Software Design Specification (SDS) Template

In the template, the parts in *italic* are parts that you are supposed to expand on. The parts in ***bold and italics*** are explanatory comments and are provided just for your understanding of the document.

Complete and tailor the document by expanding the relevant parts and removing explanatory comments as you go along.

Fill the Title and Author fields in the Properties menu with appropriate information.

Software Design Specification Document

“Relevance ranked QnA system for live YouTube's embedded stream on a RWA.”

<Team Name/Number>

Sr. No.	Name	Roll No.	Exam Seat NO.
1.	Ms. Anjali Chougule	40	1724618
2.	Ms. Janvi Pampattiwar	41	1724477
3.	Ms. Pragati Phand	42	1724480
4.	Ms. DurreAfshan Shaikh	43	1724765

CERTIFICATE

This is to certify that the Project entitled

Project title

Is

Submitted By

Sr. No.	Name	Roll No.	Exam Seat NO.
1.	Ms. Anjali Chougule	40	1724618
2.	Ms. Janvi Pampattiwar	41	1724477
3.	Ms. Pragati Phand	42	1724480
4.	Ms. DurreAfshan Shaikh	43	1724765

(Prof. P.S.R. Patnaik)
Project Guide Head

(Dr A.M. Pujar)
Dept of Computer Sc. & Engg.

(Dr. S. A. Halkude)
Principal

DEPARTMENT OF COMPUTER SCIENCE AND TECHNOLOGY

WALCHAND INSTITUTE OF TECHNOLOGY SOLAPUR

(2020-2021)

Table of content

1. Introduction

The following subsections of the Software Design Specifications (SDS) document should provide an overview of the entire SDS. The thing to keep in mind as you write this document is that you are telling how the system should do what it is supposed to do, so that the system can be implemented.

1.1 System Overview

This document provides a system overview of Software. It is best approach to implement the following features:

Embedding videos is just like creating backlinks to your site. Like in SEO (Search Engine Optimization), embedding your videos in a site behaves exactly like a back link and thus enabling your videos to get placed in search engine results and get more views. The more and more views your videos receive, your video popularity and brand image increase too. Not only video popularity, your product and site popularity too increase thus increasing your sales and profits.

Video embedding is the process of adding a video player to our website using an online video platform. For many website building or social media platforms, it is as simple as copy and pasting a link. Video embedding works by adding an embed code from your video hosting platform to the code of your website. It allows you to integrate live streaming on our website.

Relevance ranking is to be provided for live chat of the live video stream. Relevance ranking is the process of sorting the chats so that those Questions which are most likely to be relevant to the topic are shown at the top of the chat window with the answer.

1.2 Definitions, Acronyms, and Abbreviations

Provide the definitions of all terms, acronyms and abbreviations required to properly interpret the SDS

1.2.1 Definition:

Browser: A software application used to locate and display web pages.

Database: A database that stores data. It is a collection of interrelated data that contains information relevant to enterprise.

MYSQL: Most widely used query language for creating databases.

Internet: Worldwide networks of computers from where anyone can take information.

Homepage: The first page when you go to a worldwide website on the internet.

HTML: It is a computer language specifying the content & formats of web documents. It allows additional text to include codes that define fonts, layouts, graphics & hypertext.

CSS: It is language for describing presentation of Web pages, including colors, layout, and fonts. It allows one to adapt the presentation to different types of devices, such as large screens, small screen.

JavaScript: It is used to program the behaviour of web pages. JavaScript is a text-based programming language used both on the client-side and server-side that allows you to make web pages interactive.

PHP: It is called Hypertext Pre-processor.

User Interface (UI) Design: It creates fewer problems, increases user involvement, perfects functionality and creates a strong link between your customers and your website.

User Experience (UX) Design: It is about making your product/website predictable and easy to use. It's about offering a design that does not confuse the user when they proceed with their intended actions.

Webpage: Pages of information placed on a network that may contain text, graphics, images, moving pictures, sound files & other types of electronic information.

Website: Collection of files called webpage, which can contain text & images.

Python: One of the most commonly used programming languages.

Algorithms: A set of rules or instructions given to an AI, neural network, or other machines to help it learn on its own; classification, clustering, recommendation, and regression are four of the most popular types

Machine learning: Using example data or experience to refine how computers make predictions or perform a task. A facet of AI that focuses on algorithms, allowing machines to learn without being programmed and change when exposed to new data.

Module: It encapsulates a specific machine learning algorithm, function, or code library in the workspace

1.2.2 Abbreviations:

Sr. No.	Term	Description
1.	MYSQL	Structured Query Language.
2.	HTML	HyperText Markup Language
3.	CSS	Cascading Style Sheets
4.	JS	JavaScript
5.	PHP	Hypertext Pre-Processor
6.	UI	User Interface
7.	UX	User Experience
8.	DB	Database
9.	ML	Machine Learning

1.3 References

List any references or related materials here.

In this section:

(1) Provide a complete list of all documents referenced elsewhere in the SDS
(2) Identify the document by title, report number (if applicable), date, and publishing organization
(3) Specify the sources from which the references can be obtained

This information can be provided by reference to an appendix or to another document.

