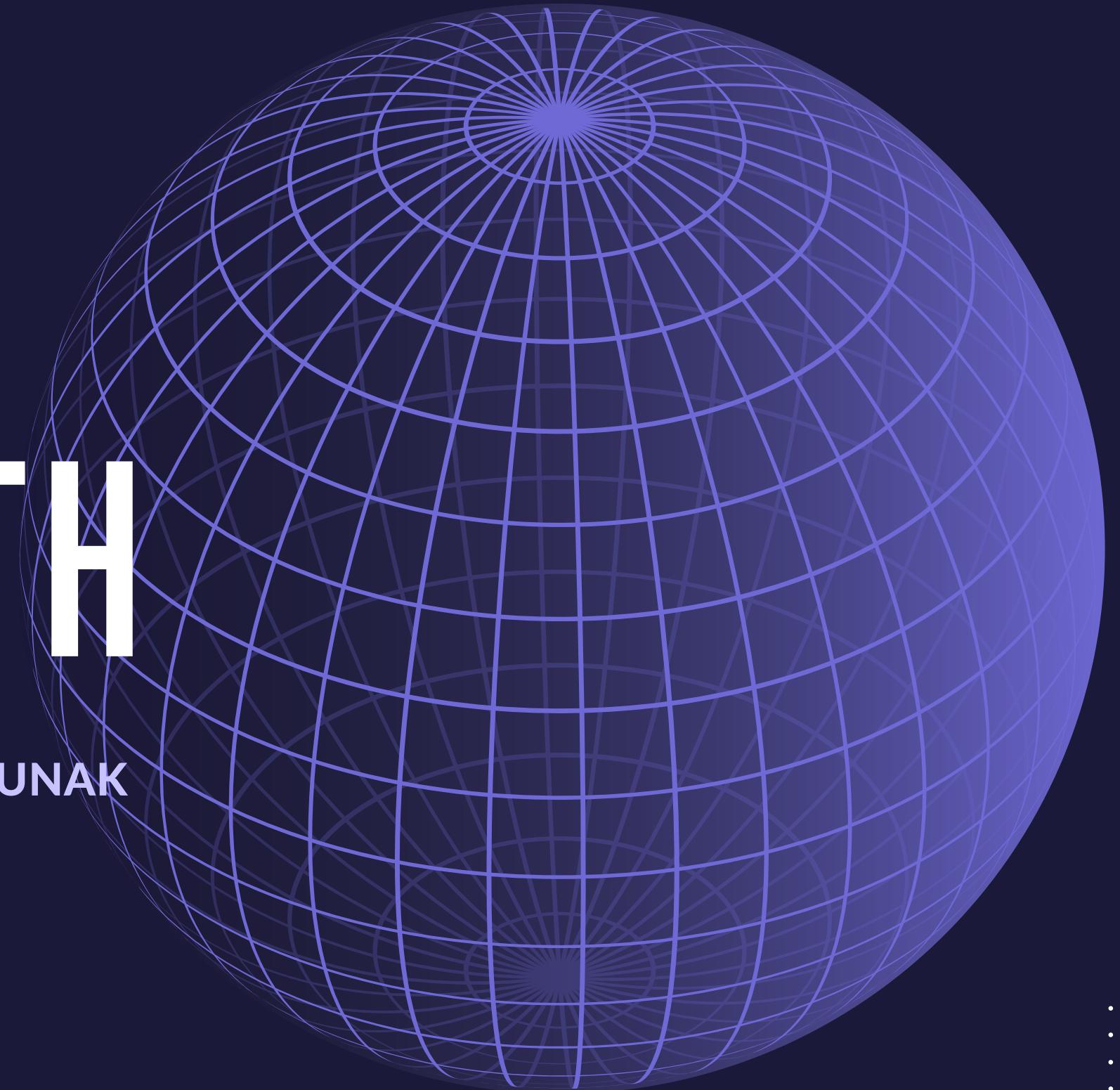




# LEARN PATH

PROJECT UAS REKAYASA PERANGKAT LUNAK





# MEET OUR TEAM



Talitha Khansa Fahira  
12230004



Michiko Tamilla Nathania R  
12230002



Yasser Arafat  
12230009



# WELCOME TO OUR PRESENTATION!

Today's Agenda



- Latar Belakang
- Pengenalan Learn Path
- Penjelasan UML
- Prototype UI/UX
- Coding





# LATAR BELAKANG

Pada era ini, pengelolaan aktivitas belajar menjadi tantangan besar bagi pelajar, mahasiswa, dan pendidik, dengan banyaknya platform yang digunakan secara terpisah. Hal ini menyebabkan ketidakefisienan dalam mengakses materi, mengumpulkan tugas, dan berkolaborasi.

kami mengajukan pengembangan website "Learn Path" sebagai platform terpadu yang dirancang untuk membantu pengguna mengelola aktivitas belajar dengan lebih efisien.





# LEARN PATH

Merupakan Website yang dirancang untuk mengintegrasikan semua fitur yang diperlukan pengguna, menciptakan pengalaman belajar yang lebih terorganisir, efisien, dan mendukung kolaborasi



Fitur yang disediakan oleh Learn Path mempermudah tiap pengguna mengelola kelas satu sama lain. Dalam setiap kelas yang dibuat, terdapat fitur :

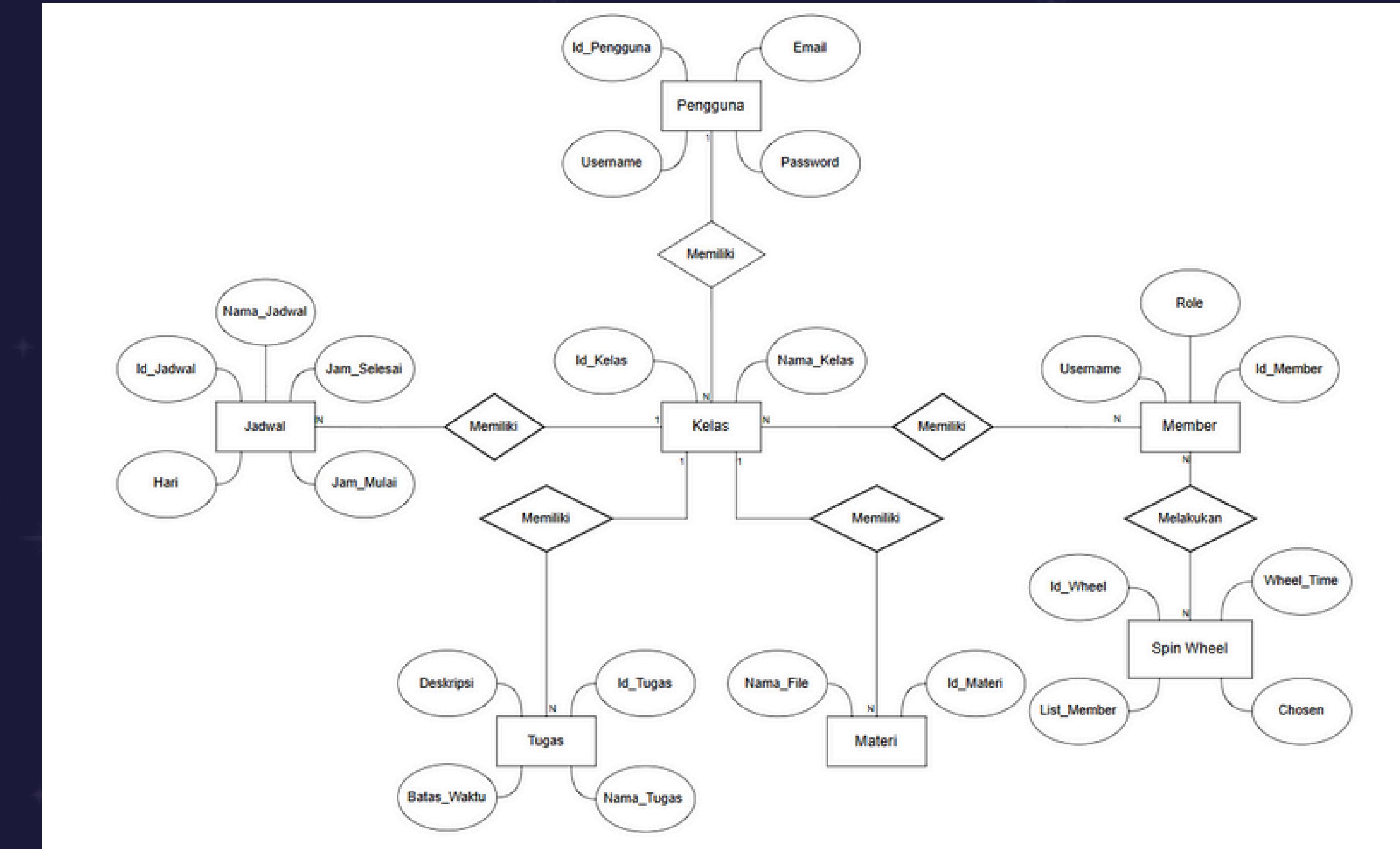
- Tambah Jadwal
- Tambah list tugas
- Tambah materi
- Spin Wheel



