

Chapter 1 : INTRODUCTION

1.1 Problem Statement:

The existing higher education system faces several challenges in effectively managing information, facilitating seamless communication, and providing easy access to campus resources for students. These challenges create a fragmented and disconnected campus experience, leading to decreased student engagement, inefficiencies in administrative processes, and limited utilization of campus services. To address these issues, there is a need for a comprehensive solution that leverages innovative technologies to create a smart campus platform specifically designed for students.

The specific problems that the Student Link project aims to address include:

1. Inefficient Information Management: The manual systems for managing student profiles, academic records, and administrative tasks are time-consuming, error-prone, and hinder the efficiency of information management processes. There is a need for a more streamlined and automated system to ensure easy access to accurate information.
2. Fragmented Communication Channels: Existing communication channels, such as email, notice boards, and physical announcements, result in fragmented and inconsistent communication across the campus. This leads to missed opportunities, confusion, and a lack of engagement among students and faculty.
3. Limited Accessibility to Campus Resources: Accessing campus resources, such as canteens, libraries, and information desks, can be challenging and time-consuming for students. The lack of centralized and user-friendly platforms leads to inefficiencies, long waiting times, and reduced utilization of these resources.
4. Disconnected Campus Experience: The combination of inefficient information management, fragmented communication, and limited accessibility results in a disconnected campus experience. Students may feel isolated, uninformed about events and activities, and face difficulties in engaging with the campus community.

The Student Link project aims to overcome these challenges by developing a Smart Campus Platform that integrates modules such as a blockchain-based student profile system, an online canteen system, an online library management system, and a chatbot for information and support.

By leveraging technology and providing a unified platform, the project aims to enhance student engagement, streamline administrative processes, and create a more connected and student-centric campus experience.

1.2 Aim and Objectives:

The aim of the Student Link project is to create a Smart Campus Platform specifically designed for students. The platform will leverage innovative technologies to address the challenges faced in higher education, with a focus on enhancing student engagement, streamlining administrative processes, and providing easy access to campus resources.

The objectives of the project include:

- Improving information management: By leveraging innovative technologies, the platform aims to streamline information management processes, ensuring that students have easy access to relevant and up-to-date information.
- Enhancing communication: The platform will provide seamless communication channels, enabling efficient and effective communication between students, faculty, and staff members.
- Increasing accessibility to resources: Through the development of online systems such as the JSS CANTEEN and SJCE LIBRARY PORTAL, the platform aims to improve accessibility to campus resources, allowing students to conveniently access meals and library materials.
- Fostering student engagement: The platform will facilitate student engagement by providing features such as the SJCE Chatbot, which can provide information about university events, clubs, and other happenings.
- Leveraging innovative technologies: The project will explore the use of technologies such as blockchain for secure student profile management and AI-powered chatbots for personalized assistance and support.

1.3 Applications:

1. Student Profile using Blockchain Technology:

The student profile module utilizes blockchain technology to create a secure and tamper-proof digital record of students' academic achievements and personal information. By leveraging blockchain's decentralized and transparent nature, this application ensures the integrity and authenticity of student profiles. Students can access and update their profiles, view their academic progress, and showcase their accomplishments to potential employers or educational institutions.

Here's more information on these technologies and their roles in the system:

1. Ganache: Ganache is a personal blockchain for Ethereum development. It allows for the creation of a local Ethereum network that can be used for testing and development purposes. In the context of the Student Profile System, Ganache is used to simulate the mining of blockchain blocks using Ethereum as the currency for transactions. This enables the system to create a secure and decentralized record of student profiles.
2. Metamask: Metamask is a browser extension that serves as a wallet and a connection between web applications and the Ethereum blockchain. It allows users to manage their Ethereum accounts, securely store private keys, and interact with decentralized applications (DApps). In the Student Profile System, Metamask is used to connect Ganache (the local blockchain) with the web application that hosts the student profiles. It provides a seamless and secure interface for students to access and update their profile information.
3. InterPlanetary File System (IPFS): IPFS is a decentralized file storage system that uses a peer-to-peer network to store and retrieve files. It enables the system to store student profile information in a distributed manner, ensuring redundancy and resilience against data loss. IPFS assigns a unique hash to each file, which serves as its identifier and allows for easy retrieval and verification.

2. JSS CANTEEN:

The JSS CANTEEN is an online canteen system designed to streamline the process of ordering meals for students. It offers a user-friendly interface where students can browse through the menu, place orders, and make payments digitally. This eliminates the need for physical queues

and reduces waiting times, providing a convenient and efficient dining experience. Students can also view nutritional information, track their order status, and provide feedback on their dining experience.

Here's more information on the features and benefits of the system:

1. Digital Menu: The online canteen system allows students to access a digital menu that provides a comprehensive list of food items available for ordering. Students can browse through various categories, such as breakfast, lunch, snacks, or beverages, and view detailed descriptions, ingredients, and pricing for each item.
2. Convenient Ordering: Students can place their food orders directly through the online platform. They can select the desired food items, specify any customizations or dietary preferences, and add them to their virtual cart. This eliminates the need to wait in long queues or physically visit the canteen to place an order.
3. Personalized Preferences: The system can offer personalized recommendations or highlight food options based on a student's dietary preferences, allergies, or previous ordering history. This feature helps students discover suitable meal choices and ensures a tailored dining experience.
4. Order Tracking and Updates: Once an order is placed, students can track the status of their order in real-time. They can receive updates on the preparation and delivery progress, estimated delivery time, and notifications when the order is ready for pickup or on its way for delivery. This transparency provides students with better visibility and reduces uncertainty regarding their orders.
5. Secure Payment Options: The online canteen system offers secure payment options, allowing students to make digital payments for their orders. They can use various payment methods such as credit/debit cards, digital wallets, or campus-specific payment systems. This eliminates the need for students to carry cash or visit a separate payment counter, adding convenience and security to the payment process.

6. Reduction in Waiting Times: By enabling students to order their meals in advance through the online platform, the system reduces waiting times at the canteen. Students can schedule their orders according to their preferred meal times or peak hours, ensuring that their meals are ready for pickup or delivery when they need them. This feature improves efficiency and reduces congestion at the canteen.

7. Feedback and Ratings: The system can incorporate a feedback and rating mechanism, allowing students to provide reviews and ratings for the ordered meals and overall dining experience. This feedback loop helps the canteen management gather insights, address any issues, and continuously improve the quality of food and services provided.

3. SJCE LIBRARY PORTAL:

The SJCE LIBRARY PORTAL is an online library management system that enhances the accessibility and utilization of library resources for students. It provides a centralized platform where students can search for books, journals, and other materials, check their availability, and make online reservations. The portal also allows students to track their borrowed items, renew loans, and receive notifications for due dates. This application simplifies the process of accessing and managing library resources, saving students time and effort.

Here's more information on the features and benefits of the system:

1. Online Book Borrowing: The system allows students to browse the library's collection and borrow books online. They can search for specific titles, authors, or subjects, view book availability, and initiate the borrowing process with just a few clicks. This eliminates the need for physical visits to the library for book borrowing, saving time and effort for students.

2. Reservation Capabilities: If a book is currently checked out by another student, the online library management system enables students to reserve the book. They can request a hold on the book and receive notifications when it becomes available. This ensures fair access to popular or limited copies of books and enhances the overall efficiency of the borrowing process.

3. Resource Availability Tracking: The system provides real-time information on the availability of library resources. Students can easily check the status of books, journals, or other materials to determine if they are currently available on the shelves or if they are checked out by other students. This feature helps students plan their research or study activities more effectively.

4. Online Renewal and Returns: The system allows students to renew their borrowed materials online if they need more time with the resources. This can be done without the need to physically visit the library or interact with library staff. Additionally, students can also initiate returns through the online platform, making the process more convenient and flexible.

5. Personalized Recommendations: The online library management system can provide personalized book recommendations based on a student's borrowing history, interests, or academic program. By analyzing the student's reading preferences and patterns, the system can suggest relevant books or related resources, helping students discover new materials aligned with their interests or academic needs.

