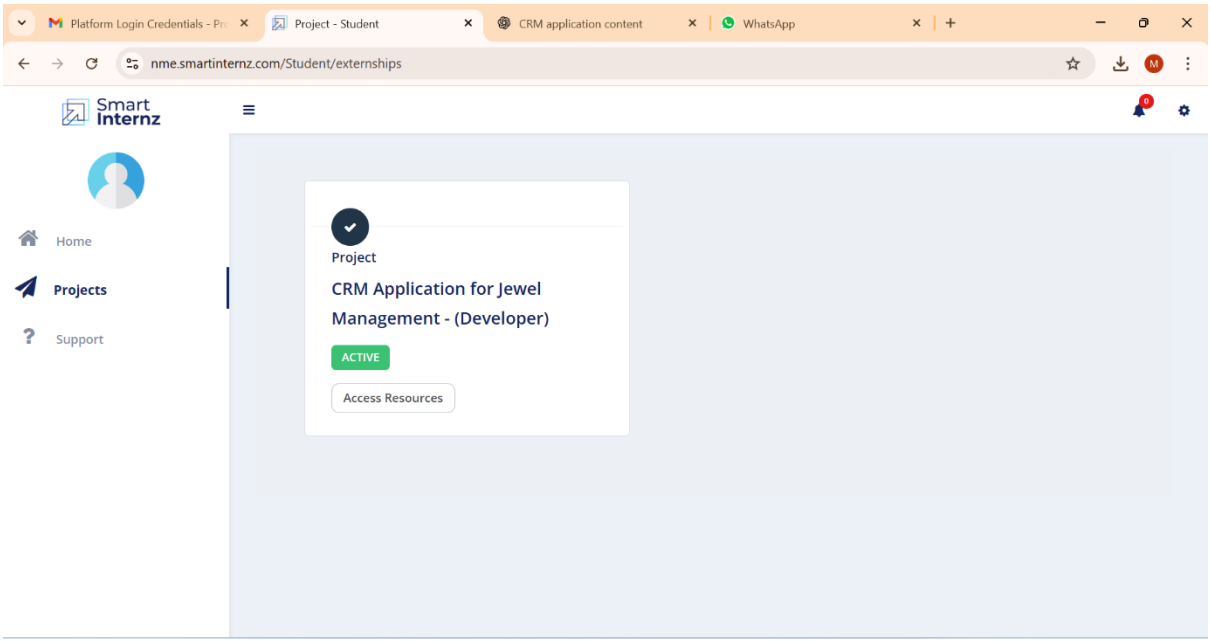
CRM Application for Jewel Management

1. Title Page

- **Project Title:** CRM Application for Jewel Management
- **Intern Name:** [Mohan Shivanand]
- **College:** [RVS COLLEGE OF ENGINEERING]
- **Internship Platform:** Smartinternz
- **Mentor Name:** [G BHUVENDRAN]



