
Group09

**Video web
Software Architecture Document**

Version 1.1

Video web	Version: 1.1
Software Architecture Document	Date: 30/07/2022

Revision History

Date	Version	Description	Author
16/07/2022	1.0	<ul style="list-style-type: none"> These sections are fulfilled with limited information from Group09, in particular: <ul style="list-style-type: none"> Introduction: <i>Dương Đình Hiếu</i> Architectural Goals and Constraints: <i>Dương Đình Hiếu</i> Project Organization: <i>Nguyễn Trương Hoàng Thái</i> Use-case model: <i>Trương Quốc Huân, Nguyễn Trương Hoàng Thái, Phan Thanh Thúy</i> Logical View: <ul style="list-style-type: none"> ❖ Component Sign up: <i>Dương Đình Hiếu, Đỗ Phương Nam</i> ❖ Component Login: <i>Dương Đình Hiếu, Đỗ Phương Nam</i> ❖ Component Video interact: <i>Nguyễn Trương Hoàng Thái, Trương Quốc Huân, Phan Thanh Thúy</i> ❖ Component Account method: <i>Nguyễn Trương Hoàng Thái, Trương Quốc Huân, Phan Thanh Thúy</i> ❖ This version will be upgraded with more information in future. 	<ul style="list-style-type: none"> ❖ Nguyễn Trương Hoàng Thái 20127625 ❖ Trương Quốc Huân 20127172 ❖ Đỗ Phương Nam 20127418 ❖ Phan Thanh Thúy 20127350 ❖ Dương Đình Hiếu 20127675
30/07/2022	1.1	<ul style="list-style-type: none"> These sections are fulfilled with limited information from Group09, in particular: <ul style="list-style-type: none"> Deployment: <i>Nguyễn Trương Hoàng Thái, Trương Quốc Huân</i> Implementation View: <i>Nguyễn Trương Hoàng Thái, Trương Quốc Huân</i> Classify classes (each class is corresponded to a component in item 4.0): <ul style="list-style-type: none"> Nguyễn Trương Hoàng Thái 	<ul style="list-style-type: none"> ❖ Nguyễn Trương Hoàng Thái 20127625 ❖ Trương Quốc Huân 20127172

Video web	Version: 1.1
Software Architecture Document	Date: 30/07/2022

Table of Contents

Introduction	4
Architectural Goals and Constraints	4
Use-Case Model	5
Logical View	6
Component: Sign up	6
Component: Login	7
Component: Video interact	8
Component: Account method	9
Component: Uploader functions	11
Deployment	12
Implementation View	13

