



COLLEGE CODE : 9605

COLLEGENAME : Cape Institute of Technology ,
College in Levinjipuram, Tamil Nadu

DEPARTMENT : B.TECH Information Technology 3RDYEAR

STUDENT NM-ID : ODAB8E1179A9C4099F343C783816FDF6

ROLL NO. : 960523205010

DATE: 04-10-2025

COMPLETED A PROJECT NAME AS Phase- 5

Technology Project Name

**IBM-FE-Login Authentication
System**

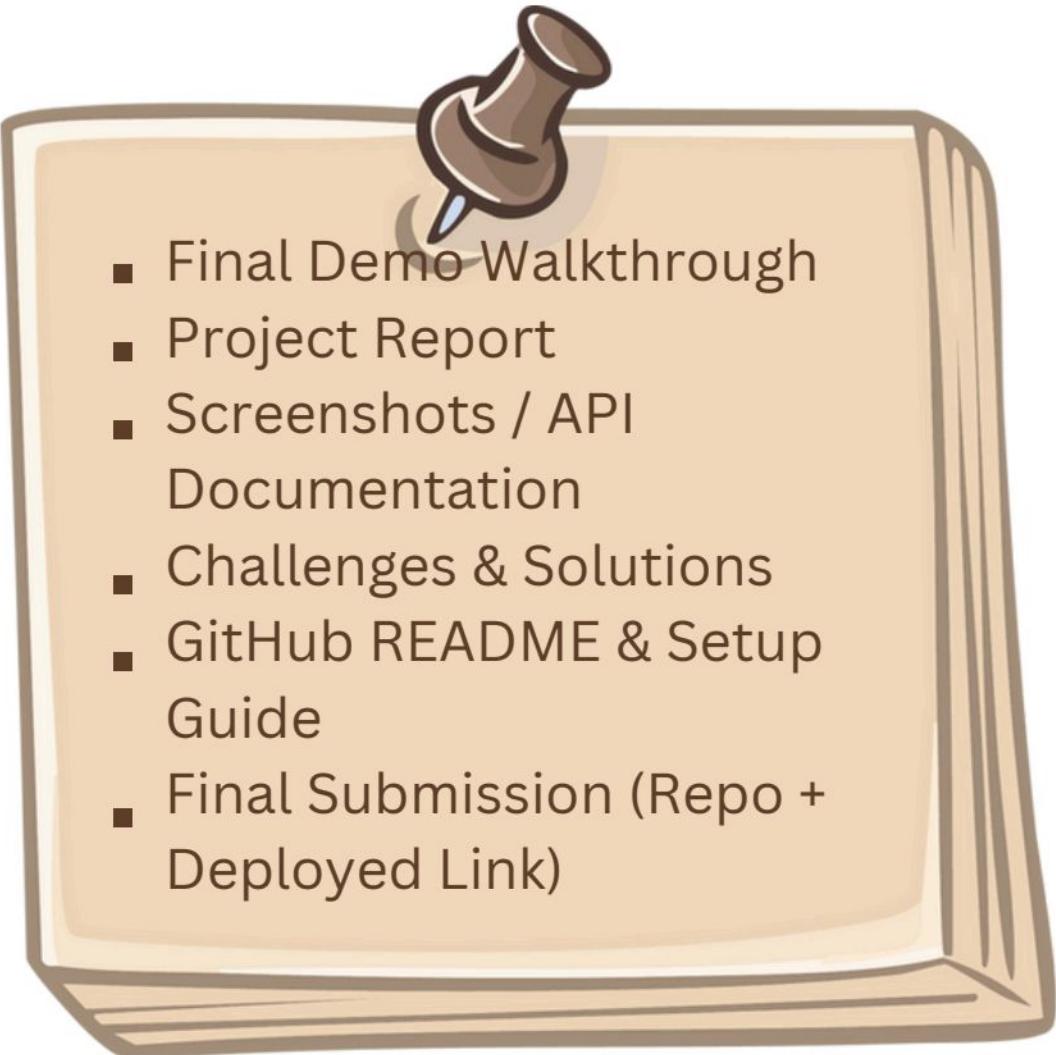
SUBMITTED BY,

NAME: GANESH BABU A

MOBILE NO: 9345776799

IBM-FE-Login Authentication System

Phase 5 – Project Demonstration & Documentation (Deadline - Week 10)