1. https://www.quora.com/How-can-I-embed-a-live-stream-with-a-live-chat-on-my-website-without-leaving-my-website
2. https://www.peggyktc.com/2016/05/embed-livestream-and-chat.html?m=1
3. https://youtu.be/s5xzhKZtTsw
4. https://youtu.be/WL11dCzHDuM
5. https://www.mockplus.com/blog/post/interaction-design-principles
6. https://www.dacast.com/blog/how-to-embed-streaming-video-to-your-website/
7. https://thechurchco.com/blog/2020/03/17/live-streaming-with-youtube/
8. https://www.youtube.com/watch?v=9Za8glNbhcw

1.4 Document Map

Define all major sections of this document and provide a one-sentence summary of each.

Section 1: Introduction: Overview of project is explained with some referred documentation.

Section 2: Design Consideration: Minimum requirements of hardware and software is mentioned along with limitation and assumptions considered in developing the system.

Section 3: Architectural (High-Level) Design: Detailed designed view of interaction diagram and deployment diagram in a logical view of the system explained.

Section 4: Low Level Design: Detailed internal structure of the system along with its components.

Section 5: User Interface Design: Explains how user is going to interact with Website.

2. Design Considerations

These subsections describe issues that need to be addressed or resolved prior to or while completing the design as well as issues that may influence the design process.

2.1 Assumptions:

Describe any assumption, background, or dependencies of the software, its use, the operational environment, or significant project issues. These are things that you are assuming to be true and that directly affect the design.

1. Smooth and uninterrupted internet connectivity required.
2. The input provided by the user is assumed to be correct.
3. Preprocessed data is good and correct for ML up to its extent.
4. The input (URL) provided by the user helps to know the correct result.

2.2 Constraints:

Describe any constraints on the system that have a significant impact on the design of the system. (e.g., technology constraints, performance requirements, end user characteristics). These are things the customer has told you that directly influence the design (e.g., the DB must be an open-source, freely available DBMS).

1. DB must be open source.
2. Users are knowledgeable enough that they can put the correct URL up to its best extent.

2.3 System Environment:

Describe the hardware and software that the system must operate in and interact with.

2.3.1 Hardware Requirements:

1. Ethernet connection (LAN) or a wireless adaptor (Wi-Fi)
2. Proper working Microphone

2.3.2 Software Requirements:

1. Browser (Internet Explorer, Firefox, Google Chrome)

2.3.3 Operating System Requirement:

1. Processor: 64 - bit Operating System, x64 - based processor, 2.5 GHz Quad Core, MAC, Linux.

2. RAM: 8 GB

2.4 Design Methodology:

Summarize the approach that will be used to create and evolve the design for this system. This is not a rehash of your project lifecycle or change-management plan. This is for stating whether you will use object-oriented design, formal specifications, or other specific methodologies. Most people will use some object-oriented technique with UML.

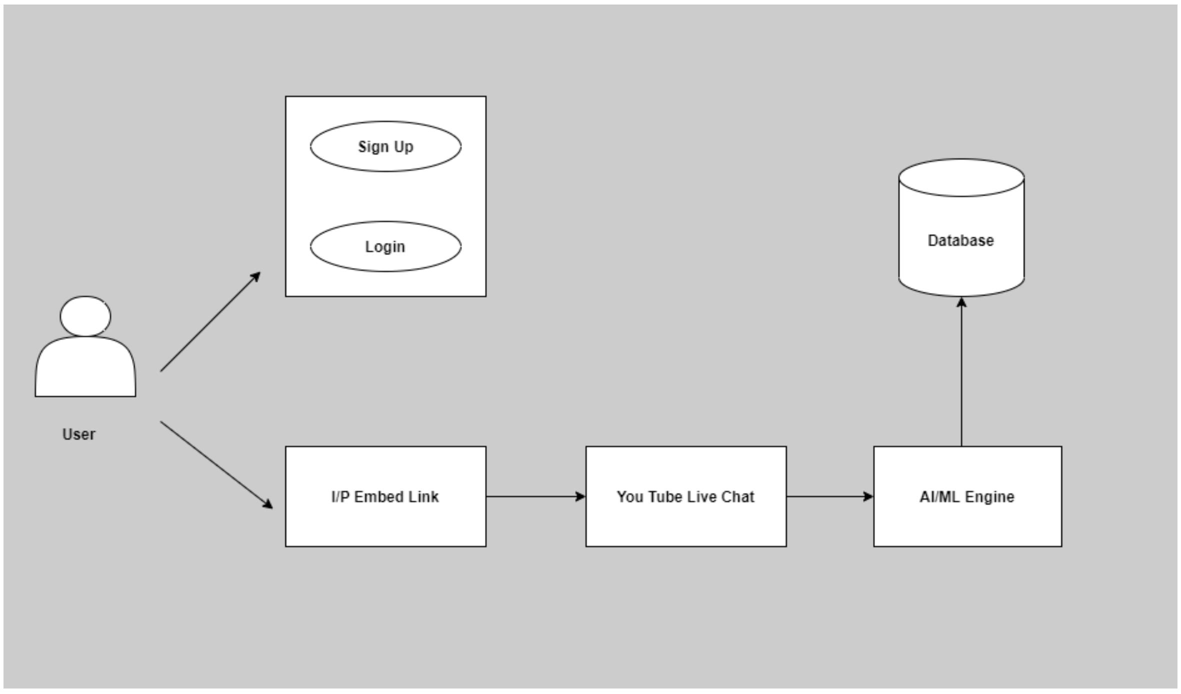


Fig. Use Case Diagram

3. Architectural (High-Level) Design

The architecture provides the top-level design view of a system and provides a basis for more detailed design work. These subsections describe the top-level components of the system you are building and their relationships. For an OO implementation in Java, for example, our components could become packages (or set of packages, depending on the level of granularity considered and the size of the system).

