VIETNAM GENERAL CONFEDERATION OF LABOR

**TON DUC THANG UNIVERSITY**

**FACULTY OF INFORMATION TECHNOLOGY**

A red and blue logo

Description automatically generated

Nguyen Hoang phuc– 521H0509

KIEU THANH PHAT – 521h0125

Developing a social network app using ASP.NET Core and React JS

Information Technology Project

COMPUTER SCIENCE – SOFTWARE ENGINEERING

Advised by

**MSc. Mai Van Manh**

**HO CHI MINH CITY, 2024**

VIETNAM GENERAL CONFEDERATION OF LABOR

VIETNAM GENERAL CONFEDERATION OF LABOR

**TON DUC THANG UNIVERSITY**

**FACULTY OF INFORMATION TECHNOLOGY**

A red and blue logo

Description automatically generated

Nguyen Hoang phuc– 521H0509

KIEU THANH PHAT – 521h0125

Developing a social network app using ASP.NET Core and React JS

Information Technology Project

COMPUTER SCIENCE – SOFTWARE ENGINEERING

Advised by

**MSc. Mai Van Manh**

**HO CHI MINH CITY, 2024**

VIETNAM GENERAL CONFEDERATION OF LABOR

HO CHI MINH CITY, 2024

ACKNOWLEDGEMENT

We would like to express my sincere thanks to MSc. Duong Huu Phuoc of the subject "Information Technology Project" and the Faculty of Information Technology for making conditions for me to do this report so that I can add the final score column in my process, as well as helping and guiding us throughout the process of studying and making reports so that I and my friends won’t be late the process.

Writing the report has helped me practicing more presentation skills, as well as many other skills. Due to lack of experience in report writing as well as limited knowledge and reasoning ability, in this report there will certainly be errors, looking forward to receiving comments and suggestions from my grateful teacher to help me improve myself.

*Ho Chi Minh city, August 1st 2024*

*Authors*

*(Sign and write full name)*

Nguyen Hoang Phuc, Kieu Thanh Phat

DECLARATION OF AUTHORSHIP

We hereby declare that this is my own research project and is under the scientific guidance of MSc. Duong Huu Phuoc. The research content and results in this topic are honest and have not been published in any form before. The data in the tables for analysis, comments, and evaluation were collected by the author from different sources and clearly stated in the reference section.

In addition, the Project also uses a number of comments, assessments as well as data from other authors and other organizations, all with citations and annotations of the original source.

**If any fraud is detected, I will take full responsibility for the content of my Project.** Ton Duc Thang University is not involved in copyright violations caused by me during the implementation process (if any).

*Ho Chi Minh City, August 1st 2024*

*Authors*

*(Sign and write full name)*

Nguyen Hoang Phuc, Kieu Thanh Phat

social NETWORK app for sharing posts that focuses on connecting with friends.

The social media network similar to it predecesor Facebook and Instagram , is build on the basis of enhancing the connectivity of friends with an endless reach around the world. User can posts feed and post describing their experiences while maintaining a length full and informal conversation through their chat group

As a result of building the application with Asp.Net Core and React JS , the application have fullfill all criterial of a high quality application. using React JS interactive libraries with additional help of Prime React component libraries for the seemless UI/UX design. Additonally , user identity and information security are maintian with Asp.Net Core Identity and Entity Framework

Contents

[Chapter 1 INTRODUCTION 11](#_Toc189382230)

[1.1 Justifying Topic Choice 11](#_Toc189382231)

[1.2 Topic Implementation Goals 11](#_Toc189382232)

[1.3 Subject And Scope of Research. 12](#_Toc189382233)

[1.4 Research Method 12](#_Toc189382234)

[1.5 Practical Significance of The Topic. 12](#_Toc189382235)

[Chapter 2 LITERATURE REVIEW 12](#_Toc189382236)

[2.1 Vite 13](#_Toc189382237)

[2.2 Axios 13](#_Toc189382238)

[2.3 React JS 14](#_Toc189382239)

[2.3.1 TSX 15](#_Toc189382240)

[2.3.2 Components 15](#_Toc189382241)

[2.3.3 Props 16](#_Toc189382242)

[2.3.4 State 16](#_Toc189382243)

[2.3.5 Prime React 17](#_Toc189382244)