- 
- Final Demo Walkthrough
 - Project Report
 - Screenshots / API Documentation
 - Challenges & Solutions
 - GitHub README & Setup Guide
 - Final Submission (Repo + Deployed Link)



IBM-FE-Login Authentication System

Phase 5 — Project Demonstration & Documentation

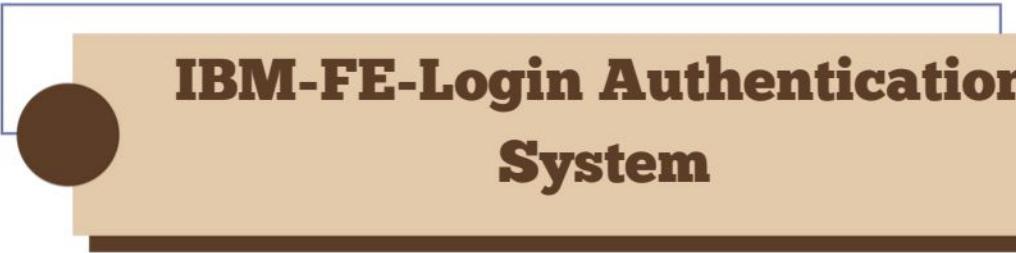
1. Final Demo Walkthrough

Objective:

To demonstrate the complete working of the Login Authentication System, developed using front-end technologies.

Demo Steps:

1. Open index.html in any modern web browser.
2. Display the login form with the following elements:
 - Email input
 - Password input
 - “Remember Me” checkbox
 - Social-login buttons (Google / GitHub placeholders)
3. Test client-side validation:
 - Invalid email → error message displayed
 - Password fewer than 8 characters → error shown
4. Demonstrate password visibility toggle.
5. Check the Remember Me option functionality.
6. Click Sign in:
 - Valid input → success message displayed
 - Invalid input → error feedback
7. Show placeholder redirects for social login buttons.
8. End the demo with a confirmation or success alert.



IBM-FE-Login Authentication System

2. Project Report

■ *Project Overview:*

The Login Authentication System is a secure, responsive login interface for web applications. It provides intuitive validation and visual feedback to ensure user-friendly authentication.

■ *Key Features*

- Email / Password login
- Client-side validation with error handling
- Password visibility toggle
- “Remember Me” option
- Social login buttons (Google & GitHub placeholders)
- Fully responsive layout for desktop & mobile

■ *Technologies Used:*

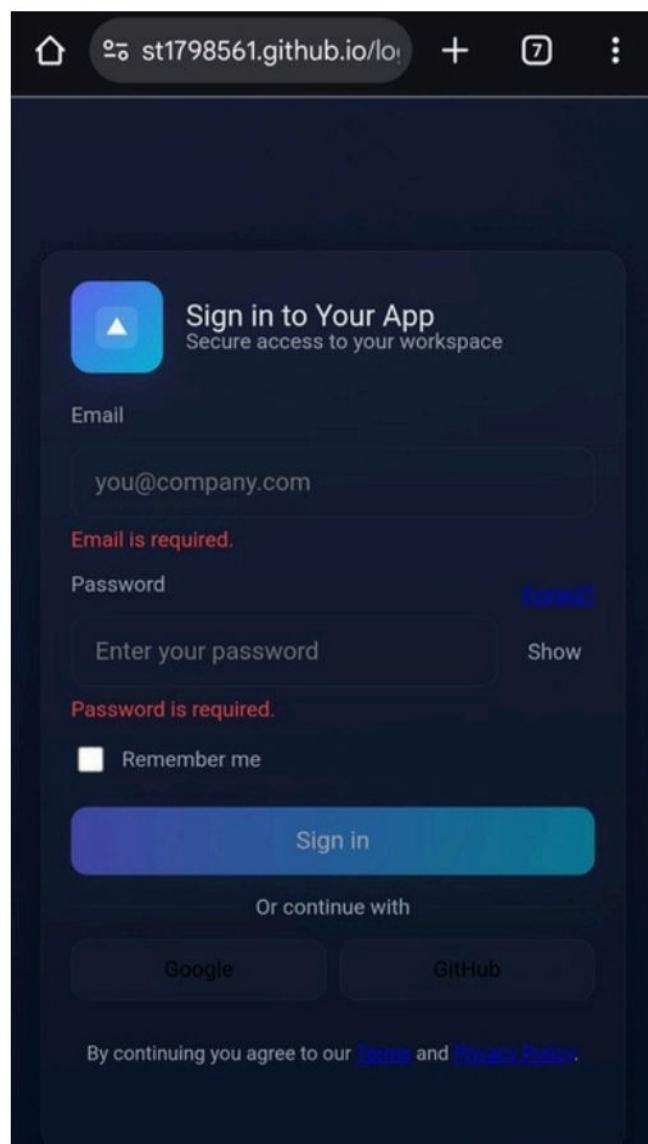
- °HTML5 – structure
- °CSS3 – styling, Flexbox, media queries
- °JavaScript (ES6+) – form validation & interactivity

