HYPE

Project DocumentationName:



Seat no:



This is to certify,

**CERTIFICATE**

***Kartik Jadhav***

***Nuvesh Mankar***

***Yadnesh Ingale***

Students of Third Year of B.sc (Computer Science), have submitted aProjectReporton

***“HYPE”***

Submitted to

Bhonsala Military College, Nashik Savitribai Phule Pune University

As a partial fulfilment of Bachelor of Science (Computer Science) for the Academic Year2022-2023.

**Project Guide HOD**

**Internal Examiner External Examiner**

# Index

## Introduction

* 1. motivation
  2. problem statement
  3. purpose/objective and goals
  4. literature survey
  5. Project scope and limitations

## System Analysis

* 1. Existing systems
  2. Scope and limitations of existing systems
  3. Project perspective, features
  4. Requirement analysis- Functional requirements,

## System Design

* 1. Design Constraints
  2. System Model
  3. Data Dictionary
  4. User Interface

## Implementation Details

* 1. Software/hardware specifications

# Testing

1. **Conclusion**

# Future Scope

1. **References9.**

# INTRODUCTION

Hype is a social media platform that allows users to share photos and videos with their followers. We have Creating a web of Hype while developing a Hype that had help us improve our skills in our programming language as well as web development and gain experience building a complex web based application.

To create anHype web project in Python, you will need to have knowledge of web development frameworks like Django and Sql, as well as front-end technologies such as HTML, CSS, and JavaScript. You will also need to have a basic understanding of databases and how to interact with them using Python.

Creating anHype web project in Python can be a challenging task.

# Motivation

Hike was the first Indian social media platform. But it does not hold the competition with other social applications such as whatsapp, instagram and telegram. So we tried to build an Indian social media platform that can stand up to these platform and give them competition.

# ProblemStatement.

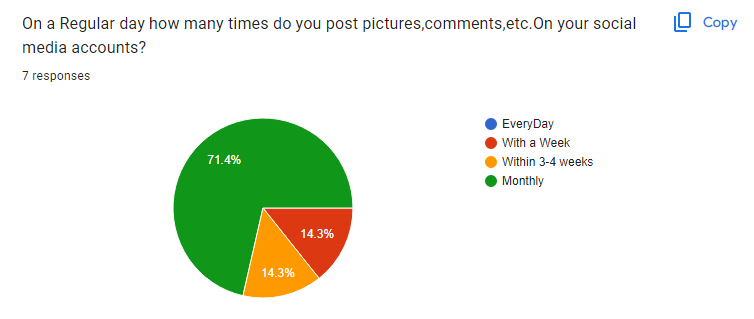
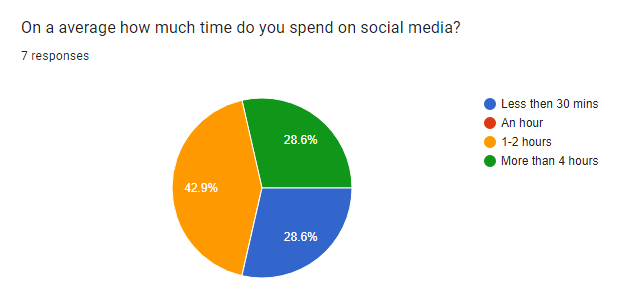
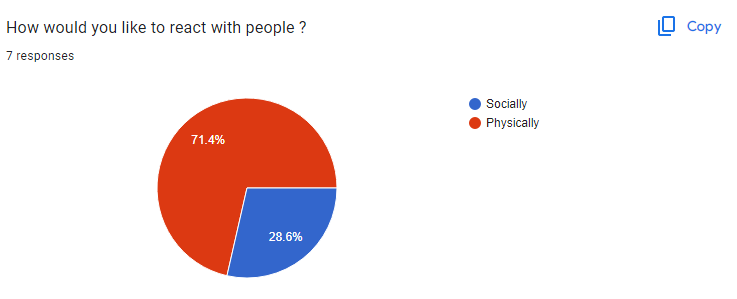
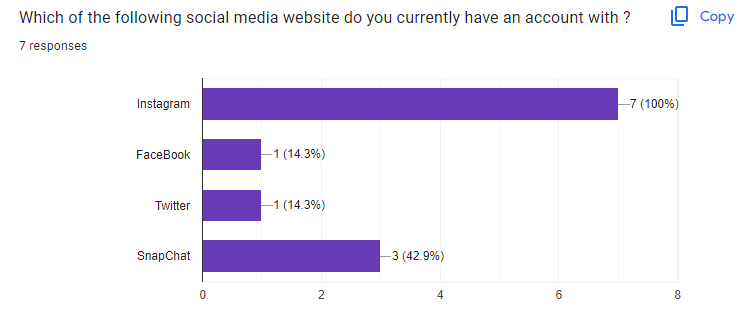
1. We Cannot follow other people
2. We are unable to comment on post.
3. We cannot able to edit our profile.
4. Chats are not private because they are saved on cloud storage
5. Chats cannot be recover

# Purpose/ObjectiveandGoals

1. Connect people socially
2. Maintain chat securities
3. Users are able to post photos to improve their experience
4. Can follow people to see their post

# Literature Survey

According to Our Survey:



# Project Scope and Limitations

* **Projectscope.**
  + - This system allows users to send messages to the people they are following.
    - Users can upload photos so that the user can be more connected with followers
    - Users can follow people they know or someone they would like to know.
    - Users can have the feature to search for someone.
    - Users can edit their profile.
* **Limitations.**
  + Features like messaging have not been advanced to the extent where a user can send images
  + User cannot do a voice or video-call with an other person.
  + Posting videos is unavailable.

# SYSTEMANALYSIS

* 1. **Existing System Study**
     + After reviewing and researching existing systems, the developer observed that there are many existing systems in the market which are in some manner or other social networking websites. All of them had the basic functionality that every other SNS which are messaging, posting (text/images/videos)and receiving notifications. **Messaging**: The messaging function allows the user to send messages to other usersthatareontheplatformorinsomesystemstheycanonlysendmessagestopeoplewhoaretheirfriends or someone they are following
     + **Notifications**: The user receives notifications when someone follows them or when someone likes or comments on their posts. They also receive notifications if they get a message from someone.
     + **Posting**: The user can post images with some caption or they can just post images without caption. They also have the option to share/post videos.
     + **Public/Private Account**: The user has the option to make their account public or private. If the account is public anyone can see the images unless they are blocked by the user. If the account is private, only the people who follow are allowed to see the posts.

# Scope and Limitations of Existing systems

* **Scope of Existing System**
* Hike had affiliations with big brands such as MCD, dominos, CCD to provide vouchers/discount coupons to the customers. Users appreciated the app for rolling out such benefits which eventually led them to get new consumers
* Hike targeted customers via the incorporation of local languages and regional stickers in the app to draw a personal connection. This helped the company gain the attention of local people who could now communicate in their own language comfortably..
* Hike was the first messaging application to introduce Text to Stickers. This was an amazing feature that allowed any text typed by the user to be converted into a cool sticker to make chats more vibrant

.

## LimitationsofExistingsystems

* Hike had too many features such as Hike daily, Hike direct, hidden mode, Hike offline, two way chat theme, news updates, games, Hike cricket and horoscope. While this wouldn’t have been a challenge for its young users, the older generation could have been perplexed by it, and hence not considered it an alternative. Moreover, new features were launched every six months, which made the app very complex to use.
* The global dominance of tech giants was one of the major factors of Hike falling out. Hike couldn’t beat its competitors WhatsApp, Telegram, WeChat and a few more to become the top of its gameAniPhoneorAndroiddeviceisnecessaryifyouwanttousethefullfunctionality, including posting. There is some limited desktop functionality. This makes it less than ideal for scheduling and posting our staff don't own smart-phones or don't want to use them for work.
* No USP(Unique selling point)Initially, their USP was providing sticker stores. But consequently, WhatsApp too added the sticker features, making Hike just like any other messenger.

# ProjectPerspective, Features

* **Project Perspective**
  + **From A friendship perspective, our** site is the place where young people can maintain and nurse their existing (offline) friendships and create new (online)friendships.
  + **The more interactive place our** site has features of posting photos and stories which helps them to interact with more people
  + **The network perspective of our** site is the place where young people learn the crucial importance of being able to network which they can be fit from their future professional life.
  + **The social perspective of our** site makes young people more social and helps them communicate with others. The site help youngsters cope with shyness or loneliness.
  + **The communication tool perspective** our site is merely a communication tool for young people and they use this it’s similar to how they use their mobile phones.
* **Features**

1. **Add and Manage multiple accounts from this a me device**:

Users can make multiple accounts using same device and manage them separately

## Photo and story Post:

Users can PostPhotos and stories

## Like And Comment:

Users can like posts and they can comment on it.

## Search People:

Search People through Username.

## Edit Profile:

People can edit their profile. They can change Profile picture, edit their username and change their password.

## Messages:

People can message their followers from a messaging page.

## Notification:

People receive notification from the system. They will know when someone follows them. When someone likes their comment, they will get a notification.

# Requirement Analysis -Functional requirements

## Account Object\*

Each user who wants to use the site must create an account. This is the head class that all other objects use to determine what a user does and when the user did it. The account's information has four purposes: holdtheorigininformation,holdfriendinformation, hold profile information, and hold privacy information with such a large amount of information to keep track of the Account class would be very large and difficultto work with.There for to ease the load,the Account class was brokenup into three different classes. There is the actual account class which keeps track of login information and friend information.It also hold he other two classes with in it. Profile information was out sourced to the Account Details class, and privacy settings were out sourced to the Privacy Settings class.The only time the Account class needs to be updated is when the user changes his/her username and/or password. All other settings are handled by the Account Details and Privacy Settings classes

## Account Details Object

* + An Account Details object is a helper class created whenever a new Account object is created. The object contains all the information that shows up in the user's profile. The user can edit this by modifying his/her profile. Overall, this class has no other purpose button be a helper class to its account object.

## Privacy Settings Object

* + A Privacy Settings object is the other helper class created whenever a new Account object is created. This object contains all the privacy settings that a user has, such aswhocanviewhis/hermediaor custom pages. Thisclassiscalledanytimeauservisitsa profile or content created by another user. However, it does not directly interact withtheothermodelclasses,onlytheview.