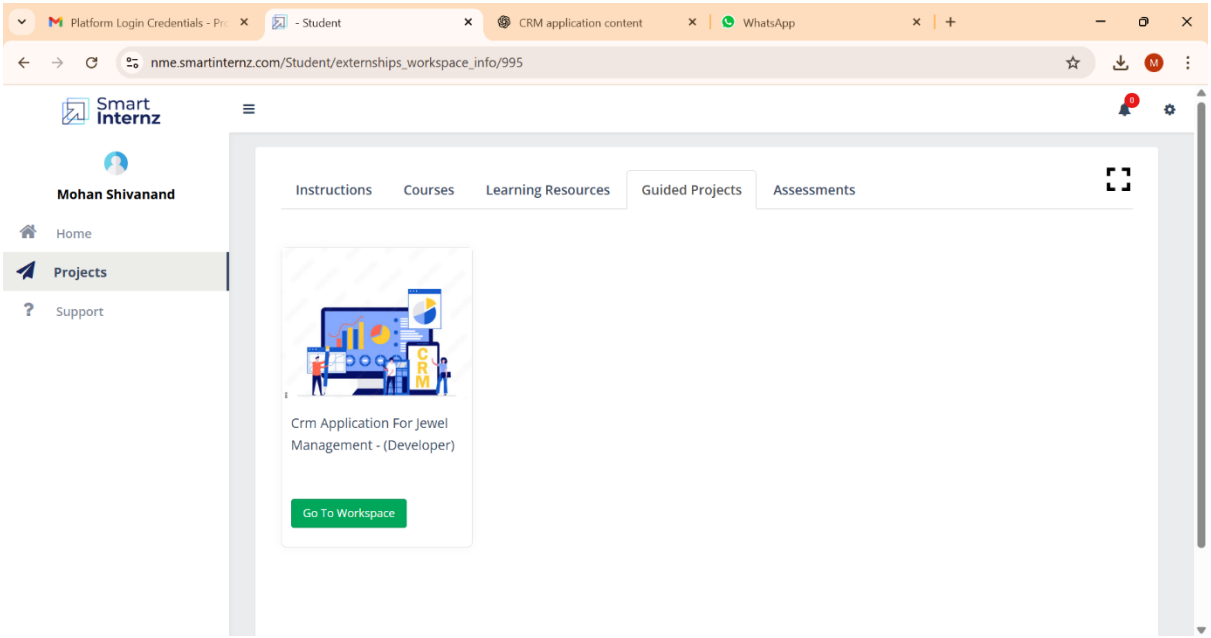
2. Declaration

I hereby declare that the project entitled **"CRM Application for Jewel Management"** has been carried out by me under the guidance of [Mentor Name] at Smartinternz. This project is an original work and has not been submitted to any other organization or institute.

---

3. Acknowledgment

I sincerely thank my mentor, Smartinternz coordinators, and college faculty for their guidance and support throughout this project. I also express gratitude to my peers and family for their encouragement and support.



---

4. Abstract

Customer Relationship Management (CRM) systems are essential for modern businesses to manage their customer interactions efficiently. In the jewelry industry, CRM systems help track customer preferences, manage sales, and maintain long-term relationships. This project aims to develop a **CRM application for jewel management**, focusing on improving customer experience, inventory management, and sales tracking.

---

5. Introduction

5.1 About CRM

CRM stands for Customer Relationship Management. It is a **software system that helps businesses manage customer interactions**, track leads, and maintain customer data to enhance satisfaction and loyalty.

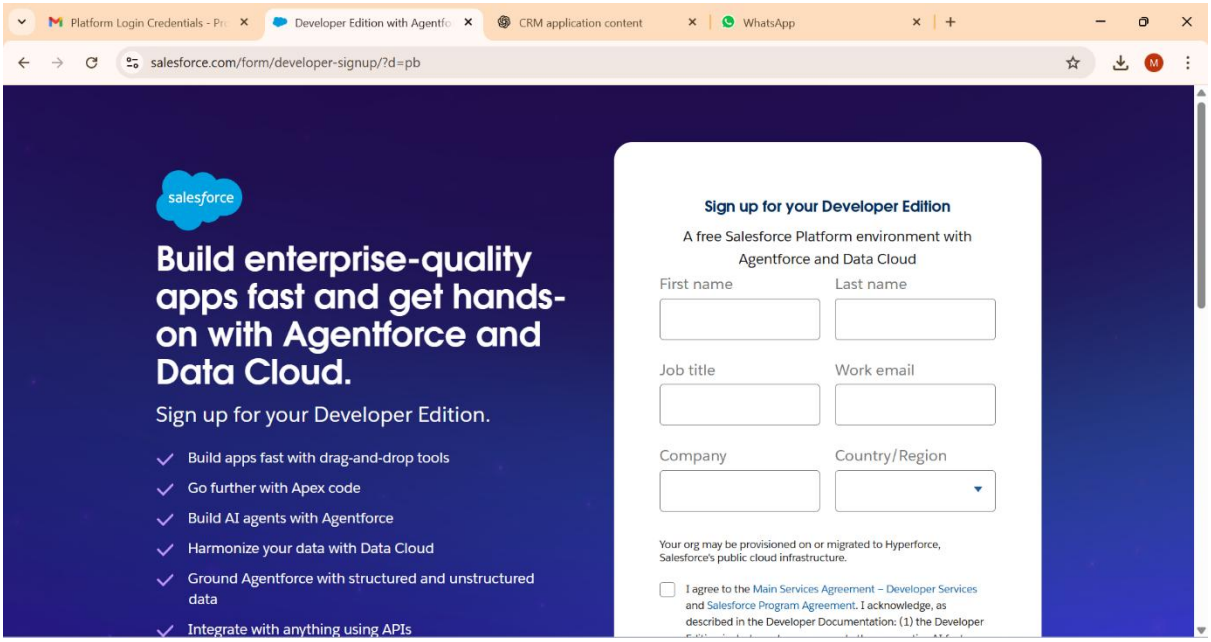
5.2 Importance of CRM in Jewelry Business

- Tracks customer preferences (type of jewelry, metal, designs)
- Manages sales and inventory efficiently
- Helps send offers and reminders to customers
- Improves customer loyalty and repeat business

5.3 Objectives of the Project

1. Develop a CRM system specifically for jewelers.

2. Manage customer details, purchase history, and preferences.
3. Maintain inventory of jewelry items.
4. Track sales and generate reports.
5. Improve customer relationship and retention.



