







**Version History**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Date** | **Change** | **Author** | **Reviewed by** |
| 1.0.0 | 19.09.2023 | Test Plan, Test activities on all features and UI testing | Arifa Akter | Abdul Quadir |
| 1.0.1 | 21.09.2023 | Schedule section updated and Features to tested updated | Arifa Akter | Abdul Quadir |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Term** | **Abbreviation & Definition** | | | |
| DRIVER | Data for Road Incident Visualization, Evaluation, & Reporting. This is the official name of the application. | | | |
| RCF | Road Crash Form. Special Entry screen to input brief information. This is seen on the Form input section. | | | |
| ARF | Accident Report Form. Detail Incidence Information Entry Screen in WebInstance of DRIVER. It is similar to the Accident Report Forum (ARF) used by the Dhaka Metropolitan Police to record traffic accident data. | | | |
| Blackspot | Location in a road where the highest number of traffic accidents occurs is called a Black Spot. | | | |
| Incident (blue dots) | Locations where incidents have occurred shown on the map as blue dots. | | | |
| Heatmap | Visual representation of the frequency and distribution of accidents that have occurred in a given area. | | | |
| GUI | Graphical User Interface | | | |

**Abbreviation List**

**Table of contents**



[1. Introduction 4](#_heading=h.gjdgxs)

[2. Reference 4](#_heading=h.1fob9te)

3. [High Level Test Objective](#_heading=h.3dy6vkm) 5

[4. Test Strategy 5](#_heading=h.1t3h5sf)

[Strategy 5](#_heading=h.4d34og8)

[Test Types 5-6](#_heading=h.2s8eyo1)

[5. Features to be Tested 6](#_heading=h.17dp8vu)

[6. Features not to be Tested 7](#_heading=h.26in1rg)

[7. Test Estimation 7](#_heading=h.lnxbz9)

[8. Release Procedure 8](#_heading=h.35nkun2)

[9. Test Suspension Criteria 8](#_heading=h.1ksv4uv)

[10. Test Acceptance Criteria: 9](#_heading=h.44sinio)

[11. QA Task List and Testing Process 9](#_heading=h.2jxsxqh)

[12. Test Environment 10](#_heading=h.z337ya)

[Hardware Requirement: 10](#_heading=h.3j2qqm3)

[Software Requirement: 10](#_heading=h.1y810tw)

[Network Requirement: 10](#_heading=h.4i7ojhp)

[Tools to be used: 10](#_heading=h.2xcytpi)

[13. Schedule 11](#_heading=h.1ci93xb)

[14. QA Summary Report 11](#_heading=h.3whwml4)

[15. Roles and Responsibilities 12](#_heading=h.2bn6wsx)

[16. Risk and Contingencies 12](#_heading=h.qsh70q)

[17. Test Exit Criteria 13](#_heading=h.3as4poj)

[18. Bug Status Explanation 13](#_heading=h.1pxezwc)

[19. Test Deliverables 14](#_heading=h.49x2ik5)

[20. Test Plan Approvals 14](#_heading=h.2p2csry)

# Introduction



The DRIVER (Data for Road Incident Visualization Evaluation and Reporting) project, a free and open-source web-based platform that has been developed, deployed and piloted by the World Bank and GRSF and holds the potential to redefine approach to collecting, analyzing, and managing road crash data.

DRIVER utilized the robust Laravel backend language to efficiently gather, process, and report data on road accidents. This open and free-to-access website serves as a central hub for road safety data, featuring a range of powerful tools and interfaces, including graph and map visualizations, custom reporting and filtering capabilities, support for multiple concurrent users, and seamless data export functionality.

The primary purpose of this Test Plan is to serve as the fundamental framework for our testing strategy, with the primary goal of ensuring the reliability, functionality, and readiness of the DRIVER system for deployment. In the following sections, we will provide a detailed overview of our methodology, goals, scope, and testing methods, all with the aim of aligning our efforts with our mission to improve road safety and make informed decisions. It provides a clear and comprehensive overview of how testing will be executed, documented, and reported throughout the software development lifecycle.

