**Regional College**

Of Professional Studies & Research, Bareilly

Affiliated to Mahatma Jyotiba Phule Rohilkhand university (Bareilly),

Bareilly-243001

Major Project Report On

**“Online Examination System”**



SESSION: 2024-2025

BACHELOR OF COMPUTER APPLICATION

**Under the Guidance of:                Submitted By:**

**Mrs. Nidhi Agarwal** **Rahul Maurya (230031160029)**

**Mrs. Priyanka Pradhan**            **Mohd Tanzeem(230031160024)**

**Rahul Kumar (230031160028)**

**(BCA VI SEMESTER)**

**ACKNOWLEDGEMENT**

I Would like to take this opportunity to express my sincere gratitude towards my supervisor, **Mrs. Nidhi Agarwal** And **Mrs. Priyanka Pradhan** for his invaluable guidance during the progress from the beginning to the completion of the project. I also wish to thank all the faculty member of MJPRU Bareilly who gave my valuable knowledge through their lectures. At this place, I have got plenty of fundamental and advanced knowledge for my career. I am indebted my mother, My father and friends for all the encouragement and moral support…

**Submitted By:**

**Rahul Maurya (230031160029)**

**Mohd Tanzeem (230031160024)**

**Rahul Kumar (230031160028)**

2

**Candidate’s Declaration**

I declare that the work presented in this Project report entitled, **“Online Examination System”** in partial fulfillment for the 6th semester of Degree of “Bachelors of Computer Application” in Computer science with Specialization in Web Technology & PHP (PhpMyAdmin) in Backend & submitted in Department of computer science, Mahatma Jyotiba Phule Rohilkhand University Bareilly, is a record of my own investigations carried under the guidance of **Mrs. Nidhi Agarwal** And **Mrs.Priyanka Pradhan**, (Department Of Computer Science)…

I have not Submitted the matter presented in this Report anywhere for the award of any other Degree

**Submitted By:**

**Rahul Maurya (230031160029)**

**Mohd Tanzeem (230031160024)**

**Rahul Kumar (230031160028)**

3

**Introduction**

The Online Examination System is a web-based application developed to facilitate the process of conducting examinations online. It aims to replace the traditional pen-and-paper testing system with a more efficient, secure, and accessible digital platform. This system enables students to take subject-wise quizzes from anywhere, while administrators can manage users, questions, and results seamlessly.

The system supports multiple programming subjects such as C, C++, Java, Python, Node.js, and React.js, each with its own dynamic quiz set. Users can register, log in, take quizzes, view their performance, and download certificates upon completion. The administrator panel allows for effective management of users, exams, and feedback, providing full control over the examination process.

**The system also integrates advanced features like:**

* Real-time quiz evaluation
* AI-based chatbot assistance
* Dynamic result generation with PDF certificates
* Authentication With Email
* Responsive Bootstrap UI for better user experience

This project is especially useful for educational institutions and e-learning platforms that aim to digitize their examination process while ensuring security, scalability, and ease of access.

4

**System Specification**

System Specification refers to the detailed description of the hardware and software requirements needed to develop and run the application smoothly. It outlines what kind of physical devices (hardware) and programs (software) are required for both the development team and end-users. In our Online Examination System, we have used basic web technologies such as HTML, CSS, JavaScript, PHP, and MySQL which work efficiently on a standard computer setup.

This system is designed to be lightweight, responsive, and compatible with modern browsers and devices, making it suitable for both desktop and mobile platforms. Open-source tools like XAMPP have been used to reduce development cost and ensure easy deployment.

**Purpose of System Specification:**

The purpose of this section is to outline the hardware and software requirements necessary to develop, deploy, and operate the Online Examination System efficiently

5

**1.Hardware Requirements:**

This system requires minimal hardware resources for proper operation. It can be easily installed and run in standard computer labs.