6. Digital Resource Access: In addition to physical books, the system may also provide access to digital resources such as e-books, scholarly articles, online journals, and databases. Students can access these resources remotely, allowing them to conduct research or access study materials from anywhere and at any time.

4. SJCE Chatbot:

The SJCE Chatbot serves as a virtual assistant and information provider for students. Using artificial intelligence, the chatbot can understand and respond to queries related to the university, events, clubs, and other campus-related information. Students can interact with the chatbot through a messaging interface and receive instant and accurate responses to their queries. The chatbot can provide information on upcoming events, club activities, campus facilities, academic deadlines, and more, enhancing student engagement and facilitating their involvement in campus. Here is some more information on the chatbot's functionality and benefits:

1. Instant and Accurate Information: The chatbot is designed to provide real-time responses to student queries, ensuring they receive instant access to the information they need. Whether it's

finding details about upcoming events, club activities, campus facilities, or general university information, the chatbot can quickly retrieve the relevant information and present it to the student.

2. 24/7 Availability: The chatbot operates round the clock, allowing students to seek information at any time of the day, even outside regular office hours. This availability enhances accessibility and ensures that students can get the information they need, irrespective of their schedule or location.

3. Personalized Assistance: The chatbot can offer personalized recommendations and suggestions based on a student's preferences and interests. It can provide tailored information about clubs, events, and activities that align with the student's profile, helping them discover new opportunities and enhance their engagement with the campus community.

4. Seamless Integration with Campus Services: The chatbot can be integrated with various campus services and systems, allowing students to perform actions and access resources directly through the chat interface. For example, students can check their course schedules, submit assignments, reserve study rooms, or access online resources through the chatbot, eliminating the need to navigate multiple platforms.

5. Enhanced Student Engagement: By providing an interactive and conversational interface, the chatbot promotes student engagement and involvement in campus activities. It serves as a friendly and accessible resource that encourages students to explore different opportunities, participate in events, join clubs, and make the most out of their university experience.

6. Efficient and Scalable Support: The chatbot can handle a large volume of inquiries simultaneously, providing efficient support to multiple students at once. This scalability ensures that students receive timely responses and reduces the burden on administrative staff who may otherwise be overwhelmed with repetitive queries.

These applications collectively contribute to creating a more connected and efficient campus experience for students, enabling them to access important services, resources, and information seamlessly.

1.4 Existing Solution:

The existing solutions in higher education often involve manual and traditional methods of managing various aspects of campus operations. Here is some more information on the existing solutions related to student profiles, canteen operations, library services, and information provision:

1. Student Profiles: Many institutions rely on manual systems for managing student profiles, which typically involve paperwork and physical files. These profiles may be stored in centralized databases or paper-based records, making it time-consuming and cumbersome to update and access student information. This can lead to inefficiencies, data inaccuracies, and difficulties in maintaining the security and integrity of student records.

2. Canteen Operations: Traditional canteens in higher education institutions often operate with physical queues and manual order processing. Students typically need to wait in line to place their orders, which can result in long waiting times and overcrowding during peak hours. The payment process may involve cash transactions or separate payment counters, adding to the complexity and potential for errors. These manual operations can be inefficient, prone to delays, and may not meet the evolving expectations of digitally savvy students.

3. Library Services: Libraries in higher education institutions may have limited digital access to their resources. Students often need to physically visit the library to search for books, check their availability, and borrow materials. The process of borrowing books may involve manual paperwork and barcode scanning. This can be inconvenient for students who prefer digital access and remote availability of library resources.

4. Information Provision: Institutions often rely on human staff, such as receptionists or administrative personnel, to provide information and assistance to students. This can lead to delays and inconsistencies in responses, especially during busy periods or outside regular

working hours. Students may face challenges in accessing accurate and up-to-date information about university events, clubs, academic matters, and other campus-related queries.

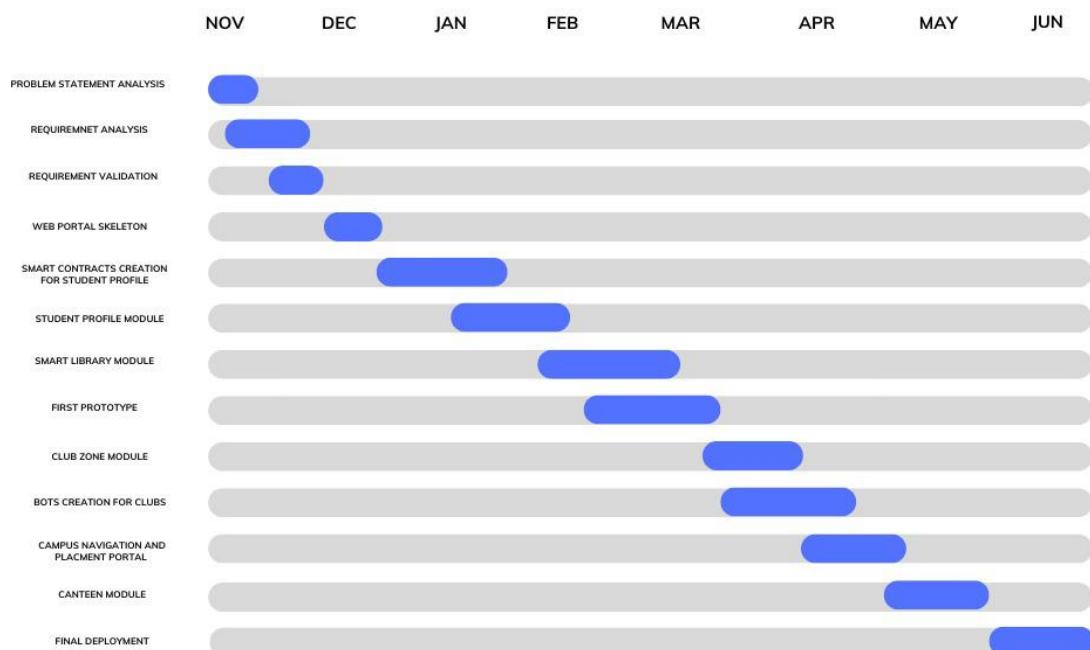
1.5 Proposed Solution:

The proposed Smart Campus Platform, "Student Link," offers an integrated and technologically advanced solution to overcome the limitations of existing systems. It introduces modules such as the blockchain-based student profile system for secure and efficient management of student data. The online canteen system (JSS CANTEEN) and library portal (SJCE LIBRARY PORTAL) bring convenience and accessibility to students in terms of meals and library resources. The implementation of the SJCE Chatbot provides instant and accurate information, enabling students to access relevant information quickly and efficiently.

1.6 Gantt Chart:

The Gantt chart helps in identifying critical milestones, ensuring that tasks are completed in a timely manner, and facilitating effective collaboration among team members. It serves as a roadmap for the project, allowing for efficient project management and successful delivery of the Smart Campus Platform.

WORK PLAN



CHAPTER 2 : LITERATURE REVIEW

SURVEY PAPER - 1

TITLE - “Online College Portal”

AUTHOR/S - Tejaswini Chavan , Deb Dutta, Michelle Gomez and Alvino Vaz

Description - The major purpose of the online college portal system, which strives to increase the effectiveness of college information management, is to handle and keep notifications and information. Two key functional requirements for the system are the administrator and the students. In comparison to users, the Administrator will have more access (enable/disable/update). This document helps to automate the manual system that is already in place. This work is digital. Remote monitoring and control are both possible. It gives precise information while requiring fewer steps. The gateway can be made more authentic while malpractice can be decreased.

This study contributes to the automation of the current manual system. This is a paperless work. It can be monitored and operated remotely. It reduces the number of employees required and delivers reliable information. Malpractice can be reduced. All information obtained throughout the years can be saved and retrieved at any time. As a result, the data saved in the repository assists management in making decisions. As a result, a Web-based system is preferable. All stakeholders, faculty, and management can obtain the necessary information without delay. This system is vital at colleges and universities.

