**Destiny Discover**

**TEST STRATEGY DOCUMENT**

**2021 Release – Destiny Discover – Back to School**

**Table of Content**

[**1. Document Control** 4](#_Toc68749102)

[**1.1. Change History** 4](#_Toc68749103)

[**1.2 Approvals** 4](#_Toc68749104)

[**2. Introduction** 5](#_Toc68749105)

[**2.1 Document Purpose** 5](#_Toc68749106)

[**2.2 General Project Description** 6](#_Toc68749107)

[**2.3 Project Definition** 6](#_Toc68749108)

[2.3.1 Release features 6](#_Toc68749109)

[**3. Test Objectives** 9](#_Toc68749110)

[**3.1 Testing Scope** 10](#_Toc68749111)

[3.1.1 In Scope 11](#_Toc68749112)

[3.1.2 Out of Scope 12](#_Toc68749113)

[3.2 Quantitative Goals 12](#_Toc68749114)

[4. Test Approach 14](#_Toc68749115)

[**4.1 Test Data Dependencies** 19](#_Toc68749116)

[4.2 Test Type 19](#_Toc68749117)

[4.2.1 Smoke Testing 19](#_Toc68749118)

[4.2.2 Functional & UI Testing 19](#_Toc68749119)

[4.2.3 Regression Testing 19](#_Toc68749120)

[4.2.4 End to End Testing 20](#_Toc68749121)

[4.2.5 API Testing 20](#_Toc68749122)

[5. Entry and Exit Criteria 20](#_Toc68749123)

￼

￼

[5.3 Resumption Criteria 22](#_Toc68749126)

[6. Defect Logging and Management 23](#_Toc68749127)

[6.1 Defect Management Plan - 23](#_Toc68749128)

[6.2 Defects Status Definitions 24](#_Toc68749129)

[6.3 Defects Severity Definitions 25](#_Toc68749130)

[6.4 QA Sign off Criteria 27](#_Toc68749131)

[7.1 Devices Testing 27](#_Toc68749132)

[7.2 Accessibility Testing 27](#_Toc68749133)

[7.2.1 Accessibility Testing Approach 28](#_Toc68749134)

[7.3 Device Matrix 31](#_Toc68749135)

[7.4 Performance Testing 31](#_Toc68749136)

[8. Automation Testing Strategy 32](#_Toc68749137)

[8.1 Automated Web Services Testing 32](#_Toc68749138)

[8.1.1 In Scope 32](#_Toc68749139)

[8.1.2 Out of Scope 32](#_Toc68749140)

[8.2 Front End Automation 33](#_Toc68749141)

[8.2.1 Front End Automation Approach 33](#_Toc68749142)

[9.1 Tools Used 34](#_Toc68749143)

[10. Photon Contacts 34](#_Toc68749144)

[10.1 Roles and Responsibilities 34](#_Toc68749145)

[11. Testing Requirements 35](#_Toc68749146)

[12. Additional Requirement 35](#_Toc68749147)

[12.1 Communication Plan 35](#_Toc68749148)

[12.2 QA Reporting 36](#_Toc68749149)

# 

# **1. Document Control**

## **1.1. Change History**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version No. | Date | Changed Sections | Change Description | Author |
| 0.1 | 02/26/2020 | All | Draft Version | Lakshmi/Arshath |
| 0.2 | 03/01/2021 | All section | Updated detailed feature list, scenarios, program plan | Puviyarasan |
| 1.0 | 04/07/2021 | All | All section reviewed and updated | Puviyarasan |

## **1.2 Approvals**

The Quality Control Test has been developed using the Business Requirements, Functional requirement document, Non Functional Requirement Document and technical documentation provided by the Business Development Team / Application development team, the business functionality stated in the Program Plan and the documentation provided by the Business Analysis Team.

Test Scenarios will be created by: Photon QA Team

We the undersigned have reviewed the Test Requirements, which identifies the Testing Requirements, Process and the Test Scenarios to be executed. We all approve of the content as it relates to our areas of expertise and will use it as our testing guideline. We also understand that any modifications in deliverables, user requirements, or technical requirements will impact this plan as well as its accompanying test schedule.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Role | Reviewer | Approver | Approval Date |
| Saikumar T S | Photon Delivery Manager | ✓ |  |  |
| Elenchezhiyan Shanmugasundaram | Technical Architect | ✓ |  |  |
| Balaji Ramamurthy | Photon QA Manager | ✓ |  |  |
| Bharathiraja | Photon Project Manager | ✓ |  |  |
| Ramkumar | Photon Lead BA | ✓ |  |  |
| Meganathan M | Technical Lead - Mobile | ✓ |  |  |
| Suresh R | Technical Lead - Mobile | ✓ |  |  |
| Puviyarasan | Photon QA Lead | ✓ |  |  |

# **2. Introduction**

## **2.1 Document Purpose**

* The purpose of the test strategy document is to clearly convey the scope and approach of functional, non-functional and regression testing that is to be performed for the Destiny Discover- Reimagine Digital transformation. It also provides details of the testing methodology, approach and tools used for testing.
* This is a test strategy composition and focuses on the aspects of testing approach for the program.