In defining your architectural design, you can follow one of the organizational styles seen in class (shared data repository, shared services and servers, and abstract machine/layered) or pick a different one if none of those is appropriate for your system.

3.1 Overview

This section provides a high-level overview of the structural and functional decomposition of the system. The section should list the different components and concisely discuss the major responsibilities and roles such components must play.

Front End: The front end will take the link as input from user.

Database: Use to store data.

Data Pre-Processing: For transforming the raw data into useful format, there are four steps-Data cleaning, Data integration, Data transformation, Data reduction.

ML Module: After training of the model, it should give questions closely related to topic with minimal risk.

3.2 Rationale

This section discusses why you are using the architecture you have chosen.

The above system will help us to give the output which satisfies the project goal and implement the tasks efficiently. Also, this architecture increases scope to extend and reuse the project.

3.3 Conceptual (or Logical) View

This section should provide and describe a diagram that shows the various components and how they are connected. The conceptual view shows the logical/functional components of the system, where each component represents a cluster of related functionalities. For UML, this would typically be a component diagram or a package diagram.

Front End: User gives the input through a web page.

Input: Embedded link is given by user.

DB: Live Chat is stored in a database.

Raw Data: Dataset: to train the ML module.

Data Pre-Processing: It is a data mining technique used to transform the raw data in a useful and efficient format.

Dataset: A data set is a collection of related data that is used for training ML modules.

Machine Learning: Training to sort data related to given topic and give only data closely related to topic.

ML Module: After training, the ML module is ready for testing.

Output: The responsive website to embed the live YouTube video and display the questions closely related to the video topic.

