LOW LEVEL DOCUMENT

Project Title	Tuber
Created By	Avnish Yadav "avnish@ineuron.ai"
Organisation	iNeuron Private Intelligence Limited
Document Version	0.1
Created Date	18th April 2022
Update Date	18th April 2022

Document Version Control

VERSION	DATE	Description	AUTHOR	COMMENTS
0.1	18th April 2022	Initial Draft	Avnish Yadav	Introduction & Architecture defined

Reviews

VERSION	DATE	REVIEWER	COMMENTS
0.1	18th April 2022	Ketan Gangal	

Approval Status

VERSION	DATE	APPROVER	COMMENTS
0.1	18th April 2022	Ketan Gangal	

Abstract

“XYZ Pvt Ltd” is an Edtech organisation with the vision of providing courses at an affordable price. Initially "XYZ Pvt Ltd" started providing professional training in the field of Artificial intelligence and Data Science, with trending demands of other technologies such as blockchain and full stack web development. To facilitate professionals training in trending technology "XYZ" has on boarded many educational youtuber to bring revolutionary improvements . Tubers platform helps "XYZ Pvt Ltd" management team to engage youtubers along with students will have opportunities to connect with their favourite youtubers.

1. Introduction

1.1 Why is this Low-Level Design document?

The purpose of this document is to present a detailed description on how a web application can be designed and deployed at any deployable server such as Azure App service, Elastic Beanstalk or in any Virtual Machine. We will be using Azure App Service to deploy our web application. DevOps is an engineering discipline that aims to unify development operations to standardise and streamline the Continuous Integration, Continuous Testing and Continuous Deployment of high-performing web applications in production. LLD 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 objective of the project is to define the complete lifecycle of a web application project. Automation will be carried in respect to Continuous Integration, Continuous Testing, and Continuous Deployment approaches. If there was no DevOps time consumption and expensive process leads to failure of the whole application. DevOps is considered as a vital part in development project because:

• DevOps is a combination of Research/Development and operations.

• Numerous manual tasks related to integration, testing and deployment can be automated through the process of Continuous Integration, Continuous Testing and Continuous Deployment. It speeds up the process of adding a new feature to add in an existing application or any improvement needs to be incorporated in the existing application.

1.2. Scope

This designed system will be a web application built using the popular python based web application development framework Django. Application allows administrators to add youtuber detail. Students can search for youtuber and submit contact forms so that meetings can be scheduled at regular intervals. Students need to create an account at tuber platform. Accounts can be created either by providing details or through any social account. DevOps pipeline will be established to facilitate the functionality of Continuous Integration, Continuous Testing and Continuous Deployment at Azure App Service.

1.3. Constraints

Admin users can be created from command for the first time. Admin user has privilege to convert any user to admin by logging to user account and by enabling a boolean field super_user. Youtuber details and Team can be managed only by admin users. Students have two options to create an account.

1. By submit registration form

2. Using Social account

1.4 Risks

Risk is associated with database migration in case of new database new migrations need to be done properly along with proper backup of data in case of failure. Databases should be up and running all the time as applications are tightly dependent on databases. Version Control System should not be avoided to prived issues with code versioning.

