APPLICATION DEVELOPMENT DOCUMENTATION GUIDELINES

Files Size: A4
Font: Arial - 12 - Double Spacing
Document Spacing: Top, Right, Bottom -1 Inch, Left - 1.5 Inches

This outline provides a standard structure, ensuring all key academic and technical components of a system project are properly addressed.

Preliminaries

- **Title Page:** Includes the full title of the project, authors, and submission date.
- **Abstract:** A concise (200-300 word) summary covering the problem, methodology, key findings, and conclusion.
- **Table of Contents, List of Figures, List of Tables**

Chapter I: The Problem and its Background

This chapter sets the stage for the entire project.

- **1.1 Project Context and Introduction:** Provide the general background, rationale, and significance of the study. Explain *why* this problem needs to be solved.
- **1.2 Statement of the Problem:** Clearly articulate the specific problems (general and specific) that the system aims to solve. These should often align directly with your objectives.
- **1.3 Objectives of the Study:** List the specific, measurable goals (usually 3-5) that the project is designed to achieve.
- **1.4 Scope and Delimitations:** Clearly define what the system **will** and **will not** cover, including target users, features, and hardware/software limits.
- **1.5 Review of Related Literature and Studies (RRL/RRS): (CRITICAL MISSING SECTION)** Survey and discuss previous studies, existing technologies, and literature that are relevant to your project. This establishes your project's theoretical foundation and justifies its novelty.
- **1.6 Conceptual or Theoretical Framework:** This is where you include your **Frameworks** (e.g., the Input-Process-Output model, or the specific software development life cycle you follow).
- **1.7 Definition of Terms:** Define key technical, operational, and conceptual terms used throughout the document.

Chapter II: Methodology and System Design

This chapter explains **how** the system was built and implemented.

- **2.1 Project Methodology:** Clearly identify the specific Software Development Life Cycle (SDLC) model used (e.g., Agile, Waterfall, Spiral). Detail the steps taken within that model.
- **2.2 Technical System Architecture:** Describe the overall structure of the system (e.g., client-server, 3-tier).
    - **2.2.1 System Environment (Hardware and Software Requirements):** List all necessary tools, programming languages, libraries, and hardware specifications.
- **2.3 System Modeling and Blueprints:**
    - **2.3.1 System Flowchart/Data Flow Diagrams (DFD):** Graphical representation of the system's process flow.
    - **2.3.2 Wireframes and User Interface (UI) Prototypes:** Visual mock-ups showing the screen layouts and user experience.
    - **2.3.3 Database Structure and Design:** Include the **Entity-Relationship Diagram (ERD)** and the detailed database table structure (your *table of the database and its usage*). Explain the normalization level.
- **2.4 Implementation Procedures:** Describe the actual coding, testing, and deployment processes.

Chapter III: Results and Discussion

This chapter presents the outcome and analysis of the project.

- **3.1 Presentation of the System Features:** Detailed walkthrough of the final system, usually presented via screenshots or functional descriptions corresponding to the objectives.
- **3.2 Discussion of Results:** Analyze the system's performance, how it met the objectives outlined in Chapter I, and explain any deviations or compromises made during development.
- **3.3 System Testing and Evaluation: (CRITICAL MISSING SECTION)** Document the results of testing (e.g., unit tests, user acceptance testing (UAT)). Include the instrument used (like a System Usability Scale or quality matrix) and present the computed data.

Final Sections

- **Conclusion:** Summarize the main achievements of the study and reiterate the most important findings. This must directly address the problems and objectives from Chapter I.
- **Recommendations:** Provide specific suggestions for future enhancements, modifications, or related research.
- **Bibliography/References:** A complete list of all sources cited in the RRL, Frameworks, and Methodology.
- **Appendices:** Include supplementary material (e.g., source code snippets, raw data, ethics clearances, testing forms, resumes of developers).