

|  |
| --- |
| **WEB APPLICATION**  **Institute Of Fine Arts** |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Developed by**  **Members:**   |  |  |  | | --- | --- | --- | | **No.** | **Student Name** | **Student ID** | | 1 | Adnan | Student1553724 | | 2 | Daniyal Qasim | Student1471406 | | 3 | Syed Hunain Haider Zaidi | Student1434075 | | 4 | Muhammad Ali Khan | Student1459107 | | 5 | Abdul Latif | Student1471801 | | 6 | Aliza Rashid | Student1459145 | | 7 | Aman Fatima | Student1458756 |  * **Class No : PR2-202305E +03E TTS** * **Start Date: 25/10/2024** * **End Date: 20/11/2024** * **Name of the Coordinator: Sir Asad Ali** * **Date of Submission: 25/11/2024** | |
| **System Summary:**  **ANALYSIS**  To build a web-based platform for managing competitions, submissions, evaluations, and exhibitions efficiently, replacing the manual pen-and-paper process currently in use.  **Key Modules:**   1. Administrator Module:  * Manage details of staff, students, competitions, awards, and exhibitions. * CRUD (Create, Read, Update, Delete) operations for all relevant entities. * Generate reports on student participation and competition outcomes.  1. Staff Module:  * Oversee student records, competition details, and submissions. * Evaluate paintings based on creativity and accompanying descriptions, assigning marks (e.g., "Best," "Better"). * Manage exhibition paintings, including pricing, sale status, and customer details.  1. Student Module:  * View competition details, awards, and feedback. * Upload, edit, or delete paintings, poems, or descriptions before competition deadlines. * Monitor exhibition status, including painting sales and payments.  1. Manager Module:  * View comprehensive reports on students, competitions, awards, and exhibitions. * Maintain exhibition details and oversee staff and student activities.  1. Home Page:  * Showcase upcoming and ongoing competitions. * Highlight award winners and statistics (e.g., total submissions).   **Functional Requirements:**  **For Students:**   * Competitions**:**   + - Submit designs (JPEG) with descriptions (e.g., poems, quotations).     - View competition details (conditions, awards, deadlines)     - Update or delete submissions before the deadline. * Feedback**:**   + - Access evaluations and remarks on their submissions     - View highlighted works (e.g., "Best" paintings) and areas for improvement. * Exhibitions**:**   + - View exhibition details, painting status, and sales history.   **For Staff:**   * Student Records:   + - Manage and retrieve student and competition data. * Evaluations:   + - Assign marks and provide remarks for submissions.     - Disqualify late submissions. * Exhibitions**:**   + - Select top paintings for exhibitions and manage their status.   **For Manager:**   * Manage:   + - Staff and student details.     - Competition and exhibition schedules.     - Awards and reports generation. * Oversee:   + - Student submissions and evaluations.     - Exhibition details, including sales and customer data.   **TECHNOLOGIES TO BE USED**    **Technologies** -Frontend: HTML5,CSS3, Angular 18, JavaScript  -Backend: API, ASP .net  -Database: SQL Server  **Tools** -Development Environment: Visual studio Code, Visual Studio  -Data Management: SQL Server |

|  |
| --- |
| **CLASS DIAGRAM** |

|  |
| --- |
|  |

|  |
| --- |
| **DATABASE DESIGN** |

|  |
| --- |
| **Database Name: institute**  **Number of tables:19**  **Tables Names:**  C:\Users\kk\Downloads\DaimgaramPNG.PNG  **Schematic Diagram Of Database:**  Schematic diagram of the database depicting the primary and foreign key  relationships. |

