### **1.0 INTRODUCTION**

E-campus 2.0 revolutionizes information systems in educational institutions by prioritizing accuracy and efficiency. Focused on managing student attendance, assignment and exam scores, the system utilizes a robust database. With a swift and user-friendly interface, it ensures rapid data retrieval and efficient maintenance. Similar for students, staff, and administrators, role-based login privileges enhance security. Students access their individual marks, while staff can view and modify all student data. Admin can control permissions, development of secure data management.

The system calculates internal grades based on assignment, exam marks, and attendance. Offering a discussion forum, it facilitates communication among students and faculty with customizable comment visibility. The report generation feature ensures professional and accessible data storage, promoting easy information sharing. E-campus 2.0 marks a significant advancement in student data management, enhancing usability and efficiency across educational institutions.

E-campus 2.0 has the potential to completely transform how colleges and universities handle student data. Focused on enhancing accuracy and efficiency, this system uses a strong database design to highlight the careful management of exam score, internal assessments, and attendance records. Its user-friendly design ensures swift data retrieval and storage, while a sophisticated graphical user interface further enhances accessibility.

This whole system provides different privileges based on positions to meet the different requirements of administrators, staff, and students. Faculties have the ability to review and modify each student's mark, students can quickly access and view their marks. Admin hold pivotal roles, possess the authority to grant or revoke permissions for other user roles, ensuring a hierarchical and secure access framework.

E-campus 2.0 data report generate can store data and generate reports as output from the data in the database quickly which can be accessed from anywhere through the internet in other hand it can also customize the report, this makes it easier to share information students and staffs. Overall, the report generation in E-campus 2.0 can improve the ease of use and efficiency of the data management of the student data.

E-campus 2.0 is a comprehensive information system revolutionizing student data management in educational institutions. The system efficiently calculates students' internal assessments based on assignment and exam marks, merge with attendance records. Faculty staff, it allows them to post, upload, and alter student data, presenting the information in an organized report format. A unique feature is the inclusion of a discussion forum encourage communication between staff and students through comments.

**2.0 SYSTEM STUDY**

**2.1 Existing System:**

The paper-based system for managing student marks, once a cornerstone of education, now struggles to meet the demands of the 21st century. With advancements in educational technology such as Learning Management Systems (LMS) and student information systems, offer opportunities to streamline grading, enhance data security, improve accessibility, and provide real-time analytics. By embracing these modern solutions, educational institutions can better meet the evolving needs of students and users, creating a more efficient and effective learning environment.

One significant drawback of paper-based grading is its reliance on manual data entry, consuming valuable instructional time and increasing the risk of errors. These errors can have cascading effects, leading to inaccurate grade reports, unfair student evaluations, and unnecessary stress for both students and teachers. Moreover, inaccuracies in grades can profoundly impact student motivation, self-esteem, and academic trajectory, potentially affecting eligibility for academic programs and scholarships, while also damaging teacher credibility and creating friction with parents.

The physical nature of paper records also poses challenges for information sharing and collaboration among educators, leading to delays in feedback and hindering a holistic understanding of student progress. Furthermore, paper records are susceptible to loss, damage, or misplacement, hindering historical analysis and long-term tracking of student progress. Additionally, the lack of robust data analysis capabilities makes it difficult for educators to identify trends or track the effectiveness of instructional strategies.

E-campus 2.0 offer data analysis tools that provide real-time insights into student performance, enabling educators and administrators to generate customizable reports with ease. However, the reliance on paper records also raises concerns about data security and privacy, as paper documents lack robust access controls and audit trails. Unauthorized access to student information can have serious legal and ethical repercussions for educational institutions.

**2.2 Proposed system:**

The E-campus 2.0 can generate a more accurate and efficient information system. This system mainly concentrates on managing student attendance system, keeps track of the internal and other assignment and exam scores through a database and allows staffs to manage student-related data needs in college and universities.

E-campus 2.0 is user friendly because the retrieval and storing of data is fast and data is maintained efficiently. Moreover, the graphical user interface is provided in the proposed system. This system allows the students, staffs and admins to login and can have their privileges according to their roles. The student can view only their marks, but this system allows the staff to view all of their students marks and alter. Then the admins can give and revoke the usage permissions form other roles such as the staffs and students.

This system can calculate the students internal according to the data like assignment & exam marks, attendance etc. The staffs are able to post and edit or alter their student’s data. It also displays the data in a report format. This system provides a discussion forum, it enables to the staffs and students to communicate in a form of a comment with other students and faculties. The students and staffs can set visibilities of their comment on the forum. The flexibilities of the comment settings are depending on their login access mode. The admins are also allowed to remove others comments.

An innovative feature of E-campus 2.0 is its ability to calculate student’s internal assessments based on diverse data inputs such as assignment and exam marks, as well as attendance records. Faculty members have the flexibility to post, edit, and alter student data, presented in a user-friendly report format. The system also incorporates a discussion forum, fostering communication among students and faculties through comments. Users can control the visibility of their comments based on login access modes, and administrators retain the authority to remove inappropriate content.

3.0 SYSTEM SPECIFICATIONS

**3.1 HARDWARE REQUIREMENT**

Processor : Intel core i3, 8th Generation

Memory : DDR3 4GB RAM

Hard disk : 500GB HDD

**3.2 SOFTWARE REQUIREMENT**

Operating System: Windows 7

Front End: PHP Laravel 8.1, HTML5 and CSS3

Back End : MySQL Server 8.0

**4.0 SYSTEM DESIGN**

**4.1 INPUT DESIGN**

Input Design plays a vital role in the life cycle of software development. It requires very careful attention of developer. The input design is to feed data to the application as accurate as possible. So inputs are supported to be designed effectively so that the errors occurring while feeding are minimized. According to Software engineering Concepts, the input forms or screens are designed to provide to have a validation control over the input limit, range and other related validations. This system has screens in almost all the modules. Error message have been developed to alert the user whenever he commits some mistakes and guide him in the right way.

The objectives of input design are as follows

The important features are

* The input screen is not crowed, as the user can understand the information from the screen.
* The input validation is being done at program level to check errors & help messages are to be provided.

The following Input are

* Login Process
* Theory
* Events
* Practical
* Internal Marks

**4.1.2 DATABASE DESIGN**

**i. STUDENT(TABLE)**

Table: Student

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Stud\_rno** | **Stud\_name** | **Stud\_dept** | **Stud\_batch** | **Stud\_pwd** | **Stud\_year** | **Stud\_counslor** |

**ii. STAFF(TABLE)**

Table: Staff

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Staff\_no** | **Staff\_name** | **department** | **batch** | **counselor** |

**iii. ASSIGNMENT(TABLE)**

Table: Assignment

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Assign\_id** | **Stud\_rno** | **C\_code** | **assignment** | **1** | **2** | **3** | **4** | **average** |

**iv. Practical (TABLE)**

Table: Practical

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **no** | **Stud\_rno** | **C\_code** | **mark** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **average** |

