

# Individual Homework 2 (HW2)

[Start Assignment](#)

- Due Oct 15 by 11:59pm
- Points 100
- Submitting a file upload
- File Types zip
- Available Sep 1 at 12am - Oct 17 at 11:59pm

## Change Log

**Bold has been added to items to draw your attention to them.**

### Task 7. Has been changed to the following:

Produce a set of screencasts as follows, using either .mp4 or .mov file formats. The **goal** of each screencast is for you to **show** and **explain** to the **grader** why the grader should believe that you have done every item listed for each deliverable and that the item is correct. Use Zoom to capture the screen and your voice, but not your face. Your HW2, after being converted into a ZIP archive, **must be small enough** to be submitted to Canvas. **Do not place your videos in some other publicly available repository!**

---

### Task 7 1) Has been changed to the following

Show each of the User Stories in your **User Stories.pdf** from Task 2 and explain why each is needed by the requirements for this assignment. This screencast should be **at least two minutes long and no longer than five minutes**. Name the Screencast “**Screencast1**” (with the appropriate filename extension) and place it in your HW2 folder created in Task 1.

---

### Task 7. 4) Has been changed to the following

Show and explain the new code that you added to TP1 to implement the CRUD functionality, the User Stories, and the input validation from Task 2. Also, show and explain how your new code and its internal documentation are in alignment with the structure of the provided FoundationsF25 application, code, and documentation style used in your solution to Task 4. This screencast should be **at least two minutes long and no longer than five minutes**. Name the Screencast “**Screencast4**” (with the appropriate filename extension) and place it in your HW2 folder created in Task 1.

---

### Task 7. 5) Has been changed to the following

Demonstrate the **entire HW2** application, explain how the code works, and how it satisfies the CRUD and input validation requirements. This screencast should be **at least three minutes long and no longer than six minutes**. Name the Screencast “**Screencast5**” (with the appropriate filename extension) and place it in your HW2 folder created in Task 1.

---

### Task 7. 6) Has been changed to the following

Demonstrate the execution of each of the test cases in your **Test Cases.pdf** and explain how the output supports your assertion that the code passed each test. This screencast should be **at least three minutes long and no longer than six minutes**. Name the Screencast “**Screencast6**” (with the appropriate filename extension) and place it in your HW2 folder created in Task 1.

---

**Task 7. 7) Has been changed to the following**

Show and explain the outline in your **Staff Epic.pdf** you produced for Task 6. This screencast should be **at least two minutes long and no longer than five minutes**. Name the Screencast “**Screencast7**” (with the appropriate filename extension) and place it in your HW2 folder created in Task 1.

---

## Introduction

Using the provided foundational code and materials, as well as HW1 and/or TP1, each student must design, document, create, and then test a standalone application demonstrating a set of Create, Read, Update, and Delete (CRUD) and input validation operations for each of the following items with sample data appropriate for the Team Project.

- a post class
- a reply class
- a class that supports storing all current posts as well as any subset of the posts (e.g., a subset reflecting the results of a search)
- a class that supports storing all replies to all stored posts and any subset of replies (e.g., a subset reflecting the results of a search)

Note: The above-mentioned subsets might be empty, might contain one or more elements, and may be arbitrarily large. There is no fixed upper limit to the number of elements in a list.

You are urged to use Ed Discussion to see the kinds of messages students sent and the kind of replies students generated. What information is present in the Ed Discussion system that should be part of the Team Project program? Your testing is expected to show **realistic data** (e.g., use actual Ed Discussion questions and answers).

You are expected to engage with others in the class, directly or via Ed Discussion, to identify potential attributes for the post and reply classes, and use the insights you have gained to produce what you believe is required. Do not blindly implement what you believe others are doing, and most definitely, **do not copy** their work!

Produce a set of test cases that cover the important aspects of this assignment. Write simple statements about what and how to test each of these important aspects.

Document your code and your automated testing in a manner consistent with the provided materials. Remember, the code tells the computer (and humans reading the code) what to do, but does not explain **why this code is being used or why some other, possibly simpler, solution was not used**. These are the kinds of comments that are truly valuable!

With the requirements satisfied, you must create a short screencast that shows that the application performs each requirement and passes the tests that you identified.