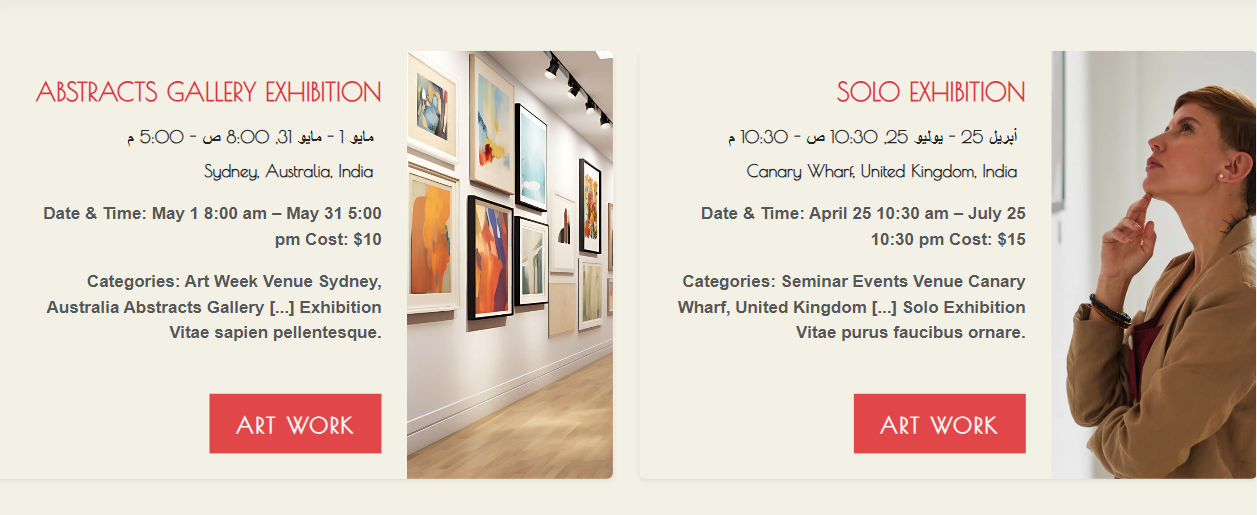
|  |
| --- |
| **TABLE DESIGN** |

|  |
| --- |
| **Database Name: institute**    **C:\Users\kk\Downloads\user.png1.users:**    **2.allCourses:**  **C:\Users\kk\Downloads\allcourese.png**  **C:\Users\kk\Downloads\Award.png3.Awards:**  **4.Competitions:**  **C:\Users\kk\Downloads\competition.png**  **C:\Users\kk\Downloads\regcom.png5.competitionstudent:**  **6.contact:**  **C:\Users\kk\Downloads\contact.png**    **7.exhibitionRegistration:**  **C:\Users\kk\Downloads\ExREg.png**    **8.exhibitions**  **C:\Users\kk\Downloads\Exhibition.png**  **9.Markings:**  **C:\Users\kk\Downloads\Marking.png**    **10.paintingsubmit:**  **C:\Users\kk\Downloads\paintingsubmit .png** |

