

# INDIACOM WEBSITE

# (A Conference Management System)

**A DISSERTATION**

**SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS**

**FOR THE AWARD OF THE DEGREE**

**OF**

# MASTER OF COMPUTER APPLICATIONS

# (MCA)

**BY**

## SANCHIT TALREJA

## (Enroll. No.: 35211604422)



**BHARATI VIDYAPEETH’S INSTITUTE OF**

**COMPUTER APPLICATIONS & MANAGEMENT (BVICAM)**

**(GURU GOBIND SINGH INDRAPRASTHA UNIVERSITY) NEW DELHI**

**2024**

**CANDIDATE’S DECLARATION**

I, hereby, declare that the work which is being presented in this dissertation, entitled “**INDIACOM CONFERENCE WEBSITE**” for partial fulfilment of the requirements for the award of the degree of **Master in Computer Applications (MCA),** at **Bharati Vidyapeeth’s Institute of Computer Applications and Management (BVICAM), New Delhi,** is an authentic record of my own work carried out during the period January 2024 to 2024 under the supervision and guidance of Dr. Saumya Bansal (Assistant Professor, BVICAM).

I have not submitted the matter embodied in this dissertation anywhere for the award of any degree or diploma.

**SANCHIT TALREJA**

Enroll. No. 35211604422

**ACKNOWLEDGEMENT**

It is my proud privilege to express my profound gratitude to the entire management and staff members of Bharati Vidyapeeth’s Institute of Computer Applications and Management (BVICAM), New Delhi, for providing me with the opportunity to avail the excellent academic facilities and infrastructure. The knowledge and values inculcated have proved to be of immense help at the very start of my career. Special thanks to Hon’ble Founder, Bharati Vidyapeeth, Pune for having provided us an excellent infrastructure at BVICAM, New Delhi. I am also thankful to Prof. M. N. Hoda (Director, BVICAM) for his astute guidance, support and motivation during the academic period of MCA programme.

I would like to thank my dissertation guide Dr. Saumya Bansal (Assistant Professor, BVICAM) for her valuable support. She took interest in observing my progress throughout the training/internship span and studied all the monthly update I report to him. She came up with regular suggestion and encouragement.

I would like to pay my sincere gratitude to Ms. Ambooj (Assistant Professor, BVICAM) for providing an opportunity to work on project, under her guidance at BVICAM.

I would like to thank Training and Placement Cell of BVICAM for providing me with an opportunity to pursue my internship.

I am expressly grateful to all those names that have not appeared in this acknowledgement but have contributed in significant measure towards the completion of the dissertation.

**SANCHIT TALREJA**

Enroll. No. 35211604422

**INSTITUTE CERTIFICATE**

**ABSTRACT**

This project details the redevelopment of IndiaCom, a website aimed at a conference management system. The original site suffered from usability issues including a cumbersome login system, a clunky review process, and an outdated user interface (UI). This project addresses these shortcomings through a comprehensive redesign.

A new, user-friendly UI streamlines the user experience. The WordPress platform empowers content management for the website without coding expertise. Additionally, a revamped review system featuring manageable research papers and user systems enhances reviewer workflow by allowing in-platform paper review by eliminating the need for downloads.

These improvements collectively enhance the user experience for website visitors and reviewers, establishing India.com as a more effective platform for conference management.

There are mainly 4 modules of the system: user registration, creating a new research paper, updating the research paper, and managing users and the research paper. All the modules are tested by following unit, integration, and load testing mechanisms and found correct in all respects. After successful testing, the system is ready to be shifted in phase 2 and to make this full system we still need more time and manpower.

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Description** | **Page No.** |
| A-1.1 | Use Case Diagram | 51 |
| A-2.1 | DFD Level 0 | 52 |
| A-2.2 | DFD Level 1 | 52 |
| A-3 | Sequence Diagram | 53 |
| A-4 | Activity Diagram | 55 |
| 3.1 | ER Diagram | 28 |
| A-5.1 | Screenshot- Home Page | 56 |
| A-5.2 | Screenshot- My account/ login page | 56 |
| A-5.3 | Screenshot- registration page | 57 |
| A-5.4 | Screenshot- upload research paper page | 57 |
| A-5.5 | Screenshot- update research paper Page | 58 |
| A-5.6 | Screenshot- add co-authors Page | 58 |
| A-5.7 | Screenshot- manage users Page | 59 |
| A-5.8 | Screenshot- manage research paper Page | 59 |
| A-5.9 | Screenshot-generated reports | 60 |
| A-5.10 | Screenshot- load testing | 60 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **Table No.** | **Description** | **Page No.** |
| 1.1 | Deliverables | 13 |
| 3.1 | Structure of research paper Database table | 29 |
| 3.2 | Structure of wp\_users Database table | 29 |
| 3.3 | Structure of wp\_xyz\_ips\_short\_code Database table | 30 |
| 3.4 | Structure of wp\_usermeta Database table | 30 |
| 3.7 | Login Screen Design | 31 |
| 3.8 | Registration Design | 31 |
| 3.9 | Submit paper Design | 32 |
| 3.10 | Update paper Design | 33 |
| 3.11 | Add co-author Design | 33 |
| 3.12 | Manage research paper Design | 33 |
| 3.13 | Manage users Design | 34 |
| 3.16 | User Login Use Case | 34 |
| 3.17 | Submit paper Use Case | 35 |
| 3.18 | Update paper Use Case | 35 |
| 3.19 | Add co-authors Use Case | 36 |
| 3.20 | Manage users Use Case | 37 |
| 3.21 | Manage research paper Use Case | 37 |
| 4.1 | Functional Testing Test Cases | 42 |
| 4.2 | Functional Testing Report | 43 |

**LIST OF ABBREVIATIONS**

|  |  |
| --- | --- |
| **Abbreviation** | **Description** |
| H/W | Hardware |
| S/W | Software |
| MB | Megabyte |
| GB | Gigabyte |
| RAM | Random Access Memory |
| HTTP | Hypertext Transfer Protocol |
| URL | Uniform Resource Locator |
| UI | User Interface |
| DFD | Data Flow Diagram |
| ASP | Active Server Pages |
| DFD | Data Flow Diagram |
| MCA | Master in Computer Applications |
| WP | WordPress |
| RP | Research Paper |

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
|  | | **Page No.** |
| **CHAPTER 1: INTRODUCTION** | | |
| 1.1 | Problem statement | 12 |
| 1.2 | Proposed statement | 12 |
| 1.3 | Deliverables | 13 |
| **CHAPTER 2: PROJECT DESCRIPTION** | | |
| 2.1 | System Interfaces | 15 |
| 2.2 | System Specifications | 17 |
|  | 2.2.1 H/W Requirement | 17 |
|  | 2.2.2 S/W Requirement | 17 |
| 2.3 | Methodology and Tools used | 18 |
|  | 2.3.1 Requirement Phase | 18 |
|  | 2.3.2 Design Phase | 20 |
|  | 2.3.3 Development Phase | 20 |
|  | 2.3.4 Implementation Phase | 21 |
|  | 2.3.5 Testing Phase | 21 |
| 2.4 | Constraints | 22 |
| 2.5 | Assumptions & Dependencies | 23 |
| 2.6 | User Characteristics | 23 |
|  |  |  |
| **CHAPTER 3: FUNCTIONALITIES** | | |
| 3.1 | Local Database Design | 26 |
| 3.2 | Logical Database Design | 29 |
| 3.3 | Use case Description | 34 |
|  |  |  |
| **CHAPTER 4: TESTING** | | |
| 4.1 | Test Activities | 39 |
| 4.2 | Unit Testing | 41 |
| 4.3 | System Testing | 41 |
| 4.4 | Test Reports and Debugging | 43 |
| **CHAPTER 5: CONCLUSION AND FUTURE SCOPE** | | |
| 5.1 | Conclusion | 47 |
| 5.2 | Limitations of the System | 47 |
| 5.3 | Future Scope for Modification | 47 |
|  |  |  |
| **BIBLIOGRAPHY** | | |
| **ANNEXURES** | | |
| A-1 | Design diagrams | 51 |
| A-2 | Data flow diagrams | 52 |
| A-3 | Interaction diagrams | 53 |
| A-4 | Flowchart | 55 |
| A-5 | Screenshots | 55 |
|  |  |  |

|  |  |
| --- | --- |
| **CHAPTER 1**  **INTRODUCTION** | |
| 1.1 | Problem Statement |
| 1.2 | Proposed Solution |
| 1.3 | Deliverables |
|  |  |

**1. Introduction**

IndiaCom, a website catering to researcher’s conference management, underwent a thorough review to identify areas for improvement. This project prioritizes user experience (UX) and aims to establish IndiaCom as a more efficient and user-friendly platform.

User testing revealed several shortcomings hindering user experience. Login issues suggest the need for enhanced testing and debugging procedures. Furthermore, the website's data schema suffers from poor design, leading to inefficiencies and data management challenges. Updating the website is reportedly a time-consuming process further hampered by the lack of a support community for the legacy programming language.

Beyond functionality, both the admin dashboard and the user interface were flagged for user-unfriendliness and outdated aesthetics. These issues likely discourage user engagement and hinder overall website effectiveness.

This project addresses the identified issues through a comprehensive redesign built on a user-centered approach. By prioritizing ease of use and clear navigation, we aim to create a more intuitive experience for both website visitors and reviewers.

The following sections will detail the specific solutions implemented:

A Streamlined User Interface: A modern and user-friendly UI will be implemented, emphasizing clarity and ease of navigation.

Empowering Content Management: We will migrate IndiaCom to the WordPress platform, enabling content updates without requiring coding expertise.

Enhancing the Review Process: A revamped review system will be introduced, featuring manageable research paper and user systems. This will allow reviewers to seamlessly access and review papers directly within the platform, eliminating the need for downloads.

Data Management Made Easy: A well-designed data schema will be implemented, streamlining data organization and facilitating efficient management by using limited table structures.

By addressing these areas, this project aims to transform IndiaCom into a more efficient, user-friendly, and visually appealing platform, ultimately serving its intended purpose more effectively.

**1.1 Problem Statement**

Our review of the current website has identified several areas that require improvement. A recurring issue reported by faculty is encountering problems upon login, which suggests deficiencies in the website's testing and debugging procedures. Furthermore, the current database schema suffers from poor design, leading to inefficiencies and presenting challenges in data management. Updating the website takes a lot of time, and fixing code problems is even harder because the old programming language we use doesn't have much community support. Finally, the user interface and admin dashboard require modernization, as they currently lack user-friendliness and visual appeal.

**1.2** **Proposed Solution**

To improve the user experience and streamline administrative tasks, we propose a new online review system. This system would eliminate the need for faculty to download papers for review, saving them valuable time and effort. Additionally, the system would be designed with a visually appealing interface to enhance user satisfaction. Furthermore, a well-structured and easy-to-understand database like My SQL that contains single table instead of multiple ones would be implemented to facilitate efficient data management. Finally, the system would be built on a platform like WordPress, which utilizes a user-friendly admin module, allowing for straightforward maintenance and updates.

The main objectives of the work are:

* Provide a new review system so reviewers don’t have to download papers to review them.
* Make it visually appealing.
* Providing a simple database instead of a complex one by using limited table structures.
* Provide an admin module by WordPress so it can be maintained and updated easily.

1.3 Deliverables

The list of deliverables is provided in Table 1.1.