## **2.2 General Project Description**

Scope of the project is to enable the digital transformation of Destiny Discovery app.

* New feature of the application
* Reskin of the existing application

Detailed list of new features and reskin list are mentioned in 2.3.1 Release features section.

Photon will perform Creative, Feature/Function and Technical Discovery that will help define the strategy both from a user journey, app capabilities and technical integrations standpoint which is then followed by the agile software development track.

## **2.3 Project Definition**

* The goal is to test the Destiny Discover app which has the new capabilities that are getting built. Based on release plan, Photon QA to do smoke, functional/UI, regression testing and accessibility testing for the following functionalities
* To test the application after reskinning, based on the existing test case shared by Follet team, we will execute it manually and report the results.

### 2.3.1 Release features

* Below are the new features for the Destiny Discover- Reimagine Digital transformation application as specified in the Program Plan. Detailed list will be available in below follett share point path:

<https://follettcorp.sharepoint.com/:x:/r/sites/ReimagineDigital/_layouts/15/Doc.aspx?sourcedoc=%7B0BB63600-75C4-4DDA-B924-7CA19EB1683A%7D&file=PHTN%20FOLLETT%20Destiny%20Discover%20B2S%20Implementation%20Plan%20Ver%206.0.xlsx&action=default&mobileredirect=true>

List of features based on program plan

* + User Landing Screen - Personalized Home Page
  + Keyword & Advanced Search
  + Predictive & Intelligent Search
  + Search Results
  + Filter & Sort
  + Dashboard - Reading Progress
  + Create own Challenge or Join Ongoing Challenge
  + Challenge Dashboard
  + Find and Add Reading Program
  + Reading Program
  + Reading Challenge
  + Updates to Product Detail Page
  + Setting Recommendation Preference
  + Recommendations
  + Reader Performance Improvements
  + eBook Reader - Read Along
  + Audio Book Player - Audio Book Improvements
  + Comic Book Support
  + History
  + Push Notifications
  + Gamification
  + Student performance dashboard
* Also provided list of pages we are going to reskin as a part of this program plan.
  + Search/Browse School
  + Browse Site, Landing Page
  + User Authentication -
    - User Login,
    - Multiple Device Login,
    - Create Account,
    - My Profile
  + Landing Screen, Customize Landing, Add/Edit Links
  + Global Navigation - Global Menu
  + View Collections -
    - View Featured/Public/Private Collections,
    - Sort/Filter Collections,
    - Copy to personal collection,
    - Report collection,
    - Pin/Unpin Collection,
    - Follow/Unfollow,
    - Schedule Pins,
    - Organize/Arrange Collections order
  + Create/Edit Collection -
    - Create/Public/Private Collection,
    - Upload Image,
    - Edit Collection,
    - Share Collection,
    - Copy Collection,
    - Delete Collection,
    - Attach Standards,
    - View Activity History,
    - Generate PDF
  + Update Collection1 (Add/Update Items)
    - Grid/List View
    - Sort By
    - Add URLs
    - Item Tagging
    - Add Notes
    - Add Playlist
    - Upload from Cloud
    - Add from collections
    - Organize/Arrange Items order
    - Remove
    - Move To
    - Copy To
    - Information (i)
  + PDP -
    - Book Details page
    - Book Information
    - Call Information
    - Book Label
    - Add/Remove Carousal
    - Explore Tags
    - Add to collections
    - Add Citation
  + eBook Reader -
    - Table Of Content
    - Export Notes
    - Highlight text
    - Annotate Text
    - Add Bookmark
    - Search Within Ebook
  + Audio Book Player
  + Bookmarks
  + Lightbox Integration,
  + Ratings - Add Ratings, View Ratings
  + Reviews, Add Reviews, View Reviews
  + Awards - View Awards
  + Checkout -
    - Hold Book
    - Checkout/Return Book
    - Renew Book
    - Play Audio Book
    - View eBook
    - Assignments
  + Fines - View Fines
  + Inventory - Book Status, Book Availability
  + History - Checkout History, Active Checkouts, Active Holds
  + Favorites - Add/Remove Favorites, View Favorites
  + Wishlist - Add/Remove Wishlist, Search to add to Wish, View Wishlist
  + Widgets - Configure Widgets
  + Download - Download e-Books, Automatic Download
  + User Feedback, Feedback to Follet
  + Miscellaneous - Help Link, Third party links
  + Print Catalog Optimization

Note- Above release features are picked up from current version of program plan.

# **3. Test Objectives**

The objectives for Photon QA validation of this project’s system functions are:

* **Test Effectively** – To ensure the overall test coverage and reduce the number of defects going to later stages of project; limit test misses through traceability of requirements to test cases, identification and timely resolution of ambiguities, and gap analysis on functional and business requirements to the government contract
* **Test Efficiently** – By effectively using resources, limit redundancy in testing across teams and use automation for Smoke and Regression Testing
* **Test Smartly** – By using enablers and adapting newer testing techniques
* **Test Strategically** – By aligning testing goals to that of the program, focusing on high priority requirements testing and defect resolution

## **3.1 Testing Scope**

This section outlines the various types and levels of testing that will be performed. It will also provide high level details of what will be out of testing scope. Any deviations/changes from the proposal/contract should also be identified here.

