

PROJECT REPORT:

**FACEBOOK USER INTERFACE
CLONE**

By:
MOHAMMED RAYAN HUSSAIN

ID:MST04-0046

Abstract

Creating a Facebook UI clone using HTML, CSS, and JavaScript is an engaging project that allows developers to explore the design principles and functionalities of a modern, interactive social media platform. The clone replicates essential features of Facebook's user interface, such as a dynamic news feed, user profile sections, a navigation bar, and interactive elements like like buttons, comment sections, and post inputs. By recreating these features, the project demonstrates the effective use of front-end technologies to deliver an intuitive and visually appealing experience.

HTML is employed as the foundational structure to organize the UI components, while CSS is used extensively to style elements, ensuring a responsive and aesthetically consistent design. Advanced CSS techniques, such as flexbox and grid, are utilized to mimic Facebook's complex layouts, while media queries ensure adaptability across various devices and screen sizes. JavaScript enhances interactivity by enabling functionalities such as dynamic updates to the news feed, modal popups for creating posts, and real-time updates for user actions, making the clone feel more dynamic and responsive.

This project is not only a valuable learning exercise for front-end development but also serves as a practical demonstration of how individual technologies come together to create a cohesive user experience. It highlights the importance of clean code structure, responsive design principles, and interactive functionalities in building user-centered web applications. The Facebook UI clone offers developers an opportunity to sharpen their skills and gain insights into the design and functionality of large-scale social media platforms.

Table of Contents:

1.Introduction

2.Project Overview

3.Technologies Used

4.System Architecture

5.Implementation Details

6.Challenges Faced

7.Future Enhancements

8.Conclusion

9.References

1. Introduction:

The Facebook UI clone is a project aimed at replicating the visual design and basic functionality of one of the world's most popular social media platforms. By building this clone using HTML, CSS, and JavaScript, developers can gain hands-on experience in creating a polished, responsive, and interactive user interface. The project focuses on implementing core features such as a navigation bar, a news feed with posts, user profiles, and interactive elements like comments, likes, and shares. This provides an opportunity to understand the design intricacies and functional components of a complex web application.

HTML serves as the backbone of the project, providing a well-structured layout for various UI elements. CSS brings life to these structures by adding styles, colors, and responsive designs that adapt to different devices. The use of CSS techniques such as flexbox, grid, and media queries ensures the layout remains consistent and user-friendly. JavaScript further enhances the project by introducing dynamic functionality, such as toggling menus, submitting posts, and updating the UI in real-time. Together, these technologies create a cohesive and interactive web application that mirrors the user experience of the original Facebook platform.

This project is ideal for aspiring front-end developers who want to enhance their skills in web development. It offers a deep dive into designing and developing scalable UI systems and provides practical knowledge of combining HTML, CSS, and JavaScript to create modern web interfaces. Beyond technical skills, building this clone encourages developers to think critically about user experience and responsive design, making it a comprehensive learning experience in crafting engaging web applications.

2. Project Overview:

The Facebook UI clone project focuses on replicating the core features of Facebook's interface and functionality using HTML, CSS, and JavaScript. The development includes two main components: a login page and a home page. The login page features a fully functional form with validation, ensuring user inputs are correct before authentication. Upon successful login, an example authentication mechanism transitions the user to the home page, providing a seamless experience. The home page includes an interactive and visually appealing layout, designed to resemble Facebook's original interface.

The home page integrates multiple working features that enhance user interaction. A profile card displays user information prominently, and like buttons have been implemented to function just as they do on Facebook, offering a realistic interaction experience. Additionally, clicking on the Facebook icon refreshes the page, mimicking the behavior of the actual platform. A logout option is also functional, allowing users to securely exit the session and return to the login page. These features demonstrate a well-rounded understanding of both front-end development and user experience design.

Pages Implemented:

- **Login Page:** Features a functional login form with validation and example authentication to access the home page.
- **Home Page:** Displays an interactive interface with a working profile card, like buttons, and page refresh functionality on clicking the Facebook icon.
- **Logout Functionality:** Allows users to securely log out and return to the login page.

This project highlights the thoughtful integration of UI and functional components to create a user-centric application. By combining responsive design principles and JavaScript interactivity, the clone not only captures the aesthetic essence of Facebook but also mirrors some of its basic functionalities. The result is a streamlined application that serves as a practical showcase of web development skills, emphasizing the implementation of authentication, interactive elements, and user navigation within a cohesive interface.

3. Technologies Used:

1. HTML5

- Used for structuring the website's content and layout.
- Ensures semantic markup for better accessibility and SEO.

2. CSS3

- Applied for styling and designing the website, ensuring an aesthetically pleasing and consistent look.
- Techniques like Flexbox and Grid were used for responsive layouts.

3. JavaScript (ES6)

- Enables interactivity and dynamic features such as cart functionality and menu filtering.

4. Responsive Design Tools

- **Media Queries:** Ensures compatibility across various devices and screen sizes.

5. Version Control

- **Git & GitHub:** For version control and collaborative development.

6. Design Tools

- Tools like Figma or Adobe XD were used for prototyping and UI/UX design

4. System Architecture:

The system architecture of the Facebook UI clone is designed to mimic the basic functionalities of a social media platform with a focus solely on front-end implementation. Below is a structured overview of the architecture:

1. Presentation Layer (Frontend)

The presentation layer consists of HTML, CSS, and JavaScript, creating the user interface and handling interactions.

- **HTML:**

- Provides the structure for the login page and home page.
- Elements include forms (login), profile card, like buttons, and navigation elements (Facebook icon).

- **CSS:**

- Responsible for styling the interface to resemble Facebook's design.
- Includes responsive design for compatibility across devices.

- **JavaScript:**

Handles interactivity and dynamic behavior, such as:

- Form validation for login.
- Example authentication logic.
- Functionality for like buttons and profile interactions.
- Page refresh behavior on clicking the Facebook icon.
- Logout functionality to return to the login page.

2. Application Logic Layer

This layer manages the functional logic and connects the front-end behaviour:

- **Form Validation:**
 - Ensures proper user input before login authentication.
- **Authentication Simulation:**
 - Mimics user login and transitions to the home page upon successful authentication.
- **State Management:**
 - Tracks the logged-in state to enable logout functionality and redirection.

3. Flow Diagram

i. Login Page

- User enters credentials → JavaScript validates input → Redirects to Home Page if valid.

ii. Home Page

- Displays user profile card, news feed, and like buttons.
- User interactions update the interface dynamically (e.g., liking posts).
- Clicking the Facebook icon refreshes the page.

iii. Logout Process

- Logout button clears the logged-in state and redirects to the login page.