SURVEY PAPER - 2

TITLE - “A Systematic Review on Technologies and Applications in Smart Campus: A Human-Centered Case Study”

AUTHOR/S - Yuchen Zhang, Christine Yip, Erwan Lu, Zhao Yang Dong

This paper focuses on a thorough analysis of the technological tools employed in the creation of a smart campus that is primarily human-centered. Campus closure had been sporadically applied in 188 countries due to the COVID-19 pandemic, which has had an impact on the health of hundreds of millions of students. Emerging ICT technologies have been thoroughly studied, pushed, and utilised in the field of education to deal with the effects and challenges of the long-term school closures, and the pandemic has brought smartness in education to a new level of prominence. It throws light on significant modern information and communication technology

(ICT) as well as intelligent software that can be used to upgrade campuses to the next level of smartness and to build intelligent communities for its patrons and students.

3. Creating a Student's Profile using Blockchain Technology

The research paper deals with data extraction and security. Due to the amount and accuracy of the data, data collecting is a very important activity in the higher education industry. The data must be carefully examined and protected with advanced security measures. One of the most important technologies that has changed how data is seen is blockchain. It has numerous uses in education, namely for keeping student profiles.

4. Chatbot Development for Educational Institute

Numerous programmes attempt to replicate human interaction by integrating a human appearance. We can create a variety of chatbots utilising AI. The paper is based on the observation made on a chatbot that was created to resolve queries regarding college. The fields include Enquiry process, Fees structure, Course specifics, Description of the eligibility requirements, and Admission. In order to create a dialogue operator that could be used in college, this paper demonstrates how we can deal with identifying the most important realities in works that portray the life of an actual individual.

5. AI And Web-Based Interactive College Enquiry Chatbot

Artificial intelligence is now playing a significant role in a wide range of disciplines, from industries in product manufacture to customer care in public relations. There are numerous online Artificial Intelligence (AI) systems or chat bots that assist users in solving their problems. This paper deals with building an AI-powered virtual assistant that can answer any college-related question. This will function as a College Oriented Intelligence machine.

Chapter 3: System Requirements and Analysis

3.1 Input Requirements:

Input requirements refer to the data and information that the Smart Campus Platform "Student Link" needs to receive and process. Some examples of input requirements for the project include:

- Student Profile Information: The system needs to collect and store student information such as personal details, academic records, and other relevant data. This information can be inputted by students themselves or imported from existing university databases.
- Menu and Food Item Details: The online canteen system requires input regarding menus, food items, their descriptions, prices, and availability. This information can be provided by the canteen staff or administrators.
- Library Resources: The library management system needs input on books, journals, and other resources available in the library. This information may include titles, authors, ISBN/ISSN numbers, publication details, and categorization.
- User Queries and Interactions: The chatbot module requires input from users in the form of questions, queries, or requests for information. The system needs to process and analyze these inputs to provide appropriate responses.

3.2 Output Requirements:

Output requirements define the expected results and information that the Smart Campus Platform "Student Link" should produce. Some examples of output requirements for the project include:

- Student Profiles: The system should generate and display student profiles, including their personal information, academic records, and any additional details relevant to their campus activities.
- Online Canteen Orders: The system should generate order confirmations and receipts for students who place orders through the online canteen system. These outputs may include details of the ordered items, payment information, and estimated delivery/pickup times.

- Library Resource Availability: The system should provide real-time information on the availability of library resources. This can include displaying the status of books (available, on loan, reserved) and providing due dates for borrowed materials.
- Chatbot Responses: The system should generate accurate and timely responses to user queries and interactions with the chatbot. The responses may include text-based information, links to relevant resources, or notifications about upcoming events or deadlines.

3.3 Functional Requirements:

Functional requirements describe the specific functions and capabilities that the Smart Campus Platform "Student Link" should possess. Some examples of functional requirements for the project include:

- User Registration and Authentication: The system should allow students to create accounts, log in securely, and manage their profiles. It should implement authentication mechanisms to ensure data privacy and security.
- Student Profile Management: The system should provide features for students to update their profile information, add academic achievements, and track their progress.
- Online Canteen Ordering: The system should enable students to browse menus, select food items, customize orders, make payments securely, and track the status of their orders.
- Library Resource Management: The system should facilitate search and retrieval of library resources, allow students to request book reservations, manage borrowing and returning of materials, and send reminders for due dates.
- Chatbot Functionality: The system should incorporate natural language processing and AI capabilities to understand user queries, provide relevant information, and engage in interactive conversations.

4.4 Non-Functional Requirements:

Non-functional requirements specify the quality attributes and constraints that the Smart Campus Platform "Student Link" should adhere to. Some examples of non-functional requirements for the project include:

- Security and Privacy: The system should implement robust security measures to protect sensitive student information, financial transactions, and user data. It should comply with relevant data protection regulations.

- Usability and User Experience: The system should have an intuitive and user-friendly interface that is easy to navigate and understand. It should provide clear instructions, feedback, and error handling to enhance the user experience.
- Performance and Scalability: The system should be capable of handling a large number of concurrent users, processing requests quickly, and ensuring a responsive user interface. It should be scalable to accommodate future growth and increasing demands.
- Reliability and Availability: The system should be reliable, ensuring minimal downtime and disruptions. It should have backup and recovery mechanisms in place to handle system failures or data loss.
- Compatibility and Integration: The system should be compatible with different web browsers, operating systems, and devices to ensure broad accessibility. It should also have integration capabilities to interact with other existing campus systems or external services.

3.5 Software Requirements:

Software requirements outline the specific software components and technologies required for the Smart Campus Platform "Student Link" project. Some software requirements for the project may include:

- Programming Languages: The system may require programming languages such as Java, Python, or JavaScript for backend and frontend development.
- Web Development Frameworks: Frameworks like Django, Flask, or Node.js may be utilized for web application development.
- Database Management System: A database management system like MySQL, PostgreSQL, or MongoDB may be used to store and manage data.
- Blockchain Technology: For the secure student profile system, blockchain platforms such as Ethereum may be employed along with tools like Ganache and Metamask.
- Natural Language Processing and AI: Libraries and frameworks such as NLTK (Natural Language Toolkit), TensorFlow, or Dialogflow may be used for chatbot functionality and AI capabilities.

3.6 Hardware Requirements:

Hardware requirements specify the necessary physical components and infrastructure for hosting and running the Smart Campus Platform "Student Link." Some hardware requirements for the project may include:

- Servers: The system may require dedicated servers or cloud-based server infrastructure to host and deploy the web application and databases.
- Network Infrastructure: A stable network connection with adequate bandwidth is necessary to ensure seamless communication between users and the system.
- End-User Devices: The system should be compatible with various devices such as desktop computers, laptops, tablets, and smartphones, allowing users to access the platform from different devices and operating systems.
- Storage Devices: Sufficient storage capacity is needed to store data, including student profiles, canteen orders, library resources, and chatbot interactions.

Chapter 5 : Tools and Technology Used

The selection of these tools and technologies indicates the project's focus on utilizing blockchain technology for secure student profiles, web development frameworks for frontend and backend development, and machine learning capabilities for the chatbot module.

1. Ganache: Ganache is a personal blockchain for Ethereum development. It provides a local development environment where you can test and deploy smart contracts, mine blocks, and simulate Ethereum network behavior. It is often used for testing and debugging blockchain applications.

2. Truffle: Truffle is a development framework for Ethereum that provides tools and utilities for building, testing, and deploying smart contracts. It simplifies the process of creating and managing blockchain applications by providing a suite of development tools and a standardized project structure.

3. Metamask: Metamask is a browser extension that serves as a cryptocurrency wallet and allows users to interact with the Ethereum blockchain. It provides a secure way to store private keys, manage Ethereum accounts, and seamlessly connect with decentralized applications (DApps) through the browser.

4. Solidity Programming: Solidity is a programming language specifically designed for writing smart contracts on the Ethereum blockchain. It is used to define the behavior and logic of smart contracts, enabling the execution of decentralized applications on the Ethereum network.

