**OceanWatch : Empowering Ocean Safety and GPS Tracking**

**Project Description:**

OceanWatch is a comprehensive tracking and safety software designed for fishermen operating in tropical areas. With the aim of enhancing the safety of fishermen venturing into the deep sea, OceanWatch provides real-time tracking and communication capabilities, ensuring that fishermen can be located promptly in case of emergencies. The software makes possible smooth coordination among fishermen, rescue teams, weather forecasters, and administrators, thereby minimizing the risks associated with fishing in remote and hazardous environments.

The system's core classes include Fisherman, Boat, Rescue Team, and Weather Forecast. Fisherman and Boat details, along with tracking device specifics, contribute to a robust database that powers the user interface. Users, including rescue teams, can effortlessly access real-time and historical locations, boosting situational awareness. For search and rescue operations, the software offers precise last-known positions, aiding teams in planning and executing targeted missions effectively. In summary, OceanWatch signifies a monumental leap in maritime safety, employing technology to safeguard lives in challenging deep-sea environments. By seamlessly integrating GPS tracking, real-time data processing, and proactive alerts, , OceanWatch transforms every deep Sea journey into a secure and monitored experience.

**Problem Statement:**

Dhaka, July 22 (bdnews24.com) — Over 3,500 people died and around 11,000 others went missing in 2,400 incidents of fishing trawler sinking in the coastal areas last 20 years, according to information revealed at a seminar. Green Club of Bangladesh organized the seminar in the city on Friday on the accidents in the sea and coastal areas. Fishing in tropical areas poses significant risks to fishermen, particularly when navigating the vast expanse of the deep sea. Incidents of fishermen going missing or encountering emergencies, such as adverse weather conditions or equipment failures, are unfortunately not uncommon. These incidents often result in tragic outcomes, with fishermen lost at sea and rescue efforts hindered by the lack of timely information on their whereabouts. The absence of an efficient tracking and communication system exacerbates the challenges faced by rescue teams in locating and rescuing stranded fishermen promptly.

**Proposed Solution :**

OceanWatch proposes a robust tracking and safety solution to address the challenges faced by fishermen operating in tropical areas. The software comprises several key features aimed at enhancing the safety and well-being of fishermen:

*Real-Time Tracking:* OceanWatch utilizes GPS technology to track the location of fishing vessels in real-time. Fishermen can be monitored continuously, allowing administrators and rescue teams to pinpoint their exact whereabouts at any given time.

*Emergency Alert System:* In the event of an emergency, fishermen can activate an SOS alert through the OceanWatch software. This alert triggers an immediate response from nearby rescue teams, who can mobilize quickly to assist the distressed fishermen.

*Weather Forecast Integration:* OceanWatch integrates real-time weather forecasts into its platform, providing fishermen with up-to-date information on changing weather patterns. Fishermen can make informed decisions regarding their fishing activities, minimizing the risk of encountering hazardous weather conditions.

*Two-Way Communication:* OceanWatch facilitates seamless communication between fishermen, rescue teams, and weather forecasters through a two-way messaging system. This enables fishermen to relay important information to rescue teams and receive guidance or assistance as needed.

*Administrative Dashboard:* Administrators have access to a centralized dashboard where they can monitor fishing activities, track vessels, and coordinate rescue operations efficiently. The dashboard provides valuable insights into fleet management and ensures swift responses to emergencies.

By implementing OceanWatch, fishermen can navigate the deep sea with greater confidence, knowing that they have access to a reliable tracking and safety system. The software not only enhances the safety of fishermen but also streamlines rescue operations, ultimately saving lives and preventing tragedies at sea.

**Process Model: Agile Software Development**

****

*Analysis:*

The nature and environment of the OceanWatch software project demand a flexible and iterative approach to development. Given the dynamic and evolving requirements, as well as the need for continuous feedback and adaptation, Agile Software Development emerges as the most suitable method for this project.

*Reasoning:*

Iterative Development: Agile methodologies, such as Scrum or Kanban, emphasize iterative development cycles. This aligns well with the evolving nature of the project, allowing for incremental enhancements and adjustments based on user feedback and changing requirements.

Customer Collaboration: Agile methodologies prioritize customer collaboration throughout the development process. In the case of OceanWatch, close collaboration with fishermen, rescue teams, and other stakeholders is essential to ensure that the software meets their needs effectively.

Flexibility: Agile methodologies offer flexibility in accommodating changes to requirements, priorities, and technical constraints. This is particularly beneficial for a project like OceanWatch, where the exact specifications and functionalities may evolve over time in response to user feedback and emerging challenges.

Continuous Improvement: Agile methodologies promote continuous improvement through regular reflection and adaptation. This allows the development team to address issues promptly, refine processes, and enhance the quality of the software incrementally.

Database Operations and CRUD Operations: Agile methodologies support the integration of database operations and CRUD (Create, Read, Update, Delete) operations seamlessly into the development process. Database design and implementation can be iteratively refined as part of each development sprint, ensuring that the database schema aligns with evolving requirements and supports efficient data management.