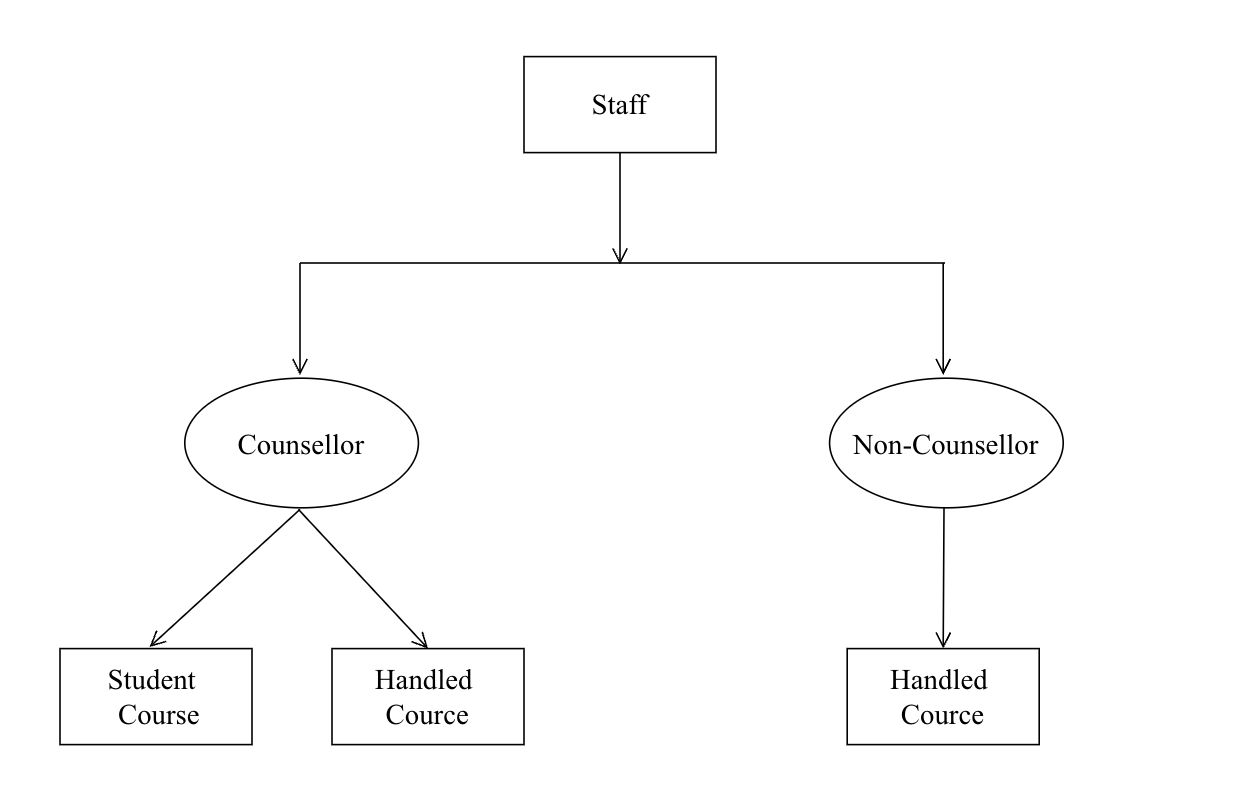
**v. Event(TABLE)**

Table: Event

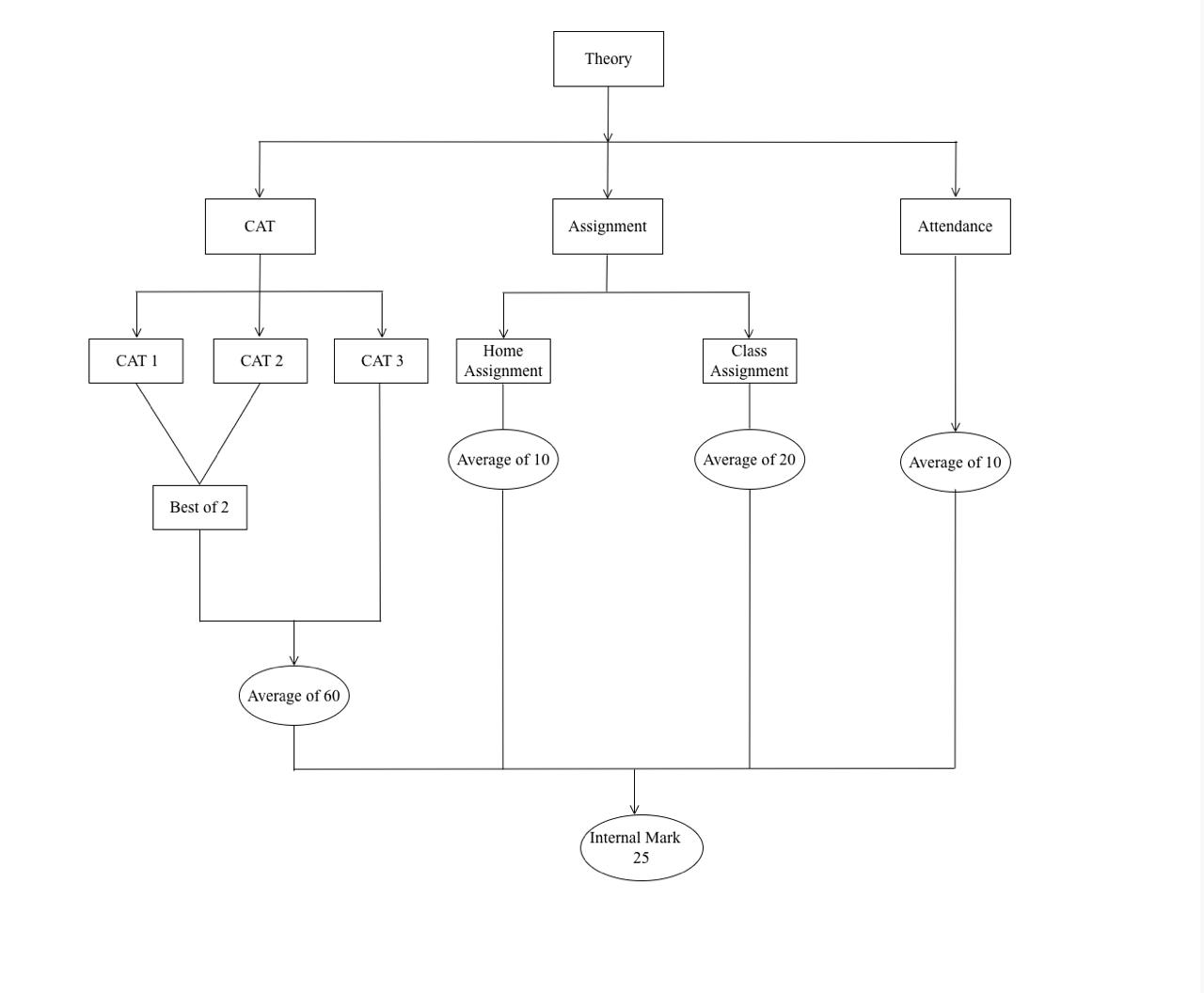
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Event\_id** | **Event\_name** | **Dept\_id** | **description** | **Event\_date** |

**4.1.3 FLOW DIAGRAM**

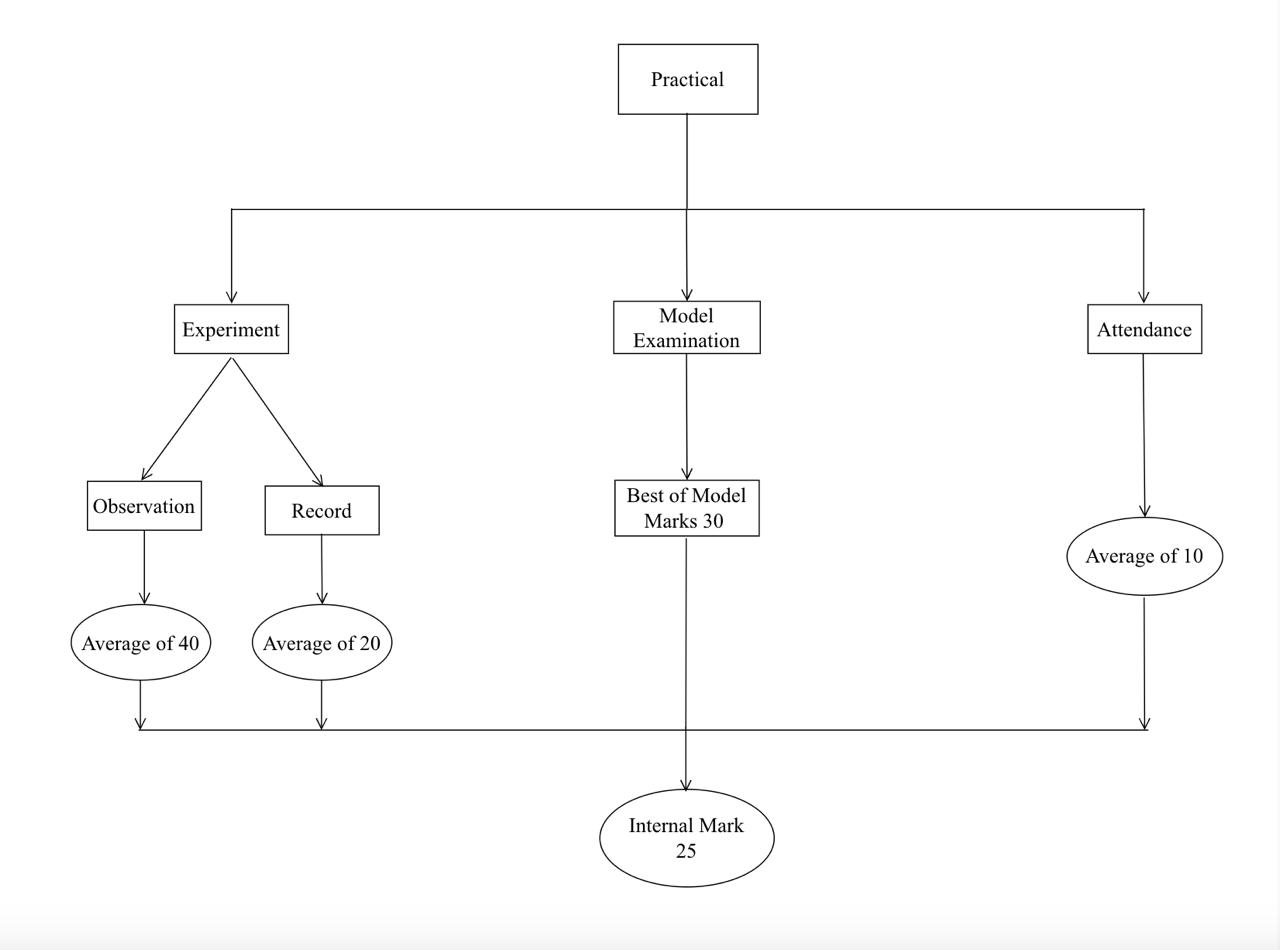
**LOGIN**



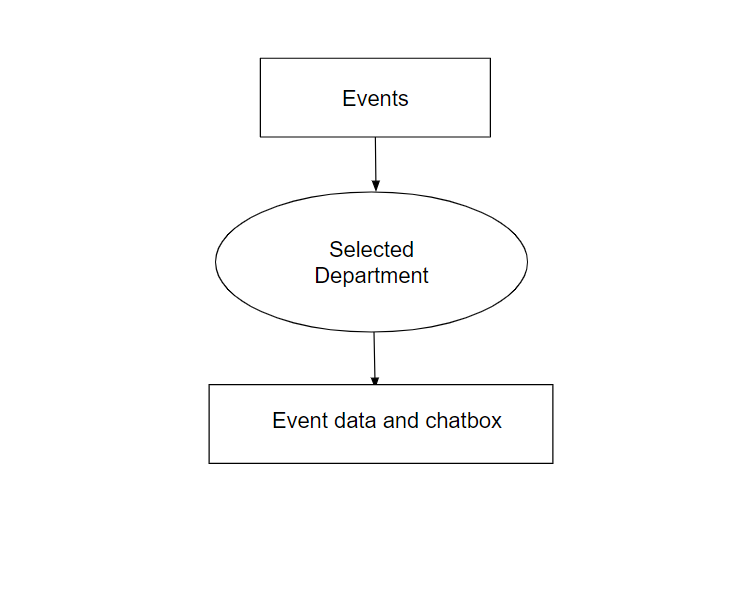
**THEORY**



**PRACTICAL**



**EVENTS**



**4.2 OUTPUT DESIGN**

The report generation module serves as a dynamic platform within the system. The user can take report in the modules of assignment marks, observation and record marks, attendance, internal marks and etc. The report automatically generates from the database. These reports are accessible from anywhere with an internet connection, providing flexibility and convenience. Users can customize reports to focus on specific areas and share them easily with colleagues or students. The users can evaluate student data and thereby provide improved support for their students' academic performance.

* Dynamic reporting: Users can access reports on various aspects like assignments, attendance, and internal marks.
* Automated generation: Reports are automatically pulled from the database, saving time and effort.
* Accessibility: Reports can be accessed from anywhere with an internet connection.
* Customization: Users can tailor reports to focus on specific areas of interest.
* Collaboration: Reports can be easily shared with colleagues or students.
* Improved student support: Educators can leverage data to provide targeted support for student performance.

The user can import the data while clicking the import button in the module, the import function allows users to upload large amounts of student data efficiently using a common file format (CSV) and the imported data’s are stored in the table. If the data is in the table, it does not import the duplicate entries by checking for existing data. This ensures data accuracy and avoids confusion caused by multiple records for the same student.

* Efficient data upload: Large amounts of student data can be uploaded using a common CSV format.
* Data accuracy: The system checks for existing data to prevent duplicate entries.

The user can export the data while clicking the export button in the module, the export function allows users to download data from reports into CSV files and the downloaded file store in the user system. This creates a portable format that can be easily analyzed using popular spreadsheet software. The export functionalities enable users to effectively handle student data, analyze it in more detail, and foster better communication regarding student performance.

* Portable data: Users can download reports as CSV files for further analysis in spreadsheet software.
* Detailed analysis: Exporting data allows for deeper analysis of student performance.
* Enhanced communication: Sharing exported data facilitates better communication about student progress.

**5.0 MODULES**

**5.1 LOGIN:**

The system's login module allows users to gain access with different features according to the roles they have been assigned, such as staff, student, or admin. This module acts as the first interface through which users enter their login credentials such as the username and password, starting a sequence of actions that eventually determine their access rights and functionalities.

In login.html interface, the system channels control and variable data to log.php a component designed to interface with the underlying PHP and MySQL database. Through this interaction, the system verifies the existence of the user and proceeds to ascertain their assigned role within the system hierarchy.

For faculty members, their roles are further describe into two distinct categories: counsellor and non-counsellors. Counsellor enjoy broader access privileges within the system. They possess the capability to view and interact with all courses available to their students, as well as those courses directly under their supervision. This expansive access allows councilors to oversee and manage a diverse array of academic offerings, enabling a complete approach for academic administration and student support.

Conversely, non-counsellor faculty members operate within a more limited scope. Their access is restricted to courses that fall under their direct purview. While their access may be more constrained, non-counsellor faculty members still wield significant authority within their designated courses. They are empowered to perform various tasks such as uploading data, viewing student marks, and generating reports tailored to their specific administrative needs.

Students, as a distinct user group within the system, possess access privileges similar to their unique role. Their interaction within the system primarily revolves around accessing and reviewing their own academic performance. Through the system interface, students can view their respective marks and track their progress across various courses and assignments.

At the apex of the system hierarchy lies the admin role, endowed with the highest level of access and control. Admins occupy a central position within the system, x. In addition to possessing capabilities akin to faculty members, admins have the unique ability to virtually access all courses within the system. This broad access enables admins to maintain a comprehensive view of system activity and intervene as necessary to ensure optimal functionality and user experience.