5. Implementation Details:

The implementation process of the Facebook UI clone involves breaking down the project into distinct stages, from setting up the project environment to implementing features and final testing. Below is a step-by-step guide to help you through the process:

1. Project Setup

- **Create Project Directory:**

Set up a folder for your project where you will store all the HTML, CSS, and JavaScript files.

- **Initialize Files:**

Create basic files:

- index.html (for the login page)
- home.html (for the home page)
- style.css (for styling the pages)
- script.js (for adding interactivity)

2. Structure the HTML Layout

Start by creating the structure for the login and home pages.

- **Login Page (index.html):**

- Create a simple form with fields for the username and password.
- Add validation attributes (e.g., required) to ensure basic input validation.
- Add a button to submit the form for authentication.

- **Home Page (home.html):**

- Create the layout for the home page, including the profile card, news feed, and like buttons.

- Include the Facebook logo icon and a logout button.

3. Style with CSS

Style both the login and home pages to resemble Facebook's interface.

- **Login Page Styling:**

- Use CSS to center the login form and add styles to input fields and buttons.
- Apply a simple, clean design for the login screen (similar to Facebook's login page).

- **Home Page Styling:**

- Use flexbox or grid for arranging the profile card, posts, and navigation elements.
- Style the news feed, profile card, like buttons, and logout button.
- Add responsiveness to ensure the layout adapts well to different screen sizes (desktop, tablet, mobile).

4. Implement Interactivity with JavaScript

Add the functionality to the UI components using JavaScript.

- **Login Page:**

- **Form Validation:**

- Implement JavaScript to validate the input fields for username and password. Ensure that empty fields are not submitted.

- **Authentication Simulation:**

- For now, use example credentials (e.g., username = "user" and password = "password"). If the user inputs the correct credentials, redirect them to the home page.
- If the credentials are incorrect, display an error message.

- **Home Page:**
 - **Like Button Functionality:**
 - Use JavaScript to make the like buttons interactive. When clicked, the like button should toggle between "liked" and "unliked".
 - **Profile Card:**
 - Populate the profile card with example user information (name, profile picture).
 - **Page Refresh on Icon Click:**
 - Add a clickable Facebook icon that refreshes the page when clicked.
 - **Logout Functionality:**
 - Implement a logout button that clears the login state and redirects the user back to the login page.

5. Testing the Features

After implementing all the features, conduct testing to ensure the functionality works as expected:

- **Test the login process:**
 - Check if the form validation works correctly.
 - Ensure the authentication simulation redirects users to the home page.
- **Test the home page interactions:**
 - Verify that the profile card displays the correct information.
 - Test if the like buttons toggle correctly when clicked.
 - Ensure the logout button logs the user out and returns to the login page.
 - Check the refresh functionality on the Facebook icon click.

6. Final Refinements

- **Refactor the Code:**
 - Clean up the code, removing any unnecessary lines or comments.
 - Ensure all files are organized and named appropriately.
- **Optimize for Mobile:**
 - Make sure the layout is responsive and works well on different screen sizes.
- **Cross-Browser Testing:**
 - Test the project in multiple browsers (Chrome, Firefox, Edge, etc.) to ensure compatibility.

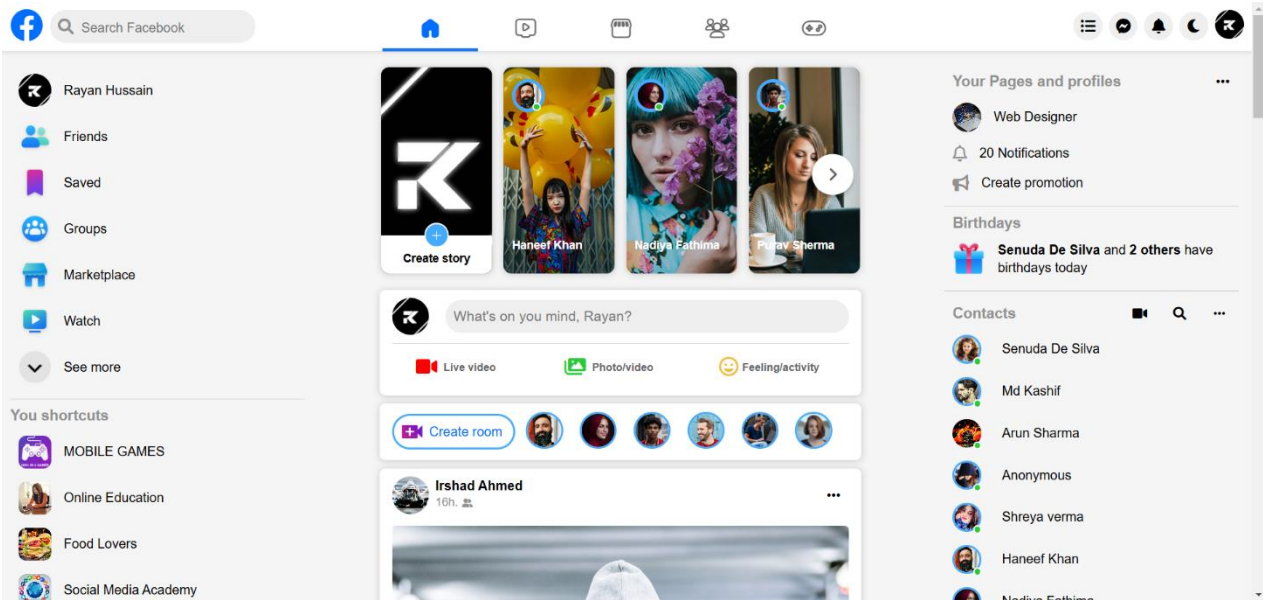
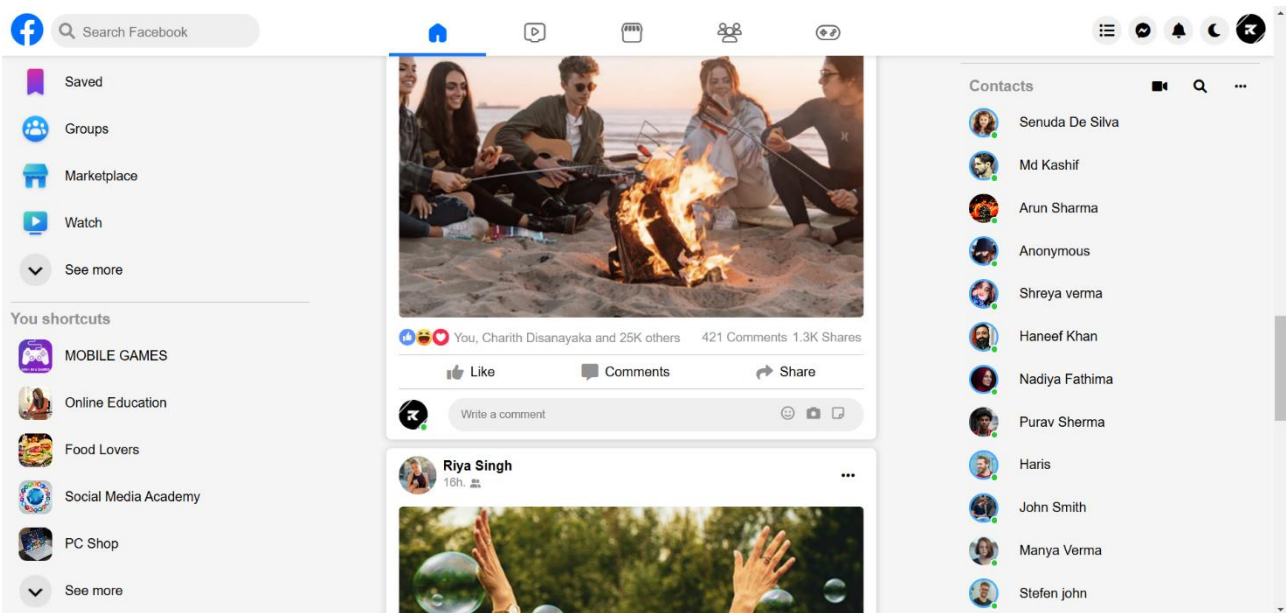