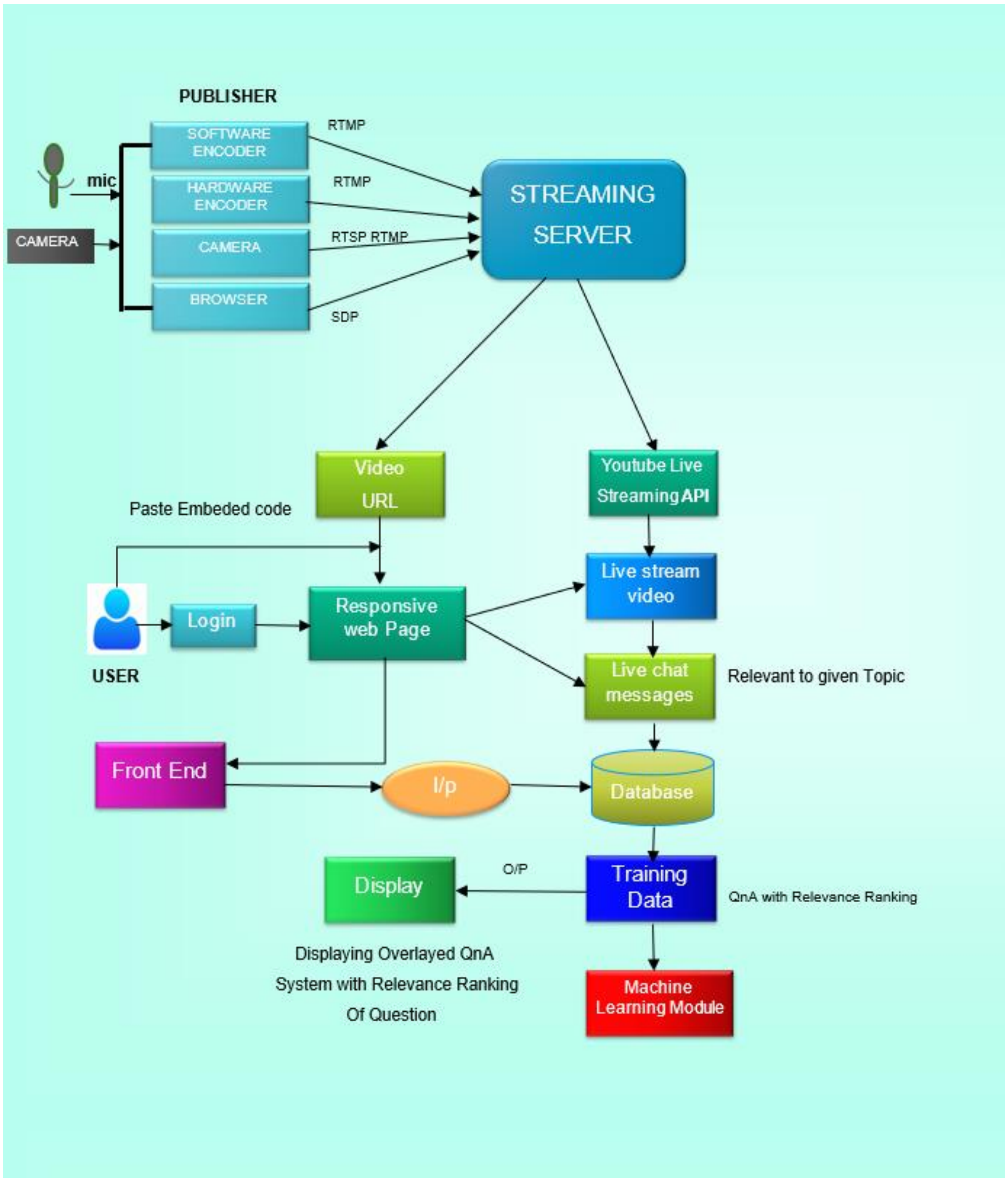


Fig. High-Level System Architecture

- *RTMP is a live streaming protocol that transmits video files from the encoder to an OVP. Most encoders use the RTMP stream format. RTMP stands for Real-Time Messaging Protocol. RTMP streaming is a delivery method designed for live-streaming.*
- *Real-Time Messaging Protocol (RTMP) is an open source protocol owned by Adobe that's designed to stream audio and video by maintaining low latency connections. Clients use a handshake to form a connection with an RTMP server which then allows users to stream video and audio.*
- *The Session Description Protocol (SDP) is a format for describing*

multimedia communication sessions for the purposes of session announcement and session invitation. Its predominant use is in support of streaming media applications, such as voice over IP (VoIP) and video conferencing.

- *The Real Time Streaming Protocol (RTSP) is a network control protocol designed for use in entertainment and communications systems to control streaming media servers. The protocol is used for establishing and controlling media sessions between endpoints. Clients of media servers issue VHS-style commands, such as play, record and pause, to facilitate real-time control of the media streaming from the server to a client (Video on Demand) or from a client to the server (Voice Recording).*

3.4 Other Views

High-level designs are most effective if they attempt to model groups of system elements from a number of different views. Beside the Conceptual/Logical view, examples of additional viewpoints are:

(a) ***Process View: this represents the runtime view of the system. The components are threads, processes, or distributed components. In UML, this would typically be a process interaction diagram.***

(b) ***Physical View: this view is for distributed systems. The components are physical processors that have parts of the system running on them. For UML, this would be a deployment diagram.***

Note that it is not necessary to document all these views. For many smaller applications, the conceptual view is all that is necessary. Document those views that will help you design and implement the system and create a subsection for each one of them.