One of the key responsibilities entrusted to admins is the creation and management of user accounts within the system. Admins possess the authority to create new user accounts, whether they be staff or students, thereby shaping the composition and dynamics of the user ecosystem. This administrative function plays a crucial role in ensuring the integrity and security of the system while facilitating seamless user onboarding and engagement.

In summary, the login module serves as a critical gateway within the system, dictating the scope of user access and interaction based on their designated roles. If it is staff, students, or admins, each user group is afforded distinct privileges and capabilities tailored to their respective responsibilities and requirements within the educational framework.

**5.2 THEORY:**

**5.2.1 ASSIGNMENT**

The E-Campus 2.0 system provides a centralized platform for managing student data and generating reports of Class Assignment and Home Assignment based on the user privilege. Users can access a assignment module to view and edit the assignment marks for all courses offered to students. This functionality allows for efficient management of student progress across various subjects. Users can filter marks based on specific assignments and course codes.

The system empowers users to generate detailed reports on student performance, course engagement, and other relevant metrics. These reports are automatically generated from the database, saving time and effort. Users can tailor reports to their specific needs by filtering them based on criteria like assignment type (e.g., Class Assignment, Home Assignment) and course code.

The system offers import and export functionalities in a common CSV (comma-separated values) format. This enables users to upload large amounts of student data efficiently. The system intelligently checks for existing data to prevent duplicate entries, ensuring data accuracy and avoiding confusion caused by multiple records for the same student.

Users can also export data displayed in generated reports into downloadable CSV files. This creates a portable format that can be easily analyzed using popular spreadsheet software, allowing for further in-depth analysis of student performance. Additionally, this export functionality facilitates the sharing of data with external parties.

In essence, the E-Campus 2.0 system caters to the specific needs of users within the educational environment by providing a tool for managing student data and generating reports. The ability to view and edit marks, filter data, generate customized reports, and import and export information in a user-friendly format empowers users to efficiently manage student progress, identify areas for improvement, and gain valuable insights into student performance. This centralized platform promotes better data management, fosters collaboration, and ultimately contributes to a more effective learning environment.

**5.2.2 CAT**

The E-Campus 2.0 system provides a centralized platform for managing student data and generating reports of CAT marks based on the user privilege. Users can access a dedicated module for viewing and editing student marks across various courses and categories of CAT. This allows for efficient monitoring of student progress in different subject areas. Filtering options by year and category further enhance this functionality.

The system empowers users with comprehensive reporting capabilities. They can generate reports encompassing student performance, course engagement, and other relevant metrics. These reports can be similar to specific needs by selecting categories and years. It allows users to gain valuable insights into student learning.

Data import and export functionalities further enrich the user experience. Users can upload large datasets of student information in a common CSV format (comma-separated values). This eliminates manual entry and saves significant time. The system safeguards data integrity by preventing duplicate entries during import, ensuring only new information is added.

The users can export data displayed in generated reports into downloadable CSV files. This creates a portable format that can be easily analyzed using popular spreadsheet software. This functionality proves valuable for further analysis and fosters better communication regarding student performance.

The system automatically generates reports based on data stored within the database. These reports provide an information at a glance, including class averages, pass percentage, standard deviation, potential instances of academic (malpractice), number of absent, and overall exam failures. This comprehensive data analysis saves users significant time and effort.

**5.3 PRACTICAL:**

**5.3.1 EXPERIMENTS:**

The E-Campus 2.0 system provides a platform to the specific needs of users. The system offers different privileges based on user roles. Users can access a dedicated module to view and edit student marks across various courses. This functionality allows for efficient management of student progress in different subjects. Additionally, users can filter marks based on specific criteria, such as course code or experiment type (e.g., observation, record), enabling them into granular details of student performance.

The system offers a powerful reporting engine that leverages data from the database to generate comprehensive reports. Users can tailor these reports to their specific needs by filtering them based on relevant metrics such as experiment type, course code. This empowers them to gain valuable insights into student performance, course engagement, and other educational trends.

The system facilitates the efficient import and export of student data using a common CSV (comma-separated values) format. This allows users to upload large amounts of student information in bulk, significantly reducing manual data entry and streamlining record management. The system safeguards data integrity by intelligently checking for existing entries within the chosen table. This ensures that only new information is imported, preventing duplicate records.

Reports generated within the system are accessible from anywhere with an internet connection. This provides users with flexibility and convenience for data analysis. Additionally, the ability to export data in a portable CSV format allows for further analysis using popular spreadsheet software, fostering deeper exploration of student performance trends. Furthermore, the export functionality empowers users to share data with colleagues or external parties, facilitating better communication and collaboration regarding student progress.

The E-Campus 2.0 system implements a robust role-based access control mechanism. This ensures that users only have access to the functionalities and data relevant to their specific roles within the educational environment. This approach safeguards data security and streamlines workflows for all users, promoting a more efficient and secure learning management experience.

The E-Campus 2.0 system provides to empower users with efficient data management, insightful reporting, and seamless collaboration. This system offers a significant advantage for educational institutions by streamlining administrative processes, fostering better communication, and ultimately, promoting a more effective learning environment for all users.

**5.3.2 MODEL**

The E-Campus 2.0 system provides a different user experience based on assigned roles. Users with access to the model module can view and edit marks for all courses offered to students. This functionality allows for efficient management of student progress across various subjects. Additionally, these users possess full authority over reports generated from the database. This empowers them to create detailed reports on student performance, course engagement, and other relevant metrics. They can tailor reports to their specific needs by filtering based on course code, enabling a deep dive into granular details of student performance in different areas.

The system facilitates the efficient import and export functionalities in CSV (comma-separated values) format. Users can upload large amounts of student data efficiently, reducing manual entry and streamlining record management. The system intelligently checks for existing data to prevent duplicate entries, ensuring data accuracy. Users can also export data displayed in generated reports into downloadable CSV files. This provides a portable and user-friendly format for further analysis in spreadsheet software. This functionality facilitates sharing information with colleagues or external stakeholders, fostering collaboration and communication.

For users managing specific courses within the system, they can view and edit course content, ensuring students have access to the most current materials. Similar to users with broader access, they can generate reports specific to their courses. By filtering reports by course code, they gain valuable insights into student performance within their designated areas of responsibility. The system also allows them to import and export data related to their courses using CSV files. This functionality enables efficient uploads of student information specific to their assigned subjects, while maintaining data integrity by preventing duplicates. They can export data from generated reports into downloadable CSV files for further analysis or sharing relevant data with colleagues or external parties.

**IMPORT:**

The import module system offers a powerful set of tools for counselors and faculty to manage student data efficiently and effectively. These tools revolve around the concept of uploading information in a common format called CSV (comma-separated values). This format essentially organizes data into a table where each row represents a student record, and each column contains a specific piece of information. Counselors can import function to upload student information for all available categories in bulk. This eliminates the manual data entry for large groups of students, saving significant time and effort.

The system acts intelligently by checking for existing data within the chosen table. This ensures that only new information is imported, preventing the creation of duplicate records. This not only maintains data accuracy but also streamlines record management, making it easier to find and update student information. Similar to counselors, faculty can import data related to the courses they manage using CSV files. This proves particularly beneficial for uploading student information specific to their assigned subjects. The faculty member can simply upload a CSV file containing the new student data, eliminating the need to manually enter information for each student within the system.

This significantly reduces administrative burden and allows faculty to dedicate more time to teaching and interacting with students. Just like counselors, faculty benefit from the system's ability to prevent duplicate entries during import. This ensures data integrity and eliminates the confusion that can arise from having multiple records for the same student.

**EXPORT:**

The export module system offers a powerful set of tools for counselors and faculty to manage student data efficiently and effectively. These tools revolve around the concept of downloading information in a common format called CSV (comma-separated values). This format essentially organizes data into a table where each row represents a student record, and each column contains a specific piece of information. The counselors can export the data displayed in generated reports into downloadable CSV files.

This provides them with a portable and user-friendly format for further analysis using common spreadsheet software like Excel. This allows counselors to manipulate and analyze the data in various ways, gaining deeper insights into student performance or identifying areas that require additional support. Additionally, the ability to export data in a CSV format facilitates sharing information with colleagues or external stakeholders, such as parents or educational researchers.

This fosters collaboration and communication, enabling a more comprehensive understanding of student progress. Additionally, faculty can export data from generated reports into downloadable CSV files. This provides them with a portable format for further analysis. The ability to export data in a CSV format also empowers faculty to share relevant information with colleagues, such as other instructors teaching the same course, or with external parties. This can be particularly useful for sharing data with external stakeholders, such as parents or educational researchers, can also be facilitated through CSV exports.

**EVENT:**

The events module serves as a dynamic platform within the system, fostering collaboration and engagement among users through the organization and participation in various events. Staff members and admins are empowered to create new events, providing essential details such as event name, description, and date via the event. php interface.

Upon creation, these events are automatically integrated into the system, appearing alongside other listed events on the event.php page. This visibility ensures that all users, including admins, staff, and students, have access to the list of upcoming events within the system.

Clicking on any listed event redirects users to the event-cb.php page, where a tailored interface awaits. Divided into two distinct columns, event-cb.php presents users with essential event details, such as description and date, on the left side. This information serves as a comprehensive overview, offering insights into the nature and scope of the selected event.

Complementing the event details is a user-friendly chatbox located on the right side of the page. Designed related to popular messaging platforms like WhatsApp, this chatbox serves as a centralized hub for event-specific communication. Users across all roles: admins, staff, and students, can actively engage in discussions, share updates, and coordinate activities related to the selected event.

Each event within the system exhibits its own dedicated chatbox, ensuring that conversations remain focused and relevant to the respective event. This segregation promotes efficient communication and collaboration, enabling users to exchange ideas, coordinate plans, and foster a sense of community around specific events.

In essence, the events module acts as a trigger interaction and collaboration within the system. By providing users with the tools to create and engage with events, the module fosters a vibrant ecosystem conducive to shared learning, networking, and collective achievement. The events module embraces the essence of collaboration through smooth integration and simple user interfaces, enabling users to connect, communicate, and work collaboratively towards common objectives.

**6.0 SYSTEM IMPLEMENTATION**

System implementation is like making a new system design actually work. This involves training users, getting the site ready, and converting files to install the new system. It's important that this process doesn't disrupt how the organization works. The main aim is to smoothly start using the tested system while keeping costs, risks, and personal irritation to a minimum. Before using a computer system, it's checked to see if it works well. We test it based on how well it performs and how effective it is. These tests are done in a well-planned and organized way. Implementation is the stage where the theoretical design becomes a real working system. Let's break down the steps in the implementation plan. First, we make sure users are trained to use the new system. Then, we prepare the site to make sure it's ready for the system. Lastly, we convert files carefully to move data to the new system. This detailed implementation plan aims to transition from the design to the working system smoothly while making sure the organization runs without any big disruptions.

* Test the system with sample data.
* Detection and correction of errors.
* Make necessary changes in the system.
* Check with the existing system.
* Installation of hardware and software.
* Training and involvement of user personal

**7.0 SYSTEM TESTING**

**7.1 TESTING**

Testing is a series of different tests that whose primary purpose is to fully exercise the computer-based system. Although each test has a different purpose, all work should verify that all system element has been properly integrated and performed allocated function. Testing is the process of checking whether the developed system works according to the actual requirement and objectives of the system. The philosophy behind testing is to find the errors. A good test is one that has a high probability of finding an undiscovered error. A successful test is one that uncovers the undiscovered error. Test cases are devised with this purpose in mind. A test case is a set of data that the system will process as an input. However, the data are created with the intent of determining whether the system will process them correctly without any errors to produce the required output.

