

Report

a. Requirements:

Phase 3:

User Dashboard:

The user dashboard gives the user a comprehensive picture of the system, allowing them to access the most critical data, operations, and controls. In practice, a dashboard frequently serves as a homepage. This Dashboard is designed for users who wish to use our application. User Dashboard functionalities are different when compared with other Dashboards. Here we help users choose the Designer of their choice. Users will not be given access to add or remove users or add plans. The features that users can experience are listed below.

- **Task 1:** New users are allowed to register into the application by clicking on the register button on the login page where they redirect to registration for requesting details to be filled like email id password and more.
- **Task 2:** Users should be able to view Designers profiles to know their past work.
- **Task 3:** Users are given an option to chat with the designers they like to know more about their work.
- **Task 4:** User are allowed to give feedback to the designers based on conversation.
- **Task 5:** Users have access to view their details and can verify them.
- **Task 6:** Users can also change the password if there is an emergency.

Features of User:

- **Chat:** The users are allowed to chat with the designers to choose the designer of their choice.
- **View Stylish:** The users can view designers and check their profiles to know their past work and interact with them.
- **Feedback:** After a chat with the designer, the users can give feedback based on the conversation.
- **My Details:** The users can view the details given during registration, like their email or chosen plan.
- **Change Password:** The users will have an option to change their passwords in case of an emergency.

b. UML design. You must include the following diagrams:

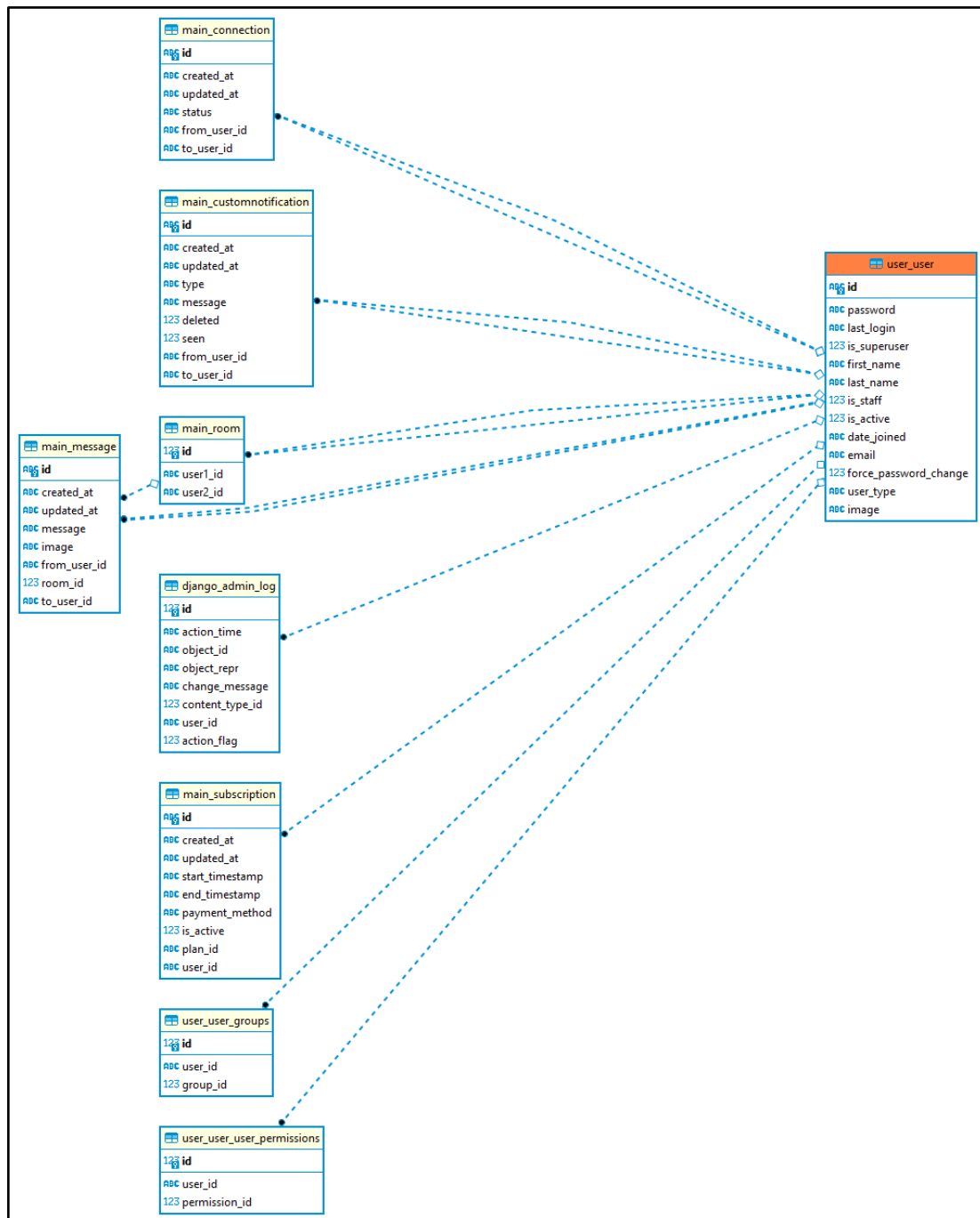


Fig . ER - Diagram

Class Diagram:

A UML class diagram is used to not only describe a system's object and information structures but also to show how the application communicates with its users. It can be used for everything from modelling the static view of an application to expressing system responsibilities. Classes represent an abstraction of entities having similar features in a UML class diagram. Static relationships between classes are represented by associations. Aggregation is a sort of connection in which components are put together or modified to make a more complicated object.

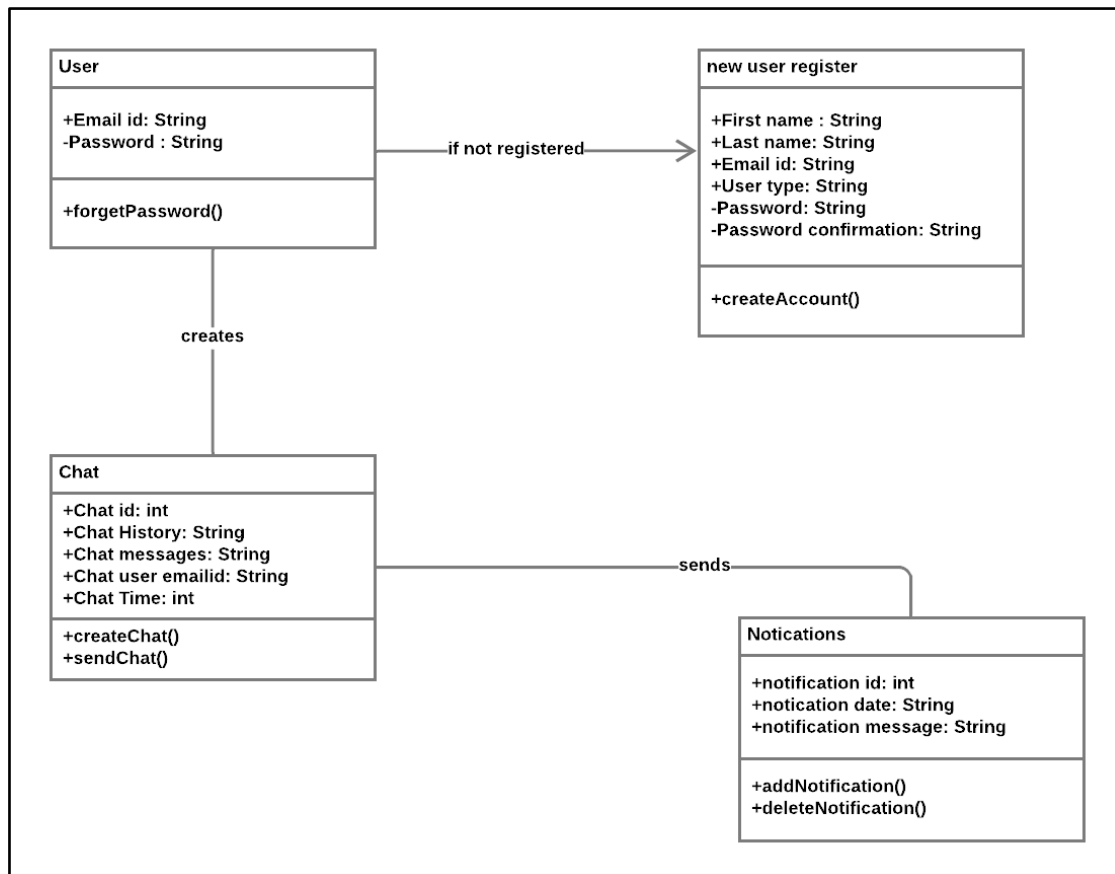


Fig. Class Diagram

Sequence Diagram for Admin:

In UML, the analysis and design phases are represented by a sequence diagram. It is an interaction diagram that shows how operations are performed. A sequence diagram is frequently used to show a use case is chronologically structured event flow. It does an excellent way of highlighting how objects communicate with one another and what messages activate those communications.