## Chat Session Object

* + Theseobjectscontainuniqueinformationforaparticulartypeofactionauserperforms.
  + Allofthesecontain refer except other account the towns them.Eachobjectis a workingcopy of an object in the Server. Anytime one of these objects is created on the Server, an entry of its creation isaddedtothe news feeddatabase.

## Message Object

* + AMessageobjectiscreatedwhenausercomposesanewmessagetobesenttoafriend.After it is confirmed that the friend is located in the database, the Message object addsits information to the database. When a user checks his or her inbox, a list of messagesthatweresenttotheuserwillbeshowindescendingorderofwhentheywerereceive

## Friends

* + The most important feature is being able to add and remove friends. In our Social Networking site, making friends is a fairly straightforward process. Users can type in the name of a friend in the search bar at the top of their homepage. The database is queried for an account that has the search term contained in the full name, any media files with the search terms in the description, any pages with the search terms in the titles, any links with the search term in the titles, and any notes with the search terms in the title. For example, User A could search for User B in the search bar. After clicking on User B's profile, User A will see a button that says Send Friend Request. Clicking on it will send an alert to User B that User A wants to be a friend. The friend request will now be in the friends list of User B, where he/she can either accept it or ignore it, letting it sit there indefinitely, If user B accepts the request, User A will beaded to User B's friends list and vice versa. Being friends has its advantages. For example, only friends can chat to each other. Also, friends can view any part of a profile that is marked as friends only. Finally, for a user to view his/her friends and incoming friend requests, he/she just clicks on the friends tab which brings up a frame.

## Account Creation

* + When a user accesses the site for the first time, he/she must create an account beforeusinganyofthesitefeaturesTheaccountcreationprocessisbrokenintothreesectionsThefirstsectiondealswiththelogininformationandis required or the user to fill out.
  + This includes the email, password, and password confirmation. The purpose behind the password confirmation is to ensure that the user didn't accidentally mistype when creating a password. These creations section deals with information about who you are such as name, location, and gender. Most of these fields are optional except for your name and gender.Itwouldn'tbemuchofasocialnetworkifeveryonewasnamedanonymous.

The final section deals with information about the users likes and dislikes, such as interest sand activities. Unlike the other two sections, this section is completely optional. Once the user clicks create account, a new account, account details, and privacy settings are added to the server, and the user is brought back to the login page

## Chatting

* + One of the advantages of having friends is the ability to have live communications with the MIA chat. When a friend is online, he/she will appear in the chat available friend to chat with. To start a chat session, simply click on then a of the friend which will begin a new chat session. The little Bart the bottom will change from chat disabled to chatting with a friend name as seen in the figure below. Also, the friend receiving the chat will get a notification that a new chat session has started.
  + Then the two friend scan chat with each other until one or both log out. The chat feature is a little more complex than others as it requires a combination of JavaScript and jQuery to work. When a user clicks on a friend's name to begin a chat, a chat request is sent to the database. On every page there is a jQuery function that queries the server for any new chat requests once a second. It needs to be jQuery, so the client can request for the server to run a check and return any relevant information. If it was pure PHP, the page would never load completely, because the page cannot finish loading until the server is done with all preprocessing. The JavaScript portion Of jQuery allows the client to ask the server to do more processing after the page has been loaded If there is a new chat request, a ponders generated, notifying the receiver of the new chat that is starting or a new pending chat if the user is in another.

## Messaging

* + Unlike with chatting, users can send a message to any other user. For user A to send a message to user B, he/she simply goes to the message center tab and clicks compose message. He then fills out the form as shown in the figure below with the email of the recipient, the message title, and the message content. The new message will then be inUserB'sinboxinthemessagecenter.Messagingisimplementedbystoringarecordofthemessageintheserver.Firstamessageobjectisconstructedthat takes in to from accounts, the message body, and the date it was sent. The message body is retrieved from a user submitted form, the from account is retrieved from the user session, and the receiver account is retrieved from the recipient email address The receiver account is validated, to ensure the sender didn't try to send a message to anon- existent receiver. Finally, the new message is stored in the server. Retrieving messages sent to the user is a simple method of querying the server for any messages that have been sent to the user, and this is done by checking the receiver ID of each message against the account ID of the user. Any matches are returned to the inbox of the user as shown in the figure below.

## Events

The receiver account is validated, to ensure the sender didn't try to send a message to anon- existent receiver. Finally, the new message is stored in the server. Retrieving messages sent to the user is a simple method of querying the server for any messages that have been sent to the user, and this is done by checking the receiver ID of each message against the account ID of the user. Any matches are returned to the inbox of the user as shown in the figure belowAfter filling out the form, the user will be brought to the standard confirmation page He/she can then view the new event by clicking on the Events tab and clicking on then event. If the user wishes to edit some information about the event, he/she can click on the edit link next to the event where a similar form to creating an event will be displayed with all the fields populated with the current event information.

## Media Uploading

* + Aspeopleusetheirsocial networking account, they willwanttobe able ouploadfunnyorinterestingimages,video,music,etc.

To share withtheirfriends.Themediaupload section will be located at the media tab, where users are able to specify a file to Upload as well as provide short description of the file to be uploadedIn order to prevent users from uploading potentially malicious files such a executables ,only certain file extensions are supported. These allowed extensions cover popularimage extensions such as png ,jpg ,gif ,and bit map, video extensions audio extensionsmp3,wma, and wav, and some document files including txt, rtf, doc, and pdf. The inner working so this uploading process are surprisingly simple ,since the HTTP server take scare of requesting the file from the client automatically. After the user clicks upload, the file is sent to the server and stored in a temporary location.

# System Design\*\*

## Design constraints

* + - Engage Quickly

Engagement is crucial for the success of any website. You need to make sure that visitors are immediately drawn into your site, either through great content, a compelling call to action, or some

GIVE VISIT OR SOMETHING TODO

Your home page should present visitors, both new and returning, with something to*-do*. Logging in or signing up is the most obvious thing for visitors to do, but think about other options. Give them the opportunity to explore what the site is all about before they sign up. Let them search for people they already know on the site. Give them a chance to see why they should sign up before forcing them to. It builds a sense of trust between your site and its users right from the start.

PROMOTEINTER ESTING CONTENT FROM FOLLOWERS

Show your users what their followers are doing. From the moment someone logs in, they should be able to see what their followers have been doing, posting, and otherwise promoting. Most site approach this with a news feed or similar listing of all the activities your follower startup-to.

MAKE IT EASY TO FIND FOLLOWERS

There’s nothing sadder than a social network account with few or no followers. Make it easy for your users to find followers, both new and old. Letting users search by name, and Username makes it more likely they’ll engage with a lot of their users, improvingEveryone’s user experience. The more followers a user has, the more active their profile and news feeds will be, meaning they’re more likely to come back often.

* + - Let Users Express Themselves

Self-expression is one of the hall mark so social media. Some’s its approach this by giving users almost full control over the way their profile page looks. Others restrict the design options but let users add content to suit their own preferences. The degree to which your social network allows users to customize and personalize their profiles is up to you; just make sure there’s some functionality in that area.

* + - Be Dynamic

Dynamic content is the lifeblood of Web 2.0 sites. Social networks are no different. Content should change constantly, with the newest, most popular, and most valuable information continually pushed to the forefront for users.

