CENG-322

Deliverable 1

FINAL DOCUMENT LAYOUT

Students who are not part of a team, will get 0.

Please submit **a single PDF document per group**. This means that the document should be professionally organized and have a uniform style throughout. It should look as though it came from one team, not 4 separate students. Please note that instructors may choose to run your submission through TurnItIn or compare the submission with other students from other sec8ons for academic integrity purposes.

A single **PDF document** with the following sections denoted using page numbers, headers/footers and a table of contents:

Must follow the sequence below:

Use proper formatting

1. Document Title
2. Team Name
3. Project Name
4. Team Number
5. Student's names and IDs
6. Table of Contents
7. Team Contract (from CENG-322_TEAM_contract_v2.0)
8. Signature of each student who participated in the work (team members who did not participate and where **their signature is missing will get 0.**

9. **Create Google Play Developer Account,** take a screenshot showing account has been approved and created (one or more members).

https://support.google.com/googleplay/android-developer/answer/6112435?hl=en#zippy=

10. Link GitHub Repo. All members must contribute to the repo.
11. Each member must have a minimum of **three** commits. Missing commits **will get 0. Commits must be done from Android Studio.**
12. Screenshot of github invitation.
13. **Project Background and Description:**
    1. Describe the project goals and final vision.
    2. Describe the software aspect and hardware.
    3. Describe the screen flows.
    4. How you Incorporated the feedback provided through the interview.
    5. Demonstrate how you are planning to satisfy to read / write from the DB which is hosted on the cloud.
14. **Project Scope:** Describe the technical scope of the project by talking about the project plan, and how you will know when the project is complete.
15. **Describe** how to integrate the software with the hardware components.
16. Project layout: write how you plan to implement Navigation Drawer, TabsLayout or Bottom Navigation

https://m2.material.io/components/navigation-drawer/

https://m2.material.io/components/bottom-navigation

https://m2.material.io/components/tabs/android

How to use them, why to use, use cases of existing applications that utilize them. one vs the others, and use cases if existing apps that use the above.

17. **Write Minimum of 1 theme: for the Theme write:**
    1. **Two epics, 3 stories for each epic, 3 tasks for each story.**
18. If document not in pdf, or not the proper name (marks will be deducted!)

19. Please save the PDF document, i.e. TeamName_ProjectName_GroupNumber_Deliverable_1.pdf

20. The last submitted pdf file will be marked.

Android Studio Project pushed into in github:

21. Please Update your github Id, to be YourFullName4Digits, i.e. JohnSmith1111.
22. Replace John with your firstname, Smith with your last name, and 11111 with your studentId.

23. Use Navigation Drawer, Tabbed layout or Bottom Navigation
24. Minimum API 29.
25. Target API 34
26. Create a new project, File -> New -> New Project.
27. Project name must be YourProjectName
28. Module name must be YourModuleName.
29. Package name must be ca.teamname.it.projectname (i.e. ca.security.solutions.it.home.security)
30. Java classes must start with upper case.
31. Use proper name for your Java classes that reflects the functionality.
32. All resources in strings.xml, no hardcoding of text.
33. Must Support **English and French**.
34. Full Name, student ID, and section number, must be on ALL JAVA classes as **a comment**. Name and student id of all participants.
35. Full Name, student ID, and section number, must be on both strings.xml as **a comment**. Name and student id of all participants.
36. In strings.xml, app_name must be your project name.
37. **MUST** use Fragments for the individual screens.
38. Demonstrate the use of styles in objects.
39. Use homescreen icon different than the default Android icon. Use an icon that reflects the app behaviour.
40. Intercept user press on back key and display Alert Dialog whether user wants to stay on app or exit. Use proper icon for the dialog.
41. Every time you want to commit, select root directory, Commit using Git -> Commit Directory.
42. Application must not have compilation errors.
43. Application must not crash.
44. Design the application splash screen, that has application image and description of the application. Splash screen appears for 3 seconds, then user lands on main screen.

Github Requirements: Follow instructions below

Repository name must be **YourProjectName, no spaces**.

45. Make your repository **private** (not public).
46. One of the team members to invite the professor as a collaborator (invite haki11). Take a screenshot of the invitation. **Screenshot must show the other team members and the professor.**
47. Must Commit on your machine and push into github.
48. After each commit, push.
49. Each member must have a minimum of 3 commit/pushes into master branch.
50. Every time you want to commit, select root directory, Commit using Git -> Commit Directory. The entire project must be committed and not just the module.
51. Add .gitignore file with the appropriate content into the project folder. Add team name as a comment into the first line of .gitignore (Then add all members names and students ids).
    https://www.toptal.com/developers/gitignore

    Please note the file .gitignore must be at project level and not module level (**Delete** the one in your module).

52. Add README.md file into the project folder, with the appropriate content explaining about your project. Use the appropriate tagging format. **Add Project name as first line in the file README.md**
53. Add members names and students ids into the file README.
    https://www.makeareadme.com/

54. **Include project image in your README as in the example:**
    https://github.com/johncodeos-blog/CheckInternetAndroidExample
55. All members names and student's ids as second line.
    https://www.makeareadme.com/

56. Add project link into the README file.

    Please note file must be at project level and not module level.

57. At the project level, create a directory called docs, under docs create deliverable1. Copy the pdf into the directory. Commit and push.
58. The entire project should be committed and pushed where it can be cloned.
59. Your submission must be a single pdf file as described in step 18.