**Table 1.1: List of deliverables**

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No.** | **Phase** | **Deliverables** | **Page No.** |
| **1.** | Requirement Analysis | Use Case | 52 |
| **2.** | Design | Entity Relationship Diagram | 26 |
| **3.** | Design | Use case Diagram | 53 |
| **4.** | Design | Data Flow Diagrams | 54 |
| **5.** | Design | Interaction Diagram | 55 |
| **6.** | Design | Flowchart | 57 |
| **7.** | Unit Testing | Test Results | 42 |

|  |  |
| --- | --- |
| **CHAPTER 2**  **PROJECT DESCRIPTION** | |
| 2.1 | System Interfaces |
| 2.2 | System Specifications |
|  | 2.2.1 H/W Requirement |
|  | 2.2.2 S/W Requirement |
| 2.3 | Methodology and Tools used |
|  | 2.3.1 Requirement Phase |
|  | 2.3.2 Design Phase |
|  | 2.3.3 Development Phase |
|  | 2.3.4 Implementation Phase |
|  | 2.3.5 Testing Phase |
| 2.4 | Constraints |
| 2.5 | Assumptions & Dependencies |
| 2.6 | User Characteristics |

**2. Project Description**

**2.1 System Interfaces**

The overall System has the following Interfaces:

* **Login Page:** 
  + This interface allows users to log in to the system.
  + Provides Form-based authentication.
  + Performs the authentication and authorization process.
* **Home Page:**
* This is the landing page after the user is logged in
  + This page will be accessible to both researchers and admin users.
  + This section displays all the tabs like Contact Us, Submit Paper, My Account etc.
* **Submit Paper Page:**
* This page is reached in case the Admin/User clicks on the submit paper link.
  + This section allows the researchers to generate the new paper
  + This is the initial step for the creation of a new paper ID.
* **Update paper Page:**
* This page is reached in case the Admin/User clicks on the update paper link.
  + After the order is created, it is displayed in pending orders where all the details can be viewed unless the payment is made.
* **Get Membership Page:** 
  + This page is reached in case the Admin/user clicks on the get membership Tab.
  + This page allows the person to create a new user and will be gone for approval to the admin side.
* **My Account Page:**
* This page is reached in case Users/Admin click on my account Tab.
  + This page gives us information about the user and also has features like password change etc.
* **Add Co-Authors Page:**
* This page is reached when the Admin /user clicks on the update co-authors link.
  + This page helps in adding co-authors in the specific row of paper IDs submitted by the researcher.
* **Users page:** 
  + This page is reached when the Reviewer/Admin clicks on the Indiacom logo and after that user's tab.
  + Admin can manage users and approve or deny from here and also edit the values of the users if they want to.
  + This is a wp\_users dashboard pushed to the front end.
* **Research Paper Page:**
* This page is reached in case the Reviewer/Admin clicks on the Indiacom logo and after that research paper tab.
  + Admin can manage all the research papers & can edit them according to anything they want but there is no delete functionality.
* **Wp-admin Page:**
* This page is reached when the Admin writes wp-admin after the website in url.
  + This page can handle full website features, pages, posts, plugins etc.

**2.2 System Specification**

**2.2.1 H/W Requirements**

* **Architecture:**
  + 64 bit hardware architecture
  + Intelprocessor with compatible Motherboards
* **Processing Power**
  + i3 4th gen 2.4-gigahertz (GHz) processor or faster
* **Memory**
  + 4 GB of RAM (8 GB is recommended)
* **Secondary Storage**
  + 50 GB of available space on the hard disk

**2.2.2 S/W Requirements**

* **Platform:**
  + Web Browser compatible with 1024 x 768 or better resolution
* **Services**
  + WordPress:Version 6.5.2
  + WP\_Plugins: Elementor (basic,pro,essential addons,Ea-pro) , insert php code snippet, user registration,user role editor ,wp data access, wp frontend admin, disable gutenberg
  + Server: Apache (using mamp)
  + Languages: PHP 8.3.4, HTML, CSS
  + Database: My SQL 5.2.0
* **Web Browser**
  + Browser compatible to request active server pages over HTTP.

**2.3 Methodology and Tools Used**

The concerned project uses the agile methodology of iterative development where requirements and solutions evolve through collaboration between self-organizing cross-functional teams.

Agile web development is a model for the development of web applications. It is more efficient and powerful within a shorter timeline than other models incorporates face-to-face communication, and includes technical personnel as well as customers as part of the team. Agile web development uses project managers, and business analysts, emphasizing clear goals, planning, and iterative delivery. Agile web development ensures the successful completion of the product at the end of the iteration.

**2.3.1 Requirement Phase**

The requirement phase guided the development team through designing a prototype and setting goals.

The discovery phase included these steps:

* Built the employee wish list
* Performed interviews, group discussions, brainstorming sessions
* Identified the project team
* Established the development environment
* Identify client requirements
* Set the project scope and schedule

**Requirements of the System:**

* All users must log in with their username and password.
* Role Based Security Needed: Admin and User
* Administrator manages users, website, research papers and traffic.
* User can view and manage his Profile on the application.
* Validate the information entered by the user on web pages.
* Role-based security should be there. A researcher should not be able to edit the information of other researchers.
* System should save the state of the application persistently by monitoring database transactions having multiple queries.

Thus, the first deliverable functionality will be divided into the following 3 parts:

* **Authentication and Authorization**: This includes login functionality which includes that there should be some security provided on the way users log in so that only Admin has access to the administration page where users, posts, and issues can be modified, added or deleted.
* **Administration Functionality**: This includes a functionality where admins can administer the software and is able to modify the records.
* **Researcher’s Functionality:** This includes functionality where researchers can view, upload, update or add co-authors to the paper.

**2.3.2 Design Phase**

The design phase is a vital turning point in a project where abstract ideas solidify into concrete plans. It starts by taking the requirements and goals identified earlier and translating them into technical specifications and blueprints. This might involve creating mockups, user flows, or outlining functionalities. The entire process is crucial because it ensures everyone involved is on the same page and minimizes the risk of errors or rework later in the project. By meticulously crafting a detailed design, you create a roadmap that guides the development process towards a successful outcome.

**2.3.3 Development Phase**

The System was developed by following 3-tier architecture with bottom up strategy defined as follows:

* **Data Access Layer**: A data context has been defined for this layer to manage connections to the database tables, providing easy access. A framework has been created at this layer to define a database context class that manages the flow of information in and out of this layer.
* **Business Logic Layer**: This layer serves as a bridge between the interface layer and the data access layer, housing all business and modification logic.
* **Presentation Layer**: This layer defines the generic interface through which the user interacts. It contains classes that define functions for presenting data retrieved from the Business Access Layer to the user in the required formats.

The tools which were used are as follows:

* **Microsoft Visual Studio Code**

It is an integrated development environment (IDE) from Microsoft.

* **MY SQL from my PHP admin for Database Management**

phpMyAdmin is a free and open-source administration tool for MySQL and MariaDB. As a portable web application written primarily in PHP

**2.3.4 Implementation Phase**

The project was brought to life on a local development environment using MAMP. This software package provided a simulated server environment, allowing for testing and development without affecting the live website. Within MAMP, a MySQL database was created and managed using phpMyAdmin. This database serves as the central storage for website content. WordPress, a popular content management system (CMS), was then installed. WordPress empowers non-technical users to easily manage website content through a user-friendly interface. However, for functionalities not natively offered by WordPress, custom code forms were created using HTML, CSS, and PHP. This approach allowed for the development of bespoke features tailored to the specific needs of the project. By integrating these custom forms seamlessly within the WordPress framework, we ensured a user-friendly content management experience while extending the website's capabilities.

**2.3.5 Testing Phase**

Bottom up testing was performed on deliverable of the project starting from Data Access Layer.

* **Unit Testing**
* **Submit Paper Form:** Ensure all essential fields are required and have informative error messages for missing information.
* **Update Paper Form:** Validate file uploads, check for file size limits, and confirm correct storage of uploaded files.
* **Add Co-Authors Form:** Test user search functionality for accuracy and selecting/adding co-authors with limitations (if applicable).
* **Login System:** Verify username/password validation, error messages for invalid credentials, and successful session creation with redirection to the appropriate dashboard.
* **Integration Testing**
  + Integration testing within the deliverable involved integrating all the layers and checking the secure flow of Information under the defined roles.
  + Integrity checks were tested on the Data Passed to the Data Access Layer from the User Interface Layer.
  1. **Constraints**
* Display of content varies from Browser to Browser it is recommended to use Google Chrome or similar browsers to view the site perfectly.
* It is recommended that your internet speed should be more than 850kbps to load the website faster.
* It is recommended to use the website after login to access all the features of the user.
* Data should be posted in the research paper tab so that there is no mismatch left.
* The testing process is carried out on a limited number of hosts.

**2.5 Assumptions and Dependencies**

We assume that the client will make available the required software and access permissions for all the environments required for the purpose of project execution.

The client will provide all documents that are relevant to the application and are within the scope of the project. These documents include but are not limited to the following:

* Requirement documents.
* Screen Mock-ups and working Prototypes (wherever available) approved by business users.
* Use Case.
* Design and Architecture document and list of batches/interfaces.
* All design documents for all components.
* The client would provide Test data for testing purposes.

Dependencies will be as follows:

* Design Specifications sign-off.
* Research paper data should be posted in the research paper tab for the processing of further rounds and payments if selected.
* The client’s resources should be available for any query/clarification.

**2.6 User Characteristics**

Under Role Based Security three types of Users have been defined: The Admin user the researcher user and the Reviewer.

The researcher has the following rights in the application –

* Login to the system using their “User Name” and “Password”.
* Upload or generate a new research paper
* View submitted research papers
* Update the research paper generated already by the current user
* Add co-authors who helped in making the research paper

The Reviewer has the following rights in the application –

* Login to the system using their “User Name” and “Password”.
* Can access the admin dashboard which contains the modules like managing users or research paper
* Can’t access all the features like admin have main wp-admin dashboard is not accessible

Admin Users have all rights throughout the application. They have the following additional rights in the application along with employee rights:-

* Login to the system using their admin “User Name” and “Password”.
* Grant access to modules to the users according to their job assigned.
* Has full access to the database and all the rights any of the Reviewers and researchers have.
* Can access the wp-admin dashboard also.

|  |  |
| --- | --- |
| **CHAPTER 3**  **FUNCTIONALITY** | |
| 3.1 | Logical Database Design |
|  | 3.1.1 ERD |
|  | 3.1.1Table Structures |
| 3.2 | Input and Output Design |
| 3.3 | Use case Description |

**3. FUNCTIONALITY**

**3.1** **Local Database Design**

This ER diagram represents the database schema for an "Indiacom Conference Website." It includes four main tables: research\_paper, wp\_xyz\_ips\_short\_code, wp\_users, and wp\_usermeta. Here’s a breakdown of each part:

1. research\_paper

Primary Key (ID): The unique identifier for each research paper.

Attributes:

SESSION: The session in which the paper is presented.

TITLE: The title of the research paper.

EMAIL: Contact email of the author.

MID: An identifier related to the author or paper.

NAME: Name of the author.

CONTACT AUTHOR ID: Identifier for the contact author.

DATE OF LAST UPDATE: The date when the paper was last updated.

EVENT ID: Identifier for the event.

STATUS: Current status of the paper.

PLAG PATH: Path to the plagiarism report.

TRACK ID: Identifier for the track.

DATE OF SUBMISSION: The date when the paper was submitted.

DOCUMENT PATH: Path to the document.

PRESENTATION PATH: Path to the presentation.

2. WP\_XYZ\_IPS\_SHORT\_CODE

Attributes:

ID (PK): The unique identifier for each short code entry.

TITLE: Title of the short code.

CONTENT: Content associated with the short code.

SHORT\_CODE: The actual short code.

STATUS: Status of the short code.

Relation to research\_paper: Indicates that short codes are fetched for display purposes related to research papers.

3. wp\_users

Primary Key (ID): The unique identifier for each user.