*Evidence:*

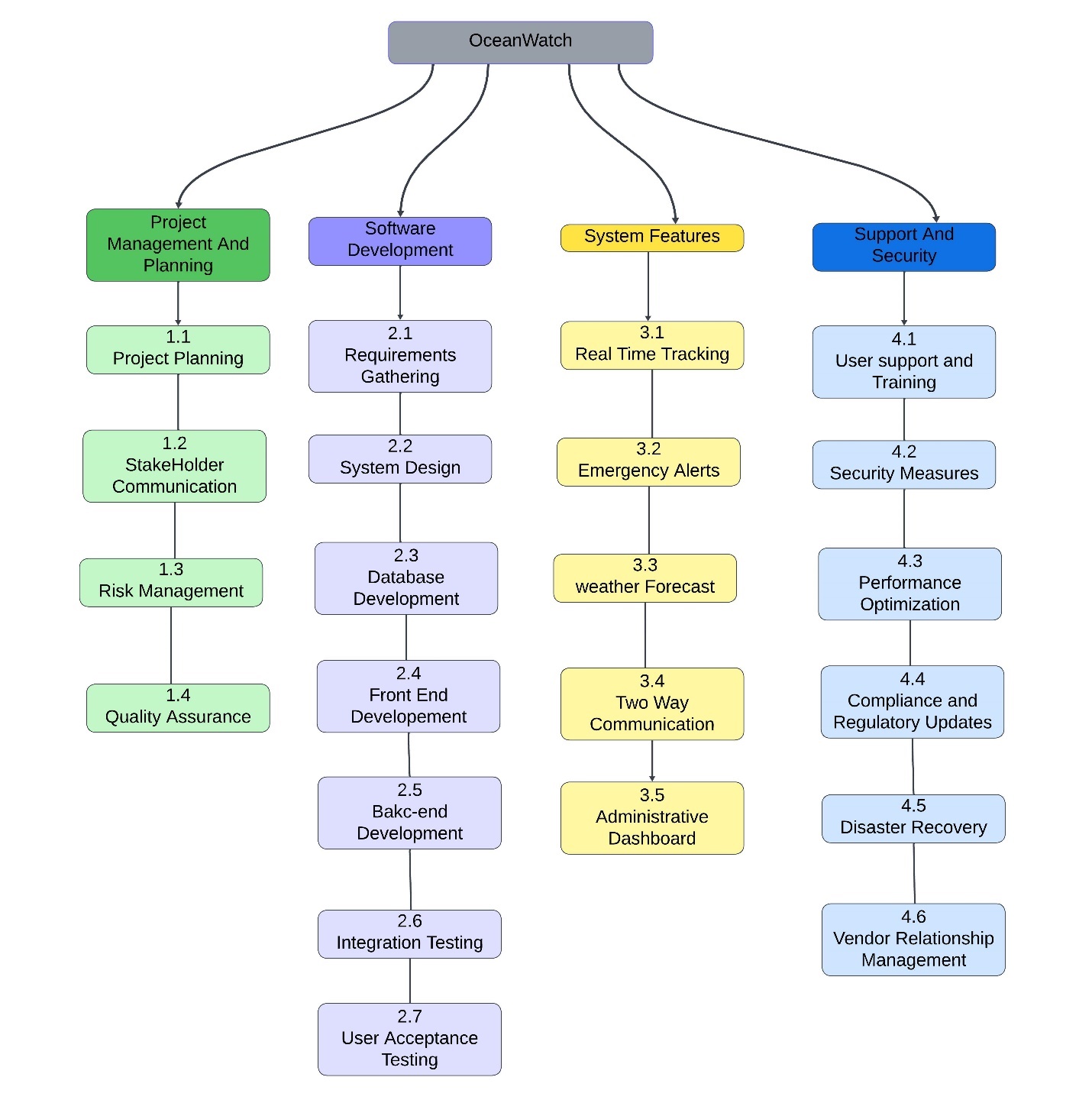
Case Studies: Numerous case studies demonstrate the successful application of Agile methodologies in projects involving database operations and CRUD operations, highlighting the adaptability and effectiveness of Agile approaches in such contexts.

Industry Best Practices: Agile Software Development has become widely adopted across various industries due to its proven effectiveness in delivering high-quality software that meets customer needs efficiently. This widespread adoption serves as evidence of the reliability and suitability of Agile methodologies for diverse software projects.

Feedback from Stakeholders: Feedback from stakeholders, including fishermen, rescue teams, and administrators, can provide valuable insights into the effectiveness of the Agile approach in addressing their needs and facilitating the development of the OceanWatch software.