Video web	Version: 1.1
Software Architecture Document	Date: 30/07/2022

Software Architecture Document

1. Introduction

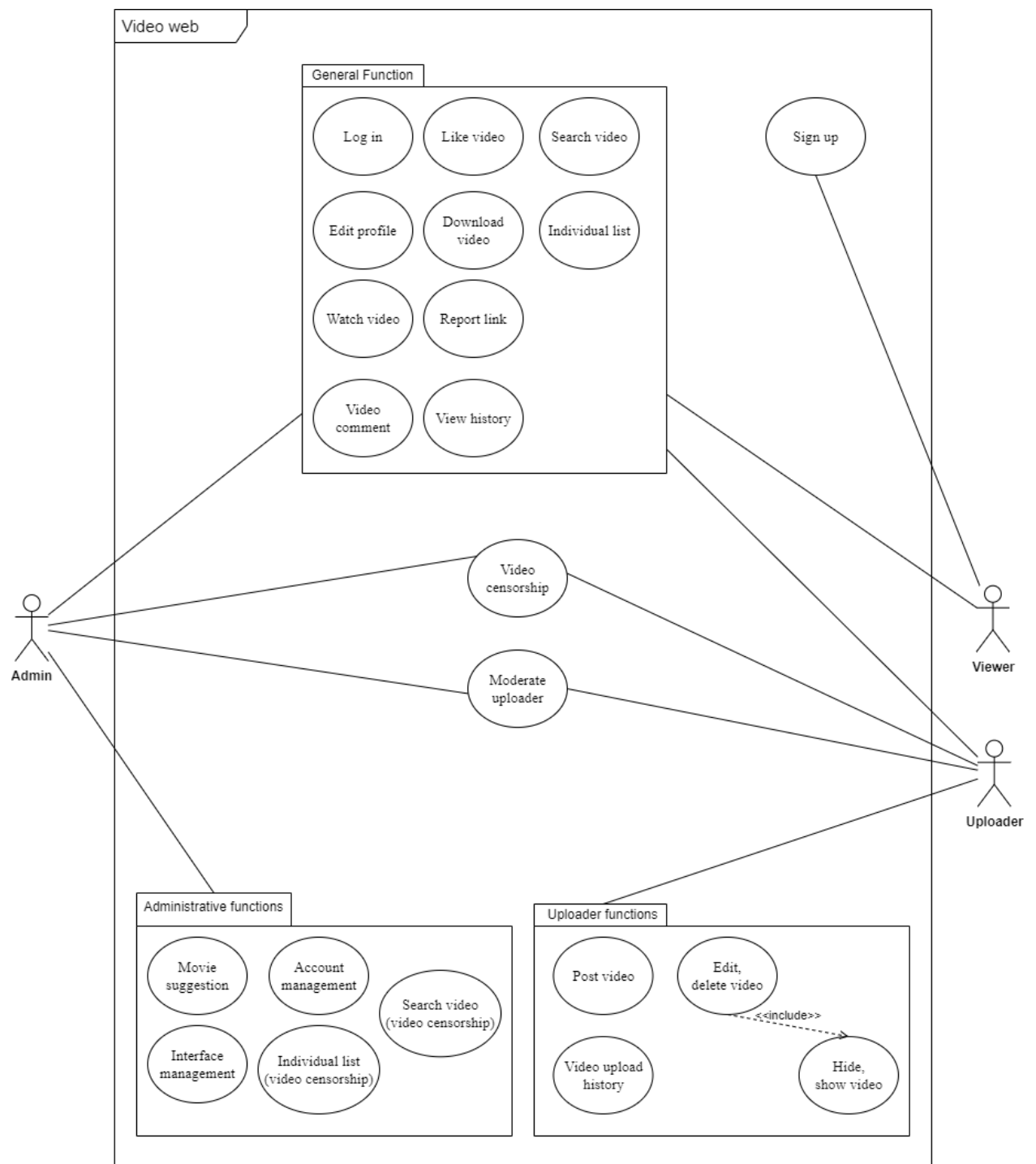
- This document provides an architectural overview of the video web application by presenting the architecture as a use case diagram, deployment view, logical view (with components view) and implementation view. These views are presented using the use case diagram and class diagram.

2. Architectural Goals and Constraints

- Users can access images, audio, and video.
- Website accessible 24/7 (except during maintenance).
- The maximum search speed is 10s, if it can't load, it will report a page error.