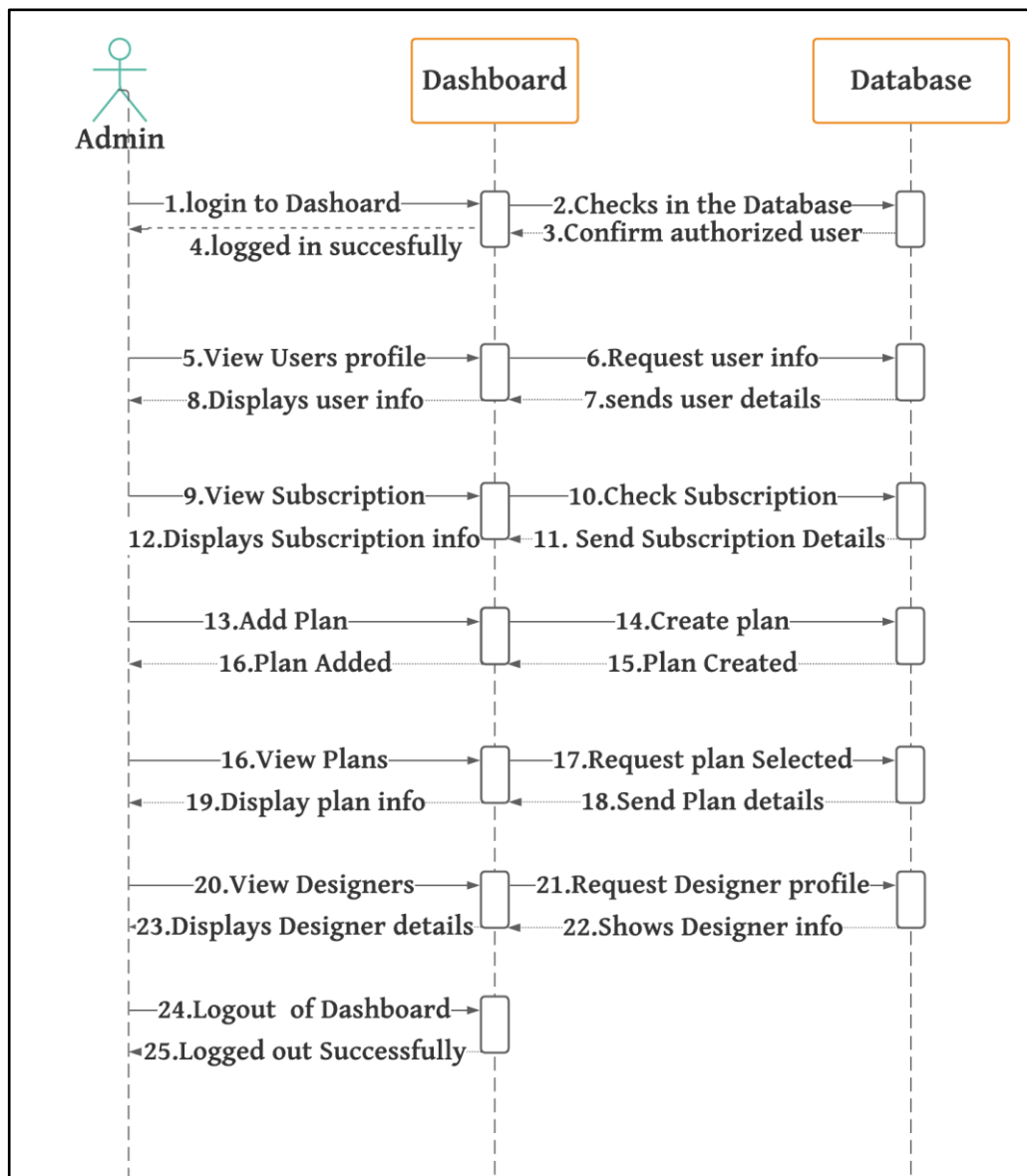


Fig. Sequence Diagram for Admin

Description: The above sequence diagram depicts the orderly execution of actions between the admin dashboard and the database.

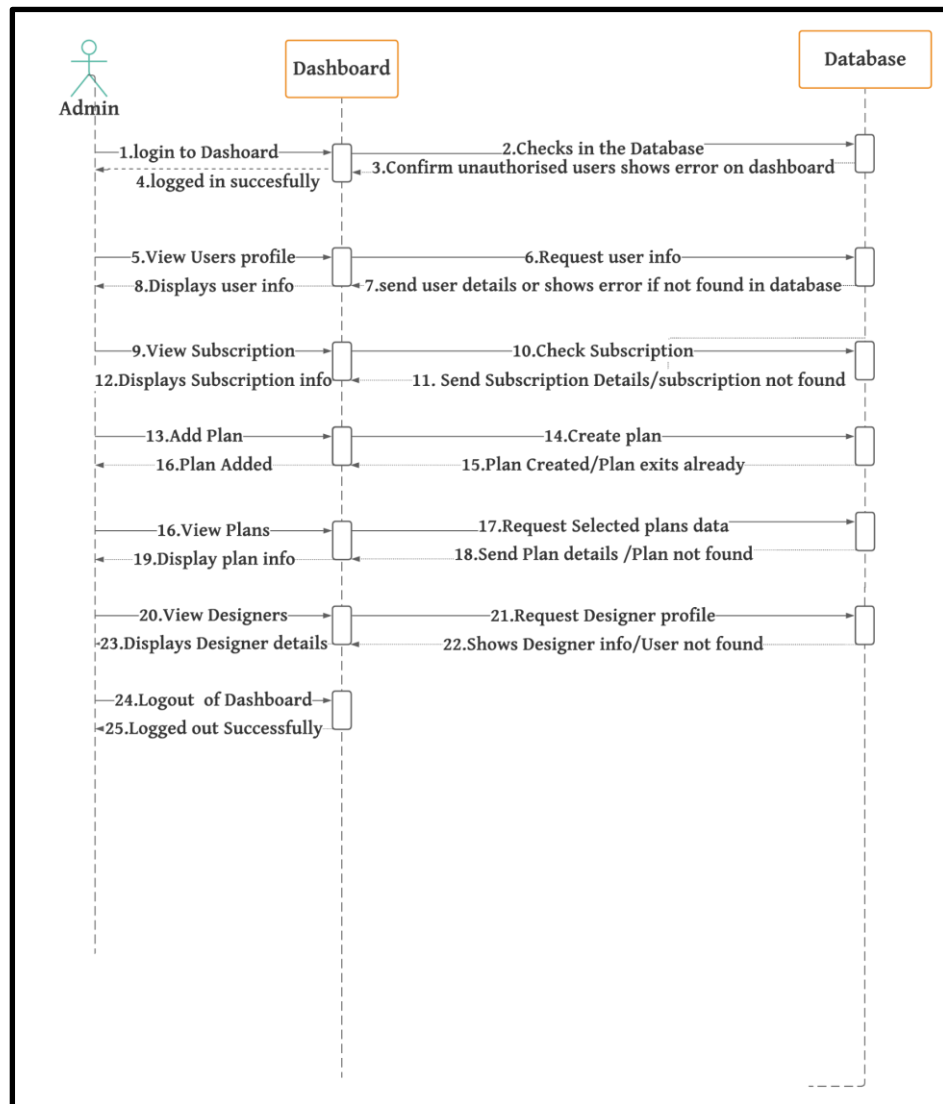


Fig: Error case Sequence Diagram for Admin

Description: The above sequence diagram depicts the orderly execution of actions between the admin dashboard and the database.

Sequence Diagram for Designer:

In UML, the analysis and design phases are represented by a sequence diagram. It is an interaction diagram that shows how operations are performed. A sequence diagram is frequently used to show a use case is chronologically structured event flow. It does an excellent way of highlighting how objects communicate with one another and what messages activate those communications.

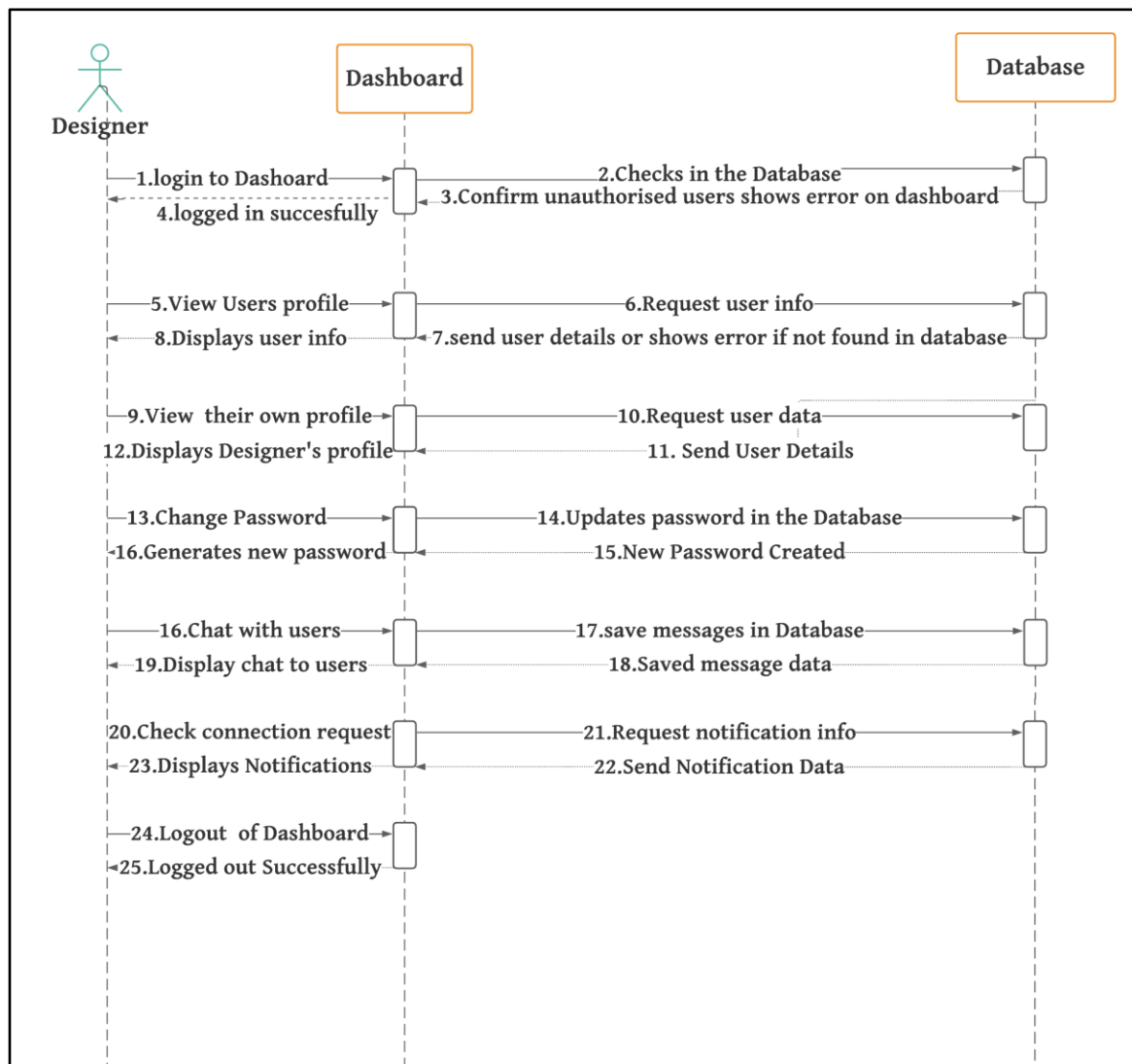


Fig. Sequence Diagram

Description: The above sequence diagram depicts the orderly execution of actions between the Designer's dashboard and the database.

