Name \_\_\_\_\_\_\_\_\_Yu-Hong, Jhuo\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Create testing cases and Jasmine tester for Q1**
2. **Coding app for question#1**
3. **Restructure Q1 code for viewmodel and inheritance**
4. **Create test cases and jasmine tester for Q2**
5. **Construct Q2 code**
6. **Restructure Q1 and Q2 code to reduce duplication and apply MVVM model**
7. **Create Q3 testing cases and Jasmine tester**
8. **Construct Q3 App**
9. **Coding Q4 test cases and tester**
10. **Construct Q4 app**

**ITERATION PLAN**

❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒

🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 goal

🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 planned tasks in sequence

🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 time estimate for each task [10 minute blocks]

🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 record of the actual time

**DESIGN LEVEL CLASS DIAGRAM**

❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒

**PLAN FOR HOW FEATURE WORKS.**

❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒

NOTE: AT LEAST ONE of each of the following

🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 UML dynamic diagram

🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 use case

🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 storyboard

🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 wireframe

🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 pseudocode

🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 ‘Planning A Complex Algorithm’ worksheet

🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 🔿 spike solution

**TEST PLANS**

❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒

**WORKING PROGRAM WHICH PASSES STANDARDJS VALIDATION**

❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒

❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒

❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒

**TEST RESULTS**

❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒

**ERROR LOGS EACH RECORDING AT LEAST 10 ERRORS**

❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒

**PIP**

❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒ ❒