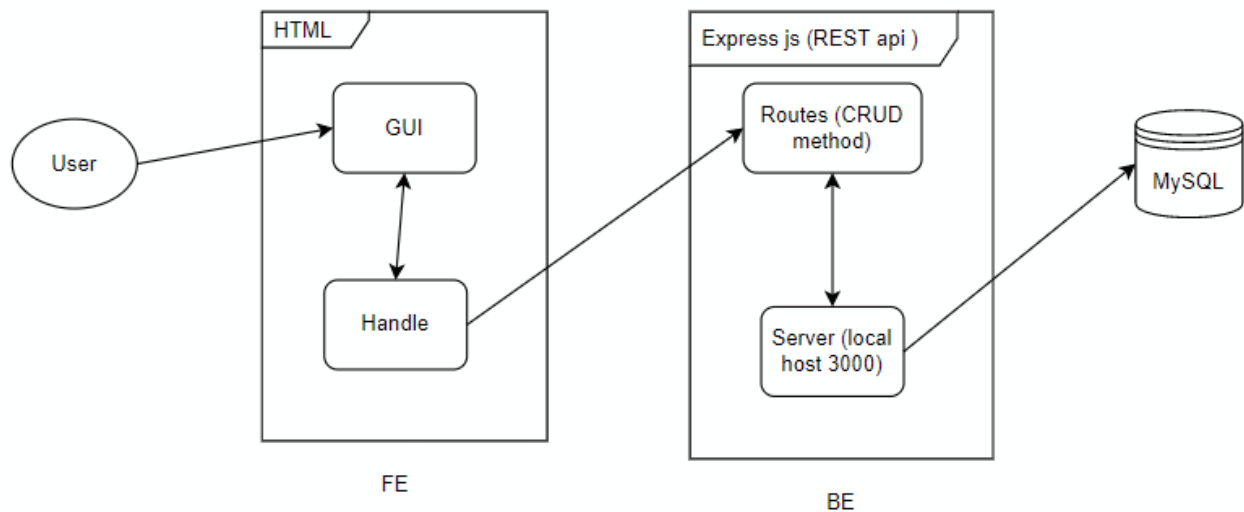
Video web	Version: 1.1
Software Architecture Document	Date: 30/07/2022

3. Use-Case Model



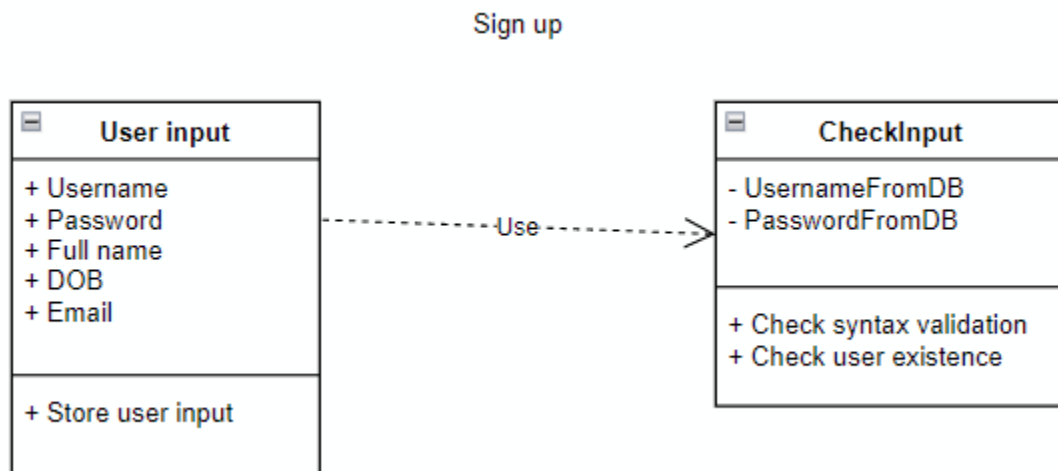
Video web	Version: 1.1
Software Architecture Document	Date: 30/07/2022

4. Logical View



- Example with account registration function:
 - Users will see on the interface (GUI) a REGISTER button so they can click (Handle) to register an account
 - After clicking the subscribe button, HTML through the API to use methods to help subscribers to the DB