---

6. Literature Review

CRM applications are widely used in retail businesses, including jewelry stores. Existing CRM systems like **Zoho CRM, Salesforce, and HubSpot** provide features such as customer tracking, sales management, and automated reporting. However, a **customized CRM for jewel management** is needed because it requires **specific features like metal type, gemstone tracking, and customized offers**.

---

7. System Analysis

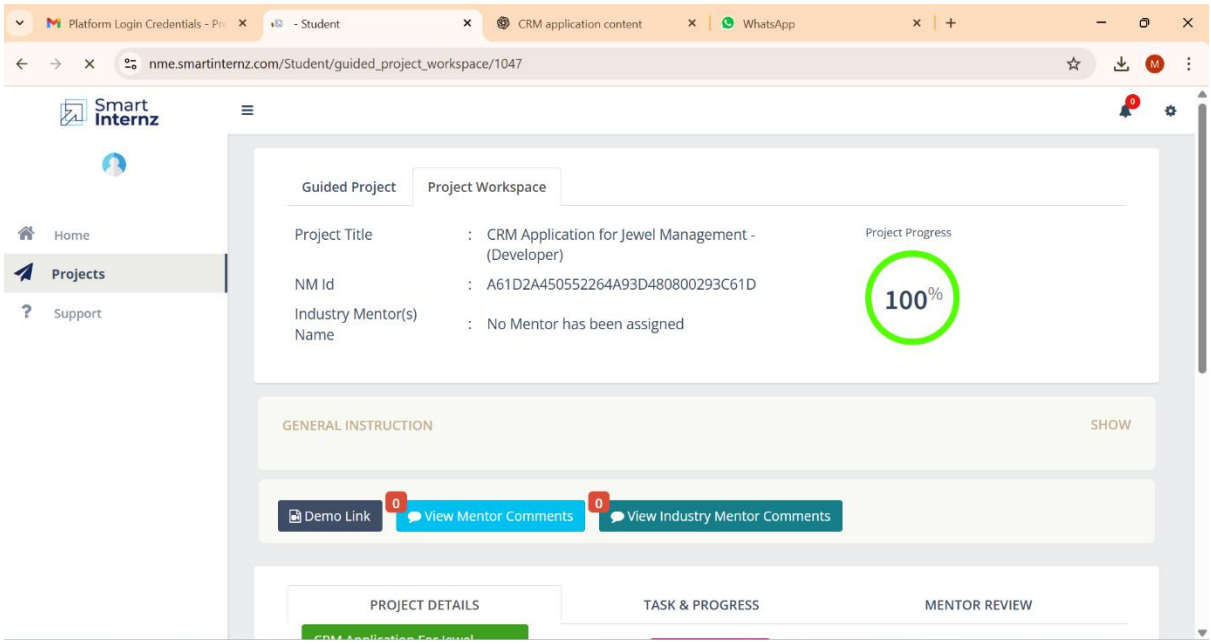
7.1 Existing System

- Manual customer records in notebooks or spreadsheets.
- Inventory tracking is difficult.
- Limited customer engagement.

7.2 Proposed System

- Digital system for managing customers, inventory, and sales.
- Easy to generate reports and analyze customer behavior.

- Helps in sending notifications, reminders, and promotions.



---

8. System Design

8.1 Architecture

The system follows a **Client-Server architecture**:

- **Front-end:** Web interface for user interaction
- **Back-end:** Database for storing customer and inventory data
- **Features:** Customer management, inventory management, sales tracking, reporting