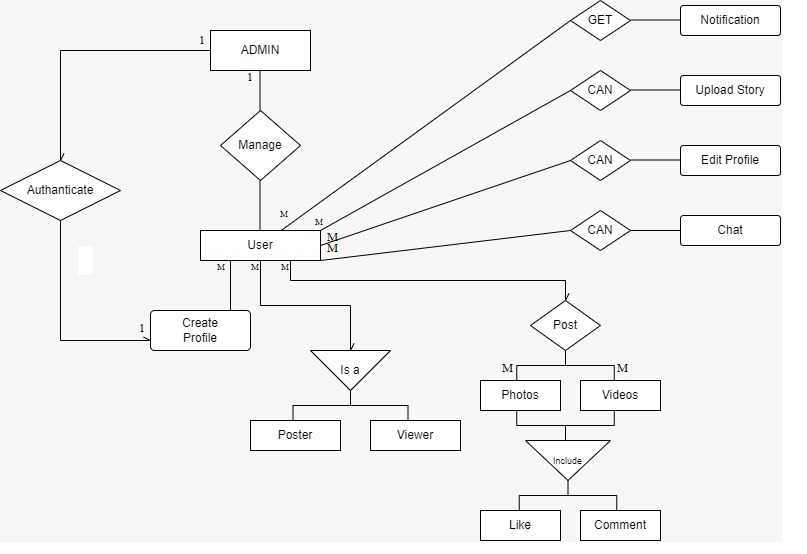
* + - HAVEREGULARLY-CHANGINGCONTENT

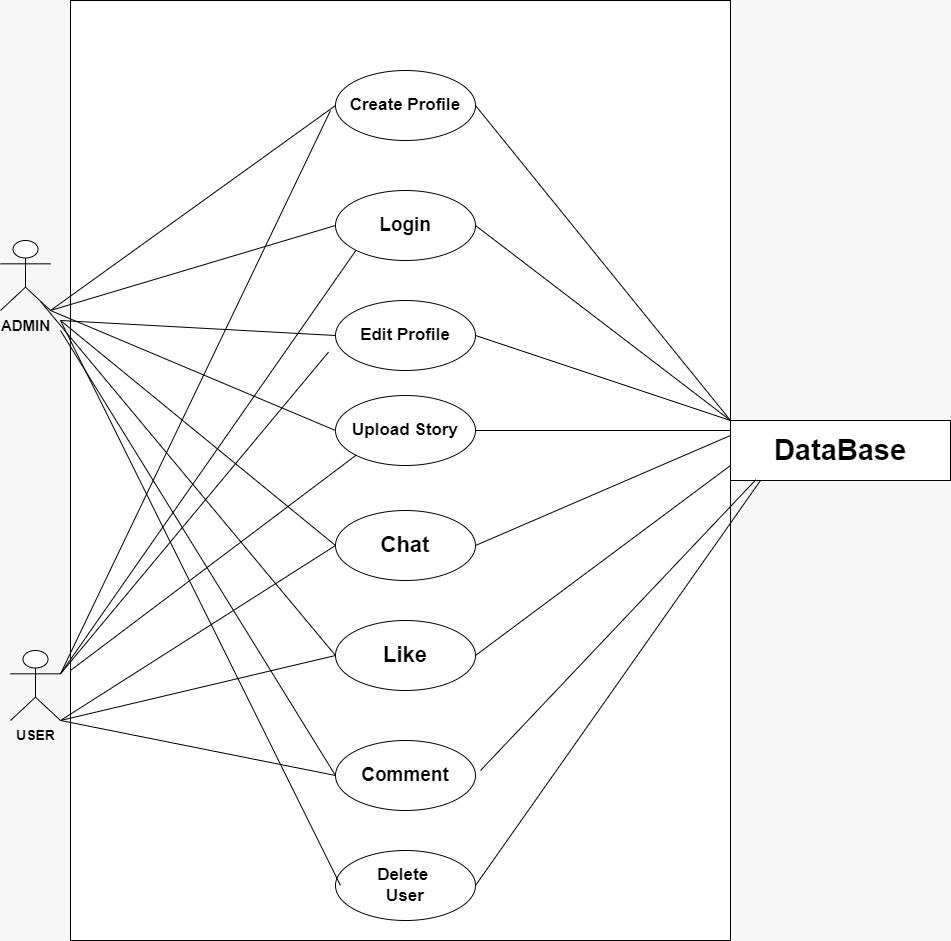
Because of the nature of social networking sites, there’s new content constantly available from users .Take advantage of this by including content, both on the home page and on individual users’ profiles or main pages, compose do for these updates .Updated content keeps users coming back ,as there’s more to see each time they visit.

* WHATREALLYNEEDSTOBEHERE?

When designing your user interface, ask yourself this question repeatedly. Is it really necessary for ascertain bit of information or an option to be included on a given page? If the answer is no, then don’t include it there. Only give your users the minimum necessary information to perform the tasks you want them toper form. Just make sure if there’s additional information some users might want that you it easy enough for them to find it.

