**Dashboard for National Accounts**

**Software Requirements Specification**

**INT 219 & INT 220**

**Frontend and Backend**

**Shudhanshu Kumar Singh - 12320289**

**Vikash Gupta - 12321380**

**Hemesh Raj - 12325006  
Aayush Shah - 12314896**

Prepared for

Continuous Assessment 2

Spring 2025

**Table of Contents**

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Scope 1

1.3 Definitions, Acronyms, and Abbreviations 1

1.4 References 1

1.5 Overview 1

2. General Description 2

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Characteristics 2

2.4 General Constraints 2

2.5 Assumptions and Dependencies 2

3. Specific Requirements 2

3.1 External Interface Requirements 3

3.1.1 User Interfaces 3

3.1.2 Hardware Interfaces 3

3.1.3 Software Interfaces 3

3.1.4 Communications Interfaces 3

3.2 Functional Requirements 3

3.2.1 <Functional Requirement or Feature #1> 3

3.2.2 <Functional Requirement or Feature #2> 3

3.5 Non-Functional Requirements 3

3.5.1 Performance 3

3.5.2 Reliability 3

3.5.3 Availability 3

3.5.4 Security 3

3.5.5 Maintainability 3

3.5.6 Portability 3

3.7 Design Constraints 3

3.9 Other Requirements 3

4. Analysis Models 4

4.1 Data Flow Diagrams (DFD) 4

5. Github link…………………………………………………………………………………………………….5

A. Appendices

A.1 Appendix 1

A.2 Appendix 2

# 1. INTRODUCTION 1.1 Purpose This Software Requirements Specification (SRS) aims to clearly define the functionality, features, and design constraints of the "Dashboard for National Accounts" system. The primary audience includes project managers, software engineers, database administrators, testers, documentation writers, and stakeholders. This document will guide the development and testing teams throughout the software development lifecycle (SDLC). 1.2 Scope The "Dashboard for National Accounts" is a web-based system developed using PHP and MySQL. It provides an interface for administrators and authorized users to manage, analyze, and report on key national economic data. The system will: - Provide a secure login system for authentication. - Display real-time and historical statistical economic data in a dashboard format. - Include a module for uploading and managing news relevant to national accounts. - Support report generation, viewing, and exporting features. - Offer a simple UI for managing user accounts and sessions. The system will not: - Perform predictive analytics or machine learning. - Allow public data modification (only read access unless authorized). 1.3 Definitions, Acronyms, and Abbreviations SRS – Software Requirements Specification PHP – Hypertext Preprocessor DBMS – Database Management System UI – User Interface DFD – Data Flow Diagram CRUD – Create, Read, Update, Delete LAMP – Linux, Apache, MySQL, PHP 1.4 References - IEEE Guide to Software Requirements Specifications - PHP Documentation (https://www.php.net/docs.php) - MySQL Documentation (https://dev.mysql.com/doc/) - HTML/CSS/JavaScript MDN Web Docs 1.5 Overview This document contains detailed information about the project's general description, functional and non-functional requirements, system interfaces, design constraints, and validation elements such as client approvals and deployment details. It is structured to follow IEEE standards for SRS documents.

# 2. GENERAL DESCRIPTION 2.1 Product Perspective The product is an independent web application. It does not require integration with third-party services but must run on a compatible web server environment. It follows a client-server architecture, where the front end interacts with the backend to retrieve and manage data. 2.2 Product Functions - User Authentication: Secure login/logout and session management. - Dashboard: Display data metrics in graphs, cards, and tables. - News Management: Add, update, delete national economic news. - Report Module: Generate, export, and print data reports. - User Account Management: Update profile details, change password, etc. 2.3 User Characteristics - Administrators: Tech-savvy personnel managing data, reports, and users. - Data Analysts: Professionals using the system to extract insights. - Viewers: Users with limited permissions to view dashboards and reports. 2.4 General Constraints - Must run on servers supporting PHP ≥ 7.4 - Browser support for latest versions of Chrome, Firefox, Edge - Built without using modern PHP frameworks (e.g., Laravel) 2.5 Assumptions and Dependencies - MySQL and Apache/Nginx installed and configured - JavaScript enabled in browser - Stable internet connection for live data updates