Attributes:

USER\_URL: URL associated with the user.

USER\_EMAIL: Email of the user.

USER\_ACTIVATION\_KEY: Activation key for the user.

USER\_STATUS: Status of the user.

USER\_LOGIN: Username for login.

USER\_PASS: Password for login.

USER\_NICENAME: Nicename of the user.

USER\_REGISTERED: Registration date of the user.

DISPLAY NAME: Display name of the user.

4. wp\_usermeta

Primary Key (UMETA\_ID): The unique identifier for each user meta entry.

Attributes:

USER ID: The user ID this meta data is associated with.

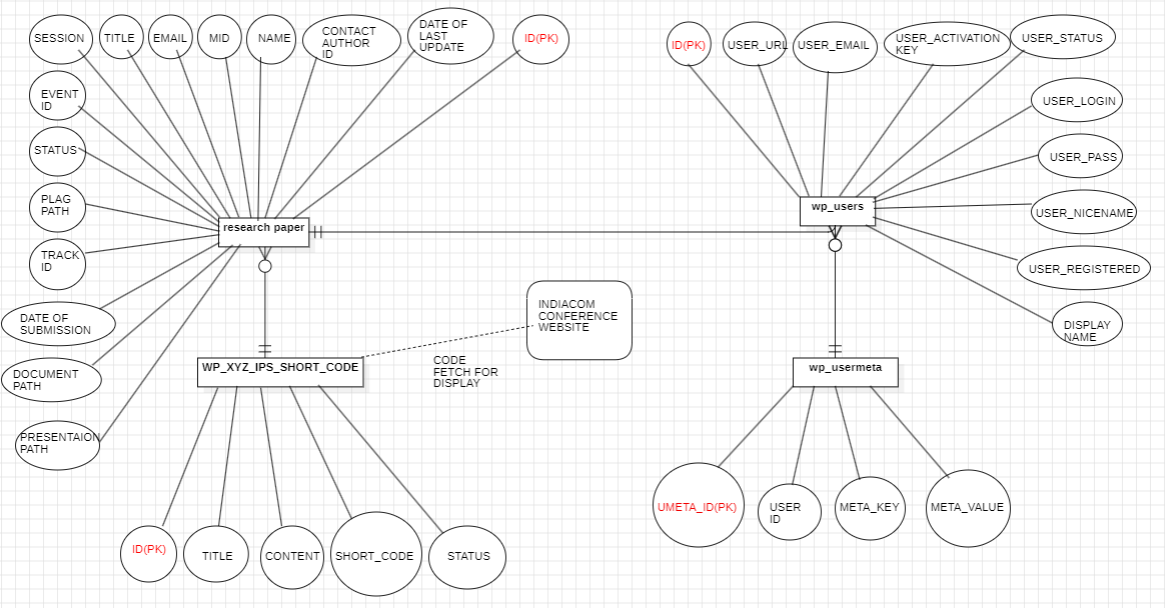
META\_KEY: Key for the meta data.

META\_VALUE: Value for the meta data.

Relation to wp\_users: Stores additional metadata for users.

**3.1.1Entity Relation Diagram (ERD)**

The ER diagram of the developed system is shown in Figure 3.1



**Figure 3.1 E-R Diagram of System**

**3.1.2 Table Structures**

The structure of various database tables designed for the “INDIACOM Conference Website” is described in Tables 3.1 to 3.4.

**Table 3.1: Structure of research paper Database table**

| **S. No.** | **Name of Column** | **Data Type** | **Description** |
| --- | --- | --- | --- |
|  | paper\_id | Int(11) | Primary Key, Auto Increment |
|  | name | varchar(255) | Name of the user |
|  | mid | int(11) | login id of the user |
|  | email | varchar(255) | Email ID of the user |
|  | title | varchar(255) | Title of research paper |
|  | date\_of\_submission | varchar(220) | Submission date of research paper |
|  | date\_of\_last\_update | varchar(220) | Update date of research paper |
|  | event\_id | Varchar(255) | Event details |
|  | track\_id | varchar(255) | Who is going to check |
|  | session | varchar(50) | Session details |
|  | document\_path | varchar(255) | Research paper store path |
|  | plag\_path | varchar(255) | Plagiarism store path |
|  | presentation\_path | varchar(255) | Presentation store path. |
|  | contact\_author\_id | varchar(210) | Co-author id details |
|  | status | enum | Status of the paper |

**Table 3.2: Structure of wp\_users Database table**

| **S. No.** | **Name of Column** | **Data Type** | **Description** |
| --- | --- | --- | --- |
|  | ID | Bigint(20) | Primary Key, Auto Increment |
|  | User\_login | Varchar(60) | username |
|  | User\_pass | Varchar(255) | User Password |
|  | User\_nicename | Varchar(50) | short name |
|  | User\_email | Varchar(100) | email |
|  | User\_url | Varchar(100) | Access URL |
|  | User\_registered | DateTime | Time of user made |
|  | User\_activation\_key | Varchar(255) | Activation key |
|  | User\_status | Int(11) | Status of user |
|  | Display\_name | Varchar(250) | Name to be displayed |

**Table 3.3: Structure of wp\_xyz\_ips\_short\_code Database table**

| **S. No.** | **Name of Column** | **Data Type** | **Description** |
| --- | --- | --- | --- |
|  | Id | Int(11) | Primary Key, Auto Increment |
|  | Title | varchar(1000) | Title of code |
|  | Content | Longtext | Actual code |
|  | Short\_code | Varchar(2000) | Shortcode |
|  | Status | Int(11) | Active or paused |

**Table 3.4: Structure of wp\_usermeta Database table**

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No.** | **Name of Column** | **Data Type** | **Description** |
|  | umeta\_id | Bigint(20) | Primary Key, Auto Increment |
|  | User\_id | bigint(20) | Secondary key, user ID |
|  | Meta\_key | varchar(255) | Secondary key, user ID |
|  | Meta\_value | longtext | Values are stored here in +1 format |

**3.2** **Input Output Design**

The Input-Output Design of various Screens designed for the “INDIACOM conference website” is described in Tables 3.7 to 3.13.

**Table 3.7: Input-Output Design of Login Screen**

|  |  |
| --- | --- |
| **Purpose** | * Enables user, Reviewer or Administrator to log into the System |
| **Description of field(s)** | * User Name: The user enters his/her login name to log into the System. * Password: User enters Password. * Login Button: Logs the user in or displays an error on the mismatch of username and password. |
| **Validation Checks** | * The username should be a registered User Name. * Password should match the entered username in the database. |

**Table 3.8: Input-Output Design of Registration (Get Membership) Screen**

|  |  |
| --- | --- |
| **Purpose** | * This form helps in signing up as a user |
| **Description of field(s)** | * Salutation: The user will choose what he is actually a doctor, Mr, Mrs, miss, professor * User email: here the user will type the email address that is currently he is using. * First name: here user will write his first name * Last name: hare user will write his last name * Username: here user will choose his username if the username is already taken then it will show the error * User password: here the user will enter is user password which will need to be strong enough so the website will accept it. * Confirm password: here user will write the password again if the password does not match the user password then this field will show the error. * Address: here user will type his address * Country: The user will select his country * Pin code: here user will type his pin code * State: here the user will type the state he lives in. * Mobile number: here user we will type is phone number * User bio: here user will type his bio. * Interested event: your user will select the events is interested in. * CSI membership number: here user will type a CSI membership number if any * IEEE membership no.: here user will type his IEEE membership number if any * Organisation name: here user will type the organisation name he belongs to. * Category: here the user will choose the category if he is a research scholar, full-time student, faculty or industry representative |
| **Validation Checks** | * Same username is not allowed * Same email address is not allowed * Password should be strong * All fields are mandatory with an asterisk symbol. |

**Table 3.9: Input-Output Design of submit paper Screen**

|  |  |
| --- | --- |
| **Purpose** | * Enables researchers to submit a new paper. |
| **Description of field(s)** | * Event: This field is used to select the event for which the paper is being uploaded. * Track: This field is used to select the track for which the paper will be on which track. * Session: This field is used to select the session for which the paper is uploaded. * Title: this field will be filled with the title of the paper. * Upload paper: this field is used to upload the paper. * Upload plagiarism: this field is used to upload plagiarism reports. * Terms and conditions: This field is used to get acceptance for the terms and conditions. |
| **Validation Checks** | * All fields are mandatory with an asterisk symbol. * Upload paper should be in pdf format. * Plagiarism report should be in pdf format. |

**Table 3.10: Input-Output Design of Update Paper Screen**

|  |  |
| --- | --- |
| **Purpose** | * Enables to update the existing paper of the current user. |
| **Description of field(s)** | * Select the paper ID first in which updates are to be made. * Upload revised paper. * Upload plagiarism report. * Upload the presentation. |
| **Validation Checks** | * Mandatory fields need to be filled. * Revised paper should be in pdf or Word format * Plagiarism report should be in pdf format * Presentation should be in pptx or ppt format. |

**Table 3.11: Input-Output Design of add co-authors Screen**

|  |  |
| --- | --- |
| **Purpose** | * Enables researchers to add co-authors to the submitted paper. |
| **Description of field(s)** | * Select the paper ID first in which updates are to be made. * Select the co-authors whom you worked with. * Selected co-authors will be shown here before submission. |
| **Validation Checks** | * Mandatory fields need to be filled. |

**Table 3.12: Input-Output Design of Manage Research Paper Screen**

|  |  |
| --- | --- |
| **Purpose** | * Used to manage all the research papers. |
| **Description of field(s)** | * Helps in downloading the paper. * Helps in updating the paper if necessary. * Helps in updating the status of the paper. |
| **Validation Checks** | * Nothing is mandatory. |

**Table 3.13: Input-Output Design of Manage Users Screen**

|  |  |
| --- | --- |
| **Purpose** | * Used to manage all the researchers. |
| **Description of field(s)** | * Helps in approving the researchers. * Helps in editing the researcher's profile. |
| **Validation Checks** | * Nothing is mandatory. |

**3.2** **Use Case Description**

The Use Case Description for “INDIACOM CONFERENCE WEBSITE” is described in Tables 3.16 to 3.21.