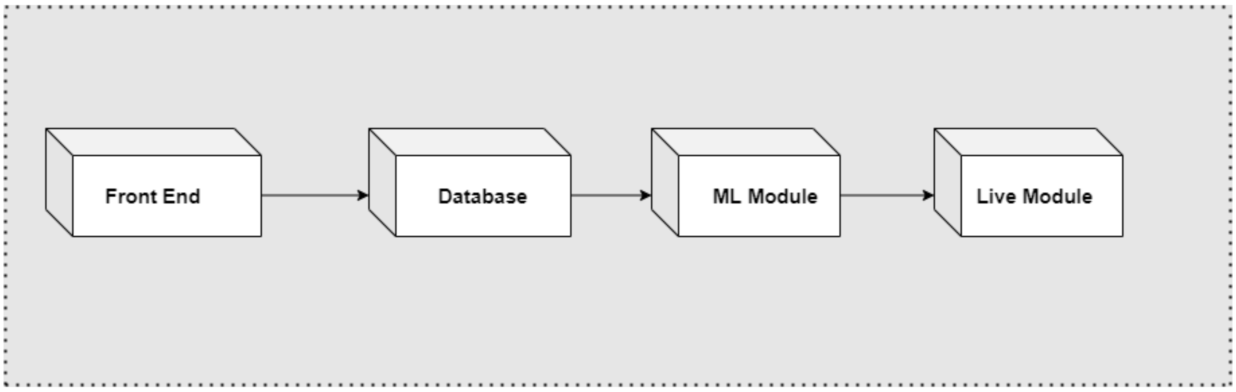


Fig. Deployment Diagram

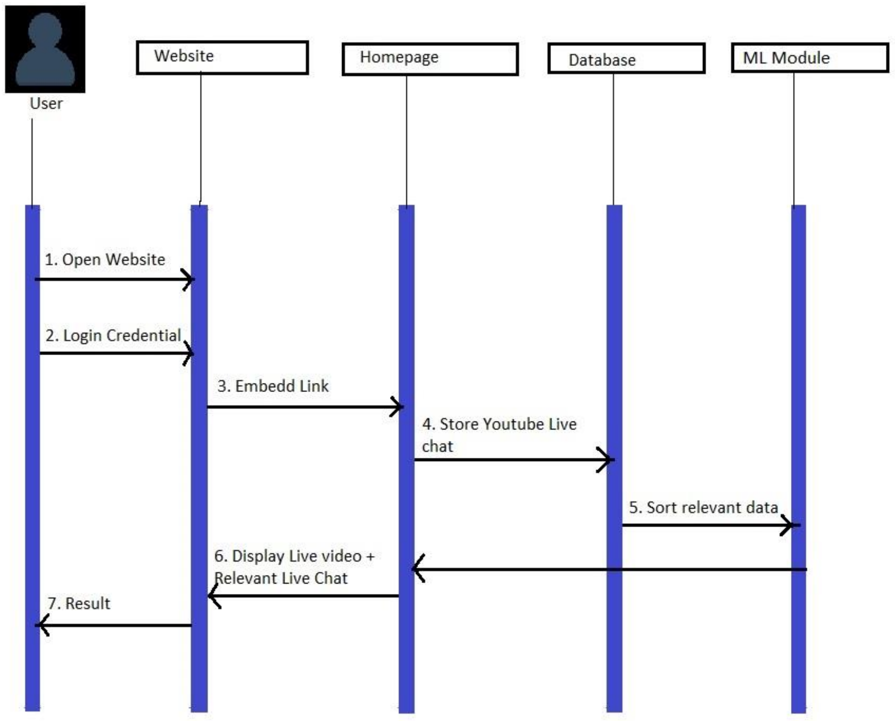


Fig. Sequence View

4. Low Level Design

This section provides the low-level design for each of the system components identified in the previous section. For each component, you should provide a subsection that shows its internal structure. In the case of an OO design, this internal structure would typically be expressed as an UML class diagram that represents the static class structure for the component. For smaller systems, you may have a single UML class diagram that each component description refers to.
As discussed above, these subsections should provide and discuss detailed diagrams of each software module. For at least some of the components, you should provide diagrams that show a dynamic view of the component internals (i.e., that show the dynamic interaction between classes). In the case of an OO design, UML state or interaction diagrams can be used to this end.

4.1 Component 1

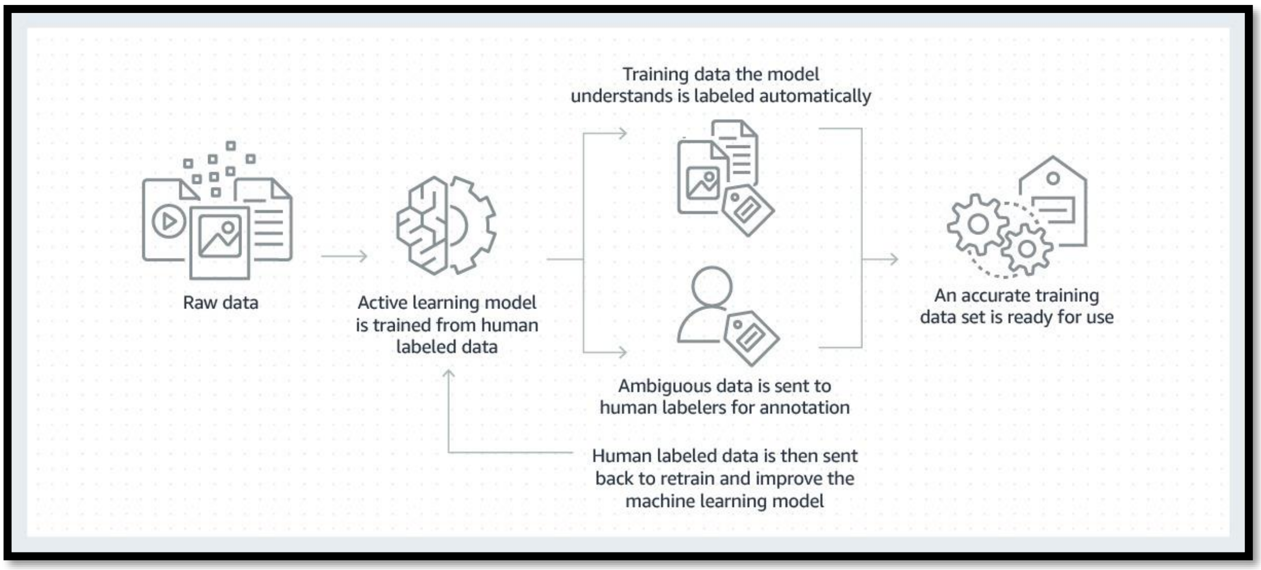

Raw data
Active learning model is trained from human labeled data
Training data the model understands is labeled automatically
Ambiguous data is sent to human labelers for annotation
An accurate training data set is ready for use
Human labeled data is then sent back to retrain and improve the machine learning model