* **Minimum Hardware Requirements:**
* **Processor:** Intel Core i3 or equivalent
* **RAM:** 2 GB
* **Hard Disk:** 100 MB
* **Display:** 1024x768
* **Input Devices:** Keyboard and Mouse
* **Internet Connection:** (for email/OTP or API based features)
* **Recommended Hardware Requirements:**
  + **Processor:** Intel Core i3 or higher
  + **RAM:** 4 GB
  + **Hard Disk:** 256 GB SSD
  + **Monitor:** Full HD display

**2. System Requirements:**

* **Operating System:** windows
* **Frontend:** HTML, CSS, Bootstrap, JavaScript
* **Backend:** PHP 7.x or later
* **Database:** MySQL
* **Web Server:** XAMPP (Apache + MySQL + PHP)

6

* **Browser:** Google Chrome
* **IDE/Editor:** VS Code
* **PDF Generator:** if certificate generation is included

**3.Programming Languages Used:**

* **PHP:** (for backend logic and database interaction)
* **HTML/CSS/Bootstrap:** (for UI structure and styling)
* **JavaScript:** (for client-side interactivity)
* **SQL:** (for database queries)

**4.System Architecture:**

You can also include a 2-tier or 3-tier architecture diagram showing:

* **Client Layer:** (Browser)
* **Application Layer:** (PHP scripts)
* **Database Layer:** (MySQL)

**5.Responsiveness and Compatibility:**

* Mobile-friendly design
* Ability to work across various browsers

7

**6.Deployment Environment:**

The Online Examination System has been deployed on **Infinity Free**, a free web hosting platform that supports PHP and MySQL. This platform allows the system to be accessed online from any location using a standard web browser, ensuring greater accessibility and flexibility.

* **The use of InfinityFree for deployment offers the following advantages:**
* Free hosting with support for PHP 7.x and MySQL databases
* Up to 5 GB of disk space and unlimited bandwidth
* Web-based control panel for file and database management
* Ability to test the system in a real-world online environment

**Hosting Platform Used:** InfinityFree (<https://infinityfree.net>) **Hosting Type:** Shared Free Hosting **Database Used:** MySQL (phpMyAdmin via InfinityFree cPanel)  
**Supported Technologies:** PHP, HTML, CSS, JavaScript, MySQL

8

**DBMS**

PhpMyAdmin is a free, open-source web-based application written in PHP that is used to manage MySQL or MariaDB databases using a graphical user interface (GUI).  
Instead of writing complex SQL commands in the terminal or command line, phpMyAdmin allows users to perform all kinds of database operations with just a few clicks through a web browser.

It is especially useful for beginners, web developers, students, and even experienced programmers who want to manage their databases quickly and visually.

**Key Features of phpMyAdmin**

**1. Database Management**

* Create, rename, or delete entire databases with ease.
* View all tables inside a database.
* Perform operations like copying or exporting an entire database.

**2. Table Management**

* Create, edit, or delete tables inside a database.
* Modify table structure by adding/removing columns.
* Change data types, default values, indexes, and primary keys.

9

* Sort or filter data easily without writing queries.

**3. Run SQL Queries**

* Execute any custom SQL commands directly.
* View results instantly in a clean tabular format.
* Great for testing SELECT, JOIN, and other queries.
* Supports multiple query executions at once.

**4. Data Operations**

* Browse table data row by row.
* Edit values directly in the browser.
* Delete, insert, or update records easily.
* Search for specific values in tables using filters.

**5. User and Privilege Management**

* Create new MySQL users.
* Assign privileges (read, write, delete, etc.) to users per database or table.
* Change or reset passwords secusrely.

10

**System Development Life Cycle**

**1. Requirement Analysis**

**Requirement Analysis** is the process of identifying, gathering, analyzing, and documenting the needs and expectations of users and stakeholders for a proposed system. It is the first and one of the most crucial phases in the Software Development Life Cycle (SDLC). During this phase, all the functional and non-functional requirements of the system are clearly defined to ensure that the final product fulfills the user’s needs and business objectives.

This phase involves regular communication with users, interviews, questionnaires, and documentation reviews to understand what the system should do and how it should behave. The output of this phase is usually a **Software Requirements Specification (SRS)** document that serves as a guideline for the next stages like design and development.

In simple words, requirement analysis lays the **foundation of the entire project**, and any mistake in this stage can lead to project failure or costly modifications later.

**2. Feasibility Study**

**Feasibility Study** is a detailed analysis performed in the early stages of a project to determine whether the proposed system is **practically achievable** and **worth developing**. It involves evaluating the technical, economic, operational, legal, and time-related aspects of

11

the project to ensure that the idea can be turned into a successful

and sustainable system.

