Individual Homework 4

Start Assignment

- Due Apr 9 by 11:59pm
- Points 100
- Submitting a file upload
- File Types pdf
- Available Jan 13 at 12am - Apr 11 at 11:59pm

Introduction

Using the provided materials and the team's TP3 submission, each student must:

- Use the provided materials, the student's HW3 submission, and/or the team's TP3 submission to discuss with the rest of the team about expanding the staff role epic into a set of User Stories consistent with the epic and the other User Stories. **Do not share** actual User Stories! Based on the ideas from the discussion, produce a set of User Stores that cover the staff role epic and are consistent with the provided User Stories.
- Use the User Stories you just created to design, document, create, and JUnit test the CRUD (create, read, update, and delete) functionalities in an application called HW4.
- Produce/update internal documentation, generate Javadoc output like the TP3 submission, and produce a PDF of the generated Javadoc output.

With the requirements satisfied, you must create a short screencast that shows that HW4 performs each requirement and successfully performs and tests the implementation of the User Stories. Use the provided MS Word template, fill it out, and submit it as a PDF where links can be copied and/or clicked to access your screencast and code within your personal GitHub. The detailed requirements for this homework are specified below.

Tasks

This homework requires you to perform the following tasks.

1. Fill in the cover page of the template as required.
2. Collaboratively with your teammates, discuss ideas about the staff role epic, as described above, and capture these ideas in a written form.
3. Produce a coherent set of staff role User Stories building on the ideas from Task 1.
4. Design, create, test, and **document** the staff role user stories created in Task 2 and name it HW4. The documentation must be equivalent to the team's TP3 submission. HW4 must be tested to show that it functions as required.
5. Produce a screencast that covers the following:
    - the implementation of the Task 2 User Stories and how they work

    - the implementation of JUnit tests and how they work
    - the results of running the JUnit tests and the output produced using Javadoc
6. Store your HW4 application in your personal GitHub repository. Make sure that only you and your grader have access to your repository.
7. Store your screencast in your personal GitHub repository. Make sure that only you and your grader have access to your repository.
8. Produce a PDF that contains
    - the resulting ideas from the Task 1 discussion.
    - the set of User Stories from Task 2.
    - the professional-looking Javadoc output from your HW3 application and the JUnit tests described in Task 3.
    - Provide links that can be copied and/or clicked to access your HW4 code and screencast from within your GitHub repository.
9. Submit this PDF 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**!

Deliverables

A PDF document must be produced that covers the following items. The links in the PDF must be copyable or clickable to access the GitHub copies of the code and screencast.

- Use the following Template for your PDF submission. Fill in this template with the results of the following tasks and submit it. (5%)

HW3-Submission-Template.docx

- Task 1: Cover page complete with your name. (5%)
- Task 2: A list of the staff role epic ideas you captured. (5%)
- Task 3: A list of the staff role User Stories you produced. (10%)
- Task 4: The User Stories have been implemented, documented, and functions as required. The application is named HW4. (25%)
    - HW4 has internal and Javadoc documentation that is equivalent to that in TP3. (10%)
    - The JUnit tests have been tested, and the output shows HW4 works as required. (15%)
- Task 5: A screencast has been produced that shows and explains the following. (30%)
    - the implementation of the Task 3 User Stories and how they work (15%)
    - the implementation of JUnit tests and how they work (5%)
    - the results of running the JUnit tests and the output produced using Javadoc (5%)
    - The code is readable, and the explanations are clear and audible. (5%)
- Task 6: Store your HW3 in your personal GitHub Repository and provide a usable link in the PDF. (5%)
- Task 7: Store your HW3 Screencast in your personal GitHub Repository and provide a usable link in the PDF. (5%)

- The grader must be able to examine the code in your GitHub, see that it is nicely formatted with internal documentation, and determine that it is consistent with the provided code so most people would assume it had been written by the same author. (10%)