[2.4 Asp.Net Core 18](#_Toc189382245)

[2.4.1 Dependency injection 18](#_Toc189382246)

[2.4.2 Direct Dependency 18](#_Toc189382247)

[2.4.3 Inverted Dependency 19](#_Toc189382248)

[2.4.4 Model-View-Controller 20](#_Toc189382249)

[2.4.5 Asp.Net Core Identity 20](#_Toc189382250)

[2.4.6 Asp.Net Core Jwt 21](#_Toc189382251)

[2.5 Rest Api 22](#_Toc189382252)

[2.6 SQL server management studio 23](#_Toc189382253)

[2.7 WebSocket 24](#_Toc189382254)

[2.8 SignalR 25](#_Toc189382255)

[Chapter 3 REFERENCES 27](#_Toc189382256)

LIST OF FIGURES

[Figure 1 Vite Logo 12](#_Toc189382185)

[Figure 2 Axios Logo 13](#_Toc189382186)

[Figure 3 React Logo 14](#_Toc189382187)

[Figure 4 JSX flowchart 14](#_Toc189382188)

[Figure 5 Props Flowchart 15](#_Toc189382189)

[Figure 6 State Flowchart 16](#_Toc189382190)

[Figure 7 Prime React Logo 16](#_Toc189382191)

[Figure 8 Performance Indicator 17](#_Toc189382192)

[Figure 9 Dependency Graph 18](#_Toc189382193)

[Figure 10 Inverted Dependency Graph 18](#_Toc189382194)

[Figure 11 MVC Graph 19](#_Toc189382195)

[Figure 12 JWT Flowchart 20](#_Toc189382196)

[Figure 13 Token Image 21](#_Toc189382197)

[Figure 14 Body Image 21](#_Toc189382198)

[Figure 15 Rest Flowchart 22](#_Toc189382199)

[Figure 16 SQL Server Management Logo 22](#_Toc189382200)

[Figure 17 WebSocket Flowchart 23](#_Toc189382201)

[Figure 18 SignalR Flowchart 25](#_Toc189382202)

LIST OF TABLES

ABBREVIATIONS

|  |  |  |
| --- | --- | --- |
| BFF | Backend For Frontend |  |
| CNN | Convolutional Neural Network |  |
| CSS | Cascading Style Sheets |  |
| GPS | Global Positioning System |  |
| HTTP | HyperText Transfer Protocol |  |
| HTML | HyperText Markup Language |  |
| OTP | One Time Password |  |
| ORM | Object Relational Mapping |  |
| JWT | Jsonwebtoken |  |
| PV | Persistent Volume |  |
| PVC | Persistent Volume Claim |  |
| RMIT | Royal Melbourne Institute  Technology | of |
| SDLC | Software Development Life Cycle |  |
| TCP | Transmission Control Protocol |  |
|  |  |  |

# INTRODUCTION

## Justifying Topic Choice

In today’s interconnected world there is a great need for digital communication, communication that will enable friends and family to connect with each other through a wide range of distances. Moreover, with information being so redundant people are gaining much more information and knowledge through a wide range of mediums than there was before, but that doesn’t mean that the information being received is informative. An example of this is disinformation which can potentially cause harm depending on how the individual perceives it.

The social media network of today is creating solutions for harnessing our world interconnected infrastructure. These social networks are creating a vast database of users and users’ content with both flexibility and growth in mind, in addition to preventing bad actors from creating a hostile environment for other users. The forefront of These social media network is Facebook and Instagram, these two websites have created an environment that encourages users to stay engaged with the latest trend and connect with their friend personal experiences, with an additional benefit of discovering their own hobby and communities. Even with the vast function of these social media networks, they still lacking the localization and feature tailor to the preferences and cultural aspect of a specific country like Vietnam.

Our social network application is set on fixing these issues, by customizing the UI/UX design to meet the preferences of Vietnamese user and incorporating features that are tailors toward them. This will result in a more friendly application for Vietnamese users enhancing their social tie, sense of nationality and comfortability when using our social media networks.