**Types of Testing**

* Unit testing
* Integration testing
* Validation testing
* Output testing
* White box testing
* Black box testing

**Unit Testing**

Unit testing involves a series of tests done before combining a unit into bigger systems. In unit testing, each part is tested on its own. In our project we tested about totally of 5 individual modules such as login, Theory, Practical, Event and Internal Mark. We check every part of a module separately to make sure the data flow is right and that it exits normally. This helps us find out if each piece is working correctly before putting everything together.

**Integration Testing**

Integration testing is similar to assembling puzzle pieces to make sure they work well as a team. It involves gradually combining different modules within the system and testing to confirm that everything functions properly when they are integrated. For instance, let's consider the student login module. Here, when a student enters their username and password, the system sends this information to the database to check its validity. If the condition is met, the username and password is accepted; otherwise, a message pops up, indicating "Invalid password." This step-by-step approach of adding modules and assessing their interaction ensures that the entire system operates seamlessly without any glitches. Integration testing acts like a quality check to ensure that all the pieces of the system fit together smoothly. It's a meticulous process where each component is combined gradually to guarantee that the overall functionality aligns seamlessly. The objective is to identify and address any potential issues that may arise when different parts of the system are integrated. This method is essential for creating a robust and cohesive system where each module interacts correctly. By testing the integration of these pieces, the development team can be confident that the final product will work harmoniously. Integration testing is an integral part of the software development process, ensuring that the various elements of the system come together seamlessly, just like assembling puzzle pieces to create a complete and functional picture.

**Validation Testing**

Validation testing is the step where we make sure a new system design works properly. It's a phase where success means the software functions in a way that meets the customer's expectations. This is done through tests that show the software follows the given requirements. In places where users need to input information, we do validation testing. For instance, in a login authentication module, it's necessary for the user to enter both the username and password. If any of these fields are left empty, asking the user to fill in the missing information. These validation checks make sure the system works correctly and that users provide the necessary information.

**Output Testing**

Output testing is the next step after validation testing, next step is output testing for the proposed system. This involves checking how the system displays or creates output, based on the format desired by the user. The output format is considered in two ways: on-screen and printed.

For on-screen output, the format is usually correct as it's designed during the system phase to match what the user needs. The information is presented on the screen in a way that is clear and follows the user's requirements. Similarly, for hard copy output, the system generates output according to the specifications requested by the user. This ensures that when information is printed, it follows the desired format and layout as per the user's needs. Output testing is essential to make sure the system provides information in the right format, both on the screen and in print. This step ensures that users receive outputs that are easy to understand, follow their requirements, and meet their expectations. It's a crucial part of making sure the system delivers accurate and user-friendly results.

**White box testing**

White box testing, also known as Clear Box Testing or Structural Testing, is a method of testing software where the person doing the testing knows about the internal structure or design of what they are testing. It's also referred to as Open Box Testing, Glass Box Testing, Transparent Box Testing, and Code-Based Testing. The tester actively selects inputs to follow different paths through the code and then determines what the correct outputs should be. This testing method gets its name because, for the tester, the software is like a white or transparent box. This means that they can see inside and understand how it works internally. During white box testing, the focus is on the internal mechanisms of the software, making sure that each part is functioning correctly. The tester essentially peeks into the "box" to verify that everything is operating as it should. In summary, white box testing is about understanding and testing the internal workings of the software to ensure it functions correctly, and the term "white box" comes from the idea of looking into a transparent box to see what's inside.

* White-box testing: Testing based on an analysis of the internal structure of the component or system.
* White-box test design technique: Procedure to derive and/or select test cases based on an analysis of the internal structure of a component or system.

**Black box testing**

Black box testing, also called Behavioral Testing, is a way of testing software where the person testing doesn't know about the internal structure or design of what they are testing. It's like working with a sealed box, and the tester focuses on how the software behaves without needing to understand how it works internally. These tests can be about how well the software functions (functional testing) or how it performs in terms of speed or other aspects (non-functional testing), although it's mostly about how the functions work. The main goal of black box testing is to find errors or issues in different categories of the software without knowing its internal details. The tester is like a user, trying to see if the software does what it's supposed to do without needing to know the technical details. This method helps ensure that the software meets its intended requirements and works as expected from the user's perspective.

* Incorrect or missing functions
* Interface errors
* Errors in data structures or external database access
* Behavior or performance errors
* Initialization and termination errors
* Black box testing: Testing, either functional or non-functional, without reference to the internal structure of the component or system.
* Black box test design technique: Procedure to derive and/or select test cases based on an analysis of the specification, either functional or non-functional, of a component or system without reference to its internal structure.

**7.2** **SYSTEM TESTING**

System testing is the phase where we prepare everything to test each module independently. This means getting the test data ready and checking if the fields meet the validation requirements. In this context, testing is about ensuring that the system operates correctly and smoothly before it goes live. During testing, we run the program to identify and fix any errors, ensuring everything is in good shape for actual operation. The main aim is to guarantee the system performs accurately and efficiently. This stage is crucial for catching and addressing potential issues before the system is actively used. It's like a final check to make sure everything is working as it should. The process involves running various scenarios to simulate how users would interact with the system, ensuring it can handle different situations. The goal is a seamless and error-free performance when the system is officially implemented. System testing helps in building confidence that the software will meet expectations and function smoothly in real-world situations.

.

**8.0 SOFTWARE FEATURES**

**About PHP**

PHP is a powerful server-side scripting language for creating dynamic and interactive websites. PHP widely used; free and efficient alternative to competitors such as Microsoft’s ASP.PHP is perfectly suited for Web development and can be embedded directly into the HTML code. The PHP syntax is similar to pearl and C.

PHP is open source that it is readily available and absolutely free. Stability, flexibility and speed are chief qualities that attract to choose PHP.PHP have multiple extensions and is extremely scalable.

**Server-side scripting**

This server-side scripting is the most traditional and main target field for PHP. Programmer needs three things to make this work. Programmer need to run the web server, with a connected PHP installation. Programmer can access the PHP program output with a web browser, viewing the PHO page through the server. All these can run on your home machine if programmers are just experimenting with PHP programming.

**Command line scripting**

Programmer can make a PHP script to run it without any server or browser. Programmers only need the PHP parser to use it this way. This type of usage is ideal for scripts regularly executed using croon (on\*nix or Linux) or Task Scheduler (on Windows). These scripts can also be used for simple text processing tasks.

**Features of PHP**

* PHP runs on different platforms (Windows, Linux, UNIX, etc.)
* PHP is compatible with almost all servers used today.
* PHP is free to download from the official PHP resource: [www.php.net](http://www.php.net).

**About PHP Laravel framework:**

PHP Laravel framework is a powerful and popular open-source framework that provides a robust set of tools and features for building web applications. Known for its elegant syntax, expressive syntax, and developer-friendly approach, Laravel simplifies the development process and accelerates the creation of high-quality applications.

**Features of PHP Laravel:**

* Laravel provides a simple and elegant routing system that allows developers to define application routes using expressive syntax.
* Laravel prioritizes security, helping developers build robust and secure web applications.

**About MYSQL**

MYSQL is an open-source relational database management systems (RDBMS), is developed, distributed and supported by MYSQL AB. MYSQL is a popular choice of database for use in web applications MYSQL can be scaled by deploying it on more powerful hardware, such as a multi-processor server with gigabytes of memory. MYSQL is easy to use, yet extremely powerful, secure, and scalable. And because of its small size and speed, it is the ideal database solution for Web sites.

**MYSQL is a database management system**

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amount of information in a corporation network. To add, access and process data stored in a computer database we need a database management system such as MYSQL server. Since computers are very good at handling large amount of data, database management system plays a central role in computing.

**MYSQL is a relational database management system**

A relational database stores separate data in separate tables rather than putting all the data in one big storeroom. This adds speed and flexibility. The SQL part of “MYSQL” stands for “Structured Query Language”. SQL is the most common standardize language used to access database and is defined by the ANSI/ISO SQL standard. The SQL standard has been evolving since 1986 and several versions exist.

**MYSQL software is open source**

Open-source means that it is possible for anyone to use modify the software. Anybody can download the MYSQL software uses the GPL (GNU General Public License), to define what we may and may not use do with the software.

**MYSQL Server works in Client/ Server or embedded systems**

The MYSQL database software is a client/server system that consists of a multi-threaded SQL server that supports different backend, several different client programs and libraries, administrative tools and a wide range of Application Programming Interface(APIs). A large amount of contributed MYSQL software is available:

Modern day websites seem to be relying more and more on compel the Structured Query Language is a very popular database language, and its standardization makes it easy to store, update and access data. One of the most powerful SQL servers out there is called MYSQL and surprisingly enough, it’s free.

Some of the features of MYSQL include: Handles large databases, in the area of 50,000,000+records. No memory leaks. Tested with a commercial memory leakage detector (purify). A privilege and password system which is very flexible and secure, and which allows host-based verification. Passwords are secure since all password traffic when connecting the server is encrypted.

**Features of MYSQL**

* Client/server Architecture: MYSQL is a client/server system. There is a database server (MYSQL) and arbitrarily many clients (application programs), which communicate with the server. The clients can run on the same computer as the server or on another computer.
* SQL Compatibility: As before said SQL is a standardized language for querying and updating data and for the administration of a database. Through the configuration setting sol-mode we can make the MYSQL server behave for the most part compatibly with various database systems.
* Stored procedures: Stored procedures (SPs for short) are generally used to simplify steps such as inserting or deleting a data record.
* Triggers: Triggers are SQL commands that are automatically executed by the server in certain database operations INSERT, UPDATE, and DELETE, MYSQL has supported triggers.
* Replication: Replication allows the contents of a database to be copied (replicated) onto a number of computers to increase protection against system and to improve the speed of database queries.
* Platform independence: MYSQL can be executed under a number of operating systems. The most important are Apple Macintosh OS X, Linux, Microsoft Windows, and the Unix.
* Speed: MYSQL is considered a very fast database program.

**9.0 CONCLUSION**

In conclusion, E-campus 2.0 signifies a noteworthy advancement in the domain of student information management and communication in academic establishments. E-campus 2.0 improves communication between faculties, students, and administrators by consolidating tasks like attendance monitoring, grading, and data reporting into one accessible interface. E-campus 2.0's advanced capabilities, which include customized reporting which can be derived from the data present on the database that are feeded into by the users to generate internal marks, role-based access restriction, and discussion forums, encourage openness, productivity, and successful communication to all of which contribute to the development of an environment that is favorable to learning and academic success. In addition to addressing the shortcomings of conventional paper-based systems, this comprehensive solution opens the door for upcoming improvements and developments in educational technology.

**9.0 FUTURE ENHANCEMENT**

There are a few major areas that stand out for possible development in the future

E-campus 2.0 additions. The utilizing machine learning algorithms offers a unique chance to apply complex data analysis to individualized student assistance and intervention plans. Machine learning can provide customized recommendations for academic development through the study of student data, increasing the quality of educational plans.

There is great potential to improve accessibility and the user engagement by creating a mobile application version of E-campus 2.0. By providing essential features via a mobile platform, users can obtain information and features while on the go, facilitating easy cooperation and communication between educators, administrators, and students.