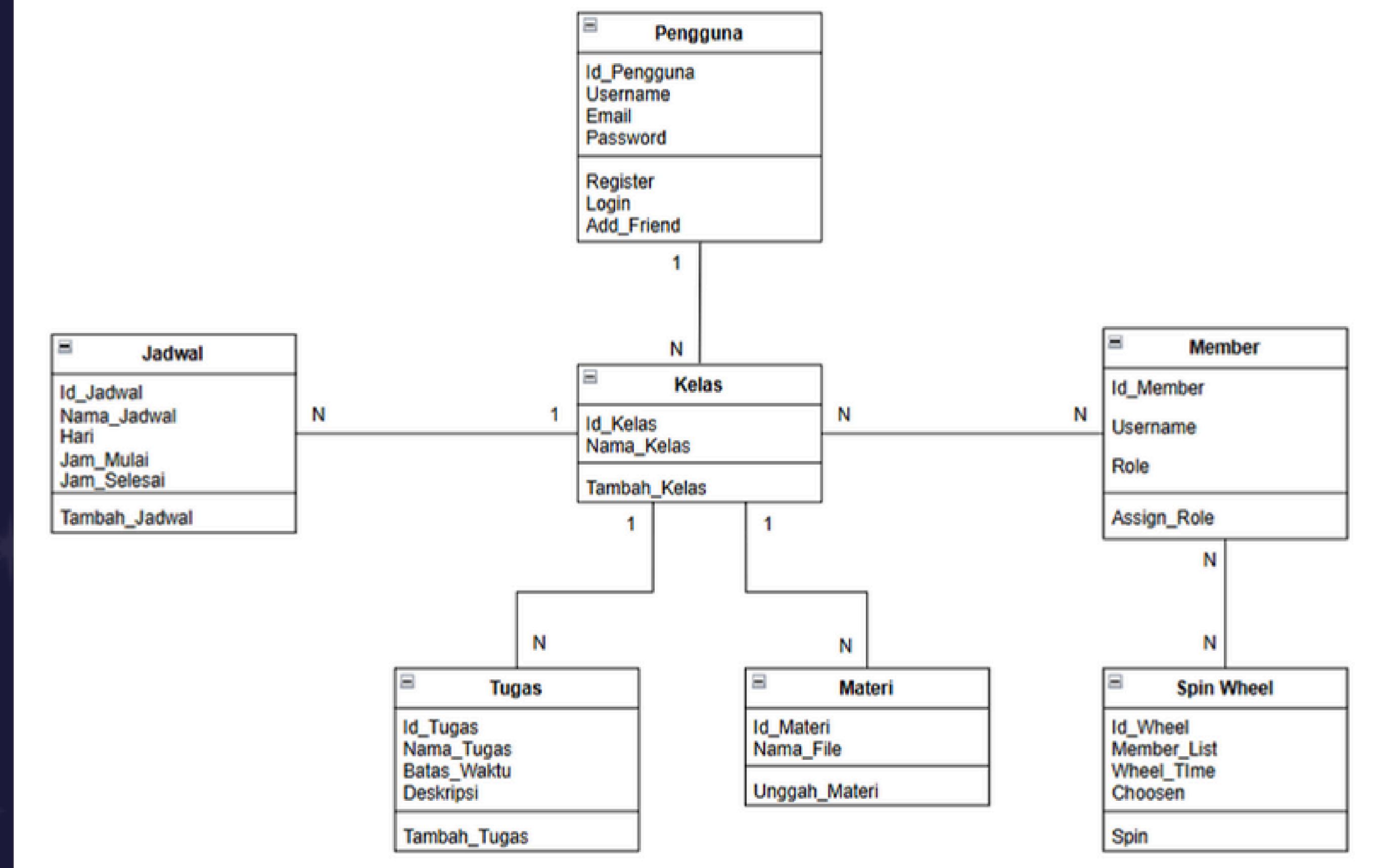
# UML



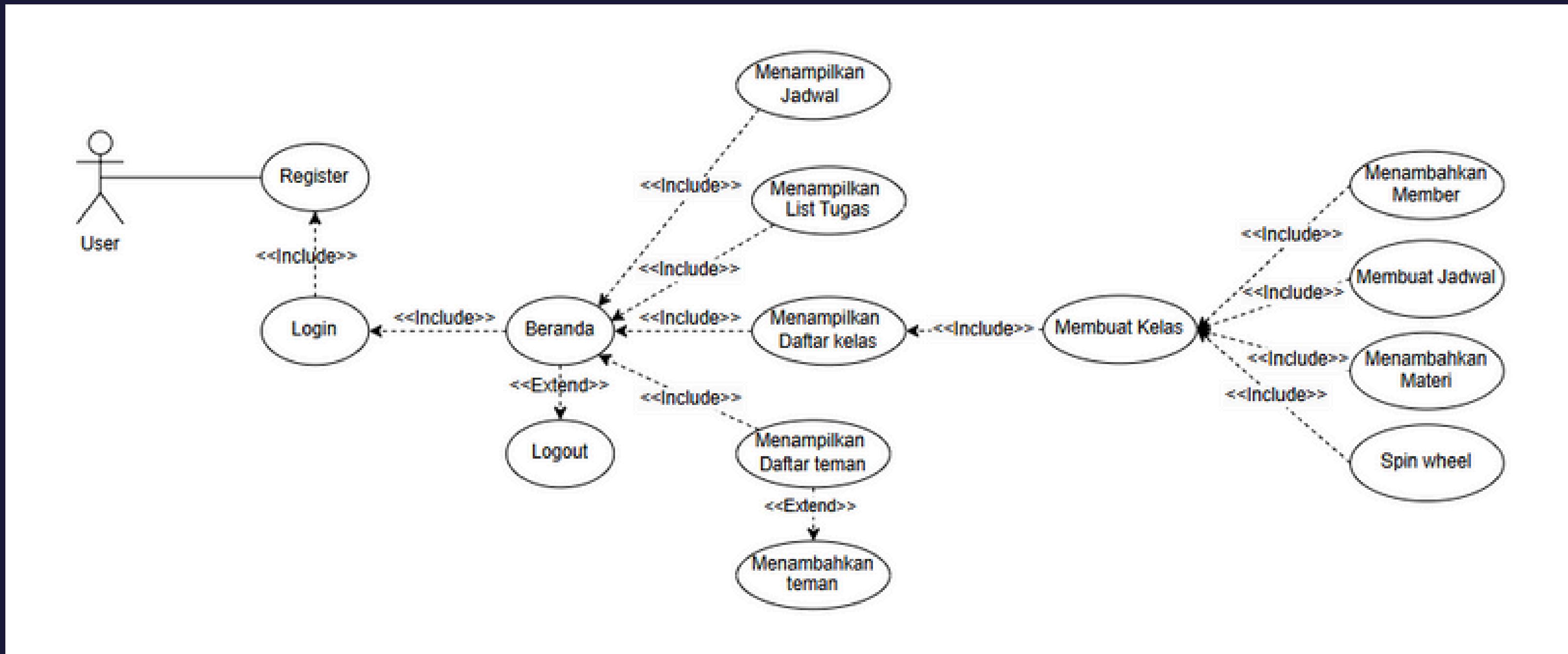
# ERD



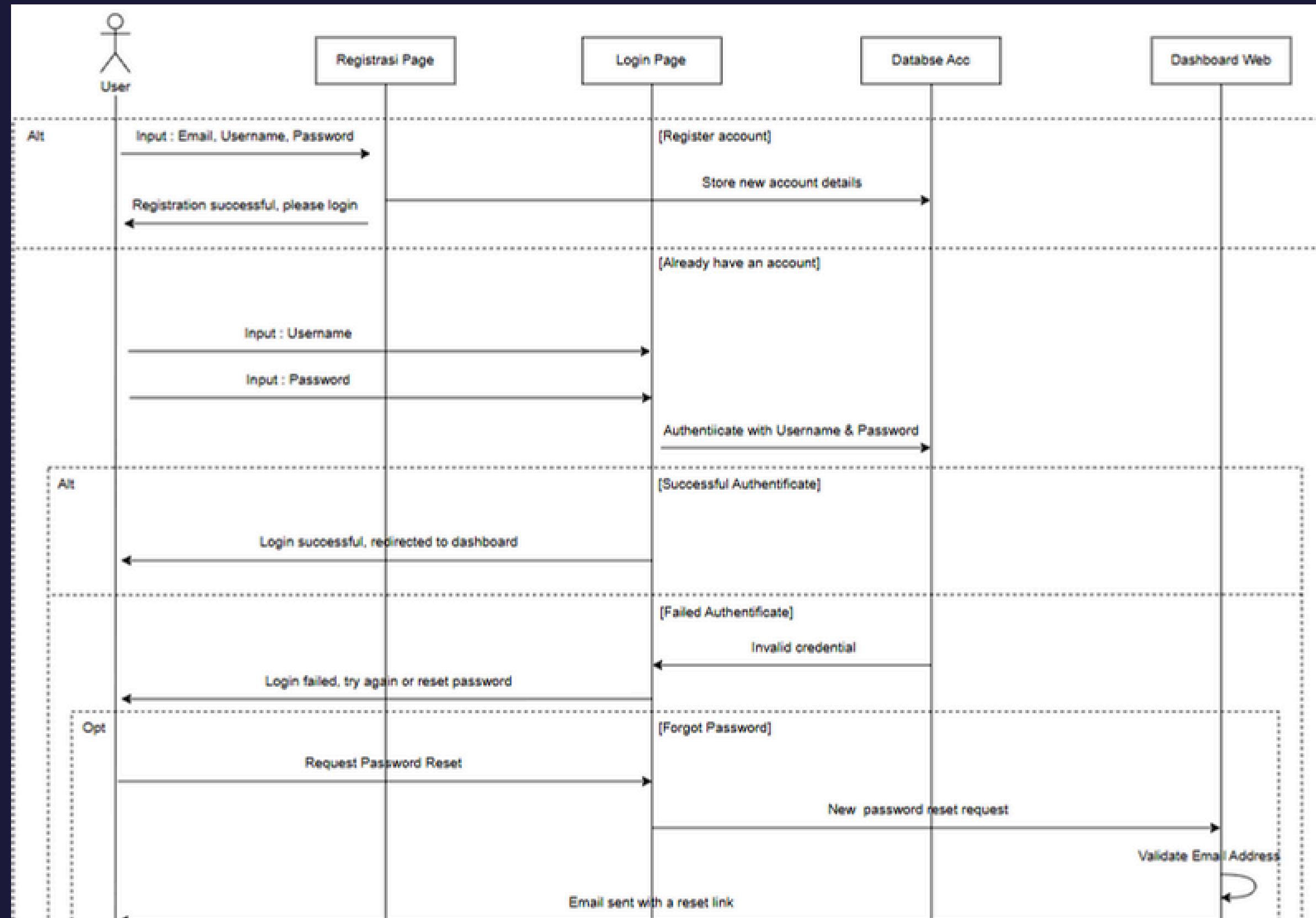
# LRS



# USE CASE

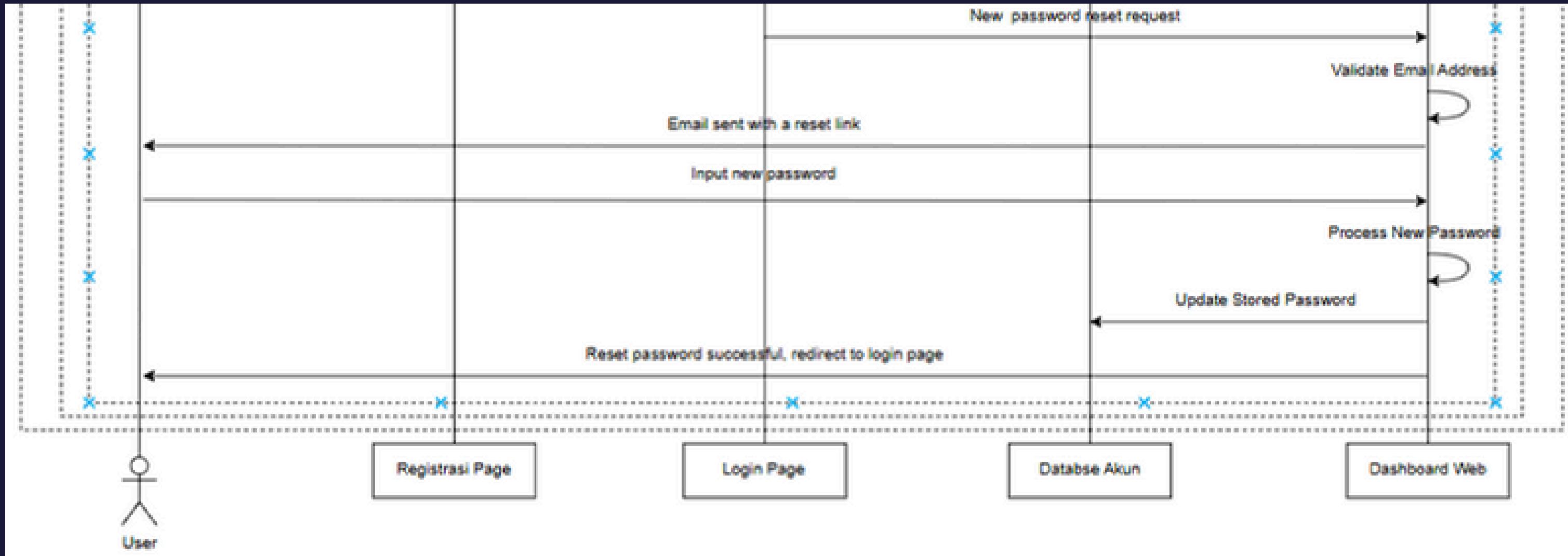


# SEQUENCE DIAGRAM



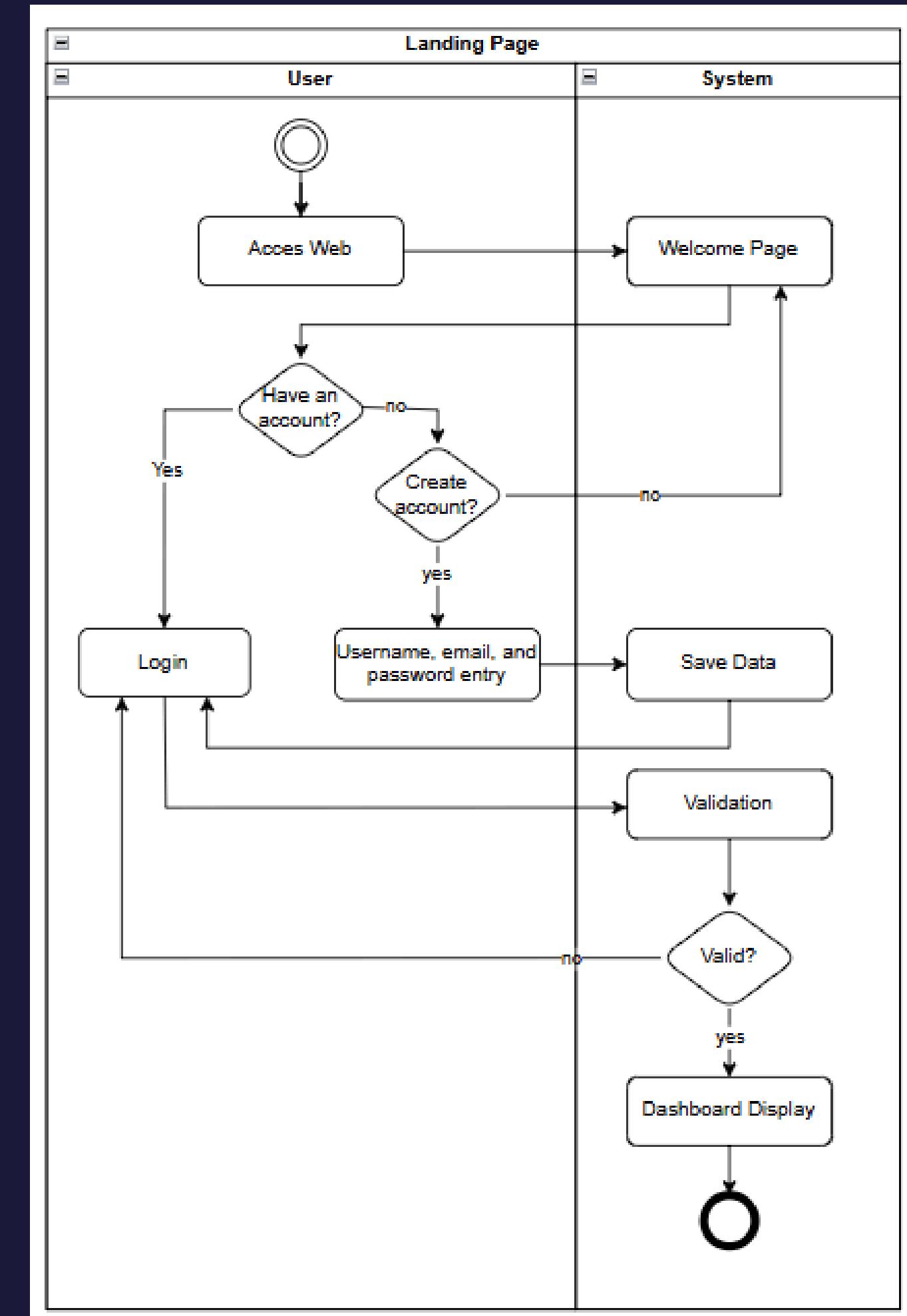