For the new features that are being built, following test will be performed

* Unit testing
* Smoke testing
* Functional testing
* Performance testing
* Accessibility testing
* End to End testing
* User acceptance testing

For the UI that is reskinned , we will execute regression test cases from Follett team and report the results after reskinning.

### 3.1.1 In Scope

Features available in the below program plan are scope for our testing:

<https://follettcorp.sharepoint.com/:x:/r/sites/ReimagineDigital/_layouts/15/Doc.aspx?sourcedoc=%7B0BB63600-75C4-4DDA-B924-7CA19EB1683A%7D&file=PHTN%20FOLLETT%20Destiny%20Discover%20B2S%20Implementation%20Plan%20Ver%206.0.xlsx&action=default&mobileredirect=true>

Based on the NFR we will be covering browser and device coverage

<https://teams.microsoft.com/_#/files/General?threadId=19%3A05c91ce21d834721b38753599de000c3%40thread.tacv2&ctx=channel&context=Non%2520Functional%2520Requirements&rootfolder=%252Fsites%252FReimagineDigital%252FShared%2520Documents%252FGeneral%252FFrom%2520Photon%252FNon%2520Functional%2520Requirements>

**Browser Coverage for Web Testing**

|  |  |  |
| --- | --- | --- |
| PLATFORM | BROWSER VERSION | MOBILE / BROWSER NAME |
| Windows | Latest Stable version (89) | Chrome (Smoke and E2E Testing) |
| Mac | Latest Stable version (14) | Safari (Smoke and E2E Testing) |
| Windows | Latest Stable version (87) | Edge - Only smoke testing. |
| Windows | Latest Stable version (84) | Firefox - Only smoke testing. |
| iOS Mobile | 61 to 88 | Chrome - Only smoke testing. |
| iOS Mobile | 6 to 14 | Safari - Only smoke testing. |
| Android Mobile | 61 to 88 | Chrome - Only smoke testing. |
| Android Mobile | 12 to 23 | Samsung Internet - Only smoke testing. |
| Chrome OS | Latest stable version | Chrome - Only smoke testing. |

**Device Coverage for Native Mobile Testing:**

|  |  |  |
| --- | --- | --- |
| PLATFORM | OS VERSION | MOBILE / BROWSER NAME |
| iOS | Combination of OS version 13 and above | * iPhone 12 mini (E2E Testing) * iPhone 12 Pro Max (E2E Testing) * iPhone 11 (Smoke Testing) * iPhone 11 Pro (Smoke Testing) * iPad Pro (E2E Testing) * iPad Air 4 (Smoke Testing) * iPad mini 5 (Smoke Testing) |
| Android | Combination of OS version 6 and above | * Samsung Galaxy Note 20 (E2E Testing) * Samsung Galaxy S10 Plus (Smoke Testing) * Samsung Galaxy S9 (Smoke Testing) * Google Pixel 4 (Smoke Testing) * Google Pixel 3XL (Smoke Testing) * Samsung Galaxy Tab S6 (E2E Testing) |
| Kindle | 10Th Gen | * Amazon Fire HD 10 (Smoke Testing) |

### 3.1.2 Out of Scope

|  |  |  |
| --- | --- | --- |
| Out of Scope Items | Why Out of scope for QA | Who needs to test - Responsible Team |
| Location based testing | From offshore exact locations based validation is not possible | Onsite team to validate the locations based testing |

# 3.2 Quantitative Goals

|  |  |
| --- | --- |
| Goal | Description |
| Obtain Baseline and Signed Off Requirements | Photon QA to obtain baseline and signed off documents of Use Cases, and NFR documents before start of Test execution phase as the testing strategy and approach is dependent on the same |
| Requirements Walkthrough | Photon QA team should be involved during the User Stories walkthroughs and help BA to identify the additional scenarios to be included in User stories and convert it as test scenarios |
| Traceability | **<TBD>** |
| Defect Tracking | All defects would be tracked in DevOps and sufficient details to be provided by DevOps around the defects. Dev partners to change the defect status and update the root cause of the defects in DevOps  All defects will be tracked within the Azure DevOps.  Azure Devops-   * Project Name: Destiny Discover - Reimagine digital transformation. * The DevOps Application will be used for recording, managing and reporting on defects. Defects review meetings will be held to discuss and review the Photon QA defects. |
| UAT Defects | All defects logged in by the Business will not be recorded as test misses. Photon QA will perform analysis at the end of testing. Defects identified by UAT after completion of Photon QA testing would be considered as test misses if it is in Photon QA scope. All defects reported by UAT should be logged by business in DevOps with detailed description. |
| Status Reporting | Status report would be shared by Photon QA on a daily basis in Dev/QA week. Execution status in the status report should match the numbers in DevOps and the project plan. |