Furthermore, the integration of email functionality into E-campus 2.0 promotes effective user-to-user contact by enabling prompt notifications, reminders, and announcements. This feature encourages effective means of communication throughout the educational community and improves user engagement.

Finally, the seamless integration with third-party systems, such as learning management systems or student information systems, can enhance interoperability and facilitate seamless data exchange across educational platforms. By embracing interoperability, E-campus 2.0 can streamline administrative processes and enhance collaboration among within the education ecosystem. using this application, and we aim to make it user-friendly. This ongoing development ensures that the software stays relevant and continues to meet the evolving needs of its users.

**11.0 BIBLIOGRAPHY**

**BOOK REFERENCES**

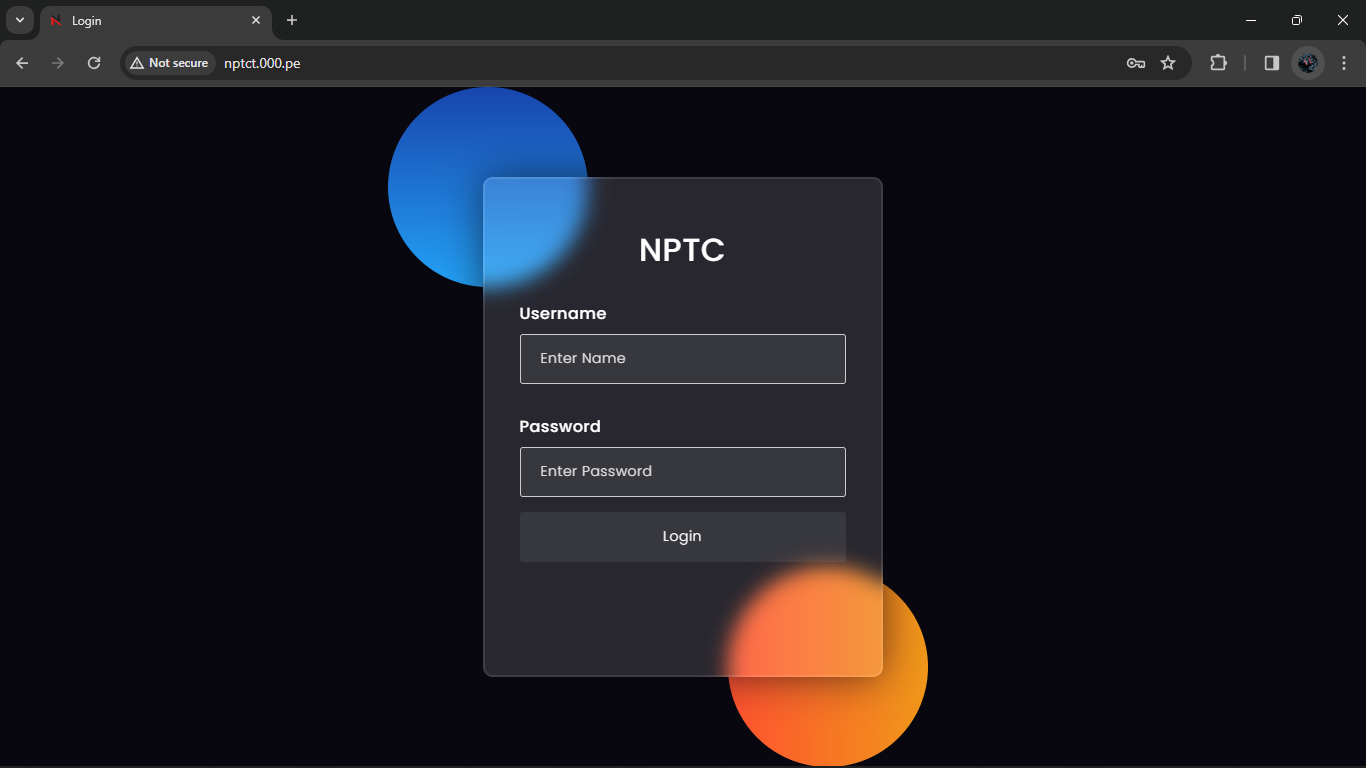
1. Jesus Castagnetto, Sascha Schumann, “Professional Php Programming”, Addison wisely Publication, Fifth Edition.
2. Jay Greenspan, Brad Bulgar, “MySQL/Php Database Applications”, Tata McGraw-Hill Publishing Company, Third Edition.
3. Joel murach, “Murach php and mysql”, mike murach and associate Inc, Third Edition.
4. Kevin tatroe, “Programing php: creating dynamic wesites”,o-reilly, Third Edition.
5. Abraham Silberschatz, Henry F. Korth and S. Sudarshan, Database System Concepts, McGraw-Hill Education (Asia), Fifth Edition, 2006

**REFERENCES WEBSITE**

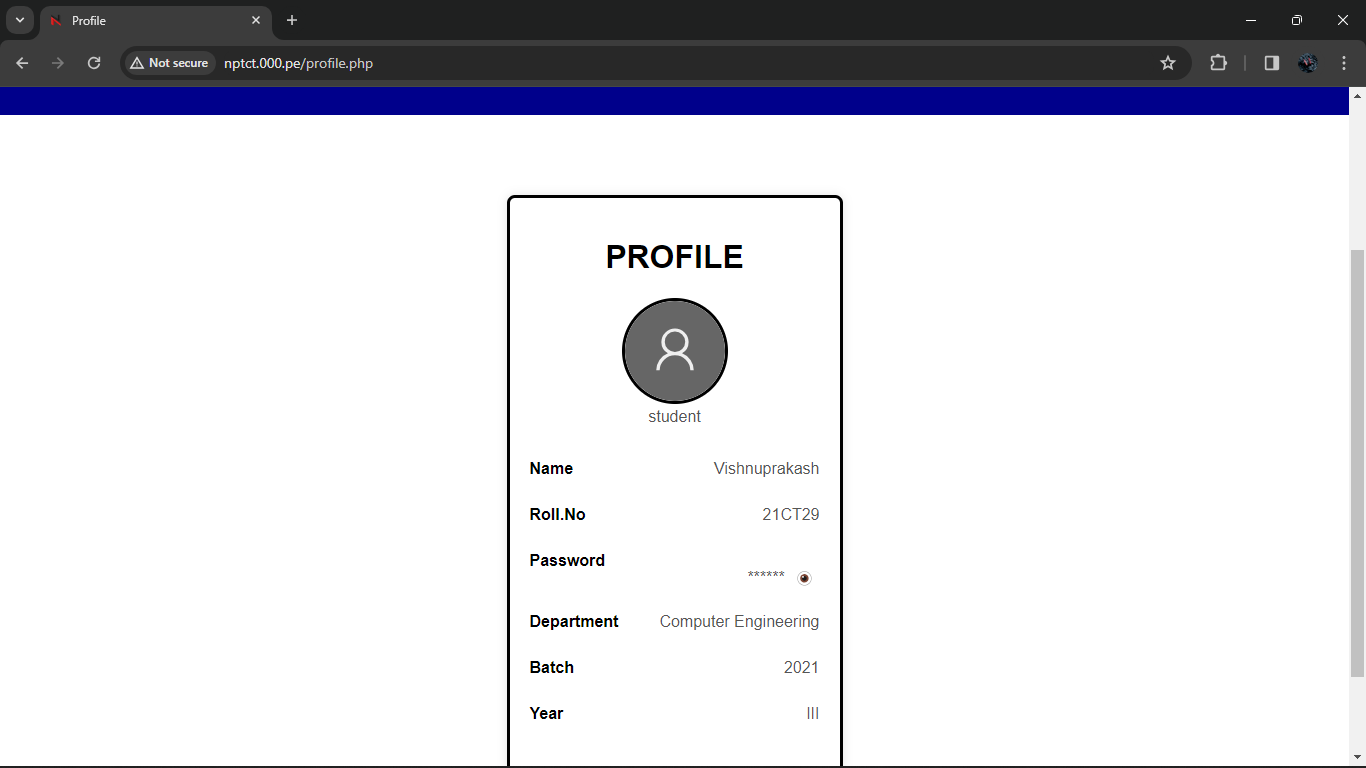
1. [www.onlinetutorial.com](http://www.onlinetutorial.com)
2. [www.chatgpt.com](http://www.chatgpt.com)
3. [www.oreilly.com](http://www.oreilly.com)
4. [www.databasestar.com](http://www.databasestar.com)
5. [www.webdesign.about.com](http://www.webdesign.about.com)
6. www.w3schools.com