# SEQUENCE DIAGRAM



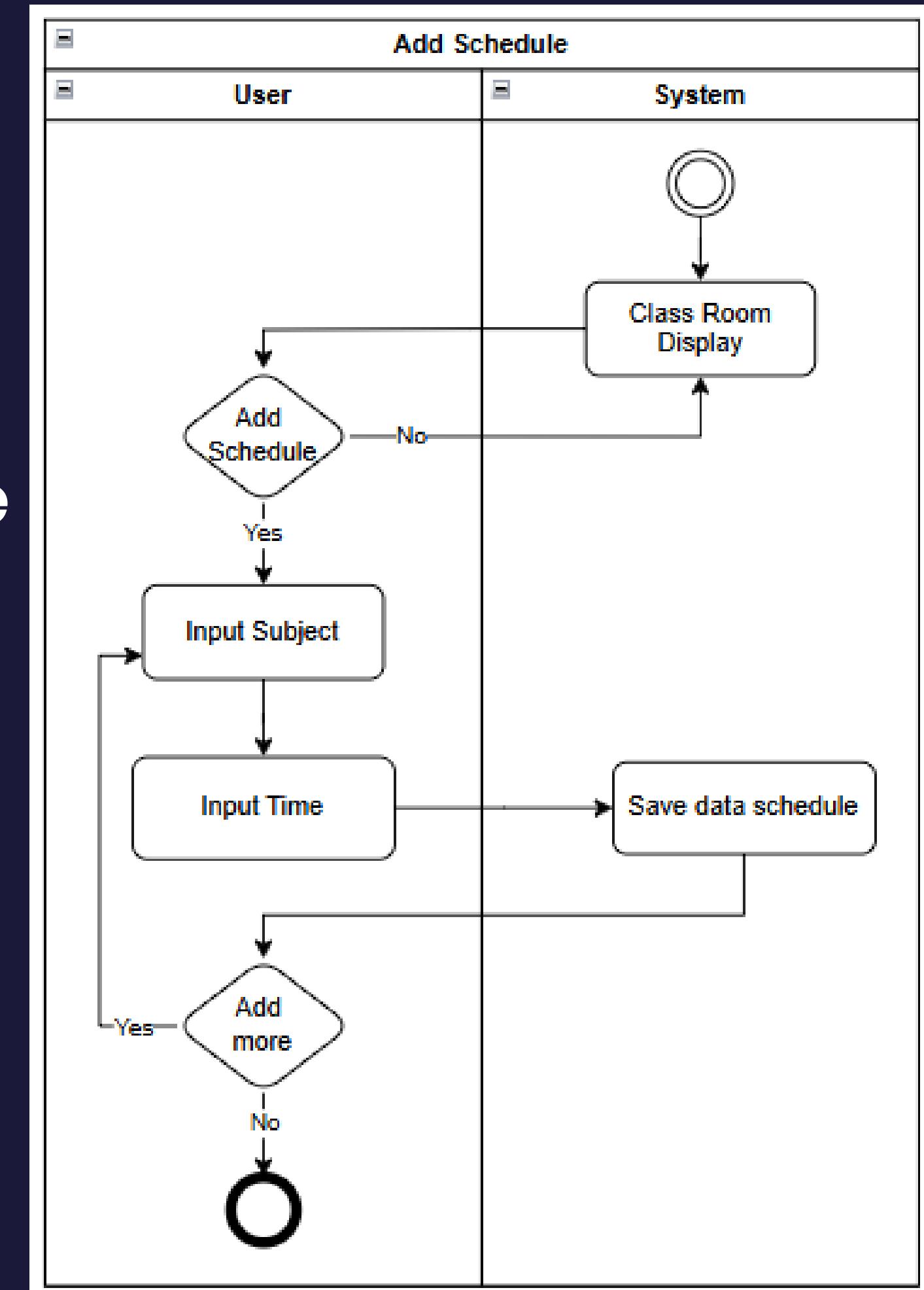
# ACTIVITY DIAGRAM

## Landing Page



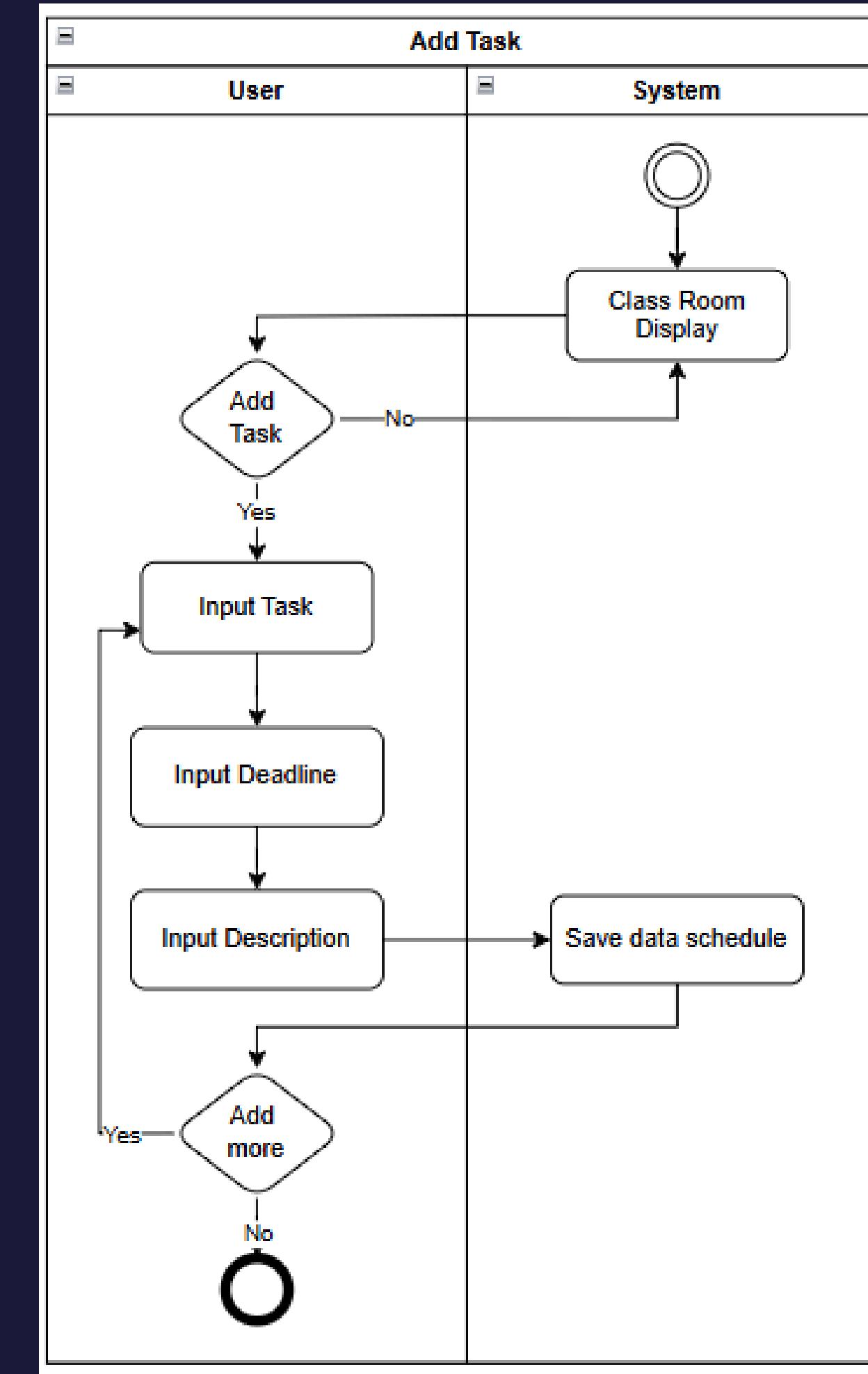
# ACTIVITY DIAGRAM

## -----Add Schedule



# ACTIVITY DIAGRAM

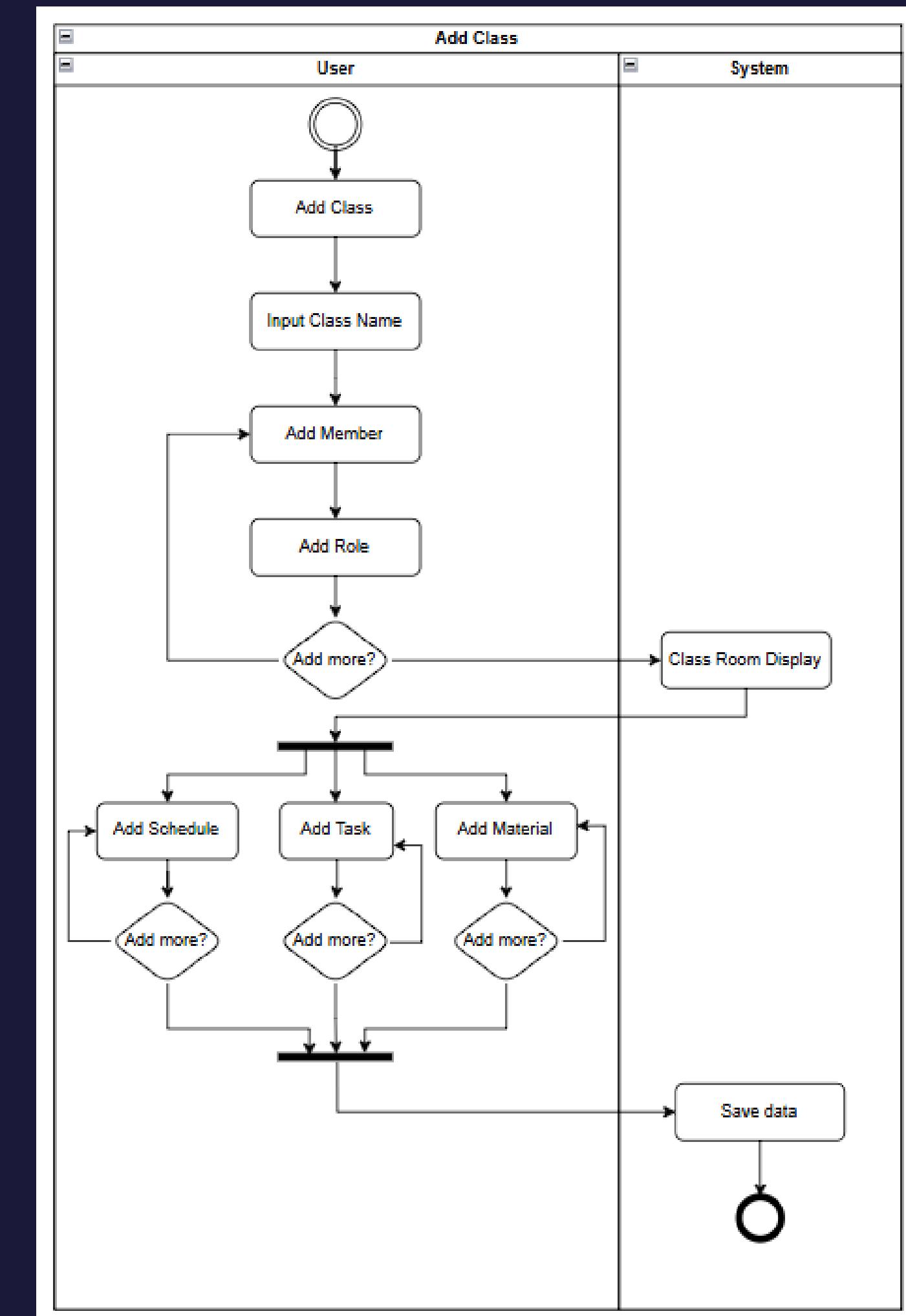
