1.INTRODUCTION

The NSS Nirmala project aims to engage students in meaningful community service, fostering a sense of social responsibility and leadership. Through various initiatives, NSS encourages participants to address local issues, promote social welfare, and develop skills while making a positive impact on society. This project not only enhances personal growth but also strengthens community bonds, empowering youth to be proactive change-makers.

NSS Nirmala is an NSS management system which is an online platform developed to facilitate the coordination and management of NSS events and activities. Centralizes the work of various committees—such as the Program, Attendance, Media, Report, and Bill Committees—allowing each group to efficiently carry out their respective duties. By enabling features such as event creation, attendance tracking, report submission, and media management, the platform simplifies the entire process of organizing and documenting NSS events. The platform helps in reducing manual paperwork, improving communication, and tracking event progress systematically.

1.1 OBJECTIVE OF THE PROJECT

The National Service Scheme (NSS) serves as a platform to enhance students' skills through social and community service activities. The success of each NSS program depends on various processes, including attendance tracking, program committee management, billing and purchasing, report making and media coordination.

Our "NSS NIRMALA" mini project aims to streamline these maintenance tasks, thereby enhancing the overall efficiency of NSS operations. This project will automate critical functions such as attendance tracking, program details management, billing and purchasing for NSS programs, media control, and report generation. By implementing an online platform, we intend to simplify these tasks for program officers, secretaries, and volunteers. Currently, these responsibilities are managed manually, which is both timeconsuming and labor-intensive.

2.SYSTEM ANALYSIS

The system analysis of the NSS Website Management System focuses on streamlining the management of NSS activities by integrating eight interconnected modules—Admin, Programme Officer, Attendance Committee, Program Committee, Report Committee, Media Committee, Bill Committee, and Volunteers. Each committee plays a distinct role, ensuring that event planning, financial management, attendance tracking, media documentation, and report creation are handled effectively. The system is designed for collaboration, with the Programme Officer reviewing and approving or rejecting the details provided by each committee, while the Admin has a read-only view for monitoring purposes. This approach enhances transparency, ensures accountability, and reduces manual errors through automation, particularly in attendance tracking and financial management. The centralized structure allows for real-time updates and efficient coordination across committees, ensuring that NSS activities are well-organized and executed smoothly. By analyzing the workflow and interactions between various modules, the system enhances operational efficiency, leading to improved management of NSS initiatives.

2.1 EXISTING SYSTEM

The NSS system currently operates through manual processes managed by volunteer, covering attendance tracking, program management, billing, purchasing, and media coordination.

Drawback of existing system:

- Time-consuming attendance tracking and submission.
- Limited scalability.
- Cumbersome billing and purchasing processes.
- Difficulty in coordinating and managing media-related tasks.
- Increased likelihood of errors and delays in operations.
- Risk of data loss in manual system.

2.2 PROPOSED SYSTEM

The proposed Website Management System is an advanced, role-based platform that brings together all key participants involved in managing National Service Scheme (NSS) activities under one comprehensive system. It allows each stakeholder—Admin, Programme Officer, and members of the Program, Attendance, Media, Report, and Bill Committees—to access a dedicated module tailored to their specific responsibilities. The Program Committee can easily create, propose, and update event details, while the Attendance Committee tracks volunteer participation with precision, logging each volunteer's attendance. The Media Committee uploads event photos and videos, creating a visual archive for documentation and promotional purposes. The Report Committee compiles detailed reports about the event's objectives, outcomes, and lessons learned, while the Bill Committee ensures accurate financial tracking and expense management.

Advantages of proposed System:

- Efficiency Boost: Automates attendance tracking, program management, billing, purchasing, media control, and report generation, reducing manual effort and time consumption.
- Streamlined Operations: Simplifies tasks currently managed manually, such as attendance marking and bill management, enhancing overall operational efficiency.
- Improved Accessibility: Provides an online platform for easy access to NSS program details, including photos and reports, facilitating transparency and accessibility for stakeholders.
- Enhanced Coordination: Enables better coordination among program officers, secretaries, and volunteers by centralizing information and communication channels.
- Error Reduction: Minimizes errors associated with manual data entry and processing, ensuring
- Centralized Data Management: All information is stored in one location, making it easy to access, update, and retrieve records as needed.

- Customizable User Dashboards: Personalized dashboards allow users to focus on relevant information, making it easier to track tasks and responsibilities.
- Environmental Benefits: Reduced paperwork and keeping digital record.

2.3 SYSTEM REQUIREMENT SPECIFICATION

A software requirements specification (SRS) is a comprehensive description of the intended purpose and environment for software under development. The SRS fully describes what the software will do and how it will be expected to perform. An SRS minimizes the time and effort required by developers to achieve desired goals and also minimizes the development cost. A good SRS defines how an application will interact with system hardware, other programs and human users in a wide variety of real-worked situations.

Customer requirements

- The system should be fast
- User friendly
- Maintaining security of data
- Efficiency in data retrieval and management

What the developer needs to know?

- Must know the existing system and its drawbacks
- Must know what will be needed in the proposed system

Business Requirements

The system should be feasible both to the developer and client. It should be effective and should be able to complete in time. Developer should be responsible for developing the system, install the software and update the software whenever necessary, conducting any user training that may be needed for using the system.

User Requirements

The user requirement(s) specification is a document usually that specifies the requirements the user expects from software to be constructed in a software project.

- Admins need a comprehensive view of all activities.
- Programme Officers must review and approve/reject events.
- Committees require portals for managing tasks.
- Volunteers need a simple interface for event participation.
- Secure logins for privacy and data integrity.
- Real-time updates for seamless communication.

Functional Requirements

Functional Requirement defines what a system is supposed to do. They can perform the following functionalities:

- Admin can view details from all committees.
- Programme Officer can approve/reject events proposed by the Program Committee.
- Program Committee manages and proposes events.
- Bill Committee calculates and manages event costs.
- Media Committee uploads event photos/videos.
- Report Committee creates and submits event reports.
- Attendance Committee tracks volunteer attendance.
- Volunteers register, view, and participate in events.

2.3.1 Hardware Specifications

Processor	: intel(R) Pentium or Intel(R) Core(TM) i5
Speed	: 2.30 GHz or higher
System bus	: 32bits or 64 bits
Memory	: 8.00 GB or higher

Hard disk	: 256 GB or higher
Monitor	: 14" LCD monitor.
Keyboard	: QWERTY keyboard with 104 keys.
Pointing Device	: Integrated touchpad

2.3.2 Software Specifications

Operating System	: Windows 11
Front End	: PHP, HTML, CSS Scripting
Scripting Language	: JavaScript
Back End	: MYSQL Server Web Server Wamp 2.0
Browser	: Google Chrome

2.3.3 Front End

HTML

Hypertext Markup Language (HTML), the languages of the World Wide Web (WWW), allows users to produce Web pages that include text, graphics and pointer to other Web pages (Hyperlinks). HTML is not a programming language, but it is an application of ISO Standard 8879, SGML (Standard Generalized Markup Language), but specialized to hypertext and adapted to the Web. The idea behind Hypertext is that instead of reading text in rigid linear structure, we can easily jump from one point to another point. We can navigate through the information based on our interest and preference. A markup language is simply a series of elements, each delimited with special characters that define how text or other items enclosed within the elements should be displayed. Hyperlinks are underlined or emphasized works that load to other documents or some portions of the same document.

CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG, and XUL, and is applicable to rendering in speech, or on other media.

Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

CSS is designed primarily to enable the separation of document content from document presentation, including aspects such as layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility, and offer greater control in specifying presentation characteristics. It enables multiple HTML pages to share the same formatting by specifying styles in a single stylesheet.

JAVASCRIPT

JavaScript is a script-based programming language that was developed by Netscape Communication Corporation. JavaScript was originally called Live Script and renamed as JavaScript to indicate its relationship with Java. JavaScript supports the development of both client and server components of Web-based applications. On the client side, it can be used to write programs that are executed by a Web browser within the context of a Web page. On the server side, it can be used to write Web server programs that can process information submitted by a Web browser and then update the browser's display accordingly. Even though JavaScript supports both client and server Web programming, we prefer JavaScript at Client- side programming since most of the browsers supports it. JavaScript is almost as easy to learn as HTML, and JavaScript statements can be included in HTML documents by enclosing the statements between a pair of scripting tags,

<SCRIPTS>…</SCRIPT>.

<SCRIPT LANGUAGE = "JavaScript">

JavaScript statements

</SCRIPT>

2.3.4 Back End

PHP

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. Originally created by Rasmus Lerdorf in 1994, the PHP reference implementation is now produced by The PHP Group. PHP originally stood for Personal Home Page, but it now stands for the recursive backronym PHP: Hypertext Preprocessors.

PHP code may be embedded into HTML code, or it can be used in combination with various web template systems, web content management systems, and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications.

MYSQL

MySQL is an open-source relational database management system (RDBMS). A relational database organizes data into one or more data tables in which data may be related to each other; these relations help structure the data. SQL is a language that programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups. MySQL databases are data containers used to store and manage information. Users may manage MySQL databases using included command-line tools or install MySQL Workbench, a GUI-based tool available as a separate download

2.4 FEASIBILITY ANALYSIS

Feasibility analysis is an essential step to evaluate the practicality and potential challenges of developing the NSS Website Management System. It includes assessing whether the project can be successfully implemented given the available resources, technologies, and support from the stakeholders. The analysis is broken down into four primary categories: technical feasibility, economic feasibility, operational feasibility, and behavioral feasibility.

Technical Feasibility

Technical Feasibility: The NSS Website Management System is technically feasible due to the availability of well-established web technologies such as HTML, CSS, and JavaScript for the frontend, and PHP or Node.js for backend development. These technologies are widely supported, making it easy to find resources and developers with expertise in these areas. Additionally, MySQL or other relational databases provide reliable data storage and retrieval mechanisms. Hosting solutions for web applications are cost-effective and scalable, ensuring that the system can handle increasing numbers of events, volunteers, and data over time without significant infrastructure changes. The adaptability of the platform also allows for future upgrades, making it sustainable for long-term use.infrastructure can support scalability.

Economic Feasibility

This project is economically feasible as it reduces manual workload and reliance on paperwork, lowering overall operational costs. While the initial cost of development might include hiring developers, server hosting, and domain purchase, the long-term cost savings in managing NSS events and tracking attendance are substantial. Additionally, open-source technologies and tools can be used to minimize expenses, and the system can be developed incrementally to match budget constraints. Over time, the reduction in labor costs and increased efficiency make the investment worthwhile. The automation of repetitive tasks, such as attendance tracking and report generation, helps cut down administrative overhead, which further justifies the system's financial feasibility.

Operational Feasibility

Operational Feasibility: Operationally, the system is designed to streamline and simplify the management of NSS events by offering intuitive modules for each committee. The clearly defined roles for each user (e.g., Admin, Programme Officer, and committees) ensure smooth day-to-day operations. By automating attendance tracking, event management, and report generation, the system eliminates manual errors and delays, making the workflow more efficient. Furthermore, the centralized data access improves coordination among committees, ensuring that each task is completed in a timely manner with minimal confusion or overlap. The system's ease of use minimizes the learning curve, allowing users to quickly adapt to the platform, and enhances overall operational effectiveness.