■ *Project Flow:*

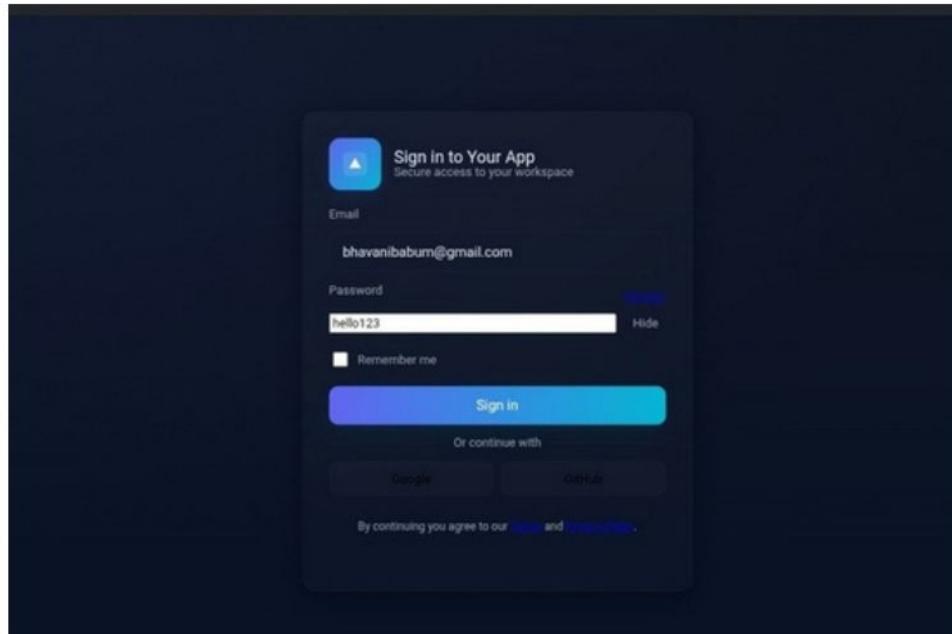
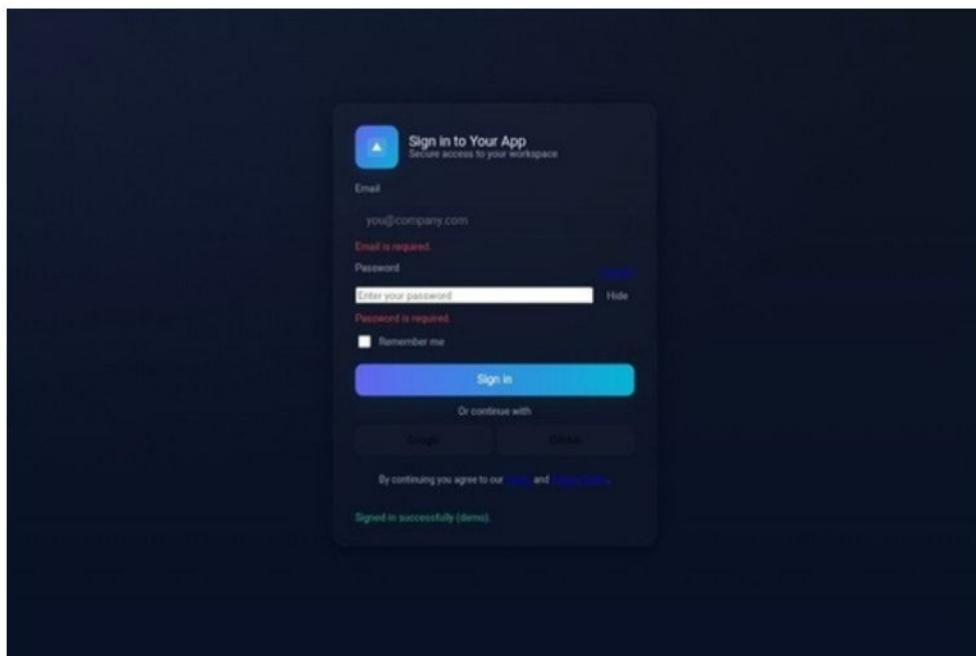
1. User opens the login page.
2. Inputs are validated on the client side.
3. Successful login shows a success message.
4. Errors are displayed dynamically.
5. Optional social login buttons redirect to placeholder pages.