# 4. Test Approach

This section provides an overview of the test approach being taken for each testing type. Also guidelines governing execution of each test type.

|  |  |
| --- | --- |
| Testing Type | Test Description |
| Smoke Testing | * Test scenarios are used from the BDD format user stories * Conduct test scenarios which are in BDD gherkin format walkthrough with relevant stakeholders * Design test scenarios in BDD gherkin format – to test the behaviors written by the business analysts. * Ensure QA environment readiness prior to execution * Execute test scenarios * This is an in sprint activity * Identify categorize (Critical, high, medium, low), fix and re-verify the defects * DevOps will be used for test case management |
| Functional & UI Testing | * Building test scenarios in BDD gherkin format * Conduct test scenarios which are in BDD gherkin format walkthrough with relevant stakeholders * Scope of functional and UI testing will be defined as per the sprints in the approved program plan * Design test scenarios that cover functionality described in User Stories. * For the UI testing the Test scenarios/Testing will be done based on the Mocks, and executed during the dev/QA week. * UI testing would be performed only for the features mentioned in User stories and screens from mocks. * Design traceability metrics to ensure complete coverage against business and functional requirements and design elements * Ensure QA environment readiness prior to execution * Test Data requirements from partners covering all test scenarios * Identify categorize (Critical, high, medium, low), fix and re-verify the defects * Complete relevant test type closure documentation * DevOps will be used for test case management |
| Regression Testing | * Building test scenarios in BDD gherkin format * Conduct test scenarios which are in BDD gherkin format walkthrough with relevant stakeholders * Regression testing will be done as per the timeline mentioned in the approved Program Plan * Identify non-testable scenarios and take them off from the testable requirements * Design traceability metrics to ensure complete coverage against business and functional requirements and design elements * Ensure QA environment readiness prior to execution * Test Data requirements from partners covering all test scenarios * Execute test scenarios in Devops Azure * Identify categorize (Critical, high, medium, low), fix and re-verify the defects * Complete relevant test type closure documentation * DevOps will be used for writing and tracking the test cases |
| End to End Testing | * End to End testing will be done as per the timeline mentioned in the approved Program Plan * Test data setup is complete for End To End testing * Identify non-testable scenarios and take them off from the testable requirements * Design traceability metrics to ensure complete coverage against business and functional requirements and design elements * Ensure QA environment readiness prior to execution * Test Data requirements from partners covering all test scenarios * Execute test scenarios in Devops Azure Identify categorize (Critical, high, medium, low), fix and re-verify the defects * Complete relevant test type closure documentation * DevOps will be used for writing and tracking the test cases |
| API Testing | * API testing will be done as per the timeline mentioned in the approved Program Plan * API testing will be based on Service Specification provided by API team * Based on the available details from Services specification doc, services will be validated * Ensure QA environment readiness prior to execution * Service assurance is a 9 point checklist of tests assuring various aspects of APIs that are consumed by the App. Refer appendix for details * Validate the services are working as expected * Report will be generated & shared to the team |
| Accessibility testing | * Accessibility testing includes requirements that are technical and relate to the underlying code rather than to visual experience * Ensure QA environment readiness prior to execution * Test Data requirements from partners covering all test scenarios * Analyzing and prioritizing the findings, making recommendations * Complete relevant test type closure documentation * DevOps will be used for writing and tracking the bugs |

## 

## **4.1 Test Data Dependencies**

NA

## 4.2 Test Type

### 4.2.1 Smoke Testing

* Smoke tests are completely automated.
* Ensure QA environment readiness prior to execution
* The smoke tests are executed on each build.
* Passing smoke testing is an entry criteria for accepting the build for testing.
* The smoke testing is usually about 10% of the regression test cases.

### 4.2.2 Functional & UI Testing

* Test scenarios are derived from use cases
* Design traceability metrics to ensure complete coverage against business and functional requirements and design elements
* Ensure QA environment readiness prior to execution
* Test Data requirements from partners covering all test scenarios
* Execute test scenarios which is in BDD gherkin format in Devops Azure Identify categorize (Critical, high, medium, low), fix and re-verify the defects
* For the UI testing will be done as an in sprint activity
* UI testing would be performed only for the features mentioned in User stories and screens from mocks.

### 4.2.3 Regression Testing

* Test scenarios will be pulled from user stories as well depending on the Critical/ High Defect Fixed made during functional testing Phase.
* All Automated cases from the Functional test suite which is obtained from user stories will be utilized as part of regression Along with Manual Test scenarios identified for Regression
* Ensure QA environment readiness prior to execution

### 4.2.4 End to End Testing

* Test scenarios are derived from use cases
* Design traceability metrics to ensure complete coverage against business and functional requirements and design elements
* Test data setup is complete for End To End testing
* Ensure QA environment readiness prior to execution
* Test Data requirements from partners covering all test scenarios
* End to End testing will be done as per the timeline mentioned in the approved Program Plan
* Critical functionalities are tested after API’s are integrated as per the program plan
* Identify categorize (Critical, high, medium, low), fix and re-verify the defects

### 4.2.5 API Testing

* API testing will be based on Service Specification provided by API team
* Based on the available details from Services specification doc, services will be validated
* Ensure QA environment readiness prior to execution
* Service assurance is a 9 point checklist of tests assuring various aspects of APIs that are consumed by the App
* Validate the services are working as expected
* Report will be generated & shared to the Destiny Discover team.