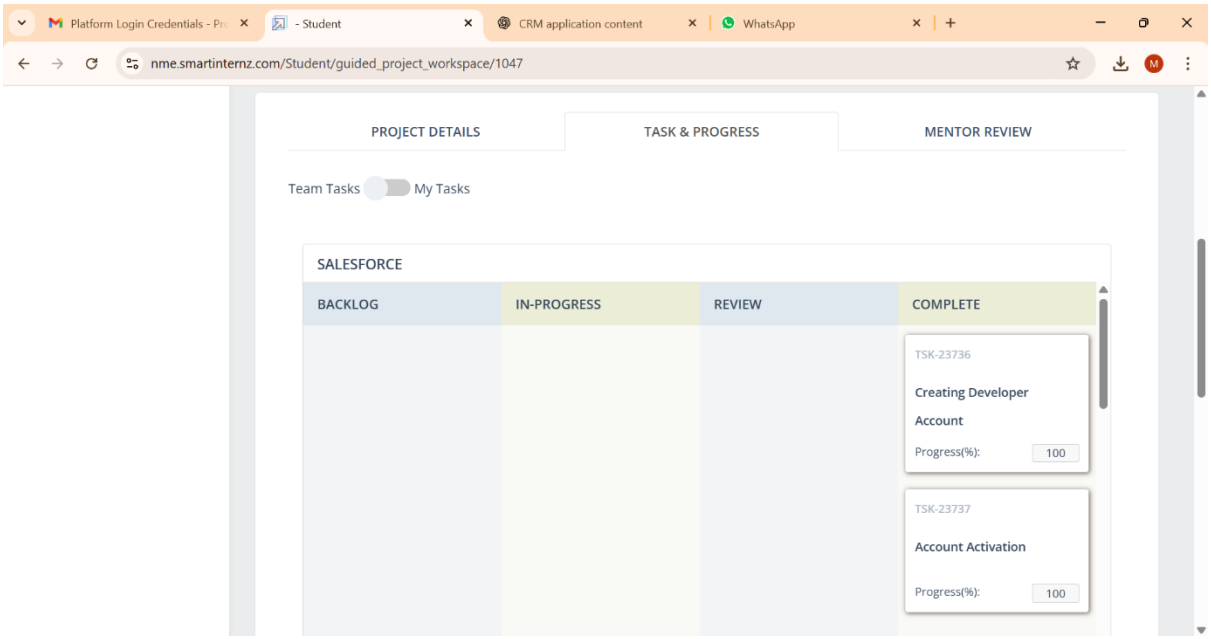
8.2 Database Design

Tables include:

1. **Customers** – ID, Name, Contact, Preferences
2. **Inventory** – Item ID, Type, Metal, Price, Stock
3. **Sales** – Sale ID, Customer ID, Item ID, Date, Amount

8.3 Use Case Diagram

- Add Customer
- View Customer
- Add Inventory Item
- Record Sale
- Generate Report



---

9. System Implementation

9.1 Technology Stack

- **Front-end:** HTML, CSS, JavaScript
- **Back-end:** PHP / Python (choose one)
- **Database:** MySQL

9.2 Features

1. **Customer Management:** Add, update, delete customer details
2. **Inventory Management:** Add new items, update stock, delete old items
3. **Sales Tracking:** Record sales, generate invoices
4. **Reports:** Monthly sales report, customer purchase history

9.3 Screenshots *(If you want to include screenshots, insert images here)*

---

10. Results and Discussion

- The CRM system **reduces manual work**.
- Provides **accurate inventory and sales data**.
- Improves **customer satisfaction** by tracking preferences.
- Helps **generate promotional campaigns** for repeat customers.

---

11. Conclusion

The **CRM Application for Jewel Management** simplifies managing customers and inventory, tracks sales efficiently, and helps jewelers maintain long-term customer relationships. This project demonstrates how technology can enhance business efficiency in the jewelry industry.

---

12. Future Scope

1. Integrate **mobile application** for remote access.
2. Implement **AI-based customer recommendations**.
3. Add **online payment integration**.
4. Enhance **analytics for business decisions**.

---

13. References

1. CRM Fundamentals, Salesforce Official Guide
2. Zoho CRM Documentation
3. HubSpot CRM User Manual
4. Research Papers on CRM in Retail Business