**Table 3.16: Use Case Description of Login Process**

|  |  |
| --- | --- |
| **Purpose** | * This use case defines the registration and Login process of the Administration and General user. |
| **Actors** | * Reviewer * Admin * Researcher |
| **Preconditions** | * User should be registered in case of login, retrieve Password, change password. * In case of Admin Login the User should be registered under Admin Role. |
| **Post Conditions** | * Home pages are displayed depending on the successful login not depending on the role of the user. |
| **Basic Flow** | * On Failure Error Message is displayed. * The system requests the actor to enter his/her username and Password. The role can be any one of the admin and the User. * The actor enters his/her username and password. * The system validates the entered name and password and Logs in the actor into the system. |
| **Alternate Flows** | * User is shown an error message with an invalid username or Password. |

**Table 3.17: Use Case Description of Submit Paper Process**

|  |  |
| --- | --- |
| **Purpose** | * Enables researchers to submit the new paper. |
| **Actors** | * Reviewer * Admin * Researcher |
| **Preconditions** | * User should be logged into the system. * All the mandatory fields need to be filled |
| **Post Conditions** | * Paper is submitted if all the details are filled. |
| **Basic Flow** | * Form to be filled * Actor clicks on ‘submit’ * Success message is shown * The actor can see the uploaded paper results below in the table. |
| **Alternate Flows** | * Invalid details are entered and an appropriate error message is displayed. |

**Table 3.18: Use Case Description of update paper Process**

|  |  |
| --- | --- |
| **Purpose** | * Enables researchers to update the existing paper of the current user. |
| **Actors** | * Reviewer * Admin * Researcher |
| **Preconditions** | * User should be logged into the system * There should be an existing paper ID of that user |
| **Post Conditions** | * All the papers are displayed of that user. |
| **Basic Flow** | * The actor selects the paper ID upon which he wants to work. * Updates the paper, plagiarism report and presentation. * Clicks on the update button. * Success message will be shown. |
| **Alternate Flows** | * Invalid details are entered and an appropriate error message is displayed. |

**Table 3.19: Use Case Description of add co-authors Process**

|  |  |
| --- | --- |
| **Purpose** | * Enables to update the co-authors of the existing paper of the current user. |
| **Actors** | * Reviewer * Admin * Researcher |
| **Preconditions** | * User should be logged into the system. * There should be an existing paper ID of that user |
| **Post Conditions** | * All the papers ID are displayed of that user. |
| **Basic Flow** | * User selects the paper id. * User selects the co-authors. * Click on submit. * Success message will be shown. |
| **Alternate Flows** | * Invalid details are entered and an appropriate error message is displayed. |

**Table 3.20: Use Case Description of Manage Users Process**

|  |  |
| --- | --- |
| **Purpose** | * Enables actors to manage the website users. |
| **Actors** | * Reviewer * Admin |
| **Preconditions** | * User should be logged into the system |
| **Post Conditions** | * All the users will be shown here. |
| **Basic Flow** | * Actor can manage the table of users. * The actor can Edit, approve and view user details. |
| **Alternate Flows** | * None |

**Table 3.21: Use Case Description of Manage Research Papers Process**

|  |  |
| --- | --- |
| **Purpose** | * Enables actors to manage the website research paper. |
| **Actors** | * Reviewer * Admin |
| **Preconditions** | * User should be logged into the system |
| **Post Conditions** | * All the research paper will be shown here. |
| **Basic Flow** | * Actor can manage the table of research paper. * The actor can Edit, approve status and view paper details of any researchers. |
| **Alternate Flows** | * None |

|  |  |
| --- | --- |
| **CHAPTER 4**  **TESTING** | |
| 4.1 | Test Activities |
| 4.2 | Unit Testing |
| 4.3 | System Testing |
|  | 4.3.1 Functional Testing |
| 4.4 | Test Reports and Debugging |

**4. Testing**

**4.1** **Testing Activities**

Testing is a process rather than a single activity. Testing must be planned, and it requires discipline to act upon it. The quality and effectiveness of software testing are primarily determined by the quality of the test processes used.

The activities of testing can be divided into the following basic steps:

* Planning and Control
* Analysis and Design
* Implementation and Execution
* Evaluating exit criteria and Reporting
* Test Closure activities

Test planning involves producing a document that describes an overall approach and test objectives. It involves reviewing the test basis, identifying the test conditions based on analysis of test items, writing test cases and designing the test environment. Completion or exit criteria must be specified so that we know when testing (at any stage) is complete.

# Purpose:

* + - * To determine the scope and risks and identify the objectives of testing.
      * To determine the required test resources like people, test environments etc.
      * To schedule test analysis and design tasks, test implementation, execution.

Control is the activity of comparing actual progress against the plan, and reporting the status, including deviations from the plan. It involves taking actions necessary to meet the mission and objectives of the project.

Test analysis and Test Design has the following major tasks:

* To review the test basis. The test basis is the information on which test cases are based, such as requirements, design specifications, product risk analysis, architecture and interfaces.
* To identify test conditions.
* To design the tests.
* To design the test environment set-up and identify the required infrastructure and tools.

Test execution involves running the specified test on a computer system either manually or by using an automated test tool. It is a Fundamental Test Process in which actual work is done. Test implementation has the following major task:

* + - * To develop and prioritize test cases by using techniques and create test data for those tests.
      * To create test suites from the test cases for efficient test execution.
      * Test suite is a collection of test cases that are used to test a software program
      * To re-execute the tests that previously failed in order to confirm a fix.
      * To log the outcome of the test execution. A test log is the status of the test case (pass/fail).
      * To compare actual results with expected result.

Evaluating exit criteria is a process defining when to stop testing. It depends on coverage of code, functionality or risk. Basically, it also depends on business risk, cost and time and vary from project to project. Exit criteria come into picture, when:

* + - * Maximum test cases are executed with certain pass percentage
      * Bug rate falls below certain level
      * When we achieve the deadlines

Test closure activities are done when software is ready to be delivered. The testing can be closed for the other reasons also like:

* When a project is cancelled
* When some target is achieved
  + When a maintenance release or update is done

**4.2 Unit Testing**

It is a level of software testing where individual units/ components of software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output. In procedural programming, a unit may be an individual program, function, procedure, etc. In object-oriented programming, the smallest unit is a method, which may belong to a base/super class.

* **Methodology Used**

Unit testing was carried out at the developer environment only. Manual testing was done. The developers review their code to check whether their respective units under tests behave as expected.

* **Tools Used**

No tools were used to do the Testing as manual testing was carried out for Unit Testing of all the modules. As Manual Testing doesn’t require any tools, so Tools are not applicable.

**4.3 System Testing**

System Testing (ST) is a black box testing technique performed to evaluate the complete system, the system’s compliance against specified requirements. In System testing, the functionalities of the system are tested from an end-to-end perspective. System Testing is usually carried out by a team that is independent of the development team in order to measure the quality of the system unbiased. It includes both functional and Non- Functional testing.

**4.3.1 Functional Testing**

* **Methodology Used**

Under this the whole system was tested under the development team. Basically, all functionalities as per requirements are tested here.

* **Tools Used**

No tools were used to do the Testing as manual testing was carried out for Functional Testing. As Manual Testing doesn’t require any tools, so Tools are not applicable.

* **Test Cases**

**Table 4.1: Functional Testing Test Cases**

| **Id** | **Test Item(s)** | **Test Case Name** | **Test Case** **Description** |
| --- | --- | --- | --- |
|  | registration | user successfully created | The person filling form to create new user |
|  | Login/logout | Authentication and logout working | Login system working properly |
|  | Upload paper | New paper id will be generated | Helps in making a new research paper by uploading them |
|  | Update paper | Existing paper can be updated | Only current user submitted paper will be updated |
|  | Add co-authors | Can add co-authors in Existing paper | Only current user submitted paper will be able to get those co-authors |
|  | Manage users | Full user control | control all the details about the user |
|  | Manage research paper | Full control over research paper table | control all the details about the research paper table |
|  | Update research paper admin control | Editing the details of the research paper table | Updating the particular paper id |
|  | User approval | Approve the user to make the user account | Approving the user to be the researcher so they can use all the user modules. |

**4.4 Test Reports and Debugging**

**Table 4.1: Functional Testing Report**

| **Id** | **Test Case**  **Description** | **Test Case**  **Input** | **Test Results** | | **Status** | **Corrective**  **Measure** |
| --- | --- | --- | --- | --- | --- | --- |
| **Expected** | **Actual** |
|  | registering as a user in the get membership page to create mid | All the fields asked in the create new user are filled properly. | user is created successfully | user is created  successfully | Pass | None |
|  | registering user with the already exsiting email id | All the fields asked in the create new user are filled properly.  with alredy exsisting email id | new paper is created successfully | new paper is created successfully | Fail | None |
|  | generating a new paper in submitting new paper | All the fields asked in the create new paper are filled properly. | new paper is created successfully | new paper is created successfully | Pass | None |
|  | updating the existing paper of the current user | All the fields asked in the update paper are filled properly. | paper is updated succesfully | paper is updated succesfully | Pass | None |
|  | adding all the co-authors of the paper of the current user | All the fields asked in the co-author are filled properly | co-authors added succesfully | co-authors added succesfully | Pass | None |
|  | admin can access the admin dashboard as well as wp admin page | login with admin and using admin dashboard | admins can use it | admins can use it | Pass | None |
|  | contributers can access the admin dashboard but not the wp admin page | login with contributer and using admin dashbaord and using wp admin page | Succeed in both the panels | success use at admin dashboard but not wp admin | Fail | None |
|  | no user can be made without the approval of admin | user approved by admin | user made successfully | user made successfully | Pass | None |
|  | user can’t update or change the values of reserch paper of any other user. | form can’t be filled  as select box is showing only that paper id which is created by that user. | can’t change others paper | can’t change others paper | Pass | None |
|  | user display data is only for the display purpose for the users.  (can be used to download their uploaded reserch paper,ppt,plagiarism) | no input needed | data displayed and ready to download | data displayed and ready to download | Pass | None |

|  |  |
| --- | --- |
| **CHAPTER 5**  **CONCLUSION AND FUTURE SCOPE** | |
| 5.1 | Conclusion |
| 5.2 | Limitations of the System |
| 5.3 | Future Scope for Modification |
|  |  |

**5. Conclusion and References**

**5.1 Conclusion**

A more user-friendly website was required to facilitate progress and enhance the overall user experience. Proposed solution includes the development of a clear and accessible database coupled with a streamlined viewing system. By integrating a WordPress admin module, The aim was to empower administrators and simplify maintenance tasks. Main focus is on creating a visually appealing platform that not only fosters a positive user experience but also encourages engagement. The vision is to establish a dynamic hub for knowledge sharing, seamlessly integrating technology with style. Ultimately, our goal is to create a platform that empowers both students and professors, simplifies tasks for faculty members.

* 1. **Limitations of the System**
* Still the review system is very time consuming and also need some better approach than by just viewing the paper in new window .
* The security of the system is not up to the mark as in some areas we used “get” method rather than “post” method which is going to be a big fuss in future.

**5.3 Future Scope for modification**

* Implement a second and third-level admin system for enhanced administrative control.

2nd level: Further the paper will be reviewed more thoroughly to check any discrepancies.  
3rd level: Reviewing done better with plagiarism checks and AI tools.

* Introduce a paper view system to facilitate a better review mechanism.
* Introducing payment systems to get a more seamless user experience in one place.
* Using auto SEO plugins for monthly SEO to give this website a wider reach internationally.
* Convert the entire system into a comprehensive mobile application for greater accessibility and convenience.