Behavioural Feasibility

Behavioral feasibility is ensured as the system promotes accountability and collaboration among the various committees and volunteers. The roles and responsibilities of each committee are clearly outlined, encouraging users to perform their duties effectively. The Programme Officer's authority to approve or reject events adds a level of oversight, ensuring quality control. Moreover, volunteers and committee members are likely to adopt the system quickly due to its user-friendly interface and the reduced manual workload it provides. Training requirements are minimal, and the system is designed to be intuitive, which makes the transition to the platform easier for all users involved. Additionally, the transparency in tracking attendance, event progress, and finances fosters trust and cooperation among the users, ensuring their willingness to engage with the system consistently.

2.5 DATA FLOW DIAGRAM (DFD)

2.5.1 INTRODUCTION TO DATAFLOW DIAGRAM

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. It differs from the flowchart as it shows the data flow instead of the control flow of the program. A data flow diagram can also be used for the visualization of data processing (structured design).

Data flow diagrams were invented by Larry Constantine, the original developer of structured design, based on Martin and Estrin's "data flow graph" model of computation.

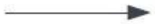
Data flow diagrams (DFDs) are one of the three essential perspectives of Structured System Analysis and Design Method SSADM. The sponsor of a project and the end users will need to be briefed and consulted throughout all stages of a system's evolution. With a data flow diagram, users are able to visualize how the system will operate, what the system will accomplish, and how the system will be implemented. The old system’s data flow diagrams can be drawn up and compared with the new system’s data flow diagrams to draw comparisons to implement a more efficient system. Data flow diagrams can be used to provide the end users with a physical idea of where the data will originate, and where it will go.

Developing a data flow diagram helps in identifying the transaction data in the data model. There are different notations to draw data flow diagrams, defining different visual representation for process, data stores, data flow, and external entities. The first step is to draw a data flow diagram (DFD). A DFD, also known as "bubble chart," has the purpose of clarifying system requirements and identifying major transformation that will become program in system design. So, it is starting point of the design phase that functionally decomposes the requirements specification down to the lowest level of details. DFD consists of series of bubbles joined by lines. The bubbles represent data transformation and the lines represent data flow in the system.

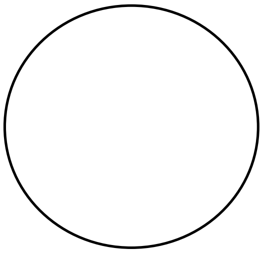
DFD Symbols

- Square - Defines source or destination of the system

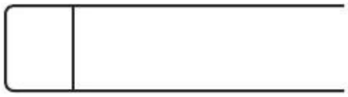
- Data flow – Identifies the data flow



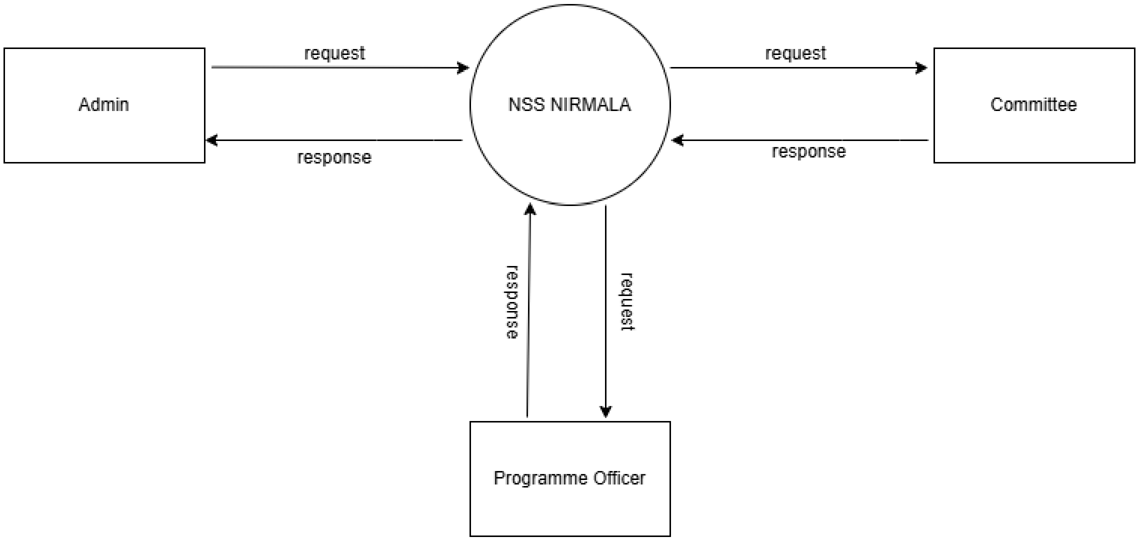
- Bubble – Represent a process that transforms incoming data to upcoming data