4.1 Component: Sign up

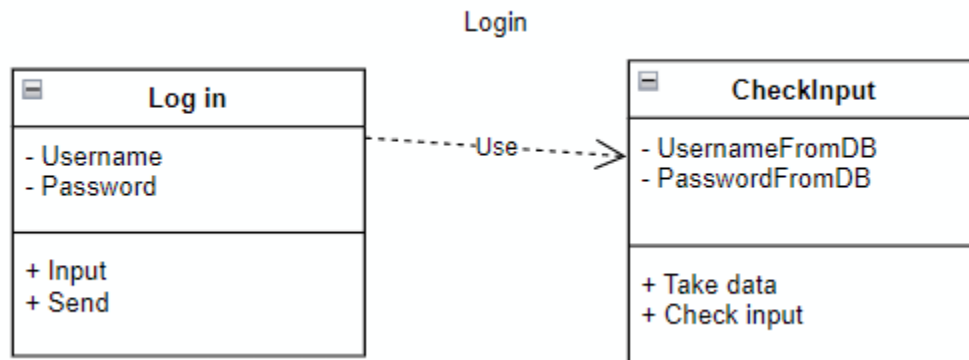


Class	Description
GUI: User input	<ul style="list-style-type: none"> • Attributes: Username, Password, DOB, Full name, Email (store user input information) • Method: Store user input (save user input data)
Routes: CheckInput	<ul style="list-style-type: none"> • Attributes: UsernameFromDB and PasswordFromDB (data stored in DB)

Video web	Version: 1.1
Software Architecture Document	Date: 30/07/2022

	<ul style="list-style-type: none"> Methods: Check syntax validation (check whether the user's syntax is correct to allow characters) and Check user existence (check whether user data already exists in the DB)
--	---

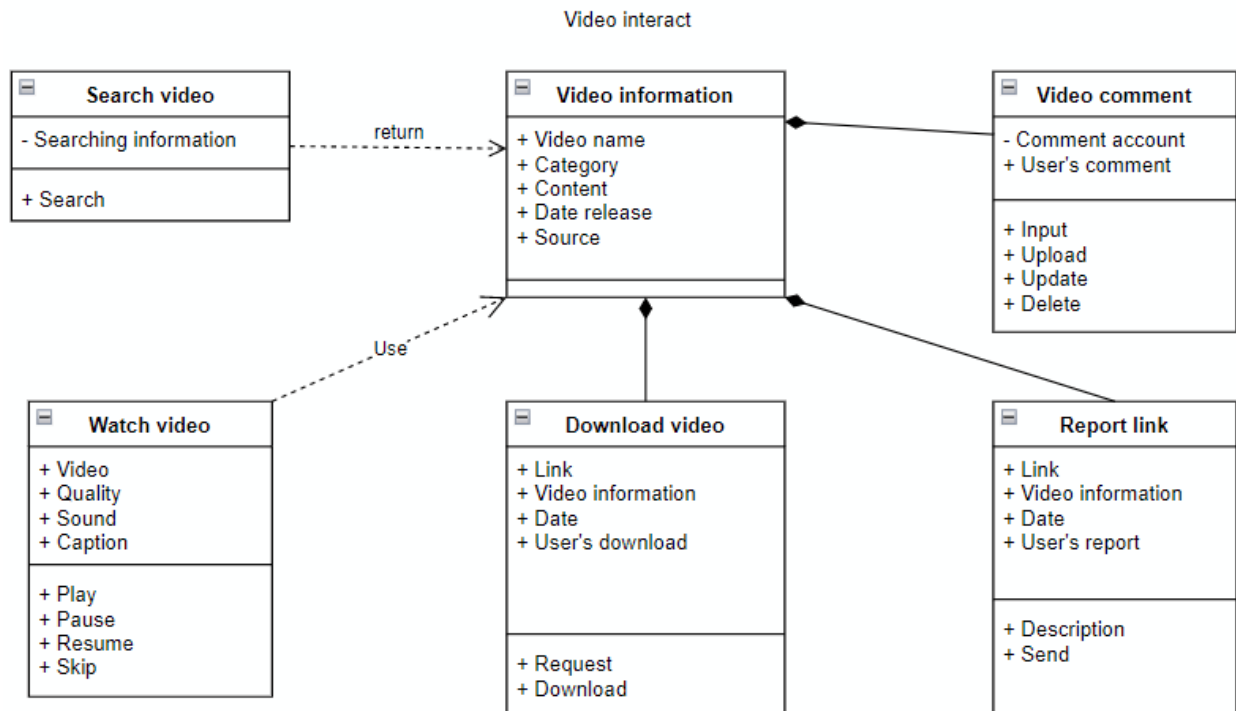
4.2 Component: Login



Class	Description
GUI: Log in	<ul style="list-style-type: none"> Attributes: username, password Methods: Input (receive data that the user enters the account and password), Send (to send the information that has just been input to CheckInput)
Routes: CheckInput	<ul style="list-style-type: none"> Attributes: UsernameFromDB and PasswordFromDB (account data and passwords in the Database have been previously stored) Methods: take data (get account and password information from DB) and check input (to compare user's account and password data entered with data in database)