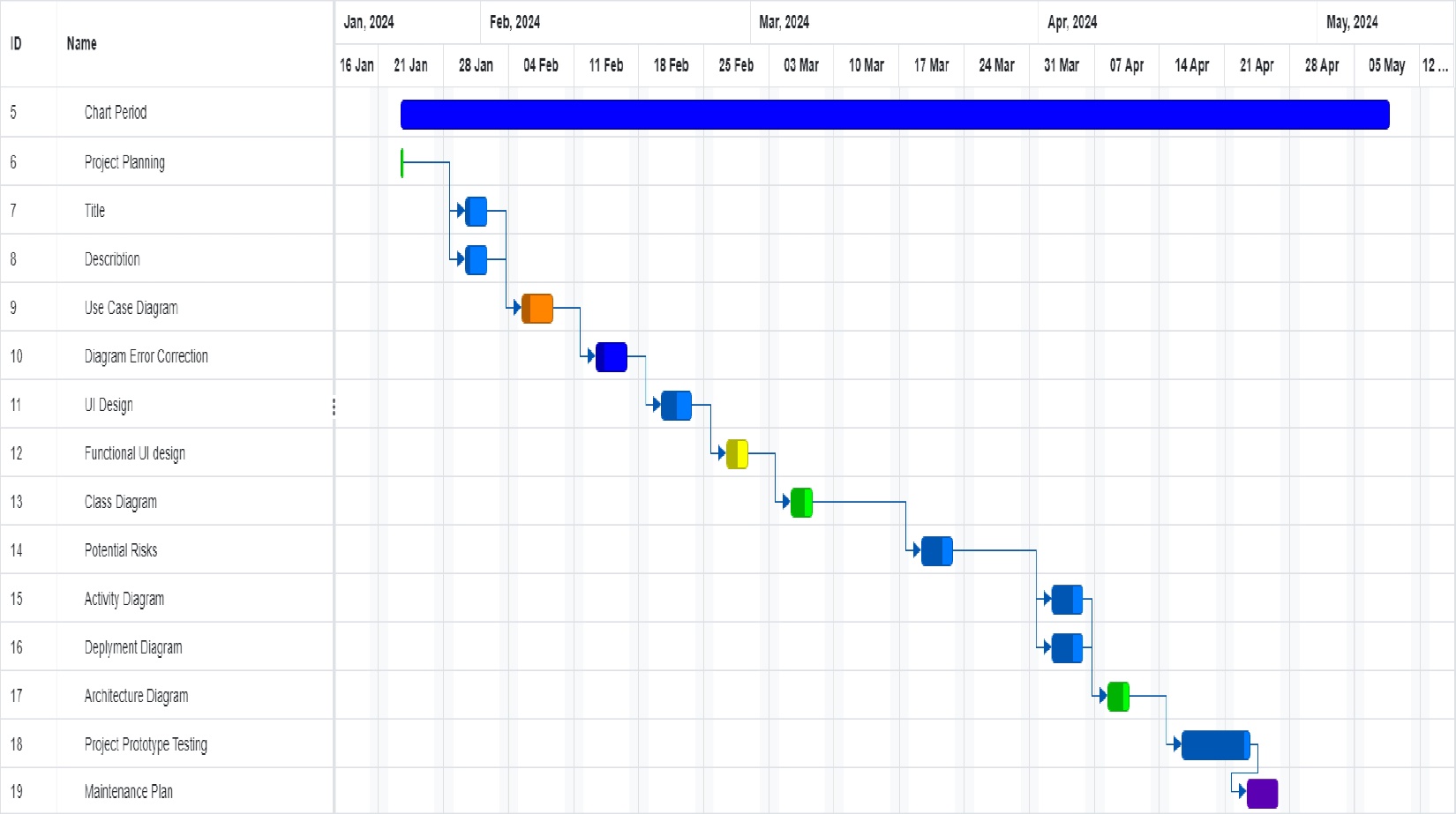
In conclusion, Agile Software Development emerges as the best-suited method for developing the OceanWatch software due to its iterative nature, emphasis on customer collaboration, flexibility, continuous improvement, and compatibility with database operations and CRUD operations. This approach will enable the development team to deliver a robust and adaptable solution that meets the evolving requirements of the project stakeholders.