You are encouraged to get help from your classmates if you have issues, and offer to help others if you have it running. Whatever you do, do not share your solution to place your solution online where others can access it!

## Tasks

This homework requires you to perform the following tasks. Be aware that team-oriented task 6 needs to be done in parallel with tasks 1 - 5 and 7 - 8.

1. **Create** a folder named “**HW2**” and use this folder to hold the various elements of the submission.
2. **Create** and **document** a subset list of the provided Student User Stories that cover what you believe is required to satisfy all the CRUD and input validation operations based on what you have read (above and elsewhere) and what you have learned from your participation in Ed Discussions and interactions with the members of the instructional team (e.g., the professors and TAs). You may need to break

one or more of these provided User Stories into smaller stories for this assignment, as they may address more than CRUD. You are **not required** to implement functions beyond CRUD and input validation, but you may choose to implement some that will help you test these requirements. Create a document named **"User Stories.pdf"** containing the results of this task. Add the PDF to the HW2 folder.

3. **Create** and **document** a list of tests that you believe will cover the required functionality (both positive and negative), including input validation and the generation of helpful error messages. Create a document named **"Test Cases.pdf"** containing the results of this task. Add the PDF to the HW2 folder.
4. Using the Team's TP1 solution as a baseline, **create, document, and implement** a Java application named HW2 that addresses the User Stories from Task 2 and provides internal documentation consistent with the provided FoundationsF25 code. (Be sure to produce four (CRUD) UML Class Diagrams and at least one Sequence Diagram using the provided samples in the "Architecture and Design Documents" portion of the "Individual Homeworks and Team Projects" Module.) Implement the test cases from Task 3, in a manner consistent with TP1, to cover the successful execution of the User Stories and detect failures that produce helpful error messages. Produce a folder named **"HW2 Source Code"**, populating it with the source code from Eclipse, and when you are finished, add the HW2 Source Code folder to HW2.
5. Test HW2 to convince yourself that it satisfies the User Stories you created in Task 2 and properly detects improper input with helpful error messages. You will be required to create a screencast that describes the User Stories, show and explain how the test cases cover those user stories, and then demonstrate that the application and the test cases work.
6. Work with your team, use the provided vision for the instructional team (staff) role, and develop a proposed set of epics (each with a scenario) to cover the role. Partition the epics so that each team member has an epic without duplication, if possible. You are then required to expand your epic and scenario by proposing a set of input values, actions to be performed, and output values that cover the epic. In addition, you must outline why each value and operation is required by the epic. The results of this task are used to produce a document named **"Staff Epic.pdf"**, and when you are finished, add this PDF folder to HW2.
7. Produce a set of screencasts as follows, using either .mp4 or .mov file formats. The **goal** of each screencast is for you to **show** and **explain** to the **grader** why the grader should believe that you have done every item listed for each deliverable and that the item is correct. Use Zoom to capture the screen and your voice, but not your face. Your HW2, after being converted into a ZIP archive, **must be small enough** to be submitted to Canvas. **Do not place your videos in some other publicly available repository!**
  - 1) Show each of the User Stories in your **User Stories.pdf** from Task 2 and explain why each is needed by the requirements for this assignment. This screencast should be **at least two minutes long and no longer than five minutes**. Name the Screencast **"Screencast1"** (with the appropriate filename extension) and place it in your HW2 folder created in Task 1.
  - 2) Show and explain each of the test cases in your **Test Cases.pdf** and how they satisfy the requirements stated in Task 3. This screencast should be **at least two minutes long and no longer than five minutes**. Name the Screencast **"Screencast2"** (with the appropriate filename extension) and place it in your HW2 folder created in Task 1.
  - 3) Show and explain each of the test cases in the code that covers all of the Text Cases in your **Test Cases.pdf**, and how they satisfy the requirements stated in Task 3. This screencast should be **at least two minutes long and no longer than five minutes**. Name the Screencast **"Screencast3"** (with the appropriate filename extension) and place it in your HW2 folder created in Task 1.
  - 4) Show and explain the new code that you added to TP1 to implement the CRUD functionality, the User Stories, and the input validation from Task 2. Also, show and explain how your new code and its internal documentation are in alignment with the structure of the provided FoundationsF25 application, code, and documentation style used in your solution to Task 4. This screencast should be **at least two minutes long and no longer than five minutes**. Name the Screencast **"Screencast4"** (with the appropriate filename extension) and place it in your HW2 folder created in Task 1.