**12.0 APPENDIX**

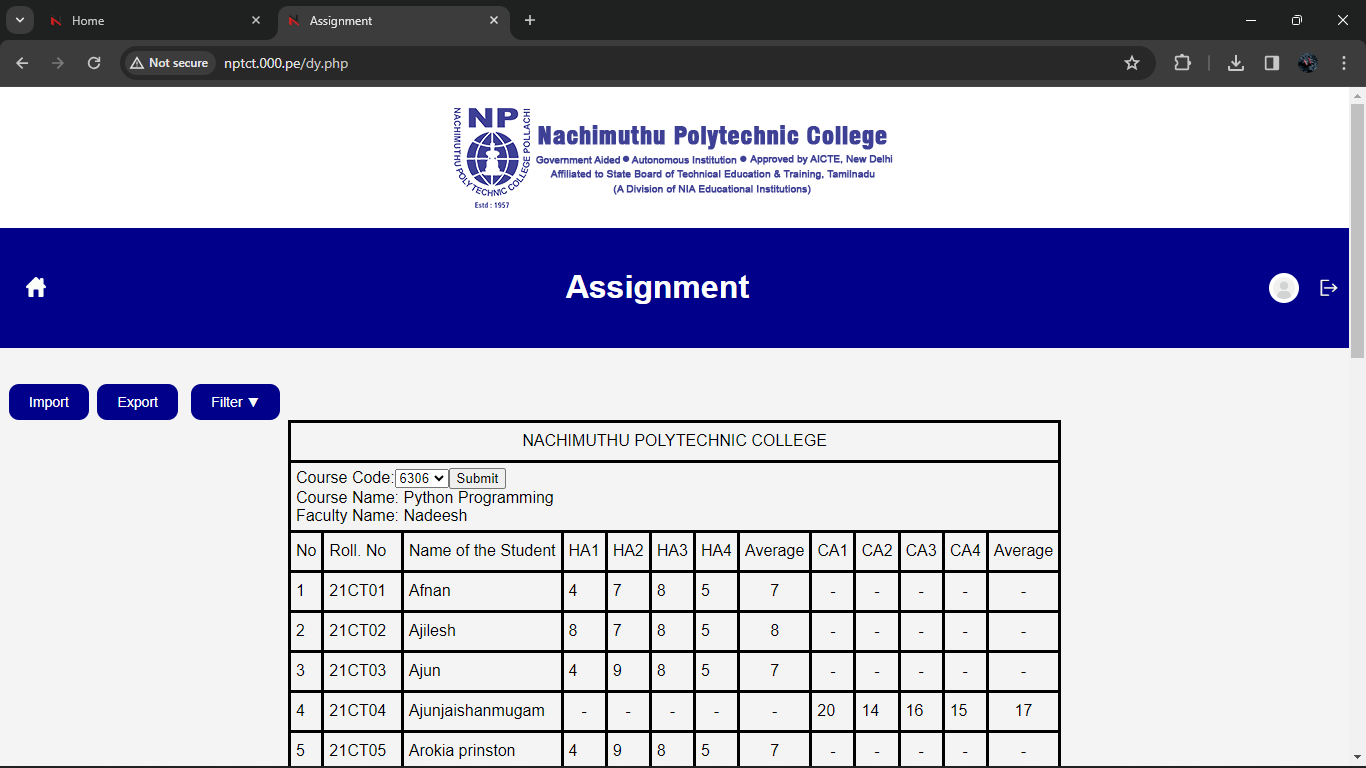
**12.1 SCREENSHOT:**

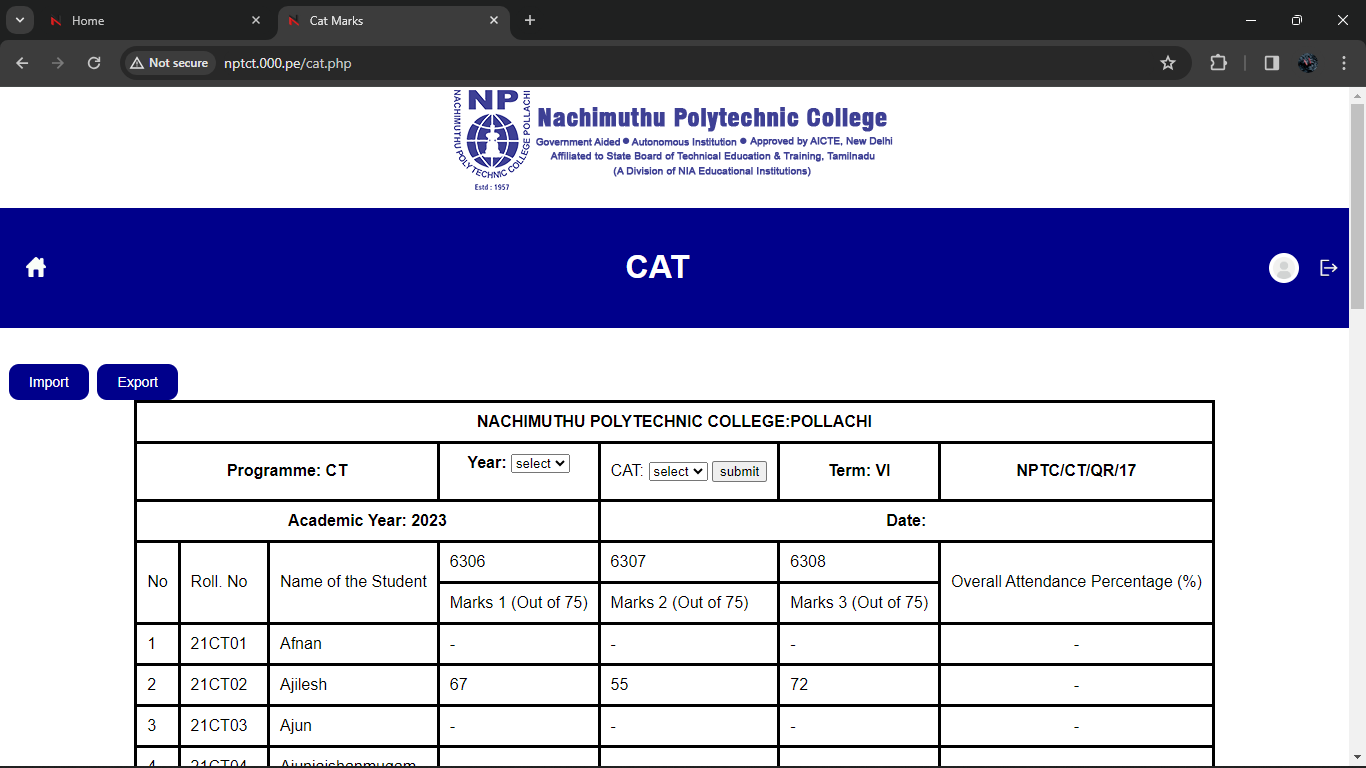


[Fig.1 Login Page]

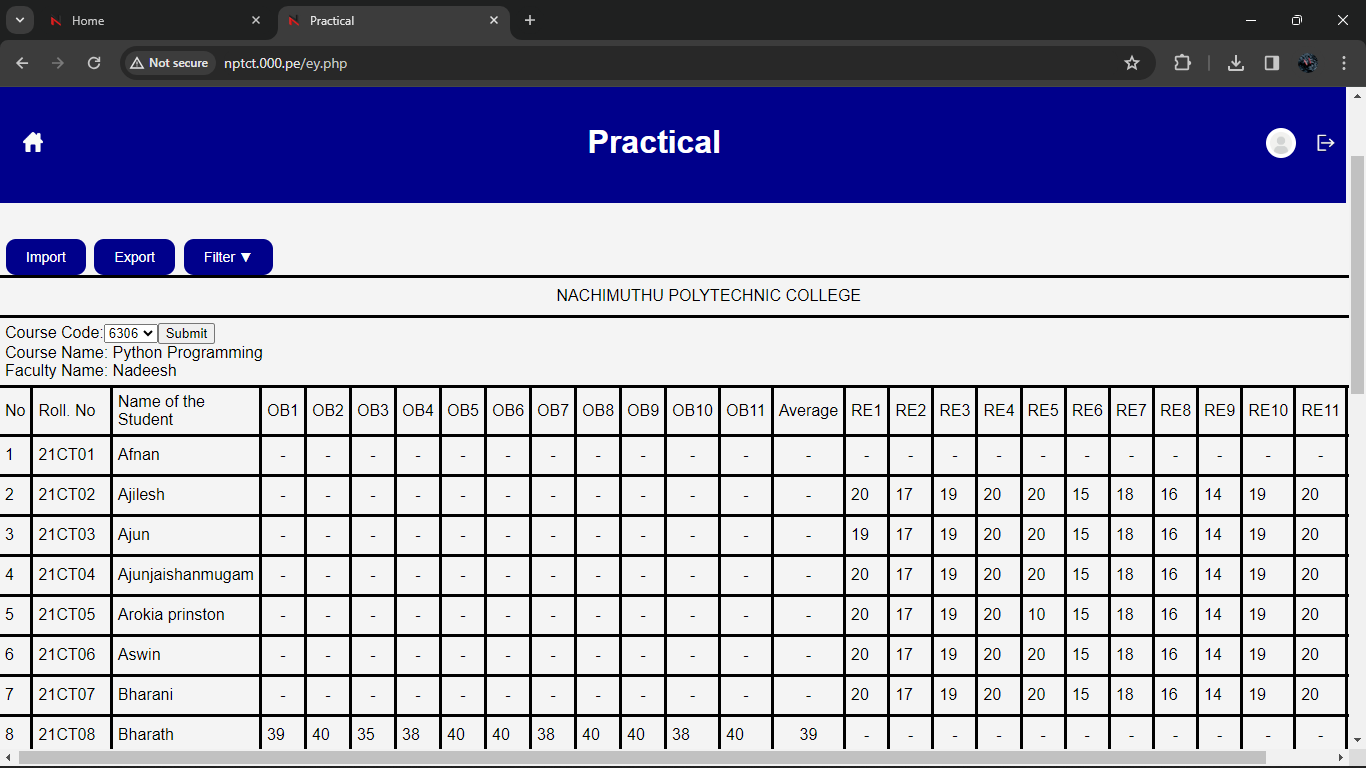


[Fig.2 Profile Page]

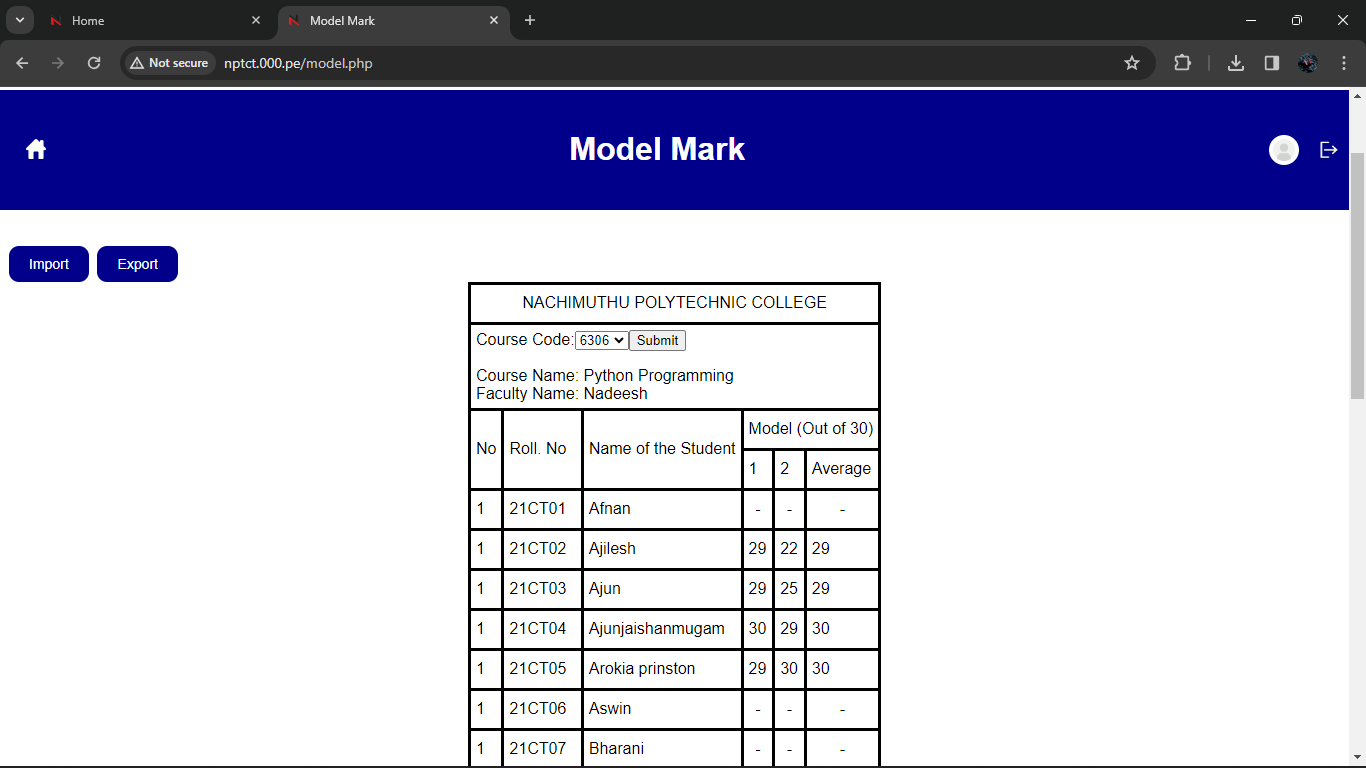


[Fig.3 Assignment Page]