Fig 1: Home page



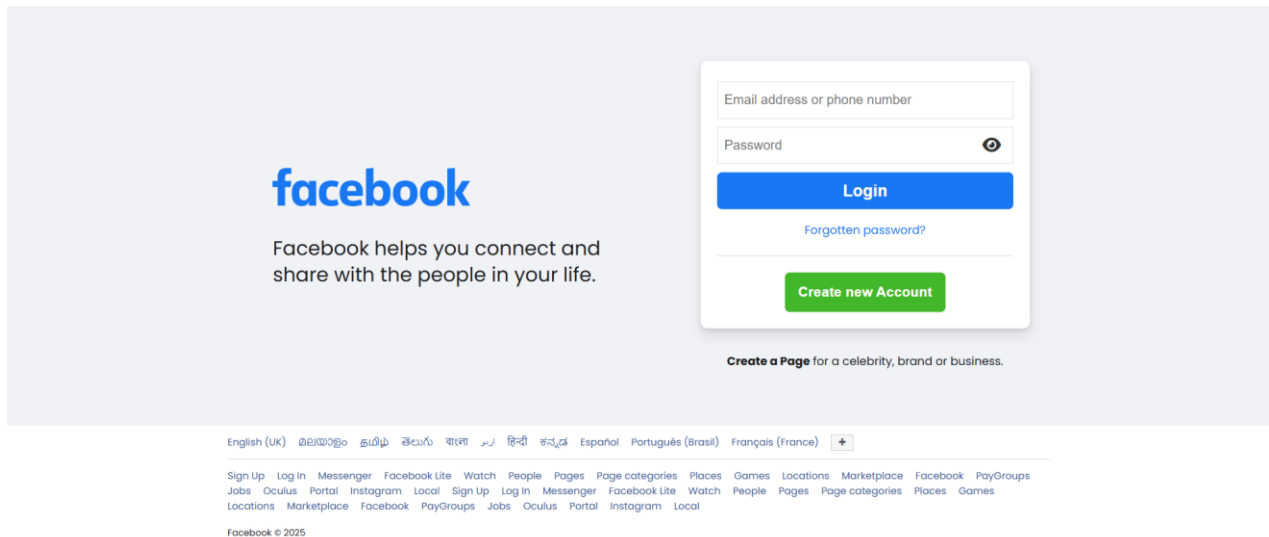
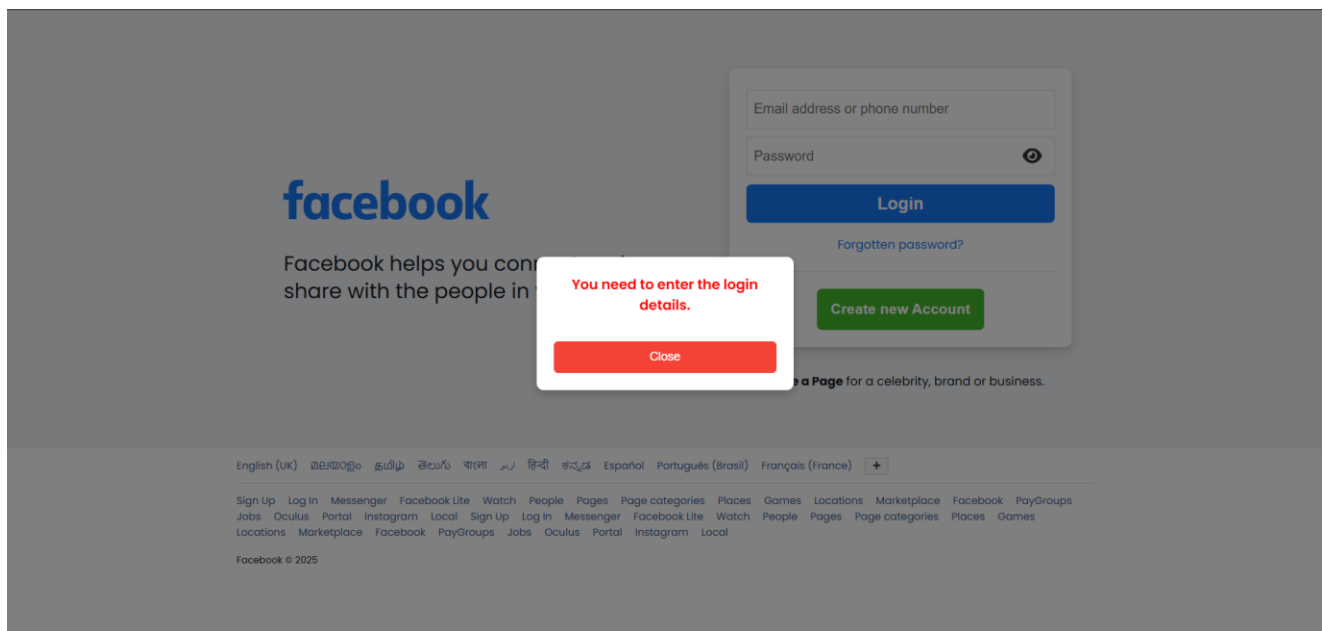