## Add Task



# ACTIVITY DIAGRAM

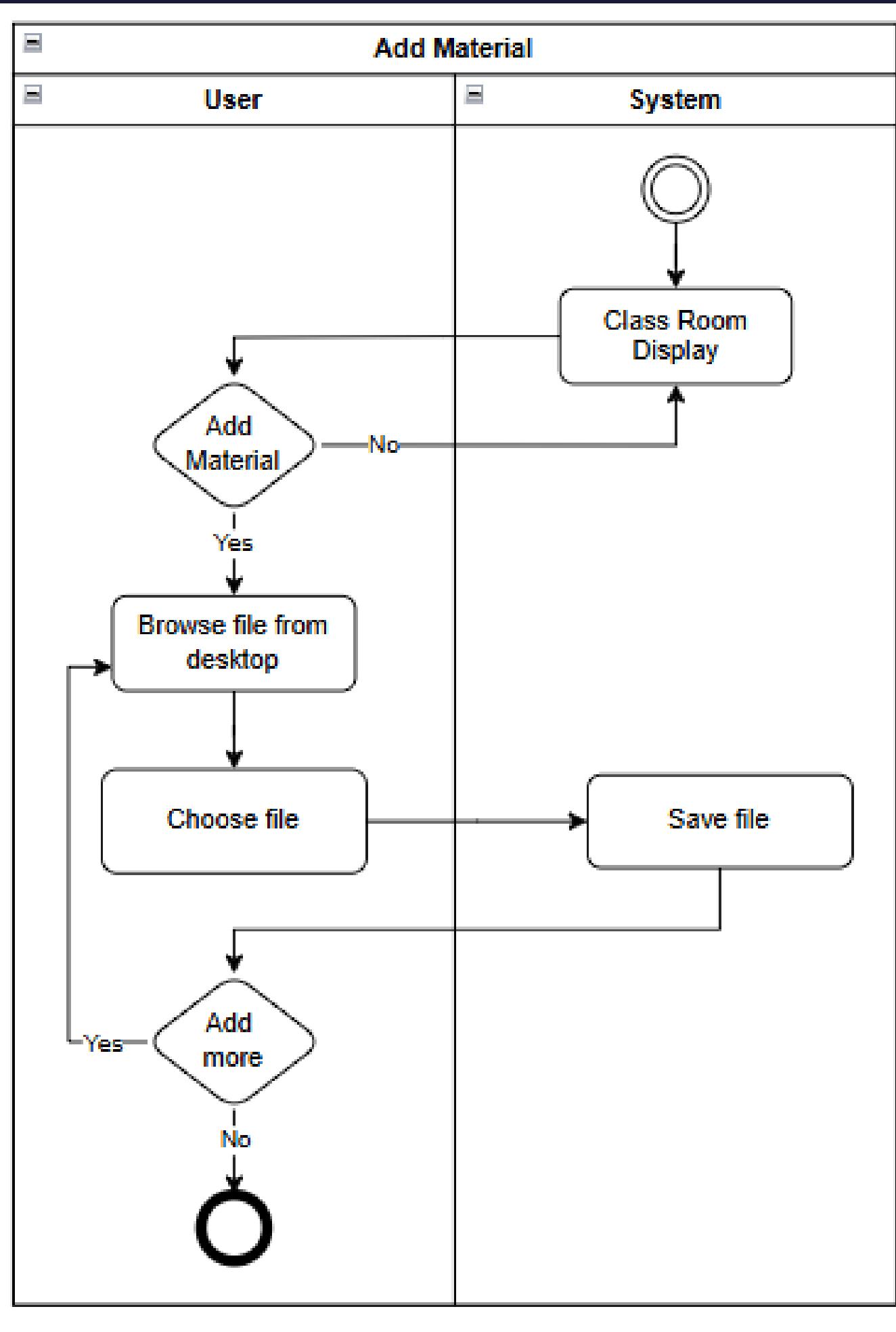
---

## Add Class



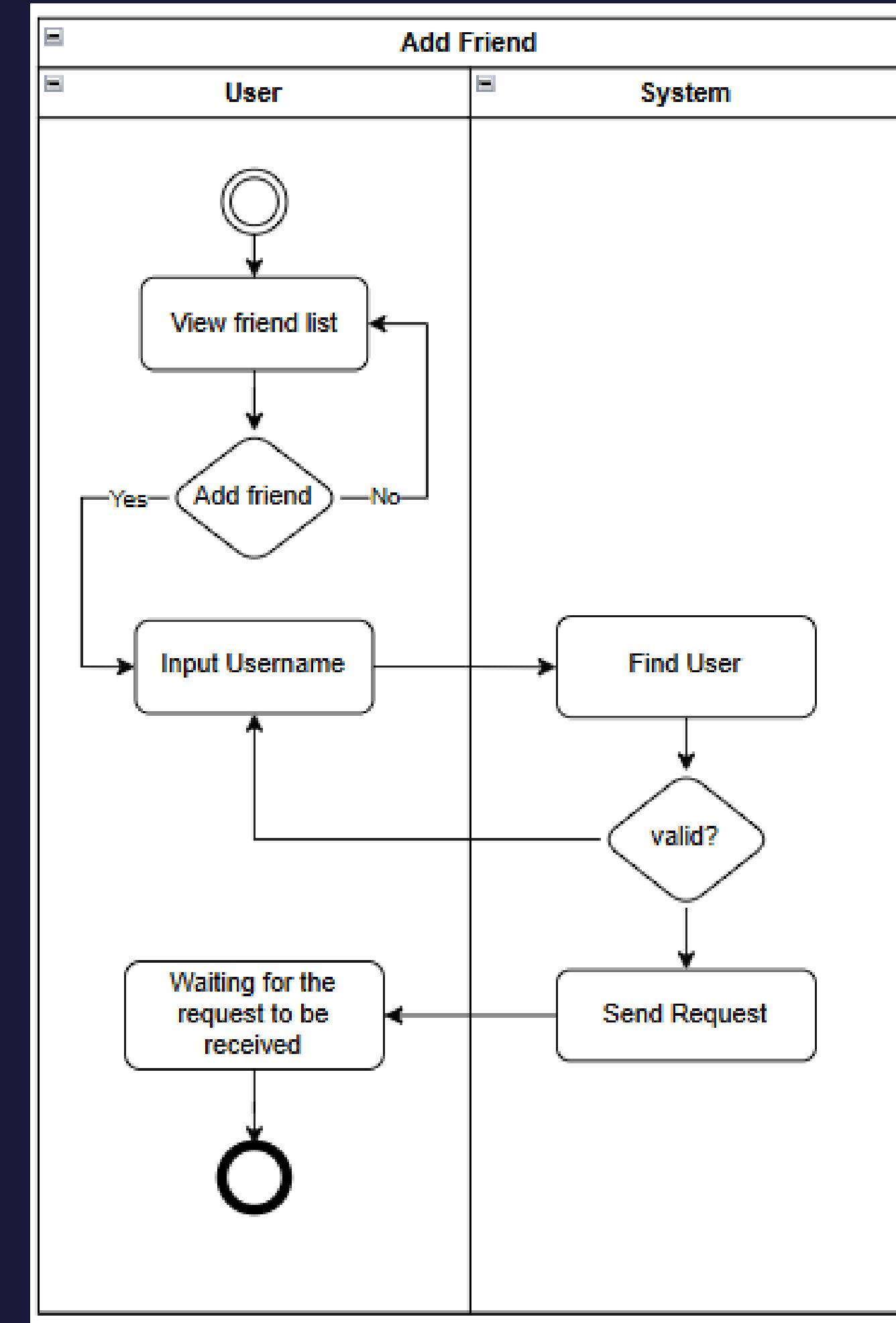
# ACTIVITY DIAGRAM

## Add Material



# ACTIVITY DIAGRAM

## Add Friend



# UI/UX

UI



The landing page for Learn Path features a dark blue header with a globe