Literature Review and Requirements specification for Existing solutions, tools and Equipment

# Introduction

In this documentation it shall provide a brief overview for our prototype that is using PHP, HTML, JavaScript, MySQL, Google Authenticator and SMFTP. These will be mainly used for the authentication, data management and email communication. This system aims to provide secure user authentication, database storage as well as email notifications using best practices.

# 1. Literature Review

# Overview of Similar Existing Solutions

Several pre-existing and existing web applications use a combination of PHP, JavaScript, and MySQL for backend and fronted development. Additionally, Google Authenticator is a widely accepted MFA this stands for (Multi-Factor Authentication) solution, while SMFTP ensures purely just email communication.

# Similar Systems and Tools

1. Content Management Systems (CMS):

* WordPress: This Uses PHP and MySQL for database management and integrates the use of Google Authenticator this is used for the MFA and supports SMTP for emails.
* Joomla: This Provides authentication plugins and secure email handling via the use of SMTP.
* Drupal: Uses PHP and JavaScript with MySQL, supporting modular authentication.

2. Authentication Platforms:

* Google Authenticator: This provides TOTP-based MFA for login security.
* Auth0: this is a cloud-based authentication service that support the use of MFA and secure logins.

3. Email Communication Services:

* SMTP this stands for (Simple Mail Transfer Protocol): This is used for sending verification and password resets emails too the users / clients.
* SendGrid & Mailgun: This is a third-party service for handling automated email transactions.

# Role of the Selected Technologies

|  |  |
| --- | --- |
| Technology | Purpose |
| PHP | Used for Backend processing and server-side logic |
| HTML & JavaScript | Used for Frontend UI/UX and client-side scripting |
| MySQL | Used for Database management for storing user details |
| Google Authenticator | Used for Multi-factor Authentication (MFA) |
| SMFTP | Used for Securing email notifications such as password resets |

# 2. Requirement Specification

# 2.1 Functional Requirements

1. User Registration & Login

* Users can register with an email and password.
* Google Authenticator generates a TOTP for MFA.
* Login requires email, password, and an OTP for authentication.

2. Database management (MySQL)

* Store user credentials (hashed passwords, authentication logs.
* Also allows the use of Tracking of login attempts for security.

3. Secure Email Communication (SMFTP)

* Send’s the account activation and passwords reset emails securely.
* This also provides the email logs in the admin panel.

4. Role-Based Access Control (RBAC)

* This is primarily the Admin, User and the Moderator roles with different privileges.

## Non-Functional Requirements

1. Security

* An SSL encryption for login and data transmission.
* Will allow us to Secure password hashing (e.g., bcrypt)
* Two-factor Authentication (2FA) this will be mainly used using the Google Authenticator app.

1. Performance

* Asynchronous JavaScript (AJAX) to improve UI responsiveness.
* Optimized MySQL queries for fast data retrieval.

1. Scalability

* This will Support multiple users concurrently without lag.

## System Architecture

The system follows a three-tier architecture:

1. Fronted (HTML, JavaScript, CSS) – This is the user Interface.
2. Backend (PHP, MySQL) – This is mainly made up of the Business logic, authentication, and database.

External APIs (Google Authenticator, SMFTP) – These will be used for the Security and communication purposes.

## 3. Interim Prototype Implementation

# Development Environment

|  |  |
| --- | --- |
| Component | Technology |
| Web Server | Apache or Nginx |
| Database | MySQL |
| Programming languages | PHP, HTML, JavaScript |
| Security APIs | Google Authenticator API |
| Email API | SMFTP |

# Prototype Features

* User Registration and Login
* Database Setup (MySQL Schema for Users, Roles, and Logs)
* Google Authenticator integration for MFA
* Email Sending Via SMFTP for password resets.
* Basic Admin Panel for Managing Users.