IBM-FE-Login Authentication System

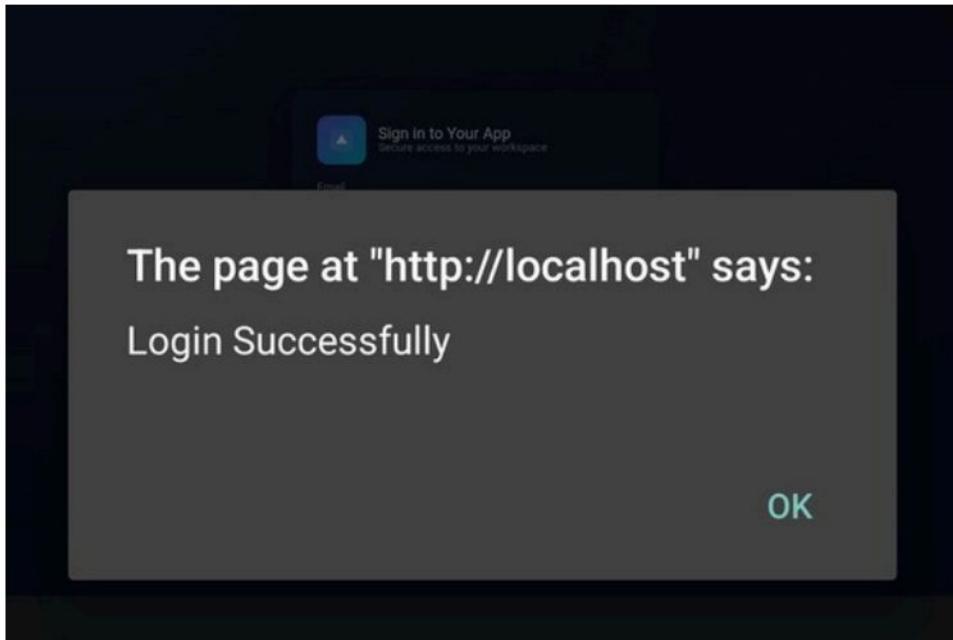
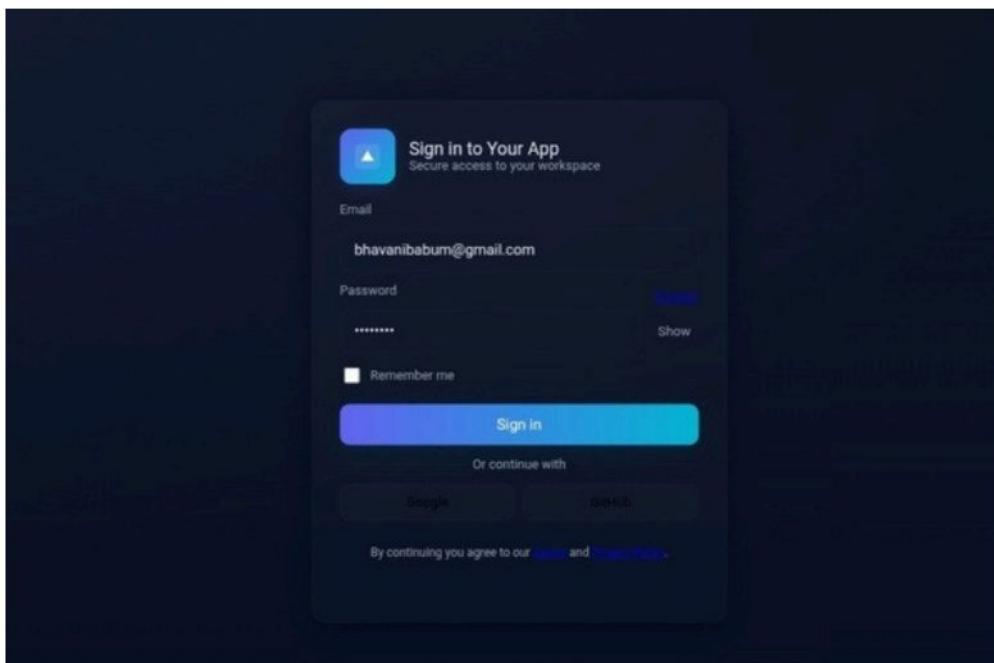
3. Screenshots / API Documentation



