**Architecture**

**Video Conferencing Application**

1. Introduction
   1. Why is Architecture Design Document?
   2. Scope
2. Architecture
   1. Project workflow and architecture
3. Deployment
   1. Heroku or AWS
4. **Introduction** 
   1. **What is Architecture design document?**

Any software needs the architectural design to represents the design of software. IEEE defines architectural design as “the process of defining a collection of hardware and software components and their interfaces to establish the framework for the development of a computer system.” The software that is built for computer-based systems can exhibit one of these many architectures.

Each style will describe a system category that consists of:

• A set of components (eg: a database, computational modules) that will perform a function required by the system.

• The set of connectors will help in coordination, communication, and cooperation between the components.

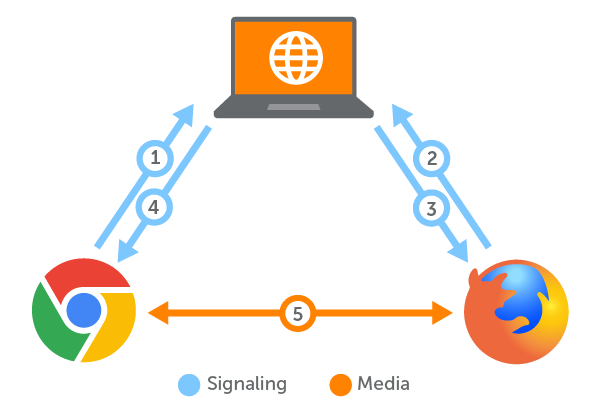
• Conditions that how components can be integrated to form the system.

• Semantic models that help the designer to understand the overall properties of the system.

**1.2 Scope**

Architecture Design Document (ADD) is an architecture design process that follows a step-by-step refinement process. The process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the design principles may be defined during requirement analysis and then refined during architectural design work.

1. Architecture



In order to make a peer-to-peer connection those peers will be connected using SDP i.e Session Description Protocol – Information about a peer that the other peer needs in order to make p2p connection. Each peer will send their sdp to other peer using signalling server and both peer will connect to signalling server using web sockets and when signalling server knows the location of each peer it can relay messages between the two peers and that way the two peers will exchange their sdp’s with each other so once those peers have each other’s sdp’s then they can establish a peer to peer connection and after that they don’t require a signalling server because then they can connect to each other using peer to peer connection and here we are implementing signalling server using django channels

So here the laptop is the django channel signalling server once the browsers connect to the django channel signalling server using web sockets and after that they exchange their sdp’s and now once they have their sdp’s they establish a peer-to-peer connection with one another and then they will communicate with each other directly without the django channel signalling server

**Implementing Chatrooms**

Chatroom using groups and groups are like rooms they keep channel names and message can be sent through all the channels in the group to their respective peers and each user which connects to the channel name essentially keeps a channel name so when a peer sends the message that message can be sent through all the channels that are in the group and in that way all the messages will be broadcasted to those peers.

**What are Channels?**

Channels builds upon the native ASGI support available in Django since v3.0, and provides an implementation itself for Django v2.2. Django still handles traditional HTTP, whilst Channels give you the choice to handle other connections in either a synchronous or asynchronous style.

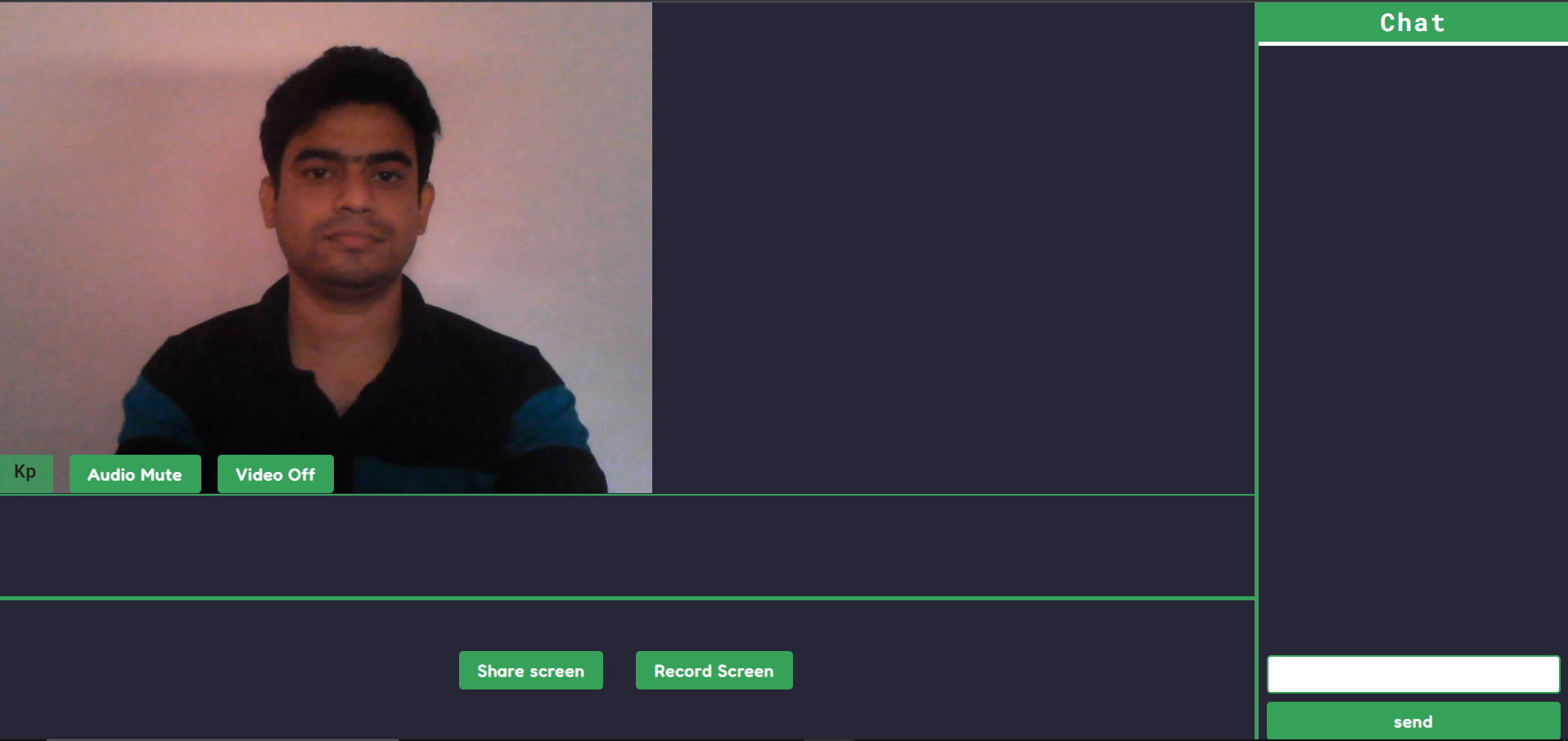
Channels is not just built around the world of HTTP and Web Sockets - it also allows you to build any protocol into a Django environment, by building a server that maps those protocols into a similar set of events. For example, you can build a Chabot in a similar style:

**Workflow**

Firstly, Peer joins the room then machine Sends message to all other peers’ indicating’s its entry. All other peer got notified of new peer through this message.Each existing peer intiaties peer connection with the new peer i.e. offers their respective SDP’s .New peer receives the each offer sdp. New peer send response for each offer sdp. Other peers receive respective sdps

New peer connected with each existing peer and that’s how they connect.

1. **Deployment**

****