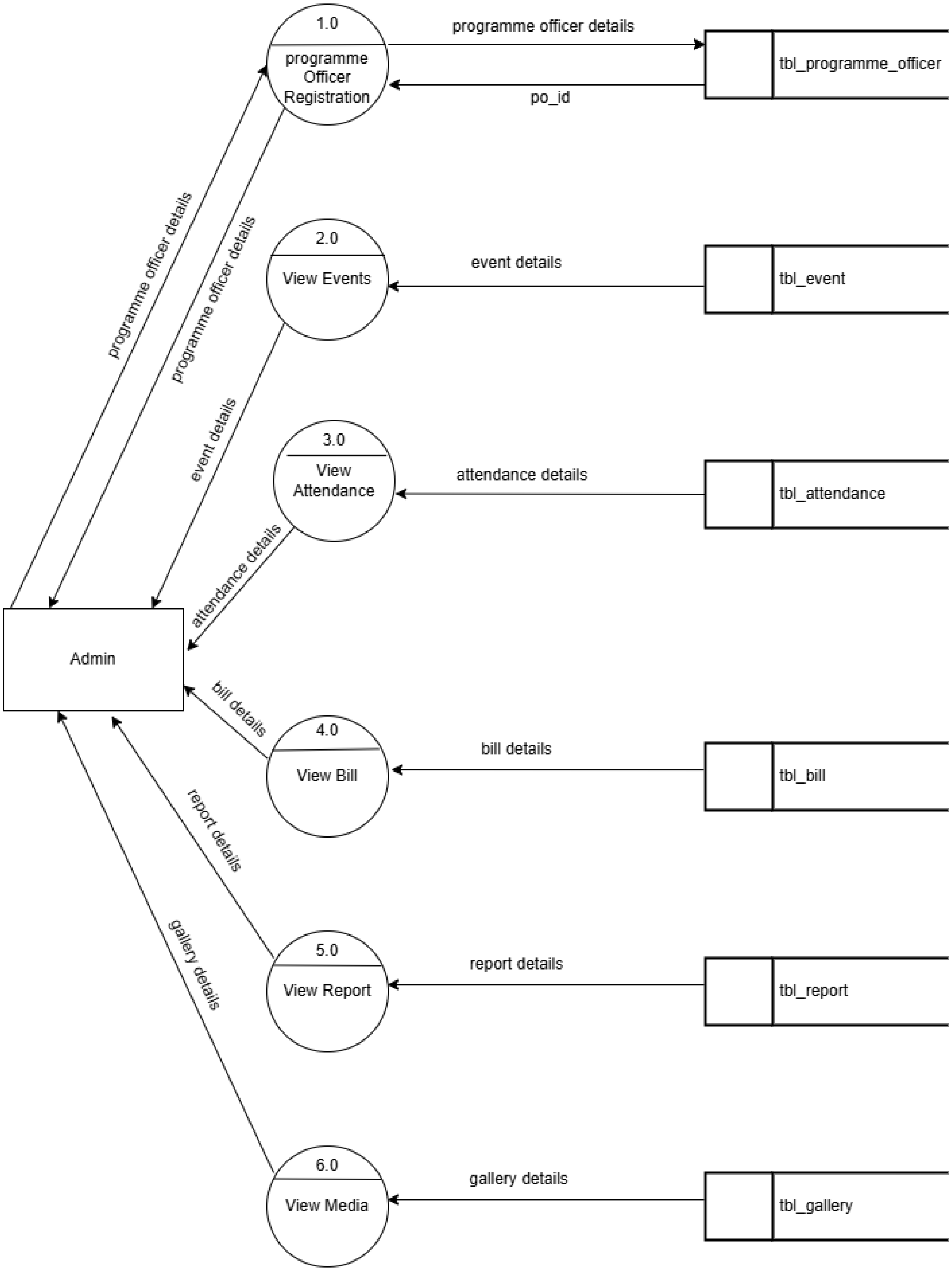
- Open Rectangle – Data Store

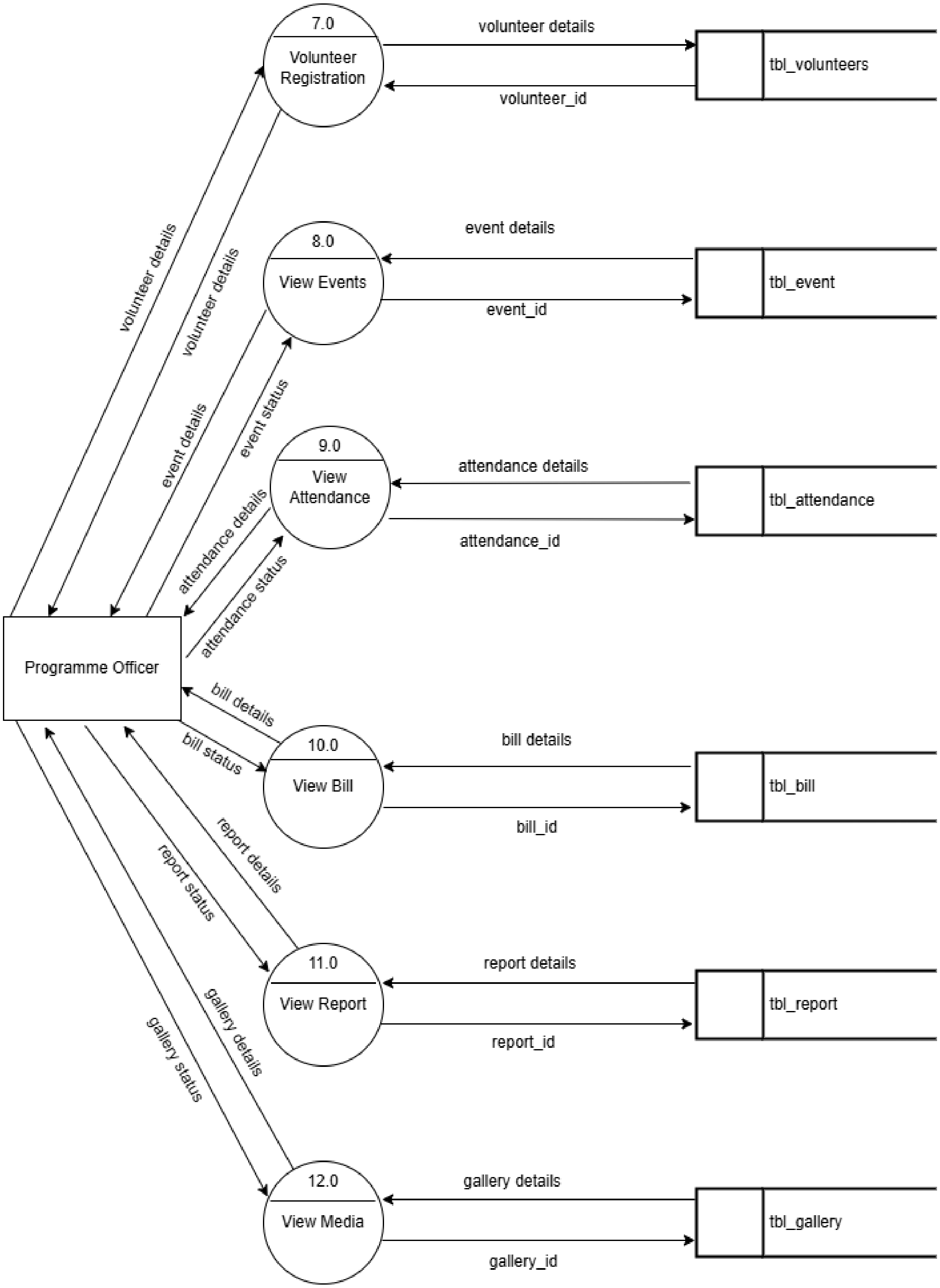


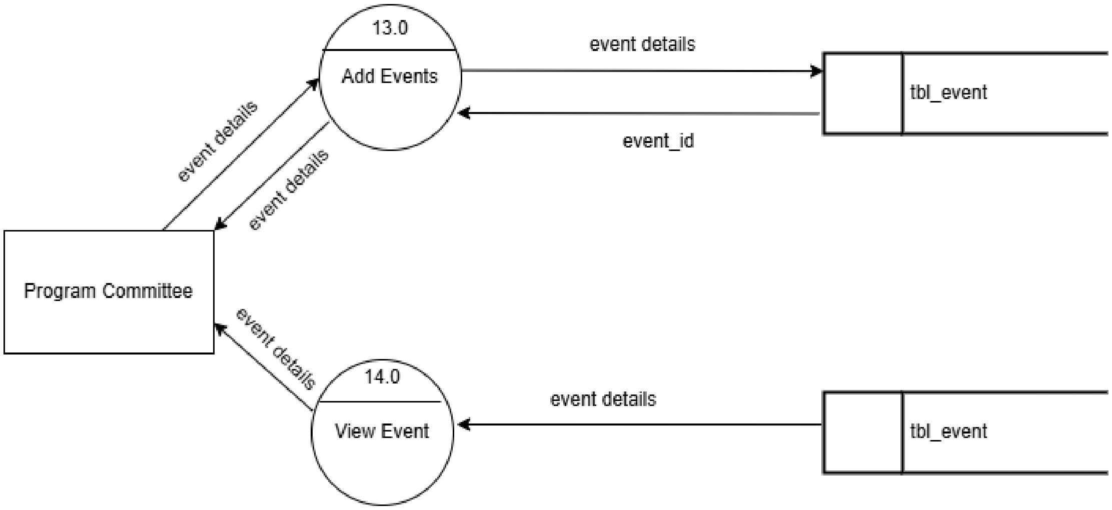
LEVEL 0

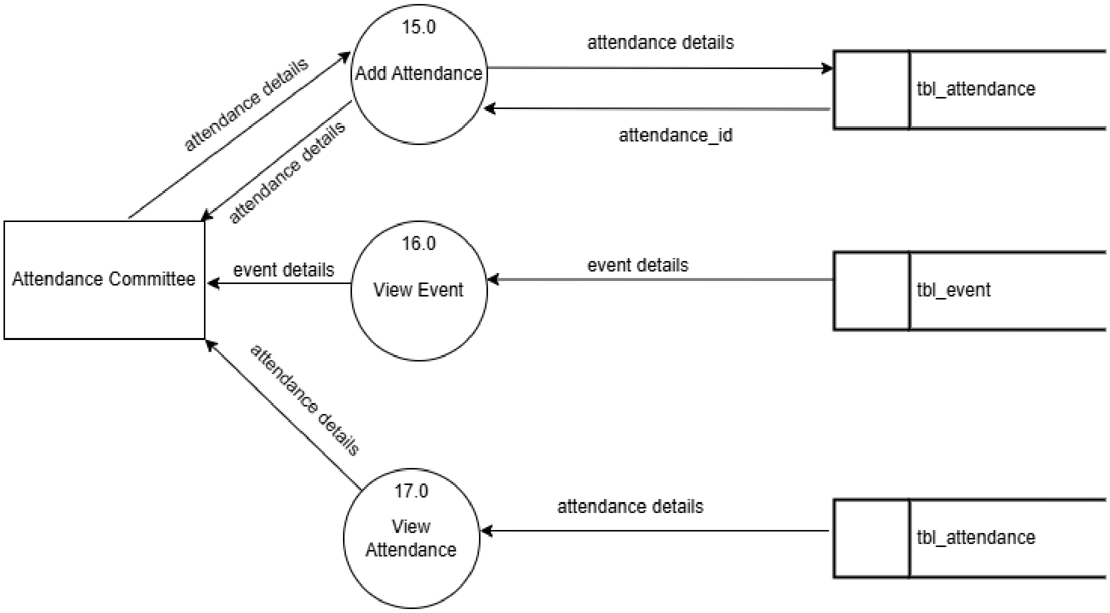

Admin
request
response
NSS NIRMALA
request
response
Committee
response
request
Programme Officer

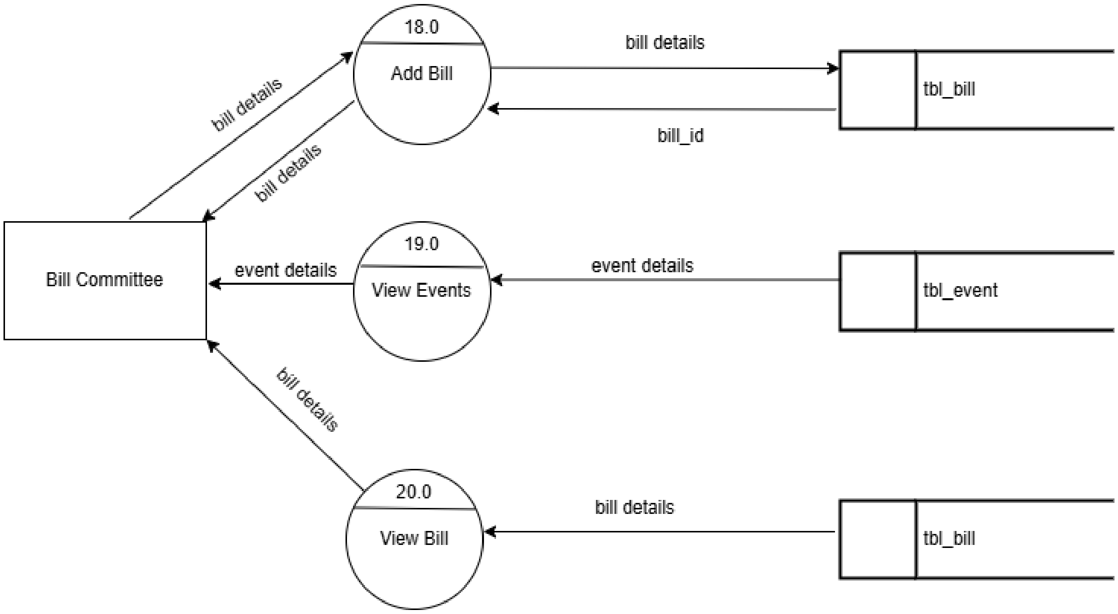
LEVEL 1

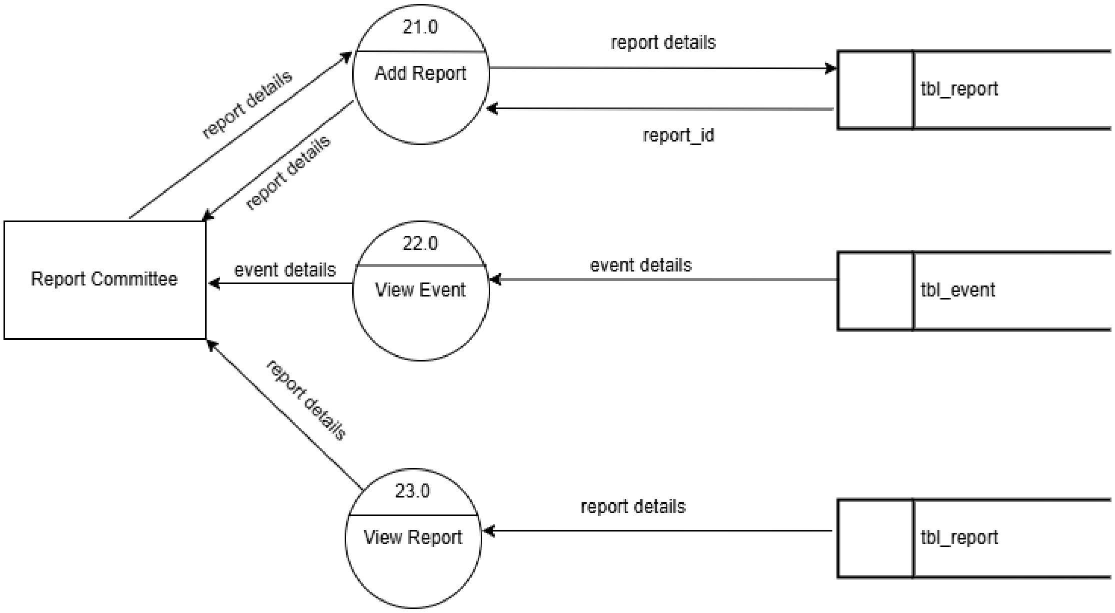

1.0
programme Officer Registration
programme officer details
po_id
tbl_programme_officer
2.0
View Events
event details
tbl_event
3.0
View Attendance
attendance details
tbl_attendance
4.0
View Bill
bill details
tbl_bill
5.0
View Report
report details
tbl_report
6.0
View Media
gallery details
tbl_gallery
Admin
programme officer details
programme officer details
event details
attendance details
bill details
report details
gallery details