IBM-FE-Login Authentication System



IBM-FE-Login Authentication System





IBM-FE-Login Authentication System

Program :

```
<!doctype html>
<html lang="en">
<head>
<meta charset="utf-8" />
<meta name="viewport"
content="width=device-width,
initial-scale=1" />
<title>Login – Your App</title>
<meta name="description"
content="Secure sign in to Your
App" />
```



IBM-FE-Login Authentication System

```
<style>
:root{
  --bg:#0f1724;          --card:#0b1220;          --
  muted:#9aa6b2;          --accent:#6366f1;          --
  accent-2:#06b6d4;          --glass:              rgba(255,255,255,0.04);  --success:#10b981;
  --danger:#ef4444;          --radius:14px;          --max-
  width:420px;          --shadow: 0 6px 30px
  rgba(2,6,23,0.6);  font-family: Inter, ui-sans-
  serif, system-ui, -
  apple-system, "Segoe UI", Roboto, "Helvetica
  Neue", Arial;
  *{box-sizing:border-box}
html,body{height:100%}
```



IBM-FE-Login Authentication System

```
body{  
margin:0;  
background: radial-gradient(1200px 600px at  
10% 10%, rgba(99,102,241,0.08), transparent),  
linear-gradient(180deg, var(--bg),  
#071026 120%);  
color:#e6eef6;  
-webkit-font-smoothing:antialiased;  
-moz-osx-font-smoothing:grayscale;  
display:flex;  
align-items:center;  
justify-content:center;  
padding:32px;  
.container{ width:100%; max-width:var(--max-  
width); background: linear-gradient(180deg,  
rgba(255,255,255,0.02), rgba(255,255,255,0.01));  
border-radius:var(--radius); box-shadow:var(--  
shadow); padding:28px; border:1px solid  
rgba(255,255,255,0.04); backdrop-filter:  
blur(6px); }
```



IBM-FE-Login Authentication System

```
.header{display:flex;gap:14px;align-items:center;margin-bottom:18px;} .logo{width:56px;height:56px;border-radius:12px;display:grid;place-items:center;background:linear-gradient(135deg,var(--accent),var(--accent-2));flex:0 0 56px;box-shadow:0 6px 18px rgba(99,102,241,0.12);} .logo svg{width:34px;height:34px;opacity:0.98} .brand{line-height:1} .brand h1{font-size:18px;margin:0;font-weight:600} .brand p{margin:0;font-size:13px;color:var(--muted)} form{display:grid;gap:12px} label{font-size:13px;color:var(--muted);display:block;margin-bottom:6px} .field{display:flex;flex-direction:column;gap:6px} input[type="email"], input[type="password"] {width:100%;padding:12px 14px;border-radius:10px;border:1px solid rgba(255,255,255,0.04);background:transparent;color:inherit;outline:none;font-size:15px;transition:box-shadow .12s, border-color .12s;} input:focus{box-shadow:0 6px 18px rgba(99,102,241,0.06);border-color:rgba(99,102,241,0.18)}
```



IBM-FE-Login Authentication System

```
.password-row{display:flex;align-items:center;gap:10px} .pw-toggle{background:transparent;border:none;color:var(--muted);cursor:pointer;padding:8px;border-radius:8px} .muted-line{display:flex;justify-content:space-between;align-items:center;font-size:13px;color:var(--muted);} .row{display:flex;gap:12px;align-items:center} .checkbox{display:flex;gap:8px;align-items:center} .checkbox input{width:16px;height:16px} .btn{width:100%;padding:12px 14px;border-radius:10px;border:0;background:linear-gradient(90deg,var(--accent),var(--accent-2));color:white;font-weight:600;font-size:15px;cursor:pointer;box-shadow:0 8px 28px rgba(6,182,212,0.08);} .btn:active{transform:translateY(1px)} .btn[disabled]{opacity:0.6;cursor:not-allowed}
```