# Reference



The following documents are used as sources of information for this test plan:

| **Ref. No** | **Document Title** |
| --- | --- |
| 1.0 | Provided Technical Specification of Deployment and Configuration of the DRIVER Software  [[DRSP] Technical Document for Deployment and Configuration of the DRIVER Software V 1.0.0.pdf - Nextcloud (bjitgroup.com)](https://nextcloud.bjitgroup.com/index.php/s/oqNqYX9QS33Ec2b?dir=undefined&path=%2FManual_SQA_Project_Docs&openfile=2362907) |
| 2.0 | Provided User Manual of DRIVER System  [A Brief User Manual of the DRIVER System V 1.0.1.pdf - Nextcloud (bjitgroup.com)](https://nextcloud.bjitgroup.com/index.php/s/oqNqYX9QS33Ec2b?dir=undefined&path=%2FManual_SQA_Project_Docs&openfile=2362909) |
| 3.0 | DRIVER RFP  [DRIVER\_RFP.pdf - Nextcloud (bjitgroup.com)](https://nextcloud.bjitgroup.com/index.php/s/oqNqYX9QS33Ec2b?dir=undefined&path=%2FManual_SQA_Project_Docs&openfile=2362908) |
| 4.0 | Module List  https://nextcloud.bjitgroup.com/index.php/apps/onlyoffice/s/oqNqYX9QS33Ec2b?fileId=2362906 |
| 5.0 | DRIVER\_Feature List & Estimation  [[Client Name] DRIVER\_Feature List & Estimation V 1.0.0.xlsx](https://docs.google.com/spreadsheets/d/1uwJRSe18HQgQmm5ffxN7QDMa-Ab5-xuX/edit#gid=1120810750) |

**Note:** The Project will be developed following a clone of Agile based methodology. Each Sprint duration will be 2 weeks. This test plan may also be changed according to the changes at any phase of testing.

# High Level Test Objective



The main objective for the DRIVER system is to make sure that the DRIVER software is thoroughly and methodically tested in order to confirm its functionality and usability. Here we have outlined the following test objectives:

* Verify that DRIVER's core functions align with requirements, ensuring it collects, analyzes, and reports road crash data effectively.
* Check the data entry and integration are seamless.
* Check the functionality and accuracy of the dashboard, including map-based visualizations, graphical representations of incidents, and filtering options to provide meaningful insights.
* Verify data export capabilities, specifically in CSV format.
* Proactively defect detection and correction to ensure quick problem resolution.
* Create extensive test plans to ensure complete testing coverage.
* Verify the accuracy and completeness of the software requirements.

# Test Strategy



## Strategy

To ensure the quality of the applications. testing will be conducted based on following approaches:

* **System Testing Strategy:**

System testing is a comprehensive method that assesses the entire DRIVER System as a one integrated system. It ensures that all system components, modules, and features work together seamlessly. This testing phase verifies the system's overall performance, stability, and functionality when different parts of the system interact. It also validates that data flows correctly between various components and that the system can handle expected loads without errors.

* **Functional Testing Strategies**

Functional testing will dissect the DRIVER System's distinct features and functions. Each module and feature is carefully scrutinized to make sure it performs in accordance with the specifications.

* **UI Testing Strategies:**

UI testing evaluates the usability and design of user interfaces.UI testing ensures that the UI components, such as buttons, forms, and menus, are well-placed and adhere to design specifications. It also evaluates language support, including proper rendering and functionality in multiple languages, such as Bengali and English.

## Test Level

Only the following test level will be conducted to ensure the quality:

**System Testing:**

We'll concentrate on system testing, which comprehensively evaluates the overall performance, functionality, and integration of the DRIVER System. This level of testing ensures that the system operates seamlessly as a unified entity, meeting all specified requirements and adhering to defined specifications.

## Test Types

To complete the test of our system, we will use a variety of testing kinds, including the following:

**Functional Testing:**

Functional testing will involve comprehensive testing scenarios, including positive and negative cases. The primary objective at this level is to ensure that the DRIVER System aligns with its defined Scope Statement and performs within the specified infrastructure and application environment. This testing phase assesses the system's functionalities in various conditions to validate its adherence to requirements and expectations. Functional testing is essential for verifying that the system performs its intended tasks accurately and reliably.

**GUI Test:**

GUI testing thoroughly assesses a graphical user interface (GUI) to make sure it adheres to design criteria and provides the best user experience. Examining user interactions, responsiveness, and the precise depiction of GUI components like buttons, text, graphics, and menus are all part of this process. Organizations can ensure that their software works exactly as intended and satisfies the highest standards of usability

**Integration Testing:**

Integration Testing ensures smooth collaboration among various system components and modules, verifying their interaction. It guarantees that different parts of the system work harmoniously together

