| **Nailah Berube, Zainab Sirajo, Tomaney Hendricks, Nick Bourne, Ayoub Osman** | |
| --- | --- |
| **Instructions** |  |
| Use your sequence diagrams, functional requirements and user stories to generate 8 test cases for your application. Remember Chapter 8 is full of examples of different types of testing. Test cases are written and they demonstrate some feature and must pass or fail. They are simple atomic operations directly related to features. Page 234 gives an example of what goes inside an automated test, and page 83 gives you a written out example of a test case as well. Put a copy of the file in this dropbox and leave a link to the test cases in your GitHub as well. | |
|  |  |
| Due on Mar 8, 2024 11:59 PM  Test 1: Create  Input:   * Enter the task that must be completed * Set priority to high, medium, or low   Test:   * Test to verify whether the user created a task * Test to verify if task priority has been set   Output:   * “Task has been set: Priority (default)” * Error: Task has not been set. Please re-enter the task.   Test 2: Read  Input:   * User inputs the position ID of the note to be retrieved. * The position ID should correspond with a valid note in the database   Test:   * Test to retrieve the note in the given position * Test displays the selected note contents   Output:   * Outputs the notes contents to the user * Outputs an error message if there are no contents in the selected position id.   Test 3: Edit  Input:   * Details for new tasks * Updated priority if the user wishes to do so   Test:   * Test to see if information updates after user saves task   Output:   * “Task has been updated” * Error: Unable to update task. Please try again.   Test 4: Delete  Input:   * Choose task the user would like to remove * Verify if there are contents in the selected task   Test:   * Test if a task has been chosen. * Test if there are contents in the selected task   Output:   * “Task has been deleted” * “Error: You did not select an existing task.”   Test 5: Login  Input:   * Valid email * Valid password   Test:   * Test to verify credentials have been entered correctly * Test to see if user has an existing account   Output:   * “Success. Logging in” * Error: Invalid email/password. Please try again.   Test 6: Collaboration  Input:   * An invite will be sent via email.   Test:   * Checks if email exists in the database.   Output:   * Invite sent * “Error: Unable to send an invite. Retype email.”   Test 7: Due Dates  Input:   * Date w/ format mm/dd/yyyy * Date clicked on calendar   Test:   * Test to see if a valid date was entered * Test to see if date was saved to corresponding task   Output:   * “Due date added” * Error: Invalid date. Please try again.   Test 8: Password Requirements  Input:   * User types in the password they want to create   Test:   * Database checks the password to see if it passes necessary requirements   Output:   * “Password created” * “Password doesn’t meet length and/or character requirements” and the user reattempts password creation | |
| Test 9: Priority Level  Input:   * User inputs priority level (low, medium , high)   Test:   * Test whether the user has chosen a priority level for the task.   Output: | |
| * The priority for this task is “(low , medium , high)” * There is no selected priority for the current task. | |
|  | |