# 5. Entry and Exit Criteria

Entry and Exit criteria is based on the terminology used in Devops Azure

|  |  |  |
| --- | --- | --- |
| Phase | Entry Criteria | Exit Criteria |
| Dev / QA Week | 1. Technology should be finalized 2. Use cases / Test Scenarios should be signed-off 3. Mocks should be signed-off 4. Feature coverage should be 100% in all the platforms. 5. Formal Release Notes Comprising of features implemented in the cadences, defects fixed, known issues and software component versions to be provided. 6. Test cases should have been reviewed and approved by the client | 1. Feature coverage should be 100% in all the platforms. 2. Formal Release Notes Comprising of features implemented in the cadences, defects fixed, known issues and software component versions to be provided. 3. Unit testing should be 100% in all platforms 4. Test coverage is 100% in all the platforms (Web, Android & iOS). |
| UAT Week | 1. Formal Release Notes Comprising of planned features implemented in the cadences, defects fixed, known issues and software component versions to be provided. 2. All completed user stories feature should be demoed to Destiny Discover QA team 3. Photon QA team to submit test execution and test coverage reports 4. Photon QA team should have provide sign off to start UAT | 1. Business requirements are met 2. All the identified defects are fixed 3. UAT/Business team provided sign-off |

## 5.1 Test Cycle Planning

List of planned work based on Iteration are tracked in program plan mentioned below:

<https://follettcorp.sharepoint.com/:x:/r/sites/ReimagineDigital/_layouts/15/Doc.aspx?sourcedoc=%7B0BB63600-75C4-4DDA-B924-7CA19EB1683A%7D&file=PHTN%20FOLLETT%20Destiny%20Discover%20B2S%20Implementation%20Plan%20Ver%206.0.xlsx&action=default&mobileredirect=true>

## 5.2 Suspension Criteria

Testing will be stopped completely when:

* Any showstopper (impediment) is encountered during test execution
* A key portion of the base functionality test fails

There is a change to the scope or plan that impacts the statistical control of the process, which results in the process being out-of-control.

## 5.3 Resumption Criteria

When the reasons for the suspension of testing have been defined and resolved, a new build should be deployed to the QA environment. Depending on the problem the testing may need to be restarted from the beginning.

|  |  |
| --- | --- |
| Suspension Criteria | Resumption Requirements |
| Non-availability of the application | Availability of the application |
| Show stopping defects, which prevent further testing along any path | Show stoppers (Impediments) are rectified |
| All the possible test scenarios have been tested with the failure documented and the fixes for the same are not available for re-testing | Modification of code in QA env with fixed defects |
| Too many critical defects in the early phase of testing | Fix the defects and repeat the impacted scripts. |

# 6. Defect Logging and Management

Azure DevOps will be used for defect logging and defect management purposes. The defects will be tracked till closure.

## 6.1 Defect Management Plan

Any defect at any point of time will have a status. The following different statuses are available in the **DevOps - Defect Life** Cycle lightly and will be finalized after discussion with project team:

|  |  |
| --- | --- |
| Defect Management Repositories | **Azure DevOps** |
| Defect management Plan | Defect Review meetings will be held on the need basis during the execution phase with development Team and BA |
| Defect Prevention activity | All the Test Scenarios and the Test Execution will be reviewed within the Photon QA team to avoid any defect leakages |
| Defect Management Quality Assurance Reporting | Azure DevOps- Defect Management Tool will be used to track the defects arising from Functional Testing.  Automation Testing will be initiated in the scope of functional Testing.  Azure DevOps - Defect Management Tool will be used for recording, managing, and reporting on defects. Defects review meetings will be held to discuss and review |

## 6.2 Defects Status Definitions

The below table has to be aligned with DevOps Azure

|  |  |  |  |
| --- | --- | --- | --- |
| **Defect Status** | **Description** | **Owner** | **Sub-Status** |
| **Open** | Defect created by Tester and Assigned to BA | QA/UAT | Awaiting Dev Response |
| **In Progress** | Development Lead has assigned to a Developer and investigation is going on the defect | BA/Dev | Awaiting Dev Response |
| **Code Review** | Development team has done the analysis and based on analysis Dev fixed the defect and code is Under review | Dev | Awaiting Dev Response |
| **Ready for QA** | Developer has fixed defects and delivered them to QA back | Dev | Awaiting QA Response |
| **QA Testing** | QA Testing (In-Progress) | Dev | Awaiting QA Response |
| **Resolved** | Defect Fixed. | QA/UAT |  |
| **Closed** | No further action required – Closed by QA. Defect is working as expected | QA/UAT |  |
| **Reopened** | Defect retesting failed should be assigned back to Dev team. | QA/UAT | Awaiting Dev Response |
| **Deferred** | Defect is going to defer for the next release (Mutually agreed by Business). Deferred defect should be intimated/informed to QA team (via email) | UAT | Dev Defer |
| **Invalid** | Based on the analysis done in triage meetings, if defect seems to be invalid due to multiple reasons like wrong test data, test team knowledge, duplicate defect etc. (Mutually agreed by QA/Dev/Business) Cancelled defect should be intimated/informed to QA team (via email) | QA/Dev/ Business | Dev Cancel |

