**ISYS3413 – Assignment 4**

**Section 1: Test Cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Cases** | **Test Data** | **Expected Result** | **Test Result** | **Pass/Fail** |
| testAddValidPost() | Post post | True | True | Pass |
| testInvalidPostTitle() | String postTitle = “Short” | False | False | Pass |
| testInvalidPostBody-Length() | String postBody = “Short body” | False | False | Pass |
| testInvalidPostType() | String postType = “Invalid Type” | False | False | Pass |
| testInvalidPostTagsEasy() | String postType = Easy  String postTags[] = {"tag1", "tag2", "tag3", "tag4"} | False | False | Pass |
| testInvalidPostBody-ForDifficult() | String postType = “Very Difficult”  String postBody = “Short body” | False | False | Pass |
| testInvalidPostEmergency-StatusForEasy() | String postType = “Easy”  String postEmergencyStatus = “Immediately Needed” | False | False | Pass |
| testInvalidPostEmergency-StatusForDifficultTypes() | String postType = “Very Difficult”  String postEmergencyStatus = “Ordinary” | False | False | Pass |
| testAddValidComment() | String comment = “This is a valid comment.” | True | True | Pass |
| testInvalidComment-Length() | String shortComment = “Too short”  String longComment = “This coment actually has way too many words to be valid.” | False | False | Pass |
| testInvalidCommentFirst-Character() | String comment = “this is an invalid comment.” | False | False | Pass |
| testTooManyComments() | 5x post.addComment(“This is a valid comment”), 1x string “This should fail.” | False | False | Pass |
| testTooManyComments-ForEasyOrdinary | String postType = “Easy”  String postEmergencyStatus = “Ordinary”  3x comments added – “This is a valid comment.”  1x comment added “This should fail.” | False | False | Pass |
| testInvalidComment-SpecialCharacterFirst-Character | String comment = “@Incalid comment with special character.” | False | False | Pass |

**Section 2: User Stories**

**User Story 1: Create New Post**

* As a user, I want to create a new post so that I can share information and seek help.

**Acceptance Criteria:**

* **Post creation:** a post can be created with a title, body, tags, type, and emergency status.
* **Validation:** the system validates the post title, body, length, type, tags, and emergency status before allowing the post to be created.
* **File writing:** upon successful creation, the post details are written to ‘post.txt’

**User Story 2: Add Comments to a Post**

* As a user, I want to add comments to a post so that I can engage in discussions, help others, and provide feedback.

**Acceptance Criteria:**

* **Comment Creation**: A comment can be added to a post with the appropriate length and first character validation.
* **Validation**: The system ensures comments meet the word count and first character requirements before adding them.
* **File Writing**: Upon successful addition, the comment details are written to comment.txt.

**User Story 3: View All Posts**

* As a user, I want to view all posts so that I can read existing information and responses.

**Acceptance Criteria:**

* **Display Posts**: The system retrieves and displays all posts from post.txt.
* **Post Details**: Each post displayed includes the title, body, tags, type, and emergency status.
* **Sorting**: Posts can be sorted by different criteria such as date created, type, or emergency status.

**User Story 4: Edit an Existing Post**

* As a user, I want to edit an existing post so that I can update or correct information.

**Acceptance Criteria:**

* **Post Selection**: The system allows the user to select a post to edit from the list of posts.
* **Validation**: The system validates the updated post details (title, body, tags, type, and emergency status) before saving changes.
* **File Update**: Upon successful validation, the updated post details are written back to post.txt.

**User Story 5: Delete a Post**

* As a user, I want to delete a post so that I can remove outdated or irrelevant information.

**Acceptance Criteria:**

* **Post Selection**: The system allows the user to select a post to delete from the list of posts.
* **Confirmation**: The system prompts the user for confirmation before deleting the post.
* **File Update**: Upon confirmation, the post is removed from post.txt and the file is updated accordingly.