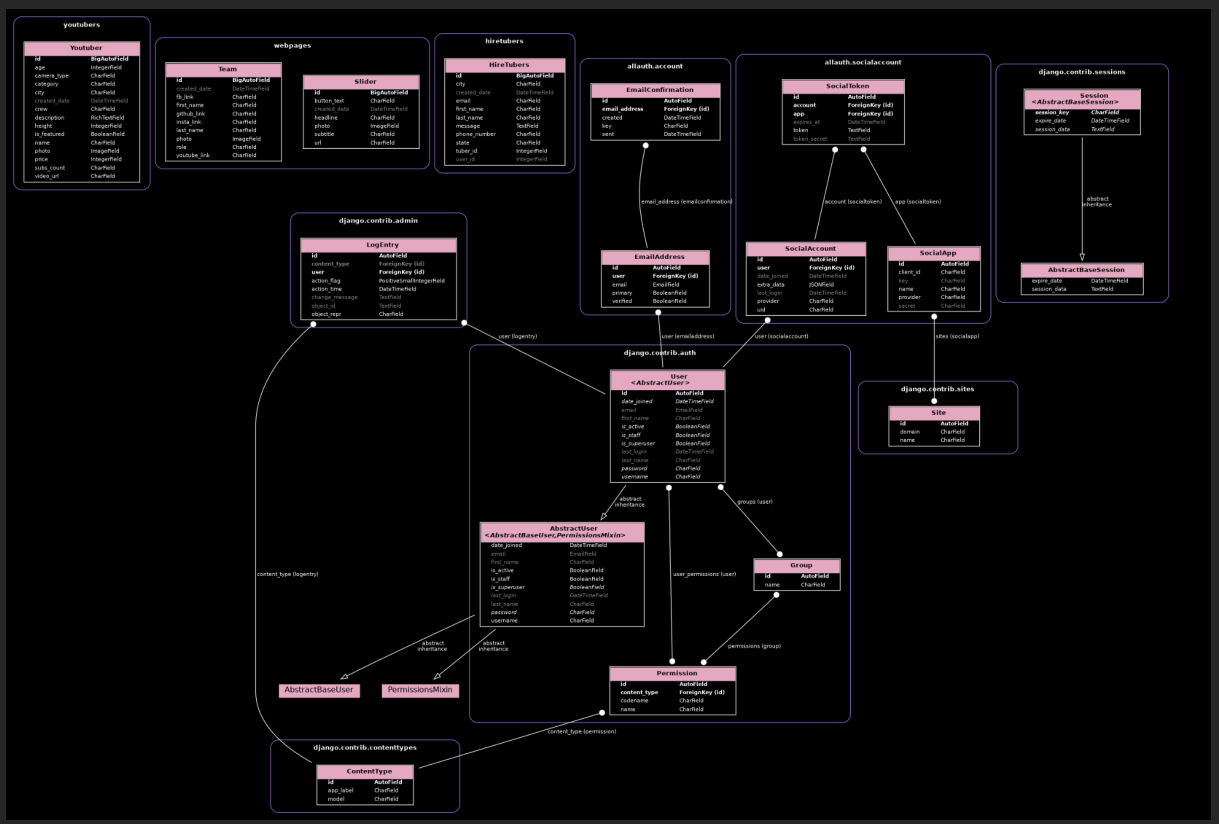
1.5 Out of Scope

Charge details for services such as database server and Azure App Service plan are not covered in this document. To understand pricing details about services, access Vendor portal.