# Features to be Tested.



| **Phases** | **Sprint** | **Features** |
| --- | --- | --- |
| 1 | 1 | * **Complete UI/UX design for DRIVER System** * **Login** * **Dashboard** * Zoom in & zoom out icon * An icon to select STREETS & SATELLITE * Analyze & filter * View all incidents * Loss amount * Detail Loss amount * Number of incidents (30/90/365 Days) * Day Name * Time * Event count mark * Description * View * **My Account** * **Manage Duplicate Records** * **Map & Statistics** * Save Filters * Clear Filters * Add a record * Radio * Checkboxes * Graphs * Export > Custom report * Export > Export CSV * Traffic Enforcement Assignments * Pop-up view * **Record List** * Occurred Date > From date * Occurred Date > To Date * Weather * SURFACE TYPE * COLLISION TYPE * ACCIDENT SEVERITY * JUNCTION TYPE * CONTRIBUTORY FACTOR * DRIVER SEX * PASSENGER SEX * Saved Filters * Clear Filters * Add a record * View * Edit * **Add a Records** * **RCF (Road Crash Form)** * **ARF (Accident Report Form)** |
| Note:  Test plan and execution may vary depending on the development progress and release. | | |

# Features not to be Tested.



| **Phases** | **Features** |
| --- | --- |
| 1. | **Pixel Perfection Testing:**  Pixel perfection testing which is under UI testing will not be conducted as part of this testing phase because design documentation is not provided such as figma.  **Non Functional Testing:**  No functional testing such as performance testing, load testing, and stress testing,security testing will not be conducted as part of this testing phase.  **Compatibility Testing:**  Testing the system's compatibility with various browsers, devices, and operating systems is not the focus of this phase.  **Database Testing:**  Testing of database functionality and performance, including database schema validation and optimization, is not within the scope of this phase.  **Regression Testing:**  Ensuring existing functionalities remain unaffected by changes.It is also excluded. |
|
|
|
|

# Test Estimation



Testing effort may depend on several factors including.

* Quality of the Test basis
* Size of the product
* Complexity of the problem domain
* Requirements for documentation
* Time pressure
* Number of defects and the amount of rework required.
* Retesting testing

# Release Procedure



Below procedures will be followed for Release:

* Step-1: Requirement Analysis
* Step-2: Start Development and make internal release for QA on Sprint first day
* Step-3: QA continue testing and report bug
* Step-4: Developer complete the rest of development and start fixing current sprint Bugs
* Step-5: Make a Final Release for current Sprint on Sprint closing day
* Step-6: QA Confirm last release bugs fixed in Final Release
* Step-7: QA make complete respective Sprint Testing Scope and record bugs
* Step-8: If there do not have any blocking issue and bug Severity is low, Application goes release otherwise release will not be done.
* Step-9: Remaining bugs will be fixed in next Sprint release

# Test Suspension Criteria



Testing will be suspended, and QA team will reject the receivables upon the following criteria:

* During testing blocking issues are identified.
* Respective bug is not fixed in the dedicated release.
* Release without release note

**Note:** If any case will happen then QA have to raise issue to respective stakeholders

# Test Acceptance Criteria:



* Application UI should be intuitive, user-friendly, and free from major usability issues.
* Application does not have any blocking issues.
* Application has 85% Test case coverage.
* Application have covered supported required browsers (Windows Chrome).

# QA Task List and Testing Process



Below Tasks will be performed by the QA Team:

* Requirement analysis
* Identify Test areas.
* Test Case writing on identified test areas.
* Prepare Test environment.
* Execute Test Cases
* Bug reporting/retest
* Deliver Test report.
* Perform Test closure activity.
* Daily morning meeting

# Test Environment



To prepare the test bed for **the** project followings are the requirement:

## Hardware Requirement:

* PC (Widows) / Laptop

## Software Requirement:

* **Operating System**: Windows 10 or above
* **Microsoft Office**
* **Google Chrome**

## Network Requirement:

* + - Internet connectivity to PC /Laptop

## Tools to be used:

* + - **Test Case management**: Microsoft Excel
    - **Document management**: NextCloud
    - **Project management**: Redmine

Device oriented testing will be conducted as per following plan:

| Platform | Browser/OS | Device | Details | Comments |
| --- | --- | --- | --- | --- |
| Windows 10 | Chrome  Version 116.0.5845.188 | Laptop/PC | Screen resolution: 1366\*768 |  |