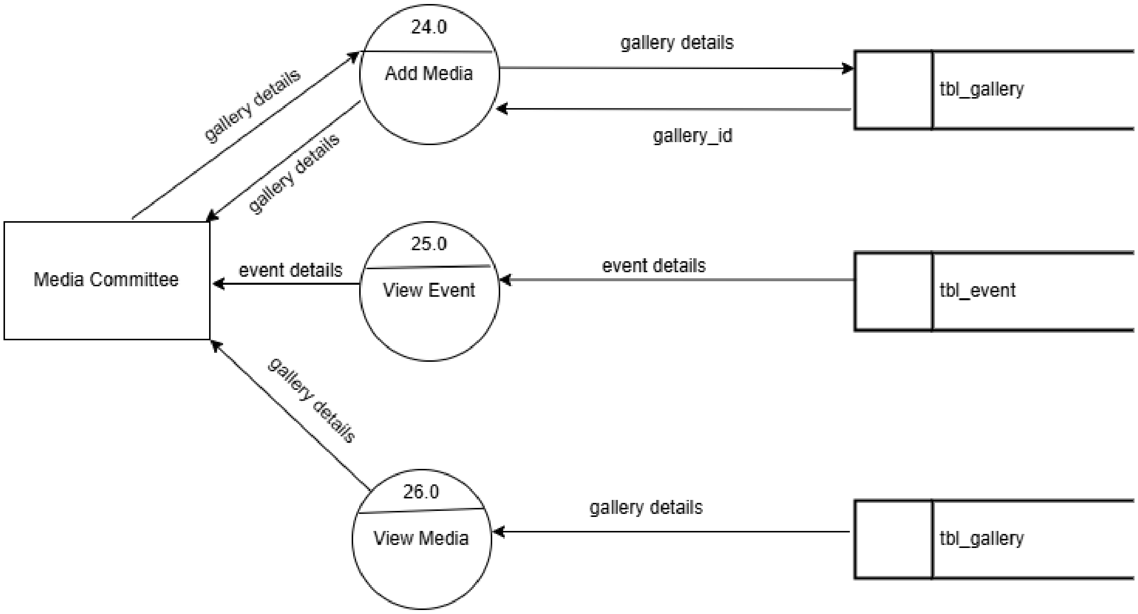

7.0
Volunteer Registration
volunteer details
volunteer_id
tbl_volunteers
8.0
View Events
event details
event_id
tbl_event
9.0
View Attendance
attendance details
attendance_id
tbl_attendance
10.0
View Bill
bill details
bill_id
tbl_bill
11.0
View Report
report details
report_id
tbl_report
12.0
View Media
gallery details
gallery_id
tbl_gallery
Programme Officer
volunteer details
volunteer details
event details
event status
attendance details
attendance status
bill details
bill status
report details
report status
gallery details
gallery status

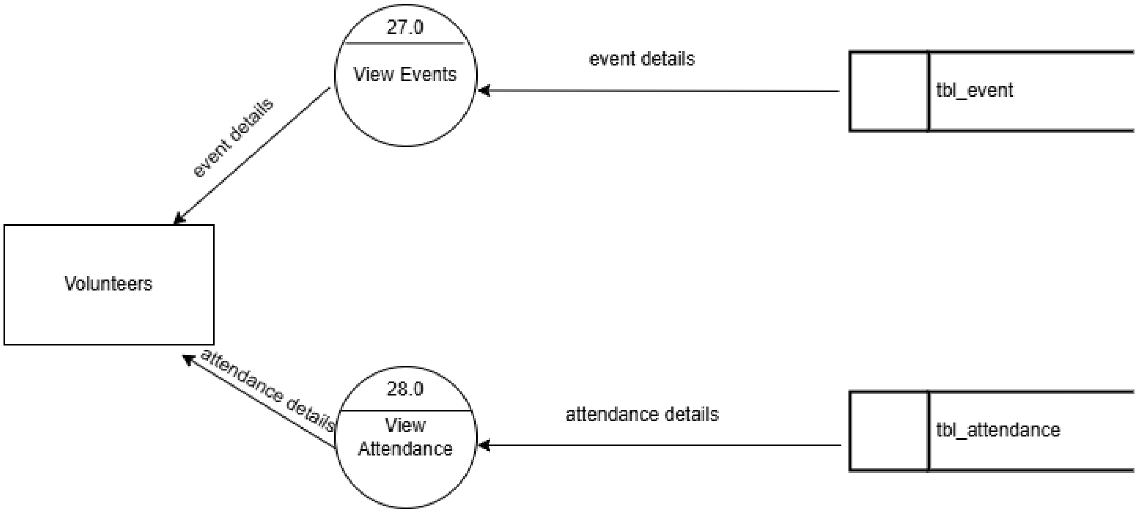

13.0
Add Events
event details
tbl_event
event_id
event details
event details
Program Committee
event details
14.0
View Event
event details
tbl_event


15.0
Add Attendance
attendance details
tbl_attendance
attendance_id
attendance details
attendance details
Attendance Committee
event details
16.0
View Event
event details
tbl_event
attendance details
17.0
View Attendance
attendance details
tbl_attendance


18.0
Add Bill
bill details
bill details
bill details
tbl_bill
bill_id
Bill Committee
event details
19.0
View Events
event details
tbl_event
bill details
20.0
View Bill
bill details
tbl_bill


21.0
Add Report
report details
report details
report details
tbl_report
report_id
Report Committee
event details
22.0
View Event
event details
tbl_event
report details
23.0
View Report
report details
tbl_report


24.0
Add Media
gallery details
gallery details
gallery details
gallery_id
tbl_gallery
Media Committee
event details
25.0
View Event
event details
tbl_event
gallery details
26.0
View Media
gallery details
tbl_gallery


27.0
View Events
event details
tbl_event
event details
Volunteers
attendance details
28.0
View Attendance
attendance details
tbl_attendance

3.SYSTEM DESIGN

3.1 INPUT DESIGN

The quality of the system input determines the quality of the system output. Input specification describes the way data enter the system for processing. Input design features can ensure the reliability of the system and procedure result from accurate data, or they can result in the production or erroneous information. The input design also determines whether the user can interact efficiently with the system. In our system almost all inputs are being taken from the databases. To provide adequate inputs we must select necessary values from the databases and arrange it to the appropriate controls.

Admin

The admin input design consists of registration fields for programme officer which contains email, password, and personal details of programme officer The admin can also view and reply for customer complaints.

Programme Officer

The programme officer input design consists of registration fields for volunteers which contain email, password, and personal details of volunteers and also the input design consists of the approval and rejection of events ,bill ,media and report of different committees

User

Users includes various committee and volunteers will do their jobs assigned to them.

3.2 OUTPUT DESIGN

One of the important features of an information system for users is the output produces. Output is the information delivered to users through the information system. Without quality of the output, the entire system appears to be unnecessary that users will avoid using it. Users generally merit the system solely by its output. To create the most useful output possible. One works closely with the user though an interactive process. Until the result is satisfactory.

Admin

Admins can view all details and updates provided by various committees. They do not have the authority to modify or approve event-related information.

Programme Officer

The Programme Officer acts as the overseer of all NSS activities. They have the authority to approve or reject events and monitor the progress of each committee.

User

Users includes volunteers and committee will do their jobs assigned for them and volunteers can view their attendance and events details.

3.3 TABLE DESIGN

Data design converts the information domain model from analysis into the data structures required for software implementation. It is based on the entity-relationship diagram and data directory. The goal of database technology is to treat data as an integrated organizational resource. A Database Management System (DBMS) protects and organizes data separately from other resources, distinguishing between logical and physical data.

The organization of data in the database aims to achieve three major objectives:

- Data integration
- Data integrity
- Data independence

The databases are implemented using DBMS package. There are 6 major steps in design process. The first 5 steps are usually done on paper and finally the design is implemented.

- Identify the table and relationships
- Identify the data that is needed for each table and relationship
- Resolve the relationship
- Verify the design
- Implement the design

A database stores data in tables, which consist of records (sets of fields). All records in a table share the same fields but contain different information. Key fields establish relationships and organize records. Primary keys uniquely identify records, while foreign keys link records to other tables.

The various database tables that are used in the project are the following:

1. Table Name : tbl_admin

Description : Admin details

Primary Key : admin_id

Field	Data Type	Size	Description
admin_id	INT	--	Unique ID of admin
admin_name	VARCHAR	50	Name of admin
admin_email	VARCHAR	200	Email of admin
admin_password	VARCHAR	50	Password of admin
admin_proof	VARCHAR	300	Proof of admin
admin_photo	VARCHAR	300	Photo of admin

2. Table Name : tbl_programme_officer

Description : Programme Officers details

Primary Key : po_id

Field	Data Type	Size	Description
po_id	INT	--	Unique ID of programme officer
po_name	VARCHAR	50	Name of programme officer
po_email	VARCHAR	200	Email of programme officer
po_password	VARCHAR	50	Password of programme officer
po_proof	VARCHAR	300	Proof of programme officer
po_photo	VARCHAR	300	Photo of programme officer

3. Table Name : tbl_programme_committee

Description : Contain the details of programme officer

Primary Key : pc_id

Field	Data Type	Size	Description
pc_id	INT	--	Unique ID of program officer
pc_name	VARCHAR	50	Name of program officer
pc_email	VARCHAR	200	Email of program officer
pc_password	VARCHAR	50	Password of program officer
pc_proof	VARCHAR	300	Proof of program officer
pc_photo	VARCHAR	300	Photo of program officer

4. Table Name : tbl_attendance_committee

Description : Contain the details of members who are part of attendance committee

Primary Key : ac_id

Field	Data Type	Size	Description
ac_id	INT	--	Unique ID of attendance committee
ac_name	VARCHAR	50	Name of attendance committee
ac_email	VARCHAR	200	Email of attendance committee
ac_password	VARCHAR	50	Password of attendance committee
ac_proof	VARCHAR	300	Proof of attendance committee
ac_photo	VARCHAR	300	Photo of attendance committee

5. Table Name : tbl_bill_committee

Description : Contain the details of members who are part of bill committee

Primary Key : bc_id

Field	Data Type	Size	Description
bc_id	INT	--	Unique ID of bill committee
bc_name	VARCHAR	50	Name of bill committee
bc_email	VARCHAR	200	Email of bill committee
bc_password	VARCHAR	50	Password of bill committee
bc_proof	VARCHAR	300	Proof of bill committee
bc_photo	VARCHAR	300	Photo of bill committee

6. Table Name : tbl_report_committee

Description : Contain the details of members who are part of report committee

Primary Key : rc_id

Field	Data Type	Size	Description
rc_id	INT	--	Unique ID of report committee
rc_name	VARCHAR	50	Name of report committee
rc_email	VARCHAR	200	Email of report committee
rc_password	VARCHAR	50	Password of report committee
rc_proof	VARCHAR	300	Proof of report committee
rc_photo	VARCHAR	300	Photo of report committee

7. Table Name : tbl_media_committee

Description : Contain the details of members who are part of media committee

Primary Key : mc_id

Field	Data Type	Size	Description
mc_id	INT	--	Unique ID of media committee
mc_name	VARCHAR	50	Name of media committee
mc_email	VARCHAR	200	Email of media committee
mc_password	VARCHAR	50	Password of media committee
mc_proof	VARCHAR	300	Proof of media committee
mc_photo	VARCHAR	300	Photo of media committee

8. Table Name : tbl_volunteers

Description : Contain the details of volunteers

Primary Key : volunteers_id

Field	Data Type	Size	Description
volunteers_id	INT	--	Unique ID of volunteers
volunteers _name	VARCHAR	50	Name of volunteers
volunteers _email	VARCHAR	200	Email of volunteers
volunteers _password	VARCHAR	50	Password of volunteers
volunteers _proof	VARCHAR	300	Proof of volunteers
volunteers _photo	VARCHAR	300	Photo of volunteers

9. Table Name : tbl_event

Description : Contain the details of event

Primary Key : event_id

Field	Data Type	Size	Description
event_id	INT	--	Unique ID of event
event _name	VARCHAR	50	Name of event
event _description	VARCHAR	100	Description of event
event _count	INT	--	Count of event
event_status	INT	--	Status of event
event_date	DATE	--	Date of particular event

