CENG- 319 Fall 2020 Software Project Outlines

Project Schema:

1- Week 3: **3 Marks**
   a. Present the topic and interview.
   b. Team members and their tasks.
   c. Project topic in word document of up to 1 page.
   d. The project topic should try to solve real world problem.
   e. As a minimum, the project must satisfy the requirements listed below.
   f. Select team members. Each team must have 3 -4 members.

2- Week 6: Milestone 1 **7 Marks**
   a. Distribution of the work among team members.
   b. Work progress: Mockups and requirements analysis.
   c. Project plan and gantt chart.
   d. Mockups, Requirement Analysis, and defining the tools used in the project.
   e. Design Documents and Architecture diagrams.
   f. Present the project to the class.
   g. Explain the logic, functionality, design, implementation, and answer questions from audience.
   h. The app should run on a simulator and a device.
   i. Prototype, UI implementation, and implementing selected functionality.
   j. Draft Test Plan.
   k. Source code committed into Git Repository.

3- Week 9: Milestone 2: **8 Marks**
   a. Expected a working beta release.
   b. The app should run on a simulator and a device.
   c. Connect to server, fetch, and display data.
   d. Demonstrate the use of Git for the source code control.
   e. Explain and run Junit testing.
   f. Update on the work progress.
   g. Store data on the cloud and on device.
   h. Write Unit and Integration test cases.
   i. Design support of Portrait and Landscape.
   j. Demo the app on a tablet and on smartphone.
   k. Source code updated into Git Repository.

4- Week 13: **12 Marks**
   a. Submit final version.
   b. Bug free software with all features and functionality.
   c. Complete test plan.
   d. Present the project to the class and interview with the teacher.
   e. Submit App documents, Android Module,…etc.
   f. Integration testing and final version of test plan.

g. Source code updated into Git Repository.

Project Minimum Requirements:

Logistics:

1- The project topic should try to solve a real world problem.
2. Three members in each team.
3. Provide Gantt chart of the work and Excel sheet with work distribution.
4. Submit a minimum of 6 screenshots!.
5. Use Git for source control.
6. This is an app that will be submitted to Google Play and competes with millions of app in the store for customer attention and use.
7. All code must be under source control and pushed into git hub. In all your submissions, you must provide git hub link for your project.

Functionalities:

8. Should have a minimum of 10 java classes.
9. Should have at least 6 images.
10. Should run on devices and simulators with Android Marshmallow 6.0 (API 23) and above.
11. The application should be able to store data locally and read it back.
12. Should be able to make network connection and run online/offline.
13. Should access remote server to fetch data.
14. Should allow for configuration setting and remember user selections.
15. Write a test plan on how to test and verify the application. The test plan should have a minimum of 10 test cases.
16. Design document and high-level architecture diagram.
17. Handle runtime permission for devices run on API 23 and above. Provide explanation for the permission requested!.
18. Use Junit to write test classes. Mock the different modules in your unit testing.
19. Support two languages: English and French.
20. Image files must support a minimum of three resolutions.

UI/UX:

1. UI color theme following Android UI guidelines.
2. Use NavigationDrawerLayout, TabLayout or both.
3. Font, color contrast, alignment, spacing, and layout should support content structure, and enhance legibility
4. Customized application icon: following color/size guidelines from Android and different than the default one.
5. Adaptive UI layout for portrait and landscape.
6. Fix the layout into portrait only if running on tablet.
7. Use the system defined gesture properly.
8. Make use of launch screen to provide use an instant launch experience

9. Incorporate floating action button in the UI for prominent action
10. Follow guideline to use proper elevation for different UI elements.

Test Plan:

Test cases should follow the following template:

- Test case number
- Test case title
- Test case Purpose
- Precondition to run the test case.
- Steps
- Expected results.