Spring 2023 CEG-4110 CEG-6110 Introduction to Software Engineering Example Project

Software Development Plan (Agile)

Project Title:

- Intellec 8 Mod 80 Emulator

Project Synopsis:

Create a To Do List app to remind the user to complete their To Do list items, on android devices

Concept of Operation (CONOPS):

The To Do List app will provide a GUI interface to allow the user to visually add and keep track of their desired list items. Additionally, the app will support scheduling list items; where the user designates the time and date an item must be completed and the app sends a notification everyday the task has not been completed before the deadline.

This app will be compatible with all android 12 and later devices

Development Methodology:

- Agile

Requirements/User Stories:

- Requirements will be traced through the code using Doxygen. ***(The code will be located in the src directory and the Doxygen output will placed in the GitHub document directory)***
- An excel spreadsheet w*ill be used to document both th*e product backlog and the sprint backlog.
- ***(The spreadsheet will be located in the GitHub repository in the document directory.)***

Design:

- A minimal design document will be developed to document key design decisions.

- ***The design document will be located in GitHub CEG-4110_2022-Fall_Example/doc/Design/***

Coverage:

- ***A combination of automated code coverage tool and code coverage analysis will be used to document the test coverage of the code. Only newly developed code will be subject to testing.***

Commits and Pull Requests:

- ***Team members will each provide commits and pull requests. The Commits and Pull Requests will be contained in GitHub.***

Minutes:

- ***Meeting Minutes will be placed in GitHub in the directory: CEG-4110_2022-Fall_Example/doc/minutes/***

Actions:

- ***Actions will be maintained in GitHub.***

Reviews:

- ***Each developer must provide at least 10 non-trivial items of feedback within Pull Requests made by someone other than yourself. You cannot review your own Pull Requests.***

Actions/GitHub Issues:

- ***Each developer must complete and close at least 5 Actions/Tasks (GitHub Issues) that are correctly stated with objective completion criteria, documented in GitHub Issues, and assigned to you. You may be required to supply evidence of Action completion.***

Identify Roles:

- Team: Byce McNary, Austin Hwang, Marwa Qureshi, Alyana Barrera
- Scrum Master: Jay Stewart

Schedule:

- Project start date: 1/30/2022
- Sprint durations: 2 Weeks starting 1/30/2023
- Project end date: 4/10/2023
- Sprints Starts:
    - 1/30/2023
    - 2/13/2023

- 2/27/2023
- 3/13/2023
- 3/27/2023
- 4/10/2023 Sprint End Date

Software Development Environment

- Operating System: MacOS and Windows
- IDE: Android Studio
- Emulator: Pixel 5
- Doxygen
- GitHub

Configuration Management

- GitHub
- Items will be configured as follows:
  - Documents, upon initial development
  - Software, upon initial development and passing
  - Software pulls will only occur when agreed upon by the team (me, myself and I).

Target Environment

- Hardware: Android 12 devices
- Operating System: Android 12

Identify The Product Backlog

- The following identifies the initial backlog.
- Updates to the backlog will be maintained as part of the sprint reviews

		Verification			
Rqt #	Requirement	T	D	A	I
	UI				
UI-1	The app shall display task items		D		
UI-2	The app shall allow adding/removing task items		D		
UI-3	The app shall allow task item scheduling		D		
UI-4	The app shall allow user input		D		
	Backend				
BCK-1	UI components shall be linked to the backend	T			
BCK-2	Tasks shall be saved locally to the android device	T			
	T - Test				
	D - Demonstrate				
	A - Analysis				
	I - Inspection				

Identify Initial Sprint Backlog

- The following identifies the initial estimate of items to be worked in each sprint.
    - 1/30/2023 Sprint 1
        - Setup development environment
        - Setup GitHub for Version Control
        - Learn Android Studio
        - Create personal branch
    - 2/13/2023 Sprint 2
        - Develop UI List to display task items
        - Link UI to backend
        - Integrate button to allow for adding items
    - 2/27/2023 Sprint 3
        - Integrate deleting task items from UI list
        - Integrate marking task items as complete
    - 3/13/2023 Sprint 4
        - Save list items locally to android device
        - Advanced User input options
    - 3/27/2023 Sprint 5
        - Create Scheduling for Task items
        - Send Android notification for Task items
    - 4/10/2023 Sprint 6
        - Stretch Goals / Additional Features
        - Prepare for demonstration

Sprint Execution:

The following will be used as an outline for the sprint review and the sprint planning meeting. The sprint review, retro and sprint planning meeting will be combined due to the brevity of the development timeline.

Sprint Planning:

Inputs:

- Product backlog
- Sprint backlog
- Any required technical Information (interface definitions, etc.)

Team Activity:

- Scrum Master – Works with the team and product owner to define the new sprint backlog.
- Development Team – Provides feedback on task efforts and sprint backlog.
- All members define what “Done” is.

Outputs:

- Refined product backlog
- Updated sprint backlog

Daily Scrum:

Inputs:

- Sprint backlog

Team Activity:

- Scrum master – polls the team as to the status
- Development Team – provides status on progress and other needs

Outputs:

- Updated sprint backlog
- Meeting minutes (Date, Time, Attendees, at a minimum)

Sprint Review:

Inputs:

- Sprint backlog

Team Activity:

- Sprint Review

Outputs:

- Delivered new software capability (Increment)