10. Table Name : tbl_attendance

Description : Contain the attendance details of particular event

Primary Key : attendance_id

Foreign Key : event_id,volunteers_id

Field	Data Type	Size	Description
attendance_id	INT	--	Unique ID of attendance
event _id	INT	--	Unique ID of event
volunteers_id	INT	--	Unique ID of volunteers
attendance _status	INT	--	Status of attendance

11. Table Name : tbl_bill

Description : Contain the bill details of particular event

Primary Key : bill_id

Foreign Key : event_id,bc_id

Field	Data Type	Size	Description
bill_id	INT	--	Unique ID of bill
event _id	INT	--	Unique ID of event
bc_id	INT	--	Unique ID of bill committee
bill _title	VARCHAR	50	Represent the event title
bill_amount	INT	--	Amount details of particular event
bill_file	VARCHAR	300	Represent the file details
bill_status	INT	--	Status of bill

12. Table Name : tbl_report

Description : Contain the report details of particular event

Primary Key : report_id

Foreign Key : event_id,rc_id

Field	Data Type	Size	Description
report_id	INT	--	Unique ID of report
event_id	INT	--	Unique ID of event
rc_id	INT	--	Unique ID of report committee
report _title	VARCHAR	100	Represent the report title
report_description	VARCHAR	100	Description of report
report_file	VARCHAR	300	Represent the file details
report_status	INT	--	Status of report

13. Table Name : tbl_gallery

Description : Contain the photos of particular event

Primary Key : gallery_id

Foreign Key : event_id

Field	Data Type	Size	Description
gallery_id	INT	--	Unique ID of gallery
event_id	INT	--	Unique ID of event
gallery _title	VARCHAR	100	Represent the gallery title
gallery_file	VARCHAR	300	Represent the file details
gallery_status	INT	--	Status of gallery

14. Table Name : tbl_complaint

Description : Contain the complaint details

Primary Key : complaint_id

Field	Data Type	Size	Description
complaint_id	INT	--	Unique ID of complaint
complaint_title	VARCHAR	50	Represent the complaint title
complaint _content	VARCHAR	100	Represent the content
complaint _reply	VARCHAR	300	Reply to the complaint
complaint_status	INT	--	Status of complaint
Complaint_date	DATE	--	Represent the complaint date

4. SYSTEM TESTING AND IMPLEMENTATION

4.1 SYSTEM TESTING

System testing is a critical phase in software development where the software's actual behaviour is evaluated against expected outcomes. The primary objective is to ensure that no defects are present in the system. To achieve this, testers intentionally run the program to identify potential errors. Although testing cannot guarantee the absolute absence of bugs, the lack of errors during testing can suggest a relatively stable system.

System testing involves identifying and addressing defects in the software. This is a crucial process for any software development team. It serves as the final stage where developers can identify and correct any errors that may have slipped through earlier development phases. The testing process is intentionally designed to uncover any issues that might cause the system to fail. In essence, system testing ensures that software products and their associated documentation meet requirements for completeness, accuracy, reliability, and maintainability.

As the initial phase of implementation, system testing aims to confirm that the system functions correctly and efficiently before going live. Its importance lies in its ability to ensure that each part of the system operates as intended. If all components perform correctly, the system's goals can be achieved. A variety of tests are conducted to verify the system's readiness for user acceptance.

The key steps involved in system testing are:

- Unit testing
- Integration testing
- Validation
- Output testing
- Acceptance testing

System Testing provides the file assurance that software once validated mast combined with all other system elements. System testing verifies whether all elements have been combined properly and that overall system function and performance is achieved. FA theintegration of modules, the validation test was carried out over the system. lt was that all the modules work well together and meet the overall system function and performance.

1. Unit Testing

Unit testing is caried out screen-wise, each screen being identified as an object. Attention is diverted to individual modules, independently to one another to locate errors. This has enabled the detection of errors in coding and logic.

Various test cases are prepared. For each module these test cases are implemented, and it is checked whether the module is executed as per the requirements and outputs the desired result. In this test each service input and output parameters are checked.

In unit testing

- Module interface was tested to ensure that information properly flows into and out of the program under test.
- Boundary condition was tested to ensure that module operates properly at boundaries established to limit or restrict processing.
- All independent paths through the control structures were executed to ensure that all statements in the modules have been executed at least once.
- Error handling paths were also tested.

2. Integration Testing

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing.

Unit tested module were taken and a single program structure was built that has been dictated by the design. Incremental integration has been adopted here.

The modules are tested separately lor accuracy and modules are integrated too.th tn. using bottom-up integration i.e., by integrating from moving from bottom to the top of the system is checked and errors found during integration are rectified. In this testing individual modules were combined and he m0duie wise Shifting was verified to be alright.

The entire software was developed and tested in small segments, where errors were easy to locate and rectify. Program builds (group of modules) were constructed corresponding to the successful testing of user interaction, data manipulation analysis, and display processing and database management.

3. Validation Testing

Validation testing is done to ensure complete assembly of the error-free software. Validation can be termed successful only if it functions in manner. Reasonably expected by the student under validation is alpha and beta testing. The student-side validation is done in this testing phase. It is checked whether the data passed to each student is valid or not. Entering incorrect values does the validation testing and it is checked whether the errors are being considered. Incorrect values are to be discarded. The errors are rectified.

In “NSS Nirmala" verifications are done correctly. So, there is no chance for users to enter incorrect values. It will give error messages by using different validations. The validation testing is done very clearly and found it is error free.

4. Output Testing

After performing the validation testing the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in a specific format.

The output format on the screen was found to be correct as the format was designed in the system design phase according to the user needs. For the hard copy also, the output comes out as specified requirement by the user. Hence output testing does not result in any Correction in the system Output This project is developed based on the user choice. It is user friendly. The output format is very clear to user. Output testing is done on Construction material System correctly.

5. Acceptance Testing

Acceptance involves running a suite of tests on the completed system. Each individual test, known as a Case, exercise particular operating condition of the operating condition of the user's environment or feature of the system, and will result in a pass fail, or Boolean outcome.

4.2 SYSTEM IMPLEMENTATION

The implementation is the final state, and it is an important phase. It involves the invalid programming system testing. user training and the operational running of developed proposed system that constitutes the application subsystems. A major task of preparing for implementation is education of users. which should really have been taken place much carrier in the project when they were belong involved in the investigation and design work. During the implementation phase system take physical shape. In order to develop a system implemented planning is very essential.

The implementation phase of the software development is concerned with translating design specification into source code. The user tests the developed system and changes are made according to their needs. Our system has been successfully implemented.

Before implementation several tests have been conducted to ensure that no errors are encountered during the operation. The implementation phase ends with an evaluation of the system after placing into the operation for a period of time.

The process of putting the developed system in actual use is called system implementation. This includes all those activities that take place to convert from old system to new system. The system can be implemented only after testing is done and is found to be working to specifications. The implementation stage is a systems project in its own right.

The implementation stage involves following tasks:

- Careful planning.

- Investigation of system and constraints.
- Design of method to achieve change over.
- Evaluation of the changeover method.

In the case of this project all the screens are designed first. For making it to be executable, codes are written on each screen and performs the implementation by creating the database and connecting to the server. After that the system, is Checked, whether it performs all the transactions correctly. Then databases are cleared and made it to be usable to the technicians.

Implementation Plans

The following are the step involved in the implementation plan of " NSS Nirmala":

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

5. SECURITY TECHNOLOGIES & POLICIES

The protection of computer-based resources that includes hardware, software, data procedures and people against unauthorized use or natural disaster is known as System Security.

System Security can be divided into four related issues:

- Security
- Integrity
- Privacy
- Confidentiality

SYSTEM SECURITY refers to the technical innovations and procedures applied to the hardware and operation systems to protect against deliberate or accidental damage from a define threat

DATA SECURITY is the protection of data from loss, disclosure, modification and destruction.

SYSTEM INTEGRITY refers to the power functioning of hardware and programs, appropriate physical security and safety against external threats such as eavesdropping and wiretapping.

PRIVACY defines the rights of the user or organizations to determine what information they are willing to share with or accept from others and how the organization can be protected against unwelcome, unfair or excessive dissemination of information about it.

CONFIDENTIALITY is a special status given to sensitive information in a database to minimize the possible invasion of privacy. lt is an attribute of information that characterizes its need for protection.

SECURITY IN SOFTWARE System security refers to various validations on data in form of checks and controls to avoid the system from failing. It is always important to ensure that only valid data is entered, and only valid operations are performed on the system.

The system employees two types check and controls:

CLIENT-SIDE VALIDATION Various client-side validations are used to ensure on the client side that only valid data is entered. Client-side validation saves server time and load to handle invalid data. Some checks imposed are:

- Forms cannot be submitted without filling up the mandatory data so that manual mistakes of submitting empty fields that are mandatory can be sorted out at the client side to save the server time and load.
- Tab-indexes are set according to the need and considering the ease of user while working with the system.

SERVER-SIDE VALIDATION Some checks cannot be applied at client side. Server-side checks are necessary to save the system from failing and intimating the user that some invalid operation has been performed or the performed operation is restricted. Some of the server-side checks imposed is:

- Server-side constraint has been imposed to check for the validity of primary key and foreign key. A primary key value cannot be duplicated. Any attempt to duplicate the primary value results into a message intimating the user about those values through the forms using foreign key can be updated only of the existing foreign key values.
- Various Access Control Mechanisms have been built so that one user may not agitate upon another. Access permissions to various types of users are controlled according to the organizational structure. Only permitted users can log on to the system and can have access according to their category. User- name, passwords and permissions are controlled the server side.
- User is intimating through appropriate messages about the successful operations or exceptions occurring at server side.
- Using server-side validation, constraints on several restricted operations are imposed.

6.MAINTENANCE

Software maintenance is the modification of a software product after delivery to correct faults, to improve performance or other attributes. Maintenance is the ease with which a program can be corrected if any error is encountered, adapted if its environment changes, or enhanced if the customer desires a change in requirements. Maintenance follows conversations to extend that changes are necessary to maintain satisfactory operations relative to changes in the user's environment.

Maintenance often includes minor enhancements or corrections to problems that surface in the system's operation. Maintenance is also done based on fixing the problems reported, changing the interface with other software or hardware, enhancing the software.

CATEGORIES OF MAINTENANCE

a. Corrective Maintenance

Corrective maintenance is the most commonly used maintenance approach, but it is easy to see its limitations. When equipment fails, it often leads to downtime in production and sometimes damages other parts. In most cases, this is expensive. Also, if the equipment needs to be replaced, the cost of replacing it alone can be substantial. The reliability of systems maintained by this type of maintenance is unknown and cannot be measured. Corrective maintenance is possible since the consequences of failure or wearing out are not significant and the cost of this maintenance is not great.

b. Perfective Maintenance

Modification of a software product after delivery to improve performance or maintainability. This term is used to describe changes undertaken to expand the existing requirements of the system. A successful piece of software tends to be subjected to a succession of changes resulting in an increase in its requirements. This is based on the premise that as the software becomes useful, the user experiment with new cases beyond the scope for which it was initially developed. Expansion in requirements can take the form of enhancement of existing system functionality and improvement in computational efficiency.

c. Adaptive Maintenance

Modification of a software product performed after delivery to keep a product usable in a changed or changing environment. Adaptive maintenance includes any work initiated as a consequence of moving the software to a different hardware or software platform. It is a change driven by the need to accommodate modifications in the environment of the software system. The environment in this context refers to the totality of all conditions and influences which act from outside upon the system. A change to the whole or part of this environment will warrant a corresponding modification of the software.

d. Preventive Maintenance

Preventive maintenance is a schedule of planned maintenance actions aimed at the prevention of breakdowns and failures. The primary goal of preventive maintenance is to prevent the failure of equipment before it actually occurs. It is designed to preserve and enhance equipment reliability by replacing worn components before they actually fail. Preventive maintenance activities include equipment checks, partial or complete overhauls at specified periods.

