

### Product Backlog ( /2.5)

- The comments from Deliverable 2 must be addressed.
- ✓ For any comments that were not addressed, deduct -2 points.
- The backlog should have been updated with respect to sprint numbers, story points, etc.
- ✓ If there is no update, deduct -2.5 points.

### Sprint Backlog 3 ( /2.5)

- The comments from Deliverable 2 must be addressed.
- ✓ For any comments that were not addressed, deduct -2 points.
- Should reflect the product backlog. That is, all the user stories that are assigned to Sprint 3 in the product backlog, should be shown in Sprint Backlog 3.
- ✓ For any missing user story, deduct -2 points.
- ✓ If Sprint Backlog 3 does not reflect the product backlog, deduct -5 points.

### Updated GitHub ( /2.5)

- It should be updated from Deliverable 2. If not, give 0.
- Contributions should be consistent, and all students contribute.
- If a student does not contribute to Github, the student will be graded as 0.

### Updated Kanban ( /2.5)

- Kanban board should show progress.
- ✓ It should be updated from Deliverable 2. If not, give 0.
- If no progress is shown, the grade for this part will be 0.

### Sprint Review 3 ( /5)

- Updated, and it follows the guideline.
- Should follow the template posted on Brightspace.
- For any missing item from the template, deduct 2 points.
- Should be for Sprint 1- 3.

### SRS ( /5) - If updated

- Need to check if the comments from Deliverables 1 and 2 have been addressed.
- For any comment that is not addressed (every single one), deduct -1 point.
- If the number of FR and NFR were lower than 15 and 10 in Deliverable 2, they have to be updated in this deliverable. If the numbers still do not meet the minimum requirements, give 0.
- If the UI is missing, give 0.
- If the UI is not updated, deduct -3 points.
- If the SRS didn't need any update, then the 5% will be added to the Use Case Model grade, and no grade will be given for SRS.
- ✓ That is, SRS will be 0, and Use Case models will be /15.

### Architecture Design (/5)

- The comments from Deliverable 2 must be addressed.
- ✓ For any comments that were not addressed, deduct -2 points.

### Use Case Models and Descriptions ( /10 or /15)

- The comments from Deliverable 2 must be addressed.
- ✓ For any comments that were not addressed, deduct -2 points.

- If the number of UCDs was below 10 or the models were below 2 in Deliverable 2, and they did not change in Deliverable 3, deduct -5 points for each.

### Sequence Diagram ( /30)

- About 7 - 15 analysis sequence diagrams plus 3 -7 design sequence diagrams.
- ✓ If the number of analysis/design sequence diagrams is below 7 and/or 3, respectively, deduct -3 points per each. For example, if there are only 5 analysis SD, deduct -6 points.
- Each diagram needs to have correct variables, methods, and parameters.
- ✓ For any mistake, deduct -2 points.
- For each diagram, you need to write a description of the steps.
- ✓ For each description that is missing, deduct -5 points.
- ✓ For each incomplete description, deduct -2.5 points.

### Domain Model ( /20)

- Students are expected to have the model of the domain with their links and relationships. The expectation is that you have at least 10 classes in their diagram.
- ✓ If the number of classes is below 10, per each missing class, deduct -2.5 points.
- If the model is too large, break it down into 2 – 4 smaller models.
- ✓ Otherwise, deduct -5 points overall.
- UML domain model needs to include attributes, links, and methods.
- ✓ Per every mistake, deduct -2 points.
- ✓ If the domain includes something like a database or UI, deduct -2 points per mistake.
- ✓ Links should be correct. If not, for each mistake, deducts -2 points.
- Students also need to have a description of the domain model as a whole and how each smaller model is related to the rest.
- ✓ If the description is missing, deduct -10 points.
- ✓ If the description is not completed, deduct -5 points.

### A new version of the application ( /15)

- If there is no progress in comparison to Deliverable 2, give 0.
- It is recommended to develop 1 to 2 of the features of your application.
- ✓ If no feature is implemented, give 0. Otherwise, if at least 1 feature is started, do not deduct any points.
- If there is no UI and class implementation such as sign-up, or login, deduct -5 points for each.
- All the dependencies, requirements etc, must be all set.
- ✓ If there is nothing, deduct -5 points.
- A README file also needs to be created.
- ✓ If there is nothing, deduct -5 points.

**Total = /100**