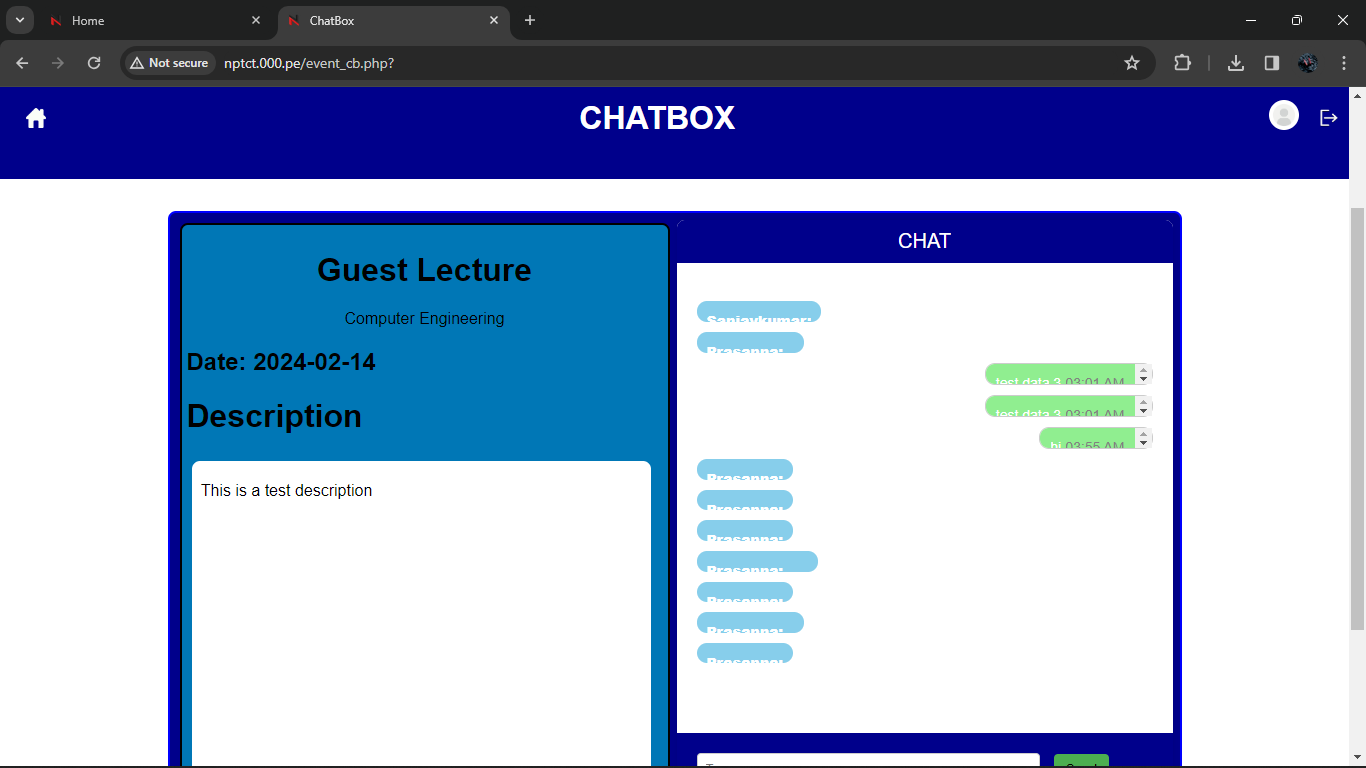
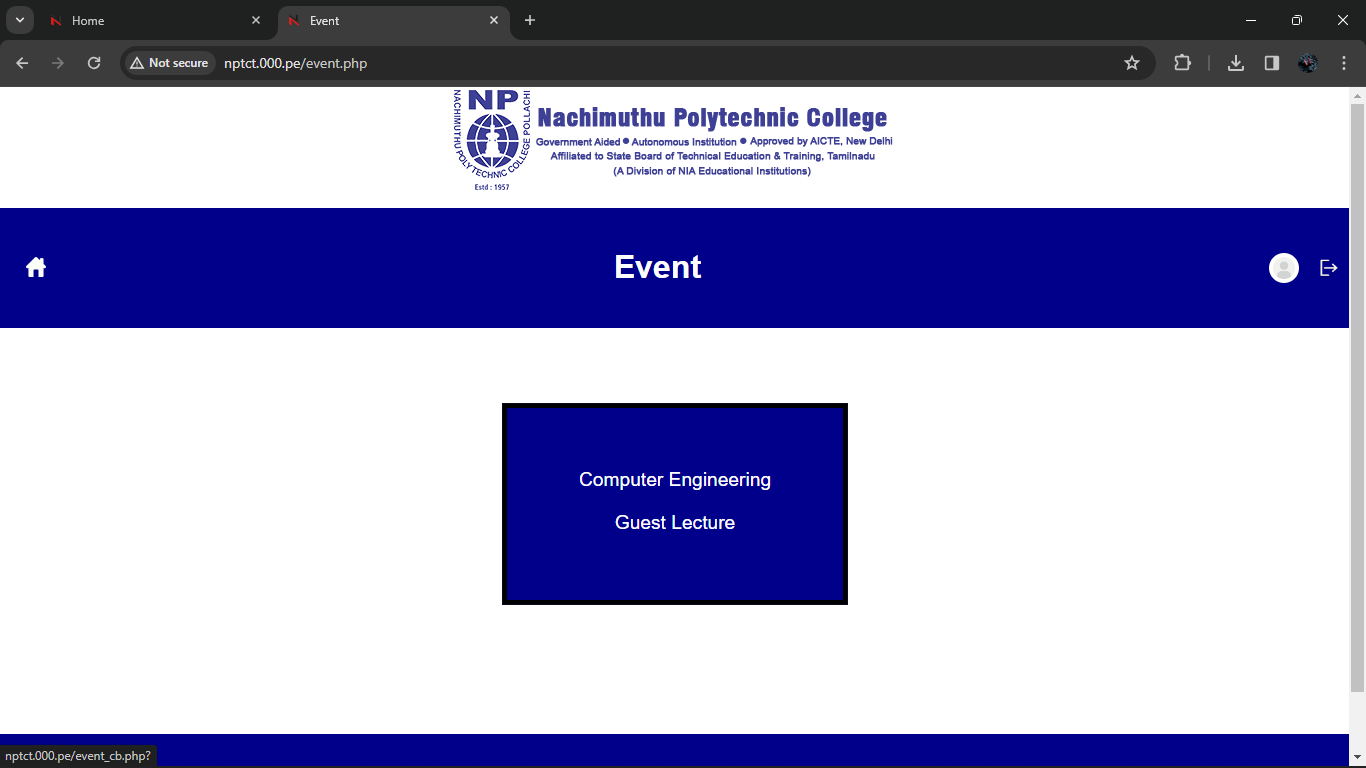
[Fig.4 Cat Page]



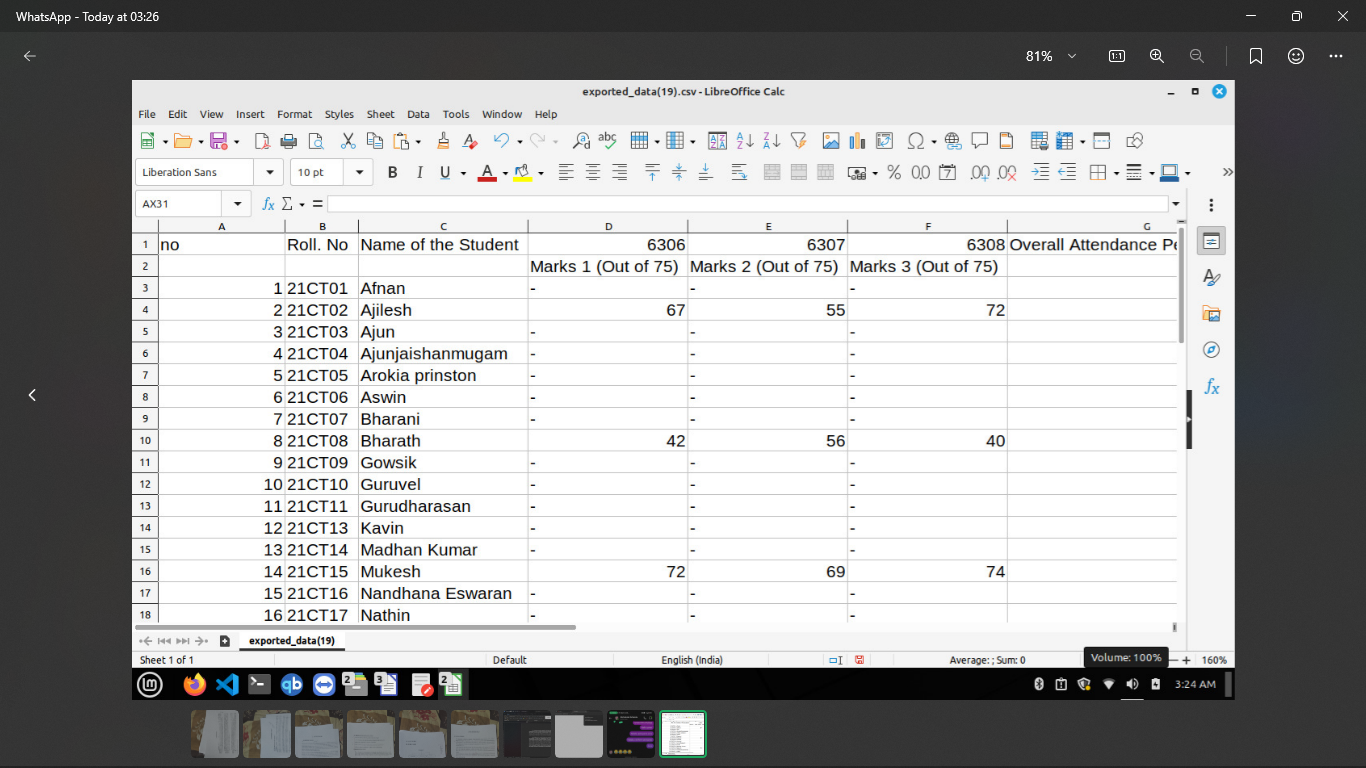
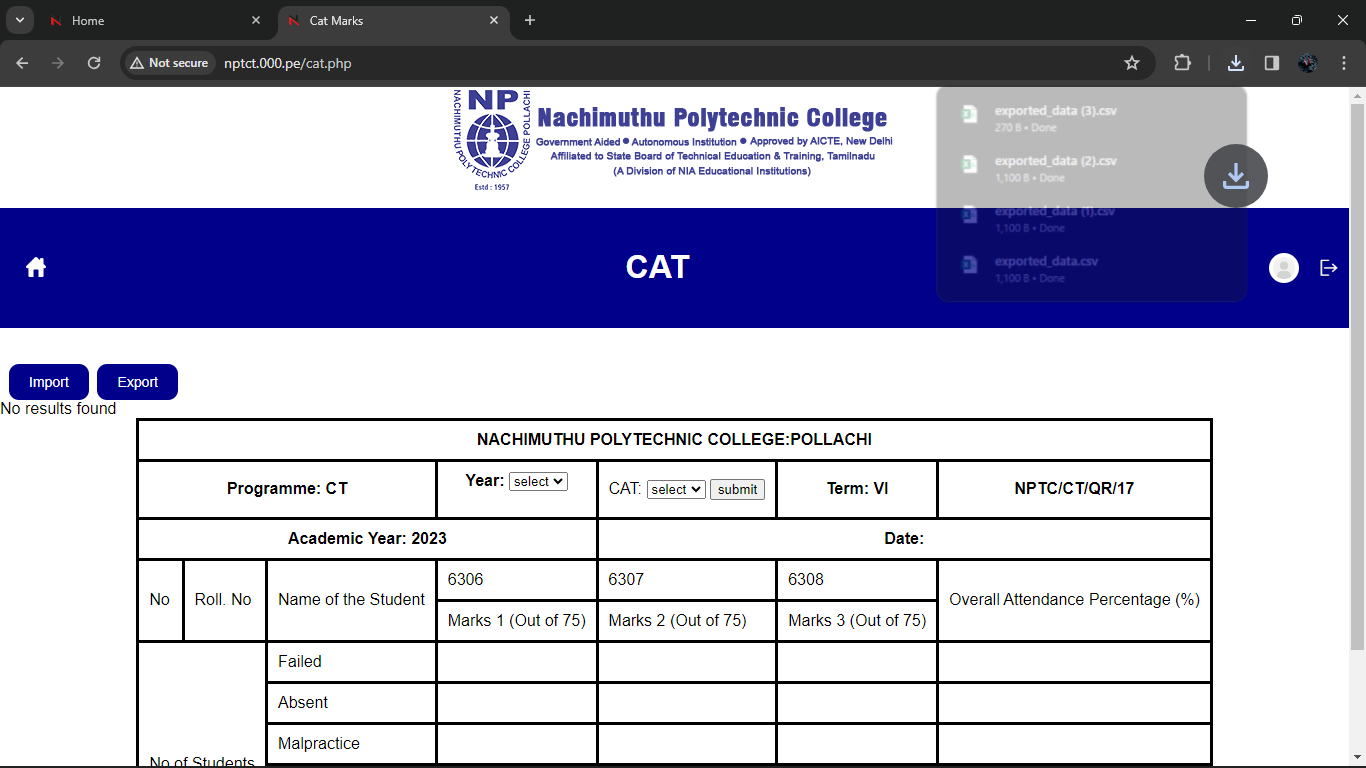
[Fig.5 Practical Page]