5. React.js: React.js is a JavaScript library for building user interfaces. It is widely used for creating interactive and dynamic web applications. React.js provides a component-based architecture that enables developers to build reusable UI components, making the development process more efficient and maintainable.

6. PHP: PHP is a server-side scripting language commonly used for web development. It is known for its ease of use, flexibility, and wide support in the web development community. PHP can be used for server-side processing and handling of data in applications such as the online canteen system.

7. Node.js: Node.js is a JavaScript runtime built on Chrome's V8 JavaScript engine. It allows developers to run JavaScript on the server-side, enabling the development of scalable and efficient web applications. Node.js is often used for server-side scripting and building backend APIs in projects.

8. MySQL: MySQL is an open-source relational database management system. It is widely used for storing and managing structured data in various applications. In the context of the project, MySQL may be utilized as the database system for storing and managing data related to the library portal.

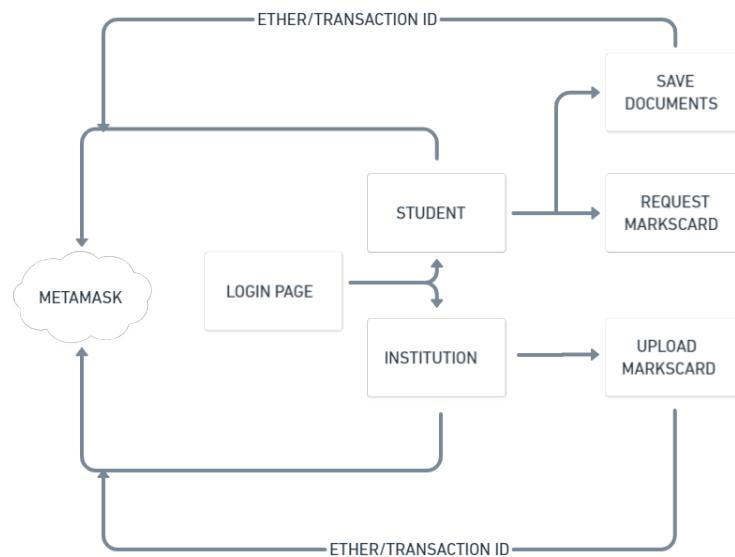
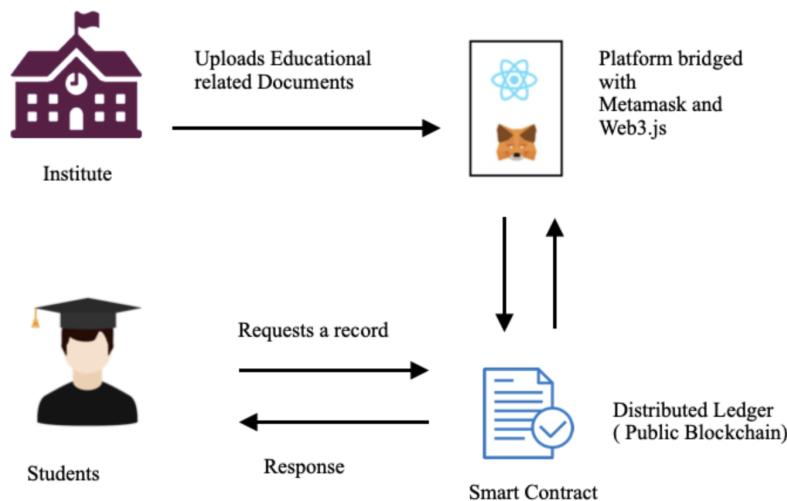
9. Python: Python is a versatile programming language with a wide range of applications. It is known for its simplicity, readability, and extensive library support. In the project, Python may be used for developing the chatbot module, utilizing libraries such as Flask for web development and TensorFlow for natural language processing and AI capabilities.

10. Flask: Flask is a lightweight web framework for Python. It is designed to be simple and easy to use, making it suitable for developing small to medium-sized web applications. Flask can be used to build the backend of the chatbot module, handle HTTP requests, and interact with other components.

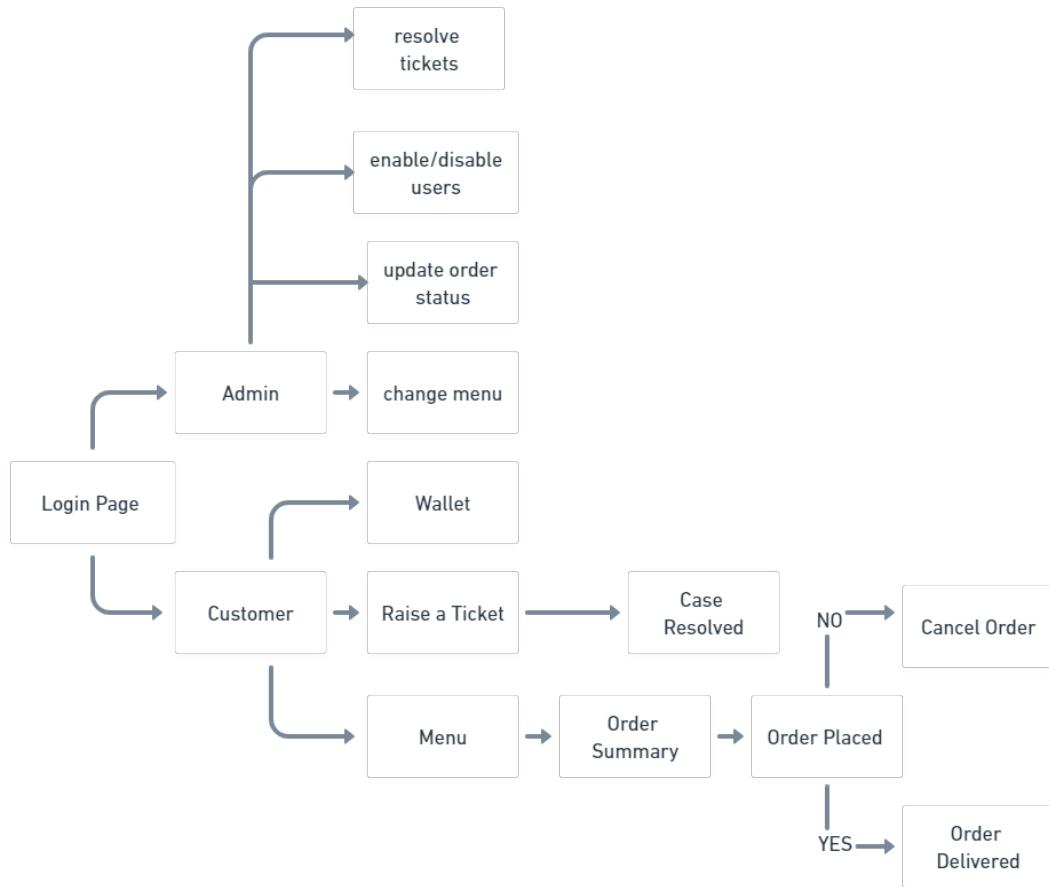
11. TensorFlow: TensorFlow is an open-source machine learning framework developed by Google. It provides a comprehensive ecosystem for building and deploying machine learning models. In the project, TensorFlow may be used for implementing natural language processing algorithms and training the chatbot model.