## Topic Implementation Goals

The main goal for this application is to provide a social network platform that is tailor for the specific need of Vietnamese users, to post and share their experiences and maintain connection through features like chat messaging. The system is further enhanced by customization preferences by the user. For example, users can choose who they want to share post with, what group of friends would the user allow to see the posts, and which post will be saved for later viewing. The application is also concerned with improving data security and integrity. This is implemented by incorporating ASP.NET identity frameworks for the authorization and authentication method, which decide what user can and cannot do. These protection barrels help keep user accounts protected from bad actors. In addition to these main features, there will also be additional advanced communication features like creating and managing group chat, assigning role and user privilege in each group chat. Creating a moderation system that allows moderators to monitor user activity and detect bad actors in the system.

## Subject And Scope of Research.

The main users of the application are people ranging from the age of 18-65 who would like to post their experiences, admins who are assigned with managing group, and company who want to expand their band to a wider audience. The software that the team will use for collaboration are GitHub, GitLab and Git kraken. Each member used a wide range of IDE including but not limited to Microsoft VS, VS Code. The front of the application is programmed using JS with the main library being React JS, using primary Prime React for the UI. On the other hand, the back of the application will use C# with the framework being ASP.NET CORE incorporating Identity and managing the database server which used SQL database management. The application is tested with 10-20 users who have used social media platforms before for better feedback.

## Research Method

System implementation: the system will use an authentication method using username and password, user can register for their new account or login with the right password and credentials. In the situation that the user forgets their password the user can used the recover password function that will send an OTP request, prompting the user to check their email in order to follow the necessary step require to create a new password for their account. When Login, the user can perform common functionality seen in other social media websites like CRUD operation on their posts and additional interaction operation to another user. All users and their users’ posts can be reported on the basis of preventing bad actors and maintaining a welcoming community for the application. On the admins side, admin can monitor user activity and posts to filter for bad actors in the system, have a detailed description of user post activity and decide which post to banned or delete if it violates community guidelines.

## Practical Significance of The Topic.

The social media networks are tailor for cultural preferences of the Vietnamese people through UI/UX and additional feature, targeting audience between the age of 18-65. The user of the social network can post about their journey and experience places that they have visited through medium like picture and text. With the added benefit of messaging and maintaining friendships with the network chatting feature. Admins can moderate and check for disturbances with the site, seeing if there are any bad actors in the system.

# LITERATURE REVIEW

This chapter is used to explain, understand and contextualize the technology used in application development. This explanation will include details such as why the technology was used, what it is used for and how we implement it in the application operation.

## Vite

Is a complied build tools that provide faster development experience for developer . Vite used native ES modules and modern browsers APIs to quickly comply your code hence the name Vice which in French mean “quick”, with no need to incorporate bundle. The building development server in Vite is equipped with an optimized system for faster reloading and hot module replacement, allowing for changes that developer made to happen in real time. In our project we used Vice to optimize our development cycle, allowing us to make rapid changes in the front end without having to worry about dependency between models or waste time rebuilding the application by using hot reload.

*A logo with lightning bolt in the middle

Description automatically generated*

Figure Vite Logo

## Axios

Axios is a promise-based HTTPS client that is used for node bases applications and applications that run on the browser. Some key feature of Axios is it isomorphic nature this mean that it have the capability to run on both the server side and client side of an application. The main difference between running on the server side is that it uses native node.js http module as opposed to XMLHttpRequests when running on the client side. There are many features in the Axios library, but our application used Axios for its ability to handle https requests, interpreted and handle these requests when sending them from the client to the server. As a result, making it easier for our application to handle the request and how to response to them.



Figure Axios Logo

## React JS

React is a JavaScript library created by Facebook to simplify the process of creating and managing the frontend interface. React optimize your rendering and frontend application by using the virtual Dom when an element render instead of changing the entered DOM, react will create a virtual DOM to apply the initial change then using a Diff algorithm apply the needed change on the real DOM, this process used to optimize rendering speed of your applications. React is also built on the principle of teach one write anywhere with it agile development and compatibility with other frameworks and library it can be added into your application anytime anywhere. We used React to simplify or frontend development thought the use of many features like components, JSX, state and props making our UI adaptable the changing UX of the application.

