Process Deliverable: Agile

Retrospective:

1. **What Went Well**:
    - We successfully sketched and visualized our product. Using high-level and low-level designs, we finally saw our ideas take shape, which was really motivating for the team.
2. **What Didn't Go as Planned**:
    - We struggled with deciding how our product should look. There were a lot of different versions and mock-ups, and we couldn't easily agree on which direction to take. It slowed us down a bit.
3. **What Can We Do Better**:
    - Next time, we should spend more time talking through what we want the final product to look like before jumping into sketches. It would make the design process smoother and save us some time.

Prioritized Tasks (Sprint Planning):

1. Start working on the software for our product:
    - Set up everything we need to start coding.
    - Work on the main feature that compiles TODO comments.
    - Focus on sorting TODOs by priority first since it's one of the most important features.
2. Write test cases for stakeholders:
    - Go through our project requirements and figure out the key features we need to test.
    - Write at least 10 test cases for things like compiling TODOs, sorting priorities, and handling edge cases.
    - Make sure the test cases are easy to understand and actually reflect what the stakeholders expect.

Black Box Test Plan

Project Name: TODO Comment Compiler
Purpose:
To ensure the tool compiles TODO comments accurately, organizes them by priority, and meets the project's acceptance criteria and design constraints.

Test Cases for TODO Comment Compiler

1. **Test Case ID:** TADS_Test1
   Description: Check if the tool compiles all TODO comments from a single file.
   Input Data: A file with 5 TODO comments.
   Expected Output: The output shows all 5 TODO comments.
   Actual Result: n/a
   Pass/Fail: n/a

---

2. **Test Case ID:** TADS_Test2
   Description: Verify the tool compiles TODO comments from multiple files.
   Input Data: 3 files containing 2, 4, and 3 TODO comments.
   Expected Output: The output shows all 9 TODO comments combined.
   Actual Result: n/a
   Pass/Fail: n/a

---

3. **Test Case ID:** TADS_Test3
   Description: Check if TODO comments are organized by priority.
   Input Data: TODOs tagged with High, Medium, and Low priorities.
   Expected Output: TODO comments are listed as: High → Medium → Low.
   Actual Result: n/a
   Pass/Fail: n/a

---

4. **Test Case ID:** TADS_Test4
   Description: Test how the tool handles TODO comments without a priority tag.
   Input Data: TODOs with missing priority tags.
   Expected Output: Comments without priority tags are flagged and placed at the bottom of the list.
   Actual Result: n/a
   Pass/Fail: n/a

---

5. **Test Case ID:** TADS_Test5
   Description: Verify that duplicate TODO comments are handled correctly.
   Input Data: A file with two identical TODO comments.
   Expected Output: Only one of the duplicates is listed in the output.
   Actual Result: n/a
   Pass/Fail: n/a

---

6. **Test Case ID:** TADS_Test6
   Description: Ensure the tool recognizes different formats of TODO comments.
   Input Data: TODOs written as TODO:, // TODO, and # TODO.
   Expected Output: All TODO formats are compiled into the output list.
   Actual Result: n/a
   Pass/Fail: n/a

---

7. **Test Case ID:** TADS_Test7
   Description: Check if the tool can export the compiled list.
   Input Data: A compiled list of TODO comments.
   Expected Output: The list is successfully exported to a file.
   Actual Result: n/a
   Pass/Fail: n/a

---

8. **Test Case ID:** TADS_Test8
   Description: Verify the summary feature works correctly.
   Input Data: A file with 10 TODO comments (4 High, 3 Medium, 3 Low).
   Expected Output: The summary shows: Total TODOs: 10; High Priority: 4; Medium Priority: 3; Low Priority: 3.
   Actual Result: n/a
   Pass/Fail: n/a

---

9. **Test Case ID:** TADS_Test9
   Description: Test the tool with an empty source file.
   Input Data: An empty source file.
   Expected Output: The output message says: "No TODO comments found."
   Actual Result: n/a
   Pass/Fail: n/a

---

10. **Test Case ID:** TADS_Test10
    Description: Check the tool’s performance with a large file.
    Input Data: A file containing 1000 TODO comments.
    Expected Output: All 1000 TODO comments are compiled within 2 seconds.
    Actual Result: n/a
    Pass/Fail: n/a