**WBS(WORK BREAKDOWN STRUCTURE ) :**

****

**Figure : WBS divided into 4 major parts**

**Project Gantt Chart:**

****

**Figure : Gantt Chart (January – May)**



**Figure : Software Risk & Impact Level**

**Project Requirements:**

**A diagram of a watch

Description automatically generated**

**Figure : Use Case Diagram**

**Prototype Design:**

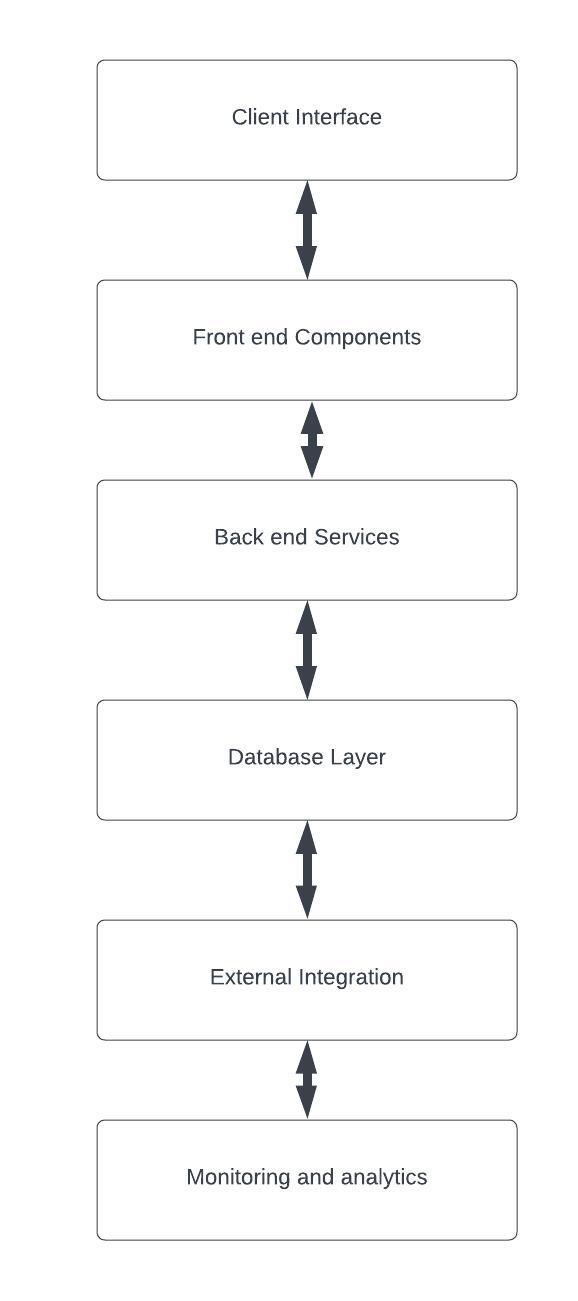
**A screenshot of a computer

Description automatically generated**

**Project UI design Playable Link:**

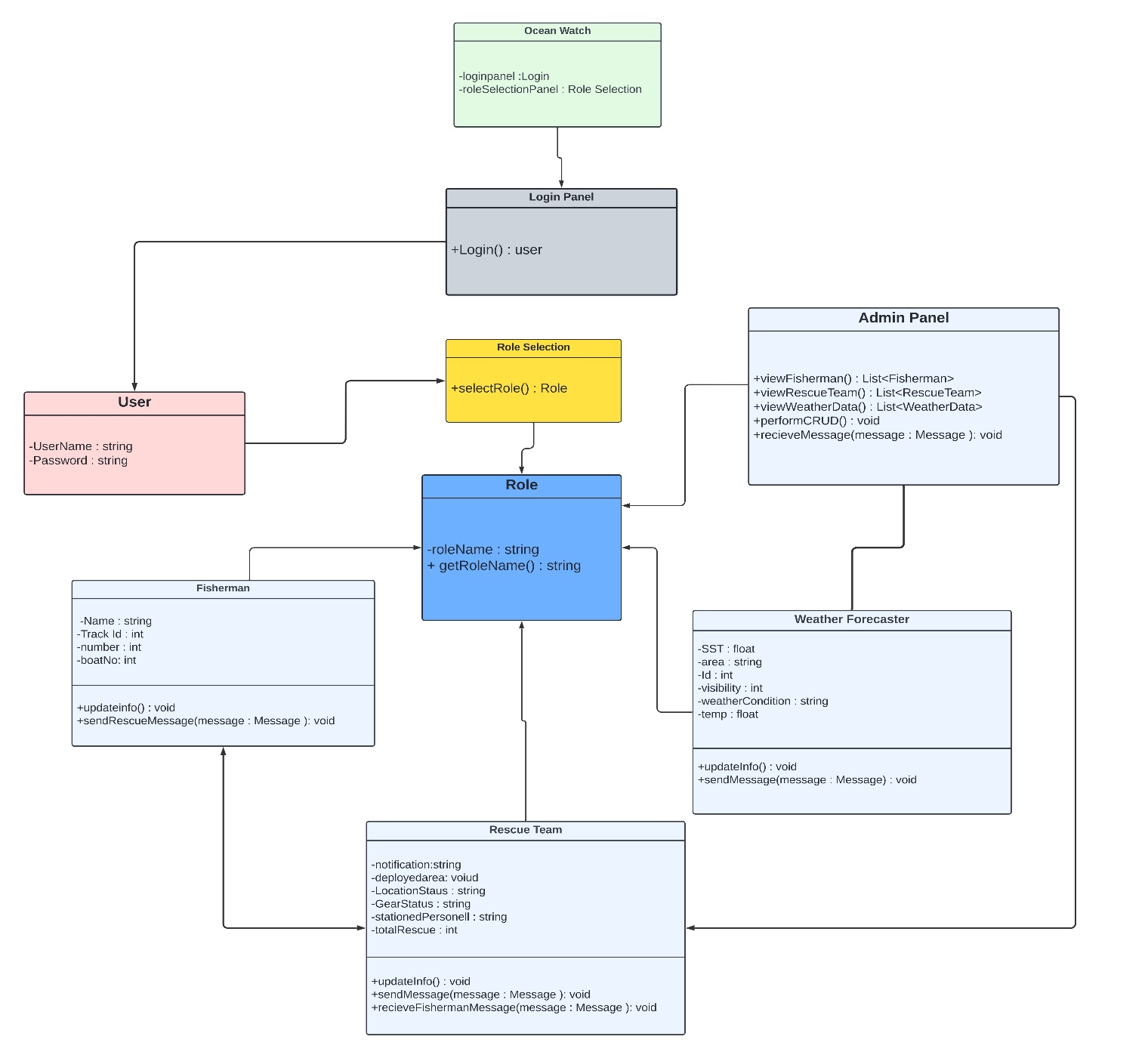
<https://www.figma.com/file/gtOF7VAtQHAyuYe8ypyUiX/OceanWatch?type=design&nodeid=0%3A1&mode=design&t=l6jSg4UrkimJauWF-1>