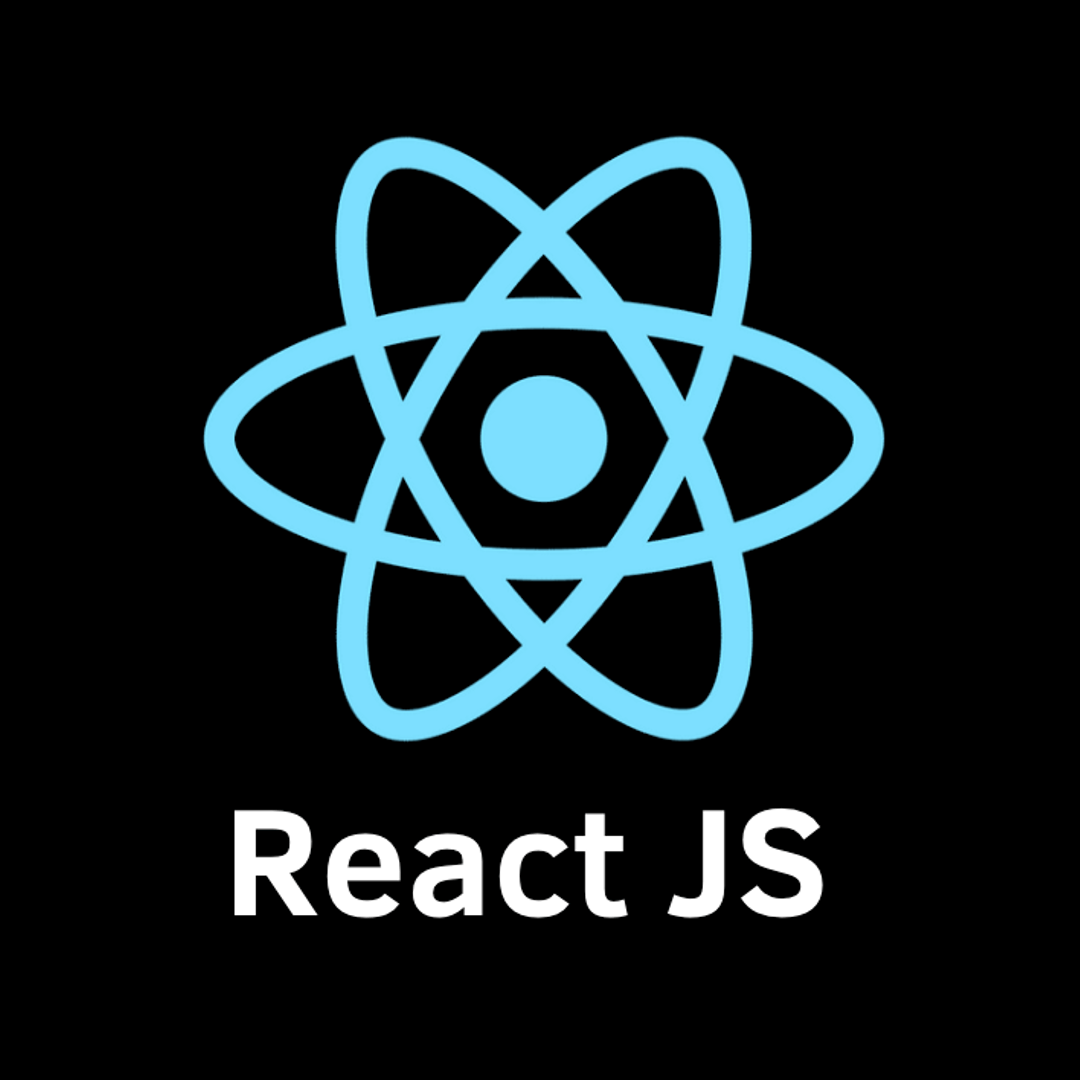


Figure React Logo

### TSX

The language that we used in your application is TSX, TSX is similar to JSX in react with the main difference being that it is written in typescript instead of JavaScript. In the same way as JSX, TSX is an extension of JavaScript that also comes with the full power of ES6, this power translates to its main feature which is the using a {} expression to combine JavaScript dropdown tag in a single file for separation of concern. This separation of concern has created a unit of call component.