Fig. Data Annotation

4.2 Component 2

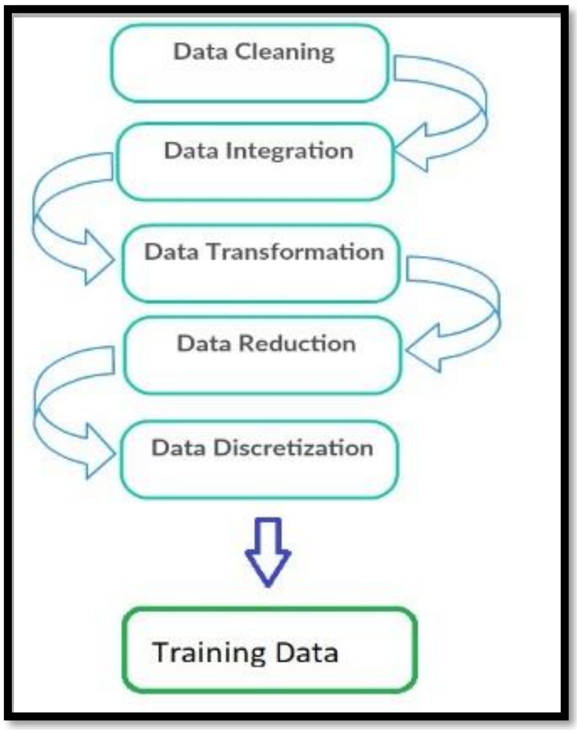


Fig. Data Pre-Processing

4.3 Component 3

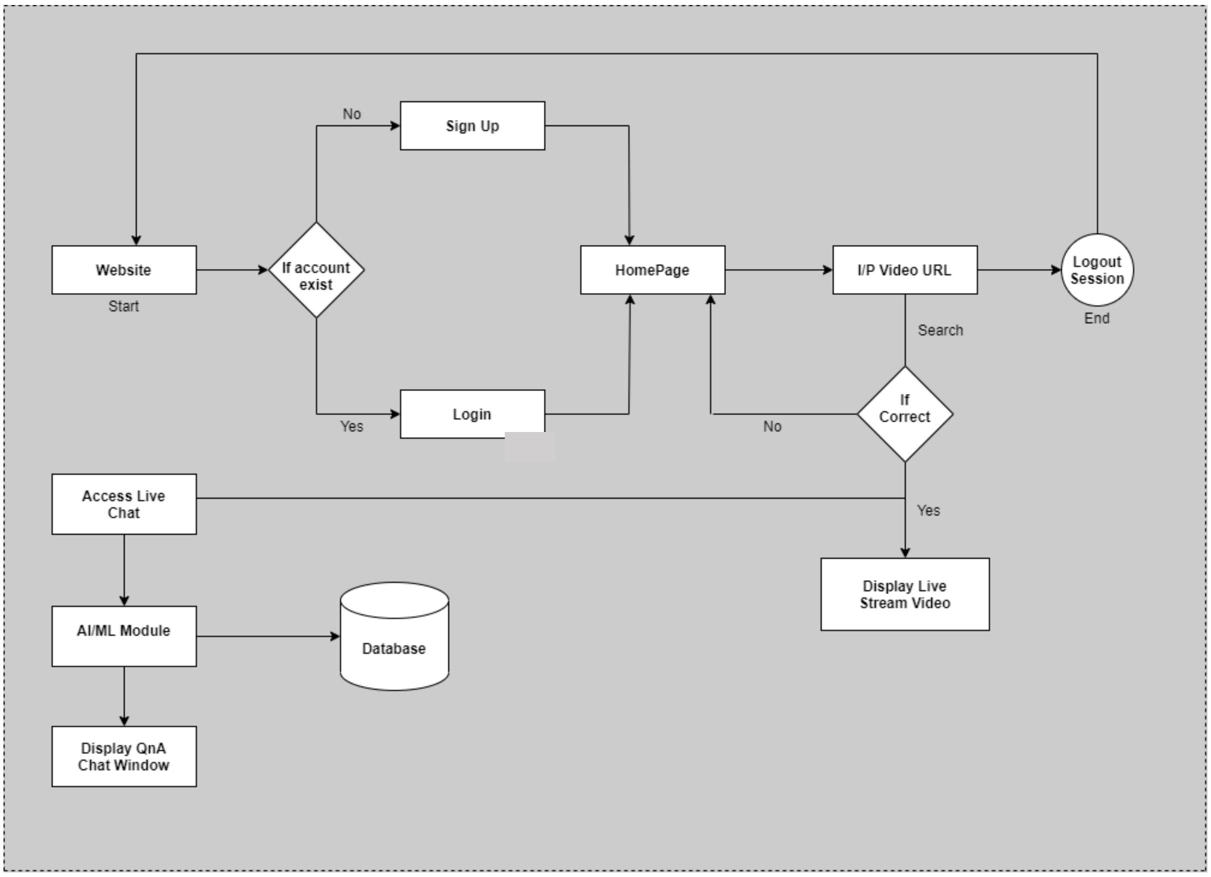


Fig. Activity Diagram

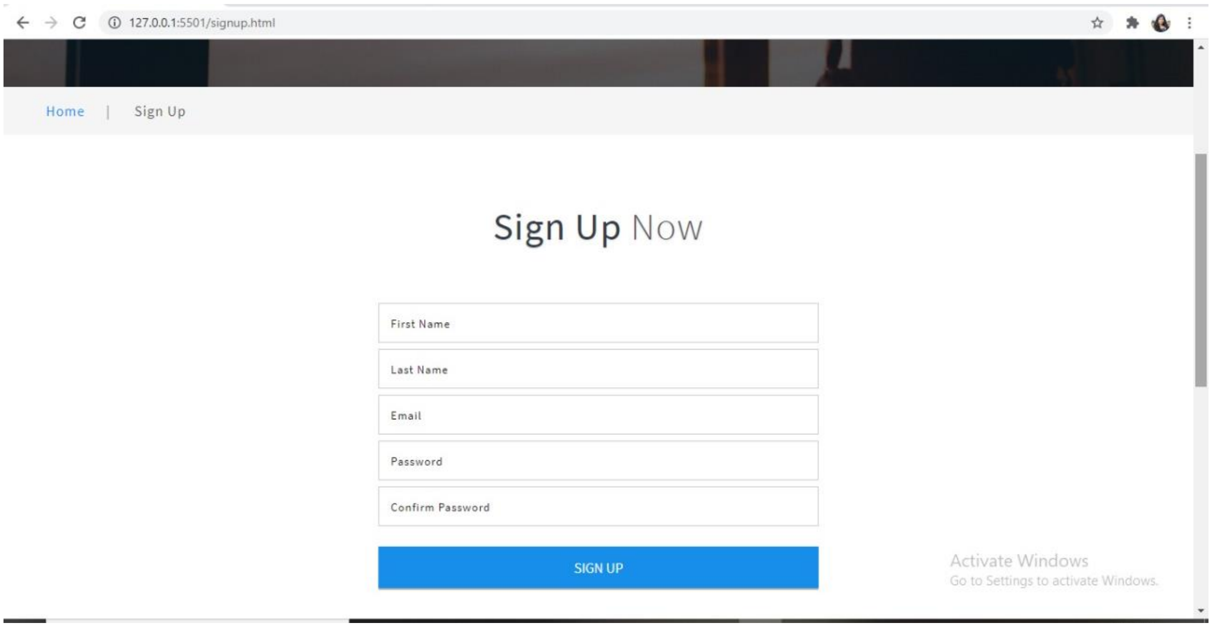
5. User Interface Design

These subsections discuss the user interface design.