**Architecture Diagram :**



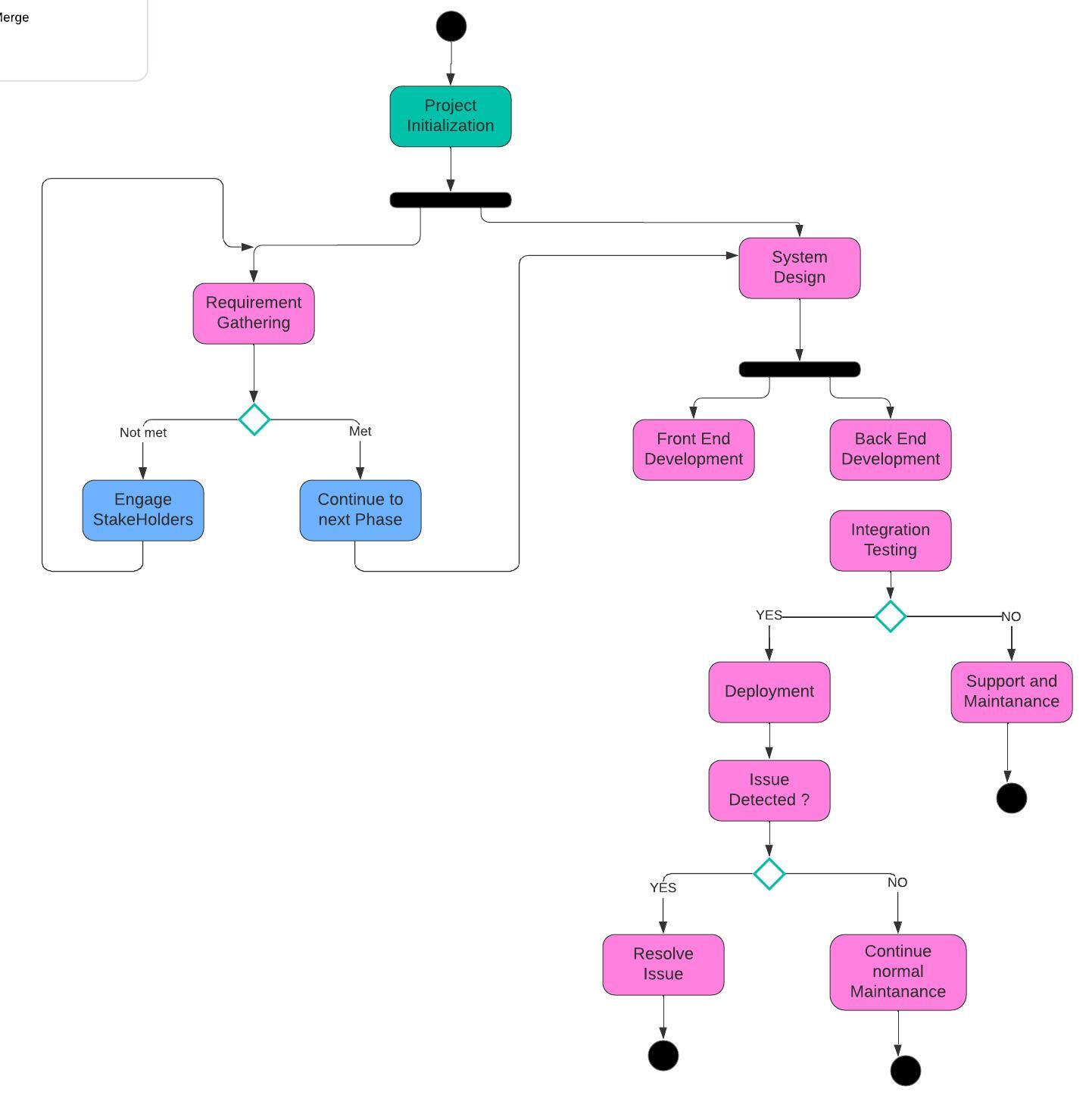
**Figure : Architecture Diagram**

**Class Diagram:**

****

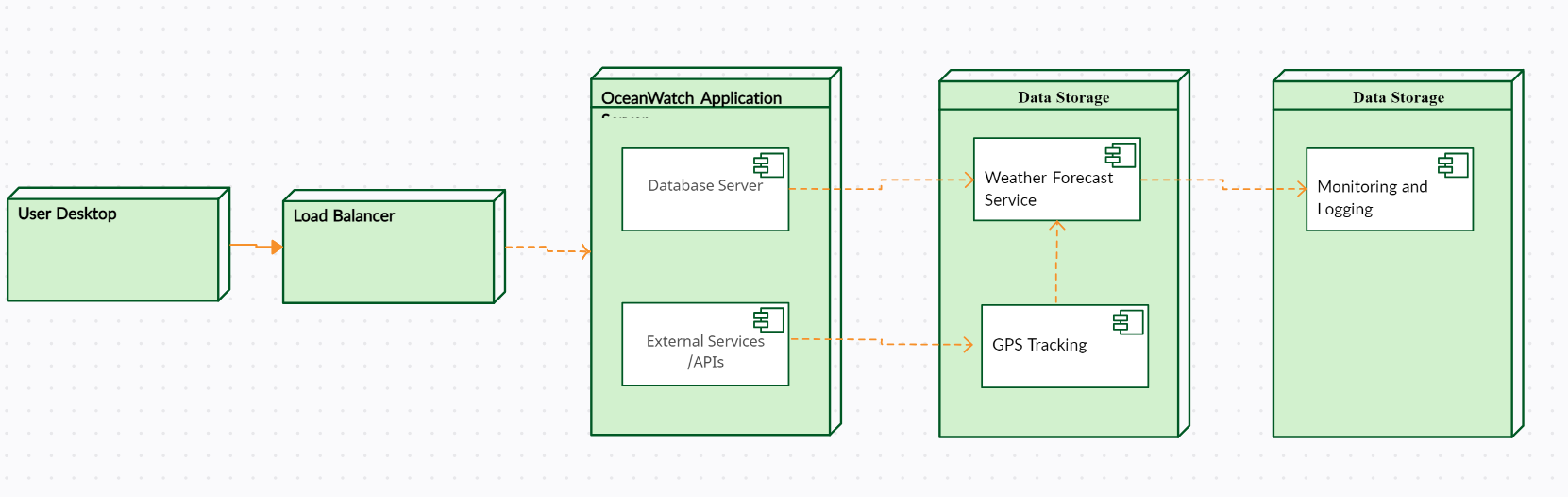
**Figure: Class Diagram**

**Activity Diagram :**

****

**Figure: Activity Diagram**

**Package Diagram :**

****

**Figure : Package Diagram**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **QA Tester’s Log** | | **Review comments from Bill incorporate in version 2.1** | | | |  |  |  |  | |  |
|  |  |  |  |  |  |  |  |  |  | |  |
| **Tester's Name** | | **Fahim** | **Date Tested** | | **16-Apr-2024** | | **Test Case (Pass/Fail/Not Executed)** | | **Pass** |  |  |
|  |  |  |  |  |  |  |  |  |  | |  |
| **S #** | **Prerequisites:** | | |  | **S #** | **Test Data** | | | |  |  |
| **1** | **User must have valid login credentials.** | | |  | **1** | **User ID: Grp7OcceanWatch** | | | |  |  |
| **2** | **User must have access to the system (not locked out).** | | |  | **2** | **Password: afnt07@** | | | |  |  |
| **3** | **Internet connectivity must be stable.** | | |  | **3** |  | | | |  |  |
| **4** | **OceanWatch system should be operational without any server downtime.** | | |  | **4** |  | | | |  |  |
|  |  |  |  |  |  |  |  |  |  | |  |
| **Test Scenario** | **Verify on entering valid userid and password, the customer can login.** | | | | | |  |  |  | |  |
|  |  |  |  |  |  |  |  |  |  | |  |
| **Step #** | **Step Details** | | **Expected Results** | | **Actual Results** | | | **Pass / Fail / Not executed / Suspended** | |  |  |
| **1** | **Open the browser.** | | **The browser should open without errors.** | | **As Expected** | | | **Pass** | |  |
| **2** | **Navigate to the OceanWatch login page.** | | **The OceanWatch login page should load successfully.** | | **As Expected** | | | **Pass** | |  |  |  |  |  |  |  |  |  |  |  |
| **3** | **Enter the provided valid User ID and Password** | | **The User ID and Password fields should be editable.** | | **As Expected** | | | **Pass** | |  |  |  |  |  |  |  |  |  |  |  |
| **4** | **Click the 'Login' button.** | | **Should redirect to the dashboard after successful login.** | | **As Expected** | | | **Pass** | |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | | TC\_0W\_002 | **Test Case Description** | | Test invalid credentials can not login | | | | | |
| **Created By** | | Fahim | **Reviewed By** | | Fahim | | **Version** | | 2.1 | |
|  |  |  |  |  |  |  |  |  |  |  |
| **QA Tester’s Log** | | Review comments from Abdullah incorporate in version 2.1 | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Tester's Name** | | Fahim | **Date Tested** | | 30 April ,24 | | **Test Case (Pass/Fail/Not Executed)** | | Pass | |