Long-term benefits of preventive maintenance include:

- Improved system reliability.
- Decreased cost of replacement.
- Decreased system downtime.

7. SCOPE FOR FUTURE ENHANCEMENT

The scope for future enhancement of the NSS Website Management System is vast, offering opportunities to further improve efficiency, user experience, and functionality. A mobile application could be developed to enable users to manage tasks and view updates on-the-go, increasing accessibility for volunteers and committee members. Advanced reporting and analytics could be integrated to track trends in volunteer participation, event success, and budget management, providing valuable insights for better decision-making. The system could also be enhanced by integrating with social media platforms for event promotion and media sharing, boosting visibility and engagement. A feedback system could be introduced to allow participants to rate and provide input on events, helping committees improve future activities. Additionally, security features like multi-factor authentication and encryption could be strengthened to protect sensitive data. Automated communication tools would streamline task assignments and event reminders, while customizable dashboards would offer users personalized views of their responsibilities. The system could also expand to include event sponsorship and fundraising modules, real-time collaboration tools, and third-party integrations with platforms like Google Drive or project management software. Volunteer recognition through certification and rewards could be added to motivate participants, and the system could be scaled to accommodate larger NSS units or other organizations, ensuring it remains adaptable to future growth and changing needs.

8. CONCLUSION

Through various initiatives, NSS encourages participants to address local issues, promote social welfare, and develop skills while making a positive impact on society. This project not only enhances personal growth but also strengthens community bonds, empowering youth to be proactive change-makers. This System is an online platform developed to facilitate the coordination and management of NSS events and activities. The NSS Nirmala Website Management System provides an efficient way for committees to collaborate and manage NSS events. It offers a platform where each role can fulfill its duties, while the Programme Officer can oversee the entire process. This system streamlines communication, automates task tracking, and ensures that every event is well-organized and documented.

9. BIBLIOGRAPHY

1. Jalotte, Pankaj. " SOFTWARE ENGINEERING", Third Edition, Spring Publishers, Narosa Publishing House.
2. Ramakrishnan, Raghu. and_ Gehrke, Johannes. DATABASE MANAGEMENT SYSTEM", Third Edition, McGraw-Hill Professional Publication.
3. Awad, Elias M. "SYSTEM ANALYSIS AND DESIGN", Second Edition, McGrawProfessional Hill publication.
4. Rob, Peter. and Coronel, Carlos. "DATABASE SYSTEM DESIGN AND MANAGEMENT", Fifth Edition, Thomson Learning Publication.
5. Pressman, Roger S. "SOFTWARE ENGINEERING", Fifth Edition, McGraw-Hill Publication.
6. Bayross, Ivan. "SQL/MySQL" Second Revised Edition, BPB Publications
7. Rasmus Leodorf and Kevin Tatore. "PROGRAMMING PHP "

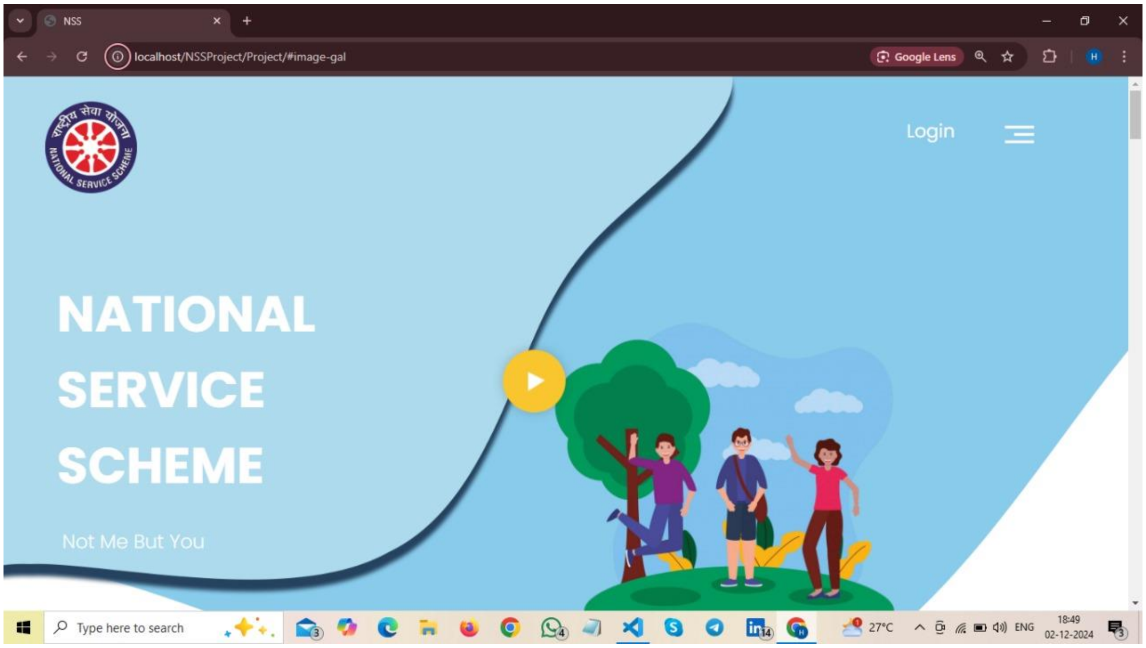
REFERENCE

1. Modulue template downloaded from: https://www.free-css.com
2. Modulue template downloaded from: https://www.w3schools.com
3. NSS Nirmala Systems details researched from: https://nss.mgu.ac.in

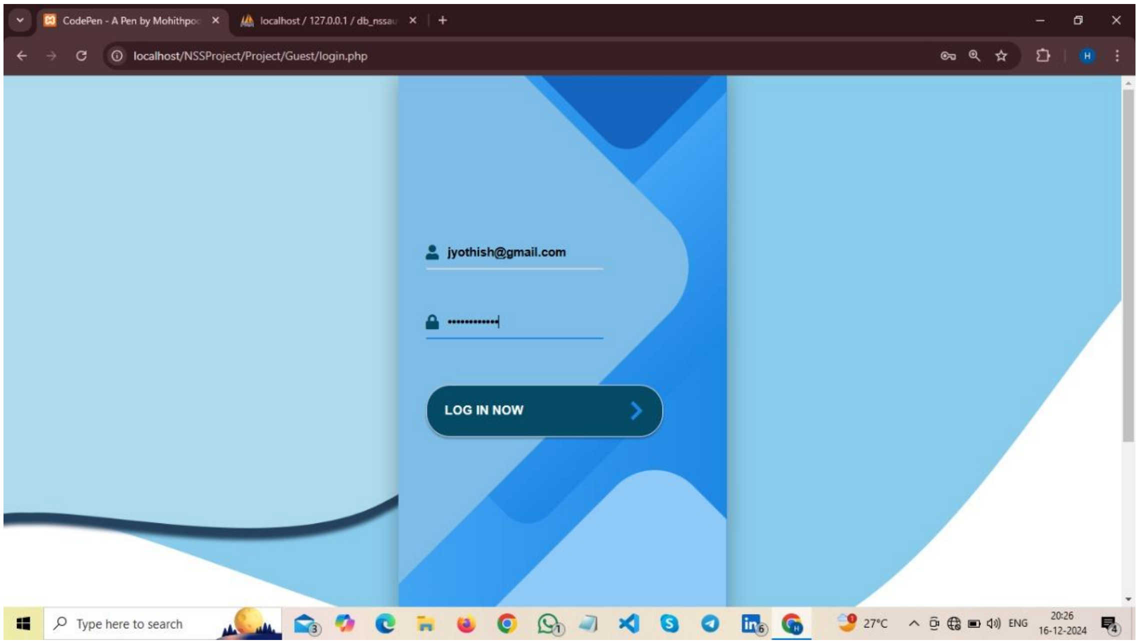
10. APPENDIX

10.1 SCREENSHOTS

HOME PAGE



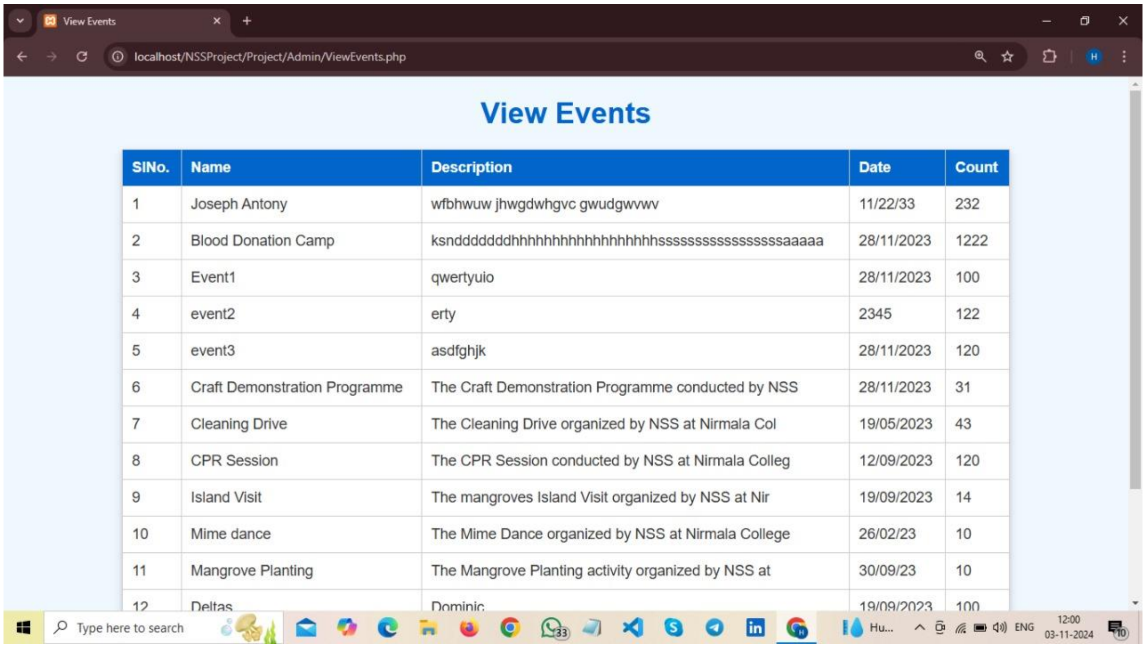
LOGIN PAGE


CodePen - A Pen by Mohithpo
localhost / 127.0.0.1 / db_nssa
localhost/NSSProject/Project/Guest/login.php
LOG IN NOW
Type here to search
27°C
ENG
20:26
16-12-2024