Sequence Diagram for User:

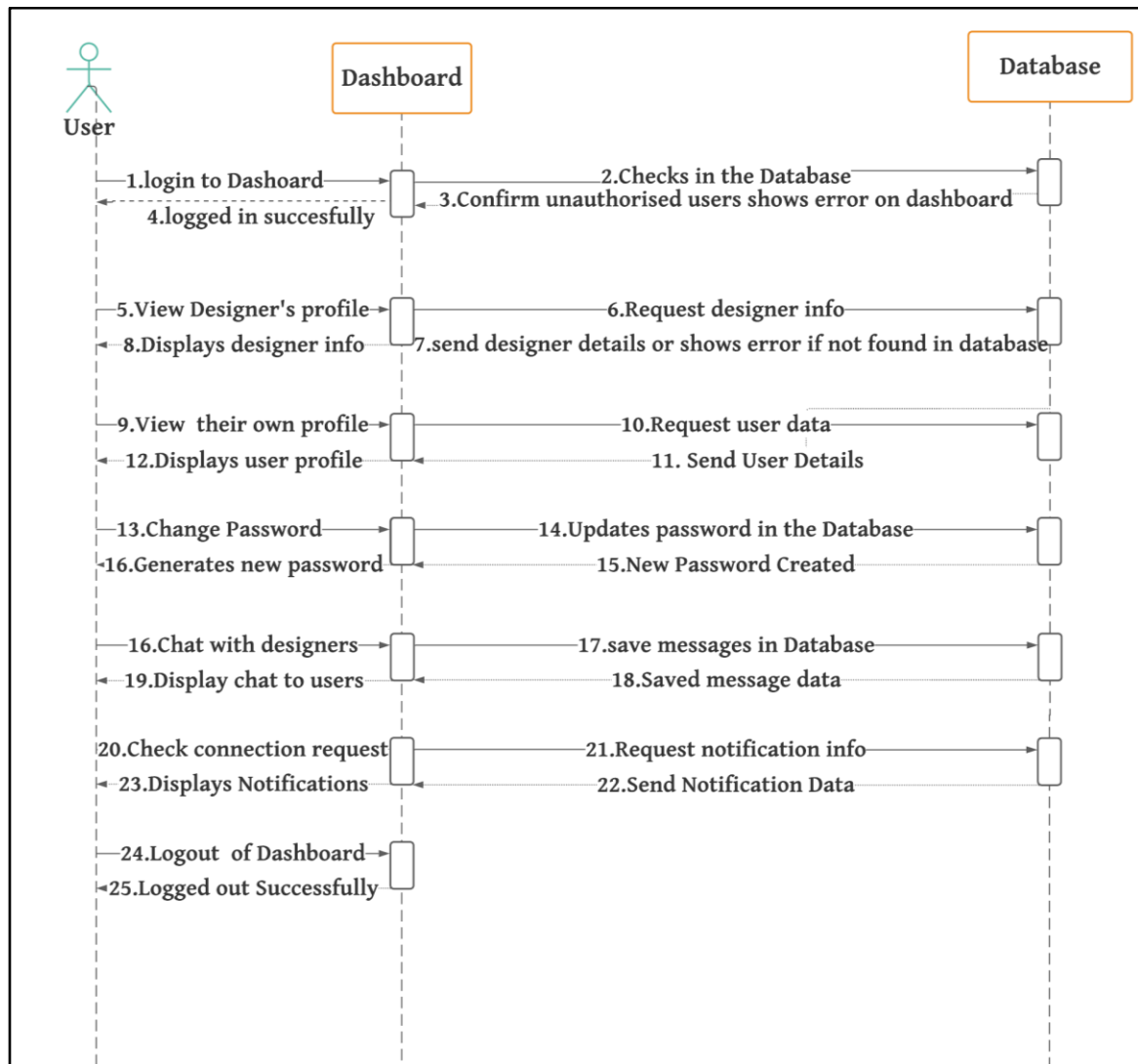


Fig. Sequence Diagram for User

Description: The above sequence diagram depicts the orderly execution of actions between the user's dashboard and the database.

Use Case Diagram for Admin:

A use case diagram is used in UML to identify system functionality during the analysis phase. It defines how users and external devices interact with the system under development. It does not go into great detail. Instead, it focuses on the connections between use cases, actors, and systems. A use case diagram usually comprises four components. Actors, systems, use cases, and relationships are all examples. The actors are anyone or anything which interacts with the system.

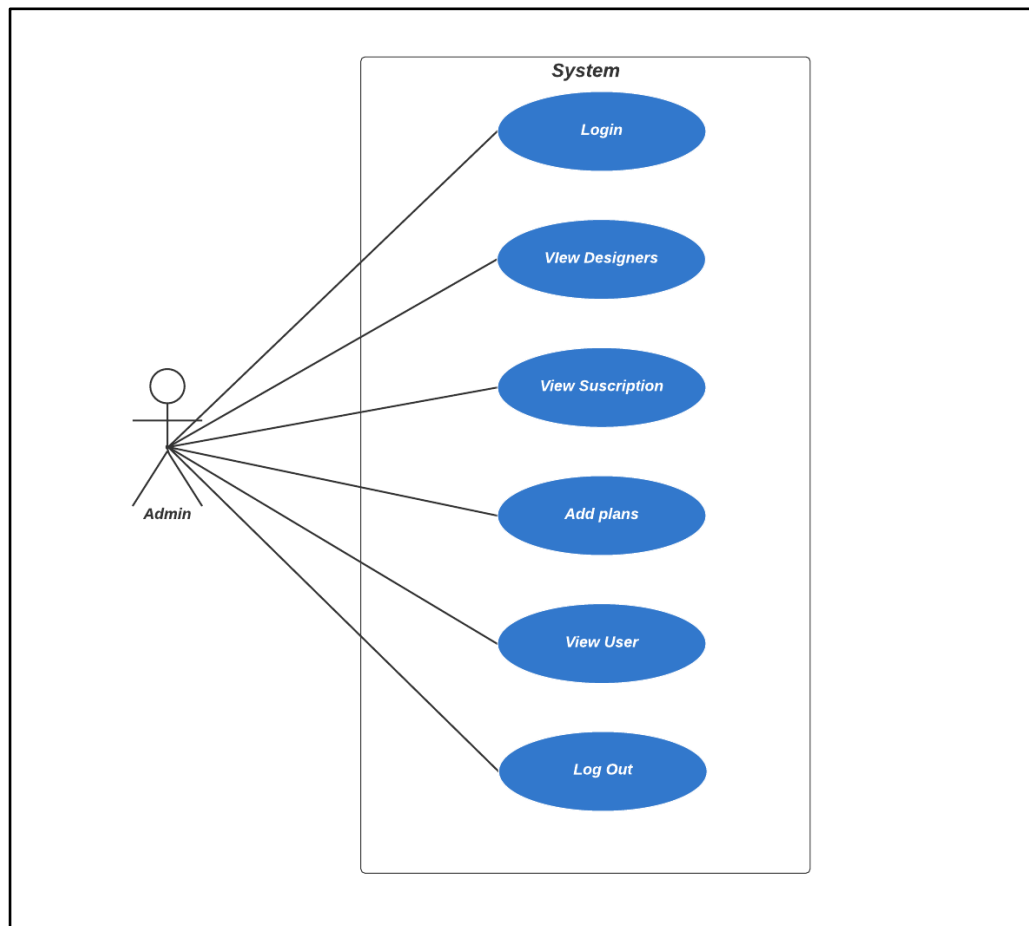


Fig. Use Case Diagram

Description: In the above use case diagram, the actors in the diagram are Admin who manages the application like adding plans, viewing subscriptions, and checking the status of the registered users.

Use Case Diagram for Designer:

A use case diagram is used in UML to identify system functionality during the analysis phase. It defines how users and external devices interact with the system under development. It does not go into great detail. Instead, it focuses on the connections between use cases, actors, and systems. A use case diagram usually comprises four components. Actors, systems, use cases, and relationships are all examples. The actors are anyone or anything which interacts with the system.