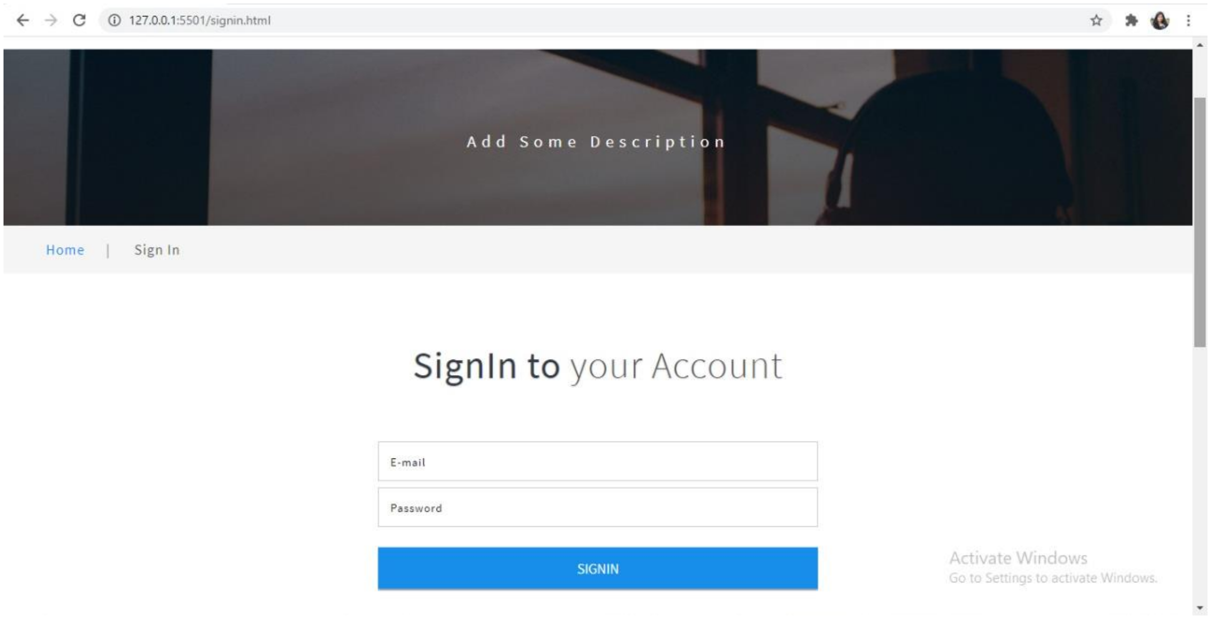
5.1 Application Control

This section details the common behaviour that all screens will have. Common look and feel details, such as menus, popup menus, toolbars, status bars, title bars, drag and drop, and mouse behaviour should be described here.