# System Model

* + 1. **ER Diagram**
    2. **Use Case Diagram**

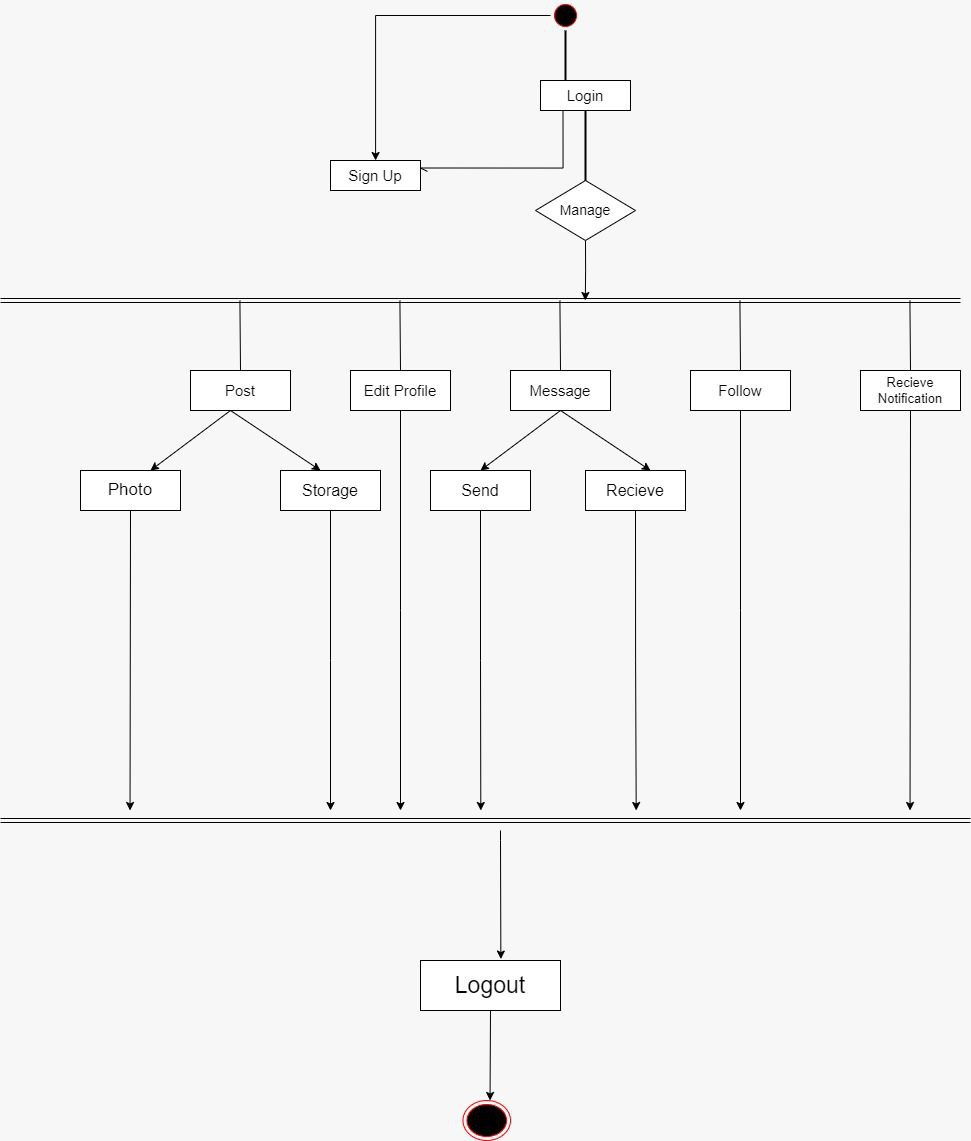


## Activity Diagram.

* + - * Admin Side

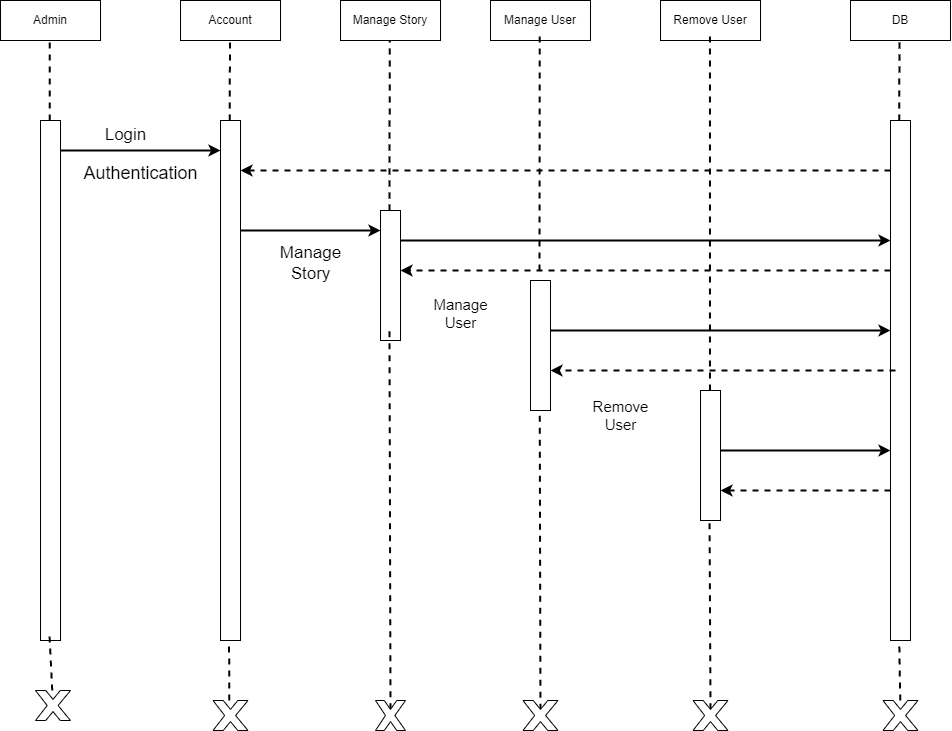
## 

* + - * User Side

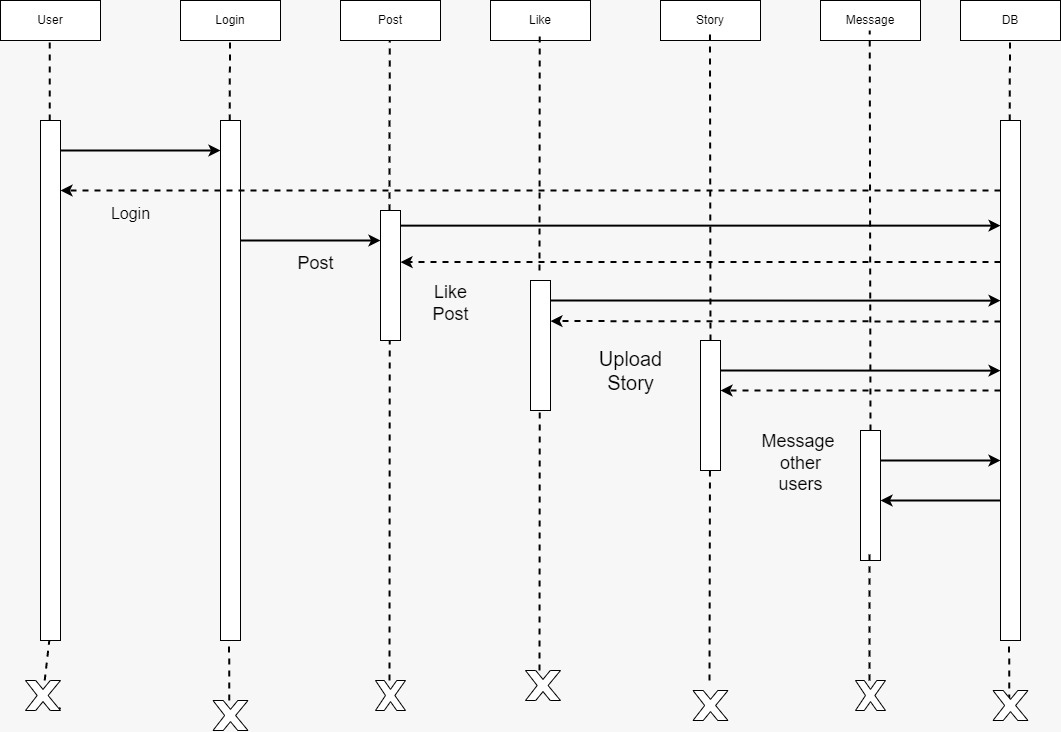


## Sequence Diagram

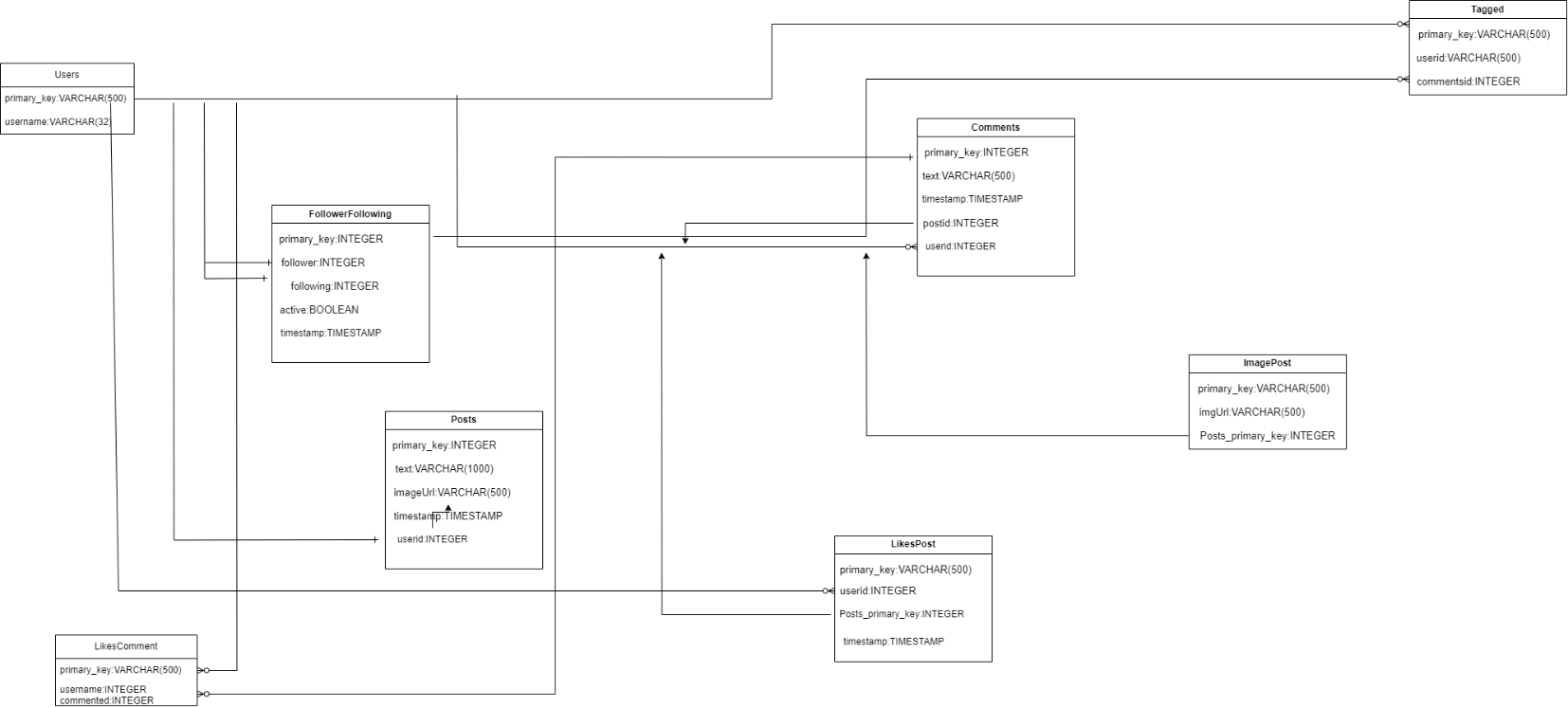
## Admin Side

****

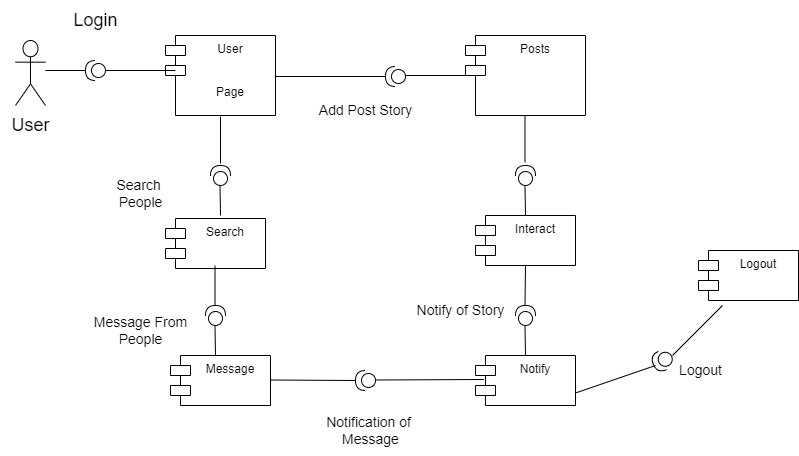
* **User Side**

****

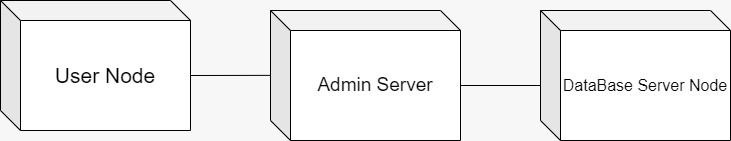
* + 1. **Class Diagram**

****

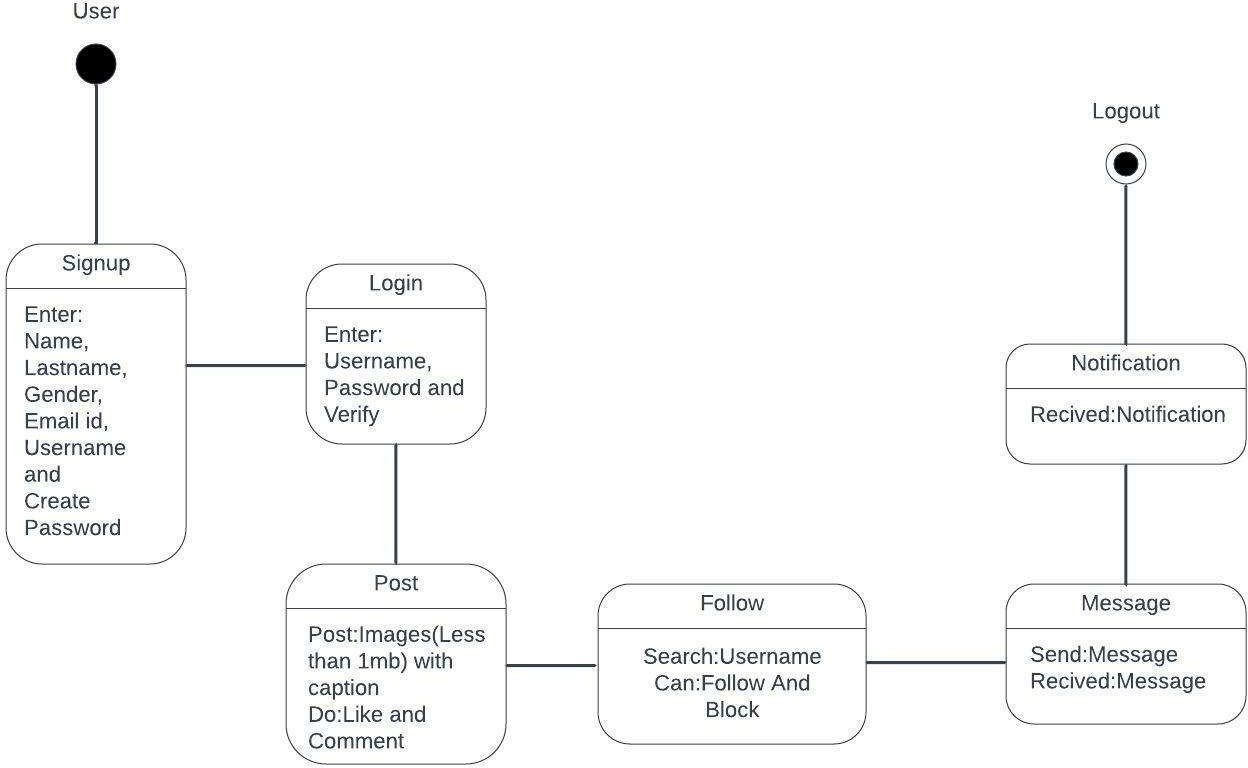
## Component Diagram

****

* + 1. **Deployment Diagram**



* + 1. **State/Transition Diagram**



# Data Dictionary

**hype\_db**

# admin

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** | **Linksto** | **Comments** | **Mediatype** |
| id *(Primary)* | int(11) | No |  |  |  |  |
| name | varchar(250) | No |  |  |  |  |
| password | Text | No |  |  |  |  |
| password\_text | Text | No |  |  |  |  |

**Indexes**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Keyname** | **Type** | **Unique** | **Packed** | **Column** | **Cardinality** | **Collation** | **Null** | **Comment** |
| PRIMARY | BTREE | Yes | No | id | 0 | A | No |  |

# Comments

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** | **Links to** | **Comments** | **Media type** |
| id *(Primary)* | int(11) | No |  |  |  |  |
| post\_id | int(11) | No |  |  |  |  |
| user\_id | int(11) | No |  |  |  |  |
| comment | Text | No |  |  |  |  |
| created\_at | timestamp | No | current timestamp() |  |  |  |

**Indexes**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Keyname** | **Type** | **Unique** | **Packed** | **Column** | **Cardinality** | **Collation** | **Null** | **Comment** |
| PRIMARY | BTREE | Yes | No | id | 0 | A | No |  |

# follow\_list

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** | **Linksto** | **Comments** | **Mediatype** |
| id *(Primary)* | int(11) | No |  |  |  |  |
| follower\_id | int(11) | No |  |  |  |  |
| user\_id | int(11) | No |  |  |  |  |