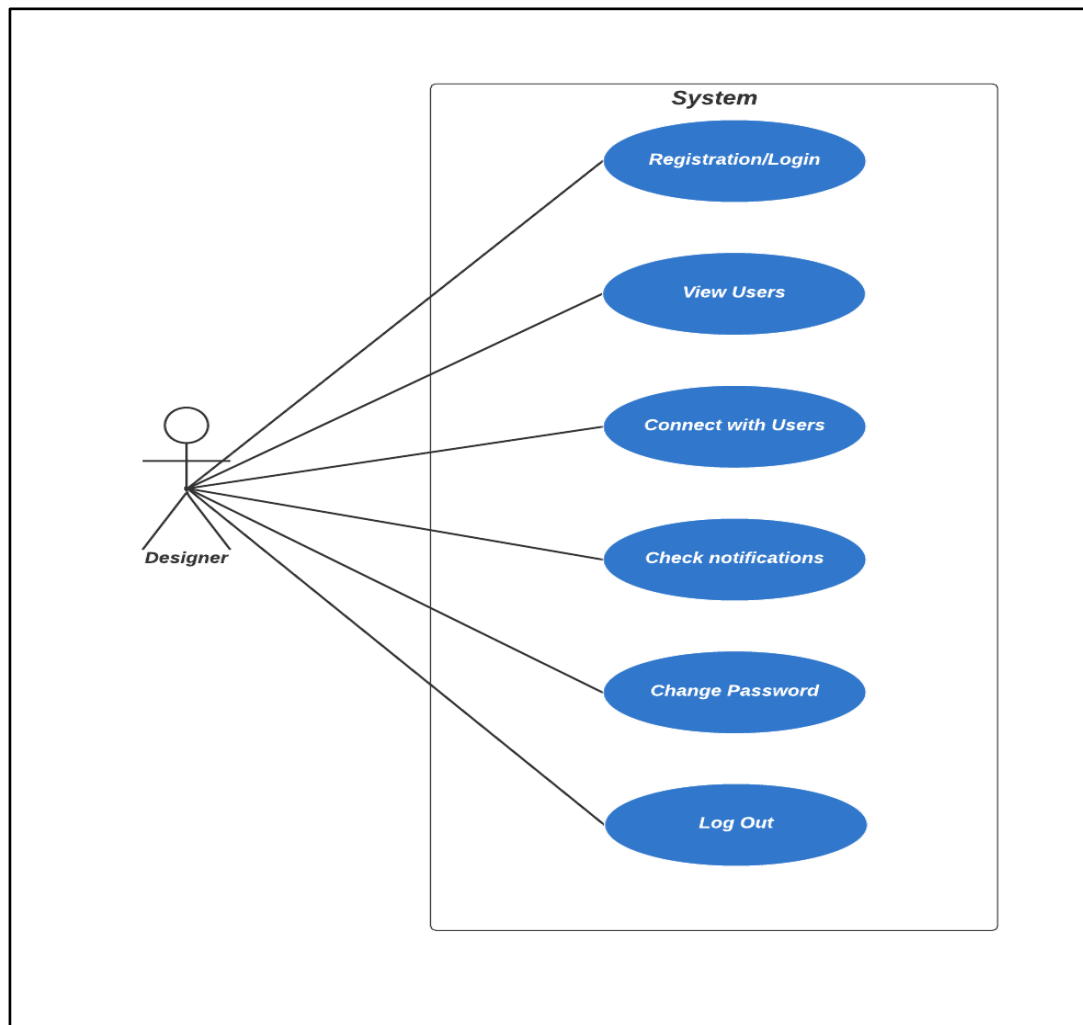


Fig. Use Case Diagram (normal case)

Description: In the above use case diagram, the actors in the diagram are Designer who has the ability to see users, read notifications, connect with people, and change their password

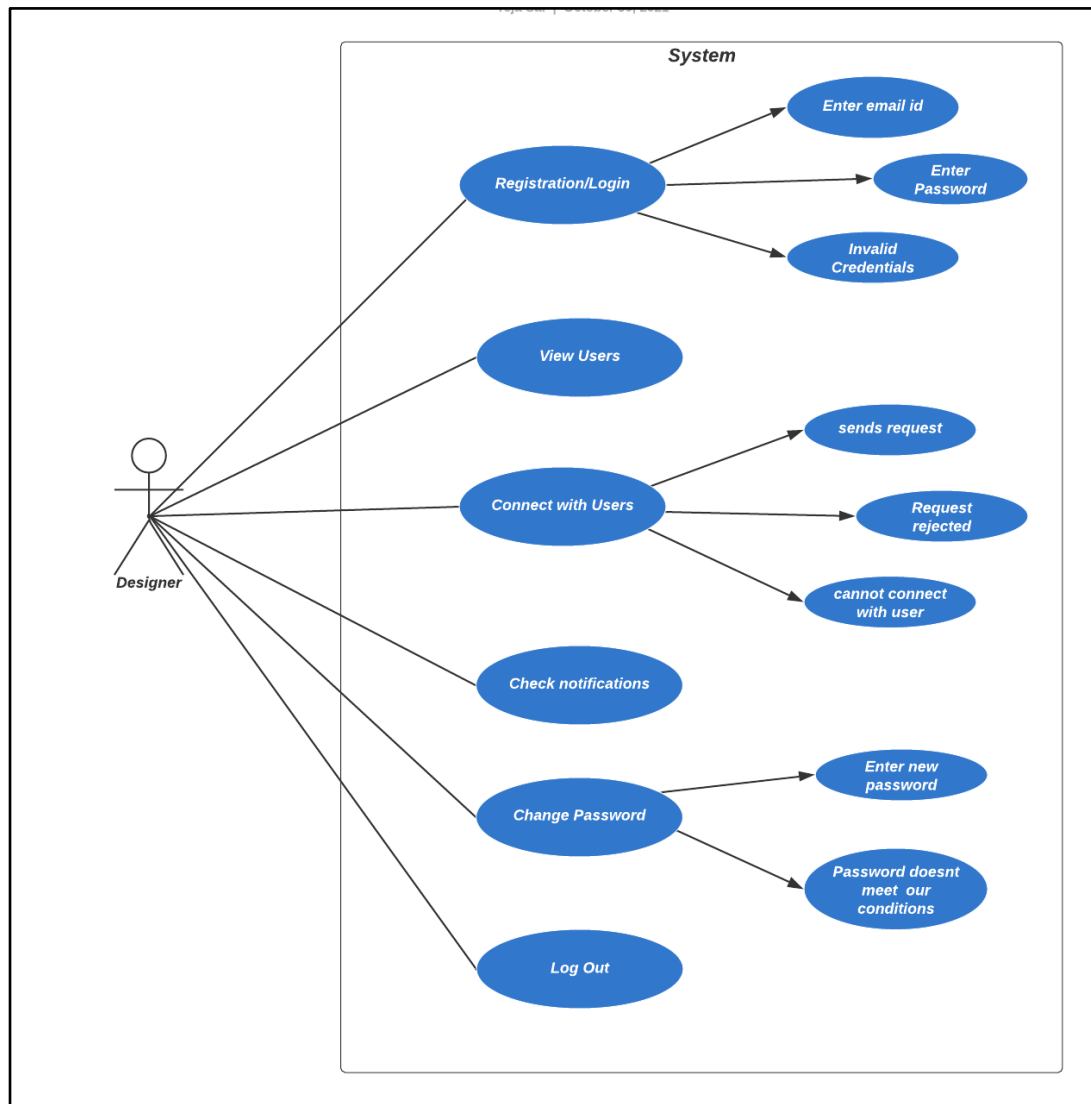


Fig. Error Case Diagram

Use Case Diagram for Users:

A use case diagram is used in UML to identify system functionality during the analysis phase. It defines how users and external devices interact with the system under development. It does not go into great detail. Instead, it focuses on the connections between use cases, actors, and systems. A use case diagram usually comprises four components. Actors, systems, use cases, and relationships are all examples. The actors are anyone or anything which interacts with the system.

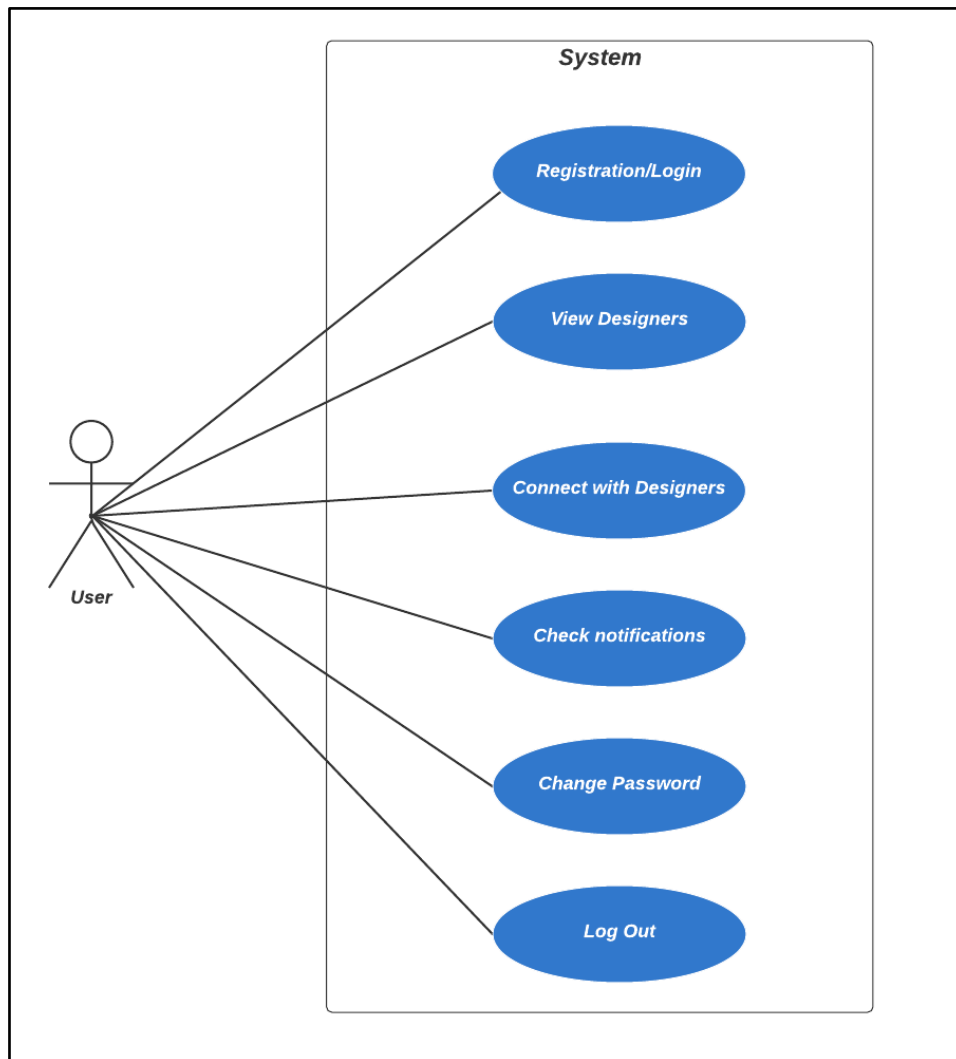


Fig. Use Case Diagram

Description: In the above use case diagram, the actors in the diagram are User who has the ability to see designers, read notifications, connect with people, and change their password.