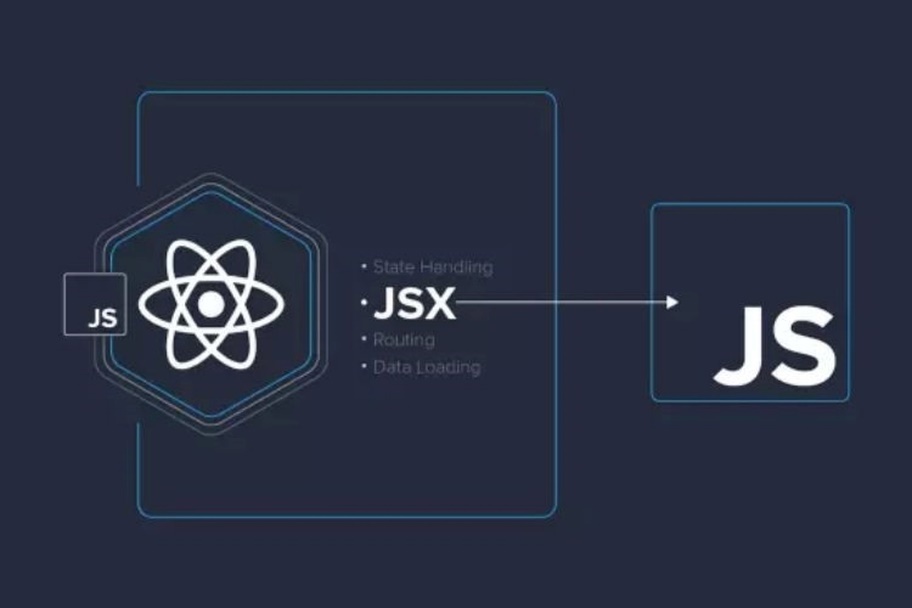


Figure JSX flowchart

### Components

Components are encapsulated pieces of react code that represent a UI element by managing the output render in the DOM. Each component can be reused multiple time in multiple different file bases on the developer need. In addition to being reusable, components can pass information to each other to meet the adaptation of the UI. This information are represented in two main ways state and props.

### Props

Props are object that are passed thought by its parent components specify using it child components argument, every parent component can pass state and information to its child component using props.

A diagram of a diagram

Description automatically generated

Figure Props Flowchart

### State

React have a build in state object assign to each component storing the data an information of a component, when the state changes the component will be rerender to adjust with the new UI changes. Moreover, state can also help pass information between layer in the react tree hierarchy, this mean that it is able to pass down the information from a component to it children using props, or lift the state up to it ancestor so that it dependencies can affect it in the desirable way.

A diagram of a state

Description automatically generated

Figure State Flowchart

### Prime React

Prime React is an open-source UI component for React containing over 80 components. Developers using prime React can choose a wide range of prebuild components and style to use in their project optimizing the time necessary to build a full-scale UI/UX interface. Moreover, Prime React is also customizable, you can incorporate a CSS library of your choice to enhance or change the theme of a Prime component. With these features in mind, our team has used Prime for over 50% of our components, this includes thinking like button, sidebar, avatars making our UI/UX development quicker and easier.

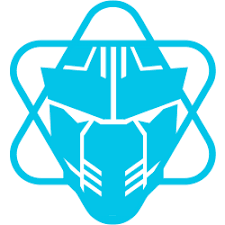


Figure Prime React Logo

## Asp.Net Core

Asp.Net Core is a cross-platform open-sources and top performance development framework for the .Net ecosystem. Enabling developers to build applications from IOT to websites. Asp.Net Core was first released in 2016 as the child of closed sources development Asp.Net. is built to run on many popular OS like window and Linux, with the added benefit of having the highest performance speed when compared to its rival Node.js and Java.

A screenshot of a device

Description automatically generated

Figure Performance Indicator

ASP.NET Core offers a wide range of features to streamline development. For our project, we utilized some of these features to enhance functionality and efficiency:

* Dependency injection
* Model-View-Controller
* Entity Framework
* Identity

### Dependency injection

Dependency injection is a design pattern where a dependency does not need to create a dependent object to use it functionality instead it will call the object when needed. This design pattern is used to achieve Inversion of control between the classes and their dependencies making it satisfy the loosely couple principle when designing application.

### Direct Dependency

Dependency inversion is a technique to combat direct dependency, take for example class A, B and C in the picture each class call the other class to use the objects’ function or variable in the dependent class. A is dependent on B so it has to create a new Object B for it to use B function, B is dependent on C, so it has to create a new Object C to used it function and so on and so on. This is reflected in the run time chart where each class will be created linearly when call.