# 3. SPECIFIC REQUIREMENTS 3.1 External Interface Requirements 3.1.1 User Interfaces - Login Interface: Simple login form with error feedback. - Dashboard: Visual cards and graphs showing statistics. - Reports Interface: Filters and buttons for generating reports. - News Module: Form-based CRUD interface. - Account Page: Profile info with update options. 3.1.2 Hardware Interfaces - Client: Desktop/laptop/mobile device with modern browser - Server: LAMP/LEMP stack with PHP and MySQL support 3.1.3 Software Interfaces - PHP 7.4+ - MySQL 5.7+ - HTML5, CSS3, JavaScript (with jQuery if applicable) 3.1.4 Communications Interfaces - HTTP/HTTPS protocols - Browser-based interaction over local or hosted network 3.2 Functional Requirements 3.2.1 User Authentication Inputs: Email and password Processing: Verify user against DB, establish session Outputs: Success redirects to dashboard; failure returns error Error Handling: Invalid email/password alerts 3.2.2 Dashboard Display Inputs: None (uses session) Processing: Fetch and format data from DB Outputs: Cards, tables, and charts 3.2.3 Report Generation Inputs: Selected filters (dates, categories) Processing: Query DB, format for display Outputs: Exportable HTML/PDF/print formats 3.2.4 News Management Inputs: Title, content, date Processing: Add/edit/delete entries Outputs: List of recent news 3.5 Non-Functional Requirements 3.5.1 Performance - Dashboard load time < 2 seconds - Report generation < 3 seconds for normal data size 3.5.2 Reliability - 99% uptime target - Failover handling through error logs 3.5.3 Availability - Available 24/7 unless scheduled maintenance 3.5.4 Security - Hashed passwords with salting - Session-based authentication - Input sanitization against SQL injection and XSS 3.5.5 Maintainability - Modular file structure - Clear code comments - Reusable include files 3.5.6 Portability - Tested on XAMPP, WAMP, LAMP environments - Responsive design for multiple screen sizes 3.7 Design Constraints - Built using plain PHP - No external PHP libraries or MVC frameworks - MySQL database only 3.9 Other Requirements - Include favicon and page title for all pages - Navigation bar with logout and page links - Dark mode toggle (optional)

# 4. Analysis Models

List all analysis models used in developing specific requirements previously given in this SRS. Each model should include an introduction and a narrative description. Furthermore, each model should be traceable the SRS’s requirements.

## 4.1 Data Flow Diagrams (DFD)

[Admin/User]

|

[User Authentication] <----> [User DB]

|

[Dashboard Display]

|

--------------------------------------------------------

| | | | |

[User DB] [News DB] [Stats DB] [Account DB] [Reports DB]

| | | | |

--------------------------------------------------------

|

[News Management] [Statistics Management] [Account Management] [Report Generation]

| | | |

|

[Admin/User]

# 5. GitHub Link

# <https://github.com/hemeshhere/Dashboard-For-National-Accounts.git>

# A. Appendices

Appendices may be used to provide additional (and hopefully helpful) information. If present, the SRS should explicitly state whether the information contained within an appendix is to be considered as a part of the SRS’s overall set of requirements.

Unless otherwise noted, the contents of the appendices are **for reference only** and do not form part of the binding requirements of the system.

#### **A.1 Appendix 1 – Conceptual Documents**

This includes the initial project outline and concept notes that helped define the scope and purpose of the "Dashboard for National Accounts" system.

**Included Documents:**

* Problem Statement and Solution Outline
* System Overview Diagram
* Feature Prioritization Matrix
* Stakeholder Interviews Summary

#### **A.3 Appendix 2 – Mockups and Wireframes**

Includes the initial screen sketches and UI wireframes used to gather feedback and refine the front-end design.

**Included Screens:**

* Login Page
* Dashboard Overview
* Reports Generation Page
* News Management Panel
* User Account Settings