**BIBLIOGRAPHY**

**Books:**

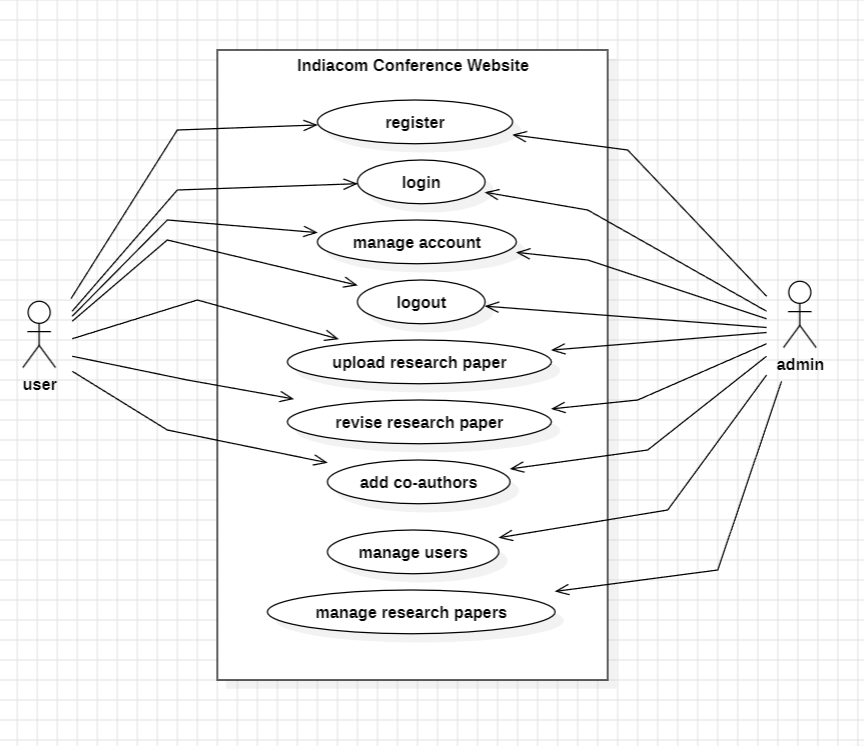
* Get Coding!(2017-2018): Learn HTML, CSS & JavaScript & Build a Website, App & Game – by Young Rewired State
* JavaScript (7th edition may 2020): The Definitive Guide, by David Flanagan

**Websites:**

* https://www.w3schools.com/
* https://wordpress.com/learn/
* https://www.coursera.org/
* https://chat.openai.com/

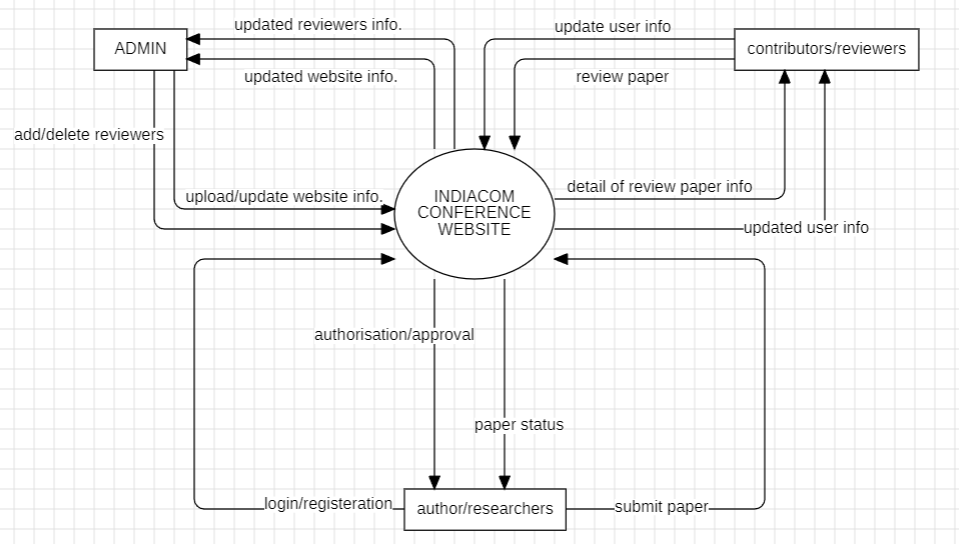
|  |  |
| --- | --- |
| **ANNEXURES** | |
| A-1 | Design Diagram |
| A-2 | Data Flow Diagram |
| A-3 | Interaction Diagram |
| A-4 | Flowchart |
| A-5 | Screenshots |

