# System Test Plan

# for

# Unravel

**Version 1.0 approved**

**Prepared by:**
**Aman Choudhary**

**CS 02**

**September 23, 2017**

# 1 Introduction

## 1.1 Purpose

We can detect the significant differences between existing and desired conditions and evaluate the system features at the same time using a process which is called Testing. With the help of system test plan document we can brief the scope, approach and scheduling of the testing activities which includes the modules, features to be tested and the kind of the testing to be performed to ensure the software will work accordingly to the description in the SRS and the integration will be tested.

The main purpose to develop the test plan are as follows:

- To determine the techniques to be utilized for testing.
- Identify that what are the features of the product that are to be tested.
- What are the approaches that will be used for testing.
- Identify the bugs and manage them.
- What will be the criteria for acceptance.

## 1.2 Project Overview

In today's world for learning anything or any subject we prefer video lectures and students always face problems in finding the proper content to understand that subject.So our project is providing a platform where user can get a sorted content , trending video, playlists and projects.

## 1.3 Objective

The main reason to make this document is to make sure that our software system meets all the requirements that are desired.Maintaining the quality of the product all the considered scenario's must be considered.Also the system testing identifies all the risks and issues to the project and makes sure that all of these will be addressed in a proper manner before releasing the product.

## 1.4 Formal Reviewing

There will be many review points during and before the System Testing.

## 1.5 Formal Review Points

- Survey Analysis Document.
- Design Documents.
- Testing Approach.
- System test progress.

# 2 Test Items

First of all we will be testing all the units separately to make sure that all the things that we plan to achieve are working well individually. Prime objective of testing our software is to make sure that all the features mentioned in the SRS are working properly without any error. To ensure this we will also do integration testing later to provide the user with the promised features.

# 3 Software Risk Items

- Login error
- search error
- System hangs at user side
- videos not playing properly
- project not available properly

# 4 Feature that are to be tested

Following are the various features that are to be tested from the ”User’s” viewpoint.This system test plan is according to the Software Requirements Specications.

- Search bar module
- Registration module
- Login module
- Playlist module
- User module

# 5 Feature not to be tested

- Security

# 6 Approach

There are basically three levels for testing as that are used in Software Engineering. These tests includes Unit Testing, Integration Testing and System Testing. As the coding phase continues we will be conducting unit testing on the modules which get completed. Once coding and unit testing is completed, we can begin with integration testing which will involve integrating the modules and testing them accordingly. Then we will proceed towards validation testing in which we will check whether all requirements have been met or not. After validation testing we will perform system testing in which our whole system will be tested.

# 7 Test Strategy

## 7.1 Types of testing

### 7.1.1 Unit testing

It is done by the developer to segregate each part of the program and test that the individual parts are working properly or not.On giving input to any function or procedure it should return an answer i.e correct if the input is valid otherwise it should show an error message or a warning message.

### 7.1.2 Integration Testing

Integration testing is done to make sure that the modules on combing gives the desired output i.e the interface between the modules works correctly and modules work well when combined to build up the system.
There are basically three types of approaches that we follow while doing integration testing:

- Big Bang Integration Testing
  In this integration we test all components or modules simultaneously, after which everything is tested as a whole
- Top-Down Integration testing:
  In this we test from top to bottom
- Bottom-Top Integration testing:
  Testing takes place from bottom to top.

### 7.1.3 Validation testing

Verification and Validation are two important tests, which are carried out on software. Verification testing help us in evaluating a software system by determining whether the product of a given development phase satisfy the requirement establish before the start of the phase.
Validation testing helps us in confirming that a product meets its intended use. Here intended use means customer's expectation.

### 7.1.4 System level testing

In this testing we test behaviour of whole system/product. It may include tests based on risks and/or requirement specifications, use cases, or other high level descriptions of system behaviour, interactions with the operating systems, and system resources. System testing is most often the final test to verify that the system to be delivered meets the specification and its purpose. System testing is carried out by specialist testers or independent testers. System testing should investigate both functional and non-functional requirements of the testing.

### 7.1.5 Performance Testing

It is to ensure that performance of the system meets the expectation of the users.

### 7.1.6 Stress Testing

It checks the error handling modules of the system whether it performs well in any kind of error.

## 7.2 Test Deliverable

- Test plan
- Test case document
- Test result

### 7.2.1 Software and Hardware needs

#### Hardware

A device in which we can play video and see the projects with the help of a browser from a website using internet.

#### Software

We only need a browser installed on any device through which we can play video and see the project from a website.

### 7.2.2 Planning Risk

The probable risk to the project are:

- Delay in developing the software.

- Major changes in the requirements of the product.

  The delay in developing the software could be either due to the members unable to fix the errors or faults encountered while developing the software.