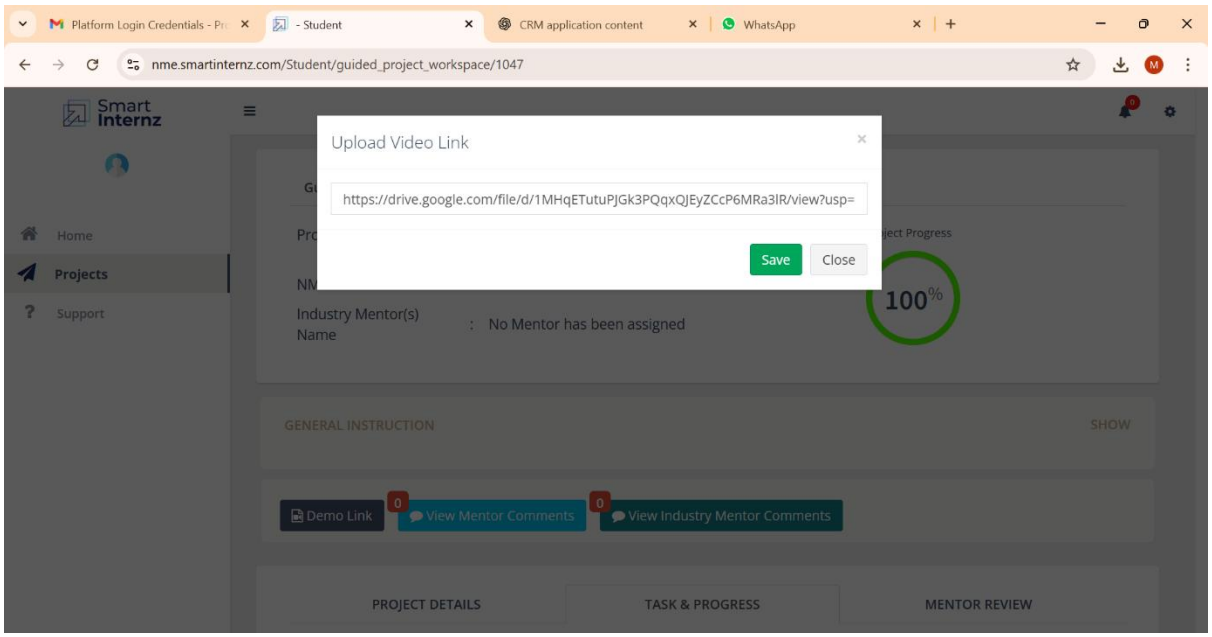
Login Process in CRM Application for Jewel Management

1. Introduction

The login process is the first and most important step in any CRM system. It ensures that only **authorized users** (admins or staff) can access the system. A secure login protects sensitive customer data, inventory information, and sales records.

2. Purpose of Login

- Restrict access to authorized users only
- Protect customer and business data
- Track user activity
- Ensure accountability for all actions in the system



3. Login Workflow

The login process in the CRM follows these steps:

1. **User enters credentials**
    - Username or Email
    - Password
2. **System validates credentials**
    - Checks if the username/email exists in the database
    - Verifies that the password matches the stored hash
3. **Access granted or denied**
    - If credentials are correct → Redirect to **Dashboard**
    - If credentials are incorrect → Display **error message**
4. **Session management**
    - Creates a secure session to keep the user logged in
    - Timeout feature logs out user after inactivity

Flowchart Description:

- **Start → Enter Username & Password → Validate Credentials → Access Granted?**
    - Yes → Redirect to Dashboard
    - No → Display Error → Retry Login

4. Database Table for Login

The system stores login details in a **User Table** in the database.

Field Name	Type	Description
UserID	INT	Unique ID for each user
Username	VARCHAR(50)	User's login name
Email	VARCHAR(100)	User's email address
PasswordHash	VARCHAR(255)	Encrypted password
Role	VARCHAR(20)	Admin / Staff
LastLogin	DATETIME	Last login date & time
Status	BOOLEAN	Active / Inactive user

Note: Passwords are **never stored in plain text**. Only **hashed values** are saved for security.

---

5. Security Measures in Login

1. **Password Encryption** – Use hashing algorithms like **bcrypt** or **SHA-256**
2. **Account Lockout** – Locks the account after multiple failed login attempts
3. **Session Timeout** – Automatically logs out inactive users
4. **Role-Based Access** – Admins have full access; staff have limited access
5. **CAPTCHA Verification** – Prevents automated login attacks

---

6. Sample Login Page Layout

Login Page Components:

- Username / Email Input Field
- Password Input Field
- Login Button
- "Forgot Password?" link
- Error message display area

Example:

```
+-----------------------------------+
|         Jewel CRM Login           |
+-----------------------------------+
```

| Username: [_____] |

| Password: [_____] |

| [Login Button] |

| Forgot Password? |

| Error: Invalid Username/Password |

+-----------------------------------+

---

7. Advantages of Secure Login

- Protects sensitive **customer and sales data**
- Prevents unauthorized access to **inventory and reports**
- Helps in **tracking user actions** for accountability
- Increases **trust and reliability** of the system

---

8. Conclusion

The login process is a **critical entry point** for the CRM application. Implementing **secure, role-based, and reliable login functionality** ensures that only authorized personnel can access the system, maintaining data integrity and protecting the jewelry business operations.