- 5) Demonstrate the **entire HW2** application, explain how the code works, and how it satisfies the CRUD and input validation requirements. This screencast should be **at least three minutes long and no longer than six minutes**. Name the Screencast "**Screencast5**" (with the appropriate filename extension) and place it in your HW2 folder created in Task 1.
- 6) Demonstrate the execution of each of the test cases in your **Test Cases.pdf** and explain how the output supports your assertion that the code passed each test. This screencast should be **at least three minutes long and no longer than six minutes**. Name the Screencast "**Screencast6**" (with the appropriate filename extension) and place it in your HW2 folder created in Task 1.
- 7) Show and explain the outline in your **Staff Epic.pdf** you produced for Task 6. This screencast should be **at least two minutes long and no longer than five minutes**. Name the Screencast "**Screencast7**" (with the appropriate filename extension) and place it in your HW2 folder created in Task 1.
8. Compress the HW1 submission folder into a zip archive and submit it. Submit it before the deadline so there is enough time for the upload of the submission to finish, for Canvas to process the submission, and for Canvas to add it to its data repository before the deadline. Just **starting** the upload before the deadline is **not adequate!** If you find yourself still working an hour before the deadline, submit what you have at that point, and then continue to work to finish the assignment. Partial credit for an incomplete submission is far better than a zero for failing to submit before the deadline.

## Deliverables

A HW1 submission archive must contain the following items.

- Task 1: The HW2 archive and its contents are named properly as described in Task 1. (5%)
- Task 2: Create and document a subset list of the provided Student User Stories. (5%)
  - Covers CRUD (1%)
  - Covers input validation (3%)
  - Does not include material unrelated to CRUD or input validation (1%)
- Task 3: **Create** and **document** a list of tests that you believe will cover the required functionality (both positive and negative), including input validation and the generation of helpful error messages: (5%)
  - Covers all of the Task 2 User Stories. (1%)
  - Includes both positive and negative tests. (1%)
  - Negative tests include helpful and context-relevant error messages. (3%)
- Task 4: **Create, document, and implement** a Java application named **HW2**: (15%)
  - The code and internal documentation are consistent with the provided FoundationsF25 code. (2%)
  - Covers the User Stories from Task 2. (2%)
  - Includes four useful CRUD UML Class Diagrams. (3%)
  - Includes a relevant Sequence Diagram. (3%)
  - Implements the test cases from Task 3 in a manner consistent with TP1. (5%)
- Task 6: Given a **staff epic and scenario**, produce an **outline** covering the epic's **input values, actions performed, and output produced**. (10%)
  - An epic and scenario are provided. (1%)
  - The remainder of the deliverable is in the form of an outline. (1%)
  - The outline includes the epic's input values, actions performed, and output produced. (2%)
  - Explain why you believe the added code fits the structure and style of the existing code. (3%)

