

# Architecture

## Django - Video Conferencing Web App

<b>Revision Number</b>	1.0
<b>Last Date of Revision</b>	18-Feb-2022
<b>Author</b>	Sai Kumar Reddy Korsapati

## Document Version Control

Date Issued	Version	Description	Author
18-Feb-2022	1.0	First Draft	Sai Kumar

## Contents

Topic	Page No
Document Version Control	2
Abstract	4
1. Introduction	5
1.1 Why this Low-Level Design Document	5
1.2 Scope	5
1.3 Constraints	6
1.4 Risks	6
1.5 Out of Scope	6
2. Technical Specifications	6
3. Technology Stack	6
4. Proposed Solution	7
5. User I/O Workflow	8
6. Exceptional Scenarios	8
7. Test Cases	9

## Abstract

For the past decade, video conferencing (VC) has become more popular and more reliable as a tool to bridge the distance gap when travel is not an option, impractical or undesired. Real-time Communication is quickly becoming a “must-have” feature in many types of applications. From customer service to telehealth, video conferencing is rapidly integrating into different industries’ workflows. Video conferencing uses audio and video telecommunications to bring people at different sites together. Understanding what is required for videoconferencing and its application has become one of the major researched topics by various learning institutions and businessmen. In this paper, an introduction to video conferencing is presented with the emphasis on its application in distance learning.

# 1 Introduction

## 1.1 Why this Low-Level Design Document?

The purpose of this document is to present a detailed description of the Django - Video Conferencing Web App. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system and will be proposed to the higher management for its approval.

The main purpose of video conferencing is to enable face-to-face communication between two or more people in different locations. It is a popular alternative to phone conferencing for businesses and provides individual users with an inexpensive means of communication with distant friends and family.

It is only recently that technology has reached a level of stability, usability and affordability which permits its use in real teaching scenarios rather than research projects. The use of video is being hailed as the next advance in electronic communication. Many companies are developing systems to support such concepts as virtual teams, telecommuting, and remote conferencing.

Video conferencing has recently become increasingly popular and dispersed in the wake of faster and cheaper internet connections and better technologies. Modern standalone video conferencing units provide advanced video and audio quality due to more efficient compression and can function over normal broadband internet connections. Growing processing power and cheaper accessories, such as webcams, have also made it possible to participate in a video conference using dedicated software on a normal personal computer without any expensive special hardware.

## 1.2 Scope

Low-level design (LLD) is a component-level design process that follows a step-by step refinement process. This process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the data organization may be defined during requirement analysis and then refined during data design work.

### 1.3 Constraints

The VC system must be user friendly, as automated as possible and users should not be required to know any of the workings.

### 1.4 Risks

Document specific risks that have been identified or that should be considered.

### 1.5 Out of Scope

Delineate specific activities, capabilities, and items that are out of scope for the project.

## 2. Technical Specifications

### 2.1. Deployment

Heroku Cloud platform is used for hosting of the project.



## 3. Technology Stack

<b>Frontend</b>	Html, Css, Js, BootStrap
<b>Backend</b>	Python, Django
<b>Database</b>	Sqlite-3
<b>Deployment</b>	Heroku