**A-1: Design Diagram**

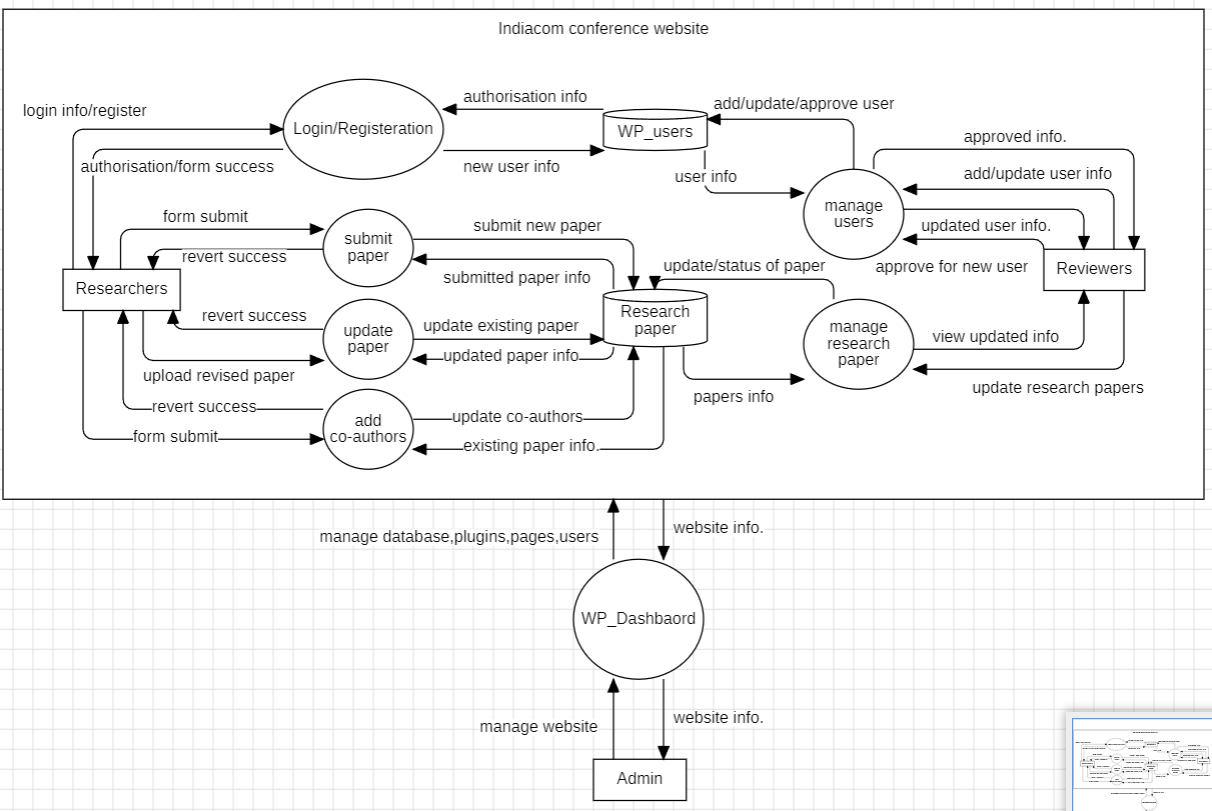


**Figure A-1.1: Use case Diagram**

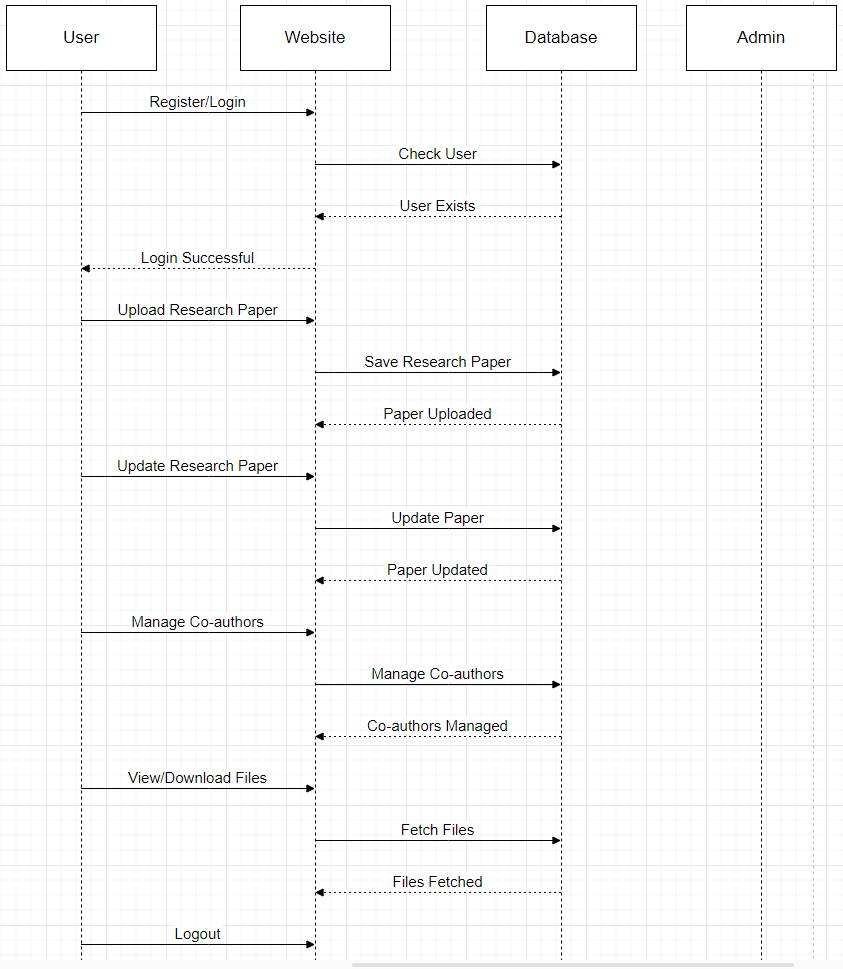
**A-2: Data Flow Diagram**

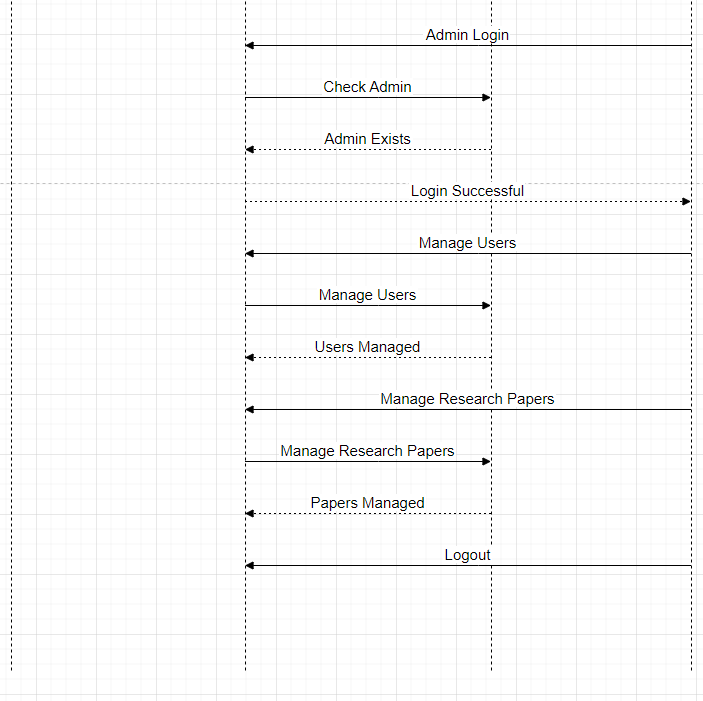
****

**Figure A-2.1: DFD Level 0**

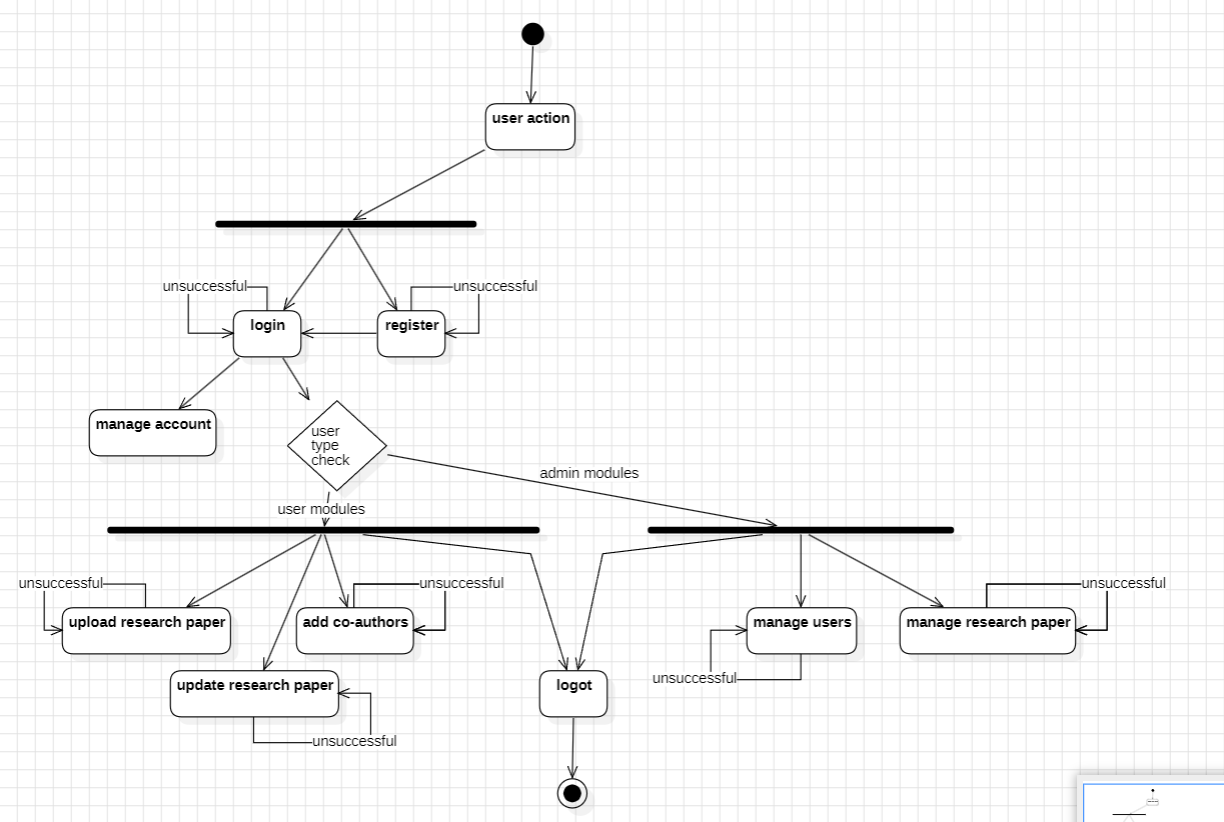