|  |  |  |  |  |  |  |  |  |  |  |
| **S #** | **Prerequisites:** | | |  | **S #** | **Test Data** | | | | |
| 1 | User must have invalid login credentials | | |  | 1 | Userid = invalid user ID | | | | |
| 2 | User must have access to the system | | |  | 2 | Pass = invalid password | | | | |
| 3 | Internet connectivity must be stable | | |  | 3 |  | | | | |
| 4 | System must not be operational with system downtime | | |  | 4 |  | | | | |
|  |  |  |  |  |  |  |  |  |  |  |
| **Test Scenario** | Verify on entering valid userid and password, the customer can login | | | | | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | **Actual Results** | | | **Pass / Fail / Not executed / Suspended** | | |
|
| 1 | Open the browser | | Site should open | | As Expected | | | Pass | | |
| 2 | Enter Invalid User id & Password | | Username and password fields should be editable | | As Expected | | | Pass | | |
| 3 | Click Login Button | | Dusplay an error message | | As Expected | | | Pass | | |
|  |  | |  | |  | | |  | | |
|  |  | |  | |  | | |  | | |
|  |  | |  | |  | | |  | | |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | | TC\_0W\_003 | **Test Case Description** | | Verify that the 'Forgot Password' link redirects to the password reset page. | |  | |  | |
| **Created By** | | Fahim | **Reviewed By** | | Fahim | | **Version** | | 2.3 | |
|  |  |  |  |  |  |  |  |  |  |  |
| **QA Tester’s Log** | | Review comments from Abdullah incorporate in version 2.3 | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Tester's Name** | | Fahim | **Date Tested** | | 30 April ,24 | | **Test Case (Pass/Fail/Not Executed)** | | Pass | |
|  |  |  |  |  |  |  |  |  |  |  |
| **S #** | **Prerequisites:** | | |  | **S #** | **Test Data** | | | | |
| 1 | User must have access to the OceanWatch login page. | | |  | 1 | <https://www.figma.com/file/gtOF7VAtQHAyuYe8ypyUiX/OceanWatch?type=design&nodeid=0%3A1&mode=design&t=l6jSg4UrkimJauWF-1> | | | | |
| 3 | OceanWatch system should be operational without any server downtime. | | |  | 2 |  | | | | |
|  |  |  |  |  |  |  |  |  |  |  |
| **Test Scenario** | Verify that clicking the 'Forgot Password' link redirects to the password reset page. | | | | | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | **Actual Results** | | | **Pass / Fail / Not executed / Suspended** | | |
|
| 1 | Open the browser | | The browser should open without errors.| | | As Expected | | | Pass | | |
| 2 | Click the 'Forgot Password' link. | | should redirect to the password reset page | | As Expected | | | Pass | | |