Chapter 6 : System Design

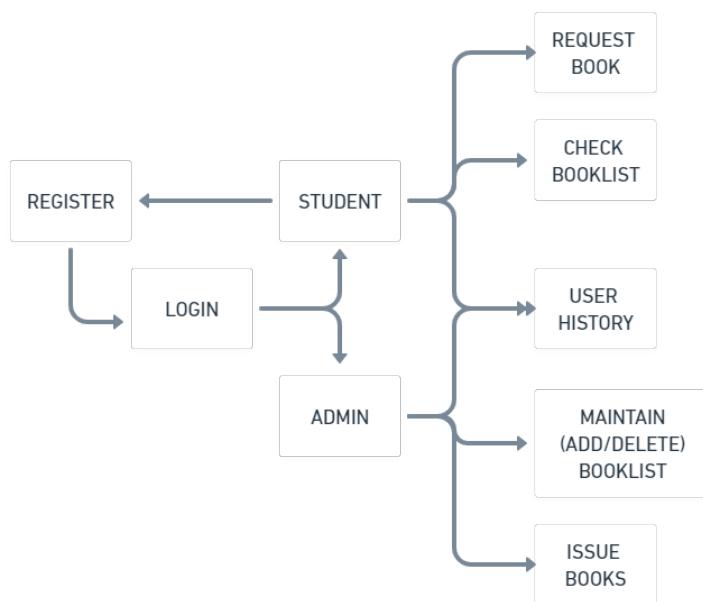
Student Profile using Blockchain Technology



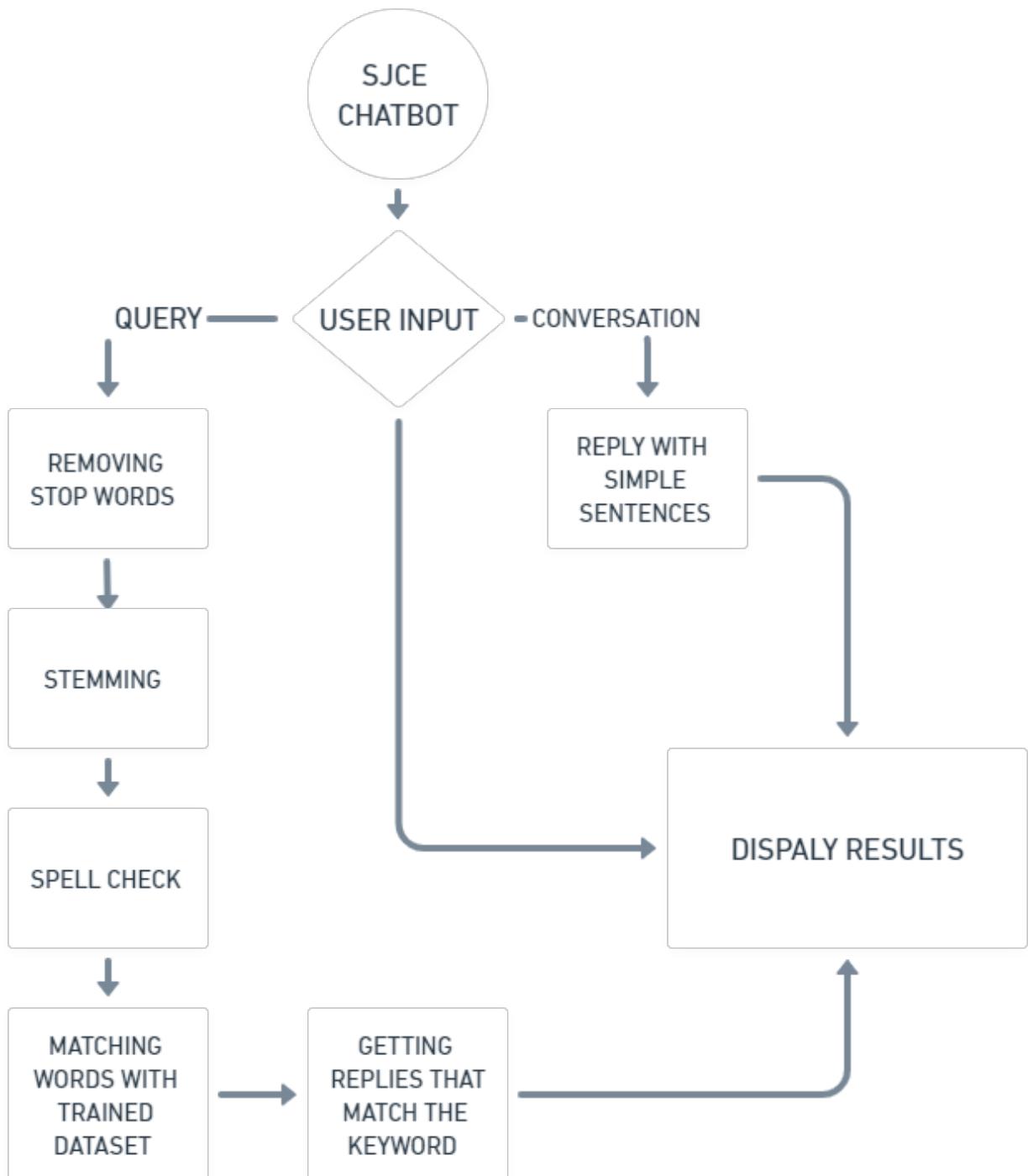
JSS Canteen Design



SJCE LIBRARY PORTAL



SJCE Chatbot Design



Chapter 7 : System Implementation

Student Profile module using Blockchain Technology

It involves the following steps:

1. Requirement Gathering:

- Understand the requirements and objectives of implementing a blockchain-based student profile system.
- Determine the data to be stored in the student profile, such as academic achievements, personal details, and other relevant information.

2. Blockchain Selection:

- Choose an appropriate blockchain platform, such as Ethereum, to develop the student profile system.
- Consider factors like scalability, security, and community support when selecting the blockchain platform.

3. Smart Contract Development:

- Develop smart contracts using Solidity programming language to define the data structure and operations for the student profile.
- Implement functions for adding, updating, and retrieving student profile information.
- Define access control mechanisms to ensure data privacy and security.

4. Blockchain Network Setup:

- Set up a private or permissioned blockchain network using tools like Ganache or Truffle.

- Configure the network with appropriate consensus algorithms, network nodes, and network parameters.

5. Integration with User Interface:

- Develop a user interface (UI) for students to interact with the blockchain-based student profile system.
- Implement web or mobile UI components using technologies like React.js or Angular.
- Integrate the UI with the blockchain network using tools like MetaMask to enable user interactions.

6.. User Authentication and Access Control:

- Implement authentication mechanisms to verify the identity of users accessing the student profile system.
- Define access control rules to allow only authorized users to modify or access specific profile information.

```

1 pragma solidity ^0.5.0;
2 pragma experimental ABIEncoderV2;
3
4 contract EducationContract {
5     struct RecordDetails {
6         string record_id;
7         string record_name;
8         string description;
9         string timestamp;
10        string ipfsHash;
11    }
12
13    RecordDetails[] public record;
14    mapping(address => RecordDetails[]) public students;
15
16    function addReport(string memory _record_id,address _student, string memory _record_name, string memory _description, string memory _timestamp, string memory _ipfsHash) {
17        students[_student].push(RecordDetails(_record_id,_record_name,_description,_timestamp,_ipfsHash));
18        return 1;
19    }
20
21    //function getPatCount() public view returns(uint) {
22    //    return record.length;
23    //}
24
25
26
27    function getAllRecordDetails(address _id) public view returns (RecordDetails[] memory)
28    {
29        RecordDetails[] memory id = new RecordDetails[](students[_id].length);
30        for (uint i = 0; i < students[_id].length; i++) {
31            RecordDetails storage tempRecord = students[_id][i];
32            id[i] = tempRecord;
33        }
34        return id;
35    }
36 }
```

JSS CANTEEN

It involves the following steps:

1. Requirement Gathering:

- Understand the requirements and objectives of the online canteen system.
- Determine the features and functionalities to be included, such as menu display, online ordering, payment processing, and order tracking.

2. Technology Selection:

- Choose appropriate technologies for developing the online canteen system, such as PHP or Node.js for backend development, MySQL for the database management system, and HTML/CSS/JavaScript for frontend development.
- Consider using frameworks like Laravel or CodeIgniter for efficient development and code organization.

3. Database Setup:

- Create the necessary database tables to store information about food items, orders, customers, and transactions.
- Define relationships between the tables to ensure data consistency and integrity.