icon on the left and three horizontal bars on the right. The main title "Learn Path" is prominently displayed in white. Below it is a subtitle "Inspiring Growth, One Connection at a Time." and a "Get Started" button. A central callout box contains the text "Waktu Indonesia Bagian Belajar" and "Dingin Tetapi Tidak Kejam!" above a world map graphic. At the bottom, there is a "Contact Us" section with four empty input fields.

Logo

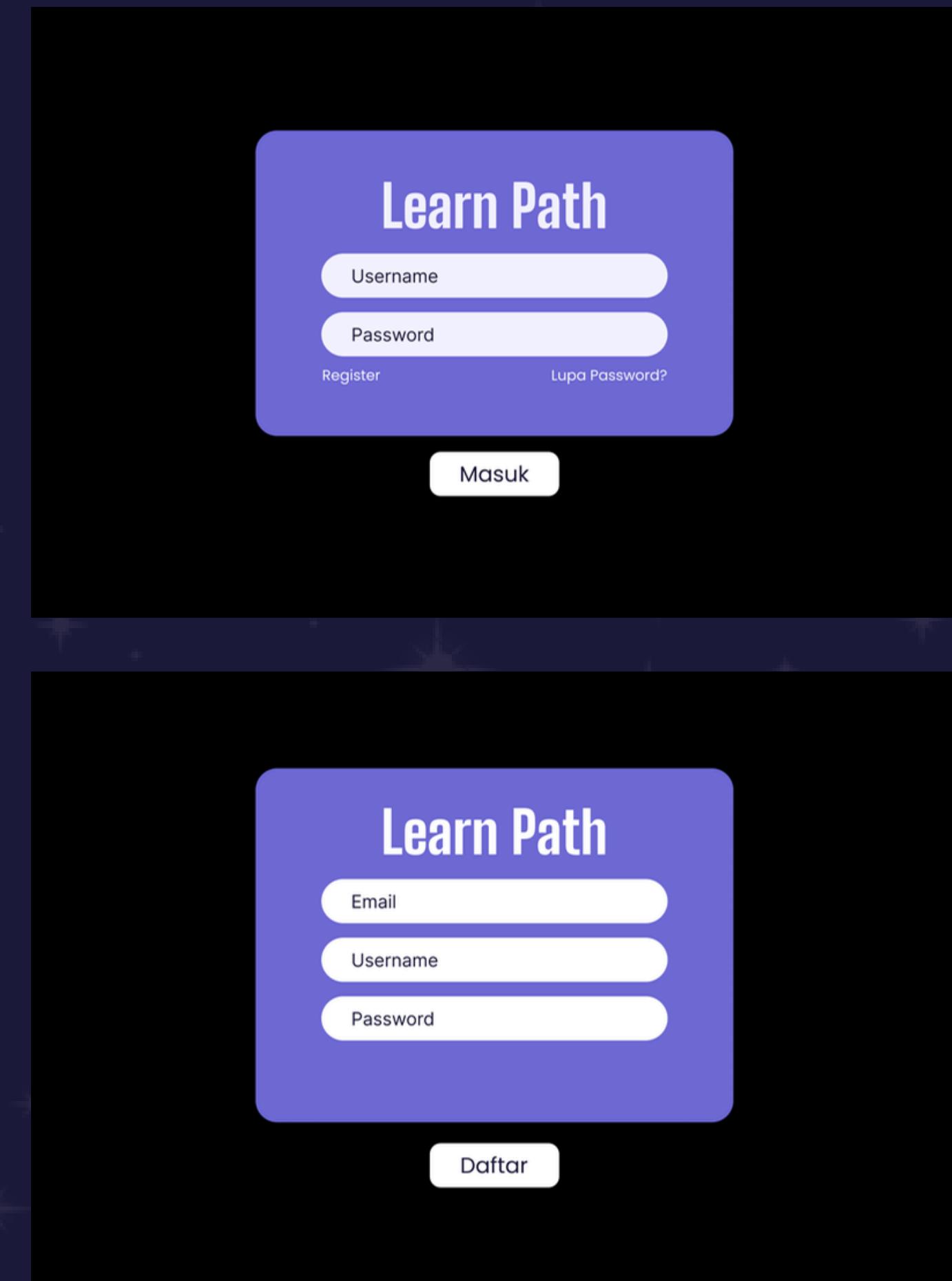
Product About

# Learn Path

Inspiring Growth, One Connection at a Time.