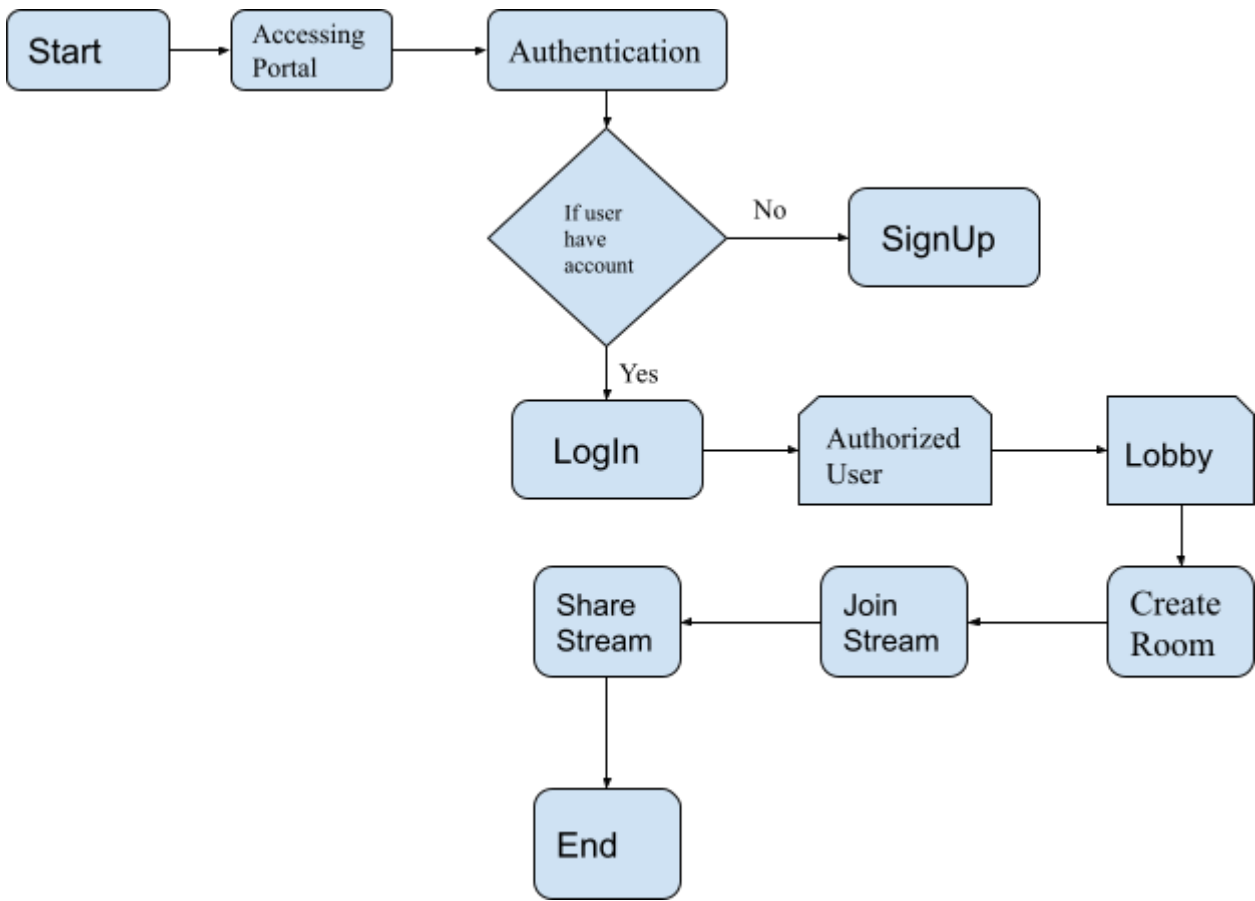
## 4. Proposed Solution

The solution proposed here is a web based Video Conference App using python - django framework can be implemented to perform above mentioned use cases .

In first case, if Video Conference Web App detects any illegal activities like Cross site scripting (XSS) protection, Cross site request forgery (CSRF) protection, SQL injection protection, Clickjacking protection, SSL/HTTPS, Host header validation, Referrer policy, Cross-origin opener policy, Session security all these will be handled by Django automatically to provide high security to our application, user sign in, sign up options provided will be used to authorize.

if user is already existing in database else will provide an option to register by themselves, further in the second use case, Video Conference Web App creates a bridge to share information to distance if travel is not an option like during covid lock down situation education classes can be held via video conference system and lastly in the final use case of Video Conference Web App, it implements meetings with video, audio, screen sharing facility to transfer information through online via web app helps viewers or students to grasp the knowledge without traveling.

### 5. User I/O Workflow



### 6. Exceptional Scenarios

Step	Exception	Mitigation	Module



## 7. Test Cases

Test case	Steps to perform test case	Module	Pass / Fail