4. User Interface Design:

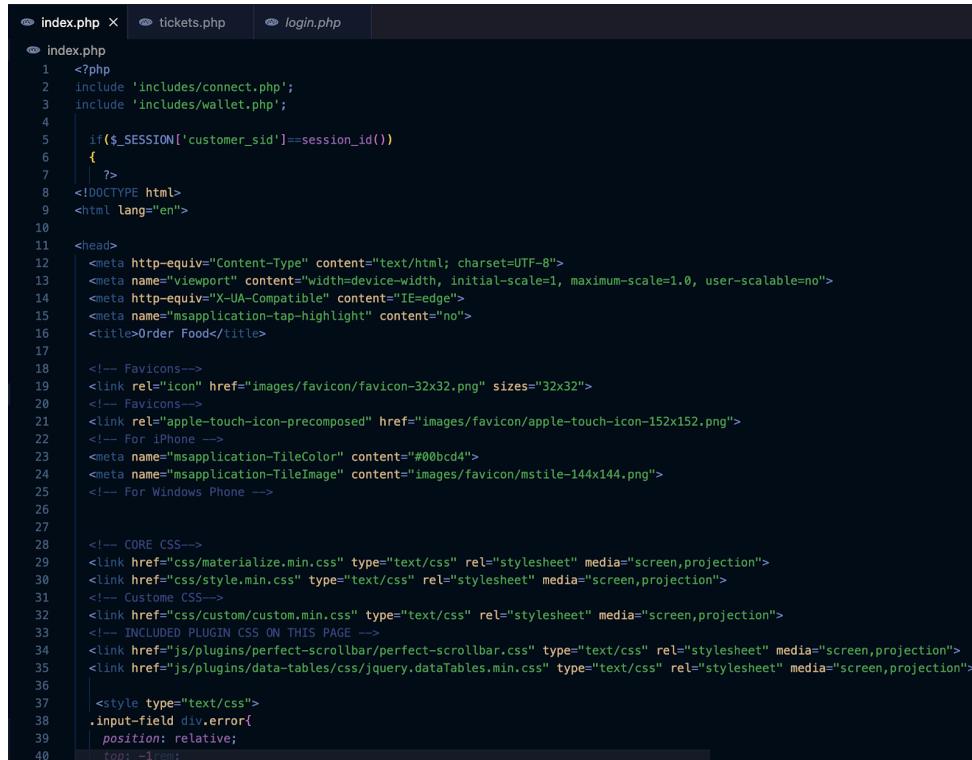
- Design a user-friendly and visually appealing interface for the online canteen system.
- Create interactive and intuitive forms for users to browse menus, select food items, and place orders.
- Implement responsive design to ensure compatibility with different devices and screen sizes.

5. Backend Development:

- Implement server-side functionality using PHP to handle user requests, process data, and interact with the database.
- Implement security measures, such as user authentication and authorization, to protect sensitive data.

6. Order Management and Tracking:

- Implement features to manage and track food orders, including order placement, order status updates, and order history.
- Develop functionality to notify users about the status of their orders, such as order confirmation, preparation, and delivery.



```

index.php X tickets.php login.php
index.php
1 <?php
2 include 'includes/connect.php';
3 include 'includes/wallet.php';
4
5 if($_SESSION['customer_sid']==session_id())
6 {
7 |
8 <!DOCTYPE html>
9 <html lang="en">
10
11 <head>
12 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
13 <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1.0, user-scalable=no">
14 <meta http-equiv="X-UA-Compatible" content="IE=edge">
15 <meta name="msapplication-tap-highlight" content="no">
16 <title>Order Food</title>
17
18 <!-- Favicons-->
19 <link rel="icon" href="images/favicon/favicon-32x32.png" sizes="32x32">
20 <!-- Favicons-->
21 <link rel="apple-touch-icon-precomposed" href="images/favicon/apple-touch-icon-152x152.png">
22 <!-- For iPhone -->
23 <meta name="msapplication-TileColor" content="#00bcd4">
24 <meta name="msapplication-TileImage" content="images/favicon/mstile-144x144.png">
25 <!-- For Windows Phone -->
26
27
28 <!-- CORE CSS-->
29 <link href="css/materialize.min.css" type="text/css" rel="stylesheet" media="screen,projection">
30 <link href="css/style.min.css" type="text/css" rel="stylesheet" media="screen,projection">
31 <!-- Custom CSS-->
32 <link href="css/custom/custom.min.css" type="text/css" rel="stylesheet" media="screen,projection">
33 <!-- INCLUDED PLUGIN CSS ON THIS PAGE -->
34 <link href="js/plugins/perfect-scrollbar/perfect-scrollbar.css" type="text/css" rel="stylesheet" media="screen,projection">
35 <link href="js/plugins/data-tables/css/jquery.dataTables.min.css" type="text/css" rel="stylesheet" media="screen,projection">
36
37 <style type="text/css">
38 .input-field div.error{
39   position: relative;
40   top: -1rem;

```

SJCE LIBRARY PORTAL

System implementation for the SJCE Library portal in the Student Link project involves the following steps:

1. System Design:

- Define the functionalities and features of the library portal, such as online book searching, borrowing, reservation, and resource availability tracking.
- Determine the user interface design, considering usability and ease of navigation for students.
- Design the database schema to store and manage library resources, user profiles, and transactional data.

2. Technology Selection:

- Choose appropriate technologies for developing the library portal, such as PHP for backend development, MySQL as the database management system, and HTML/CSS/JavaScript for frontend development.
- Utilize popular libraries and frameworks like Laravel or CodeIgniter for efficient development and code organization.

3. Database Setup:

- Create the necessary database tables to store information about books, users, reservations, and transactions.
- Establish relationships between the tables to ensure data consistency and integrity.

4. Frontend Development:

- Develop the user interface for the library portal using HTML, CSS, and JavaScript.
- Implement features like search functionality, book browsing, user registration and login, and reservation management.
- Design intuitive and user-friendly forms and interfaces for users to interact with the portal.

5. Backend Development:

- Implement server-side functionality using PHP to handle user requests, process data, and interact with the database.
- Develop APIs and web services to enable communication between the frontend and backend components.
- Implement security measures, such as user authentication and authorization, to protect user data and ensure privacy.

6. Integration and Deployment:

- Integrate the library portal with the Student Link platform, ensuring seamless navigation and interaction between different modules.
- Deploy the system on a suitable web server, ensuring it is accessible to users.
- Configure server settings, domain names, and SSL certificates to ensure secure and reliable access to the library portal.

7. Testing and Quality Assurance:

- Conduct thorough testing of the library portal's functionalities, ensuring they work as expected.
- Test different scenarios, such as book searching, borrowing, reservation, and transaction handling, to validate the system's reliability and accuracy.
- Identify and fix any bugs or issues that arise during testing.

8. User Training and Support:

- Provide training and support to library staff and users on how to effectively use the library portal.
- Develop user documentation and guides to assist users in navigating and utilizing the portal's features.

9. Continuous Improvement:

- Gather feedback from users and stakeholders to identify areas for improvement and enhancement.
- Regularly update the library portal to address user needs, improve performance, and incorporate new features and functionalities.

SJCE CHATBOT

It involves the following steps:

1. Designing the Chatbot:

- Define the chatbot's functionalities, such as providing information about the university, events, clubs, and other campus-related queries.
- Determine the chatbot's conversational flow and dialog structure.
- Design the user interface for the chatbot, considering usability and user experience.

2. Technology Selection:

- Choose appropriate technologies for developing the chatbot, such as Python for backend development and Flask as the web framework.
- Utilize TensorFlow, a popular machine learning library, for natural language processing and building the chatbot's conversational capabilities.

3. Data Collection and Preparation:

- Gather relevant data and information about the university, events, clubs, and other campus-related details.
- Prepare the data by structuring it in a format suitable for training the chatbot's machine learning model.
- Clean and preprocess the data to improve its quality and ensure accurate responses.

4. Training the Chatbot:

- Use machine learning techniques to train the chatbot on the collected and preprocessed data.
- Implement natural language processing algorithms to understand user queries and generate appropriate responses.
- Fine-tune the chatbot's model using techniques like supervised learning or reinforcement learning to improve its conversational abilities.