Get Started

Contact Us



The image shows two versions of the Learn Path login screen. The top version is for existing users, featuring fields for "Username" and "Password", and links for "Register" and "Lupa Password?". The bottom version is for new users, featuring fields for "Email", "Username", and "Password", and a "Daftar" button.

## Learn Path

Username

Password

Register Lupa Password?

Masuk

## Learn Path

Email

Username

Password

Daftar

Halo, user!

Schedule

Days	Schedule	Time
Monday	-	00.00-00.00
Tuesday	-	00.00-00.00
Wednesday	-	00.00-00.00
Thursday	-	00.00-00.00

Materi

- Statistika  
Slide Pertemuan 1
- Statistika  
Slide Pertemuan 2
- Keuangan Mikro  
Slide Pertemuan 2
- RPL  
Slide Pertemuan 2

Friends

Friend 1

User

Class

Statistika

+  
Logout

Statistika

Schedule

Days	Subject	Time
Monday	-	00.00-00.00
Tuesday	-	00.00-00.00
Wednesday	-	00.00-00.00
Thursday	-	00.00-00.00

Materi

- Statistika  
Slide Pertemuan 1

Task

Task	Class	Day	00.00
Task 1	Class	Day	00.00
Task 2	Class	Day	00.00
Task 3	Class	Day	00.00

Spin Wheel

Mulai Spin

Member

Leader Mufasa

Role 1 Mudzakar Moortad

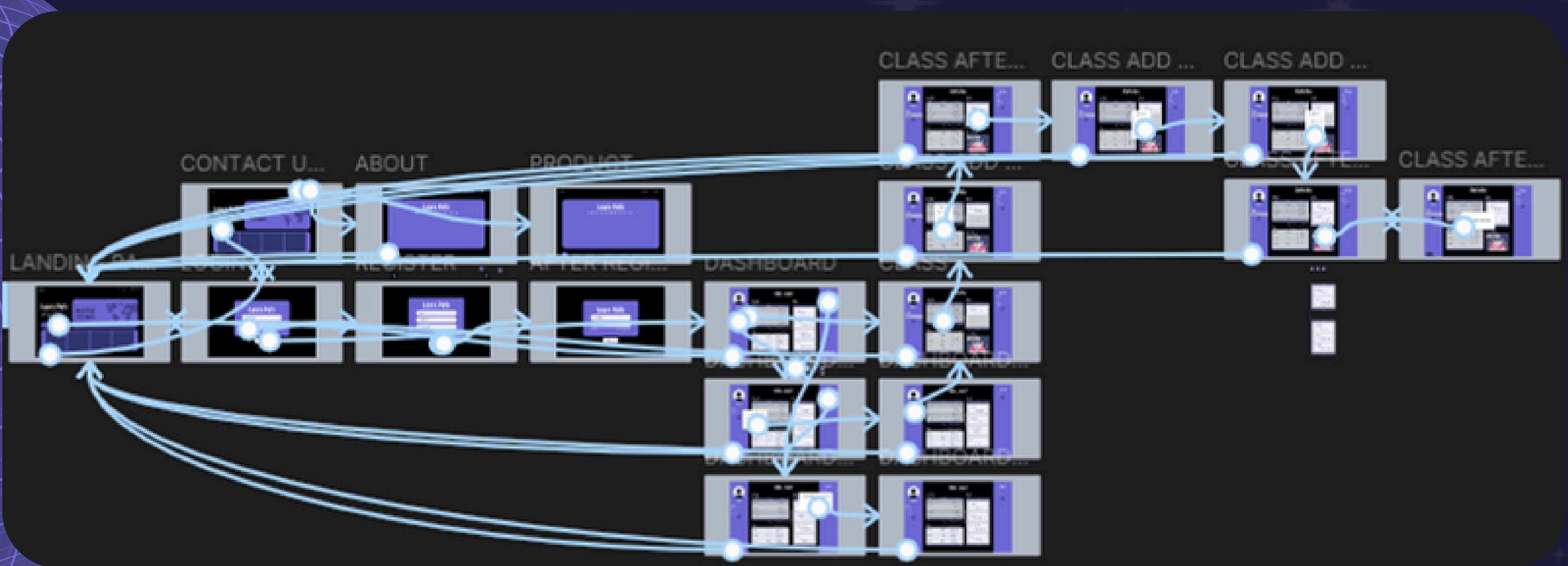
User

Class

Statistika

+  
Logout

UX



[HTTPS://WWW.FIGMA.COM/DESIGN/KQ9MICGNLFUPOMC8DIRMU/UI-RPL-KELOMPOK-2?NODE-ID=0-1&P=F&T=MICILZ1NUI8QFZPA-0](https://www.figma.com/design/KQ9MICGNLFUPOMC8DIRMU/UI-RPL-KELOMPOK-2?node-id=0-1&p=f&t=MICILZ1NUI8QFZPA-0)

# Coding



# PROGRAMMING LANGUAGES & TECHNO. USED

## 1. PHP (Hypertext Preprocessor)

- Handles backend logic such as data processing, user validation, and database interaction.
- Enables the creation of dynamic and interactive systems.

## 2. HTML (HyperText Markup Language)

- Provides the basic structure for web content.
- Organizes elements such as tables, images, forms, and text on the page.

## 3. CSS (Cascading Style Sheets)

- Controls layout, colors, fonts, and responsive design to ensure the website adapts to different devices.

## 4. MySQL

- A relational database management system used to store data.
- Stores information such as users, classes, tasks, schedules, and materials.

## 5. JavaScript

- Allows for features like spin wheels, form validation, and other dynamic functions without page reloads.

