2. Cycling 🚴‍♀️

User can log their work out details specific to Cycling like distance (Kms), time taken (Mins) and elevation gain. Using this, speed is automatically calculated in km.hr.

How to log a workout?

Click on the approximate location on the map which you have completed your workout to log a new workout. Select the type of information relevant to your workout. Hit 'Enter' to log you work out. A card is displayed on the screen with relevant information along with the marker on the map.





How to edit a workout?

Prerequisite: A workout must be created prior of editing.
Click on to re-enter values. Hit 'Enter' after editing to update the records in the database.

How to delete a workout?

Click on to remove the card from the user Interface. The record will be still maintained in the database. It is done by changing the status of the record to **'inactive'.**

```
_id: ObjectId('63323f16f91678321a783045')
id: "4237334475"
clicks: 0
date: 2022-09-27T00:08:54.475+00:00
› coords: Array
distance: 200
status: "inactive"
duration: 100
type: "cycling"
description: "Cycling on September 26"
__v: 0
elevation: 5000
```

How to run Mapty workout Application?

Prerequisite: Node Modules must be installed.
Run the application using: npm run dev.
Running on: Localhost.
To run scripts use: Npm run test

Question 2

Fill out the change request form for a user. You can download a template from the internet. (06)

CHANGE REQUEST FORM

Change Description		
Project Name: Mapty Workout Logger	**Change Name:** Ability to retain record deletion / modification	**Number**: 376
Requested By: User - Rika Ebrar	**Assign To:** Development Team – Akshatha Hebbar	**Date**: 09/8/2022
Description of Change: Feature to allow user to modify the existing record. Ability to retain record when even if the record is deleted.		
Reason for Change: Allowing the record edit and delete would better help the user to keep track and manage their workouts much more efficiently.		
Priority [Circle One]: 1. High (2.) Medium 3. Low		
Impact of Not Responding to Change (and Reason Why): It would create difficult for the user to force them to tolerate unnecessary workouts which are invalid or incorrect.		
Date Needed: 09/28/2022	Approval of Request: Project Manager	Date: 09/8/2022

Change Impact
Tasks/Scope Affected: 1.3.2 (Retain the workouts post delete in backend) 1.3.3 (Delete/modify workout)
Cost Evaluation: $2851
Risk Evaluation: Medium
Quality Evaluation: High
Additional Resources: MongoDB database community version
Duration: 12hrs
Additional Effort: 0.5hrs
Impact on Deadline: Low
Alternative and Recommendations: None
Comments:

Sign Offs	
[Circle One]: (1.) Accepted 2. Deferred 3. Rejected 4. More Info Requested	
Comments: Terminate the storage usage of browser APIs.	
Project Manager Signature:	Date: 09/9/2022
Decision Maker Signature:	Date: 09/10/2022

Question 3

Determine the type of maintenance. Describe what maintenance-related activities you performed/did not perform (like testing/ regression testing). Describe the reason why an activity was not performed. Describe the challenges you faced while maintaining the program. (08)

Type of maintenance

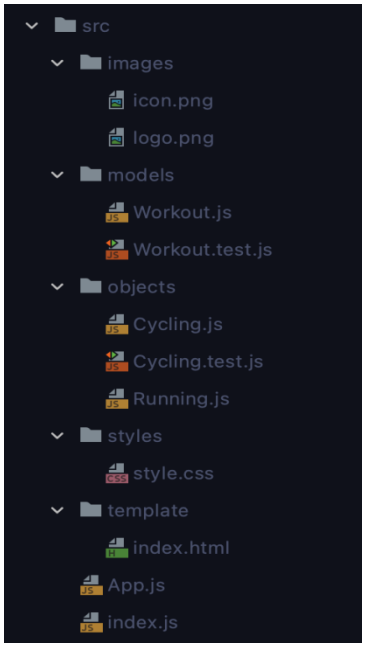
Since this a new change request, it falls under as adding a new functionality. Hence it is categorized as **perfective maintenance**.

Maintenance related activities performed

Some of the maintenance related activities are listed below:

Restructuring a monolith application into components and managing them using webpack

Prior of using webpack, the application was running on a monolithic script.js file, thus hindering an application's extensibility, maintainability. Now the code is broken down into components like objects and classes which can be easily extended and maintained. Webpack is now used to bundle our javaScript, helping us to move from a monolithic structure to a more component-based structure. It automates the code bundling process and deployment instead of manually editing complex script files and deploying them. Webpack also helps our application to improve load times(async chuck loading) and provides a suitable platform to add new features to our application.



Unit Testing

Record validation is performed by predefining values in the file. This enhances the maintenance and restrict developers to modify the code incorrectly as checks are performed before running. Below is an example of the test code.

```
import Cycling from './Cycling';

describe('Cycling', function() {
  test('should be create with correct values.', () => {
    const workout = new Cycling({ distance: 1, duration: 1 });
    expect(workout.status).toBe( expected: 'active');
    expect(workout.getJSON()).toEqual( expected: {
      'description': expect.any(String),
      'type': 'cycling',
      'speed': 60,
      'clicks': 0,
      'date': expect.any(String),
      'distance': 1,
      'duration': 1,
      'id': expect.any(String),
      'status': 'active'
    });
  });
});
```

Maintenance related activities did not perform

Manual end to end testing was performed. But this can be automated using scripts. It can greatly enhance the quality of application.

Challenges you faced while maintaining the program

Since all the code was in a monolithic structure it was difficult to analyze the objects needed to be broken to and it was confusing to decide on the model of the workout which needs to be followed by the objects. Finally, to overcome this situation we split a single script file into src/model/workout.js and src/objects/(cycling| running). This gives us platform the add new workout and modularize the application.