- Explain why each value and operation is required by the epic. (3%)
- Task 7.1: **Screencast1**: (5%)
  - The screencast is named properly, uses one of the required formats, the textual materials shown in the screencast must be readable, the audio must be clear and understandable, and the screencast is of the proper duration. (1%)
  - The screencast shows and describes each User Story from the User Stories.pdf. (2%)
  - The screencast explains why each User Story is needed by the requirements for this assignment. (2%)
- Task 7.2: **Screencast2**: (5%)
  - The screencast is named properly, uses one of the required formats, the textual materials shown in the screencast must be readable, the audio must be clear and understandable, and the screencast is of the proper duration. (1%)
  - The screencast shows and describes each Test Case from the Test Cases.pdf. (4%)
- Task 7.3: **Screencast3**: (10%)
  - The screencast is named properly, uses one of the required formats, the textual materials shown in the screencast must be readable, the audio must be clear and understandable, and the screencast is of the proper duration. (2%)
  - Show that the set of test cases in the code covers all the tests in the Test Cases.pdf with no unrelated test cases. (3%)
  - Explain how each of the test cases satisfies the requirements from Task 3. (5%)
- Task 7.4: **Screencast4**: (10%)
  - The screencast is named properly, uses one of the required formats, the textual materials shown in the screencast must be readable, the audio must be clear and understandable, and the screencast is of the proper duration. (2%)
  - Show and explain the code written to implement the CRUD functionality. (4%)
  - Show and explain that the new code and internal documentation align with the provided FoundationsF25 code. (4%)
- Task 7.5: **Screencast5**: (10%)
  - The screencast is named properly, uses one of the required formats, the textual materials shown in the screencast must be readable, the audio must be clear and understandable, and the screencast is of the proper duration. (2%)
  - Show that the application properly implements all the requirements. (4%)
  - Explain how the code works for each of the requirements. (4%)
- Task 7.6: **Screencast6**: (10%)
  - The screencast is named properly, uses one of the required formats, the textual materials shown in the screencast must be readable, the audio must be clear and understandable, and the screencast is of the proper duration. (2%)
  - Show the execution of each test case from your Test Cases.pdf works are required. (4%)
  - Explain how each test case from your Test Cases.pdf works. (4%)
- Task 7.7: **Screencast7**: (10%)
  - The screencast is named properly, uses one of the required formats, the textual materials shown in the screencast must be readable, the audio must be clear and understandable, and the screencast is of the proper duration. (2%)
  - The screencast shows that Staff Epic.pdf contains an epic, a scenario for the epic, and an outline. (2%)
  - The screencast shows that the outline describes the input values, outline values, and actions required by the epic and scenario. (2%)
  - The screencast explains why each input value, output value, and action that the outline

**The Rubrics are being updated. You will be informed when they are updated.**

**Focus on the statements above about the Deliverables and not the Rubric. If there is a disagreement between the Deliverables and the Rubric, the Rubric will be updated.**

HW 2 Rubric				
Criteria	Ratings			Pts
Task 1: The HW2 archive and its contents are named properly as described.	5 pts HW2 Archive and its contents are named properly.	2.5 pts Minor errors in the naming.	0 pts Naming of the archive and its contents not according to the description.	5 pts
Task 2.1 - Create and document a subset list of the provided Student User Stories covering CRUD.	1 pts Complete CRUD coverage.	0.5 pts Some CRUD covered.	0 pts No CRUD coverage.	1 pts
Task 2.2 - Create and document a subset list of the provided Student User Stories covering input validation	3 pts User stories fully cover input validation.	1.5 pts User stories partially cover input validation.	0 pts No input validation coverage.	3 pts
Task 2.3 - The subset list material containing irrelevant data other than CRUD and input validation	1 pts Only CRUD + input validation included	0.5 pts Some irrelevant material	0 pts Mostly irrelevant material.	1 pts
Task 3.1 - Creation and documentation of a list of tests that cover all of the Task 2 stories.	1 pts The tests cover all of the Task 2 stories.	0.5 pts The tests cover some of the Task 2 stories.	0 pts The tests do not cover any of the Task2 stories.	1 pts
Task 3.2 - Types of testcases created - (Positive and Negative)	1 pts Includes both positive and negative tests.	0.5 pts Includes either positive or negative tests.	0 pts Does not include both positive and negative tests.	1 pts
Task 3.3 - Helpful error messages in negative testcases.	3 pts All negative tests generate context relevant error messages.	1.5 pts Some of the negative tests generate context relevant error messages.	0 pts None of the negative tests generate context relevant error messages.	3 pts