Fig 2. Login page



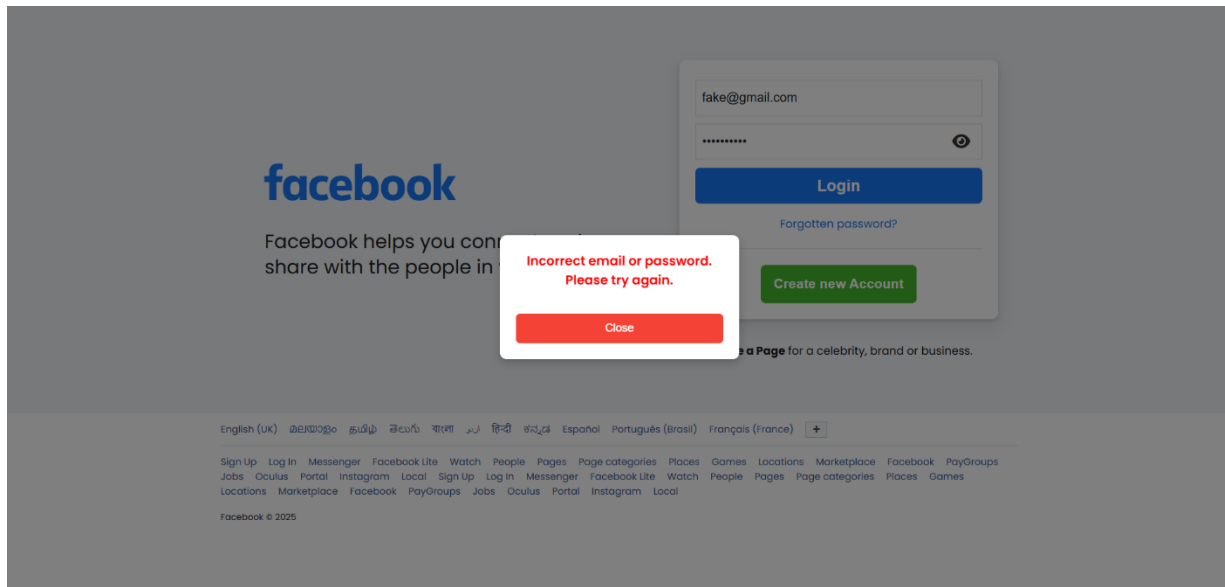


Fig 3. Form Validation of Login Page

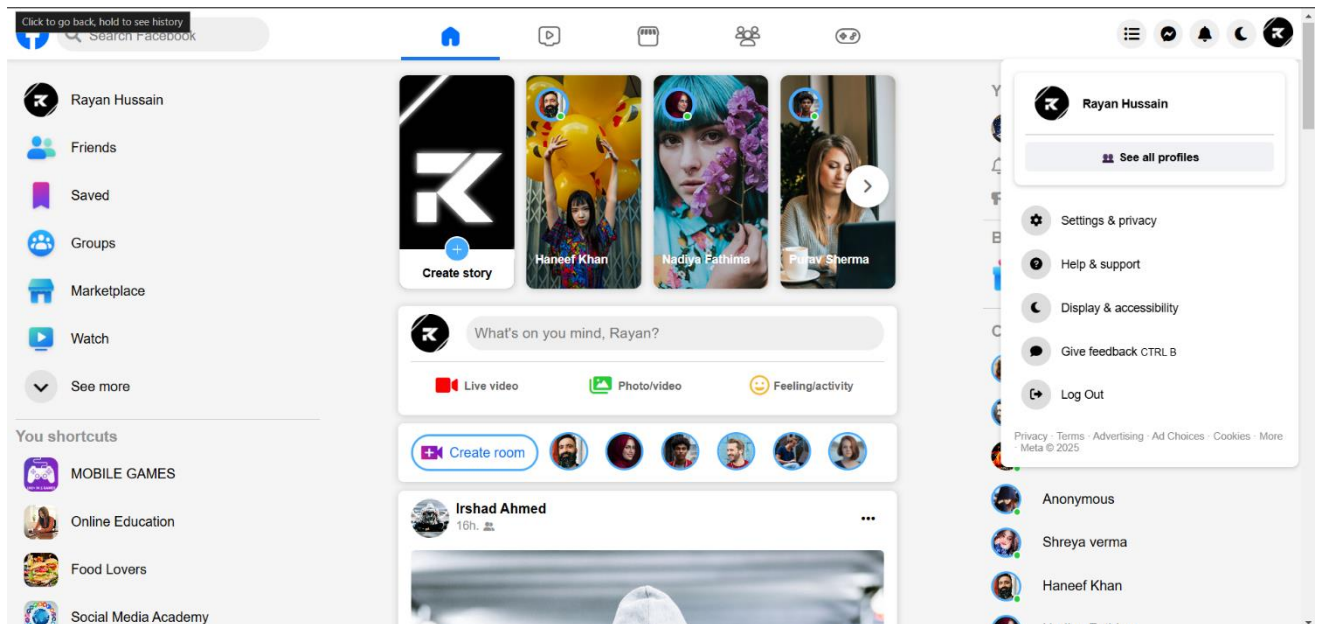


Fig 4. Profile card Overlay

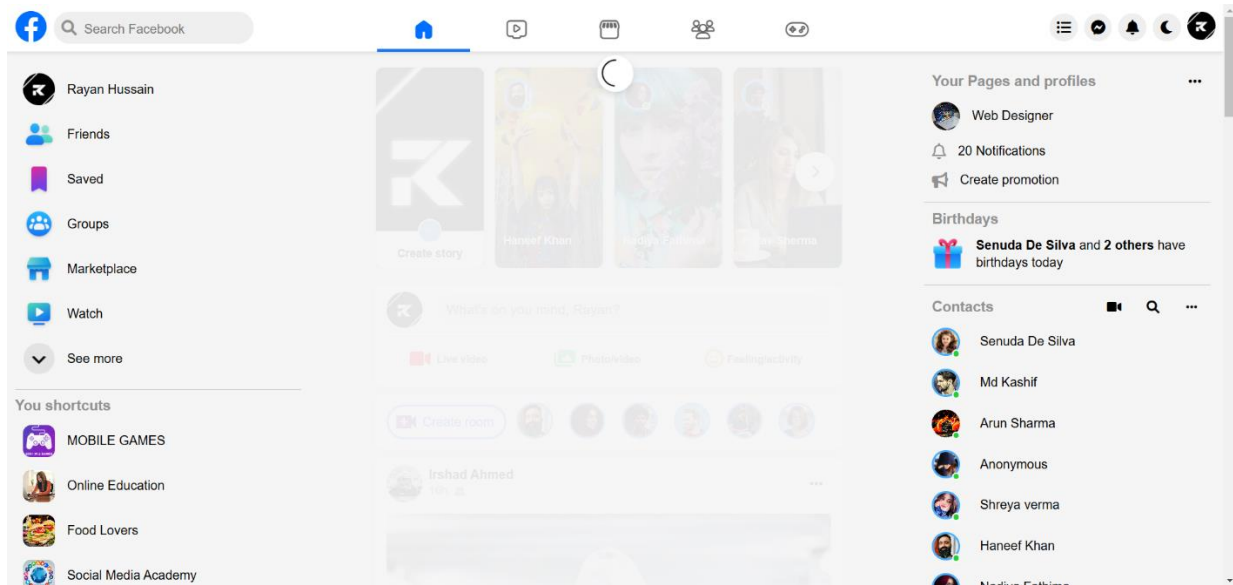


Fig 5. Refresh Home Function

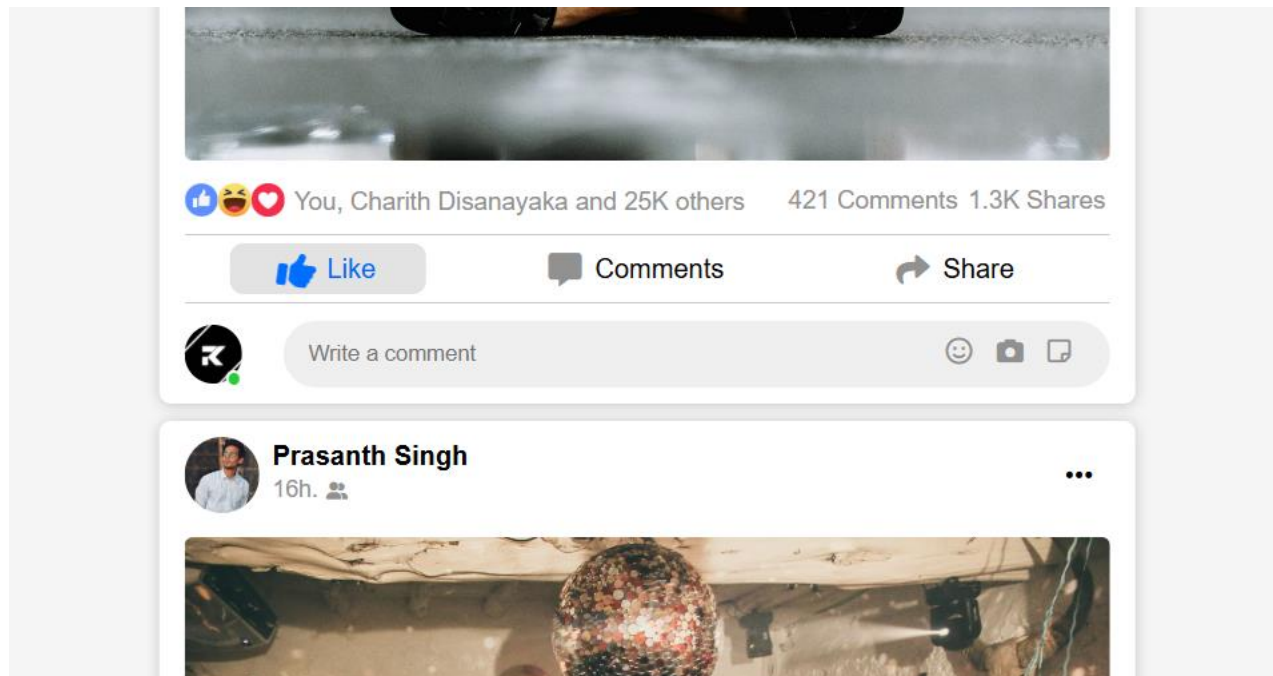


Fig 6. Working Like Button

Home Page:

Navigation Bar:

1. Left Section (Logo, Loading Spinner, Search Bar)

- **HTML Tags Used:**

- `<div class="left">`: The main container for the left section.
- `<div class="logo">`: This contains the Facebook logo, represented by an icon.
- `<div class="loading-spinner" id="spinner">`: The loading spinner container, identified by the `id="spinner"`.
- `<div class="search_bar">`: The search bar container.
- `<i class="fa-solid fa-magnifying-glass"></i>`: Font Awesome icon for the search button.
- `<input type="text" placeholder="Search Facebook">`: The text input field for the search bar.

- **Styling & Techniques Used:**

- CSS:**

- Flexbox is used for aligning items horizontally within the `.left` section to ensure responsive design.
- Icons are styled using Font Awesome, which provides scalable vector icons.
- The search bar input field is styled to match Facebook's design, ensuring the icon and input field are aligned properly.

Animation:

- The loading spinner (#spinner) is animated using CSS to display a loading effect, which could be triggered on page load or during asynchronous operations.

2. Centre Section (Main Navigation Items)

• HTML Tags Used:

- `<div class="center">`: The container for the central navigation links.
- `<div class="icon-container">`: A container for each navigation item (home, video, marketplace, etc.).
- `<i class="material-symbols-rounded">`: Material Design icons used for representing each section.
- `<div class="name-card">`: The name label for each navigation item (e.g., Home, Video).

• Styling & Techniques Used:

CSS:

- Flexbox is again used to evenly space the navigation items horizontally.
- `material-symbols-rounded` provides a modern, minimalistic design for the icons.
- Each icon-container has specific styling for padding, margin, and hover effects.
- The name-card is styled to display the name beneath each icon, ensuring they align neatly.

CSS Animation:

- Hover effects are added using CSS to change the color or scale of the icons when the user interacts with them, giving visual feedback.
- Transition effects ensure smooth hover animations for the icons and names.