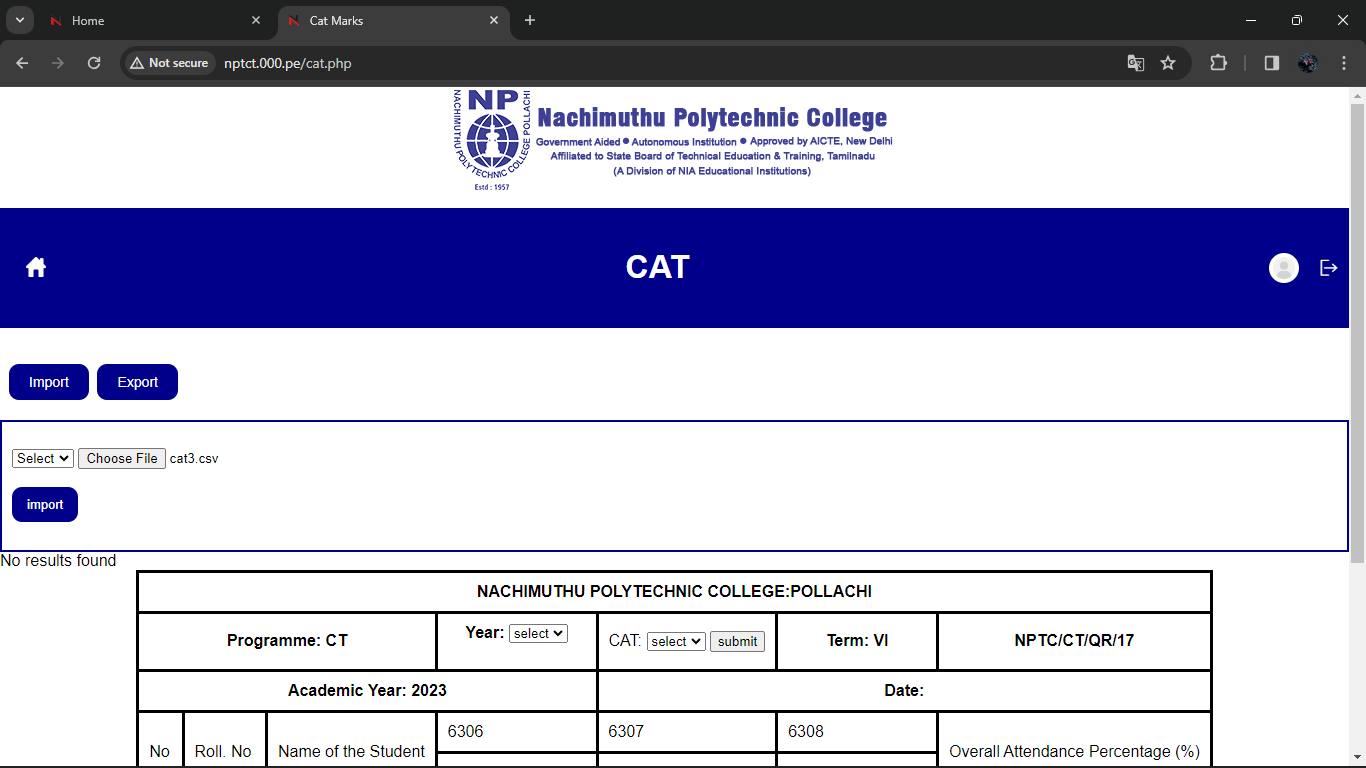
[Fig.6 Model Page]

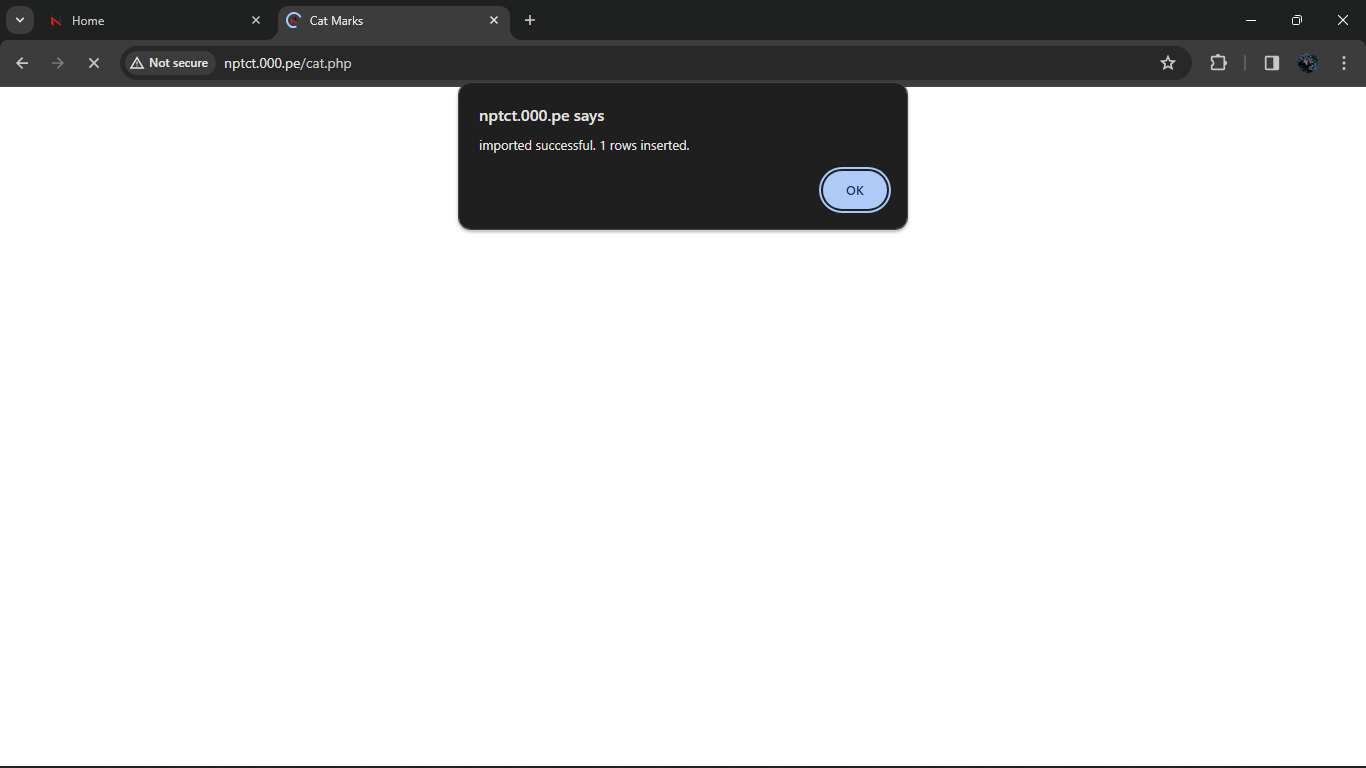


[Fig.7 Event Page]



[Fig.8 Export]





[Fig.9 Import]

**12.2 Sample coding:**

**<?php**

include 'conn.php';

session\_start();

//connection:

$conn =mysqli\_connect($host,$username,$password,$database) or die(mysqli\_error($conn));

if (!$conn)

{

die("Connection failed: " . mysqli\_connect\_error());

}

//log\_variables:

if (isset($\_POST['submit'])) {

$username = $\_POST['username'];

$password = $\_POST['password'];

}

//query setup:

$query = "SELECT \* FROM Student WHERE stud\_rno = '$username' AND stud\_pwd = '$password'";

$result = $conn->query($query);

//student verification:

if ($result->num\_rows > 0) {

echo "hi";

$row = $result->fetch\_assoc();

$\_SESSION['uname'] = $username;

$\_SESSION['password'] = $password;

$\_SESSION['previlege'] = 'student';

header("Location: home2.php");

exit();

}

else {

//staff verification:

$query = "SELECT \* FROM staff WHERE staff\_no = '$username'";

$result = $conn->query($query);

if ($result->num\_rows > 0) {

$row = $result->fetch\_assoc();

$\_SESSION['uname'] = $row['staff\_no'];

$\_SESSION['password'] = $row['staff\_no'];

$\_SESSION['previlege'] = 'staff';

header("Location: home2.php");

}//if not both:

else{

$\_SESSION['error'] = "<p style='color:red;font-size:11px;'>\*Incorrect username or password</p>";

header("Location: index.php");

if($\_SESSION['error\_count']<1){

$\_SESSION['error\_count'] = 1;

}

if($\_SESSION['error\_count']>=1){

$\_SESSION['error\_count'] = $\_SESSION['error\_count']+1;

}

if($\_SESSION['error\_count']>=5){

$\_SESSION['log\_block\_status'] = "yes";

$\_SESSION['log\_block\_time'] = time()/\*date("h:i:s")\*/;

$\_SESSION['log\_block\_off\_time'] = $\_SESSION['log\_block\_time'] + 60 \* 5;

}

else{

unset($\_SESSION['log\_block\_status']);

//unset($\_SESSION['error\_count']);

unset($\_SESSION['log\_block\_off\_time']);

unset($\_SESSION['log\_block\_time']);

}

}

}

**?>**

**<?php**

session\_start();

**?>**

<!DOCTYPE html>

<html lang="en">

<head>

<title>Login</title>

<link rel="stylesheet" type="text/css" href="log.css">

<link rel="preconnect" href="https://fonts.gstatic.com">

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css">

<link href="https://fonts.googleapis.com/css2?family=Poppins:wght@300;500;600&display=swap" rel="stylesheet">

<link rel="icon" href="favicon.ico">

</head>

<body >

<div class="rainbow">

<div class="background">

<div class="shape"></div>

<div class="shape"></div>

</div>

</div>

<form method="POST" action="log.php">

<h3>NPTC</h3>

<label for="username">Username</label>

<input type="text" id="username" name="username" placeholder="Enter Name" >

<label for="password">Password</label>

<input type="password" id="password" name="password" placeholder="Enter Password">

**<?php**

echo $\_SESSION['error'];

//echo $\_SESSION['error\_count'];

//echo $\_SESSION['log\_block\_off\_time'];

unset($\_SESSION['error']);

**?>**

<input type="submit" name="submit" value="Login">

</form>

</body>

</html>