CMPUT 301 Fa19 - INTRO TO SOFTWARE ENGINEERING Combined LAB LEC Fa19

Assignment 1

Assignment 1

Learning Objectives

- Solve a problem by constructing a simple, interactive application using Android and Java.
- Document an object-oriented design in Unified Modeling Language.

Problem Description

Consider the situation of someone who wants to track their bike rides. Make a simple, attractive, intuitive, Android mobile app to note this data. Let us call this app: RideBook.

Specifically, each ride (when it is done) has the following fields:

- date (presented in yyyy-mm-dd format)
- time (presented in hh:mm format)
- distance in km (non-negative decimal)
- average speed in km/h (non-negative decimal)
- average cadence in revolutions per minute (non-negative integer)
- comment (textual, up to 20 characters)

Only the comment field may be left blank for a ride.

The app should allow the user to:

- show a list of rides
- add a new ride (which always appends to the bottom end of the list)
- view and edit the details of an existing ride
- delete a ride
- see the total distance of all rides below the list of rides

The list need not show all the information for a ride if screen space is limited. Minimally, each ride in the list should show its date, time, and distance.

The app must assist the user in proper data entry. For example, use appropriate user interface controls to enforce particular data types and avoid illegal values.

For this assignment, the app need not be persistent.

Use your campus computing ID in the app name. Specifically, the app name must show up as *YOURCCID*-RideBook (e.g., kennyw-RideBook).

Deliverables

1. **Code Base:** (5 marks)
   Your complete source code and compiled binary, implementing the working app and its user interface, will be inspected and run by the TA. The Android Studio project and APK (Android Package Kit) binary must be included in the submission. Each class must contain comments describing its purpose, design rationale, and any outstanding issues.

2. **Video:** (1 mark)
   The video is a recording of the app's user interface. The video file must be included in the submission. Besides being instructional for using the app, the video is meant to show that the demonstration actions below actually work. No audio is needed. Maximum duration is 3 minutes. Focus on just the screen of the app, not your whole desktop. For visual clarity, do not use a handheld camera.

3. **System Documentation:** (2 marks)
   Describe the structure of your app's object-oriented design using UML class diagram(s), saved as non-lossy image file(s). Focus on the most important classes that you designed and implemented. Add notes to describe the main responsibilities of these classes.

Demonstration Actions

1. Open the app from the launcher.
2. Show the list of rides, with no rides so far. (This should be the initial screen.)

3. Add a ride with date 2019-09-27, time 14:00, distance 40.5, speed 25.5, cadence 75, and no comment.
4. Show the list and total, with this ride.
5. View/edit this ride to be distance 41.5, and comment "light wind".
6. Show the list and total, with this updated ride.
7. Add a ride with date 2019-09-28, time 15:30, distance 39.5, speed 26, cadence 76, and comment "sunny"
8. Show the list and total, with the two rides.
9. Add a ride with date 2019-09-29, time 09:05, distance 25, speed 27.5, cadence 80, and comment "cloudy"
10. Show the list and total, with the three rides.
11. Delete the ride dated 2019-09-28.
12. Show the list and total, with the two remaining rides.
13. View the details of the 2019-09-27 ride.
14. View the details of the 2019-09-29 ride.
15. Delete the ride dated 2019-09-27.
16. Delete the ride dated 2019-09-29.

Hints

This is a description of the core functionality. Often, problem statements from users lack details. As you are prototyping a design, you may uncover other behaviors that have not been specified, but make sense in the context and intent of the application. For example, think about how someone might effectively use your application. It is up to you to decide what functions your design will need, based on the given problem description and valid assumptions, in discussion with your users (the TAs and instructor). You should consider asking the customer (the instructor) what they want to see.

While you may discuss your design with other students, the code and documentation must be your own work. Code from publicly available sources may be used within reason and only if their licenses permit so. Always fully cite to give proper credit to the original developers in the source code and in the system documentation. For example, in citing a work, at least state: from whom, the date of publication, license, and URL. Do what is required by its license.

The TAs will be inspecting your code, so "major" commented-out experiments should be cleaned up so that the code is readable.

Do not skimp on the UML class diagrams in the system documentation. For neatness and readability, diagrams should be created or drawn using a vector graphics editing tool and exported in a common, non-lossy graphics format.

Besides addressing the problem correctly, your software design will be evaluated on its proper use of object-oriented design concepts, such as separation of concerns and information hiding.

Losing Marks

You may lose marks for any of the following:

- files not in properly named subdirectories
- missing APK file for the app
- cannot run the app after install
- cannot distinguish CCID from the app name
- cannot view files without specialized tools
- lossy compression used in image file for UML (e.g., JPEG)
- inadequate or improper citations

These are brown M&M rules.

Please also ensure your original files are secure from public access.

Submission Procedure

Create an assignment directory called *YOURCCID*-RideBook/ (e.g., kennyw-RideBook/), and within it have three subdirectories: code/, video/, doc/.

Your whole Android Studio project directory goes as is within code/. The compiled binary APK file(s) should be found within an app/build/outputs/apk/ subdirectory within the project directory. The video file goes in video/. The UML documentation goes in doc/.

Zip the assignment directory and upload to eClass.

Please make sure all the required files are included to build the app. The TA will test your app from the submitted code and APK file.

The app name must show up as *YOURCCID*-RideBook (e.g.,kennyw-RideBook), so that it can be easily distinguished from other submissions.

Submission status

Submission status	No attempt
Grading status	Not graded
Due date	Monday, 30 September 2019, 4:00 PM

Time remaining	7 days 4 hours
Last modified	-
Submission comments	Comments (0)

Add submission

You have not made a submission yet