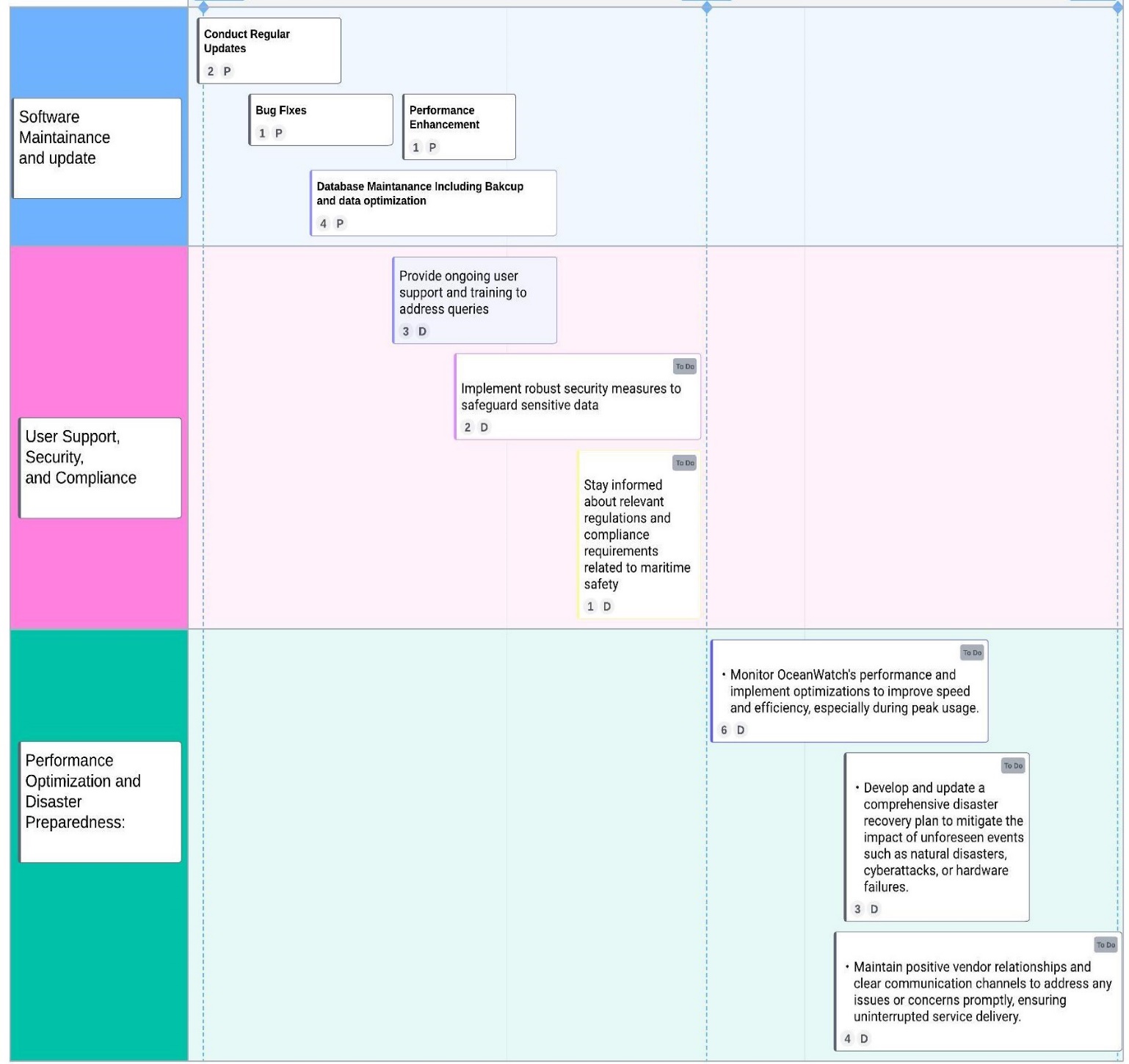
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | | TC\_0W\_004 | **Test Case Description** | | Verify that a user can successfully log out of the OceanWatch system. | | | | | |
| **Created By** | | Fahim | **Reviewed By** | | Fahim | | **Version :** | | 2.4 | |
|  |  |  |  |  |  |  |  |  |  |  |
| **QA Tester’s Log** | | Review comments from Fahim incorporate in version 2.4 | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Tester's Name** | | Fahim | **Date Tested** | | 30 April ,24 | | **Test Case (Pass/Fail/Not Executed)** | | Pass | |
|  |  |  |  |  |  |  |  |  |  |  |
| **S #** | **Prerequisites:** | | |  | **S #** | **Test Data** | | | | |
| 1 | User must be logged in to the OceanWatch system. | | |  | 1 | None | | | | |
|  |  |  |  |  |  |  |  |  |  |  |
| **Test Scenario** | Verify on entering valid userid and password, the customer can login | | | | | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | **Actual Results** | | | **Pass / Fail / Not executed / Suspended** | | |
|
| 1 | Click on the 'Logout' button. | | Should log the user out and redirect to the login page | | As Expected | | | Pass | | |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | | TC\_0W\_005 | **Test Case Description** | | Verify that the user can view historical tracking data on the dashboard. | | | | | |
| **Created By** | | Fahim | **Reviewed By** | | Fahim | | **Version :** | | 2.5 | |
|  |  |  |  |  |  |  |  |  |  |  |
| **QA Tester’s Log** | | Review comments from Fahim incorporate in version 2.5 | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Tester's Name** | | Fahim | **Date Tested** | | 30 April ,24 | | **Test Case (Pass/Fail/Not Executed)** | | Pass | |
|  |  |  |  |  |  |  |  |  |  |  |
| **S #** | **Prerequisites:** | | |  | **S #** | **Test Data** | | | | |
| 1 | User must be logged in to the OceanWatch system. | | |  | 1 | None | | | | |
| 2 | Internet connectivity must be stable. | | |  | 2 | N/A | | | | |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Test Scenario** | Verify that the user can access historical tracking data on the dashboard. | | | | | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | **Actual Results** | | | **Pass / Fail / Not executed / Suspended** | | |
|
| 1 | Navigate to the historical tracking data section on the dashboard | | Should log the user out and redirect to the login page | | As Expected | | | Pass | | |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | | TC\_0W\_006 | **Test Case Description** | | Verify that the user can search for specific fishing boats on the dashboard. | | | | | |
| **Created By** | | Fahim | **Reviewed By** | | Fahim | | **Version :** | | 2.6 | |
|  |  |  |  |  |  |  |  |  |  |  |
| **QA Tester’s Log** | | Review comments from Sarah incorporate in version 2.6 | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Tester's Name** | | Fahim | **Date Tested** | | 30 April ,24 | | **Test Case (Pass/Fail/Not Executed)** | | Pass | |
|  |  |  |  |  |  |  |  |  |  |  |
| **S #** | **Prerequisites:** | | |  | **S #** | **Test Data** | | | | |
| 1 | User must be logged in to the OceanWatch system. | | |  | 1 | Boat ID | | | | |