Video web	Version: 1.1
Software Architecture Document	Date: 30/07/2022

4.3 Component: Video interact

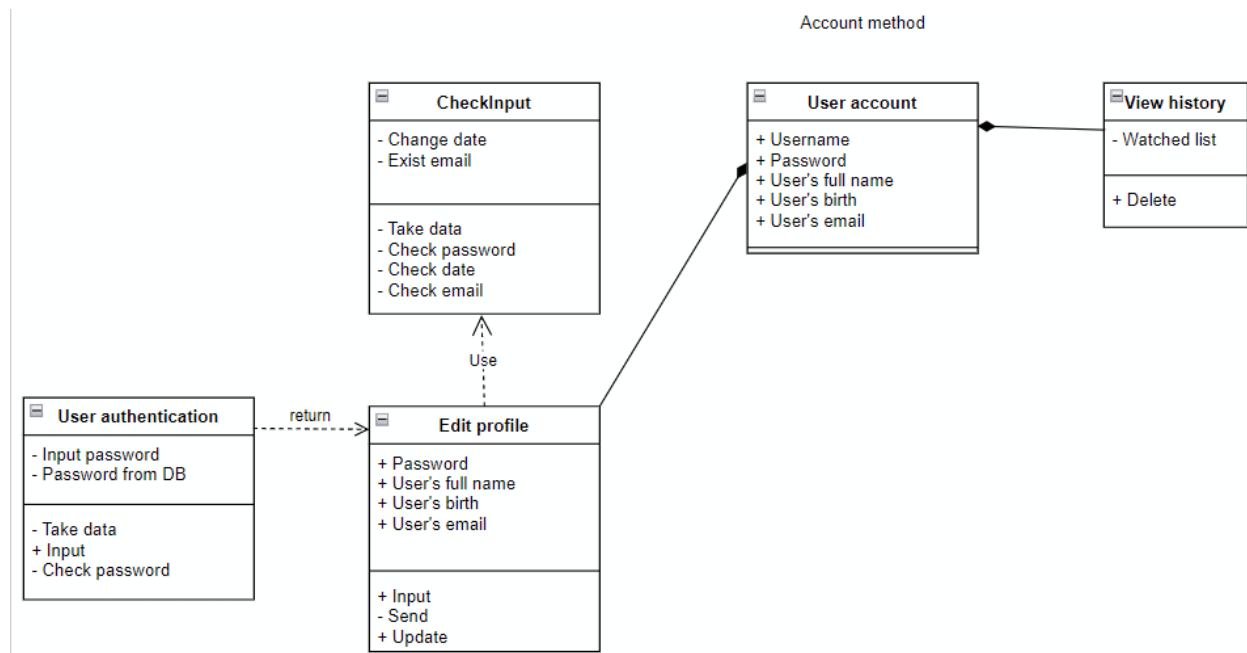


Class	Description
GUI: Search video	<ul style="list-style-type: none"> Attributes: searching information (store information such as video name, video content... to serve video search) Methods: Search (search for videos by the value of searching information)
MySQL: Video information	<ul style="list-style-type: none"> Attributes: Video name , Category , Content , Date release , Source (source of video)
GUI: Video comment	<ul style="list-style-type: none"> Attributes: Comment account (commentator's information (Full name, avatar (if any))) and User's comment (content that person comments) Methods: Input (to receive data that the user enters comment content), Upload (to post comments), Update (to edit comments), Delele (to delete comments)
GUI: Watch video	<ul style="list-style-type: none"> Attributes: Video, Quality (resolution), Sound (sound accompanied by video) and Caption (its subtitles (if any)). Methods: Play (to play the video), Pause (to pause the video), Resume (to resume playing the video if it has just been paused) and finally Skip (to skip to the next video (if available) in the queue or playback previous video (if any))