## 6.3 Defects Severity Definitions

The below table has to be aligned with DevOps Azure

This is a mandatory field and needs to be compulsorily populated depending on the severity of the defect. The following convention is followed while assigning the severity to a defect:

|  |  |  |
| --- | --- | --- |
| **Severity** | **Definition** | **Example** |
| **1 - Critical** | Application & Testing cannot continue until issue is resolved  Environment or Application not Functioning.  Defect prevents execution of major (core) functionality and has no work around  Defect has major implication to the business. | Application Page does not load  After clicking a button the application hangs  Clicking a link leads to system exception error  Row check fails for data validation |
| **2 - High** | Major functionality produces wrong results and has workaround  Environment or Application not Functioning or partially functioning  Testing partially blocked  Defect prevents further use of portion of a work product  Resulting system has reduced usability for the end user | Field validation missing  System responds incorrectly to invalid data, error handling not implemented  Once data migration is complete, the data doesn’t appear in column  Cannot save to database |
| **3 - Medium** | Functional Problem with workaround  Testing not blocked  Minor functionality produces wrong results or missed requirement | Clicking on a link takes the user to the wrong location on the same page  Scroll bar not working, incorrect labels, and instructional text, heading or sub headings. |
| **4 - Low** | A defect that does not affect the functionality of the system.  Only minor cosmetic issues. | Spelling or grammatical mistakes on web pages.  Font type or color used is not according to the specified format |

## 6.4 QA Sign off Criteria

* Build deployed in QA environment with proper release notes
* Test cases & traceability reviewed and signed-off
* 100% of critical and high defects fixed, verified & closed.
* After defect triage with business team, low severity (Sev 3, 4) open defects will be deferred

7. Non Functional Testing

## 7.1 Devices Testing

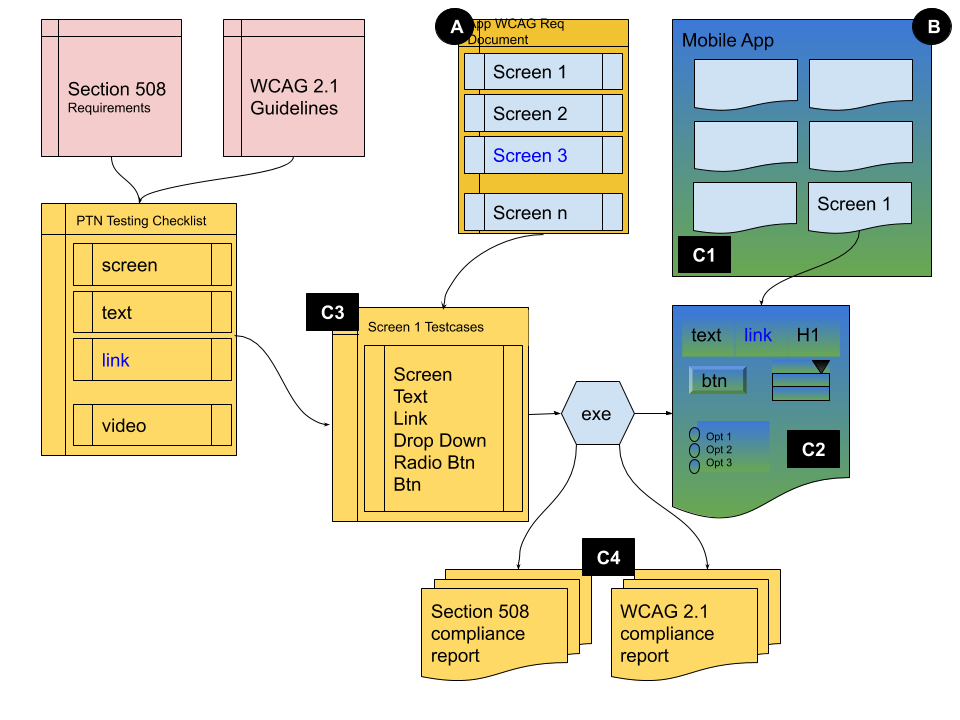
* Photon QA identifies the list of Devices based on the NFR as well as considering the Follett Customer usage metrics in the US region.
* All device testing are updated in In Scope section based on NFR details: <https://teams.microsoft.com/_#/files/General?threadId=19%3A05c91ce21d834721b38753599de000c3%40thread.tacv2&ctx=channel&context=Non%2520Functional%2520Requirements&rootfolder=%252Fsites%252FReimagineDigital%252FShared%2520Documents%252FGeneral%252FFrom%2520Photon%252FNon%2520Functional%2520Requirements>