IBM-FE-Login Authentication System

```
.or{display:flex;align-items:center;gap:10px;color:var(--muted);font-size:13px}
.or::before,.or::after{content:"";flex:1;height:1px;background:rgba(255,255,255,0.03);border-radius:2px}
.socials{display:flex;gap:8px} .social-btn{flex:1;padding:10px 8px;border-radius:10px;background:rgba(255,255,255,0.02);border:1px solid rgba(255,255,255,0.03);display:flex;align-items:center;justify-content:center;gap:8px;font-weight:600;cursor:pointer}
.help-links{display:flex;justify-content:center;margin-top:8px;gap:16px;font-size:13px;color:var(--muted)}
.small{font-size:13px;color:var(--muted)}
.legal{margin-top:14px;font-size:12px;color:var(--muted);text-align:center} .error{color:var(--danger);font-size:13px} .success{color:var(--success);font-size:13px}
@media (max-width:420px)
{body{padding:18px}.container{padding:18px}}
</style>
</head>
```

IBM-FE-Login Authentication System

```
<body>
<main class="container" role="main" aria-
labelledby="login-heading">
  <header class="header">
    <div class="logo" aria-hidden="true">
      <svg viewBox="0 0 64 64" fill="none"
        xmlns="http://www.w3.org/2000/svg" aria-
        hidden="true">
        <rect x="8" y="8" width="48"
          height="48" rx="10" fill="white"
          opacity="0.12"/>
        <path d="M20 40 L32 20 L44 40 Z"
          fill="white" opacity="0.98"/>
      </svg>
    </div>
    <div class="brand">
      <h1 id="login-heading">Sign in to Your
        App</h1>
      <p>Secure access to your
        workspace</p>
    </div>
```



IBM-FE-Login Authentication System

```
</header><form id="loginForm" novalidate
method="post" action="/login"> <div class="field">
<label for="email">Email</label> <input id="email"
name="email" type="email" inputmode="email"
autocomplete="username" required aria-
required="true" placeholder="you@company.com">
<div id="emailError" class="error" aria-live="polite"
style="display:none"></div> </div> <div class="field">
<div class="row" style="justify-content:space-
between;align-items:flex-end">
<div style="flex:1">
  <label for="password">Password</label>
</div>
<div class="small"><a href="#" id="forgotLink">Forgot?</a></div>
</div><div class="password-row">
<input id="password" name="password"
type="password" autocomplete="current-password"
required aria-required="true" placeholder="Enter
your password" style="flex:1">
```



IBM-FE-Login Authentication System

```
<button type="button" class="pw-toggle" id="togglePw" aria-pressed="false" aria-label="Show password">Show</button>
</div>
<div id="pwError" class="error" aria-live="polite" style="display:none"></div>
</div> <div class="row" style="justify-content:space-between">
  <label class="checkbox"><input type="checkbox" id="remember" name="remember"><span class="small">Remember me</span></label>
  <div></div>
</div> <div>
  <button type="submit" class="btn" id="submitBtn">Sign in</button>
</div> <div class="or">Or continue with</div> <div class="socials" role="group" aria-label="Social sign-in options">
  <button type="button" class="social-btn" id="googleBtn">Google</button>
  <button type="button" class="social-btn" id="githubBtn">GitHub</button>
</div> <p class="legal">By continuing you agree to our <a href="#">Terms</a> and <a href="#">Privacy Policy</a>.
</p> <div id="statusMsg" aria-live="polite" style="margin-top:10px"></div>
</form> </main>
```



IBM-FE-Login Authentication System

```
<script>
  (function(){
    const form =
    document.getElementById('loginForm');
    const email = document.getElementById('email');
    const password =
    document .getElementById('password');
    const toggle =
    document .getElementById(' togglePw ');
    const emailError =
    document .getElementById('emailError ');
    const pwError =
    document .getElementById('pwError ');
    const status =
    document . getElementById('statusMsg ');
    const submitBtn =
    document.getElementById('submitBtn');// Toggle
    password visibility
    toggle.addEventListener('click', ()=>{
    const isShown = password.type === 'text';
    password.type = isShown ? 'password' : 'text';
    toggle.textContent = isShown ? 'Show' : 'Hide';
    toggle.setAttribute('aria-pressed', String(!isShown));
    // keep the cursor at the end
  })()
</script>
```