jyothish@gmail.com

••••••••••••

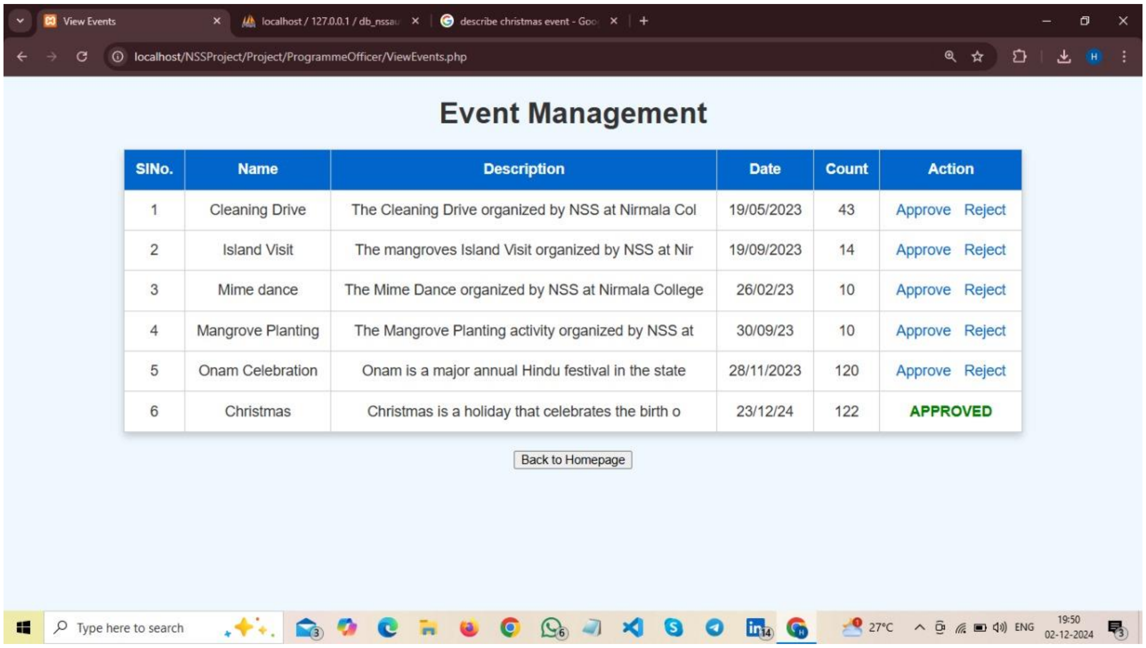
ADMIN DASHBOARD



View Events

SINo.	Name	Description	Date	Count
1	Joseph Antony	wfbhwuw jhwgdwhgvc gwudgwvwv	11/22/33	232
2	Blood Donation Camp	ksndddddddhhhhhhhhhhhhhhhhhsssssssssssssssssaaaaa	28/11/2023	1222
3	Event1	qwertyuio	28/11/2023	100
4	event2	erty	2345	122
5	event3	asdfghjk	28/11/2023	120
6	Craft Demonstration Programme	The Craft Demonstration Programme conducted by NSS	28/11/2023	31
7	Cleaning Drive	The Cleaning Drive organized by NSS at Nirmala Col	19/05/2023	43
8	CPR Session	The CPR Session conducted by NSS at Nirmala Colleg	12/09/2023	120
9	Island Visit	The mangroves Island Visit organized by NSS at Nir	19/09/2023	14
10	Mime dance	The Mime Dance organized by NSS at Nirmala College	26/02/23	10
11	Mangrove Planting	The Mangrove Planting activity organized by NSS at	30/09/23	10
12	Deltas	Dominic	19/09/2023	100

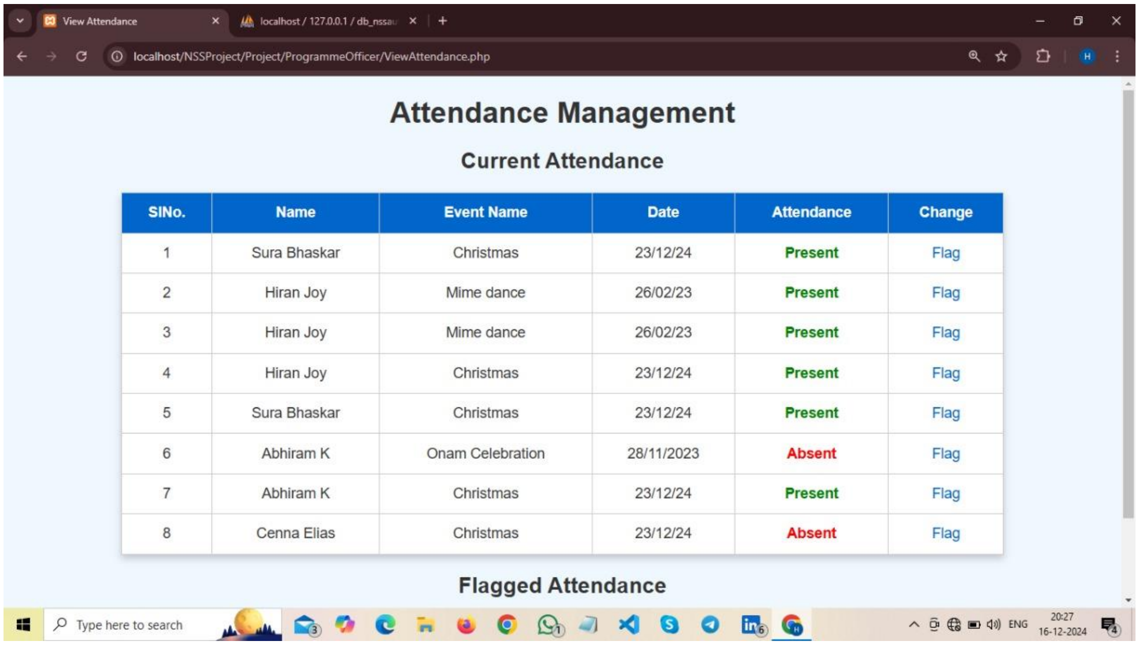
PROGRAMME OFFICER DASHBOARD


View Events
localhost / 127.0.0.1 / db_nssau
describe christmas event - Goo
localhost/NSSProject/Project/ProgrammeOfficer/ViewEvents.php
Type here to search
27°C
ENG
19:50
02-12-2024

Event Management

SINo.	Name	Description	Date	Count	Action
1	Cleaning Drive	The Cleaning Drive organized by NSS at Nirmala Col	19/05/2023	43	Approve Reject
2	Island Visit	The mangroves Island Visit organized by NSS at Nir	19/09/2023	14	Approve Reject
3	Mime dance	The Mime Dance organized by NSS at Nirmala College	26/02/23	10	Approve Reject
4	Mangrove Planting	The Mangrove Planting activity organized by NSS at	30/09/23	10	Approve Reject
5	Onam Celebration	Onam is a major annual Hindu festival in the state	28/11/2023	120	Approve Reject
6	Christmas	Christmas is a holiday that celebrates the birth o	23/12/24	122	APPROVED

Back to Homepage


View Attendance
localhost / 127.0.0.1 / db_nssau
localhost/NSSProject/Project/ProgrammeOfficer/ViewAttendance.php
Type here to search
ENG
20:27
16-12-2024

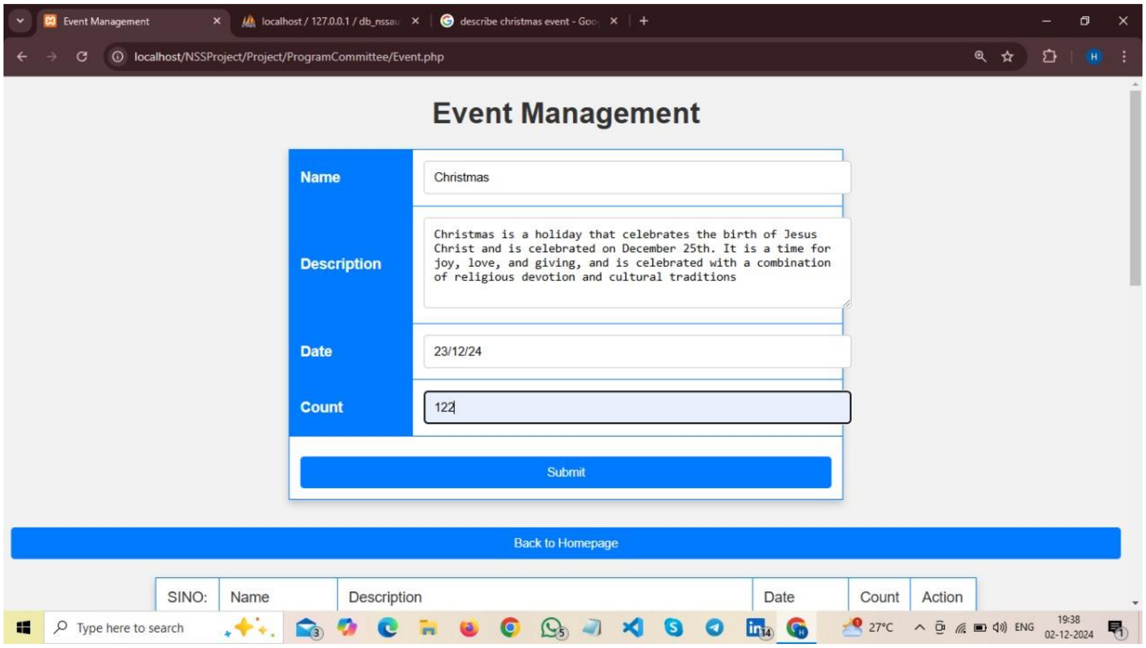
Attendance Management

Current Attendance

SlNo.	Name	Event Name	Date	Attendance	Change
1	Sura Bhaskar	Christmas	23/12/24	Present	Flag
2	Hiran Joy	Mime dance	26/02/23	Present	Flag
3	Hiran Joy	Mime dance	26/02/23	Present	Flag
4	Hiran Joy	Christmas	23/12/24	Present	Flag
5	Sura Bhaskar	Christmas	23/12/24	Present	Flag
6	Abhiram K	Onam Celebration	28/11/2023	Absent	Flag
7	Abhiram K	Christmas	23/12/24	Present	Flag
8	Cenna Elias	Christmas	23/12/24	Absent	Flag