5. Integration and Deployment:

- Integrate the chatbot with the Student Link platform, ensuring seamless communication between the chatbot and other system modules.
- Deploy the chatbot on a suitable platform or hosting service, making it accessible to users.
- Ensure the chatbot is properly connected to the platform's user interface and can receive and respond to user queries effectively.

6. Testing and Evaluation:

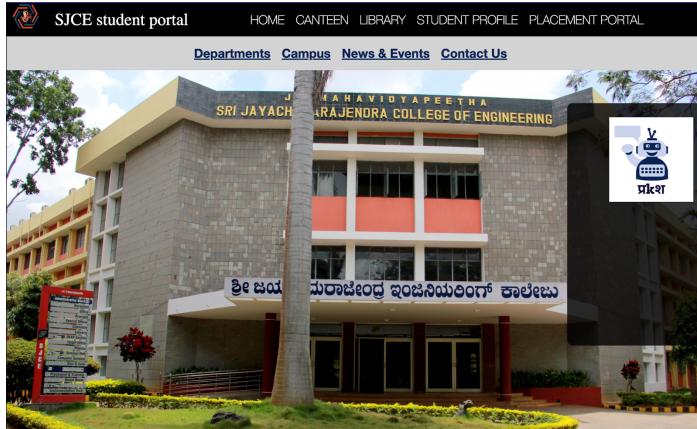
- Conduct thorough testing of the chatbot's functionality, ensuring it provides accurate and relevant responses to user queries.
- Validate the chatbot's performance against predefined test cases and user scenarios.
- Gather feedback from users to evaluate the chatbot's effectiveness and make any necessary improvements.

7. Continuous Improvement:

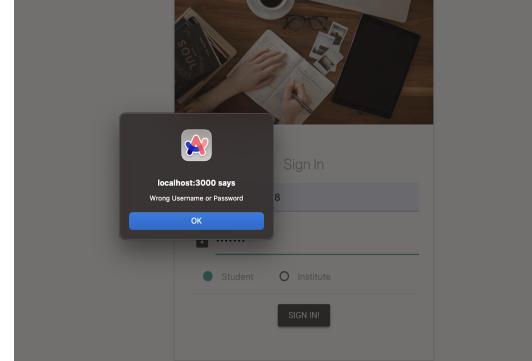
- Monitor the chatbot's performance in real-world usage and collect user feedback to identify areas for improvement.
- Regularly update and enhance the chatbot's capabilities based on user feedback, emerging needs, and advancements in technology.

Chapter 8 : System Testing and Result Analysis

Screenshots of test cases of all 4 modules with caption for them



Student portal homepage



login page(wrong password)

A screenshot of a "Records" page. At the top, there's a header with "Records", "Home", "Request Record", and "Log out". Below the header is a section titled "List of records" with a table. The table has columns for "Institute ID", "Record Name", "Description", and "Created Timestamp". One row in the table shows "123", "marks card", "one marks card for 1st sem", and "2023-01-15 11:23:45". At the bottom of the table are two buttons: a green "IPFS" button and a green "Download" button.

record page record retrieval

A screenshot of a "New Record" page. It has fields for "USN" (containing "01JST19CS128") and "Institute Address" (containing "0x171c4a223009340063f3A3fDD60b5217C5"). Below these is a timestamp "2023-06-17 22:59:18". There's a "Record Name" field with "2ND YEAR MARKS CARD" and a green "Upload to Blockchain" button at the bottom.

record request

A screenshot of a "Record upload page(institute)". It has a header with "Records", "Home", and "Logout". Below the header is a "Upload Education Reports" section with an "INSTITUTE ID" dropdown set to "IN123". There are fields for "STUDENT ADDRESS KEY*" (containing "0x0209fb39fc064e7045822596cfa4aa6a6e27830c"), "RECORD NAME*", "DESCRIPTION*", and "TIMESTAMP*". The "RECORD NAME*" field contains "3RD YEAR MARKS CARD", the "DESCRIPTION*" field contains "2ND YEAR MARKS CARD", and the "TIMESTAMP*" field contains "2023-06-17 23:04:57". At the bottom is a green "Upload to Blockchain" button.

Record upload page(institute)

Login page(JSS CANTEEN)

ordering items selection page

Tickets page

orders page

Edit details Page

SJCE LIBRARY PORTAL

ADMIN LOGIN | USER SIGNUP | USER LOGIN | LOG ME OUT

ADMIN LOGIN FORM

LOGIN FORM

Enter Username

Password

Verification code : 50306

LOGIN

SJCE LIBRARY PORTAL

DASHBOARD | CATEGORIES | AUTHORS | BOOKS | ISSUE BOOKS | REG STUDENTS | CHANGE PASSWORD | LOG ME OUT

ADMIN DASHBOARD

Books Listed
2

Times Book Issued
6

Times Books Returned
3

Registered Users
6

Authors Listed
2

Listed Categories
6

Admin login page

SJCE LIBRARY PORTAL

DASHBOARD | CATEGORIES | AUTHORS | BOOKS | ISSUE BOOKS | REG STUDENTS | CHANGE PASSWORD | LOG ME OUT

MANAGE ISSUED BOOKS

Success : Book issued successfully

#	Student Name	Book Name	ISBN	Issued Date	Return Date	Action
1	Anuj kumar	PHP And MySQL programming	222333	2023-09-18 00:11:29	Not Return Yet	[Edit]
2	Sarita Pandey	physics	1111	2017-07-15 23:32:55	Not Return Yet	[Edit]
3	test	PHP And MySQL programming	222333	2017-07-15 16:29:26	Not Return Yet	[Edit]
4	Anuj kumar	physics	1111	2017-07-15 11:53:23	2017-07-15 16:52:29	[Edit]
5	Anuj kumar	physics	1111	2017-07-15 11:49:40	Not Return Yet	[Edit]
6	Anuj kumar	PHP And MySQL programming	222333	2017-07-15 11:42:27	2017-07-15 16:45:23	[Edit]
7	Anuj kumar	PHP And MySQL programming	222333	2017-07-15 11:39:47	2017-07-15 16:45:20	[Edit]

Showing 1 to 7 of 7 entries

Admin issued books page

admin dashboard

SJCE LIBRARY PORTAL

DASHBOARD | CATEGORIES | AUTHORS | BOOKS | ISSUE BOOKS | REG STUDENTS | CHANGE PASSWORD | LOG ME OUT

ISSUE A NEW BOOK

Issue a New Book

Srudent Id*
SID002

Anuj kumar

ISBN Number or Book Title*
222333

PHP And MySQL programming

Issue Book

page to issue book

SJCE LIBRARY PORTAL

DASHBOARD | CATEGORIES | AUTHORS | BOOKS | ISSUE BOOKS | REG STUDENTS | CHANGE PASSWORD | LOG ME OUT

STUDENT DASHBOARD

1
Book Issued

1
Books Not Returned Yet

SJCE LIBRARY PORTAL

DASHBOARD | CATEGORIES | AUTHORS | BOOKS | ISSUE BOOKS | REG STUDENTS | CHANGE PASSWORD | LOG ME OUT

MANAGE REG STUDENTS

Reg Students

#	Student ID	Student Name	Email Id	Mobile Number	Reg Date	Status	Action
1	SID002	Anuj kumar	anuj.lpu1@gmail.com	9865472555	2017-07-11 21:07:05	Active	Inactive
2	SID005	sdfsd	csfd@dfsfks.com	8569710025	2017-07-11 21:11:27	Blocked	Active
3	SID009	test	test@gmail.com	2359874527	2017-07-11 21:28:28	Active	Inactive
4	SID010	Amit	amit@gmail.com	8585856224	2017-07-15 19:10:30	Active	Inactive
5	SID011	Sarita Pandey	sarita@gmail.com	4672423754	2017-07-15 23:30:59	Active	Inactive

Showing 1 to 5 of 5 entries

Testing Chatbot if it understands the intents of user perfectly

SJCE chatbot

bot
Hi, welcome to SJCE chatbot!, i am here to help you. 😊
you can ask me anything related to SJCE college....