IBM-FE-Login Authentication System

```
password.focus();
const len = password.value.length;
password.setSelectionRange(len, len);
});
function validateEmail(v){
return /^[^@\s]+@[^\s@]+\.\w+/.test(v);
}
function validatePassword(v){
    return v.length >= 8;
}
function updateErrors(){
const emailVal = email.value.trim();
const pwVal = password.value;
let ok = true;

if(emailVal === ""){
    emailError.style.display = 'block';
    emailError.textContent = 'Email is required.';
    ok = false;
} else if(!validateEmail(emailVal)){
    emailError.style.display = 'block';
    emailError.textContent = 'Please enter a valid email
address.';
    ok = false;
} else {
    emailError.style.display = 'none';
    emailError.textContent = "";
}
```



IBM-FE-Login Authentication System

```
if(pwVal === ''){
    pwError.style.display = 'block';
    pwError.textContent = 'Password is required.';
    ok = false;
} else if(!validatePassword(pwVal)){
    pwError.style.display = 'block';
    pwError.textContent = 'Password must be at least
8 characters.';
    ok = false;
} else {
    pwError.style.display = 'none';
    pwError.textContent = "";
}
submitBtn.disabled = !ok;
return ok;
}
// Live validation
email.addEventListener('input', updateErrors);
password.addEventListener('input', updateErrors);
// Prevent native constraint UI and use our own
```



IBM-FE-Login Authentication System

```
email.addEventListener('invalid', (ev)=>{
  ev.preventDefault();
  updateErrors();
});

password.addEventListener('invalid', (ev)=>{
  ev.preventDefault();
  updateErrors();
});

// Initial state
updateErrors();

form.addEventListener('submit', async (e)=>{
  e.preventDefault();

  status.textContent = '';
  // run validation one more time
  if(!updateErrors()) return;
  submitBtn.disabled = true;
  submitBtn.textContent = 'Signing in...';
```



IBM-FE-Login Authentication System

```
try{
  const payload = {email: email.value.trim(), password: password.value, remember: document.getElementById('remember').checked};
  // demo delay to simulate network
  await new Promise(r=>setTimeout(r,800));
  // demo: always succeed (replace with real auth)
  const success = true;
  if(success){
    // Show inline success message
    status.textContent = "";
    const msg = document.createElement('div');
    msg.className='success'; msg.textContent='Signed in successfully (demo).';
    status.appendChild(msg);
    // Show an alert as you requested
    alert('Login Successfully');
    // Clear sensitive fields and reset form
    form.reset();
    updateErrors(); // update UI and disable submit until valid input
    // move focus back to email for next interaction
    email.focus();
  } else {
    throw new Error('Invalid credentials');
  }
}
```



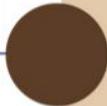
IBM-FE-Login Authentication System

```
 }catch(err){  
     status.textContent = '';  
     const msg = document.createElement('div');  
     msg.className='error'; msg.textContent =  
     err.message || 'Sign-in failed. Try again.';  
     status.appendChild(msg);  
 }finally{  
     submitBtn.disabled = false;  
     submitBtn.textContent = 'Sign in';  
 }  
});  
  
document.getElementById('googleBtn').addEventListener('click', ()=>{window.location.href =  
'/auth/google'});  
  
document.getElementById('githubBtn').addEventListener('click', ()=>{window.location.href =  
'/auth/github'});  
// Accessibility: allow Enter key on toggle to work  
like click
```



IBM-FE-Login Authentication System

```
toggle.addEventListener('keydown',
(ev)=>{ if(ev.key === 'Enter' || ev.key === ' ')
  { ev.preventDefault(); toggle.click(); }
});
})();
</script>
</body>
</html>
```



IBM-FE-Login Authentication System

API Documentation (Demo):

- *Login Endpoint:* POST /login

```
{  
  "email":  
    "user@example.com",  
  "password": "string",  
  "remember": true  
}
```

- *Response (Demo):*

```
{  
  "success": true,  
  "message": "Signed in  
  successfully"  
}
```



IBM-FE-Login Authentication System

4. Challenges & Solutions

Challenges & Solutions

Challenge	Solution
Email Validation	Implemented JavaScript regular expressions // "^(\\s@)+@[\\s@]+l.[\\s@]+\$/ onimgehir ensage
Password Rules	Enforce a minimum of 8 characters with instant validation feedback to ensure secure password
Password Toggle Functionality	Using JavaScript to switch input type between "password" and "text", uses to view or hide their
Responsive Layout	Utilized CSS Flexbox and media queries to make the login form responsive across mobile, tablet
Backend Simulation	Used JavaScript's setTimeout() to mimic a real backend login request and display success or