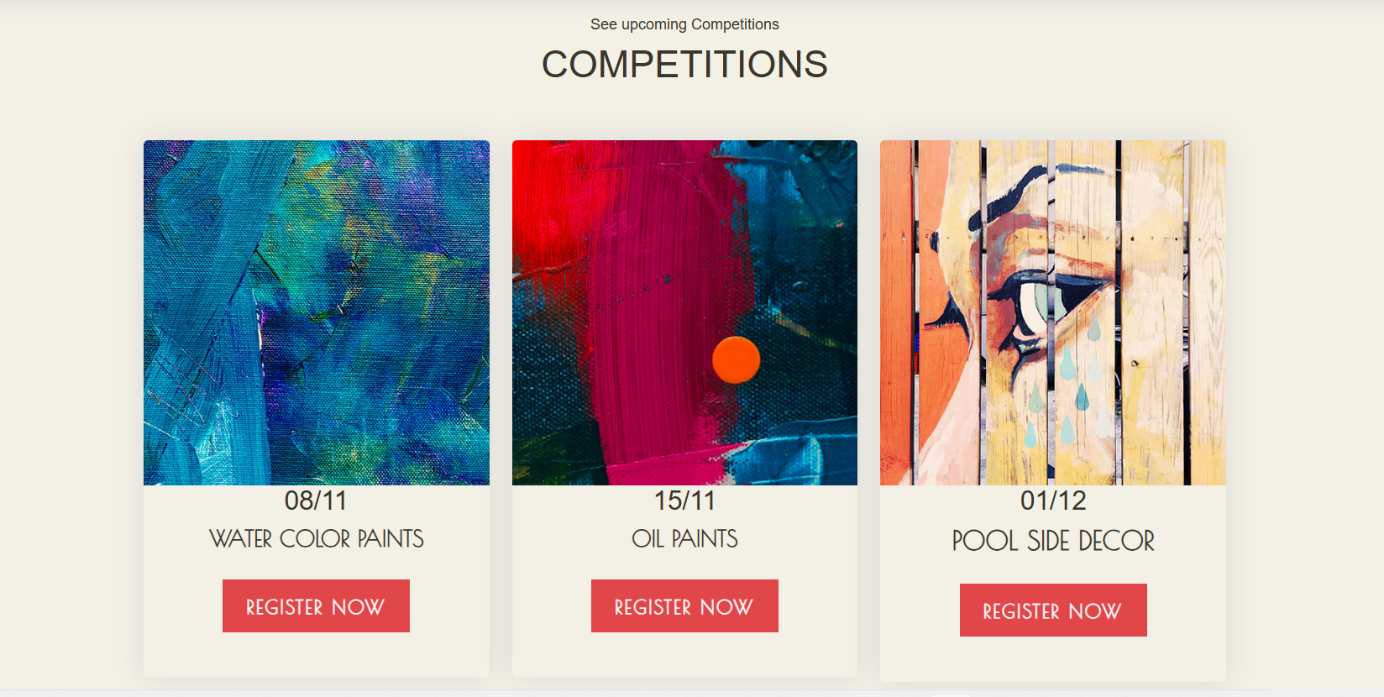
|  |
| --- |
| **SCREENSHOTS** |

|  |  |
| --- | --- |
| **Home Page**  **Navbar/header**    **Ongoing Competitions**    **Previous Competitions**    **Upcoming Competitions**    **Total Students Registered In Competition** |  |