| 2 | Internet connectivity must be stable. | | |  | 2 | Name to search for | | | | |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Test Scenario** | Verify that the user can search for specific fishing boats on the dashboard.. | | | | | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | **Actual Results** | | | **Pass / Fail / Not executed / Suspended** | | |
|  |
| 1 | Enter the Boat ID or name in the search field on the dashboard | | Should display search results matching the entered criteria | | As Expected | | | Pass | | |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | | TC\_0W\_007 | **Test Case Description** | | Verify that the user can view weather forecast information on the dashboard.. | | | | | |
| **Created By** | | Fahim | **Reviewed By** | | Fahim | | **Version :** | | 3.1 | |
|  |  |  |  |  |  |  |  |  |  |  |
| **QA Tester’s Log** | | Review comments from Sarah incorporate in version 3.1 | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Tester's Name** | | Fahim | **Date Tested** | | 30 April ,24 | | **Test Case (Pass/Fail/Not Executed)** | | Pass | |
|  |  |  |  |  |  |  |  |  |  |  |
| **S #** | **Prerequisites:** | | |  | **S #** | **Test Data** | | | | |
| 1 | User must be logged in to the OceanWatch system. | | |  | 1 | None | | | | |
| 2 | OceanWatch system should be operational without any server downtime. | | |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Test Scenario** | Verify that the user can access weather forecast information on the dashboard. | | | | | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Step #** | **Step Details** | | **Expected Results** | | **Actual Results** | | | **Pass / Fail / Not executed / Suspended** | | |
|  |
| 1 | Navigate to the weather forecast section on the dashboard. | | Should display current and future weather forecast information for the specified area. | | As Expected | | | Pass | | |  |

**Software Maintenance Plan:**

****

**Figure: Maintenance Plan Divided into 3 categories**

**Software Maintenance and Updates:**

1)Conduct regular updates to keep OceanWatch up to date with technological advancements, security patches, and user requirements.

2)This includes bug fixes, performance enhancements, and feature additions based on user feedback and emerging needs.

3)Perform database maintenance, including backups, data cleaning, and optimization, to ensure smooth operation and data integrity.

**User Support, Security, and Compliance:**

1)Provide ongoing user support and training to address queries, issues, and feedback from stakeholders.

2)Implement robust security measures, including regular audits, authentication mechanisms, encryption protocols, and access controls, to safeguard sensitive data.

3)Stay informed about relevant regulations and compliance requirements related to maritime safety and data privacy, ensuring OceanWatch's adherence through updates and modifications.

**Performance Optimization and Disaster Preparedness:**

1)Monitor OceanWatch's performance and implement optimizations to improve speed and efficiency, especially during peak usage.

2)Develop and update a comprehensive disaster recovery plan to mitigate the impact of unforeseen events such as natural disasters, cyberattacks, or hardware failures.

3)Maintain positive vendor relationships and clear communication channels to address any issues or concerns promptly, ensuring uninterrupted service delivery.