**Task 1: Requirements Analysis and Coverage Description: Please propose the optimal test data and set of test cases required to check the functionality. Make sure that proposed approach is efficient and effective in terms of required time and effort. Please use the mockup and requirements below as input.**

**Bonuses: Identify bad requirements and to which criteria they don't satisfy.**

**Assessment Criteria: Since the task is aimed at requirements analysis, test data and test cases creation skills, it is expected that you will come up with reasonable assumptions, test cases format and proper coverage of requirements with tests.**

**Task 1 Input: Overview The web-form below is used for searching of available workplaces in an organization. Each workplace has the following attributes: date, floor, list of equipment and smoking restrictions. These attributes are also used as search criteria. Mock-up of the search form is provided below.**

**Assumptions:**

1. The system is a web application.
2. The date format follows a standard format (e.g., DD/MM/YYYY).
3. The system validates data on both the client and server sides.
4. The search result table is paginated, and each page displays a limited number of results.
5. The sorting functionality is available for each column in ascending and descending orders.
6. The equipment selection is multiple-choice, allowing users to select one or more options.
7. The system provides appropriate error messages for invalid input.
8. The system's backend and frontend communicate efficiently for search and display operations.
9. The system is accessible from various browsers and devices.

**Test Data:**

**Positive Test Cases:**

1. Search with all parameters filled (date, floor, equipment, smoking).
2. Search with the minimum required parameters.
3. Search with different valid date ranges.
4. Search with each equipment type individually.
5. Search with smoking allowed and not allowed.
6. Search and sort by each available field.

**Negative Test Cases:**

1. Attempt to search with no search parameters.
2. Attempt to search with a date range in the past.
3. Attempt to search with invalid date format.
4. Attempt to search with invalid floor (non-numeric, non-existent).
5. Attempt to search with no equipment selected.
6. Attempt to search with an invalid combination of parameters.
7. Attempt to sort by a field that is not visible in the search result.
8. Attempt to navigate to a non-existent page in the search results.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case** | **Date** | **Floor** | **Equipment** | **Smoking Allowed** | **Comments** |
| TC1 | 1/2/2024 | 1 | PC, Phone | Yes | Valid search with a specific parameters |
| TC2 | 15/03/2024 | 5 | Flip chart Board | No | Valid search with a specific parameters |
| TC3 | 1/12/2024 | 1 |  | No | Valid search with a minimum parameters |
| TC4 | 15/02/2025 | -1 | PC, Phone | Yes | Invalid floor negative value |
| TC5 | 10/3/2025 | abc | Projector | No | Invalid floor (Non-numeric value) |
| TC6 | 10/5/2025 | 4 | PC, Projector | No | Valid search with smoking not allowed |

**Test cases**

I have created Test cases in Excel file for that please refer attached Web\_form\_TCD\_v1.0 Document.

**Bad Requirements**

**Requirement 1: "System shall provide search results without delay."**

**Reason:** This requirement lacks specificity and quantify ability. The term "without delay" is subjective and does not define a measurable performance target. It does not provide information on what constitutes an acceptable or unacceptable delay. Additionally, it does not consider factors such as system load, network latency, or the size of the data set being searched.

**Improved Requirement:** "The system shall retrieve and display search results within 3 seconds of the user initiating a search. This performance target includes the time taken for processing, querying the database, and rendering the results on the user interface. The response time shall be measured under typical operating conditions and a simulated maximum load to ensure optimal performance."

**Explanation:** The improved requirement introduces a measurable performance target (3 seconds) and provides clarity on what aspects of the search process contribute to the response time. It considers factors such as processing, database querying, and user interface rendering, offering a more comprehensive and actionable guideline for development and testing teams.

**Requirement2: "System shall validate provided data before search."**

**Reason:** This requirement lacks specificity and clarity regarding the nature and extent of data validation. It does not define what types of data validation should be performed, whether it includes client-side or server-side validation, and what actions should be taken in case of validation failure. Additionally, it does not specify the criteria or rules for data validation.

**Improved Requirement:** "The system shall perform server-side validation on all user-inputted data before initiating a search. Validation checks shall include ensuring that dates are in the correct format, floor numbers are valid, and equipment selections are within the predefined list. In case of validation failure, the system shall provide clear error messages indicating the specific issues and guiding the user on corrective actions."

**Explanation:** The improved requirement provides clear guidance on the type of validation (server-side), specific data elements to be validated, and the expected behavior in case of validation failure. It enhances clarity and ensures that the development and testing teams have a precise understanding of the validation expectations.

**Solution Summary:**

The proposed test cases cover a range of scenarios, including both positive and negative test cases, to ensure comprehensive testing of the system's functionality. By addressing assumptions, creating detailed test cases, and identifying and clarifying ambiguous requirements, the testing process becomes more efficient and effective.

Testers should execute these test cases in a systematic manner, documenting the actual results and comparing them against the expected results. Any discrepancies should be reported to the development team for resolution. Additionally, the identified bad requirements should be communicated to stakeholders for clarification and refinement.

By following this approach, the testing process will contribute to the overall quality of the system, helping to identify and address potential issues before the software is deployed to production.