3. Right Section (Menu, Messenger, Notifications, Dark Mode, Profile)

- **HTML Tags Used:**

- `<div class="right">`: The container for the right section of the navbar.
- `<div class="icon-container">`: Similar to the center section, each container holds an icon and its corresponding name label.
- `<i class="fa-solid fa-list-ul"></i>`: Font Awesome icon for the menu.
- `<i class="fa-brands fa-facebook-messenger"></i>`: Font Awesome icon for Messenger.
- `<i class="fa-solid fa-bell"></i>`: Font Awesome icon for notifications.
- `<i class="fa-solid fa-moon"></i>`: Icon for toggling dark mode.
- ``: Image tag to display the profile icon.

- **Styling & Techniques Used:**

CSS:

- Similar to the left and center sections, Flexbox is used to align the icons and text horizontally.
- Font Awesome and Material Design icons are utilized for a clean, recognizable look.
- **CSS Grid** could be used here to create a well-structured and responsive layout that works across different screen sizes.
- The profile-icon image is styled to maintain a circular, uniform shape and centered within its container.

Animation:

- For the dark mode icon (fa-moon), JavaScript can be used to toggle between light and dark themes, with smooth transitions handled by CSS.
- Hover effects are applied to the icons for better interactivity, as in the center section.

4. Overall Navbar Layout

- **HTML Tags Used:**

- `<nav>`: The main container element for the navigation bar. It wraps all the sections (left, center, right) and ensures proper layout alignment.
- `<div>`: Used extensively to divide the navbar into logical sections: left, center, and right.

- **Styling & Techniques Used:**

CSS:

- The navbar is styled using Flexbox to create a fluid and responsive layout. The sections are aligned to the left, center, and right to mimic Facebook's navbar structure.
- Specific width, padding, and margin values are used to space out the sections and items within the navbar.
- **CSS Media Queries:** Used to ensure that the navbar remains responsive on different screen sizes, particularly on mobile devices where the layout should adapt accordingly.

Animation:

- Smooth transitions and hover effects are added to ensure that when a user interacts with an element (such as the icons or profile section), the experience is dynamic and fluid.

```

index.html > html > body > nav
2   <html lang="en">
15  </body>
16
17  <nav>
18
19      <div class="left">
20
21          <div class="logo">
22              <i class="fa-brands fa-facebook"></i>
23          </div>
24          <div class="loading-spinner" id="spinner"></div>
25
26          <div class="search_bar">
27
28              <i class="fa-solid fa-magnifying-glass"></i>
29              <input type="text" placeholder="Search Facebook">
30
31          </div>
32
33      </div>
34
35      <div class="center">
36          <div class="icon-container">
37              <i class="material-symbols-rounded active">home</i>
38              <div class="name-card">Home</div>
39          </div>
40          <div class="icon-container">
41              <i class="material-symbols-rounded">live_tv</i>
42              <div class="name-card">Video</div>
43          </div>
44          <div class="icon-container">
45              <i class="material-symbols-rounded">storefront</i>
46              <div class="name-card">Marketplace</div>
47          </div>
48          <div class="icon-container">
49              <span class="fa-solid fa-users"></span>
50              <div class="name-card">Groups</div>

```

```

index.html > html > body > nav
2   <html lang="en">
15  </body>
17  <nav>
35      <div class="center">
52          <div class="icon-container">
53              <i class="material-symbols-rounded">home</i>
54              <div class="name-card">Home</div>
55          </div>
56      </div>
57
58      <div class="right">
59          <div class="icon-container">
60              <i class="fa-solid fa-list-ul"></i>
61              <div class="name-card">Menu</div>
62          </div>
63          <div class="icon-container">
64              <i class="fa-brands fa-facebook-messenger"></i>
65              <div class="name-card">Messenger</div>
66          </div>
67          <div class="icon-container">
68              <i class="fa-solid fa-bell"></i>
69              <div class="name-card">Notifications</div>
70          </div>
71          <div class="icon-container">
72              <i class="fa-solid fa-moon"></i>
73              <div class="name-card">Dark Mode</div>
74          </div>
75          <div class="icon-container">
76              
77              <div class="name-card">Profile</div>
78              <div id="profile-container"></div>
79          </div>
80      </div>
81
82  </nav>
83
84
85

```

Main Left (Profile Section) Section:

1. Profile Section

- **HTML Tags:**
 - `<div class="img">`, ``, and `<p>` are used to display the profile image and name.
- **CSS:**
 - Flexbox aligns the profile image and text. Border-radius is used for rounded images.
- **Animation:**
 - Hover effects can be added to scale or change colors for interactivity.

2. Navigation Items (Friends, Saved, Groups, etc.)

- **HTML Tags:**
 - Similar structure using `<div class="img">`, ``, and `<p>` for each section.
- **CSS:**
 - Flexbox ensures proper layout, and hover effects change background color or scale icons.
- **Animation:**
 - CSS transitions animate hover interactions for icons.

3. Shortcuts Section

- **HTML Tags:**
 - `<div class="shortcuts">`, ``, and `<p>` display shortcut icons and labels.
- **CSS:**
 - Flexbox or grid layout organizes shortcuts, with consistent spacing and fixed image sizes.
- **Animation:**

- Hover effects for scaling icons or changing background color.

4. “See More” Section

- **HTML Tags:**
 - `` and `<p>` are used to display the down arrow and label.
- **CSS:**
 - Icons may rotate or change color on hover.
- **Animation:**
 - Smooth transition or spinning animation for the arrow on hover.

5. Horizontal Line

- **HTML Tags:**
 - `<hr>` is used to visually separate sections.
- **CSS:**
 - Styled for width, height, and color, with possible fading effects on load.

Layout & Responsiveness

- **HTML Tags:**
 - `<div class="main">`, `<div class="left">` contain all sections.
- **CSS:**
 - Flexbox for layout, media queries for responsiveness, and consistent padding/margin.

```

87
88
89 <div class="main">
90
91
92 <!--left-->
93
94 <div class="left">
95
96   <div class="img">
97     
98     <p>Rayan Hussain</p>
99   </div>
100
101   <div class="img">
102     
103     <p>Friends</p>
104   </div>
105
106   <div class="img">
107     
108     <p>Saved</p>
109   </div>
110
111   <div class="img">
112     
113     <p>Groups</p>
114   </div>
115
116   <div class="img">
117     
118     <p>Marketplace</p>
119   </div>
120
121   <div class="img">
122     
123     <p>Watch</p>

```


Main Centre (Feed Section) Section:

1. Story Section (Top Box)

- **HTML Tags**

- `<div class="top_box">`, `<div class="story_card">`, and `` display the stories, profile images, and text.
- The `my_story_card` allows users to upload a story with a button and tag.