Video web	Version: 1.1
Software Architecture Document	Date: 30/07/2022

GUI: Download video	<ul style="list-style-type: none"> Attributes: Video link (URL link of the video to download), Video information (content, summary, ..), Date (date when the video was downloaded), User's download (account information of the downloader) methods : Request (send a request to the server to receive the download link), after receiving the link we have DownLoad (to download that video to the device)
GUI: Report link	<ul style="list-style-type: none"> Attributes: Video link (the URL link of the video is broken and needs to be reported), Video information (content, summary, ..), Date (date on which the video was reported), User's report (account information of the reporter).) methods : Description (describe the error message about the video), Send (send Description and link to Admin)

4.4 Component: Account method



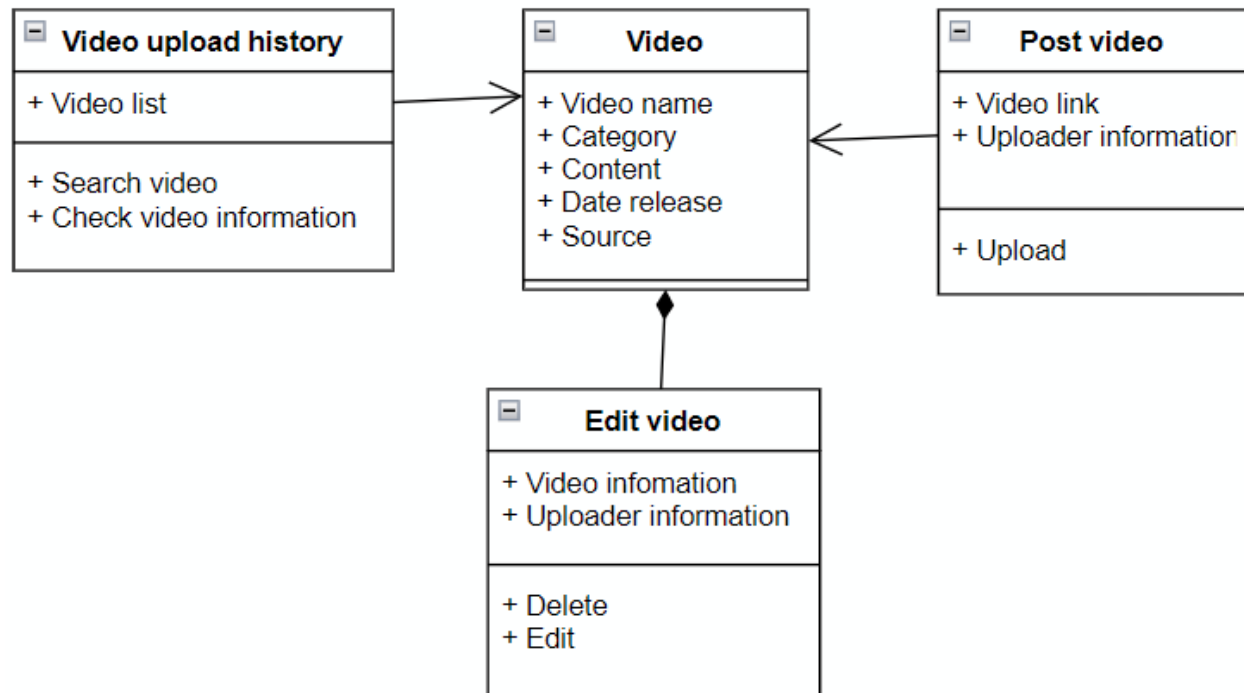
Class	Description
GUI: User authentication	<ul style="list-style-type: none"> Attributes: Input password (user's current password) and Password from DB (that account's password is already stored in DB) Methods: Input (receive data that the user enters), Take data (get data from the DB to use for checkpassword) and Checkpassword (if the data