**Task 2 – Estimation**

**Description: Please provide rough estimate with WBS for test activities planned in the previous task. Provide information about what you suggest to include and what the customer gets as a result of planned activities.**

**Bonuses: Make a high-level description of how estimated test activities sum up in a process**

**Assessment Criteria: The pair of current and previous tasks is aimed to evaluate skills on strategy elaboration and estimation, you should enlist all the assumptions used in planning/estimation and provide clear vision on what level of quality is expected as a result of proposed activities Software Testing Evaluation Task**

**Task 2 Input: Increment Description**

**The new increment of an application described in previous task will contain many new features that will extend already existing functionality. The following features are added to the project plan for a new version:**

**- User registration**

**- Booking of the office place**

**- User roles (user and office manager roles)**

**- Native mobile application**

**Task 2 Input: User Stories**

**The following set of user stories were received from the Customer as description of planned features:**

**1. As a user, I want to book available working places for future so that nobody could take the place.**

**2. As a user, I want to have ability to cancel booking so that another user could book it instead.**

**3. As a user, I want to register into the system so that I can view working places I have booked.**

**4. As a user, I want to use mobile application for system usage so that I can do it any time.**

**5. As an office manager, I want to manage working places so that system provide valid up-to-date information to system users.**

**6. As an office manager, I want to create new working places so that users could have ability to find and book them.**

**Task 2(a) - Work Breakdown Structure (WBS) and Estimation:**

**Assumptions:**

1. The team has a basic understanding of the application architecture.
2. The team is familiar with the technologies involved in user registration, office place booking, user roles, and mobile application development.
3. The development of the new increment is concurrent with the testing activities.
4. The team has access to the necessary testing environments and test data.
5. The testing team is sufficiently staffed with skilled resources.
6. The customer's requirements are well-documented and stable.

**Work Breakdown Structure (WBS):**

**Test Planning and Preparation (10%):**

* Review requirements and user stories.
* Develop the test strategy.
* Identify testing resources and tools needed.
* Create the test plan.

**Test Design (20%):**

* Develop detailed test cases based on user stories and requirements.
* Design test data for user registration, office place booking, user roles, and mobile application testing.

**Test Environment Setup (10%):**

* Set up testing environments for various testing phases.
* Configure test data for user roles, bookings, and mobile application scenarios.

**Test Execution (30%):**

* Execute test cases for user registration.
* Execute test cases for office place booking.
* Execute test cases for user roles and permissions.
* Execute test cases for native mobile application.

**Defect Reporting and Retesting (15%):**

* Log and prioritize defects.
* Verify and validate fixes.
* Retest affected areas.

**Documentation (10%):**

* Document test results, including pass/fail status.
* Update test cases based on feedback.
* Generate test summary reports.

**Estimated Effort:**

Considering the complexity of the new features and the need for comprehensive testing, the estimated effort for testing the new increment is approximately 85 person-days.

**Description for Estimated Effort:**

**Reviewing Increment Description: (2 person-days)**

* Understanding the new features added to the application increment.
* Identifying testing challenges associated with user registration, office place booking, user roles, and mobile application.

**Creating Test Strategy and Plan: (5 person-days)**

* Developing a comprehensive test strategy considering the new features.
* Creating a detailed test plan outlining testing phases, entry/exit criteria, and resource requirements.

**Test Case Design: (15 person-days)**

* Developing detailed test cases based on user stories and requirements.
* Designing test data for user registration, office place booking, user roles, and mobile application testing.

**Test Environment Setup: (8 person-days)**

* Setting up testing environments for various testing phases.
* Configuring test data for user roles, bookings, and mobile application scenarios.

**Test Execution: (35 person-days)**

* Executing test cases for user registration.
* Executing test cases for office place booking.
* Executing test cases for user roles and permissions.
* Executing test cases for the native mobile application.

**Defect Reporting and Retesting: (15 person-days)**

* Logging and prioritizing defects discovered during testing.
* Verifying and validating fixes.
* Retesting affected areas to ensure the stability of the application.

**Documentation: (10 person-days)**

* Documenting test results, including pass/fail status.
* Updating test cases based on feedback.
* Generating test summary reports.

**Contingency/Buffer: (5 person-days)**

Allocating a buffer for unforeseen challenges, additional testing requirements, or unexpected issues that may arise during the testing process.

This estimation is a rough breakdown, and the actual effort may vary based on the project's specific requirements, the complexity of the features, the team's expertise, and the efficiency of the testing process. It's crucial to involve the testing team, consider historical data, and continuously refine the estimates as the project progresses.

**Task 2(b) - User Stories Estimation:**

**User Story 1: Booking Available Working Places (15%):**

* Test the functionality of booking places for future dates.
* Verify that booked places are reserved and unavailable for other users.

**User Story 2: Ability to Cancel Booking (10%):**

* Test the cancelation process.
* Verify that canceled bookings free up the working place for other users.

**User Story 3: User Registration (20%):**

* Test user registration functionality.
* Verify that registered users can log in and view their booked places.

**User Story 4: Mobile Application Usage (20%):**

* Test the functionality of the native mobile application.
* Verify that users can perform all necessary actions using the mobile app.

**User Story 5: Office Manager - Manage Working Places (10%):**

* Test the functionality of managing working places.
* Verify that changes made by the office manager reflect in the system.

**User Story 6: Office Manager - Create New Working Places (20%):**

* Test the creation of new working places by the office manager.
* Verify that newly created places are searchable and bookable.

**Result and Quality Expectation:**

The estimated testing activities aim to ensure that the new increment meets the specified requirements and user stories. The goal is to deliver a high-quality product that is free from critical defects, performs well, and provides a seamless experience for both users and office managers. The testing process aims to identify and address any issues related to user registration, office place booking, user roles, and mobile application usage, contributing to the overall reliability and usability of the application.