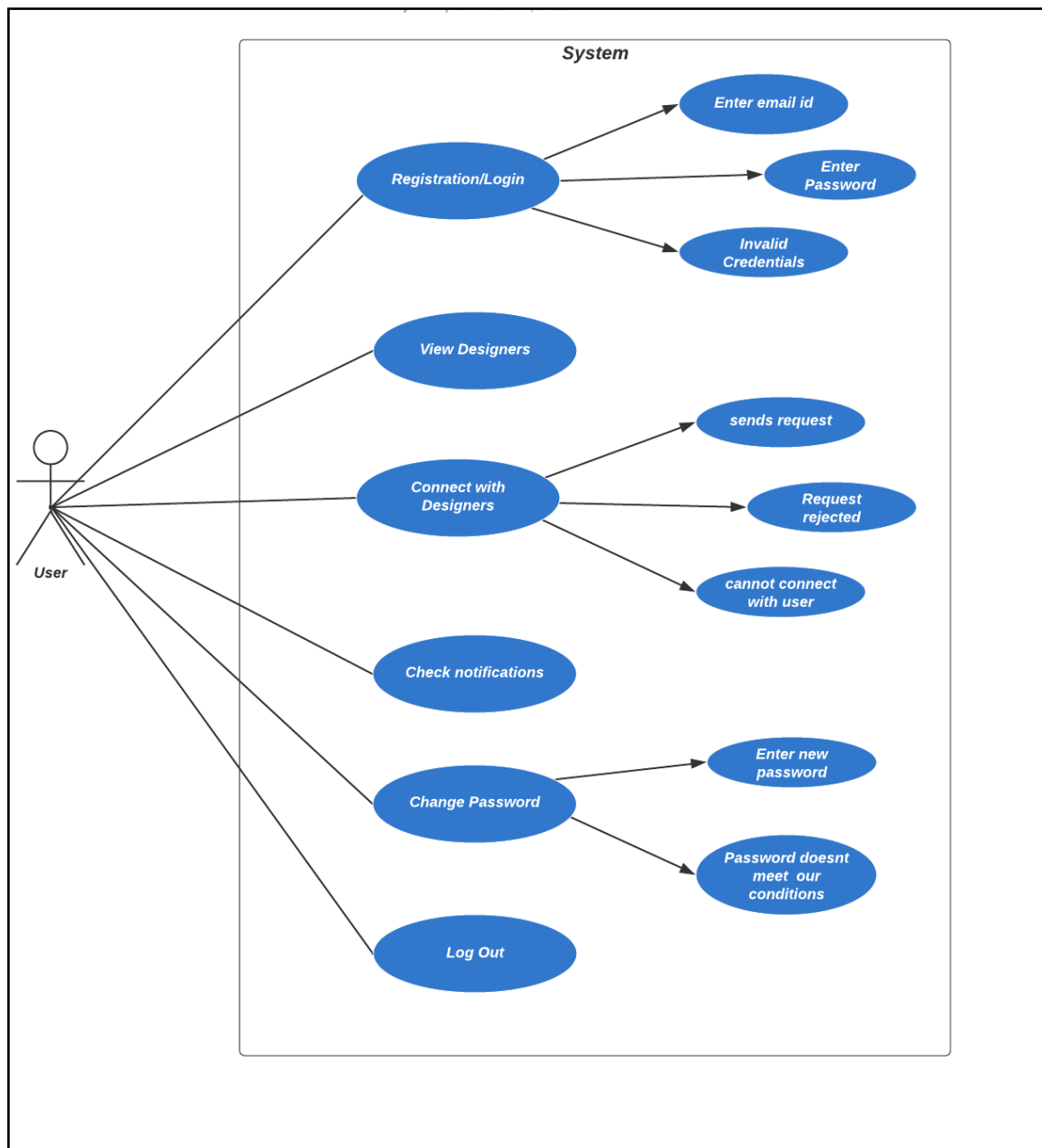


Fig. Use Case Diagram (Error case)

c. Test Cases (unit tests + system tests)

Testing: Testing is a process of determining whether the actual software product meets the expected requirements and is defect-free. It entails evaluating properties of interest by executing software/system components using human or automated techniques. Compared to actual requirements, software testing is used to find mistakes, gaps, and missing requirements. Program testing is critical because it allows any defects or errors in the software to be found and fixed before the final product is delivered. Software that has been thoroughly tested ensures dependability, security, and good performance, saving time and money.

Test Case 1 admin:

Requirement_id	Ticket_id	Requirement_description	Expected Output	Actual Output	Requirement_ticket_status
1	11	Admin tries to log in to the dashboard	Logged in successfully	Improper credentials and gives an error message user Not found	Failure
1	12	Admin login to the dashboard with proper Credentials	logged in successfully	Logged in to dashboard successfully	Success
1	13	After successful login Home Page will be displayed	admin logs in to the dashboard, the home page appears.	Home page displayed successfully	Success
1	14	Admin tries to view Users	User profiles will be displayed	User profile displayed successfully	Success

Description: The authentication of the admin is tested in the preceding test case, and only the authorized user with appropriate credentials is allowed to enter the dashboard.

Test Case 2 admin:

Requirement_id	Ticket_id	Requirement_description	Expected Output	Actual Output	Requirement_ticket_status
2	21	Admin tries to access plan to check plan details	List of plans with validity and price are displayed	Displays list of plans with its details	Success
2	22	Admin tries to update the plans	The plan will be updated	Invalid operation admin doesn't have access to update plan	Failure
2	23	Admin tries to add plans	Request to add plans is accepted and admin adds plan	Admin adds plan to users	Success

Description: The admin's access to manage plans is tested in the preceding test case, and only the limited operations are enabled for the Admin to protect the integrity of data.

Test Case 3 admin:

Requirement_id	Ticket_id	Requirement_description	Expected Output	Actual Output	Requirement_ticket_status
3	31	Admin views subscriptions	Subscription info is displayed on the dashboard	The dashboard shows subscriptions data	Success

3	32	Admin tries to update subscription	Subscription will be updated	Invalid operation admin cannot change subscription	Failure
---	----	------------------------------------	------------------------------	--	---------

Description: The admin's access to view and manage subscriptions is tested in the preceding test case, and only the limited operations are enabled for the Admin to protect the integrity of users data.

Test Case 1 designer:

Requirement_id	Ticket_id	Requirement_description	Expected Output	Actual Output	Requirement_ticket_status
1	11	The designer tries to log in to the dashboard	Logged in successfully	Improper credentials and gives an error message user Not found	Failure
1	12	Designer login to the dashboard with proper Credentials	logged in successfully	Logged in to dashboard successfully	Success
1	13	After successful login Home Page will be displayed	The designer logs in to the dashboard, the home page appears.	Home page displayed successfully	Success

1	14	The designer tries to view Users	User profiles will be displayed	User profile displayed successfully	Success
---	----	----------------------------------	---------------------------------	-------------------------------------	---------

Description: The authentication of the Designer is tested in the preceding test case, and only the authorized user with appropriate credentials is allowed to enter the dashboard.

Test Case 2 Designer:

Requirement_id	Ticket_id	Requirement_description	Expected Output	Actual Output	Requirement_ticket_status
2	21	The designer tries to access Notifications to check connection requests	A list of notifications are displayed	Displays Notification data on Dashboard	Success
2	22	The designer tries to accept the connection requests	Connect request accepted	Accepts requests and is allowed to chat with users	Success
2	23	The designer tries to add plans	Request to add plans is accepted and the Designer adds a plan	Invalid operation as a designer cannot add a plan	Failure
2	24	The Designer accepts the request and chats with users	The request gets accepted and user chat enabled	The dashboard opens a chat server and allows interaction with the user	Success

2	25	The designer tries to view messages and reply with multiple users	The designer can view and send replies to users	Messages are displayed and allowed to write a response and send to users	Failure
---	----	---	---	--	---------

Description: The Designer's access to view notifications and chat with users is tested in the preceding test case, and only the limited operations are enabled for the Designer to protect the integrity of data.

Test Case 3 Designer:

Requirement_id	Ticket_id	Requirement_description	Expected Output	Actual Output	Requirement_ticket_status
3	31	The Designer accepts the request and chats with users	The request gets accepted and user chat enabled	The dashboard opens a chat server and allows interaction with the user	Success
3	32	The designer tries to view messages and reply with multiple users	The designer can view and send replies to users	Messages are displayed and allowed to write a response and send to users	Failure

Description: The Designer's access to view and Chat with users is tested in the preceding test case, and only the limited operations have enabled the Designer to protect the integrity of users data.

Test Case 1 user:

Requirement_id	Ticket_id	Requirement_description	Expected Output	Actual Output	Requirement_ticket_status
----------------	-----------	-------------------------	-----------------	---------------	---------------------------

1	11	The user tries to log in to the dashboard	Logged in successfully	Improper credentials and gives an error message user Not found	Failure
1	12	User login to the dashboard with proper Credentials	logged in successfully	Logged in to dashboard successfully	Success
1	13	After successful login Home Page will be displayed	The user logs in to the dashboard, the home page appears.	Home page displayed successfully	Success
1	14	The user tries to view Designers	Designer profiles will be displayed	Designer profile displayed successfully	Success

Description: The authentication of the User is tested in the preceding test case, and only the authorized user with appropriate credentials is allowed to enter the dashboard.