# Schedule



Schedule will be updated as Sprint feature release:

| Phases | Feature/Modules | Test Items | # of Test Cases | Internal Release | Final Release |
| --- | --- | --- | --- | --- | --- |
| 1 | * **Complete UI/UX Design testing for DRIVER System** * **Login** * **Dashboard** * Zoom in & zoom out icon * An icon to select STREETS & SATELLITE * Analyze & filter * View all incidents * Loss amount * Detail Loss amount * Number of incidents * Day Name * Time * Event count mark * Description * View * **My Account** * **Manage Duplicate Records** * **Map & Statistics** * Save Filters * Clear Filters * Add a record * Radio * Checkboxes * Graphs * Export > Custom report * Export > Export CSV * Traffic Enforcement Assignments * Pop-up view * **Record List** * Occurred Date > From date * Occurred Date > To Date * Weather * SURFACE TYPE * COLLISION TYPE * ACCIDENT SEVERITY * JUNCTION TYPE * CONTRIBUTORY FACTOR * DRIVER SEX * PASSENGER SEX * Saved Filters * Clear Filters * Add a record * View * Edit * **Add a Records** * **RCF (Road Crash Form)** * **ARF (Accident Report Form)** | 47 | 134 | 25.09.2023 | 29.09.2023 |

# QA Summary Report



From this report all stockholders can view and judge the current project Quality

| **Sprint** | **URL** |
| --- | --- |
| 1 | QA Summary Report link here :  <https://drive.google.com/drive/folders/1tcWCDc2v7PZy0Z4OQ_VHnfd2m0sghq1L> |

# Roles and Responsibilities



| **Resource Name** | **Responsibilities** |
| --- | --- |
| N/A | General Manager |
| Abdul Quadir | Project Manager |
| N/A | BE |
| N/A | Software Architect |
| N/A | Software Engineer |
| Arifa Akter | QA |

# Risk and Contingencies



**Schedule:**

The testing timeline may be delayed for a number of reasons, including unanticipated technological difficulties, a lack of resources, or modifications to the project's scope. The total project schedule and delivery may be affected by these delays.

To mitigate schedule risks, a proactive approach will be taken. This includes:

* Identifying potential delays early in the project.
* Regularly monitoring progress and resource allocation.
* Adjusting resource allocation as needed to address delays promptly.
* Maintaining open communication with stakeholders to manage expectations.

**Testing:**

Risks associated with testing cover a range of issues that may affect how well the testing procedure is executed. Uncompleted test coverage, requirements that are not aligned, or problems with communication within the testing team are a few examples of these difficulties.To mitigate schedule risks, a proactive approach will be taken. This includes:

* Prioritizing comprehensive test coverage by identifying and documenting all relevant test scenarios.
* Ensuring that test scenarios align with project requirements and objectives.
* Fostering clear and regular communication among team members to minimize misunderstandings and streamline the testing process.
* Reviewing and updating test plans as necessary to address emerging issues promptly.

**Application Risk:**

The DRIVER System's intricacy raises concerns about application hazards. Technical difficulties, compatibility problems with various contexts, or unforeseen behavior in specific circumstances are a few ways that these hazards might appear.To mitigate schedule risks, a proactive approach will be taken. This includes:

* Categorizing identified issues by severity and addressing high-priority issues as top priorities.

# Test Exit Criteria



Testing process of the DRIVER application will be ended if following criteria are met:

* All specified functions are functioning properly.
* Major bugs are identified, resolved, and retested.
* All test cases are executed and passed.
* Testing is ongoing but the PM requested to release the system.

# Bug Status Explanation



We maintain following status of the Bug in our Test Execution report:

**New:** SQAE creates a new bug. Sets the Assignee to PM/TL/SE

**Rejected:** If the reported bug is invalid, PM/TL changes the status to Rejected & Assignee to SQAE.

**Assigned:** PM/TL/SE changes the status to Assigned

**In Progress:** Assignee changes the status to InProgress when s/he starts working. Assignee records Spent Time every day.

**Submitted:** Assignee changes the status to Submitted when s/he finished the task & set the Assignee to PM/TL. Assignee Record Spent time.

**Feedback:** If the review isn’t successful PM/TL changes the status to Feedback & Assignee to SE.

**Reviewed:** Assignee reviews and changes the status to Reviewed and Assignee to SQAE

**Fixed but Failed:** Assignee (SQAE) retest and if not fixed then changes the status to Fixed but Failed and Assignee to PM/TL

**Resolved:** Assignee (SQAE) retest and if fixed then changes the status to Resolved and Assignee to PM/TL

**Close:** PM/TL (if assigned) will close the ticket if review is successful and clarified the feedback

**Reopen:** PM/TL can set the closed task status to Reopen, if necessary.

# Test Deliverables



Followings are the deliverables from QA for **DRIVER** project:

**Test Deliverables before Testing**

* Test Plan
* Test Cases

**Test Deliverables after Testing**

* Test execution report
* Test report (each sprint)
* Known Issues in Sprint Release Note

*Known issues are well mentioned in each sprint release note.*

# Test Plan Approvals

| **Name** | **Roles** | **Signature** | **Date** |
| --- | --- | --- | --- |
| Mr. Abdul Quadir | Project Manager |  |  |