⋮  
⋮  
⋮  
⋮

1

# CODING – SQL

```
CREATE TABLE schedule (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT NOT NULL,
    class_id INT NOT NULL,
    day VARCHAR(50) NOT NULL,
    subject VARCHAR(255) NOT NULL,
    time TIME NOT NULL,
    FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE CASCADE,
    FOREIGN KEY (class_id) REFERENCES classes(id) ON DELETE CASCADE
);

CREATE TABLE tasks (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT NOT NULL,
    class_id INT NOT NULL,
    task_name VARCHAR(255) NOT NULL,
    deadline DATE NOT NULL,
    status TINYINT(1) DEFAULT 0,
    FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE CASCADE,
    FOREIGN KEY (class_id) REFERENCES classes(id) ON DELETE CASCADE
);

CREATE TABLE materials (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT NOT NULL,
    class_id INT NOT NULL,
    material_name VARCHAR(255) NOT NULL,
    file_path VARCHAR(255) NOT NULL,
    FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE CASCADE,
    FOREIGN KEY (class_id) REFERENCES classes(id) ON DELETE CASCADE
);
```

```
1 CREATE TABLE users (
2     id INT AUTO_INCREMENT PRIMARY KEY,
3     username VARCHAR(255) NOT NULL,
4     email VARCHAR(255) NOT NULL UNIQUE,
5     password VARCHAR(255) NOT NULL,
6     created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
7 );
8
9 CREATE TABLE classes (
10    id INT AUTO_INCREMENT PRIMARY KEY,
11    user_id INT NOT NULL,
12    class_name VARCHAR(255) NOT NULL,
13    FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE CASCADE
14 );
15
16 CREATE TABLE friends (
17    id INT AUTO_INCREMENT PRIMARY KEY,
18    user_id INT NOT NULL,
19    friend_id INT NOT NULL,
20    friend_name VARCHAR(255) NOT NULL,
21    FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE CASCADE,
22    FOREIGN KEY (friend_id) REFERENCES users(id) ON DELETE CASCADE
23 );
```

```
CREATE TABLE class_members (
    id INT AUTO_INCREMENT PRIMARY KEY,
    class_id INT NOT NULL,
    user_id INT NOT NULL,
    role ENUM('leader', 'vice_leader', 'treasurer', 'secretary', 'member') DEFAULT 'member',
    FOREIGN KEY (class_id) REFERENCES classes(id) ON DELETE CASCADE,
    FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE CASCADE
);

CREATE TABLE `password_resets` (
    `id` int(11) NOT NULL,
    `email` varchar(255) NOT NULL,
    `token` varchar(255) NOT NULL,
    `created_at` timestamp NOT NULL DEFAULT current_timestamp(),
    ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

# == CODING --Register HTML

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Register</title>
7      <link href="https://fonts.googleapis.com/css2?family=Big+Shoulders+Display:wght@400;700&display=swap" rel="stylesheet">
8      <link rel="stylesheet" href="css/register.css">
9  </head>
10 <body>
11     <div class="login-container">
12         <div class="login-box">
13             <h1 class="title">Registration</h1>
14             <form action="process_register.php" method="POST">
15                 <div class="input-group">
16                     <input type="text" name="username" placeholder="Username" required>
17                 </div>
18                 <div class="input-group">
19                     <input type="email" name="email" placeholder="Email" required>
20                 </div>
21                 <div class="input-group">
22                     <input type="password" name="password" placeholder="Password" required>
23                 </div>
24                 <div class="input-group">
25                     <input type="password" name="confirm_password" placeholder="Confirm Password" required>
26                 </div>
27                 <button type="submit" class="login-button">Register</button>
28             </form>
29             <div class="links">
30                 <a href="login.html">Sudah punya akun? Login</a>
31             </div>
32         </div>
33     </div>
34 </body>
35 </html>
36 . . . . .
```

# CODING

## --Process Register PHP

```
1 <?php
2 include('config.php');
3
4 if ($_SERVER["REQUEST_METHOD"] == "POST") {
5     $username = trim($_POST['username']);
6     $email = trim($_POST['email']);
7     $password = $_POST['password'];
8     $confirm_password = $_POST['confirm_password'];
9
10    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
11        die("Email tidak valid!");
12    }
13
14    if ($password !== $confirm_password) {
15        die("Password dan konfirmasi password tidak cocok!");
16    }
17
18    if (strlen($password) < 8) {
19        die("Password harus minimal 8 karakter!");
20    }
21
22    $sql_check = "SELECT id FROM users WHERE email = ?";
23    $stmt_check = $conn->prepare($sql_check);
24    $stmt_check->bind_param("s", $email);
25    $stmt_check->execute();
26    $stmt_check->store_result();
27
28    if ($stmt_check->num_rows > 0) {
29        die("Email sudah terdaftar! Silakan gunakan email lain.");
30    }
31    $stmt_check->close();
32
33    $hashed_password = password_hash($password, PASSWORD_DEFAULT);
34
35    $sql = "INSERT INTO users (username, email, password) VALUES (?, ?, ?)";
36    $stmt = $conn->prepare($sql);
37    $stmt->bind_param("sss", $username, $email, $hashed_password);
38
39    if ($stmt->execute()) {
40        header("Location: login.html?status=success");
41        exit();
42    } else {
43        echo "Terjadi kesalahan saat mendaftar!";
44    }
45    $stmt->close();
46}
47 ?>
```

# == CODING --Login HTML

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Login</title>
7      <link href="https://fonts.googleapis.com/css2?family=Big+Shoulders+Display:wght@400;700&display=swap" rel="stylesheet">
8      <link rel="stylesheet" href="css/login.css">
9  </head>
10 <body>
11     <div class="login-container">
12         <div class="login-box">
13             <h1 class="title">Login</h1>
14             <form action="process_login.php" method="POST">
15                 <div class="input-group">
16                     <input type="text" name="username" placeholder="Username" required>
17                 </div>
18                 <div class="input-group">
19                     <input type="password" name="password" placeholder="Password" required>
20                 </div>
21                 <div class="links">
22                     <a href="register.html">Register</a>
23                     <a href="forgotpassword.html">Lupa Password?</a>
24                 </div>
25                 <button type="submit" class="login-button">Masuk</button>
26             <div class="back">
27                 <a href="index.html">Kembali</a>
28             </div>
29         </form>
30     </div>
31 </div>
32 </body>
33 </html>
```

.....  
.....  
.....  
.....  
.....

# CODING ---Process Login

```
1  <?php
2  session_start();
3  include('config.php');
4
5  if ($_SERVER["REQUEST_METHOD"] == "POST") {
6      // Mengambil input dari form login
7      $username = $_POST['username'];
8      $password = $_POST['password'];
9
10     // Query untuk mencari pengguna berdasarkan username
11     $sql = "SELECT * FROM users WHERE username = ?";
12     $stmt = $conn->prepare($sql);
13     $stmt->bind_param("s", $username);
14     $stmt->execute();
15     $result = $stmt->get_result();
16
17     // Memeriksa apakah username ditemukan
18     if ($result->num_rows > 0) {
19         $user = $result->fetch_assoc();
20
21         // Memeriksa password yang dimasukkan dengan password yang tersimpan di database
22         if (password_verify($password, $user['password'])) {
23             // Jika password valid, buat sesi dan arahkan ke dashboard
24             $_SESSION['user_id'] = $user['id'];
25             $_SESSION['username'] = $user['username'];
26             header("Location: dashboard.php");
27             exit();
28         } else {
29             // Jika password salah
30             echo "Username atau Password salah!";
31         }
32     } else {
33         echo "Pengguna tidak ditemukan!";
34     }
35 }
36 ?>
37
```

# == CODING --Forgot Password

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Lupa Password</title>
7      <link rel="stylesheet" href="css/style2.css">
8      <link href="https://fonts.googleapis.com/css2?family=Irish+Grover&display=swap" rel="stylesheet">
9      <style>
10         .title {
11             font-family: 'Irish Grover', serif; /* Gunakan font Irish Grover */
12             font-size: 3em;
13             color: #ffffff; /* Warna teks */
14             text-align: center; /* Teks di tengah */
15             margin-bottom: 20px;
16         }
17     </style>
18 </head>
19 <body>
20     <div class="login-container">
21         <div class="login-box">
22             <a href="index.html" class="back-logo">
23                 
24             </a>
25             <h1 class="title">Lupa Password</h1>
26             <form action="process_forgot_password.php" method="POST">
27                 <div class="input-group">
28                     <input type="email" name="email" placeholder="Masukkan Email Anda" required>
29                 </div>
30                 <button type="submit" class="login-button">Kirim Email Reset</button>
31             </form>
32             <div class="links">
33                 <a href="login.html">Kembali ke Login</a>
34             </div>
35         </div>
36     </div>
37 </body>
38 </html>
```

. . . . .  
. . . . .  
. . . . .  
. . . . .  
. . . . .

# == CODING--Process Forgot Password

```
1  <?php
2  include 'config.php';
3
4  if ($_SERVER["REQUEST_METHOD"] === "POST") {
5      // Ambil email dari form
6      $email = trim($_POST['email']);
7
8      // Validasi email
9      if (empty($email)) {
10          die("Please enter your email address.");
11      }
12
13     // Periksa apakah email ada di database
14     try {
15         $stmt = $conn->prepare("SELECT * FROM users WHERE email = :email");
16         $stmt->execute(['email' => $email]);
17         $user = $stmt->fetch();
18
19         if ($user) {
20             // Logika pengiriman link reset password (disimulasikan)
21             echo "Password reset link has been sent to your email.";
22             // Pada kenyataannya, Anda bisa mengirim email dengan library seperti PHPMailer.
23         } else {
24             echo "Error: Email not found.";
25         }
26     } catch (PDOException $e) {
27         die("Error: " . $e->getMessage());
28     }
29 }
30 ?>
31
```

. . . . .



**THANK YOU**