You 00:33
is there hostel facility?

Bot 00:33
There are 2 hostels in the campus premises and two outside the campus for both girls and boys

You 00:35
how are placements in sjce?

Bot 00:35
It starts from july for students studying in 6th semester.
More than 200 companies take part in the process
You will be notified about it from the placement cell and keep an eye at [Placement](#)

Enter your message... Send

SJCE chatbot

bot
Hi, welcome to SJCE chatbot!, i am here to help you. 😊
you can ask me anything related to SJCE college....

You 00:35
what are the branches present in sjce

Bot 00:35
There are CSE,IS,CSBS,EC,EEE,EI which are circuit branches and MECH,CIVIL,CTM,BT,PS,IP are non-circuit branches
for more info on every dept and faculty [click here](#)

You 00:35
what is the vision of sjce

Bot 00:35
vision of SJCE is: 'Be a international leader in engineering education, research and application of knowledge to benefit society globally'

Enter your message... Send

SJCE chatbot

bot

Hi, welcome to SJCE chatbot!.i am here to help you. 😊 you can ask me anything related to SJCE college....

You 00:36 when is open day

Bot 00:36
Open day is a two day event planned on 3rd and 4th of June 2023
It is a platform where each every department showcase their best projects to the outside world along with exciting events and performances for the audience.

You 00:36 where can i see my results

Bot 00:36
results for sem end exams can be found at [results](#)

Enter your message... Send

SJCE chatbot

bot

Hi, welcome to SJCE chatbot!.i am here to help you. 😊 you can ask me anything related to SJCE college....

You 00:46 hi

Bot 00:46 Hello!

You 00:46 what can you do

Bot 00:46 I can help you to know about SJCE College

You 00:46 tell me about sjce

Bot 00:46 Sri Jayachamarajendra College of Engineering(SJCE) is a Engineering college in Mysore

Enter your message... Send

SJCE chatbot

bot

Hi, welcome to SJCE chatbot!.i am here to help you. 😊 you can ask me anything related to SJCE college....

You 00:49 whatgpt do this

Bot 00:49 I didn't get that, try again

Enter your message... Send

Chapter 9 : Conclusion and Future Work

The Student Link project aims to create a Smart Campus Platform that leverages various modules to enhance the student experience and improve operational efficiency. The implementation of a blockchain-based student profile system ensures the security and integrity of student information. The online canteen system provides a convenient and efficient way for students to order meals, while the library portal enables digital access to library resources. The integration of a chatbot enhances student engagement and provides instant information about the university and campus activities.

Through the use of tools and technologies such as Ganache, Truffle, Metamask, Solidity programming, React.js, PHP, Node.js, MySQL, Python, Flask, and TensorFlow, the project utilizes a combination of blockchain, web development, and AI technologies to achieve its objectives.

The Future Work includes:

1. Expansion of Features: The Smart Campus Platform can be further enhanced by adding more features and functionalities. For example, incorporating a timetable management system, event management system, or collaboration tools to foster student interaction and engagement.
2. Integration with Student Information Systems: Integrating the Smart Campus Platform with existing student information systems can provide a seamless flow of data, enabling real-time updates and synchronization of student profiles, course enrollment, and academic records.
3. Integration with Campus Infrastructure: The platform can be integrated with campus infrastructure such as access control systems, smart devices, and IoT sensors to enable a truly interconnected campus environment. This integration can enhance security, automate processes, and provide personalized experiences for students.

4. Data Analytics and Insights: Utilizing data analytics techniques, the platform can generate valuable insights about student behavior, preferences, and academic performance. These insights can be used to improve decision-making, personalize services, and optimize campus resources.

5. Continuous Improvement and Maintenance: The Smart Campus Platform should undergo regular updates, bug fixes, and security enhancements to ensure its smooth operation and adaptability to evolving student needs and technological advancements.

Overall, the Student Link project sets the foundation for a technologically advanced and student-centric campus environment. By embracing innovative technologies and incorporating user-friendly features, the platform aims to transform the student experience and pave the way for a more connected and efficient campus ecosystem.

APPENDIX A - Project team details

Project Title	Title of the project		
USN	Team Members	Email ID	Mobile number
01JST19CS044	Devothama GN	devothamgn321@gmail.com	6364600337
01JST19CS034	Chandana Raju	chandanaraju1503@gmail.com	9606365785
01JST19CS128	Shaman B H	shamanharish123@gmail.com	9113890793
01JST19CS042	Darshan M	darshanmahesh1257@gmail.com	9380712398



PHOTO

DEVOTHAMA GN



PHOTO

CHANDANA RAJU

SHAMAN B H

DARSHAN M

APPENDIX B - COs, POs and PSOs

Mapping for the project work (CS83P)

Course Outcomes:

CO1: Formulate the problem definition, conduct literature review and apply requirements analysis.

CO2: Develop and implement algorithms for solving the problem formulated.

CO3: Comprehend, present and defend the results of exhaustive testing and explain the major findings.

Program Outcomes:

PO1: Apply knowledge of computing, mathematics, science, and foundational engineering concepts to solve the computer engineering problems.

PO2: Identify, formulate and analyze complex engineering problems.

PO3: Plan, implement and evaluate a computer-based system to meet desired societal needs such as economic, environmental, political, healthcare and safety within realistic constraints.

PO4: Incorporate research methods to design and conduct experiments to investigate real-time problems, to analyze, interpret and provide feasible conclusion.

PO5: Propose innovative ideas and solutions using modern tools.

PO6: Apply computing knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to professional engineering practice.

PO7: Analyze the local and global impact of computing on individuals and organizations for sustainable development.

PO8: Adopt ethical principles and uphold the responsibilities and norms of computer engineering practice.

PO9: Work effectively as an individual and as a member or leader in diverse teams and in multidisciplinary domains.

PO10: Effectively communicate and comprehend.

PO11: Demonstrate and apply engineering knowledge and management principles to manage projects in multidisciplinary environments.

PO12: Recognize contemporary issues and adapt to technological changes for lifelong learning.

Program Specific Outcomes:

PSO1: Problem Solving Skills: Ability to apply standard practices and mathematical methodologies to solve computational tasks, model real world problems in the areas of database systems, system software, web technologies and Networking solutions with an appropriate knowledge of Data structures and Algorithms.

PSO2: Knowledge of Computer Systems: An understanding of the structure and working of the computer systems with performance study of various computing architectures.

PSO3: Successful Career and Entrepreneurship: The ability to get acquaintance with the state-of-the-art software technologies leading to entrepreneurship and higher studies.

PSO4: Computing and Research Ability: Ability to use knowledge in various domains to identify research gaps and to provide solution to new ideas leading to innovations.

Justification for CO-PO and PSO mapping

(Example of justification-In our project we have come across many of these Program Outcomes and Program Specific Outcomes at various stages. Basic knowledge of computing, mathematics,

science, and engineering concepts helped in getting the foundation right for the project. The motivation of this project involved in building a computer-based system to meet desired societal needs such as economic, environmental, pharmaceutical, healthcare and safety within realistic constraints which led to this project. The automated generation of QR code coupled with a scanned app in android allowed us to design and conduct real-time feasible solution.

We have used the computing knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to professional engineering practice. The project helps in building a sustainable product, i.e. on a smart phone which is used by every individual. The work was equally distributed and shared between the whole team from the beginning stages till the end of project. This created a whole learning experience, in team work and managing the team.)

		Table of Mapping of CO, PO and PSO:																		
SUBJECT	CODE	C O	P O 1	P O 2	P O 3	P O 4	P O 5	P O 6	P O 7	P O 8	P O 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3	PS O4		
Project Work	CS8 3P	C O1																		
		C O2																		
		C O3																		

Note:

Scale

0. – Not Applicable

1 – Low relevance Scale

2 – Medium relevance Scale

3 – High relevance