2. Technical Specification

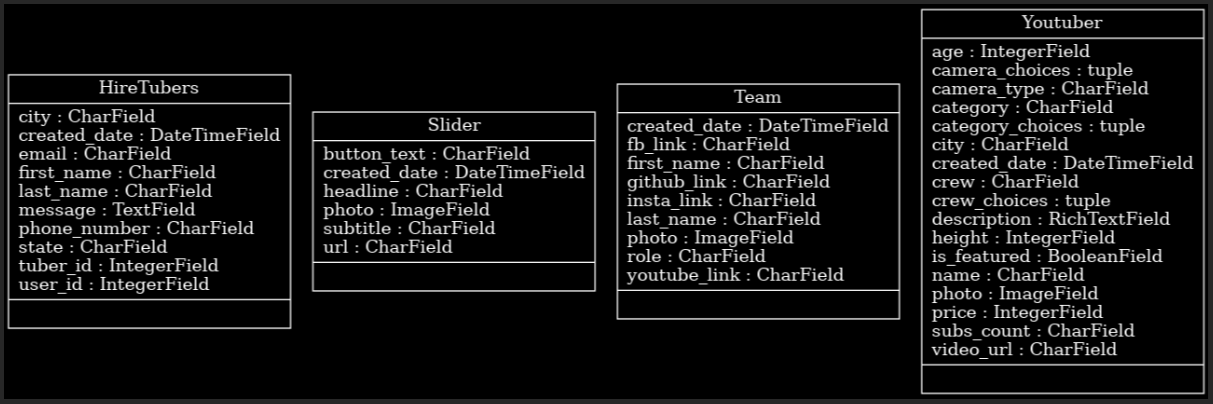
2.1 Database

System needs to store every request into the database and we need to store it in such a way that it is easy to retrive required information for debugging and investigation of certain activity. All user related, team related, and youtuber related data stored inside the database along with some of the UI part is driven by backenditself. Django supports session management and authorization and authentication at backend level means required information will be stored in the database. We have prepared a detailed database diagram.



2.2 Class Diagram

Class diagram is basically a graphical representation of the static view of the system and represents different aspects of the application. A collection of class diagrams represent the whole system.



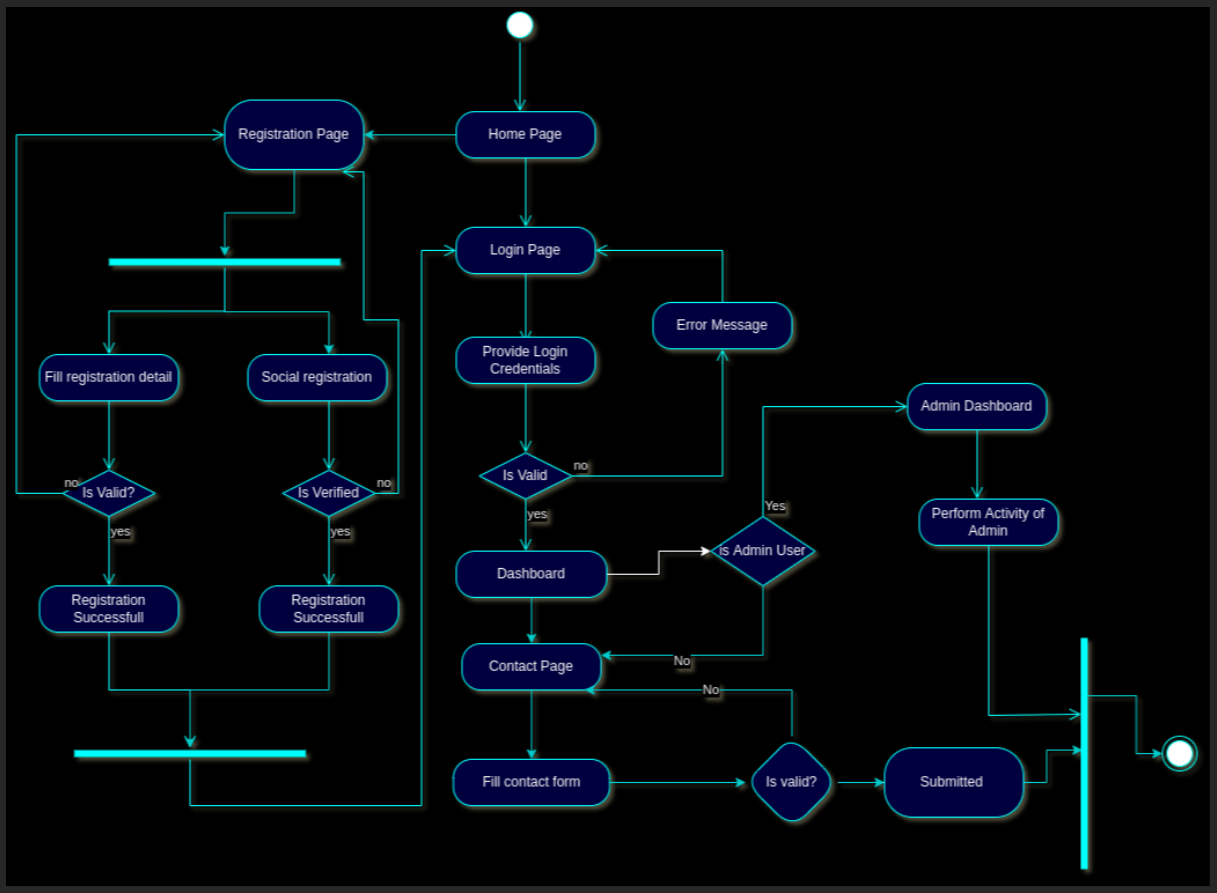
Based on analysis the tuber application team has identified four fundamental entities that need to be designed to complete the project.