A diagram of a diagram

Description automatically generated

Figure Dependency Graph

### Inverted Dependency

Inverted dependency fixes this by instead of calling the class directly call the interface that implement this class instead, class A call the interface of class B, class B call the interface of class C and so on. Making the process of changing class more adaptable and easily customized than if we were to call the class directly.

A screenshot of a graph

Description automatically generated

Figure Inverted Dependency Graph

### Model-View-Controller

Model-view-Controller is a software design patter to decouple user-interface into three main parts:

Model: for handling data and business logic

View: defines the UI or how the data should be present

Controller: used to control and deal with user command like forwarding command to the model and view handler

The pattern used “separation of concern” to provides for a better division of work and create clean code for maintenance. The flow of work goes like this the requests are routed to the Controller which will then work with the Model to perform an action or to retrieve data to be display to the View. In our application, we used the Model and Controller to define objects, mapped them to the database using Entity Framework for automatic creation, and managed them through routes and controllers.

A diagram of a model

Description automatically generated

Figure MVC Graph

### Asp.Net Core Identity

Asp.Net Core Identity is a tag along membership system within Asp.Net Core that adds login features to your application. It manages services like user sign up, login, password renewal, Roles and claim-based authorization. The Roles and claim-based authorization is one of the main functionalities of the system dealing with authentication (checking the user’s identity) and authorization (controlling the user’s privilege). Identity configuration is usually with a SQL database Server to store user info like name password and personal data. Moreover, identity is not contrast to SQL server database but can also be move to another persistent storage like Azure Table Storage

### Asp.Net Core Jwt

A diagram of a process

Description automatically generated

Figure JWT Flowchart

Asp.Net Core provides a way to create a JWT token, a JWT token is a simple and secure practice for JSON object to be transmitted between different parties. Think of a token as a access key for the JSON object, the key is used first when encrypting the token, second when decrypting the token on the backend server. The JWT token is encrypted into 3 different sections split by a dot(.)

A close-up of a number

Description automatically generated

Figure Token Image

Each part is encrypted, starting with the first red segment, which serves as the header. This section includes details such as the signature or algorithm used for the encryption process. The second part contains the payload, which holds the data within the JSON object. Finally, the third part is the secret key, used for both encryption and decryption.

When a user sends their JWT token containing their credentials to the backend server it will then be decrypted by the server secret revealing the payload content:

A screenshot of a computer

Description automatically generated

Figure Body Image

Enabling the server to Authorize and Authenticate the user of the application using Asp.Net Identity Framework

## Rest Api

Rest Api is the principle of web application development, following the principles REST architectural approach. Created by Roy Fielding in the year 2000, Rest Api quickly became popular because of its main benefit being its simplicity, stateless, scalability and lightweight. Client can interact with the backend sever thought http request to get return data in convenient format like JSON or XML, these methods include:   
GET: get the requested data.

POST: create a new data object

PUT: update an existing data object

DELETE: Remove a data object

A cloud with text and arrows

Description automatically generated

Figure Rest Flowchart

## SQL server management studio



Figure SQL Server Management Logo

SQL server management studio was created by Microsoft in the year 2005 with the purpose of creating an integrated environment for handling any SQL infrastructure from SQL Server to cloud and local Azure SQL database. It allows developers to configure and manage the database engine, deploy, oversee, update components data-tier used by the application. SQL Server can also be used to build queries and scripts for designing and managing your data warehouse both local and on the cloud

## WebSocket

WebSocket is a bidirectional, full duplex protocol. Bidirectional mean to allow communication from Two network devices, full duplex protocols mean that each devices will have 2 communication channels dedicated to receiving and sending information. Unlike it counterpart Http, WebSocket is a stateful protocol, which means that the connection will be maintain until terminated by either party and all session information will be store in the connection. Based on all these attributes, we can conclude that WebSocket is best used in real time communication like call or chat group, which requires a constant stream of data to be receive by both the client and server .As a result, we used WebSocket for the creation of our chatroom use cases ,which will allow the user to interact, chat and message with friend and love ones

A diagram of a web socket

Description automatically generated

Figure WebSocket Flowchart

WebSocket flow:

1. The client will initiate a request to the server.
2. The server will create a HandShake confirming and verifying the creation of a connection.
3. This connection will then be known as a WebSocket, which allow bidirectional and full duplex protocol
4. The WebSocket remains as a dedicated communication channel for this session, handling message request until either the server or client explicitly closes it or the connection from one of them end.
5. Status code 101 is used to notify a protocol switch to WebSocket.