Test Case 2 user:

Requirement_id	Ticket_id	Requirement_description	Expected Output	Actual Output	Requirement_ticket_status
2	21	The User tries to access Notifications to check connection requests	A list of notifications are displayed	Displays Notification data on Dashboard	Success

2	22	The User tries to accept the connection requests	Connect request accepted	Accepts requests and is allowed to chat with Designers	Success
2	23	The User tries to add plans	Request to add plans is accepted and the User adds a plan	Invalid operation as a user cannot add a plan	Failure

Description: The User's access to view notifications and chat with users is tested in the preceding test case, and only the limited operations are enabled for the user to protect the integrity of data.

Test Case 3 user:

Requirement_id	Ticket_id	Requirement_description	Expected Output	Actual Output	Requirement_ticket_status
3	31	The User accepts the request and chats with Designers	The request gets accepted and Designer chat enabled	The dashboard opens a chat server and allows interaction with the Designer	Success
3	32	The user tries to view messages and reply with multiple users	The user can view and send replies to users	Messages are displayed and allowed to write a response and send to Designers	Failure

Description: The User's access to view and Chat with users is tested in the preceding test case, and only the limited operations have enabled the user to protect the integrity of users data.

- d. A user manual that tells us how to use your program. This is meant for the end-user of the software. You may include screen shots, where appropriate.

User Manual for admin:

Step 1: To use our application, users must have access to the internet and a browser such as Chrome or Firefox. Admins can access the Admin Dashboard by visiting the login page and entering credentials (Mail Id and Password) that have been stored in our database.

Step 2: Admin should be able to see a home page in their dashboard if the credentials entered are correct to check the number of active users, active subscriptions, and available plans.

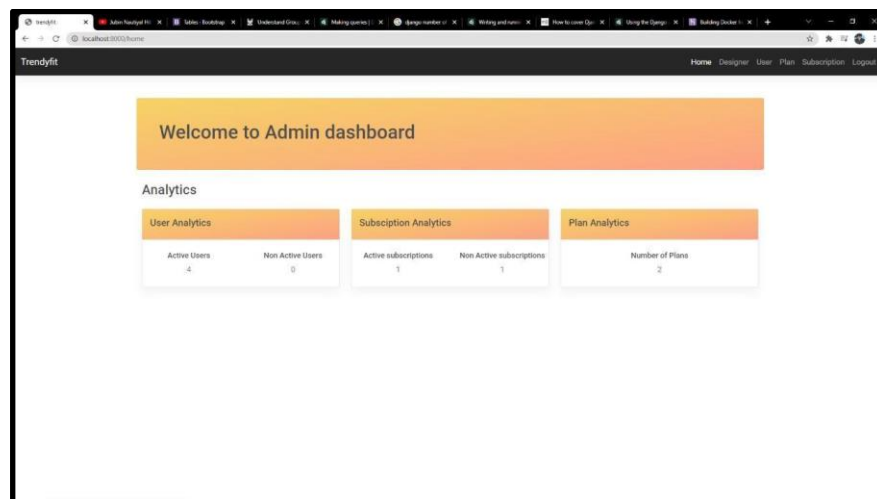


Fig.1.1. Home Page

Step 3: There are several options for the user to explore in our application, including Designer, User, Plan, and Subscription, which can be found in the right-hand navigation bar.

Step 4: Select Designer to navigate to a page where you can see a list of all the Designers who have registered with the application and their information such as the Designer's email address, registration date, and subscription details, plan name, days left, and plan status.

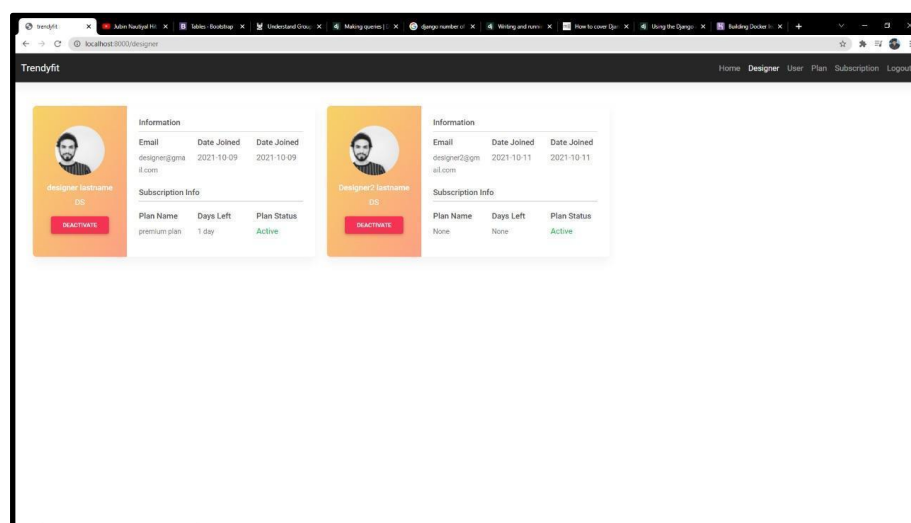


Fig 1.2. Designer Page

Step 5: Click on User to see the user's information, such as the email address they registered with, the date they joined, and other details such as their last login and status.

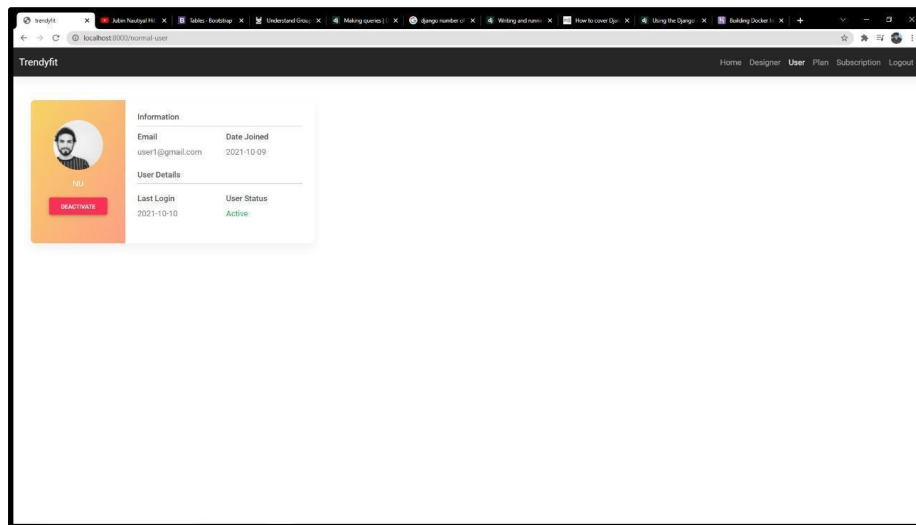


Fig 1.3. User Page

Step 6: Click Plan to view the available plans and their details, such as plan type, plan id, plan validity, and plan pricing.

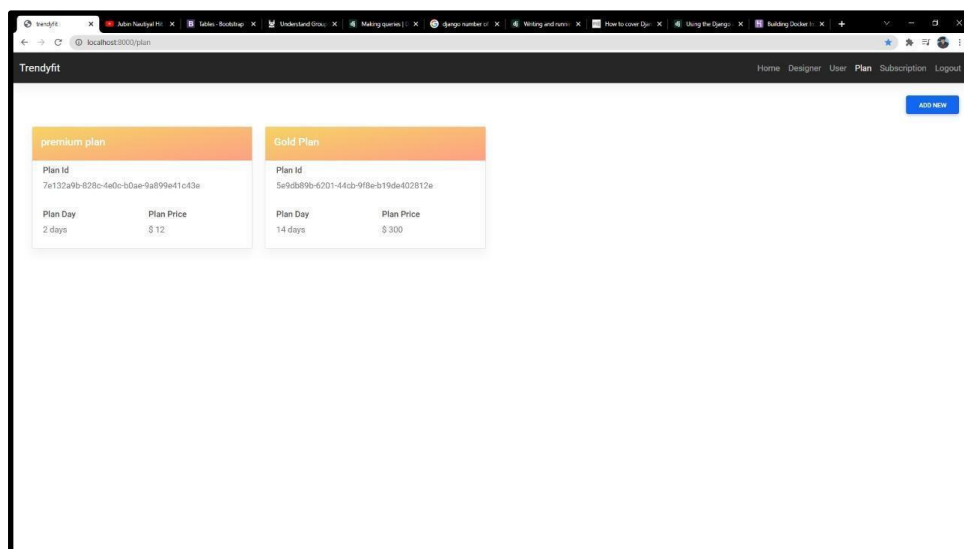


Fig 1.4. Plan Page

Step 7: Select Subscription from the Navigation Bar to access subscription details such as UserId, plan id, plan name, Validity; Plan activated date and termination date, and Payment method.

