Software Requirements Specification (SRS)

# 1. Introduction

## 1.1 Purpose

The Smart Video Tool is a Python-based desktop application that allows users to:  
- Download YouTube videos.  
- Generate QR codes for any URL.  
- Display live weather information.  
- Display current internet speed.  
  
This SRS defines the requirements for the successful development and deployment of the Smart Video Tool.

## 1.2 Scope

The software will support:  
- Video downloading (best available quality).  
- QR code generation for easy sharing.  
- Real-time display of weather updates.  
- Real-time internet speed testing.  
- Theme switching (Light Mode / Dark Mode).  
- Visual feedback through animations and progress bars.  
  
This is a standalone desktop application developed using Python and PyQt5, and packaged using PyInstaller.

## 1.3 Definitions, Acronyms, and Abbreviations

GUI: Graphical User Interface  
API: Application Programming Interface  
URL: Uniform Resource Locator  
PyQt5: Python bindings for the Qt GUI toolkit  
QR Code: Quick Response Code

# 2. Overall Description

## 2.1 Product Perspective

The application is a new, independent product. It is intended to be easy to use without requiring technical knowledge. It uses external APIs (OpenWeatherMap, ip-api) and third-party libraries like yt\_dlp, speedtest, qrcode.

## 2.2 Product Functions

- Download YouTube videos via URL.  
- Generate and display a QR code for any entered URL.  
- Detect and display user's city weather automatically.  
- Test and display download/upload internet speed.  
- Provide a dark/light theme toggle option.  
- Display video thumbnail or QR code as preview.

## 2.3 User Characteristics

- General users with basic computer skills.  
- No programming knowledge required.  
- Requires internet access for full functionality.

## 2.4 Constraints

- Internet connection required for downloading videos, weather detection, and speed testing.  
- Application runs only on Windows OS (for initial release).  
- YouTube URL must be publicly accessible.

## 2.5 Assumptions and Dependencies

- User has administrative rights to install and run applications.  
- Necessary Python libraries are bundled with the executable using PyInstaller.

# 3. Specific Requirements

## 3.1 Functional Requirements (FR)

FR1: The system must allow the user to enter a YouTube video URL.  
FR2: The system must download the YouTube video and save it locally.  
FR3: The system must generate a QR code for the entered URL.  
FR4: The system must display the current city weather information.  
FR5: The system must display internet download and upload speeds.  
FR6: The system must allow toggling between dark and light themes.  
FR7: The system must show video thumbnail or QR code in a preview section.

## 3.2 Non-Functional Requirements (NFR)

NFR1: The application should respond to user actions within 2 seconds.  
NFR2: The download progress must be shown using a progress bar.  
NFR3: The GUI must be visually appealing and user-friendly.  
NFR4: The application must maintain low CPU and memory usage during idle state.  
NFR5: The app must gracefully handle errors like invalid URL, no internet, etc.

# 4. External Interface Requirements

## 4.1 User Interfaces

- GUI made with PyQt5.  
- Buttons for Download, QR generation, Theme switching.  
- Text box for URL input.  
- Labels for weather, speed, status.  
- Progress bar for download indication.  
- Image area for preview.

## 4.2 Hardware Interfaces

- Standard keyboard and mouse for input.  
- Screen resolution recommended: 1366x768 or higher.

## 4.3 Software Interfaces

- YouTube server (for downloading videos via yt\_dlp).  
- OpenWeatherMap API (for weather data).  
- ip-api (for location detection).  
- Local filesystem (for saving downloads and previews).

- Internet connection for downloading videos and fetching weather/speed data.

# 5. System Features

- YouTube Video Download: Allow user to download YouTube videos by providing the URL.  
- QR Code Generation: Generate QR code from the entered URL and display it.  
- Live Weather Display: Detect user's city automatically and show live temperature.  
- Internet Speed Test: Check and display internet download/upload speed.  
- Theme Switching: Toggle between light and dark mode for better usability.  
- Animated GUI: Use fade-in animations and opacity effects for modern UI.

# 6. Other Requirements

- The application should be lightweight (executable size < 100MB).  
- Software must be distributed as a single .exe file for ease of use.

# 7. Appendix

**APIs Used:**  
- OpenWeatherMap API (https://openweathermap.org/api)  
- IP-API (http://ip-api.com/)  
  
**Libraries:**  
- PyQt5, yt\_dlp, qrcode, requests, speedtest-cli