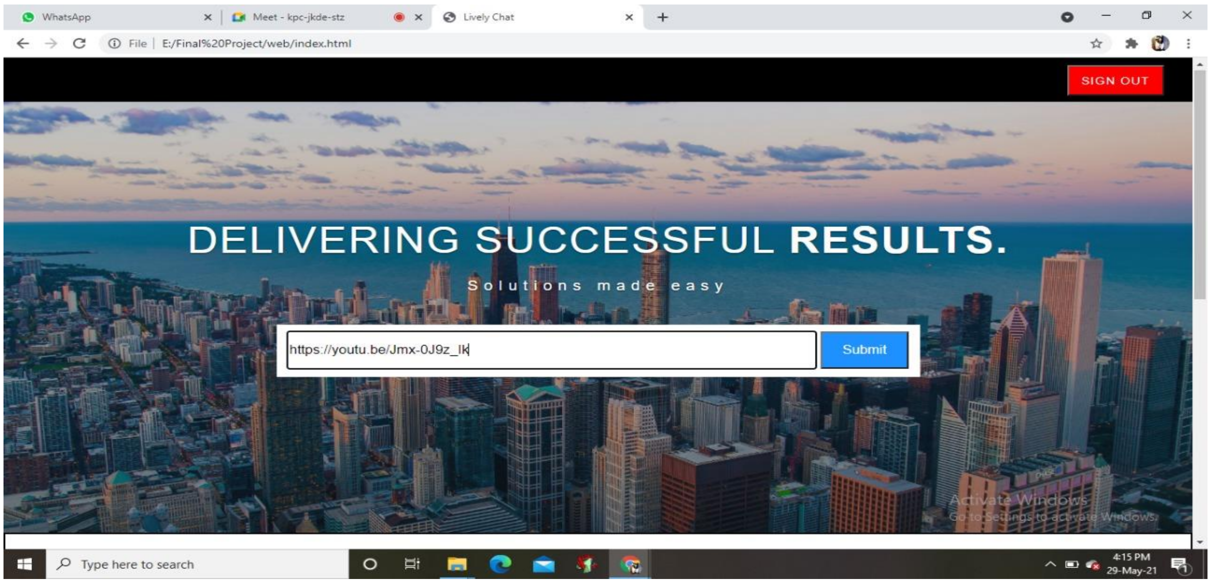
5.2 Screen 1



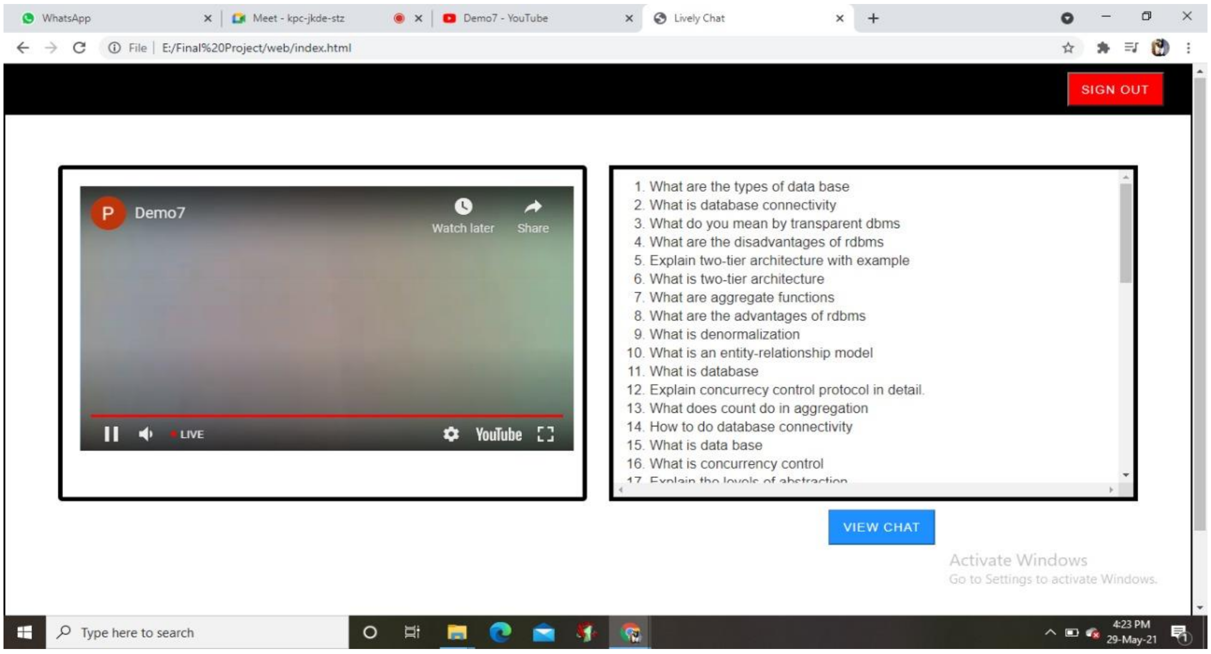
5.3 Screen 2



5.4 Screen 3



5.5 Screen 4



1. What are the types of data base
2. What is database connectivity
3. What do you mean by transparent dbms
4. What are the disadvantages of rdbms
5. Explain two-tier architecture with example
6. What is two-tier architecture
7. What are aggregate functions
8. What are the advantages of rdbms
9. What is denormalization
10. What is an entity-relationship model
11. What is database
12. Explain concurrency control protocol in detail.
13. What does count do in aggregation
14. How to do database connectivity
15. What is data base
16. What is concurrency control
17. Explain the levels of abstraction

These sections illustrate all major user-interface screens and describe the behaviour and state changes that the user will experience. A screen transition diagram or table can optionally be created to illustrate the flow of control through the various screens.

Note that these sections may not show actual screenshots (in case you have not completed the implementation yet). In these cases, they can be drawings or mock-ups created using some rapid GUI-building tool like Open source Pencil Project https://pencil.evolus.vn/