- **CSS**

- Flexbox is used to align the cards, images, and profile pictures.
- The `.overlay` and `.cricle` use absolute positioning to create overlays and circular profile images.
- Hover effects on the cards and profile images create interactivity.

- **Animation**

- CSS transitions animate hover effects for scaling, opacity change on overlay, and button interactions.

2. Post Section (My Post)

- **HTML Tags**

- `<div class="my_post">`, `<div class="post_top">`, and `<input>` create the input for a new post.
- Icons and buttons for posting a live video, photo, or activity are included.

- **CSS**

- Flexbox for layout of input and icons.
- Spacing, alignment, and hover effects are handled via CSS.

- **Animation**

- Hover effects on the icons change their background color or scale.

3. Room Section

- **HTML Tags**

- `<div class="room">`, ``, and `<p>` represent the room image and tag for creating a room.

- Multiple profile images in a room_profile are placed to represent the users in the room.
- **CSS**
 - Flexbox is used to layout the profile images and room details.
 - Proper image size and alignment are achieved using CSS.
- **Animation**
 - Hover effects are added to the room image for interactivity.

4. Friend's Post Section

- **HTML Tags**
 - `<div class="friends_post">`, ``, and `<div class="info">` show the friend's post with a profile image, time, and content.
 - The section includes a menu icon for post options and interactive components like likes, comments, and shares.
- **CSS**
 - Flexbox and grid are used to layout the post images, information, and buttons.
 - Icons are styled with specific colors for each type (like, haha, heart) for visual clarity.
- **Animation**
 - Hover effects are applied on icons and images for scaling and background changes.

Layout & Responsiveness

- **HTML Tags**
 - The section `<div class="center">` holds all components.
- **CSS**
 - Flexbox and grid layout techniques ensure that the components are aligned properly and remain responsive on different screen sizes.
 - Media queries help adjust the layout on mobile and tablet views.

```

172 <div class="center">
173
174   <div class="top_box">
175     <div class="next-button"><i class="fa-solid fa-angle-right"></i></div>
176
177     <div class="my_story_card">
178       <div class="overlay"></div>
179       
180
181       <div class="story_upload">
182         
183         <p class="story_tag">Create story</p>
184       </div>
185
186     </div>
187
188     <div class="story_card">
189       <div class="overlay"></div>
190       |
191
192       <div class="story_profile">
193
194         
195         <div class="cricle"></div>
196
197       </div>
198
199       <div class="story_name">
200         <p class="name">Haneef Khan</p>
201       </div>
202
203     </div>
204

```

```

<div class="my_post">
  <div class="post_top">
    
    <input type="text" placeholder="What's on you mind, Rayan?">
  </div>
  <hr>
  <div class="post_bottom">
    <div class="post_icon">
      <i class="fa-solid fa-video red"></i>
      <p>Live video</p>
    </div>
    <div class="post_icon">
      <i class="fa-solid fa-images green"></i>
      <p>Photo/video</p>
    </div>
    <div class="post_icon">
      <i class="fa-regular fa-face-grin yellow"></i>
      <p>Feeling/activity</p>
    </div>
  </div>
</div>

```

```

320 <div class="media">
321
322   <i class="fa-solid fa-ellipsis"></i>
323
324 </div>
325
326 </div>
327
328 
329
330 <div class="info">
331
332   <div class="emoji_img">
333     
334     
335     
336     <p>You, Charith Disanayaka and 25K others</p>
337   </div>
338
339   <div class="comment">
340     <p>421 Comments</p>
341     <p>1.3K Shares</p>
342   </div>
343
344 </div>
345
346 <hr>
347
348 <div class="like">
349
350   <div class="like_iconi">
351     <i class="fa-solid fa-thumbs-up"></i>

```

Main Right (Notification & Friends Section) Section:

1. Pages and Profiles Section

- **HTML Tags**

- `<div class="first_warpper">`, `<h2>`, `<div class="page_img">`, and `<p>` are used to display page information like your pages, notifications, and promotions.
- The `<i>` tag is used for icons like ellipsis, bell, and bullhorn to indicate menu actions and notifications.

- **CSS**

- Flexbox is used to arrange the page and profile details in a neat layout.
- Hover effects and proper spacing are used for the page icons and profile images.

- **Animation**

- Hover effects on the icons change the color or background. A transition effect on the page cards makes them appear smoother.

2. Birthdays Section

- **HTML Tags**

- `<div class="second_warpper">`, ``, and `<p>` are used to display birthday notifications, with the image representing the gift icon and the text showing the names of people celebrating their birthdays.
- The `` tag is used to style the names of users.

- **CSS**

- Flexbox is used to center the content of the birthday section.
- Proper image sizing and text styling ensure clarity.

- **Animation**

- Hover effects on the images or text can change the background or scale them.

3. Contacts Section

- **HTML Tags**

- `<div class="third_warpper">`, `<div class="contact">`, ``, and `<p>` represent the contacts in the right section.
- The `<div class="circle">` tag is used to display the status of the contact (like online or offline) with a circular background.
- Icons like video, magnifying glass, and ellipsis are included for additional interaction options.

- **CSS**

- Flexbox and grid are used to align the contact images and details properly.
- Circular status indicators are styled using border-radius and positioned with position: absolute.

- **Animation**

- Hover effects are applied to contacts for interactivity, such as scaling the contact image or changing the status circle's color.

Layout & Responsiveness

- **HTML Tags**

- The section `<div class="right">` contains all the components, ensuring modularity.

- **CSS**

- Flexbox and grid layout techniques ensure the content is aligned properly.
- Media queries adjust the layout to make the right section responsive across different devices, ensuring the sections stack on smaller screens.

```

738
739 <div class="right">
740
741   <div class="first_warpper">
742
743     <div class="page">
744
745       <h2>Your Pages and profiles</h2>
746       <div class="menu">
747         <i class="fa-solid fa-ellipsis"></i>
748       </div>
749     </div>
750
751     <div class="page_img">
752       
753       <p>Web Designer</p>
754     </div>
755
756     <div class="page_icon">
757       <i class="fa-regular fa-bell"></i>
758       <p>20 Notifications</p>
759     </div>
760
761     <div class="page_icon">
762       <i class="fa-solid fa-bullhorn"></i>
763       <p>Create promotion</p>
764     </div>
765
766   </div>
767 </div>
768
769 <hr>
770
771 <div class="second_warpper">

```

2.Login Page:

Main Login Section:

- **HTML:** Contains two columns (col-logo for the logo and description, col-form for the login form).
- **CSS:** Flexbox is used to arrange the columns. Input fields are styled for modern UI with the password toggle icon.
- **JS:** togglePassword() function switches password visibility, and validateForm() checks input validity.

