CSC 491/391 Mobile Application Development for iOS II

Programming Assignment 3
Location and Maps

Due Date & Submission

- Assignment due on Tuesday May 18, 2021, 11:59pm
- Submit your assignment in D2L Dropbox.
- Submit a single zip file that contains the contents of the project folder
    - To zip your project folder: Ctrl-click your project folder and select "Compress …" from the context menu.
- **It is mandatory to use Xcode 12 and Swift 5.3 for this assignment.**
- Include only your source files, including
    - *.swift, *.plist, *.xib, *.storyboard
    - image files
    - project files (.xcodeproj)
    - test folders
- You must use a unique prefix for the project name. (I suggest you use your last name and first initial as your prefix.) **Please use the same prefix for all your assignments.**
    - Note you only need to use the prefix for the project name. It is not necessary to use the prefix on other files in your project.
- Do not include unused or unrelated files.
- Before you submit, build and run the project, make sure everything compiles and works. Close your project before zipping the folder.
- Here are the most common reasons assignments are marked down:
    - Project does not build.
    - Project does not build without warnings.
    - One or more items in the Requirements section were not satisfied.
    - A fundamental concept was not understood.
    - Code is sloppy and hard to read (e.g. indentation is not consistent, etc.).
    - Your solution is difficult (or impossible) for someone reading the code to understand due to lack of comments, poor variable/method names, poor solution structure, etc.
- Bonus points.
    - Bonus points may be awarded to projects with *exceptional* qualities in one or more aspects.
    - Bonus points will only be awarded after all the required elements have been satisfied.
    - Bonus points will not be awarded merely for extra amount of work (or code).
    - Extra and sloppy code may cause your assignments to be marked down.
    - Bonus points are awarded at the sole discretion of the instructor.

    - Include a note in the comment in D2L describing the part(s) should be considered for bonus points.

Goals

- Building an iOS app that utilizes the functions offered by the Core Location and Map Kit frameworks.

Requirements

1. Use Core Location to determine your current location and display the location on a map. Display the information that is most relevant to the user's current location in your app, e.g., near-by stations and approaching trains and buses to these stations.
2. Dynamically update and adjust the display of information of near-by stations and the upcoming trains or buses as the user approaches a train or bus station. Imagine that you are walking around the loop, every block or so there are new bus or train stations.
3. Use the location simulation feature to test your app by creating a simulated route in a GPX file. Add the GPX file to a run scheme in Xcode to test the app.
4. The app should be robust. Even when errors or anomalies are encountered during the communication or in the data content, it should able to continue to function to the best possible extent by providing appropriate messages, responses, and the best available information in the UI.

Documentation

It is mandatory to include a brief document that discusses

- The software design of your app, including diagrams of the key components (classes) and their relations.
- Describe the main features of your app and the steps to test all the features in Xcode and the Simulator.
- What error conditions and anomalies during the communication and in the data content that you have considered in the design and implementation of your app? What measures did you take to allow your app to continue to function in the presence of these conditions?
- What measures did you take to avoid cyclic strong references in your code?

Bonus Points

Possible bonus points for this assignment in one or more of the following areas:

- Incorporate additional features using Core Location and Map Kit framework.
- Exceptional quality in the design of your code, e.g., using suitable design patterns, language features, etc. to achieve simplicity.
- Effective design of the user interface.
- Properly handle all possible scenarios that may arise during the execution of the app.
- Clean, well-organized, and easy to understand code.

Document the areas that exhibit these exceptional qualities in your app.