## 7.2 Accessibility Testing

* Meet WCAG 2.1 guideline and 508 guidelines to help physically disabled people interact with applications.
* The list of Tools & Devices are as follows and can be accessed from below NFR link:
  + Color Contrast Analyzer – Paciello
  + Magnifier/Zoom testing
  + Screen Reader
    - Voice Over – iOS (MAC, Mobile & iPad devices)
    - NVDA, JAWS – Windows
    - Talkback – Android

### 7.2.1 Accessibility Testing Approach

* Accessibility testing will be performed along with functional testing
* Accessibility implementation will be validated screen-wise by using the approved tools
* As part of accessibility testing will be performed based on approved accessibility requirements for the App
* Any issues are identified, will be logged and assigned to dev / creative teams as appropriate
* The Testing Approach is further broken down as follows.
* C1. App decomposition into Screens by Functionality.
* C2. Screen decomposition into individual components on the screen as defined in Photon's Accessibility Testing Checklist.
* C3. Test case development for each component based on Photon's Accessibility Testing Checklist against the defined Requirements for each component.
* C4. Test Execution based on the defined test cases and reported against Section 508 Compliance and WCAG 2.1 Compliance by Level A & AA.



* WCAG 2.1 checklist
* Android
  + Manual Functional testing of Accessibility is performed using TalkBack, the Android screen reader. Enabling TalkBack via the Accessibility Settings helps navigate the application like a visually disabled person would.
  + Enabling TalkBack slightly changes the way the application navigation slightly. Double Tap is used to select an item, list scrolling is performed using two fingers and navigation to different elements is done by swiping right or left across the screen.
* iOS
  + A complete Accessibility Assessment and Testing of an iOS application requires using the device’s Voice Over feature.
  + Enabling the Voiceover in the Accessibility Settings changes the user’s interaction with the device and the applications. This includes item selection and navigation through the application.
  + Testing should be performed at every screen and item level to ensure Accessibility Descriptive content is present and informative. Testing should also be performed using functional flows through the application to ensure they can be performed successfully using the Accessibility tags.
* Web
  + Web accessibility testing is performed using the screen reader NVDA (Chrome and Firefox) and Jaws (IE)
  + NVDA is free, open source, portable screen reader for Microsoft Windows whereas Jaws is a paid version
  + Testing should be performed by using the dedicated commands provided to move through object hierarchy within an application
* Other aspects of Accessibility Testing include Color Contrast Analysis to ensure there is sufficient contrast between the fore and background colors and flashing content that can lead to seizures. Color Contrast analysis is done using Color Contrast Analysis tools by capturing the screenshot of the application and comparing the fore and background colors. This will be done screen wise for all screens in the application.

7.2.2 Accessibility Tools

|  |  |
| --- | --- |
| Mobile Device Accessibility Tools | |
| Tool | Platform |
| Talkback - Screen Reader Testing (inbuilt) | Android |
| Voice Over (inbuilt) | iOS |
| NVDA, Jaws – Screen reader | Web |
| Color Contrast Analyzer Tool | Both Mobile and Web |

## 7.3 Device Matrix

Below on the NFR doc, the devices list mentioned In Scope section considered for testing the Destiny Discover app.

## 7.4 Performance Testing

Performance testing for the requirements called out in the Request and Response, Availability, Usability sections of the NFR will be validated as part of Sprint cycles.

In addition to the above, Screen load performance, Storage performance, Battery drain testing will be done to fulfill the requirements in NFR.

A combination of manual and automated approaches will be taken to assess the app performance. Screen load times can be assessed by automated scripts, while memory profiling and app footprint usage tests will be done using a manual functional flow.

7.5 Test Assumptions

This section lists the various assumptions and Dependencies in executing this release:

* The Test Scenarios will be signed off by Destiny Discover QA Team/Product Owner.
* The Test/QA environment will be available by the start date as mentioned in the Program Plan for executing the test cases.
* The object code will be fully unit tested and Unit test results along with Release notes will be shared with the testing team for every QA build deployed to the QA Environment.
* All approved Decision Requests and Change Requests in Use Cases or functional specifications will be forwarded upon approval to the Test Team Lead to ensure that the test cases are modified and the functionality are re-tested, wherever necessary.
* All the Security related use cases (from NFR) will be tested and validated.

# 8. Automation Testing Strategy

The Automation testing strategy for the Destiny Discover- Web and Mobile App-Reimagine Digital Transformation project is comprised of:

* Service Assurance
* Front end functional test automation

## 8.1 Automated Web Services Testing

Web services testing ensures that web services are functionally correct. Automation testing of web services is performed using JMeter. Using JMeter, minimizes the manual effort whenever any changes happen in the service layer. Below points represents the scope of services / API testing by Photon QA team,

### 8.1.1 In Scope

* Photon QA team will perform assurance for services that are consumed by the Hybrid App
* JMeter tool will be used for scripting services
* Service assurance is a 9 point checklist of tests assuring various aspects of APIs that are consumed by the App
* Report will be generated & shared to the B&T team

### 8.1.2 Out of Scope

* Any third party Web services/API scripting not part of Photon QA scope

## 8.2 Front End Automation

Scope of Photon QA automation is associated with the features that are defined as part of the User Stories.

* QA team will derive test scenarios from the user stories, a feasibility study of the test scenarios will be done to assess if the test scenarios are automatable or not, thus automation test scenarios are derived.
* Once test scenarios are identified for automation, the mobile automation team will start creating / adapting the feature files and once build deployed in QA environment automation team start scripting the step definitions and functions using framework
* On every new build Photon QA team will be running the scripts against the features developed in the agreed Perfecto cloud devices and will be sharing the execution reports.

### 8.2.1 Front End Automation Approach

The Photon QA team uses the internal photon proven framework which comprises below tools

* Java
* BDD
* Appium
* Selenium
* Extent Report
* Maven
* Junit
* Github will be used by the Photon Automation QA team

9. Environment Requirement

For Functional and Regression testing this section identifies the hardware, software and other resource requirements for executing the Photon QA including tools.

## 9.1 Tools Used

Below are the Photon suggested tools for validating B&T Web and Mobile application

|  |  |  |  |
| --- | --- | --- | --- |
| S.No | Tool | Version | Tool Capability |
| 1 | Azure DevOps | - | Defect Tracking & Test Management Tool |
| 2 | JMeter | 5.x | API Testing Tool |
| 3 | Talkback, Accessibility Scanner for Android | - | Accessibility Tool |
| 4 | Voiceover, Accessibility inspector for iOS |  | Accessibility Tool |
| 5 | NVDA (Chrome, Firefox) and JAWS (IE) for Web app in Windows |  | Accessibility Tool |
| 6 | Eclipse | Latest stable version | IDE(Integrated Development Environment) |
| 7 | Selenium WebDriver | Latest stable version | Web Automation Tool |
| 8 | Appium | 1.15.1 | Mobile Automation Tool |
| 10 | Extent Report | 3.0.2 | Automation Reporting Tool |
| 11 | GitHub | - | Automation Script Repository |
| 12 | Azure DevOps |  | Ci/CD environment tool |

# 10. Photon Contacts

## 10.1 Roles and Responsibilities

|  |  |
| --- | --- |
| Name | Role |
| Saikumar T S | Photon Delivery Manager |
| Elenchezhiyan Shanmugasundaram | Technical Architect |
| Balaji Ramamurthy | Photon QA Manager |
| Bharathiraja | Photon Project Manager |
| Ramkumar | Photon Lead BA |
| Meganathan M | Technical Lead - Mobile |
| Suresh R | Technical Lead - Mobile |
| Puviyarasan | Photon QA Lead |

# 11. Testing Requirements

Below link provides the testing requirements,

<https://dev.azure.com/baker-taylorAxis360Suite/Reimagine%20Digital/>

11.1 Test Deliverables and Timeline

Below program plan provides the QA deliverables and Timeline details,

<https://follettcorp.sharepoint.com/:x:/r/sites/ReimagineDigital/_layouts/15/Doc.aspx?sourcedoc=%7B0BB63600-75C4-4DDA-B924-7CA19EB1683A%7D&file=PHTN%20FOLLETT%20Destiny%20Discover%20B2S%20Implementation%20Plan%20Ver%206.0.xlsx&action=default&mobileredirect=true>

# 12. Additional Requirement

## 12.1 Communication Plan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Meeting Name | Purpose | Frequency | Owner | Intended Audience |
| User Story walkthrough | Requirement understanding/ Estimation | Baseline Requirement is available | BA | Development / QA Team |
| Estimation | Estimation Approval | Any change in the scope of Testing | Photon QA Manager | Photon QA Team |
| Photon QA test scenario walkthrough | Photon QA test scenario coverage & Photon QA test case Sign off | Test scenario design is completed | Photon QA Manager | Photon QA Team, Scrum Master, Business Team and Development Team |
| Defect call | Defect review | Daily/ weekly/ Needed basis | Photon QA Manager | Photon QA Team, Scrum Master, Business Team and remaining Development Team |
| Daily Status Meeting | Status Update | Dev/QA Week | Photon QA Manager | Photon QA Team, Scrum Master, Business Team and Development Team |

## 12.2 QA Reporting

Photon QA will produce testing reports to show testing progress. The table below lists the details of the test reports

|  |  |  |
| --- | --- | --- |
| Report Name | Details | Frequency |
| Test Case Metrics | Details of total test cases by feature | Daily |
| Test Execution Metrics | Test Execution Percentage by Feature  Test Pass Percentage by Feature  Test Fail Percentage by Feature | Daily |
| Defect Metrics | Snapshot of Open Defects by Severity  Trend of Open Defects by Severity | Daily |
| Automation Metrics | Test Automation Percentage Metrics  Automated Test Execution Metrics | Weekly |