Footer Section:

- **HTML:** Contains language options and navigation links, including "Sign Up," "Log In," and more.
- **CSS:** Styled with hover effects for links and lists, ensuring readability and interaction.
- **Accessibility:** Uses semantic tags (<small> for copyright) and ensures mobile responsiveness.

Input Fields and Password Toggle:

- **HTML:** Email, password fields, and a password visibility toggle with Font Awesome icons.
- **JS:** The password toggle switches between "show" and "hide" states.
- **CSS:** Input fields have a clean, uniform style, and the password icon is interactive.

Button and Links:

- **HTML:** Login button triggers form validation; “Create new Account” button is also provided.
- **CSS:** Buttons have hover effects and are styled for clarity.

Responsive Design:

- **CSS:** Flexbox/grid layout for desktop and mobile. Media queries ensure proper resizing of elements.

Animations:

- **CSS:** Smooth transitions and hover effects for buttons, input fields, and the password toggle.

```

17 <body>
18   <main>
19     <div class="row">
20       <div class="col-logo">
21         
22         <h2>Facebook helps you connect and share with the people in your life.</h2>
23       </div>
24       <div class="col-form">
25         <div class="form-container">
26
27           <input id="email" type="text" placeholder="Email address or phone number">
28           <input type="password" id="password" placeholder="Password">
29
30           <span id="toggle-password" style="cursor: pointer;">
31             <i class="fas fa-eye" id="eye-icon"></i> |
32           </span>
33           <button class="btn-login" onclick="validateForm()">Login</button>
34           <a href="#">Forgotten password?</a>
35           <button class="btn-new">Create new Account</button>
36         </div>
37         <p><a href="#"><b>Create a Page</b></a> for a celebrity, brand or business.</p>
38       </div>
39     </div>
40   </main>

```

3.Profile Card:

Profile Header:

- **HTML:** Contains the user's profile picture (profile-icon), name (profile-name), and a button (see-profiles-btn) for seeing all profiles.
- **CSS:** The profile header uses Flexbox for alignment. The see-profiles-btn has a styled appearance with appropriate spacing.

Profile Menu:

- **HTML:** A list (profile-menu) that includes menu items such as "Settings & Privacy," "Help & Support," "Display & Accessibility," "Give Feedback," and "Log Out."
- **Icons:** Font Awesome icons are used for a visually rich experience.
- **JS:** logout() is triggered on click of the logout option.
- **CSS:** Each menu item is styled with padding, icons, and text. The shortcut class displays a keyboard shortcut for "Give feedback."

Footer Section:

- **HTML:** Footer with privacy and terms links, and a copyright notice (Meta © 2025).
- **CSS:** Footer text is styled to appear smaller and neatly placed at the bottom.

Responsive Design:

- **CSS:** The layout adapts to different screen sizes using Flexbox, and the menu items are stacked vertically for mobile responsiveness.

Interactions:

- **CSS:** Hover effects are applied to menu items for a better user experience, and transitions are smooth for visual appeal.


```
profile.html > html > body > div.profile-box > ul.profile-menu > li > div.menu-item
2   <html lang="en">
11  <body>
12    <div class="profile-box">
13      <div class="profile-header">
14        <div class="profile-row">
17      </div>
18      <p class="profile-name">Rayan Hussain</p>
19    </div>
20    <div class="profile-line"></div>
21    <button class="see-profiles-btn"> See all profiles</button>
22  </div>
23  <ul class="profile-menu">
24    <li>
25      <div class="menu-item">
26        <i class="fa-solid fa-gear"></i>
27        <span class="menu-text">Settings & privacy</span>
28      </div>
29    </li>
30    <li>
31      <div class="menu-item">
32        <i class="fa-solid fa-question-circle"></i>
33        <span class="menu-text">Help & support</span>
34      </div>
35    </li>
36    <li>
37      <div class="menu-item">
38        <i class="fa-solid fa-moon"></i>
39        <span class="menu-text">Display & accessibility</span>
40      </div>
41    </li>
42    <li>
43      <div class="menu-item">
44        <i class="fa-solid fa-comment"></i>
45        <span class="menu-text">Give feedback</span>
46        <span class="shortcut">CTRL B</span>
47      </div>
```

6. Challenges Faced:

The challenges faced during implementation included ensuring the profile image maintained proper sizing and layout across devices, which was resolved using border-radius and max-width settings. Aligning the menu items required precise spacing with Flexbox. Mobile responsiveness posed a challenge, but was addressed with media queries to ensure smooth transitions. Styling buttons for different states and integrating Font Awesome icons required adjustments in alignment and size. Implementing smooth hover effects and a functional `logout()` required extra attention, along with ensuring accessibility through keyboard navigation and screen reader compatibility. Lastly, ensuring visual consistency and footer alignment across different screen sizes were key challenges.

7. Future Enhancements:

Future enhancements for this profile card could include integrating dynamic features like fetching user data through an API for a personalized experience. Adding animation effects for profile menu items could improve user interaction. A search functionality within the profile menu could help users find settings or support options more efficiently. Additionally, incorporating dark mode functionality to toggle between light and dark themes would enhance the visual appeal and accessibility. Improving mobile responsiveness further by introducing a collapsible menu for smaller screens could make the design more fluid and adaptable. Lastly, adding custom icons or illustrations for the menu options would enhance the user interface visually.

8. Conclusion:

In conclusion, this profile card design effectively captures the essence of a user-friendly interface by offering intuitive navigation, clear visual hierarchy, and essential functionalities like settings and support. By focusing on simplicity, accessibility, and user experience, it provides a solid foundation for users to manage their profiles easily. The future enhancements, such as dynamic data integration, mobile optimization, and visual improvements, will elevate the card's usability and adaptability. With continued refinements, this profile card could provide an even more engaging and personalized user experience across devices.

9. References:

- **MDN Web Docs** - Mozilla's official documentation for HTML, CSS, and JavaScript.
URL: <https://developer.mozilla.org/>

- **W3Schools** - A comprehensive resource for learning web development technologies.
URL: <https://www.w3schools.com/>

- **CSS-Tricks** - Offers tutorials, tips, and techniques for advanced CSS.
URL: <https://css-tricks.com/>

- **JavaScript.info** - A modern tutorial website for learning JavaScript.
URL: <https://javascript.info/>