Flagged Attendance

ADD EVENT

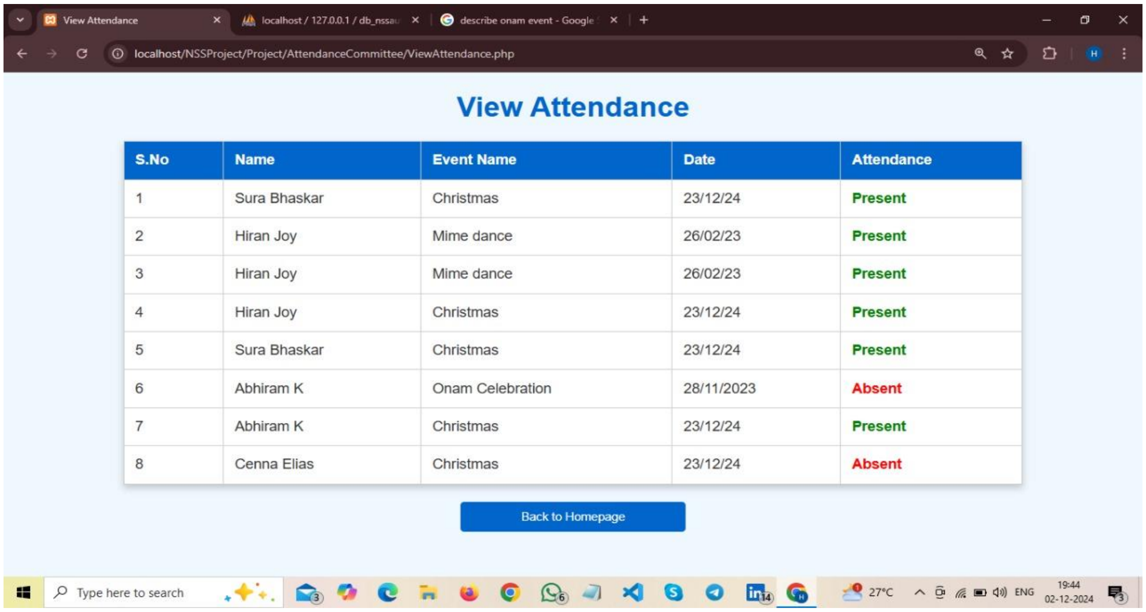

Event Management
localhost / 127.0.0.1 / db_nssau
describe christmas event - Goo
localhost/NSSProject/Project/ProgramCommittee/Event.php
Submit
Back to Homepage
Type here to search
27°C
ENG
19:38
02-12-2024

Event Management

Name	Christmas
Description	Christmas is a holiday that celebrates the birth of Jesus Christ and is celebrated on December 25th. It is a time for joy, love, and giving, and is celebrated with a combination of religious devotion and cultural traditions
Date	23/12/24
Count	122

SINO:	Name	Description	Date	Count	Action

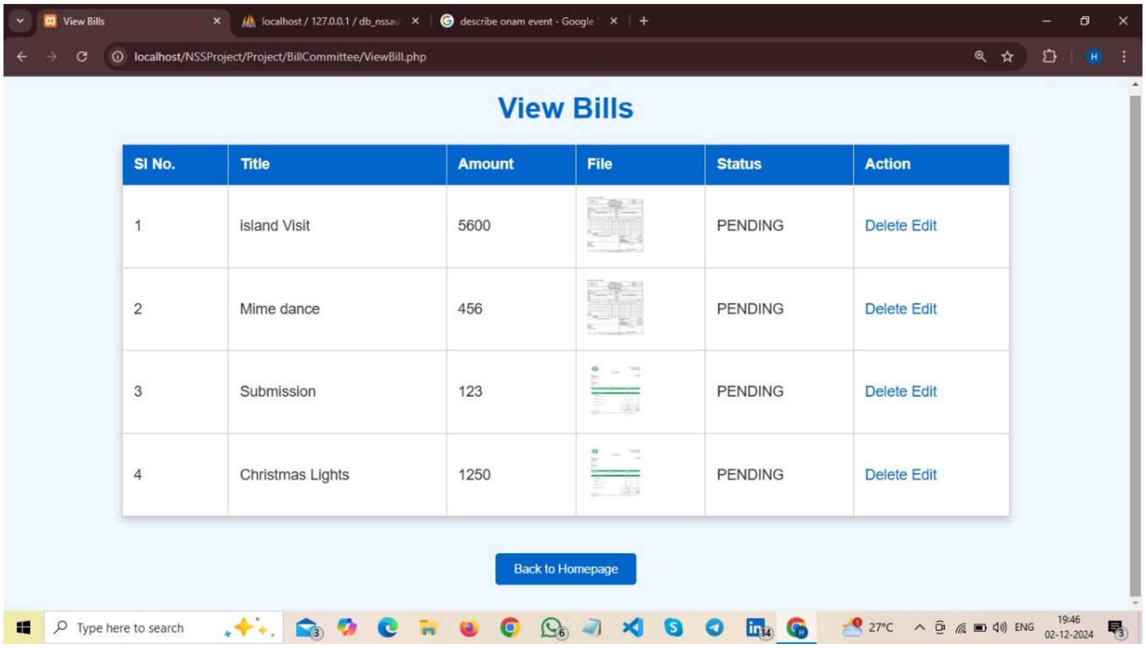
ADD ATTENDANCE


View Attendance
localhost / 127.0.0.1 / db_nssau
describe onam event - Google
localhost/NSSProject/Project/AttendanceCommittee/ViewAttendance.php
Back to Homepage
Type here to search
27°C
ENG
19:44
02-12-2024

View Attendance

S.No	Name	Event Name	Date	Attendance
1	Sura Bhaskar	Christmas	23/12/24	Present
2	Hiran Joy	Mime dance	26/02/23	Present
3	Hiran Joy	Mime dance	26/02/23	Present
4	Hiran Joy	Christmas	23/12/24	Present
5	Sura Bhaskar	Christmas	23/12/24	Present
6	Abhiram K	Onam Celebration	28/11/2023	Absent
7	Abhiram K	Christmas	23/12/24	Present
8	Cenna Elias	Christmas	23/12/24	Absent

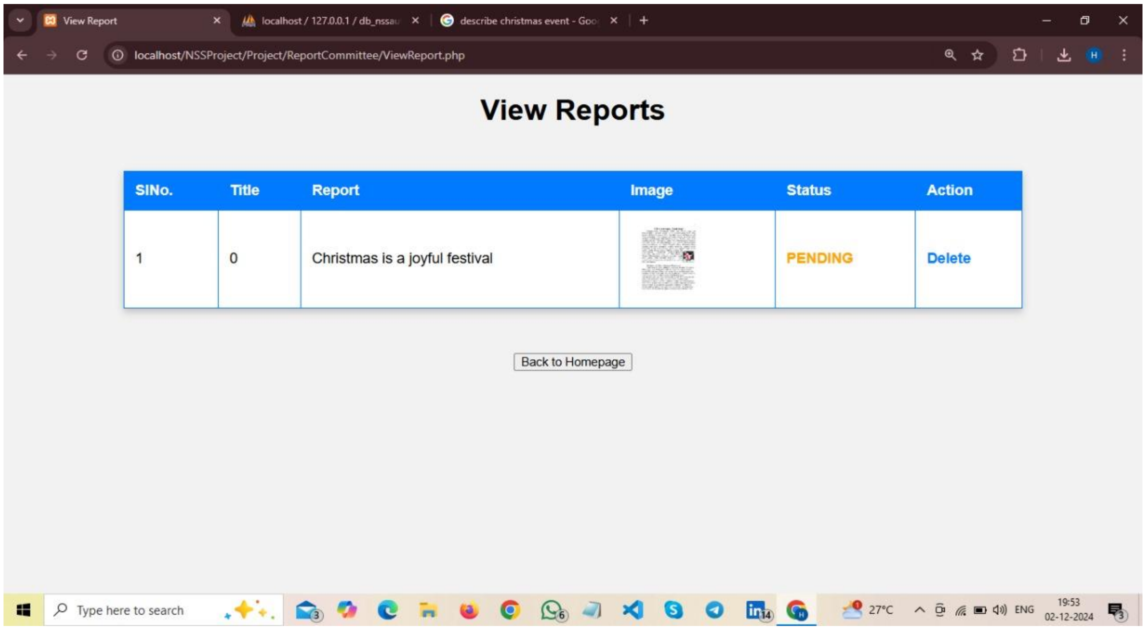
ADD BILL


View Bills
localhost / 127.0.0.1 / db_nssau
describe onam event - Google
localhost/NSSProject/Project/BillCommittee/ViewBill.php
Back to Homepage
Type here to search
27°C
ENG
19:46
02-12-2024

View Bills

Sl No.	Title	Amount	File	Status	Action
1	island Visit	5600		PENDING	Delete Edit
2	Mime dance	456		PENDING	Delete Edit
3	Submission	123		PENDING	Delete Edit
4	Christmas Lights	1250		PENDING	Delete Edit

ADD REPORT

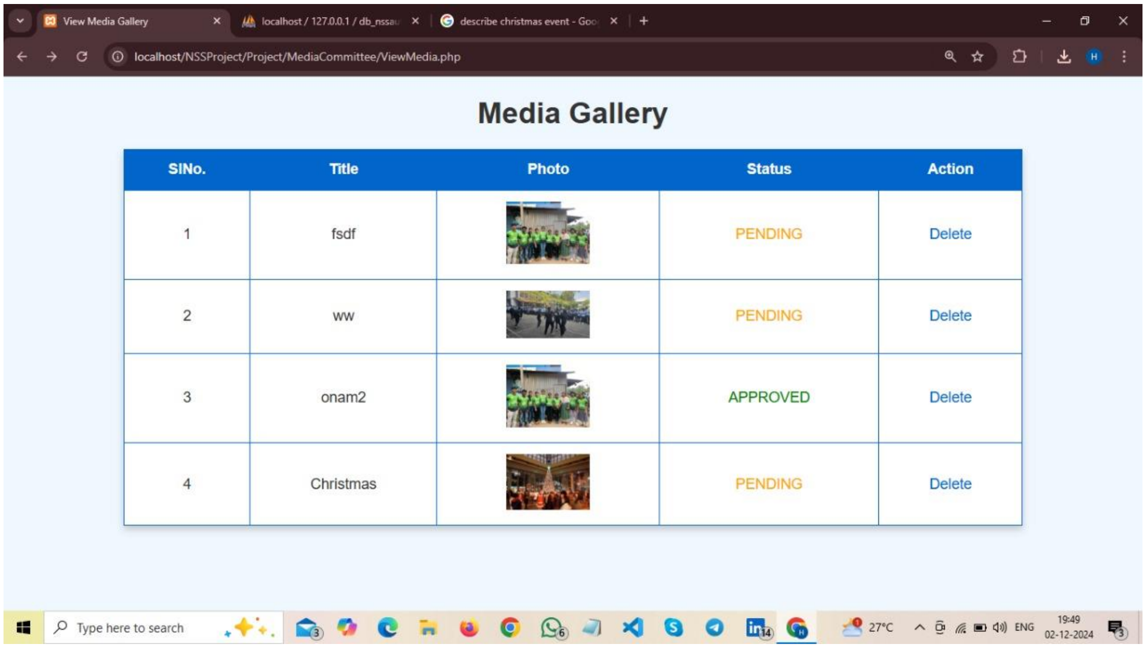

View Report
localhost / 127.0.0.1 / db_nssa
describe christmas event - Goo
localhost/NSSProject/Project/ReportCommittee/ViewReport.php
Type here to search
27°C
ENG
19:53
02-12-2024

View Reports

SlNo.	Title	Report	Image	Status	Action
1	0	Christmas is a joyful festival		PENDING	Delete

Back to Homepage

ADD MEDIA


View Media Gallery
localhost / 127.0.0.1 / db_nssa
describe christmas event - Goo
localhost/NSSProject/Project/MediaCommittee/ViewMedia.php
Type here to search
27°C
ENG
19:49
02-12-2024

Media Gallery

SlNo.	Title	Photo	Status	Action
1	fsdf		PENDING	Delete
2	ww		PENDING	Delete
3	onam2		APPROVED	Delete
4	Christmas		PENDING	Delete