**Indexes**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Keyname** | **Type** | **Unique** | **Packed** | **Column** | **Cardinality** | **Collation** | **Null** | **Comment** |
| PRIMARY | BTREE | Yes | No | id | 0 | A | No |  |

# Likes

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** | **Linksto** | **Comments** | **Mediatype** |
| id *(Primary)* | int(11) | No |  |  |  |  |
| post\_id | int(11) | No |  |  |  |  |
| user\_id | int(11) | No |  |  |  |  |

**Indexes**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Keyname** | **Type** | **Unique** | **Packed** | **Column** | **Cardinality** | **Collation** | **Null** | **Comment** |
| PRIMARY | BTREE | Yes | No | id | 0 | A | No |  |

# Messages

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** | **Linksto** | **Comments** | **Mediatype** |
| id *(Primary)* | int(11) | No |  |  |  |  |
| from\_user\_id | int(11) | No |  |  |  |  |
| to\_user\_id | int(11) | No |  |  |  |  |
| Msg | Text | No |  |  |  |  |
| read\_status | int(11) | No | 0 |  |  |  |
| created\_at | timestamp | No | current\_timestamp() |  |  |  |

**Indexes**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Keyname** | **Type** | **Unique** | **Packed** | **Column** | **Cardinality** | **Collation** | **Null** | **Comment** |
| PRIMARY | BTREE | Yes | No | id | 3 | A | No |  |

# Notifications

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** | **Linksto** | **Comments** | **Mediatype** |
| id *(Primary)* | int(11) | No |  |  |  |  |
| to\_user\_id | int(11) | No |  |  |  |  |
| message | Text | No |  |  |  |  |
| created\_at | timestamp | No | current\_timestamp() |  |  |  |
| from\_user\_id | int(11) | No |  |  |  |  |
| read\_status | int(11) | No | 0 |  |  |  |
| post\_id | Text | Yes | *NULL* |  |  |  |

**Indexes**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Keyname** | **Type** | **Unique** | **Packed** | **Column** | **Cardinality** | **Collation** | **Null** | **Comment** |
| PRIMARY | BTREE | Yes | No | id | 2 | A | No |  |

# Posts

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** | **Linksto** | **Comments** | **Mediatype** |
| id *(Primary)* | int(11) | No |  |  |  |  |
| user\_id | int(11) | No |  |  |  |  |
| post\_img | Text | No |  |  |  |  |
| post\_text | Text | No |  |  |  |  |
| created\_at | timestamp | No | current\_timestamp() |  |  |  |

**Indexes**

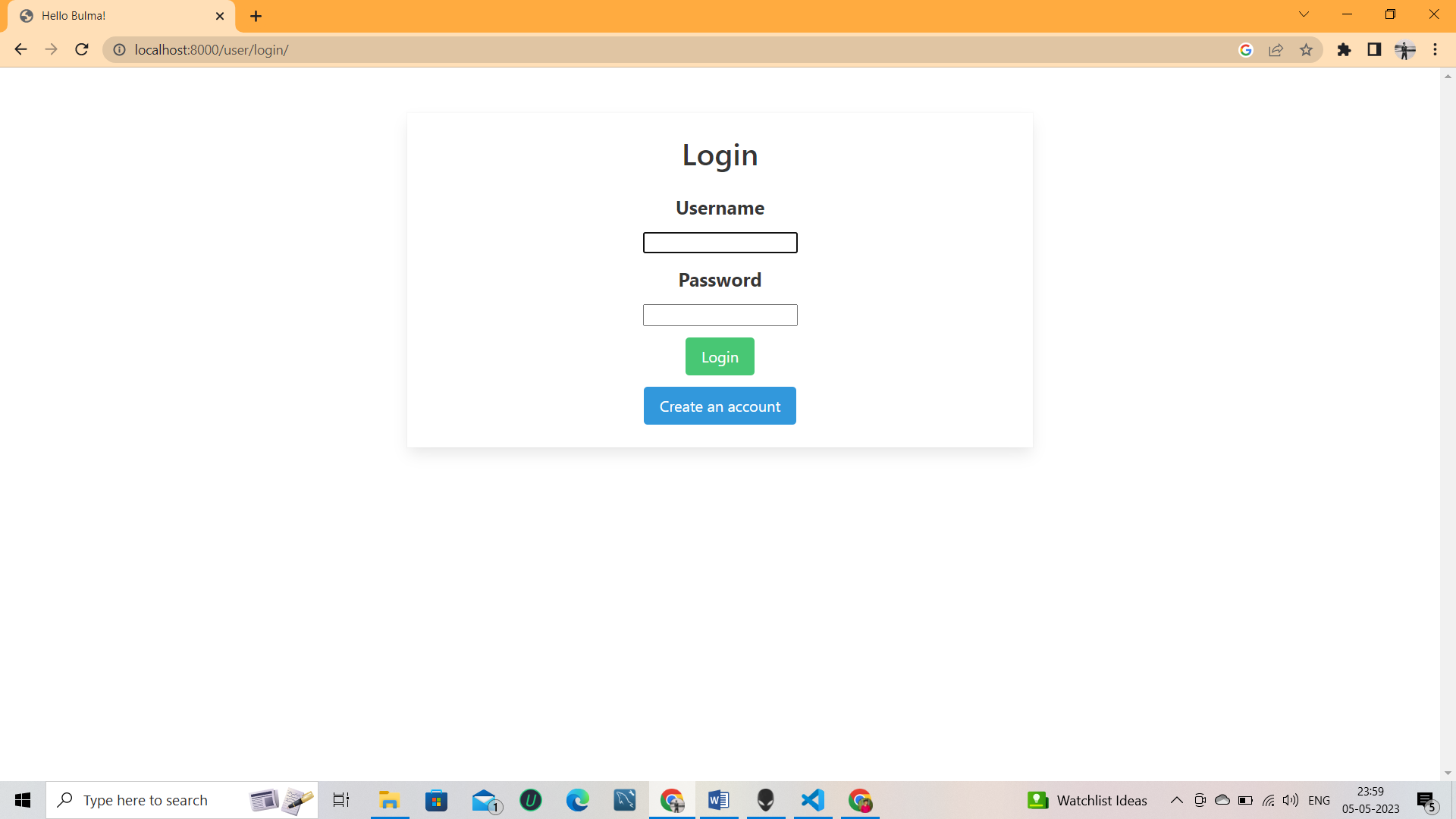
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Keyname** | **Type** | **Unique** | **Packed** | **Column** | **Cardinality** | **Collation** | **Null** | **Comment** |
| PRIMARY | BTREE | Yes | No | id | 1 | A | No |  |

# Users

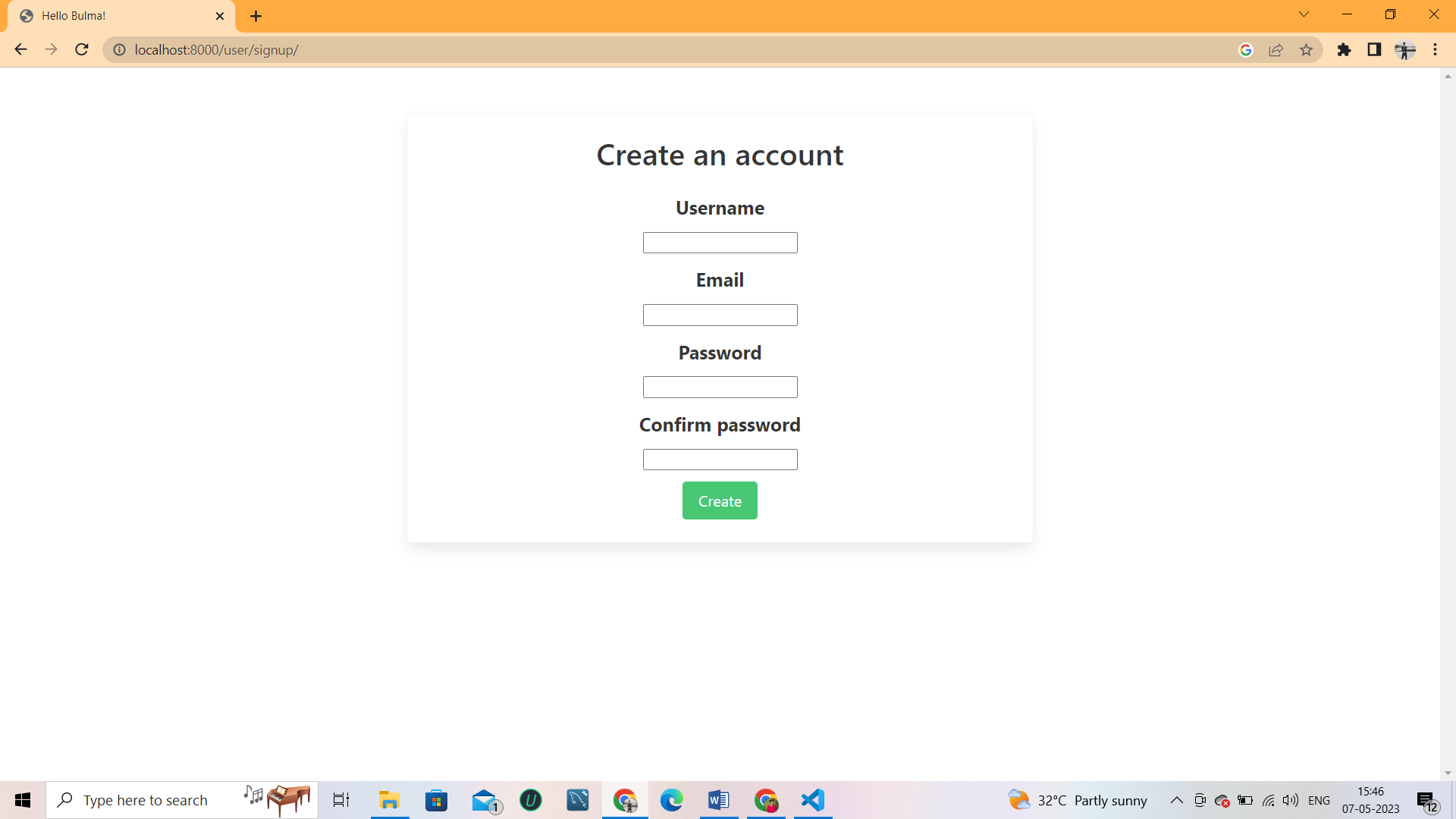
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** | **Links**  **to** | **Comments** | **Media**  **type** |
| id *(Primary)* | int(11) | No |  |  |  |  |
| first\_name | varchar(255) | No |  |  |  |  |
| last\_name | varchar(255) | No |  |  |  |  |
| gender | int(11) | No |  |  |  |  |
| email | varchar(255) | No |  |  |  |  |
| username | varchar(255) | No |  |  |  |  |
| password | Text | No |  |  |  |  |
| profile\_pic | Text | No | 'default\_profile.jpg' |  |  |  |
| created\_at | timestamp | No | current\_timestamp() |  |  |  |
| updated\_at | timestamp | Yes | current\_timestamp() |  |  |  |
| ac\_status | int(11) | No |  |  | 0=not  verified,1=active,2=blocked |  |