Entities are HireTubers, Slider, Team and Youtuber.

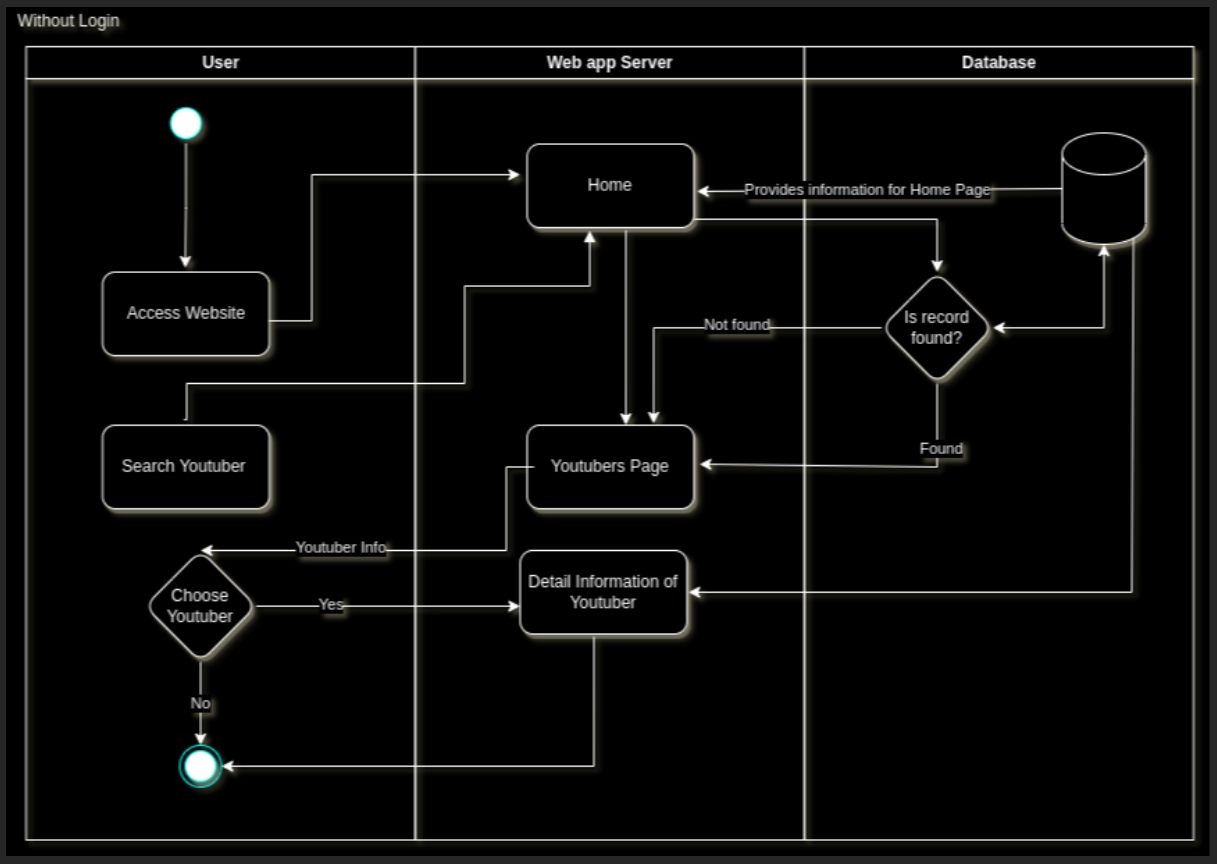
Each entity has some attributes that can be found in the above attached diagram.

