

# Full Stack Developer Take Home Challenge

## ***Architecture overview***

The application is an application made with Ruby On Rails, thus, implements the MVC design pattern. It has two models, Video and User and each request is handled by each controller. The login state is persisted and handled by a third controller, a session controller.

Communication with the Zype API is handled and organized in a directory called services (it could be a module) and it has two classes VideoService and UserService. Both inherits from an abstract class called ApiService. ApiService class is used for sharing the credentials needed for connect with Zype API as app\_key, client id and client secret.

VideoService class gets the video list and a video source link, a helper is used for generate a widget that render a video player.

UserService handles the user authorization for watching videos wich needs subscription.

The architecture used in “services”, attempts to represent a micro services architecture. To simplify the exercise, videos and user from Zype API are saved in two tables in a database.

I use Rest-client gem for connecting with Zype API.

The web design it's rough and bootstraped, but i focused in functionallity rather than design.

Next a class diagram is attached.