# User Interface

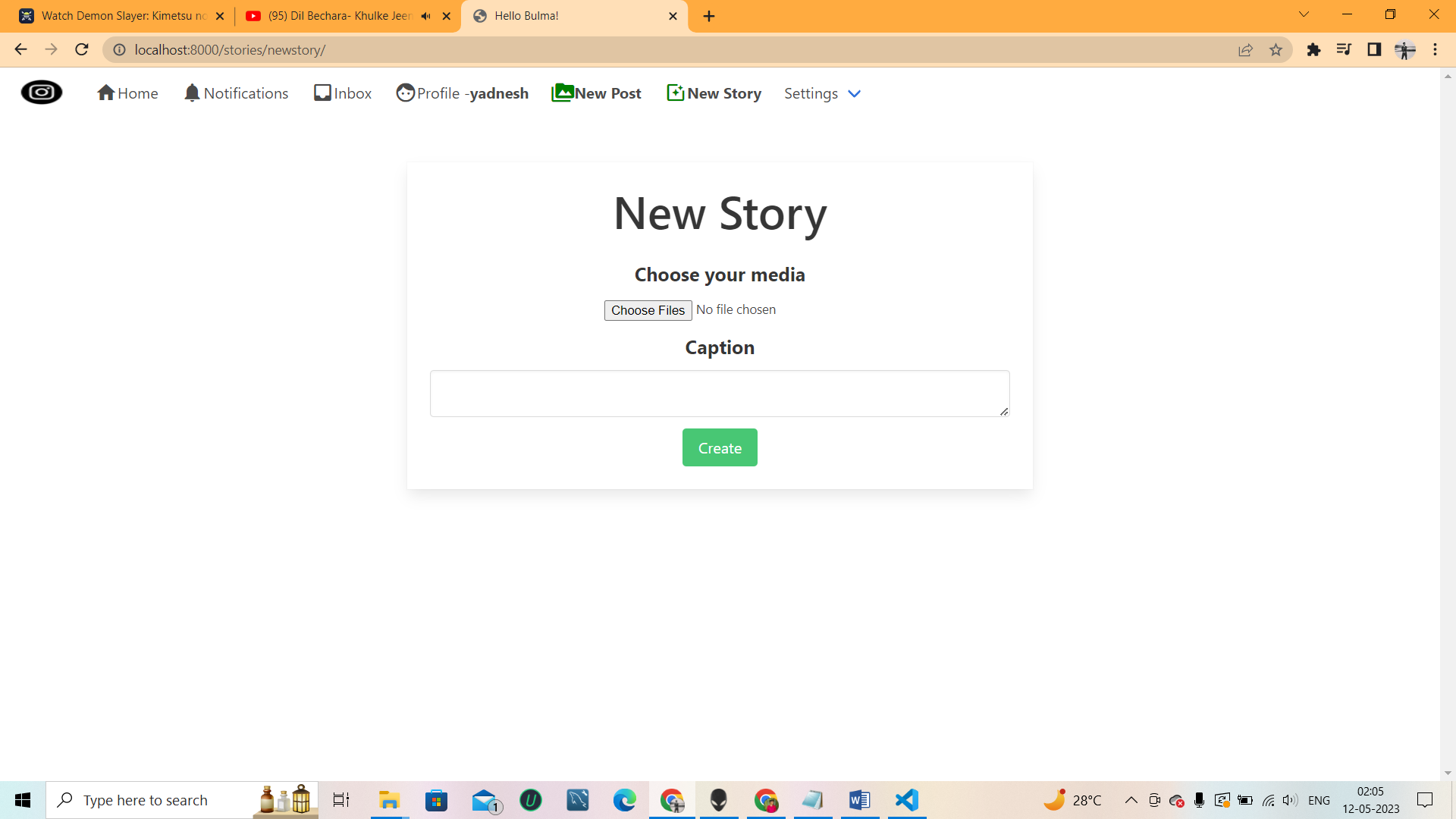
1. Login Page:



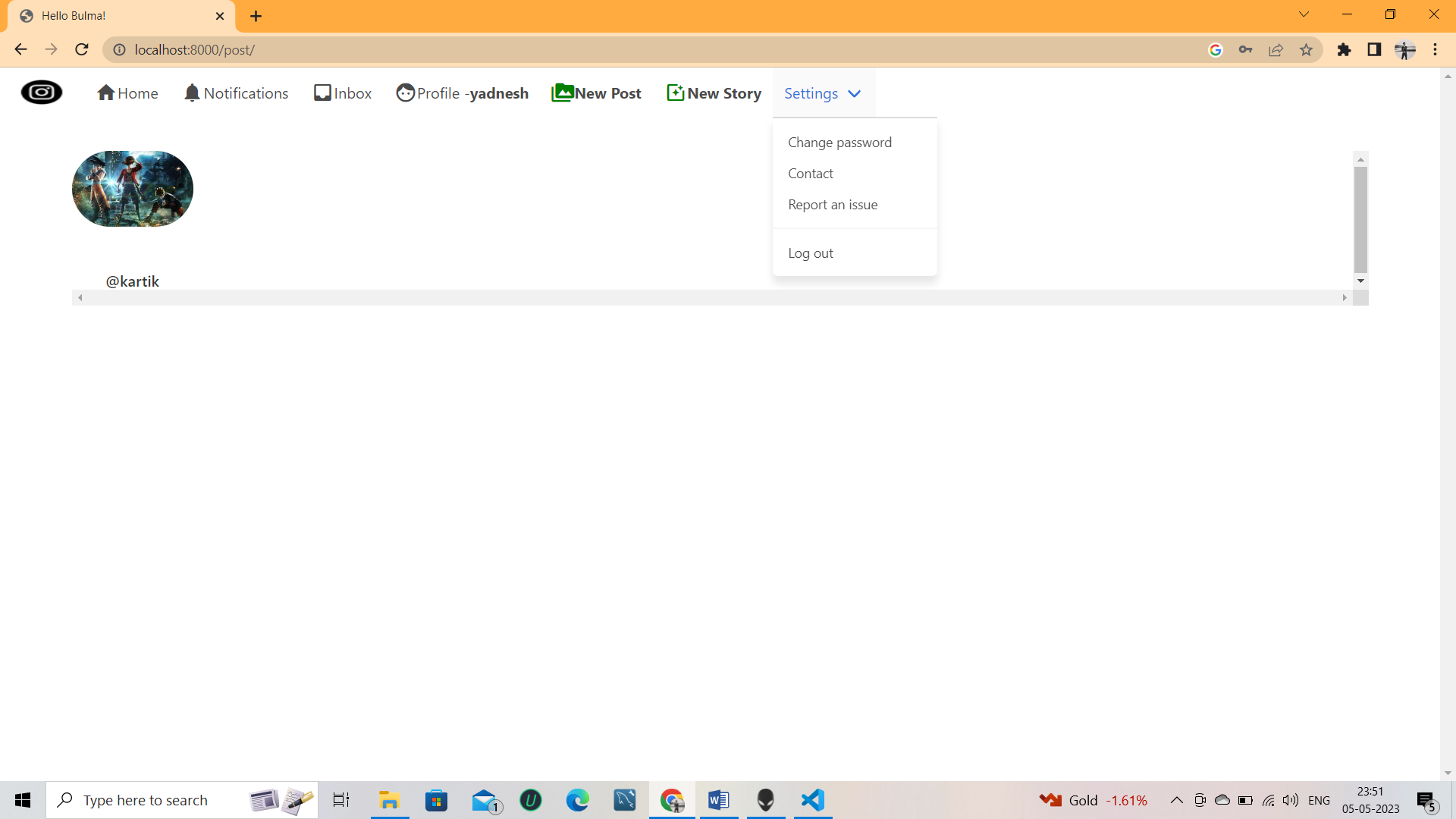
1. Signup Page



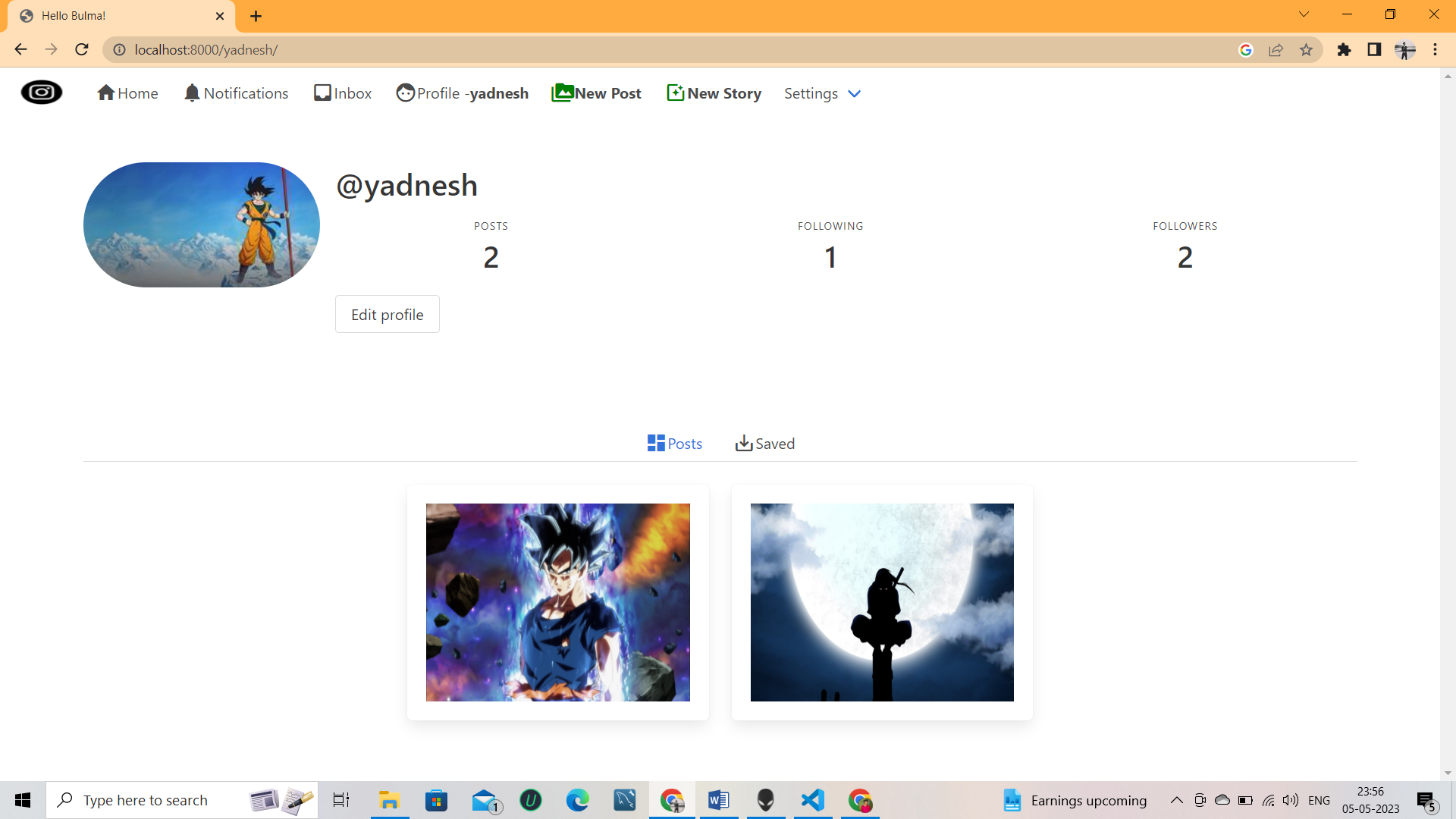
3.Story



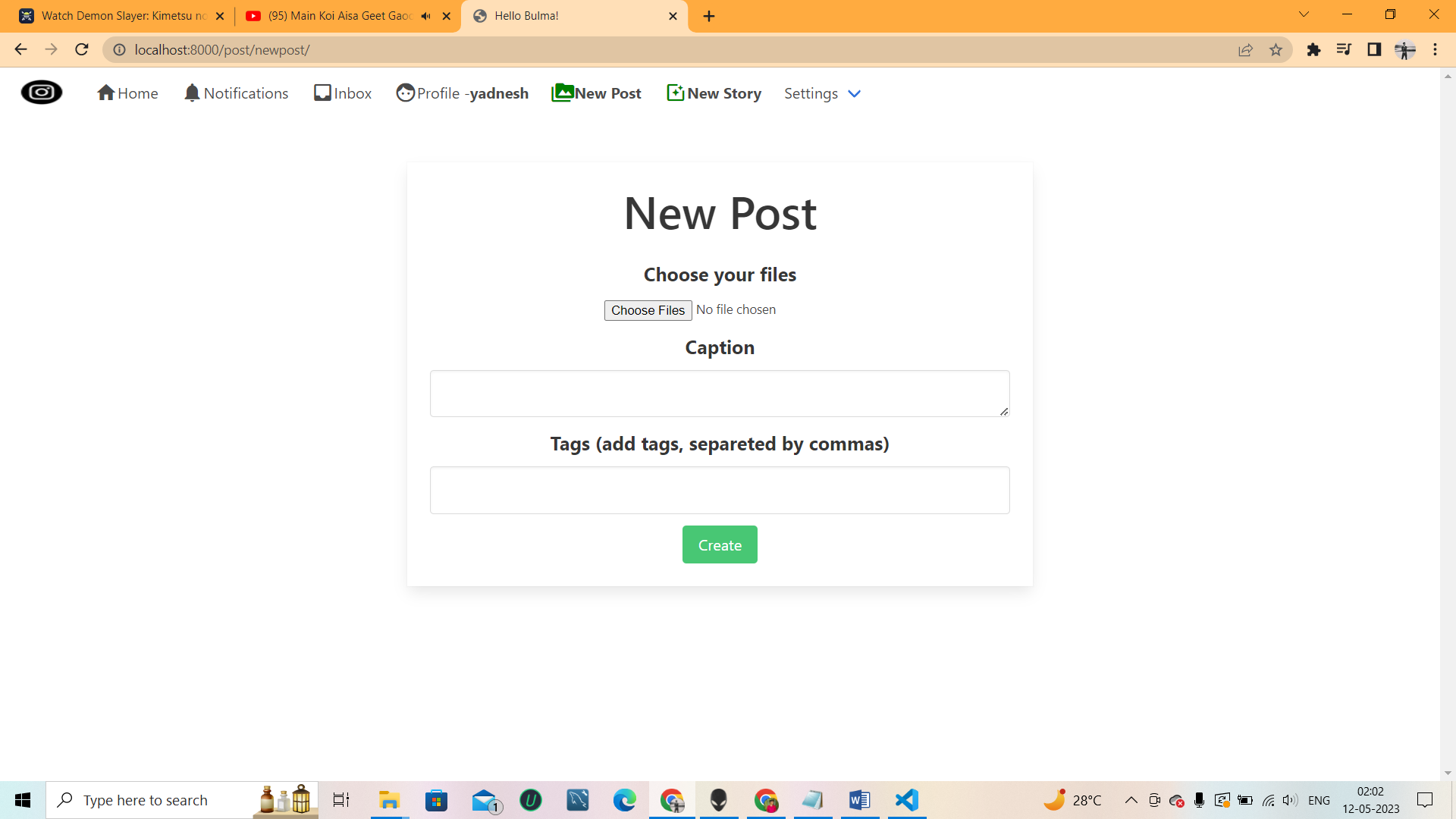
* 1. Homepage



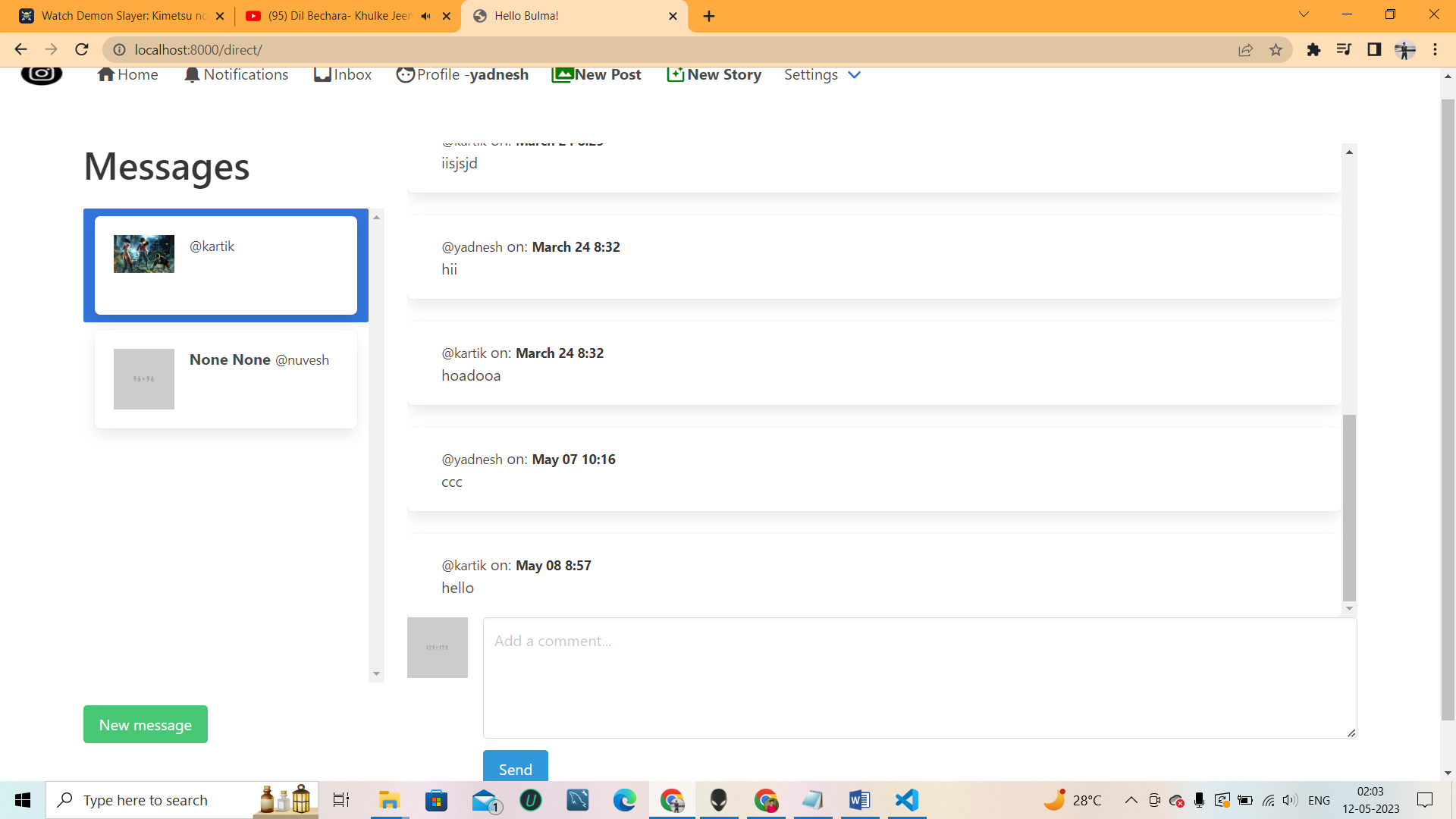
* 1. Profile page



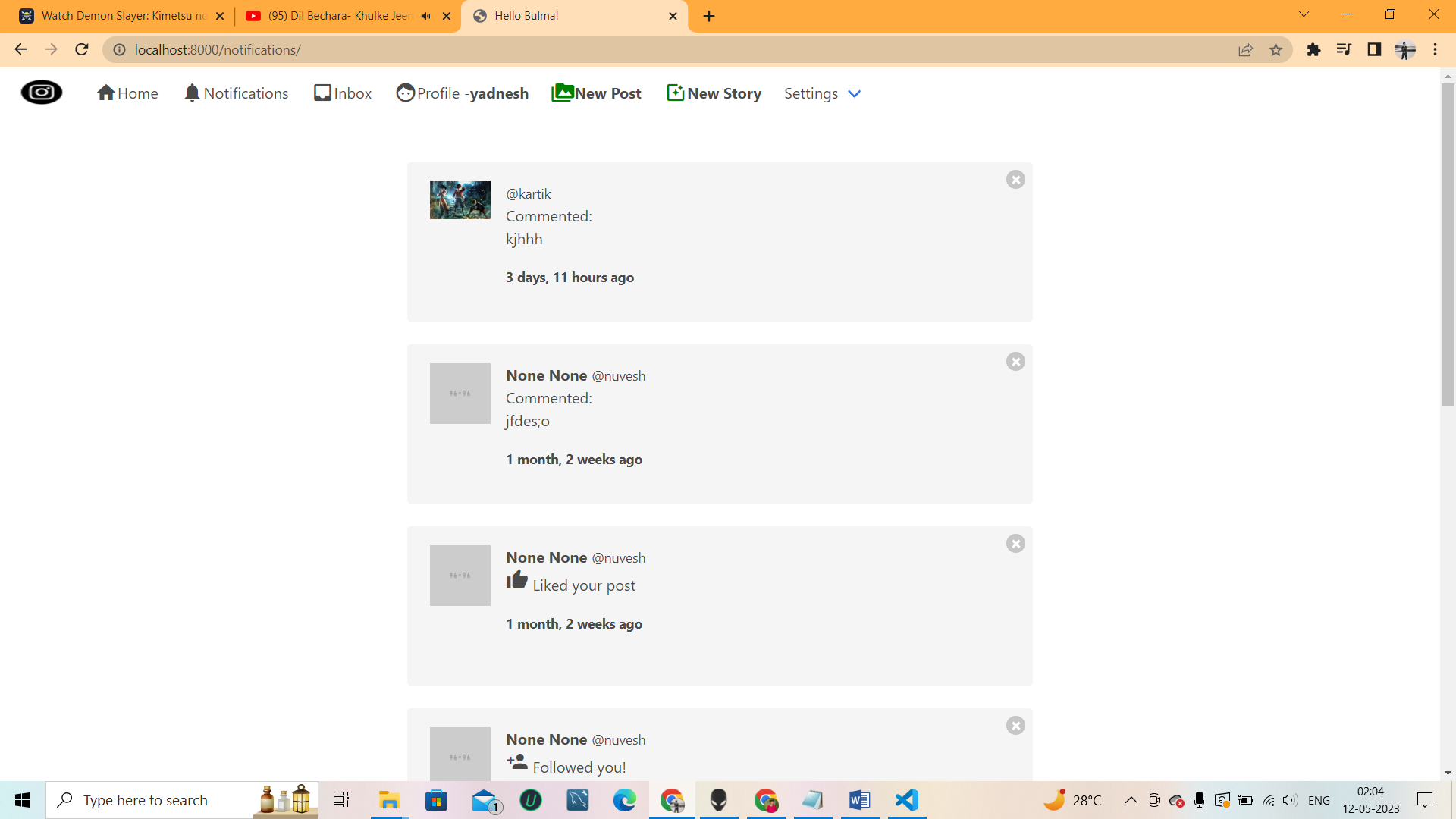
* 1. Posting



* 1. Message



* 1. Notification



# Implementation Details

* 1. Software/hardware specifications

**Hardware Interface**:

**CATEGORY WINDOWS MACOSX**

|  |  |  |
| --- | --- | --- |
| Processor | Intel Core i5 and above (such as Intel Core i7, or Intel Core i9) **OR** the following or above from AMD: Ryzen, Athlon 3000, Athlon PRO, or A-Series PRO  New Macs with the new M1 chip from Apple that is based on the ARM architecture are not currently capable of running a Windows version that can support some of the required software; as such, those are currentlyinsufficientfor your studies at CBS. | |
| HardDrive | 128Gigabytes(GB)\* | 128GB+SolidStateDisk(SSD)with atleast50GBof free disk space |
| Memory | 8 GB | |
| Wired Networking(Optional) | Ethernet LAN Port or USB Ethernet Adapter/Dongle | |
| In order to use the CBS wired network connection on your ultra-light or MacBook Air PC, youwillneedto purchase USB-to-EthernetAdapter. | |
| Wireless Networking | 802.11n | |

* 1. Operating System Requirements

SupportedOperating System

* + - Windows 11, Windows 10
* Ubuntu
  1. Software Requirements

For frontend:

* + - HTML
    - CSS
    - Bootstrap

**Backend:-**

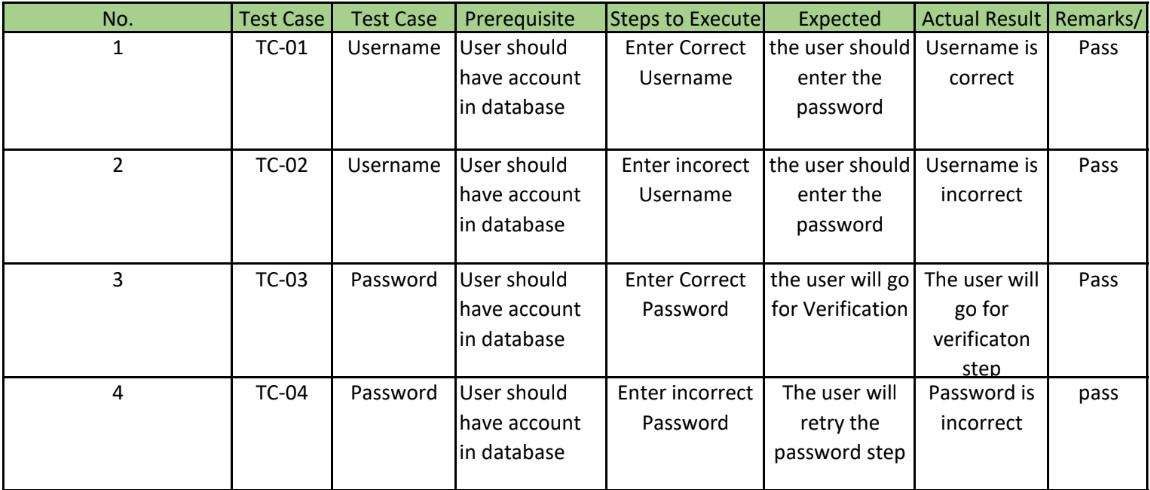
* + - * Python
    - SQL
  1. Computer Security

**AntivirusandSpywareProtection**

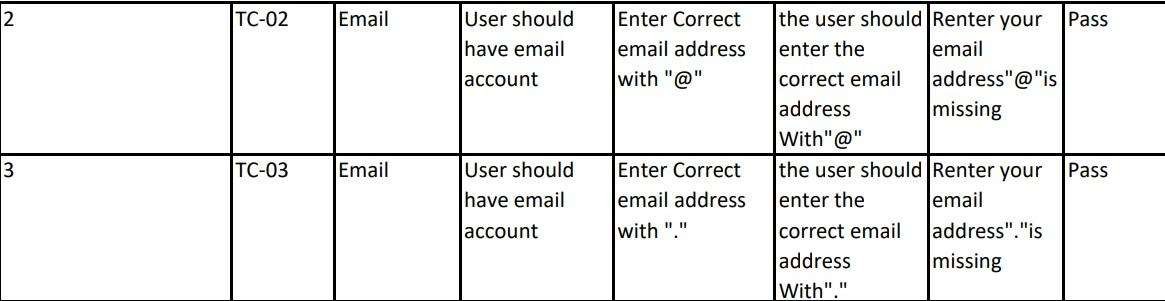
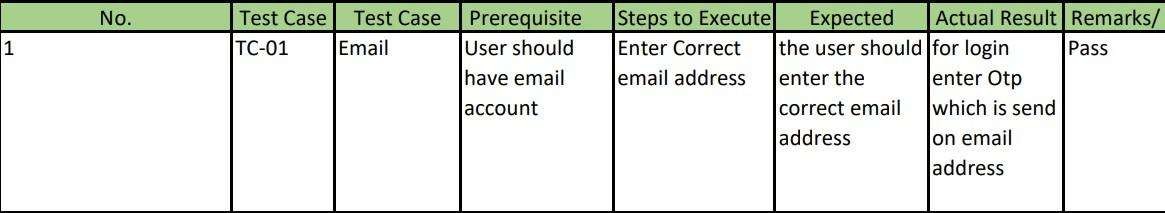
* + - Must be updated with the latest virus definitions/updates.
    - Required for both Mac and Windows.
    - **ForWindows:** We recommendusing the built-in Windows Security.

1. **Testing**

**Test Case forLogin:**



**TestCasefor Signup:**



# Conclusion

Social media is a really convenient and important communicate network for all the people nowadays. We can use it to know friends and keep contact with friends that came from different countries. Wean also share our ideas so quickly so that all the things could develop so fast because people could tell us their ideas and we could improve it immediately. We could also learn new things on social media by watching or reading the things that people shared onto the social media. People could also sell things on social media freely which could reduce the expend inure of advertisements.

There are more advantages for using social media, however, there is always advantages and disadvantages for a thing. As social media is too convenient for people, almost most of them don’t even have to 'speak out' to communicate with people. No longer, people will lost their communication skills. The more serious problem is many people utilized the power of social media and used it to bully someone. The power of social media is also same as the one in real life. A little of them used social-media to do things that against the law, which is a fool behavior.

Social media changed our life so much. Our life became more convenient because social media is a very useful tool for us in 21st century, it could help us to improve our life. However, we have to aware of how we use them. If we could use the social media smartly, having social media will become a good change for us.

# Future Scope

* + Adding option of PostVideo.
  + Adding option of VideoCalling Messages
  + Adding block feature.

# References

* Youtube:-
* Github
* Bulma