The main objective of a feasibility study is to **identify potential risks**, estimate costs and resources, and ensure that the project aligns with business goals. It helps decision-makers decide whether to proceed with the project, modify it, or abandon it altogether.

In this phase, developers and project managers assess:

* Can we build the system with available technology? (Technical feasibility)
* Will it be cost-effective and within budget? (Economic feasibility)
* Will users be able to operate and accept the system easily? (Operational feasibility)
* Can it be completed within the given timeline? (Schedule feasibility)

A feasibility study is critical because it prevents **unnecessary investment of time, money, and resources** in projects that are likely to fail.

**3. System Design**

**System Design** refers to the process of defining the architecture of a system, its components, modules, interfaces, and data flow. It is a critical step in the software development life cycle (SDLC) that

12

bridges the gap between the system’s **requirements** (what the system should do) and its **implementation** (how the system will perform its tasks). System Design is aimed at transforming **functional and non-functional requirements** into a workable model of the system.

**In Simple Words:**

Imagine you are building a **house**. Before you start construction, you need a **blueprint or a plan**. This blueprint will show you:

* The **structure** of the house
* How the rooms will be arranged
* Where the doors and windows will be
* How the electricity and plumbing will work

Similarly, in **System Design**, you are creating a **blueprint** for how your system will work. You decide:

* What **modules** (or parts) the system will have
* How these parts will interact with each other
* How data will flow within the system
* What technologies and structures will be used

**4. Implementation**

**Implementation** is the phase in the **System Development Life Cycle (SDLC)** where the actual system is built and made functional.

13

it involves translating the system design into **working software** by

writing code, integrating system components, and ensuring that the system performs as intended. In this phase, developers follow the system design specifications to create the functional components, connect different modules, and perform **initial testing** to verify that everything works as expected.

The implementation phase is crucial as it directly transforms theoretical plans into a **real, operational system**. It includes coding, setting up the development environment, creating databases, writing documentation, and resolving any issues through debugging. The success of the implementation phase determines the quality and effectiveness of the final system.

**Key Points:**

1. **Coding**: Developers write the actual code to build the system's functionalities.
2. **System Integration**: Different modules and components are connected to work together as a unified system.

**5. Testing**

**Testing** is the process of evaluating and verifying that a system or component functions as expected and meets the defined requirements. It involves executing the system or software to identify any bugs, errors, or issues, ensuring that the system is **reliable, secure, and performs** as intended.

14

The main goal of testing is to find and fix any defects before the system is deployed for end-users. It ensures that the system operates correctly in different environments and under various conditions. Testing can include **unit testing**, **integration testing**, **system testing**, and **user acceptance testing** (UAT), depending on the stage of development.

**6. Deployment**

**Deployment** is the process of installing, configuring, and making a system or application **operational** in a live environment for end-users. It involves transferring the software from a development or testing environment to a production environment where it can be accessed and used.

Deployment includes tasks such as **system setup**, **database configuration**, **server setup**, and ensuring that all components work correctly in the live environment. Once deployed, the system is available for use, and users can interact with it, starting the process of **monitoring** and **maintenance** to ensure smooth operation.

**7. Maintenance**

The **Maintenance Phase** in SDLC refers to the process of **modifying, updating, and improving a software application** after it has been delivered to the end users. This includes **fixing bugs**, **enhancing features**, **ensuring compatibility**, and **updating the system** to adapt to changes in business needs, technology, or user expectations.

15

Even though it’s the last stage, **maintenance is continuous** throughout the software’s life. It ensures the system remains **reliable**, **secure**, and **efficient** over time.

**Key Activities in Maintenance Phase:**

1. **Corrective Maintenance** – Fixing bugs or errors reported by users.

2. **Adaptive Maintenance** – Updating the system to work with new technologies, OS, or hardware.

3. **Perfective Maintenance** – Improving system performance and adding new features based on feedback.

4. **Preventive Maintenance** – Making changes to prevent future issues or system breakdowns.

5. **Monitoring & Evaluation** – Regularly checking system performance, usage, and user satisfaction.

6. **Security Updates** – Applying patches to fix vulnerabilities and protect user data.

7. **Documentation Update** – Keeping technical and user documents up to date with all changes.

16