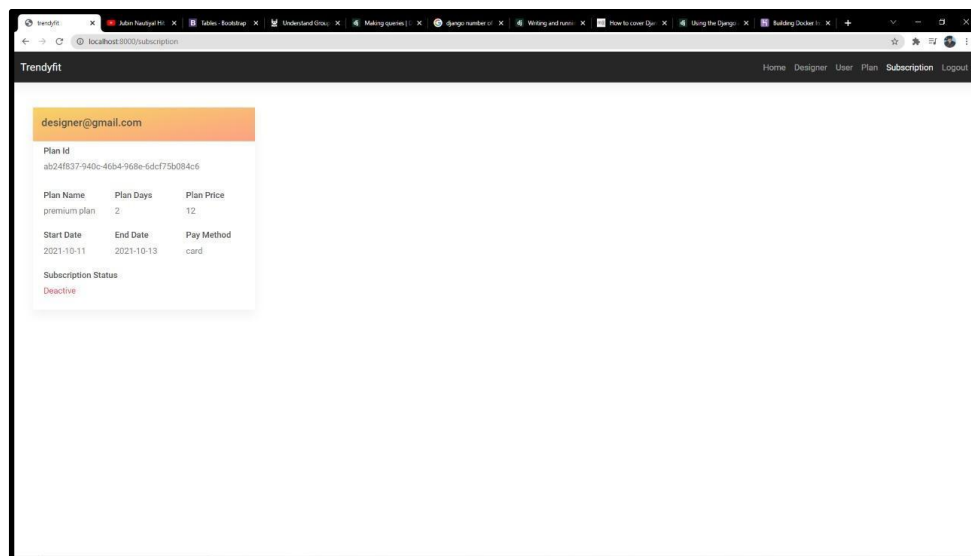


Fig 1.5. Subscription Page

User Manual for Designer:

Step 1: To use our application, users must have access to the internet and a browser such as Chrome or Firefox. Designers can access the Designer Dashboard by visiting the login page and entering credentials (Mail Id and Password) that have been stored in our database.

A screenshot of a web browser displaying the 'Sign Up' form of 'Trendyfit'. The form is titled 'Sign Up' and contains fields for 'First name', 'Last name', 'Email', 'User type' (a dropdown menu with 'Designer' selected), 'Password', and 'Password confirmation'. Below the form is an orange 'create account' button and a link that says 'Have a account? Login'.

Fig.1.1 Sign Up Form

Step 2: Designer should be able to see a home page in their dashboard if the credentials entered are correct to check the available plans.

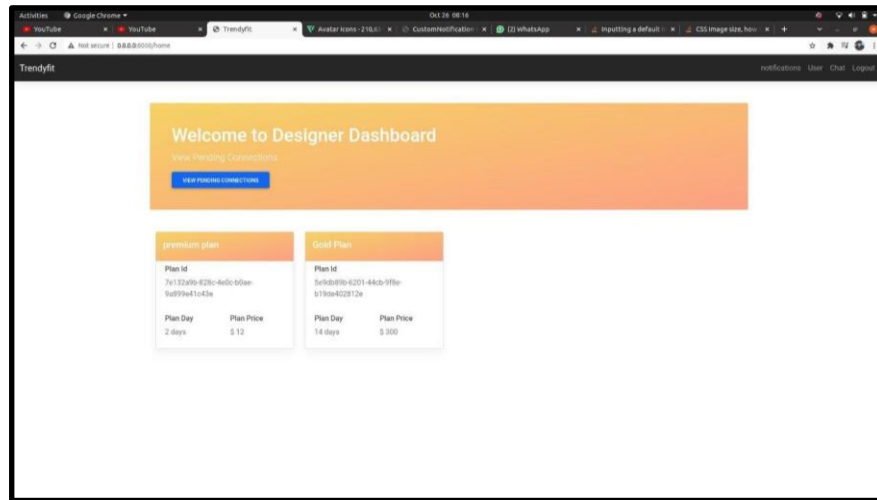


Fig.1.2. Home Page

Step 3: There are several options for the user to explore in our application, including Designer profile, View User, and Notifications, which can be found in the right-hand navigation bar.

Step 4: Select Notifications to navigate to a page where you can see a list of all the notifications, including connection requests from users, status updates of requests, and messages from the chat.

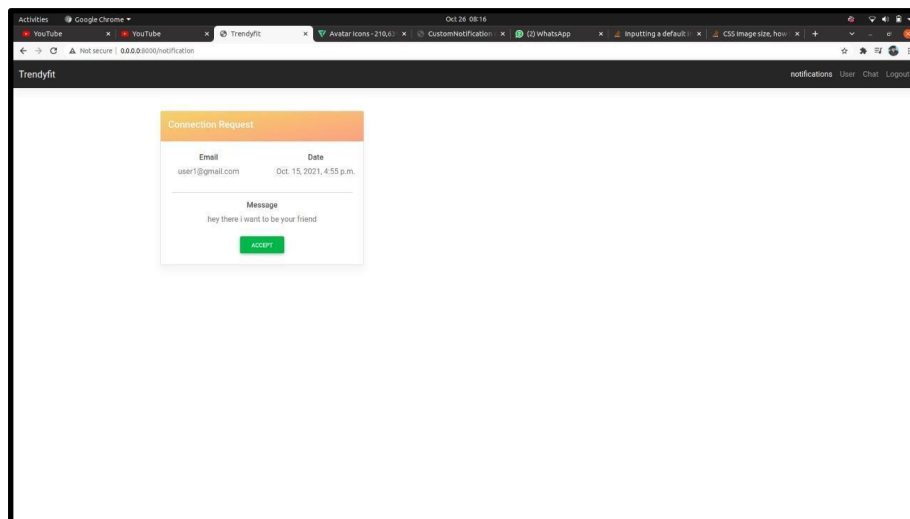


Fig 1.3. Notification Page

Step 5: Click on User to see the user's information, such as the email address they registered with, the date they joined, and other details such as their last login and status.

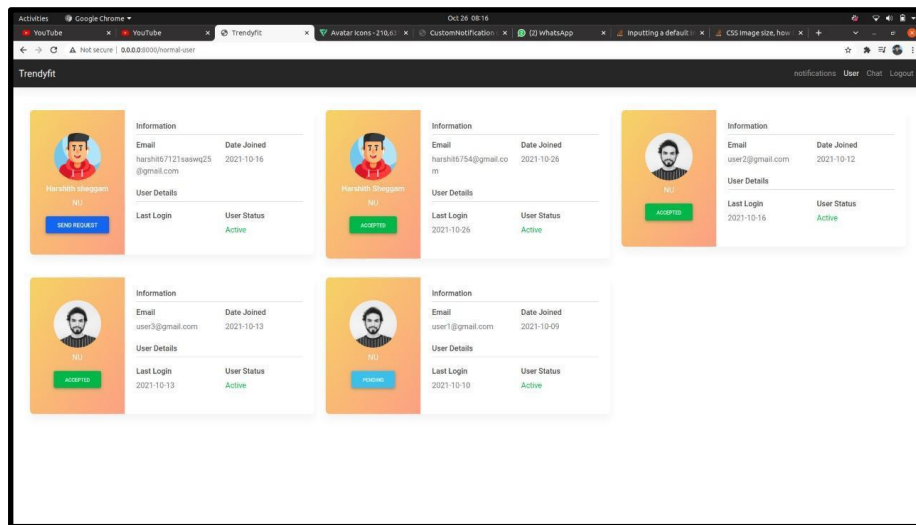


Fig 1.4. User Page

Step 6: To view the chat page, click on chat. From there, the Designer can choose to chat with users whose requests have been accepted.

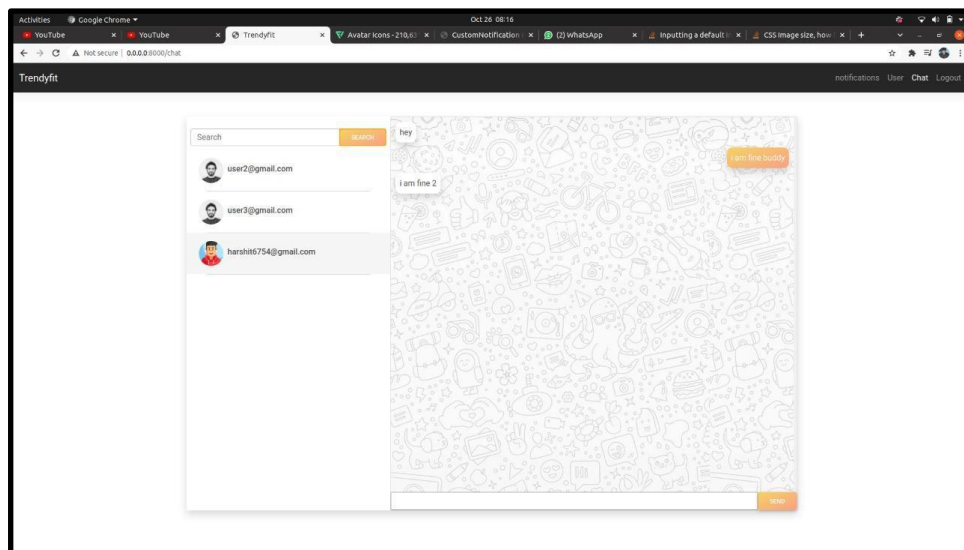


Fig 1.5. Chat Page

User Manual for users:

Step 1: To use our application, users must have access to the internet and a browser such as Chrome or Firefox. users can access the user Dashboard by visiting the login page and entering credentials (Mail Id and Password) that have been stored in our database.

The screenshot shows a web browser window with the Trendifyt Sign Up form. The form is centered on a white background with a dark header. It includes input fields for First name, Last name, Email, Password, and Password confirmation. A dropdown menu for User type is set to 'Designer'. A blue 'create account' button is at the bottom of the form, with a 'Have an account? Login' link below it.

Field	Type
First name	Text input
Last name	Text input
Email	Text input
User type	Dropdown menu (Designer)
Password	Text input
Password confirmation	Text input
create account	Submit button
Have an account? Login	Link

Fig.1.1 Sign Up Form

Step 2: User should be able to see a home page in their dashboard if the credentials entered are correct to check the available plans.

Step 3: There are several options for the user to explore in our application, including User profile, View Designer, and Notifications, which can be found in the right-hand navigation bar.

Step 4: Select Notifications to navigate to a page where you can see a list of all the notifications, including connection requests from Designers, status updates of requests, and messages from the chat.

The screenshot shows the Trendifyt Notification page. The header has a dark background with the Trendifyt logo and navigation links: notifications, Designer, Chat, and Logout. The main content area displays three notification cards. Each card has an orange header 'Connection Request', a table with Email and Date, a Message section, and a blue 'CONNECTION ACCEPTED' button.

