COMP 3350 Project Iteration 2

Group 1 (HealthTrack: The Applet for your Diet)

Planning and process (3.5/4)

- GIT (1/1)
  - Version control is being used properly for example, has more than one committer and commits are reasonable size and frequency. They are not only big commits at the end. (1/1)

Comments: All members have committed to GitHub, and the commits are reasonable size and frequency.

- Updated plan (1.5/2)
  - Plan should be up-to-date (if there is any change to the previous plan for Iteration 2 it should be explicit and justified) (0.5/0.5)
  - Big user stories for iteration 3, if it was not already in plan (0/0.5)

I cannot find any user stories for iteration 3 in your plan, I have looked at these files. (-0.5)

- Project_Submission_It2.pdf
- Project.Submission_updated_It1.pdf
- Project.Submission.pdf

  - Development tasks assigned in iteration 2 (what exactly has been done by developers) (0.5/0.5)
  - The time planned for the development tasks and detailed user stories and the actual time it took, in iteration 1 (0.5/0.5)

Comments: This comment below from previous iteration has not been addressed: *"You should clearly specify the iteration name, which the big stories are belong to! Plus detailed stories are breakdown of a given big story, so you should mention that your detailed stories are breakdown of which of your big stories. Since it is not clear which stories are belong to your 2nd iteration, therefore you will lose mark for that."*

In the "Log_File_Combined.-.Log.of.Individual.Time.Spent.csv" you need to provide the legend for authors code (SW ...). The tasks assignment is too generic. Also, in your log files, you should put the commit description aside from commit code.

- Wiki (1/1)
  - Should include description of the content of the submission. Can include other things as well. (1/1)

Comments: Well-structured wiki with proper link to all required files.

Functionality (5.5/8)

- DB support including the actual DB and stubbed version (1/2)

There is no stub under your persistence layer (-0.5), your app crash on any database activity. (-0.5)

- System performs end-to-end (including GUI) processing for all stories (1/2)

UI works fine but not able to write to database. (-1)

- Works on both emulator and tablet device. (2/2)
- The developed program conforms the updated plan (the stories that are claimed to be implemented, are indeed there) (1/1)
- No easy bug (No crashes or unexpected behavior while trying normal scenarios) (0.5/1)

App crashes upon saving user information (-0.5)

Comments: Run on device (Nexus 7) after first registration, app crashed, looking at log it was stating, not able to open the database. Then tried to open the app again and this time didn't prompt user information. Every time saving the user information, the app crashes. Basically, any write to DB crash the app, like goal submission and so on.

Trace:

```
Failed to open database '/data/user/0/club.glamajestic.healthtrack/databases/HEALTHTRACK_DB'.
android.database.sqlite.SQLiteCantOpenDatabaseException: unknown error (code 14): Could not open database at
android.database.sqlite.SQLiteConnection.nativeOpen(Native Method)
```

Implementation (6/6)

- No obvious code/design smells (2/2)
    - Classes are in the wrong package (e.g., logic is developed in the UI layer)
    - Big classes: Classes are taking too much responsibility (SRP)
    - Very long methods (over 20 lines)
    - Wrong usage of inheritance
- Proper dependency injection for DB (2/2)
- Good standard coding style (2/2)
    - Informative naming
    - Comments explain "why" and not "What"
    - No to-do
    - Too much code duplication (copy-paste)

Comments: You need to move the project architecture files from root of your application to wiki-related-files directory (update any link to them too). Aside from that, codebase under app module is properly packaged (business, persistence layers).

Unit tests (6/7)

- Automated JUnit test cases for both new features and old ones (1.5/2)

Limited number of unit tests, mainly testing some low level functions like calculate BMI, BMR and so on. You need to test other features like setting goals. (-0.5)

- Passes all unit tests for domain objects and business logic (2/2)
- Reasonable test coverage of normal and corner cases (0.5/1)

Provided unit tests are not covering some key features like “GoalsAccess”. (-0.5)

- Integration tests (actual DB in the loop) (2/2)

Comments: In general, you need to put more effort on testing, adding different set of unit tests, integration tests.

Penalties (-0.5)

- Log file (up to -2 if missing or incomplete)
- Missing libraries. Unspecified dependencies. (up to -2)
- Previous unresolved issues (up to -5)

Comments: Architecture diagram has been fixed, but again as mentioned earlier, the user stories should have iteration names in which they are supposed to be implemented. (-0.25), under your persistent layer, I couldn’t spot any stub implementation of your database, as it was part of previous iteration requirements. (-0.25)

Total (20.5/25)