****

**Figure A-2.2: DFD Level 1**

****

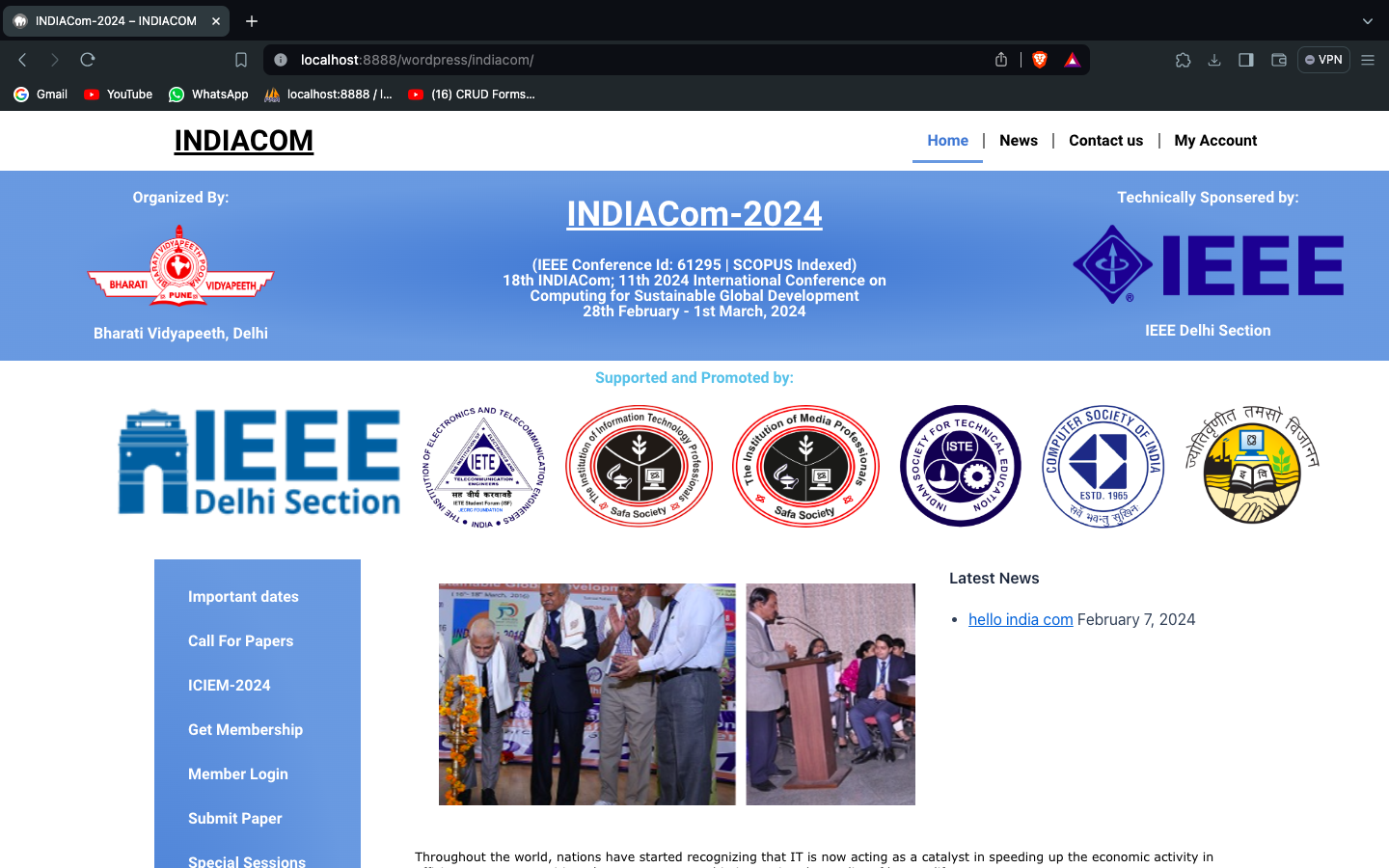
****

**Figure A-3: Sequence diagram**

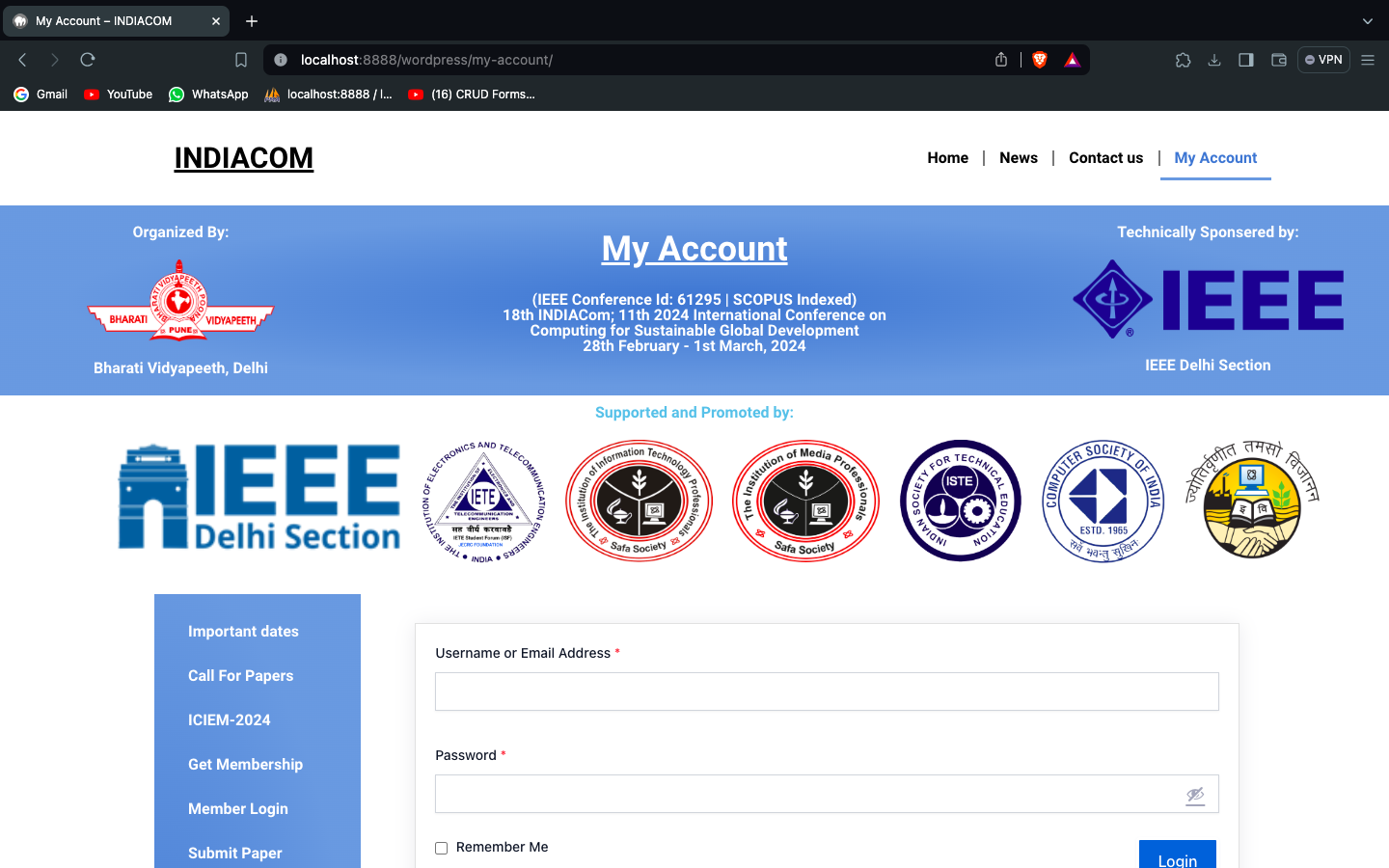
****

**Figure A-4.: Activity diagram**

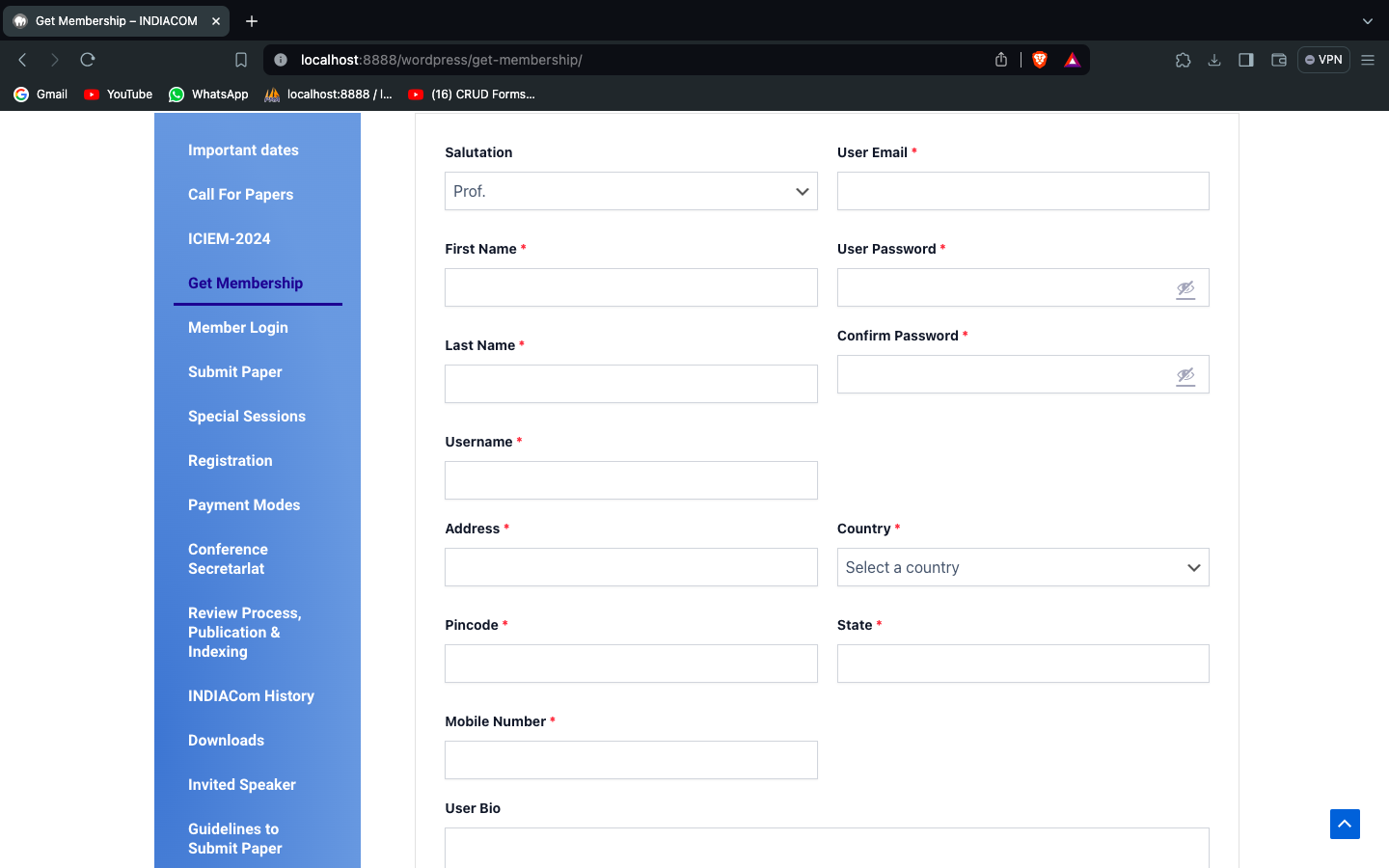
**A-5: Screenshots (Running View of the Forms/Pages)**