## SignalR

SignalR is an ASP.NET library that provides real-time web functionality for applications. This functionality allows the server to push contents to the client directly instead of waiting for a response. As a result, this functionality helps with applications that require a constant stream of data like when user messages each other, refresh page or if a page implements long polling. This approach will decrease the amount of network traffic the server receives, decrease pooling time for receiving data and maintain a dedicated pipeline for request.

SignalR work by being an abstraction over some of the transport that is used to do real time communication between server and client. Real time communication is possible using a RCP API to invoke function in the client from the server. RCP support "server push" functionality, which means that changes that happened in the database to be push back to the client. Moreover ,SignalR supports the following transport technique for handling real time communication :

* WebSocket
* Server Sent Event
* Long Polling

But prefer long WebSocket due to it optimal transport due to it efficient used for server memory , lowest response rate and containing the most underlying features. Additionally, SignalR abstraction helps with maintaining version control and shields developers from compatibility issues with WebSocket, you do not need to worry about whether WebSocket update or the application using a different order version.



Figure SignalR Flowchart

Signals used Hub to communicate between client and server. Hub is a high level pipeline that is established for client and server communication, Hub contain 2 build in protocol text bases JSON for messages and binary for binary file data. Each time a request is send signal will dispatch the request from hub between client and server. In our application, we use **Signal** to manage backend responses. Each time a chat group message is sent, a WebSocket is established for that specific chat group, enabling real-time communication.

# REFERENCES

T. (2020, November 18). Xây dựng hệ thống điểm danh. SlideShare. Retrieved March 14, 2024, from [https://www.slideshare.net/man2017/xydng-h-thng-im-danhhc-sinhpdf](https://www.slideshare.net/man2017/xydng-h-thng-im-danh-hc-sinhpdf)

A. (2023, June 18). FaceDetection\_AttendanceSystem. Retrieved January 16, 2024, from https://github.com/aryaraj132/FaceDetection-AttendanceSystem G. (2022, January 18). Flutter Docs. Flutter. Retrieved February 11, 2024, from <https://flutter.dev/>

Justin, E. (2022, January 18). What is an ORM? Prisma.IO. Retrieved March 21, 2024, from <https://www.prisma.io/dataguide/types/relational/what-is-anorm>

Mandic, V. (2024, March 20). FaceAPI. Npmjs. Retrieved March 22, 2024, from https://www.npmjs.com/package/@vladmandic/face-api

O. (2017, October 30). ExpressJS Docs. ExpressJS. Retrieved February 25, 2024, from <https://expressjs.com/>

O. (2019, July 22). MySQL docs. MySQL. Retrieved January 17, 2024, from <https://dev.mysql.com/doc/refman/8.3/en/what-is-mysql.html>

T. (2024, March 21). Tensorflow Docs. Tensorflow. Retrieved March 5, 2024, from <https://www.tensorflow.org/learn?hl=vi>

Wikipedia (2011, January 25). WebSocket. Retrieved March 1, 2024, from <https://en.wikipedia.org/wiki/WebSocket>

Kubernetes. (2022, October 12). Kubernetes. Retrieved July 11, 2024, from <https://kubernetes.io/vi/docs/concepts/overview/what-is-kubernetes/>

Docker. (2017, December 11). Viblo. Retrieved July 2, 2024, from <https://viblo.asia/p/docker-la-gi-kien-thuc-co-ban-ve-docker-maGK7qeelj2>

Docker*: Accelerated Container Application Development*. (2024, July 8). Docker. Retrieved July 2, 2024, from <https://www.docker.com/>

*Volumes*. (2024, July 12). Kubernetes. Retrieved July 30, 2024, from <https://kubernetes.io/docs/concepts/storage/volumes/>

Thư, T. (2021, May 21). *[Kubernetes storage] Học cách sử dụng Persistent Volume (PV) và Persistent Volume Claim (PVC)*. Viblo. Retrieved July 25, 2024, from https://viblo.asia/p/kubernetes-storage-hoc-cach-su-dung-persistent-volumepv-va-persistent-volume-claim-pvc-Qpmleyynlrd