Task 4.1 - Created Java application code and documentation consistency with FoundationsF25	<b>2 pts</b> The created code completely consistent with theFoundationsF25code.	<b>1 pts</b> Partially consistent with FoundationsF25.	<b>0 pts</b> No consistency or not matching theFoundationsF25code.	<b>2 pts</b>
Task 4.2 - Coverage of User Stories from Task 2.	<b>2 pts</b> The created application covers all the user stories from Task 2.	<b>1 pts</b> The created application covers some of the user stories from Task 2.	<b>0 pts</b> The created application covers none of the user stories from Task 2.	<b>2 pts</b>
Task 4.3 - Inclusion of CRUD UML Class Diagrams.	<b>3 pts</b> Includes 4 useful CRUD UML Class Diagrams.	<b>1.5 pts</b> Includes at least 2CRUD UML Class Diagrams.	<b>0 pts</b> No useful CRUD UML Class Diagrams included.	<b>3 pts</b>
Task 4.4 - Inclusion of a relevant Sequence Diagram.	<b>3 pts</b> Documentation includes a relevant sequence diagram.	<b>1.5 pts</b> Documentation includes a sequence diagram that is partially relevant.	<b>0 pts</b> Documentation doesnot include a relevant sequence diagram.	<b>3 pts</b>
Task 4.5 - Consistency of the Implementation of test cases from Task 3 with TP1	<b>5 pts</b> Test cases completely consistent with TP1.	<b>2.5 pts</b> Test cases partially consistent with TP1.	<b>0 pts</b> Test cases notconsistent with TP1.	<b>5 pts</b>
Task 6.1 - Epic +Scenario Provided	<b>1 pts</b> Epic + Scenario present.	<b>0.5 pts</b> Epic + Scenario incomplete.	<b>0 pts</b> Epic + Scenario not provided.	<b>1 pts</b>
Task 6.2 - Remainder of the deliverable is in the form of an outline.	<b>1 pts</b> Only outline provided with key highlights.	<b>0.5 pts</b> The outline is some what descriptive.	<b>0 pts</b> The outline is completely descriptive giving explanations.	<b>1 pts</b>
Task 6.3 - Inclusion of epic's input values, actions performed, and output produced.	<b>2 pts</b> Inclusion of all of theepic's input values, actions performed, and output produced.	<b>1 pts</b> Inclusion of some ofthe epic's input values, actions performed, and output produced.	<b>0 pts</b> Inclusion of none ofthe epic's inputvalues, actionsperformed, andoutput produced.	<b>2 pts</b>
Task 6.4 - Explanation of how the added code fits the structure and style of the existing code.	<b>3 pts</b> Strong justification provided.	<b>1.5 pts</b> Weak justification provided.	<b>0 pts</b> No justification provided.	<b>3 pts</b>
Task 6.5 - Explanation on why each value and operation is required by the epic.	<b>3 pts</b> Complete explanation provided.	<b>1.5 pts</b> Partial explanation provided.	<b>0 pts</b> No explanation provided.	<b>3 pts</b>
Task 7.1.1 - Screencast 1 naming, formats, readability, audio clarity and duration.	<b>1 pts</b> All of the above requirements are met.	<b>0.5 pts</b> Some of the above requirements are met.	<b>0 pts</b> None of the above requirements are met.	<b>1 pts</b>

Task 7.1.2 - Description and Depiction from UserStories.pdf.	<b>2 pts</b> All User Stories from the User Stories. pdf are shown and described.	<b>1 pts</b> Some of the User Stories from the UserStories.pdf are shown and described.	<b>0 pts</b> None of the User Stories from the UserStories.pdf are shown and described.	2 pts
Task 7.1.3 - Explanation on why each User Story is needed by the requirements for this assignment.	<b>2 pts</b> Explanation of all user stories are provided.	<b>1 pts</b> Partial explanation provided.	<b>0 pts</b> No explanation provided.	2 pts
Task 7.2.1 - Screencast 2 naming, formats, readability, audio clarity and duration.	<b>1 pts</b> All the above requirements are met.	<b>0.5 pts</b> Some of the above requirements are met.	<b>0 pts</b> None of the above requirements are met.	1 pts
Task 7.2.2 - Description and Depiction of each test case from the Test Cases.pdf.	<b>4 pts</b> All test cases from the Test Cases. pdf are shown and described.	<b>2 pts</b> Some of the testcases from the TestCases.pdf are shown and described.	<b>0 pts</b> None of the testcases from the TestCases.pdf are shown and described.	4 pts
Task 7.3.1 - Screencast 3 naming, formats, readability, audio clarity and duration.	<b>2 pts</b> All of the above requirements are met.	<b>1 pts</b> Some of the above requirements are met.	<b>0 pts</b> None of the above requirements are met.	2 pts
Task 7.3.2 - Code coverage of all tests in the Test Cases.pdf.	<b>3 pts</b> Showing that All tests in TestCases.pdf are covered in the code.	<b>1.5 pts</b> Showing that Some of the tests in TestCases.pdf are covered in the code.	<b>0 pts</b> Showing that None of the tests in TestCases.pdf are covered in the code.	3 pts
Task 7.3.3 - Explanation on how each test case satisfies the requirements from Task 3.	<b>5 pts</b> Detailed explanation provided.	<b>2.5 pts</b> Partial explanation provided.	<b>0 pts</b> No explanation provided.	5 pts
Task 7.4.1 - Screencast 4 naming, formats, readability, audio clarity and duration.	<b>2 pts</b> All of the above requirements are met.	<b>1 pts</b> Some of the above requirements are met.	<b>0 pts</b> None of the above requirements are met.	2 pts
Task 7.4.2 - Depiction and explanation of CRUD functionality implementation in code.	<b>4 pts</b> Detailed depiction and explanation provided for CRUD functionality in code.	<b>2 pts</b> Partial depiction and explanation provided for CRUD functionality in code	<b>0 pts</b> No depiction and explanation provided for CRUD functionality in code	4 pts
Task 7.4.3 - Depiction and explanation of the new code and internal documentation aligning with provided FoundationsF25 code.	<b>4 pts</b> Detailed depiction and explanation provided.	<b>2 pts</b> Partial depiction and explanation provided.	<b>0 pts</b> No depiction or explanation provided.	4 pts
Task 7.5.1 - Screencast 5 naming, formats, readability, audio clarity and duration.				2 pts