Video web	Version: 1.1
Software Architecture Document	Date: 30/07/2022

	matches the existing data in the DB, edit the profile)
GUI: Edit profile	<ul style="list-style-type: none"> Attributes: Password (customize password), User's full name (customize username), User's birth (customize date of birth), User's email (customize email) Methods: Input (receive data that the user entered), Send (send data to CheckInput), if checkInput allows, you can use Update (to update the edited information)
Routes: CheckInput	<ul style="list-style-type: none"> Attributes: Change date (this is the date limit for you to customize the information. For example, if you have just changed your password, the system will only allow you to change your current password after 30 days at yours), Exist emails (the ones that are already stored in the DB) Methods: Take data (get data from the DB (email, password)), Checkpassword (after you update the new password, the system will ask you to re-enter it again to verify the 2 passwords match. vs each other or not), Checkdate (check if your account has had enough days to change information for the current one) and Check email (check if the email you changed is valid and exists) in the DB or not)
MySQL: User account	<ul style="list-style-type: none"> Attributes: Username, Password, User's full name, User's birth, User's email
GUI: View history	<ul style="list-style-type: none"> Attributes: Watches list (list of watched videos) Methods: Delete (delete the history of watched videos)

Video web	Version: 1.1
Software Architecture Document	Date: 30/07/2022

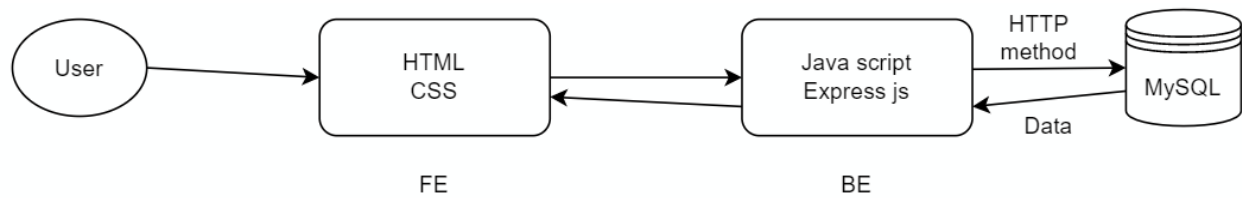
4.5 Component: Uploader functions



Class	Description
MySQL: Video	<ul style="list-style-type: none"> Attributes: Video name, Category, Content, Date release, Source
GUI: Video upload history	<ul style="list-style-type: none"> Attributes: Video list (list of uploaded videos) Methods: Search video (search for posted videos) and Check video information (see video information)
GUI: Post video	<ul style="list-style-type: none"> Attributes: Video link and Uploader information (Video uploader's information) Methods: Upload (send request to server to upload video)
GUI: Edit video	<ul style="list-style-type: none"> Attributes: Video information (Information of the video) and Uploader information (Information of the person who uploaded the video) Methods: Delete (delete posted video) and Edit (edit information of video)