2.3 Activity Diagram

Activity diagram is another important behavioural diagram in UML diagram to describe dynamic aspects of the system. Activity diagram is essentially an advanced version of flowchart that models the flow from one activity to another activity.



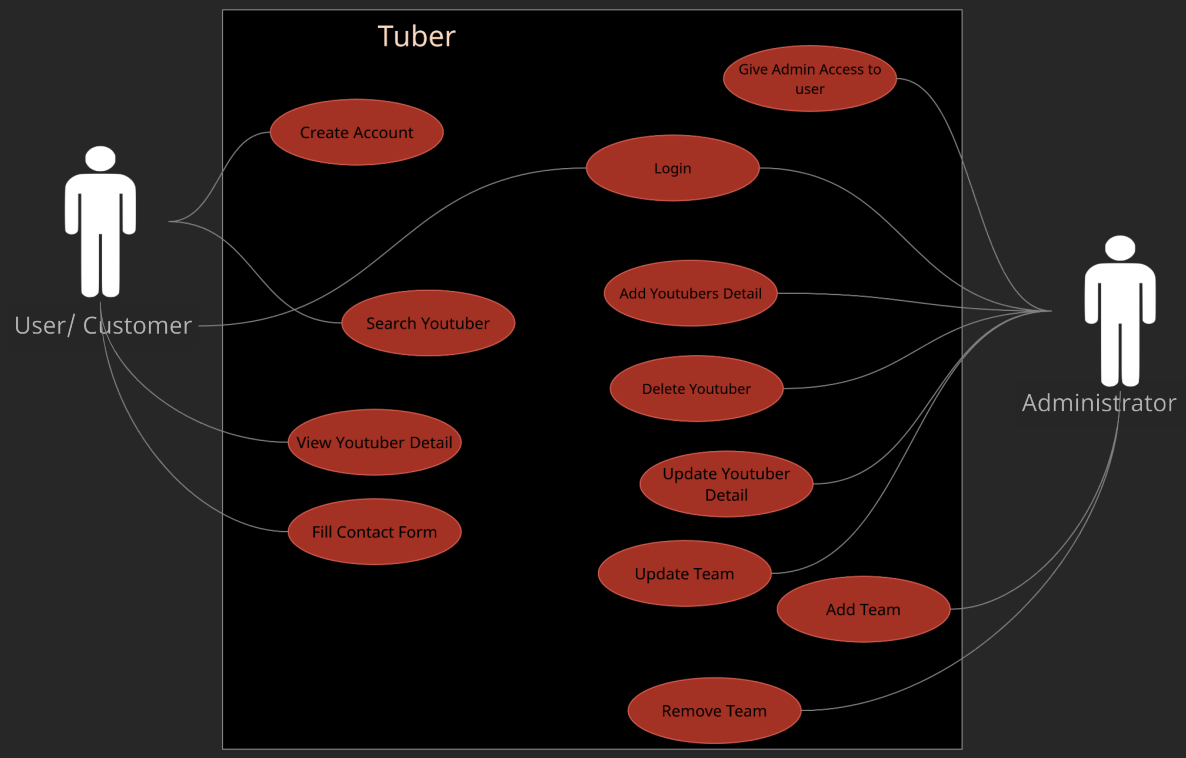
Above figure shows that any activity can be started from the home page. Account creation or sign in into account is possible from the home page. Accounts can either be created by registration form or by social login. If the logged user is admin then admin level access and functionality will be shown. Non-admin users can only search for youtubers and can submit contact forms.



2.4. Use Case Diagram

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well.

The figure below depicts different types of users along with their possible interaction with the system.



2.5. Logging

We should be able to log every activity done by the user.

- The System identifies at what step logging required
- The System should be able to log each and every system flow.
- Developers can choose logging methods. You can choose database logging/ File logging as well.
- System should not be hung even after using so many loggings. Logging just because we can easily debug issues so logging is mandatory to do.

3. Technology Stack

Frontend	HTML, CSS, JS and Bootstrap
Backend	Django
Programming language	Python
Database	PostgreSQL
Deployment	Azure App Service
Cloud	Microsoft Azure