****

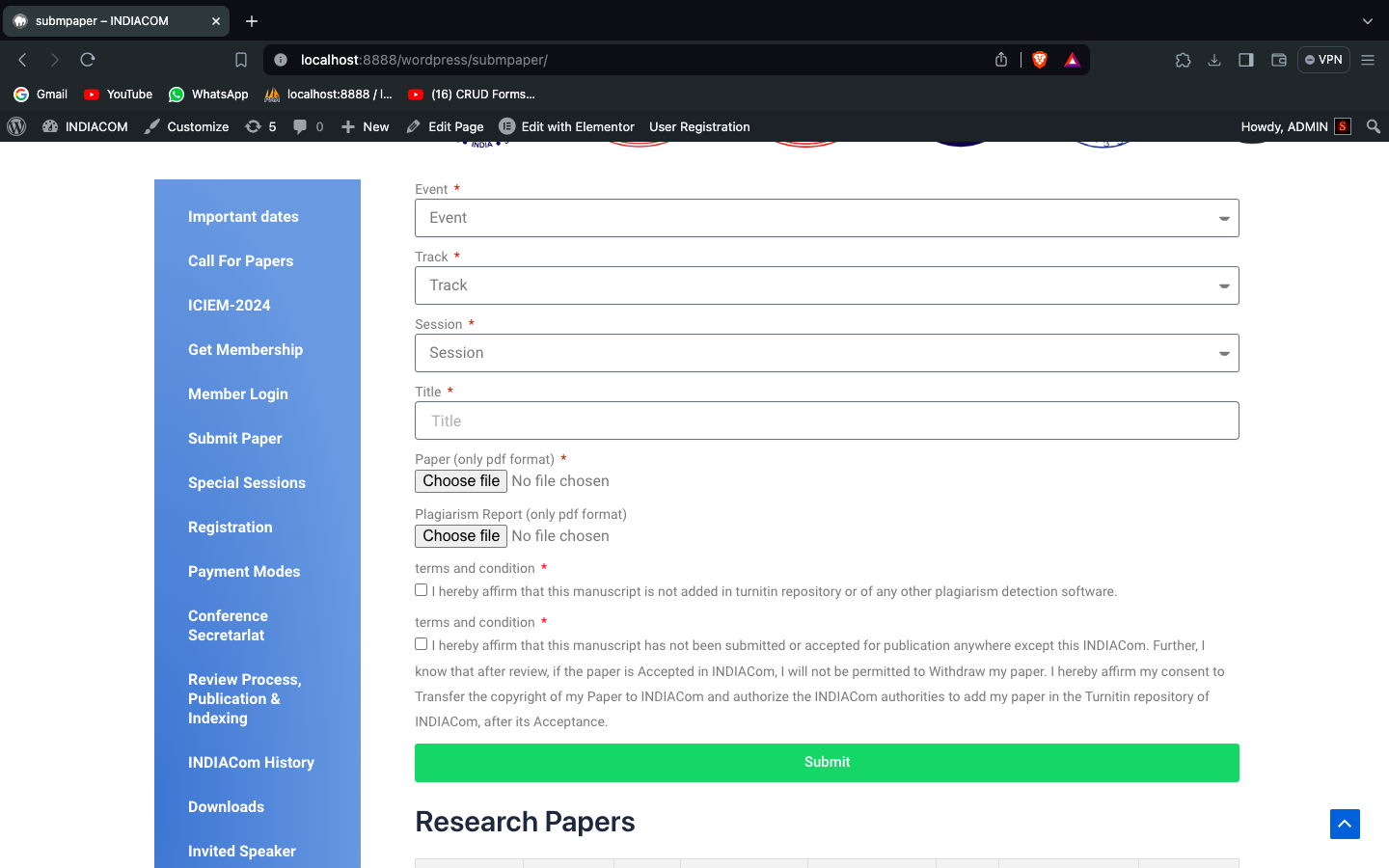
**Figure A-5.1: Home/Landing Page**

****

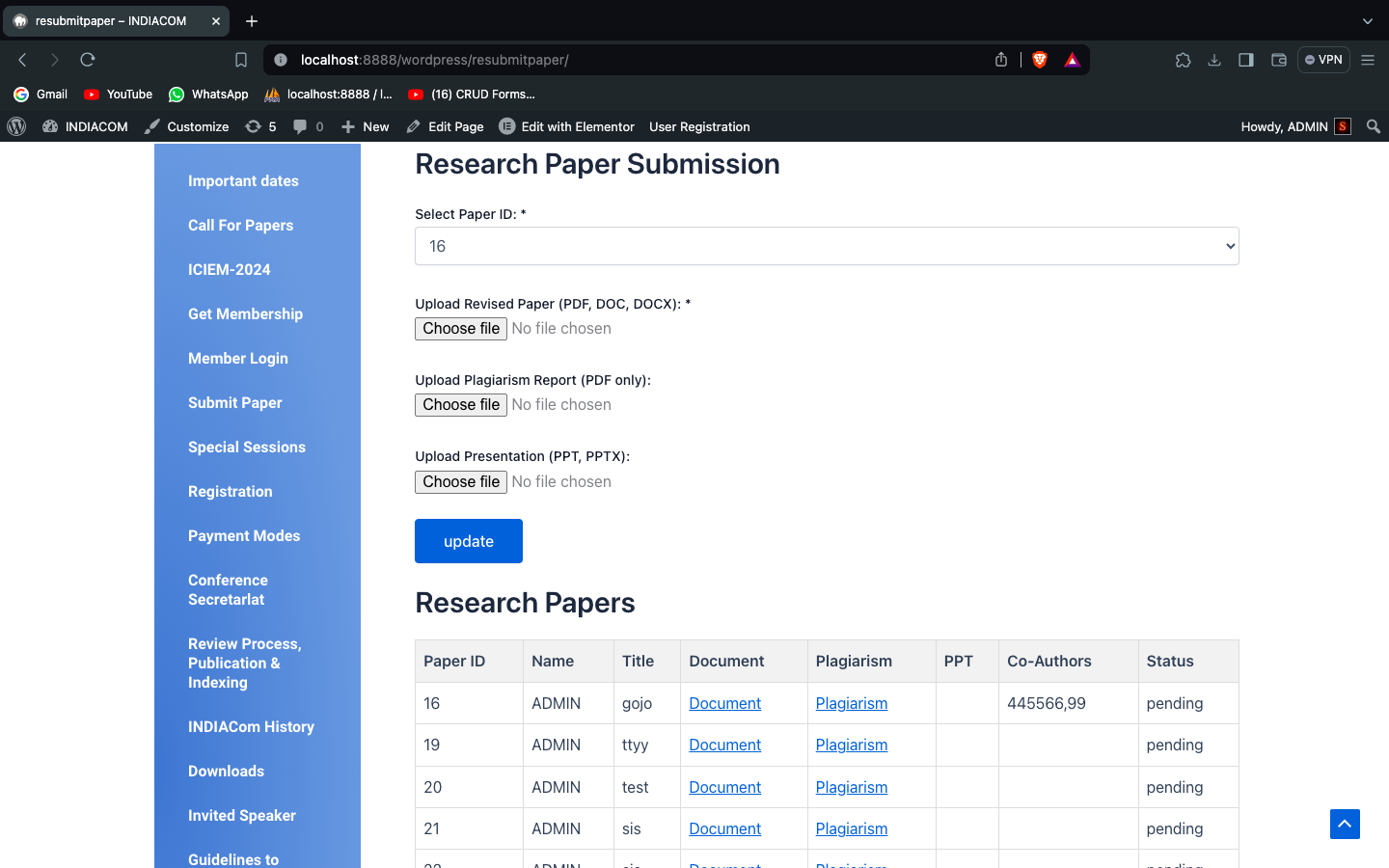
**Figure A-5.2: my account/login Page**

****

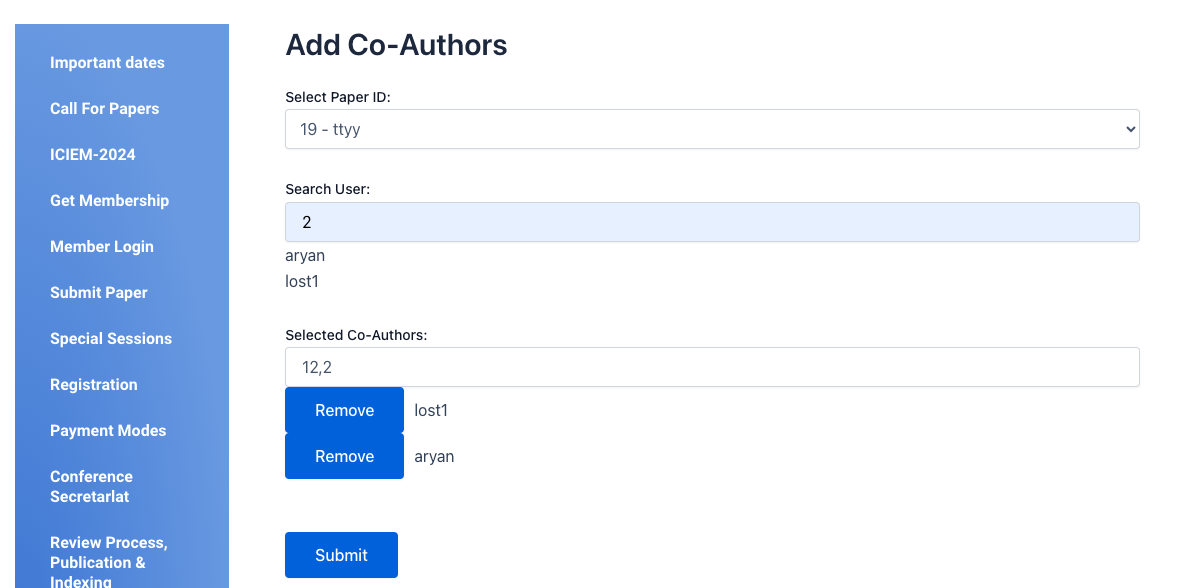
**Figure A-5.3: get membership Page**

****

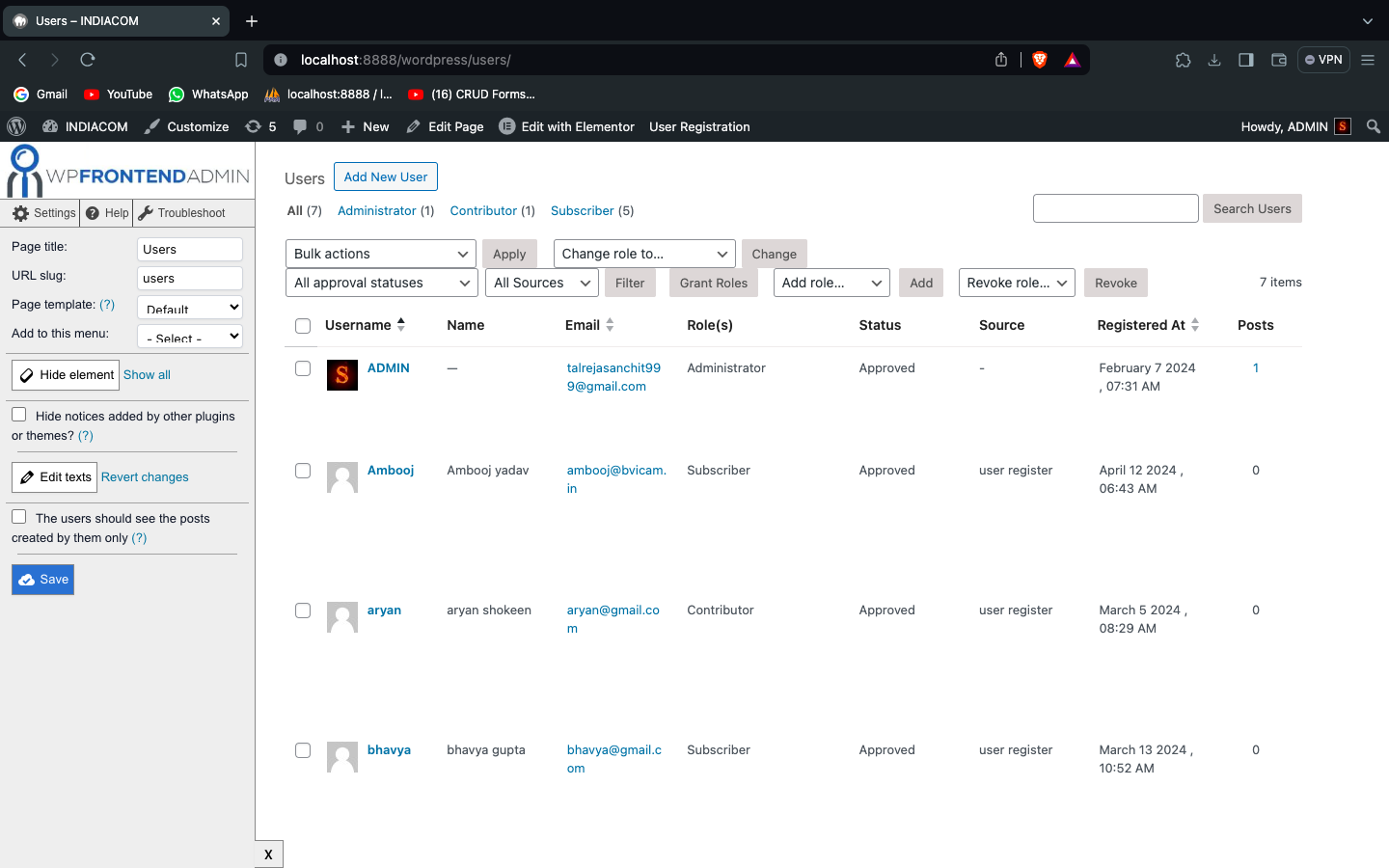
**Figure A-5.4: upload research paper Page**

****

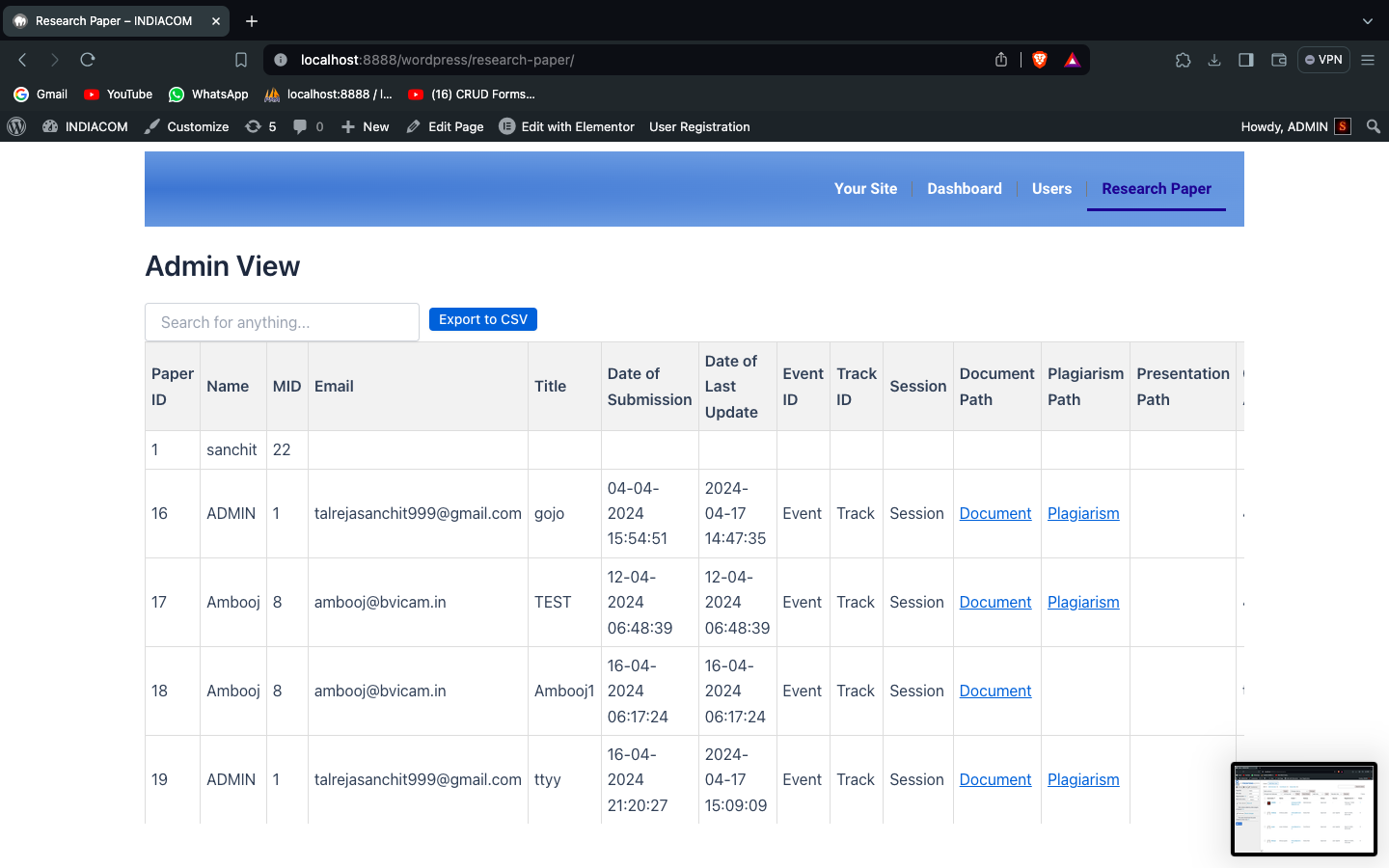
**Figure A-5.5: update research paper Page**

****

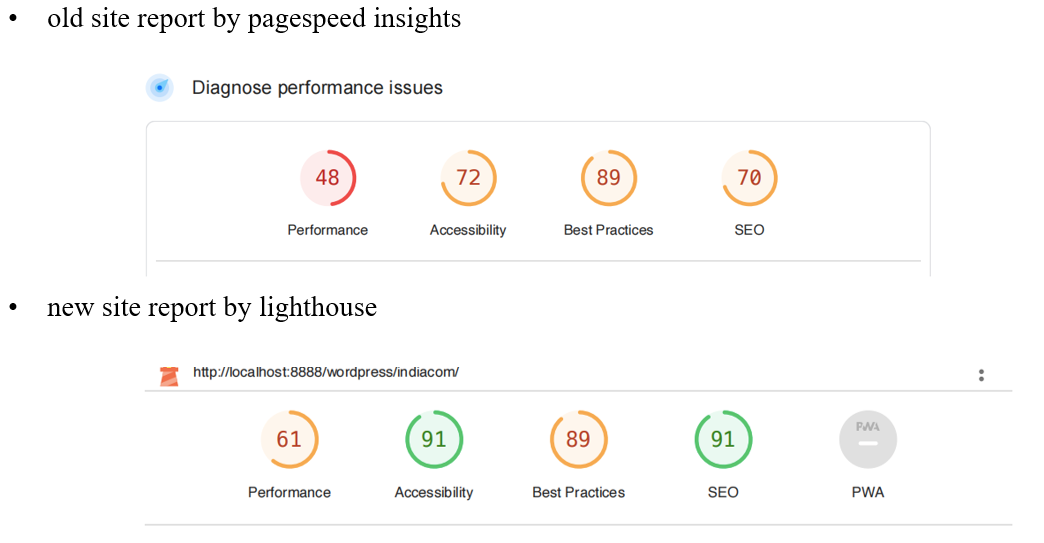
**Figure A-5.6: add co-authors Page**

****

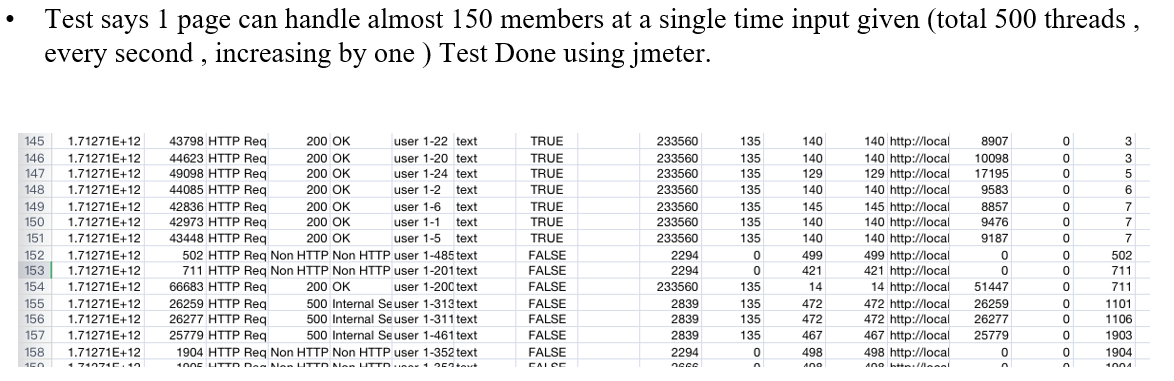
**Figure A-5.7: manage users Page**

****

**Figure A-5.8: manage research paper Pass**

****

**Figure A-5.9: generated reports**

****

**Figure A-5.10: load testing sheet**