FACULTY OF INFORMATION TECHNOLOGY CTU IN PRAGUE

Review report of a final thesis

Student:	Bc. Jan Matějka
Reviewer:	Ing. Jakub Žitný
Thesis title:	Cross-platform mobile application for safer drone operations
Branch of the study:	Web and Software Engineering

Date: 7. 6. 2020

Evaluation criterion:	*The evaluation scale: 1 to 4.*
1. Fulfilment of the assignment	*1 = assignment fulfilled,* ***2 = assignment fulfilled with minor objections,*** *3 = assignment fulfilled with major objections,* *4 = assignment not fulfilled*

Criteria description:
Assess whether the submitted FT defines the objectives sufficiently and in line with the assignment; whether the objectives are formulated correctly and fulfilled sufficiently. In the comment, specify the points of the assignment that have not been met, assess the severity, impact, and, if appropriate, also the cause of the deficiencies. If the assignment differs substantially from the standards for the FT or if the student has developed the FT beyond the assignment, describe the way it got reflected on the quality of the assignment's fulfilment and the way it affected your final evaluation.

Comments:

The main objective of the thesis was to analyze competitive landscape; and design, implement and test a new app for Dronetag. The job was fulfilled, Dronetag app is ready with reasonably complex architecture and functionality, nice UI, and is prepared for beta testings and possibly more development and release later on. However, part of the assignment, as well as crucial part of all software projects, is testing and documentation. The emphasis on these is inadequate.

Evaluation criterion:	*The evaluation scale: 0 to 100 points (grade A to F).*
2. Main written part	*75 (C)*

Criteria description:
Evaluate whether the extent of the FT is adequate to its content and scope: are all the parts of the FT contentful and necessary? Next, consider whether the submitted FT is actually correct – are there factual errors or inaccuracies? Evaluate the logical structure of the FT, the thematic flow between chapters and whether the text is comprehensible to the reader. Assess whether the formal notations in the FT are used correctly. Assess the typographic and language aspects of the FT, follow the Dean's Directive No. 26/2017, Art. 3. Evaluate whether the relevant sources are properly used, quoted and cited. Verify that all quotes are properly distinguished from the results achieved in the FT, thus, that the citation ethics has not been violated and that the citations are complete and in accordance with citation practices and standards. Finally, evaluate whether the software and other copyrighted works have been used in accordance with their license terms.

Comments:

The thesis begins by setting the context for the reader, provides motivation, describes competitor apps in very good detail. The text continues with the current Dronetag infrastructure where the mobile app is to become a part of. The author describes the Flutter framework, which was chosen by the supervisor in the assignment, and continues to how the code was organized and how the UI was designed. Lastly, the reader finds out about the usability testing and the devops environment for testings and deployment.

The logical structure of the chapters is good; there are no factual errors, the reader is not bored with unnecessary details (with some exceptions) and should have a pretty good perspective about the whole project after reading everything. Most of the facts provided are properly cited. On the other hand, there are aspects of the work that are not great. The level of English is not very good, there are grammar mistakes, and some explanations are not very clear. Furthermore, the author lists pointless "things" several times — list of entities from database model could be replaced with a nice diagram, list of entity properties could also be a part of diagram or appendix, listing elements from Cupertino UI library, components from Material Design or "visual elements" is quite useless and could have just been referenced to.

Since the student is expected to become a Software Engineer, I would expect some of the "discussions" to be more in-depth. One of the opportunities for this was in the chapter 3.1 "Flutter Analysis" — instead of actually discussing pros and cons of Flutter vs React Native, Ionic and Xamarin, the author just lists some properties of these frameworks and provides a reference with very vague comments about "less code and testing" in Flutter or reusability of Ionic code. Comparison with real native mobile environments (like Java or Kotlin for Android and Swift for iOS) could be interesting as well. Another opportunity to discuss a hot topic was in 3.5 where the author describes the BLoC pattern in Flutter — this pattern makes use of RxDart and is comparable with similar concepts in React or Elm world (flux or elm architecture), in some sense it is comparable to MVVM where we could talk about a lot of problems that non-MVVM frameworks solve.

Evaluation criterion:	*The evaluation scale: 0 to 100 points (grade A to F).*
3. Non-written part, attachments	*70 (C)*

Criteria description:
Depending on the nature of the FT, comment on the non-written part of the thesis. For example: SW work – the overall quality of the program. Is the technology used (from the development to deployment) suitable and adequate? HW – functional sample. Evaluate the technology and tools used. Research and experimental work – repeatability of the experiment.

Comments:

The primary goal of this work was to produce code that is extensible by other people in the future and will lead to successful business use. This is absolutely fulfilled, Flutter ecosystem and conventions are respected, the programmer knew what he was doing. The things that could have been better are documentation and test coverage (as mentioned above already).

The tests in this work are not sufficient; the unit test coverage of the written code is only 8%; there are no integration or automated UI tests. Usability testing was mostly focused on UI elements rather than "stories" or "flows", where testers would focus on tasks.

The text of the thesis can be seen as a documentation medium, but there are no diagrams of the database model or API schema (although the list of entities is provided). CI service description, configuration, or an overview of deployment pipelines is not provided. Also, a more complex Readme file in the repository would be nice.

Evaluation criterion:	*The evaluation scale: 0 to 100 points (grade A to F).*
4. Evaluation of results, publication outputs and awards	*90 (A)*

Criteria description:
Depending on the nature of the thesis, estimate whether the thesis results could be deployed in practice; alternatively, evaluate whether the results of the FT extend the already published/known results or whether they bring in completely new findings.

Comments:

The output is ready to be used by Dronetag, which is great.

Evaluation criterion:	*No evaluation scale.*
5. Questions for the defence	

Criteria description:
Formulate questions that the student should answer during the Presentation and defence of the FT in front of the SFE Committee (use a bullet list).

Questions:

1. How do you choose dependencies for a project? Is there a case where you would rather re-implement a small function/class/solution that is already available and open-sourced by someone else?
2. How would you compare the BLoC pattern to MVVM and Flux or Elm architecture [1]?
3. What did you use "dartz" library for? What are the benefits of the functionality that the library provides?

[1] The Elm Architecture, https://guide.elm-lang.org/architecture/

Evaluation criterion:	*The evaluation scale: 0 to 100 points (grade A to F).*
6. The overall evaluation	*75 (C)*

Criteria description:
Summarize which of the aspects of the FT affected your grading process the most. The overall grade does not need to be an arithmetic mean (or other value) calculated from the evaluation in the previous criteria. Generally, a well-fulfilled assignment is assessed by grade A.

Comments:

The quality of the text and the code is good, concepts from Flutter are respected, and future development should not be problematic. The thesis is not a useless project that will die after students graduates. However, from a software graduate from FIT, I would expect to be able to talk more about the architectural details and the challenges of whatever environment he works with. Also to focus on high-quality development practices, which include tests.

Signature of the reviewer: