**OTT Product-Test Strategy**------------------------------------------------------------------------------------------------------------------------------------------

**Table of Contents**

1 Introduction 2

1.1 Objective 2

2 Test Approach 2

2.1 Compatibility Test 2

2.2 Functional Test 3

2.3 Perfromance Test 4

2.4 Usability Test 4

2.5 Security Test 5

2.5 Test Automation 5

2.5 Test Reporting 5

# Introduction

This document describes the various types of testing that will be conducted on the OTT Product on different platforms to validate that the system performs as per the agreed requirements.

## Objective

The objective of the Test Strategy is to define the overall testing approach that is adopted for testing OTT product in multiple platforms (Web, Mobile and Smart TV).

# Test Approach

## 2.1 Compatibility Test

Compatibility testing is non-functional testing. It is carried out to verify whether an application can run on different devices, browsers, operating systems, hardware, and networks.

 In case of OTT product the compatibility test is very vital as the customer base is with different devices, browsers and use different networks.So to ensure that the applications works as intended, the application needs to be tested with multiple devices with different OS versions and needs to make sure the application will support on newer OS versions which will be releasing to the market.

**Web application:**

* **Operating System Compatibility Testing** – Application is tested by running it on different operating systems such as Windows and Mac.
* **Network Compatibility Testing** – In this type of testing, an application is checked on different networks such as WiFi, 3g, 4G, etc. for various parameters such as bandwidth, speed, etc.
* **Browser Compatibility Testing –** It tests an application’s working on various browsers such as Chrome, Firefox, Internet Explorer, Safari etc in both mobile and web browsers, to make sure it is compatible with them.

**Mobile /Smart TV Applications:**

* **Hardware Compatibility Testing**– In this type of testing, mobile application is tested on various hardware configurations devices to make sure it works on them in the desired manner.
* **Network Compatibility Testing –** In this type of testing, application is checked on different networks such as WiFi, 3g, 4G, etc .
* **Device Compatibility Testing –** This is done to check an application’s compatibility with various external devices such as connecting to chrome cast, head phones, Bluetooth speakers etc.
* **Mobile Compatibility Testing –** In this type of testing, an application is tested on various mobile devices with different operating systems such as Android, iOS, and Windows. Browser stack is one of the tool which will help to achieve this.
* **Version Compatibility Testing**– In this type of testing, an application’s compatibility with its older and newer versions is checked. It is further divided into two types-

1. **Backward Compatibility Testing –**It tests an application’s compatibility with its older versions. Eg .The application needs to be tested in older OS versions as in some devices the users cannot update to newer versions.
2. **Forward Compatibility Testing –**It is performed to make sure an application is compatible with its future versions as much as possible.

## 2.2 Functional Test

The functional testing process involves the following steps across the platforms (Web, Mobile and Smart TV)-

* Identification of business requirements or functionalities.
* Test data preparation based on the functionalities to be tested.
* Finding of the expected outcome or the expected results.
* Test case execution.
* Comparing the actual and expected results.

Major Functionalities which will be covered as part of the Functional testing for OTT application.

* 1. **User registration and Subscription-** User registration from multiple platforms and the different type of subscription methods (CC payment, Google and Apple IAP’s) needs to be tested and validated .User created from any platform should be allowed to access the application and its functionalities from other platforms as well seamlessly.
  2. **Asset Curating from Store front application**-Integration of store front application to client needs to be validated as it is important to make sure that changes made on store front is reflecting on the client end.
  3. **Video Streaming –** Streaming of asset based on the subscription package (SD/HD) needs to be validated.

In case of play back the test should be done on various devices connected to different networks to make sure the play back is not impacted due to the different version and network speed.

More user centric test like how the playback behaves while user keeps the app in background and while he receives a call needs to test to make sure it is not impacting the user experience on returning to application to continue streaming.

Play back related analytics can be validated using the tool called Conviva.

* 1. **Play back Controls-** Multiple player controls options like enabling the subtitle, audio selection needs to be tested and validated that the user selections are retained in the device on binge watch**.**

Mini player option, a most demanding feature should work as intended.

* 1. **User Profiles-** Users should be allowed to create the profile specific to rating and the content displayed for that profile should be based on the selected rating of the profile. Kids lock feature should be tested to make sure that customer is not facing any issue with kids using the application.
  2. **Download Feature (Mobile application)-** User’s should be allowed to select the video quality for download and allowed to play the downloaded content offline. All Penthera configs to be validated to make sure that studio agreements are met on allowing the download for a content.
  3. **Analytics –** All the integrated analytics (GA,Firebase, clevertap )should be tested to make sure that the events are logged as per the requirement and are available for analysis of the data.
  4. **Integrated Systems –** The integration to the additional systems like Clevertap, Branch io ,Newrelic ,Conviva needs to be validated to make sure that on all the releases the integrations are working as expected.

## 2.3 Performance Test

**Performance testing** of any application helps in identifying the performance bottlenecks in the application. In case of OTT product the performance test is important to make sure that the application will work as intended when there is a spike in the stream request majorly in case of releasing a much awaited movie/TV show.

* A performance test tool which suits the application based on the implementation or programing language support needs to be identified. Eg if the application is Java based the JMeter/Neo load can be used for testing.
* **Load testing-** It is essential to perform Load Testing over OTT network servers to understand the performance of OTT service under millions of stream requests from around the globe at once.
* **Stress testing-** Type of performance testing in which the performance of the application is evaluated under a load much higher than the expected or anticipated load. This helps in measuring the break-even point of the application – the load at which the application doesn’t respond correctly.

## 2.4 Usability Test

UI/UX testing is critical in OTT as the market is swarmed with several devices, so it is essential for the OTT service providers to ensure that the application experience is similar on all devices. In order to provide a consistent experience across multiple devices, the UI/UX design should highly flexible to accommodate a wide variety of devices of different size. Therefore, a good UI/UX design and its verification is a critical component to ensure that the OTT service delivers a smooth user experience.

* Identify the device base from the analytics and make sure that the application UI/UX supports as intended in all those devices.
* Check the application in landscape and Portrait mode for Mobile apps and for web application in mobile browsers.
* Along with ensuring that the functional application flows are working as indenting, it is also vital to check if the UI/UX design is simple and intuitive for the end-user to navigate through the controls quickly.

## 2.5 Security Test

Security testing is one of the most crucial testing types of non-functional testing. In this type of testing, the system’s readiness to fight back against any external or internal attacks is tested. It makes sure that only authorized and authenticated users are allowed to access the software, and users’ data is secured and available to them whenever required.

* Authorization and Authentication Testing- Ensure that the application is allowed to use after proper validation of user name and password for the authorized/registered users.
* DRM Protected Contents- Make sure that the streaming is available for the user’s with subscription and the DRM protected contents cannot be recorded from the device.
* Security testing need to be done using any security tool which will scan the application to identify the security breech.

## 2.6 Automation Test

Automation Test is useful to reduce the overall testing time, thus helping in faster product releases.

The tool which supports the application and which requires less maintenance need to be identified prior to go with automation.

1. **Test cases that test critical functionality of the application** -The smoke and sanity test suit can be automated to make sure that the build available for testing are acceptable.
2. **To improve the test coverage** –In case of a release, the whole regression cases cannot be executed in all the test devices, hence when we have a Sanity suite which is automated helps to make user that the application works as indenting in multiple combination devices same time and increase the test coverage.

## 2.7 Test reporting

1. **Bug Report**

* Use of label feature in Jira -Proper labeling for bugs identified in each release will help to track the defects reported in a particular version.
* Filter option – Create a filter in Jira for tracking the defect reported in the release.
* Dashboard – Jira dashboards with different widgets are highly recommended for managing the bug report and sharing to the team.

1. **Test Results**

**Test Management tool-** Test cases are created and managed in **Test Rail.**

* For each release a test plan is created and mapped with multiple test runs according to the number of test cycles and test types. While executing the test cases the test results with test evidence are attached in the Test Rail and the failed cases will be linked to the JIRA bugs.
* On completing the testing for a version the test plan created will be closed and the report can be generated.
* Report will have the total number of cases executed with Passed, failed, blocked or untested counts.

1. **Test Matrix**

On each release along with the Test report from Test Rail and the Defect report from JIRA additionally a Defect Matrix and Test summary form in excel will be shared while providing the UAT testing signoff email.

**Defect matrix-** It contains the differentiation of defects count split on basic of the severity, the type eg: Live/Regression/New Feature bug.

**Health Check Tracker-** It helps to quickly identify the health of the features releasing in a particular version. While doing the testing of the feature based on the bugs identified the feature will be highlighted in different color code.

Eg : Red means critical/high issue existing in the feature.

Green means the feature is passed and all the issues identified is fixed and closed.