**Exhibitions**

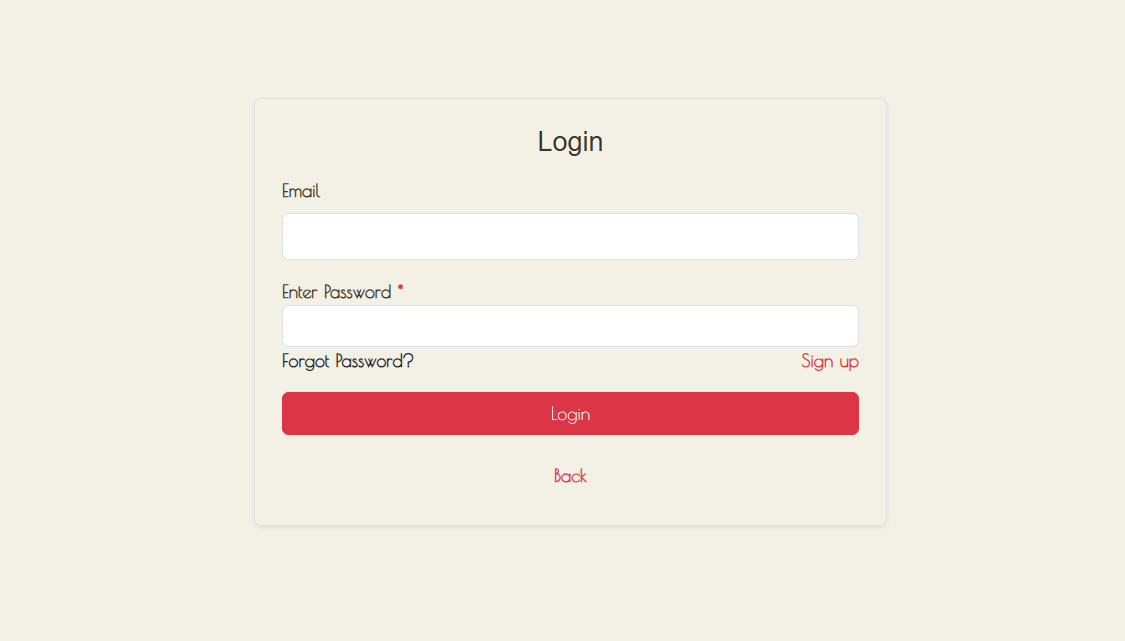
****

**Competitions**

****

**Registration Form**

**Login Form**

****

**Admin Dashboard**

**Staff Dashboard**

**Manager Dashboard**

|  |
| --- |
| **DATA FLOW DIAGRAM** |

|  |
| --- |
| C:\Users\kk\Downloads\Process Mapping Whiteboard in Magenta Yellow Bright Blue Vibrant Outlined Style.png  **Data Flow Diagram representing the control flow of Project Case Study.** |

|  |
| --- |
| **CONFIGURATION** |

|  |
| --- |
| **Software & Technology Used:** |
| **Software: - Eclipse - Mysql - Scene Builder, gitbash** |
| **Technology: - MVC model - AWS - Momo e-wallet - Socket - JavaFx, Java - Github** |

|  |
| --- |
| **PROJECT TIMELINES** |

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Task to be performed** | **Time** |
| 1. **Database design** | **- Gather relevant objects included in, define their relationships, data types - Implement Amazon RDB to store shared database** | **- 3 days** |
| 1. **MVC source code structure, store project on git** | **- Organize source code by models, controllers and views besides helper classes and referent libraries - Git management** | **-1 day** |
| 1. **Customer-side features, layouts** | **- Design layouts - Load products to customer view by categories / subcategories; search products by name - View and custom food and drinks by size, sugar and ice amount; set quantity and leave note before add to order - Add and place order, update order, send and receive notifications to servers, chefs using socket - Process payment by cash and Momo, check for discount, deposit amount to exclude from total order amount, add tip to order - QR code screening for Momo e-wallet payment - Review order after finishing order payment - Gather parts of project into unified flow** | **- 14 days** |
| 1. **Administrator-side features** | **- Log in, log out management() - User management() - Category management() - Attribute management() - Food & drinks management()  - Order management() - Reservation management() - Table management() - Discount management()  - Roles & permissions()** |  |

|  |
| --- |
| **INDIVIDUAL PROJECT SCHEDULE** |

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Activity Description** | **Responsibility** |
| **1.** | **- Gather relevant objects included in, define their relationships, data types - Implement Amazon RDB to store shared database** | **Nguyen Huu Quynh Nhu** |
|  | **- Organize source code by models, controllers and views besides helper classes and referent libraries - Git management** | **Nguyen Huu Quynh Nhu** |
|  | **- Design layouts - Load products to customer view by categories / subcategories; search products by name - View and custom food and drinks by size, sugar and ice amount; set quantity and leave note before add to order - Add and place order, update order, send and receive notifications to servers, chefs using socket - Process payment by cash and Momo, check for discount, deposit amount to exclude from total order amount, add tip to order - QR code screening for Momo e-wallet payment - Review order after finishing order payment - Gather parts of project into unified flow** | **Nguyen Huu Quynh Nhu** |
|  | **-Build layouts (****users, assign permission, role, tables, reservation, schedule of tables, schedule of reservations, discount, layout created,update of the above layout listed)** | **Vo Huynh Ngoc Tram** |
|  | **-Build functions (create, update, detele, view data) of users, assign permission, role, tables, reservation, schedule of tables, schedule of reservations, discount** | **Vo Huynh Ngoc Tram** |
|  | **-** **Encrypt with Bcrypt**  **- Remember log in when checked in the checkbox**  **-Role permissions for all users**  **-Super admin can’t delete your roles**  **-Users and super admin can’t update your status in this application**  **-Manager can’t see super admin on users view**  **-Server,manager,admin can set code for tablet to know which orders are sent from the table**  **-Remember table code when server, manager, admin has set up**  **- Users can rely on the schedule to advise customers who want to book a table, while creating it, if you enter enough information, it will load which tables are available in that time for users to choose**  **- Users can set the time customer can cancel the reservation and specify the reservation set time(ex: only canceled within 2 hours, only set a table after 2 hours now)** | **Vo Huynh Ngoc Tram** |