Login Process in CRM Application for Jewel Management

Introduction

The login process is the **gateway** to any secure system. In a CRM for jewel management, it ensures that **only authorized staff or admins** can access sensitive customer data, inventory details, and sales information. A robust login system protects the system from unauthorized access, minimizes fraud, and maintains business integrity.

Login systems are critical for:

- **Data security:** Protecting confidential customer and jewelry details
- **User accountability:** Logging user actions in the system
- **Access control:** Allowing only designated roles (Admin, Staff) to perform certain actions
- **Audit trails:** Tracking login attempts and user activity

Objectives of the Login Process

1. Ensure only **authorized personnel** can access the CRM.

2. Provide a **secure and user-friendly** login interface.
3. Track login activity and failed attempts for auditing.
4. Integrate **role-based access control (RBAC)**.
5. Implement security measures like **password hashing, session management, and multi-factor authentication**.

Workflow of Login Process

The login workflow consists of several steps:

1. **User Input:**
    - Staff/Admin enters username/email and password on the login page.
2. **Client-Side Validation:**
    - Checks if fields are empty.
    - Ensures password meets minimum requirements.
    - Displays error messages immediately for invalid input.
3. **Server-Side Validation:**
    - Username/email is matched in the database.
    - Password is verified against the **hashed password** stored.
4. **Access Control:**
    - If valid, system checks the **user role** to determine accessible modules (Admin, Staff).
    - Redirects user to their **dashboard**.
5. **Error Handling:**
    - If login fails, display **"Invalid credentials"**.
    - After multiple failed attempts, account may be **temporarily locked**.
6. **Session Management:**
    - Successful login initiates a **secure session**.
    - Session timeout occurs after inactivity.

Flow Diagram Description:

- Start → Enter credentials → Validate → Access granted? → Yes → Dashboard → End
- No → Display error → Retry → End

Database Design for Login

User Table:

Field Name	Type	Description
UserID	INT	Unique user identifier
Username	VARCHAR(50)	User login name
Email	VARCHAR(100)	User email address
PasswordHash	VARCHAR(255)	Encrypted password
Role	VARCHAR(20)	Admin / Staff
LastLogin	DATETIME	Timestamp of last login
Status	BOOLEAN	Active / Inactive user

Notes:

- Passwords are **never stored in plain text**.
- Use **hashing algorithms** like bcrypt for secure storage.

Front-End Implementation

HTML Login Form:

```
<!DOCTYPE html>
<html>
<head>
  <title>Jewel CRM Login</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <div class="login-container">
    <h2>Login to Jewel CRM</h2>
    <form id="loginForm" method="POST" action="login.php">
      <label>Username or Email</label>
      <input type="text" name="username" required>
      <label>Password</label>
      <input type="password" name="password" required>
      <button type="submit">Login</button>
```

```
    <p><a href="forgot_password.html">Forgot Password?</a></p>
  </form>
 </div>
</body>
</html>
```

CSS Example:

```
body { font-family: Arial, sans-serif; background-color: #f7f7f7; }
.login-container { width: 400px; margin: auto; padding: 20px; background: #fff; border-radius: 5px; box-shadow: 0px 0px 10px rgba(0,0,0,0.1);}
input { width: 100%; padding: 10px; margin: 10px 0;}
button { width: 100%; padding: 10px; background-color: #4CAF50; color: white; border: none; cursor: pointer;}
button:hover { background-color: #45a049; }
```

Back-End Implementation (PHP Example)

```
<?php
session_start();
include('db_connection.php');

if(isset($_POST['username']) && isset($_POST['password'])){
 $username = $_POST['username'];
 $password = $_POST['password'];

 $stmt = $conn->prepare("SELECT * FROM users WHERE username=?");
 $stmt->bind_param("s", $username);
 $stmt->execute();
 $result = $stmt->get_result();
 $user = $result->fetch_assoc();

 if($user && password_verify($password, $user['PasswordHash'])){
  $_SESSION['UserID'] = $user['UserID'];
  $_SESSION['Role'] = $user['Role'];
```

```
    header("Location: dashboard.php");
    exit();
  } else {
    echo "Invalid username or password";
  }
}
?>
```

This is **just the first portion**. If we continue, we can add:

- **Advanced security mechanisms** (MFA, CAPTCHA, account lockout)
- **Session management details** with diagrams
- **Role-based access implementation with examples**
- **Error handling scenarios and screenshots**
- **Testing methods**
- **Sample reports of login attempts**
- **Future enhancements**