CMPUT 301 Wi20 - INTRO TO SOFTWARE ENGINEERING Combined LAB LEC Wi20

Dashboard / My courses / CMPUT 301 (Winter 2020 LAB LEC) / General / Assignment 1

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 needs to monitor their blood pressure and heart rate data. Make a simple, attractive, intuitive, Android mobile app to track this data. Let us call this app: CardioBook.

Specifically, each measurement has the following fields:

- date measured (presented in yyyy-mm-dd format)
- time measured (presented in hh:mm format)
- systolic pressure in mm Hg (non-negative integer)
- diastolic pressure in mm Hg (non-negative integer)
- heart rate in beats per minute (non-negative integer)
- comment (textual, up to 20 characters)

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

The app should allow the user to:

- show a list of measurements
- add a new measurement (which always appends to the bottom end of the list)
- view and edit the details of an existing measurement
- delete a measurement
- see unusual blood pressures highlighted or flagged

Normal pressures are systolic between 90 and 140 and diastolic between 60 and 90.

The list need not show all the information for a measurement if space is limited. Minimally, each record in the list should show the date, systolic pressure, diastolic pressure, and heart rate.

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.

The app must be persistent. That is, exiting and fully stopping the app should not lose data.

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

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 demonstration of the app. The video file must be included in the submission. The video is meant to show that the demonstration actions below actually work. No audio is needed. Maximum duration is 3 minutes. You may use the screen recording software in the labs (e.g., `ffmpeg` or `simplescreenrecorder`). Focus on just the screen of the app, not the 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 measurements, with no measurements so far. (This should be the initial screen.)
3. Add a measurement with date 2020-02-01, time 22:00, systolic 126, diastolic 62, heart rate 52, and no comment.
4. Show the list, with this measurement.
5. View/edit this measurement to be systolic 106, and comment "resting".
6. Show the list, with this updated measurement.
7. Add a measurement with date 2020-02-02, time 23:00, systolic 85, diastolic 49, heart rate 60, comment "sitting"
8. Show the list, with the two measurements.
9. Add a measurement with date 2020-02-03, time 19:30, systolic 97, diastolic 63, heart rate 51, comment "laying"
10. Show the list, with the three measurements.
11. Delete the measurement dated 2020-02-02.
12. Show the list, with the two remaining measurements.
13. Exit and stop the app, showing in the running apps list that the app is no longer running.
14. Open the app again.
15. Show the list, with the two measurements.
16. View the details of the 2020-02-01 measurement.
17. View the details of the 2020-02-03 measurement.

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.

Consult the assignment rubric.

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.

Submission Procedure

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

Your Android Studio project directory goes 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-CardioBook` (e.g.,`kennyw-CardioBook`), so that it can be easily distinguished from other submissions.

Submission status

Submission status	No attempt
Grading status	Not graded
Due date	Monday, 3 February 2020, 4:00 PM
Time remaining	9 days 2 hours
Last modified	-
Submission comments	⊞ Comments (0)

Add submission

You have not made a submission yet