Card	Email	Date	Message
1	designer@gmail.com	Oct. 12, 2021, 3:59 a.m.	hey there i want to be your friend
2	user2@gmail.com	Nov. 7, 2021, 7:17 p.m.	hey i want to be your friend
3	user2@gmail.com	Nov. 9, 2021, 9:36 a.m.	hey there i want to be your friend

Fig 1.2. Notification Page

Step 5: Click on Designer to see the Designer's information, such as the email address they registered with, the date they joined, and other details such as their last login and status.

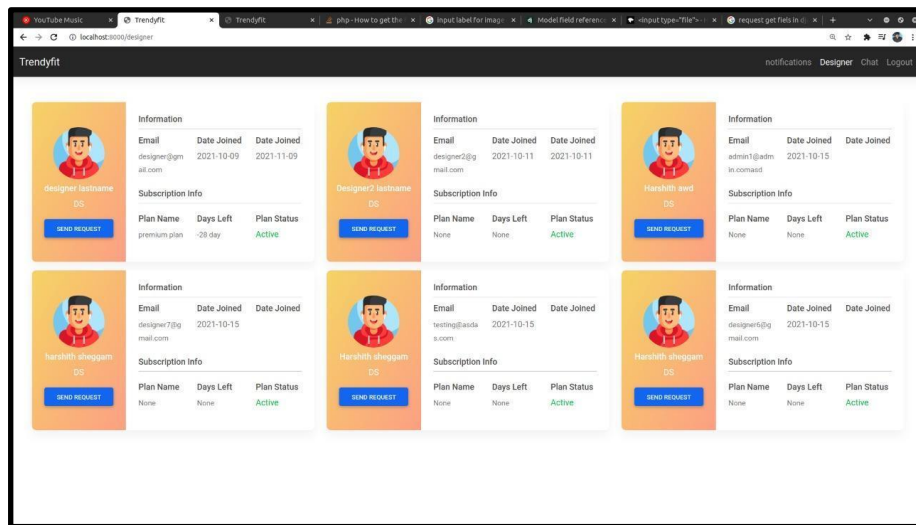


Fig 1.3. User Page

Step 6: To view the chat page, click on chat. From there, the User can choose to chat with designers whose requests have been accepted.

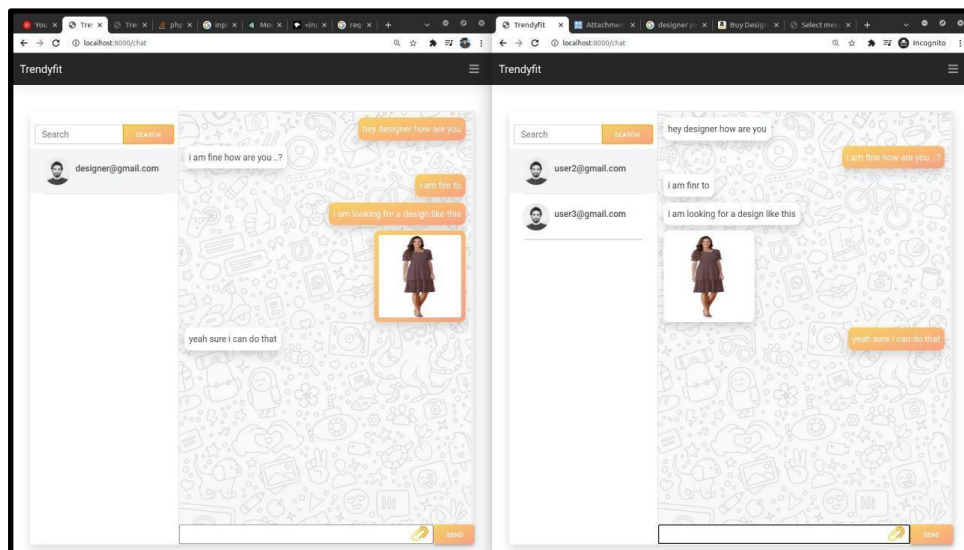


Fig 1.4. Chat Page

- e. **Clear instructions on how to compile/run both your program and your test cases (the program must compile/run).**

Instructions:

The following are the 2 methods to setup our project

- **using local run command**
- **using docker**

Note: The following commands are for linux only

Local Setup

The first thing to do is to clone the repository:

```
$ git clone https://github.com/SravanthiSilla/Mighty8.git
```

```
$ cd terndyfit
```

Create a virtual environment to install dependencies in and activate it:

```
$ python -m venv env
```

```
$ source env/bin/activate
```

Then install the dependencies:

```
(env)$ pip install -r requirements.txt
```

Note the (env) in front of the prompt. This indicates that this terminal session operates in a virtual environment set up by venv.

Once pip has finished downloading the dependencies:

```
(env)$ cd trendyfit
```

```
(env)$ python manage.py makemigrations
```

```
(env)$ python manage.py migrate
```

once you have migrated you need to

```
(env)$ cd trendyfit
```

```
(env)$ python manage.py runserver
```

or

```
(env)./manage.py runserver
```

And navigate to <http://127.0.0.1:8000/>.

Docker Setup

To execute the project in docker follow below commands. Make sure you have docker installed in your system or else please follow this link [click here](#)

```
$ cd terndyfit
```

```
$ docker-compose -f compose.yml up
```

- f. A section that describes which features you have successfully implemented and any limitations (i.e., any known problems or issues with your program) such as features that you were unable to complete on time including feedback received during peer review sessions. It should include discussions of your plans for the next phase of your project. These are the next features and/or improvements you plan to make to your program (assume that you will continue working on this project next semester).

We have implemented the below features successfully in Admin, Designer & user:

Admin:

- Login: Admin is provided with a facility to log into the application using the credentials that are registered with our application.
- Accept Designer: Admin is given access to accept requests from the newly registered designers with the application.
- View Users: Admin can view registered users where admin will verify details like their id, status and subscription plan.
- Add plans: Admin can add plans for the designer like a silver plan for three months subscriptions, gold for six months and diamond for an annual subscription.
- Manage Subscriptions: Admin has the feature to manage subscriptions on request from the registered users.
- View Designer: Admin will be able to view details like status, plan and the id of the registered Designers.

Designer:

- Registration: The designer will register using their details like email id, password, and plan for the subscription.
- Login: The designer can log into the application using the credentials obtained from registration.
- View Users: The designer will also be able to view users for helping them with their knowledge in designing.
- Chat: The designers are provided with a feature to chat with the users to interact with them, which helps users choose their designer
- Change Password: The designers will have an option to change their passwords in case of an emergency.

User:

- Chat: The users are allowed to chat with the designers to choose the designer of their choice.
- View Stylish: The users can view designers and check their profiles to know their past work and interact with them.
- Feedback: After a chat with the designer, the users can give feedback based on the conversation.
- My Details: The users can view the details given during registration, like their email or chosen plan.
- Change Password: The users will have an option to change their passwords in case of an emergency.

Feedback received during peer review sessions:

Our partner team suggested changing the UI interface and chat module to work. We have implemented that one.

Adding scrolling option to the Admin Dashboard: They suggested we add a scrolling option in the admin dashboard in the user screen. We implemented pagination but to get the perfect user experience they suggested a scrolling option. We have accepted the suggestion provided by the partner team.

UI change: Partner requested to make a few changes to UI to have a better experience.

Actions are taken: We have added a scrolling option to Admin Dashboard in the user screen and removed pagination. Also made changes to UI based on the requirements like added designer image.

Features in next phase of the Project:

- We will be adding video sharing option in chat.
 - User & designer can communication through Call Option & they can send voice messages as well.
 - Also, user can provide star ratings to designer.
- g. A brief reflection on what has been accomplished, what went well and could be improved.**

Brief Reflection:

Project phase one development has successfully completed we have implemented admin dashboard successfully feature that has been covered in the phase one development are listed below

- Admin login view
- Admin main dashboard view
- Admin designer view
- Admin user view
- Admin Plan view
- Admin Add plan view

- Admin Subscription View
- Admin logout

Project phase two development has successfully completed we have implemented Designer dashboard successfully features that have been covered in this phase two development are listed below

- Designer login view
- Designer main dashboard view
- Designer user view
- Designer Notification view
- Designer Chat view
- Designer Subscription View
- Designer logout

Project phase two development has successfully completed we have implemented Designer dashboard successfully features that have been covered in this phase two development are listed below

- User login view
- User main dashboard view
- User Designer view
- User Notification view
- User Chat view
- User Subscription View
- User logout

User:

As part of phase three, the above features are implemented in the User dashboard. Our goal was to complete all of the User's flows, which we did effectively. Notifications, connections, and the chat feature are all significant aspects of phase two.

Designer:

As part of phase two, the above features are implemented in the Designer dashboard. Our goal was to complete all of the Designer's flows, which we did effectively. Notifications, connections, and the chat feature are all significant aspects of phase two.

Admin:

The above features are implemented in the admin dashboard flow as part of the phase our target was to finish all the flow of the admin and there were successfully implemented

Improvements:

- A few Improvements can be done in the admin dashboard such as adding pagination which will not make perfect user experience, we will shift that to perfect scrolling in coming phases where user will not to click more on our dashboard that will improve our user experience.

- Push notifications when a user/admin adds to our portal a push notification will be send to the admin so that he can keep a track of the new user at real time.
- Filters can also be added to get the user/designer on search for an easy view of data.

h. Member contribution table (should describe who wrote what components or classes of the system and what parts of the report). Add more rows as needed.

Member name	Contribution description	Overall Contribution (%)	Note (if applicable)
Sravanthi	Changed designer and chat templates.	13	
Manish	Worked on adding migrations.	13	
Sharath	Worked on the User JS module.	12	
Teja	Worked on PPT documentation.	13	
Harsha	Added consumer image.	12	
Yeswanth	Worked on the Chat JS module.	12	
Manoharachary	Worked on the Designer JS module.	13	
Sai Praveen	Added URLs for the pages.	12	