Video web	Version: 1.1
Software Architecture Document	Date: 30/07/2022

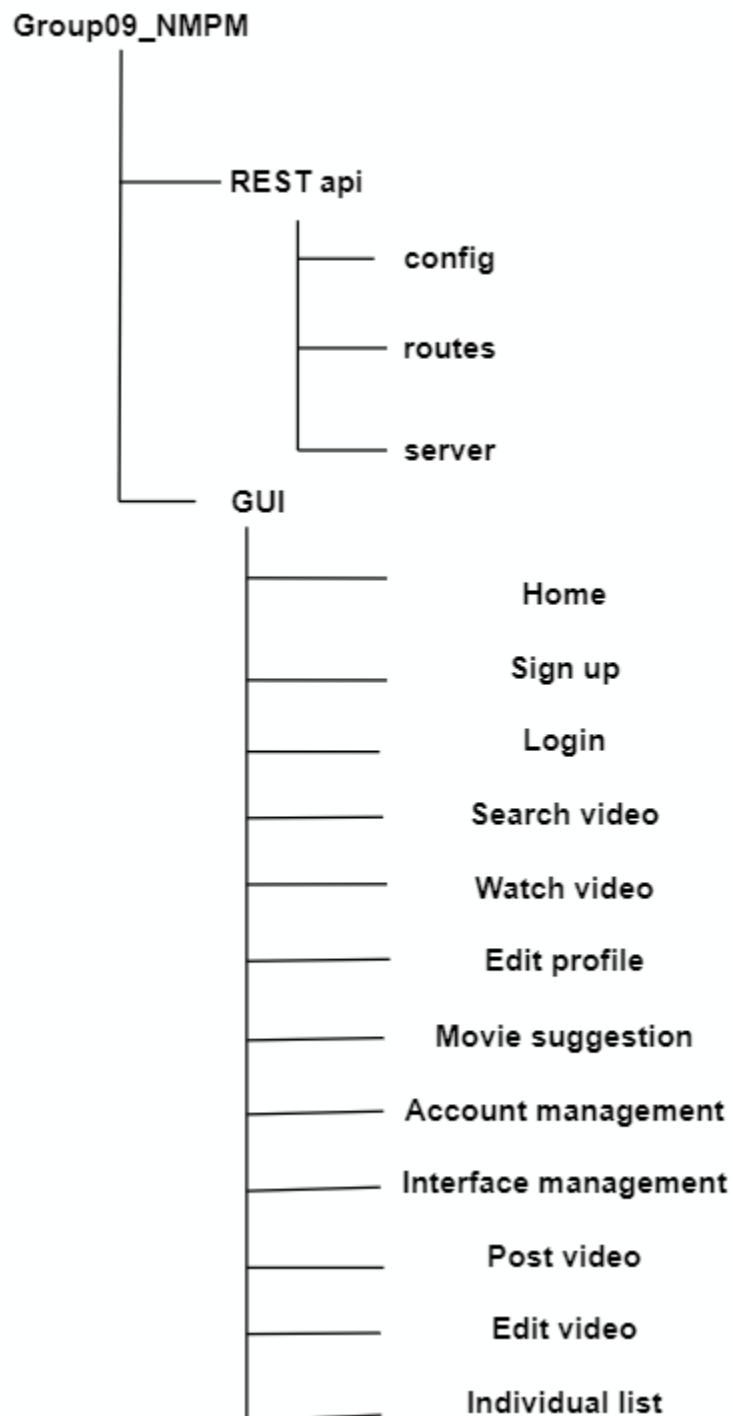
5. Deployment



- **User:** users interact with the system through user interfaces such as uploader, admin and viewer.
- **FE:** shows user interfaces such as Homepage, login, search, those interfaces are developed using HTML and CSS.
- **BE:** get the web working and use HTTP methods to interact with the database, developed using framework Express JS.
- **MySQL:** database that stores information such as user accounts, video links, comments, etc.

Video web	Version: 1.1
Software Architecture Document	Date: 30/07/2022

6. Implementation View



- **GROUP09_NMPM:** contains the directory of the whole system
 - **REST api:** contains source code of BE
 - **Config:** create a connection to the database (MySQL)
 - **Routes:** define methods (CRUD) to interact with the database
 - **Server:** setup server with port 3000

Video web	Version: 1.1
Software Architecture Document	Date: 30/07/2022

- **GUI:** contains the source code of FE
 - **Home:** home page interface
 - **Sign up:** account registration interface
 - **Login:** account login interface
 - **Search video:** video search interface
 - **Watch video:** video viewing interface
 - **Edit profile:** interface to edit personal information
 - **Movie suggestion:** interface for users to suggest videos they want to watch to Admin
 - **Account management:** user account management interface
 - **Interface management:** user interface with interface management function
 - **Post video:** interface to put videos on the web
 - **Edit video:** video editing interface
 - **Individual list:** interface to view the user's favorite video list