5. GitHub Repository & Setup Guide

Repository Name: IBM-FE-Login Authentication System

GitHub Link: [🔗](https://github.com/ganesh9345/login-project)

<https://github.com/ganesh9345/login-project/>



IBM-FE-Login Authentication System

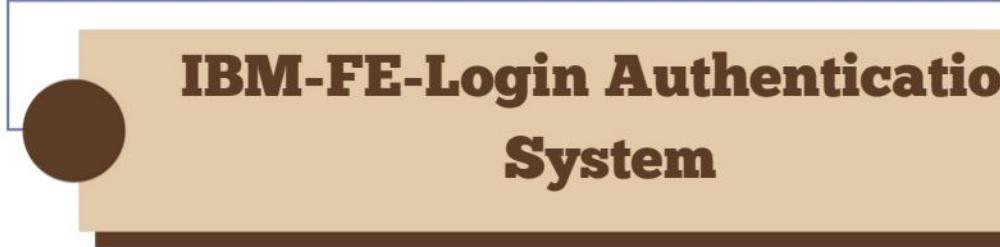
- ***GitHub Repository***

The complete project code has been uploaded to GitHub for version control and easy accessibility. It includes all HTML, CSS, and JavaScript files used in the Login Authentication System project.

- ***Setup Guide***

To run the project locally:

1. Open your browser and go to your GitHub repository link.
2. Click on “**Code**” → “**Download ZIP**” and extract the folder.
3. Open the extracted folder in **Visual Studio Code** (or any preferred editor).
4. Locate the file named **index.html**.
5. Double-click the file or right-click → **Open with** → **Browser** to view the project.



IBM-FE-Login Authentication System

5.Final Submission (Repo + Deployed Link)

Final Submission

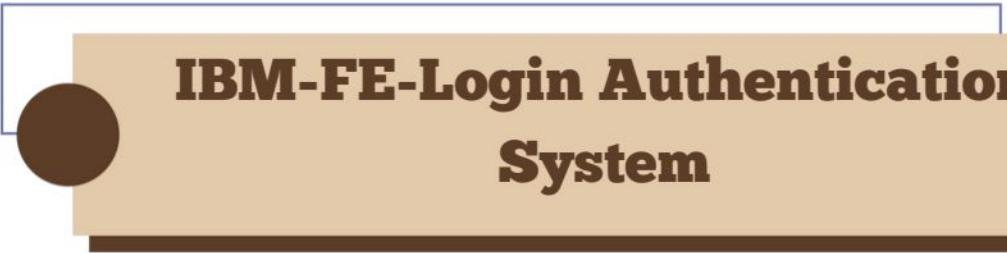
The completed project has been successfully uploaded and deployed for evaluation.

GitHub Repository Link:

<https://ganesh9345.github.io/login-project/>

Deployed Link: Nil

The links provide access to the fully functional Login Authentication System (Front-End) project developed using HTML, CSS, and JavaScript.



IBM-FE-Login Authentication System

Conclusion

As a team, we successfully developed the **Login Authentication System (Front-End)** using **HTML, CSS, and JavaScript** under the guidance of **IBM SkillsBuild** through the **Nan Mudhalvan** program. This project helped us gain strong practical knowledge in **web technologies**, including **form validation, responsive design, and interactive UI development**.

Working together enhanced our teamwork, creativity, and technical problem-solving abilities.

Overall, this experience has greatly improved our understanding of front-end development and its real-world applications.

Acknowledgement

We sincerely thank **IBM SkillsBuild** and **Nan Mudhalvan** for providing us this wonderful learning platform.

We also express our heartfelt gratitude to **our college mentors, faculty coordinators, and team members** for their continuous support and guidance throughout the project journey.

Their encouragement and feedback helped us successfully complete the project within the given timeline.



IBM-FE-Login Authentication System

Final Submission

- **Project Title:** Login Authentication System
- **Technology Used:** HTML, CSS, JavaScript
- **Organization:** IBM SkillsBuild x Nan Mudhalvan
- **GitHub Repository:**
<https://ganesh9345.github.io/login-project/>

Thank You 

- Submitted by the Project Team
- **Team Members:**
 - 1. Bhavani B
 - 2. Siva Thanu Pillai B S
 - 3. Ganesh Babu A
 - 4. Sathish Kumar S
- **Under:** IBM SkillsBuild – Nan Mudhalvan
Front-End Technology Program

