

COMP 3350 Project Iteration 2

Group 7 (To-Do-Bot)

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: No comments

- 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) (0/0.5)
 - Big user stories for iteration 3, if it was not already in plan (0.5/0.5)
 - 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:

1. As it was mentioned in it0 you were supposed to create 3-4 big stories per iteration {9-12 big stories in total}. Looking back in your original plan you have enough stories but you are not submitting or committing enough per iteration. This iteration specifically you have just one feature. One feature is below minimum, so we deduct a minimum mark of 0.5. -0.5

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

Comments: No comments

Functionality (7/8)

- DB support including the actual DB and stubbed version (2/2)
- System performs end-to-end (including GUI) processing for all stories (2/2)
- 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) (0.5/1)
- No easy bug (No crashes or unexpected behavior while trying normal scenarios) (0.5/1)

Comments:

1. Your program conforms with the plan but your plan doesn't have minimum features. -0.5
2. Bug: Steps→Add a new task, type anything, click on checkbox or set priority. Text you typed before disappears. -0.5

Implementation (5.75/6)

- No obvious code/design smells (1.75/2)
 - 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:

1. Long methods, for instance in "TaskListItemAdapter.java". -0.25

Unit tests (4/7)

Automated JUnit test cases for both new features and old ones (1/2)

Passes all unit tests for domain objects and business logic (0/2)

Reasonable test coverage of normal and corner cases (1/1)

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

Comments:

1. Unit tests included in the wrong package (should be added to "test" instead of "androidTest") -1
2. Since the tests are not in the correct package, getting "No tests found" when I try to run, whether Unit Test or Instrumentation Tests. I also tried including them in the correct package but it didn't work. -2

Penalties ()

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

Comments: **WARNING:** So far, the app provided is minimal. The only user story worked on in this iteration is about adding “priority”. Looking at the initial content of submission in ITR-0, there are a lot of big stories which are not worked on yet (ALARMS, CATEGORIES, PICTURES and WIDGETS). We deducted 1 mark for this (see comments above for Planning and Functionality), but if you do not commit in the coming iteration you may lose more marks.

Total (20.25/25)