	<b>2 pts</b> All of the above requirements are met.	<b>1 pts</b> Some of the above requirements are met.	<b>0 pts</b> None of the above requirements are met.	
Task 7.5.2 -Implementation of requirements by the application.	<b>4 pts</b> The application properly implements all the requirements.	<b>2 pts</b> The application properly implements some of the requirements.	<b>0 pts</b> The application doesnot implement any of the requirements.	4 pts
Task 7.5.3 - Explanation of how the code works on requirements.	<b>4 pts</b> Explaining how codeworks for all the requirements.	<b>2 pts</b> Some requirements missing in the explanation.	<b>0 pts</b> No requirements explained.	4 pts
Task 7.6.1 - Screencast 6 naming, formats, readability, audio clarity and duration.	<b>2 pts</b> All of the above requirements are met.	<b>1 pts</b> Some of the above requirements are met.	<b>0 pts</b> None of the above requirements are met.	2 pts
Task 7.6.2 - Depiction of the execution of testcases from your TestCases.pdf.	<b>4 pts</b> Showed the execution of all testcases from TestCases.pdf.	<b>2 pts</b> Showed the execution of some of the test cases from Test Cases.pdf.	<b>0 pts</b> Showed the execution of none of the test cases from Test Cases.pdf.	4 pts
Task 7.6.3 - Explanation of how each test case from your Test Cases.pdf works	<b>4 pts</b> Complete Explanation for all test cases provided	<b>2 pts</b> Explanation for some test cases provided	<b>0 pts</b> No explanation provided.	4 pts
Task 7.7.1 - Screencast 7 naming, formats, readability, audio clarity and duration.	<b>2 pts</b> All of the above requirements are met.	<b>1 pts</b> Some of the above requirements are met.	<b>0 pts</b> None of the above requirements are met.	2 pts
Task 7.7.2 - Staff Epic.pdf containing an epic, scenario for the epic and an outline.	<b>2 pts</b> Showing that StaffEpic.pdf contains all of the requirements.	<b>1 pts</b> Showing that StaffEpic.pdf contains some of the requirements.	<b>0 pts</b> No requirements shown.	2 pts
Task 7.7.3 - Screencast 7 showing the outline coverage of input values, output values and actions required by the epic and scenario.	<b>2 pts</b> Shows that outline covers all of the inputs, outputs, and actions.	<b>1 pts</b> Shows that outline covers some of the inputs, outputs, and actions.	<b>0 pts</b> Outline does not cover any of the inputs, outputs, and actions.	2 pts
Task 7.7.4 - Screencast Explanation of each input value, output value and action.	<b>4 pts</b> Detailed explanation of inputs, outputs and actions provided.	<b>2 pts</b> Some of the above requirements are met. Partial explanation of inputs, outputs and actions provided.	<b>0 pts</b> No explanation provided.	4 pts
Bonus				0 pts

	0 pts Full Marks	0 pts No Marks	
			Total Points: 100