**1. Title Page**

* **Project Title:** DormNest: Student Accommodation Finder Application
* **Prepared by:** [Your Name]
* **Institution:** [Your Institution/Organization, if applicable]
* **Date:** [Completion Date]
* **Mentor/Supervisor:** [Supervisor’s Name, if any]

**2. Executive Summary**

* A brief overview of DormNest, its objectives, and the problem it addresses.
* Summary of key features and technical highlights.

**3. Table of Contents**

**4. Introduction**

* **Background:** Brief context about the need for a student accommodation finder.
* **Purpose:** What DormNest aims to accomplish.
* **Scope:** The functional and non-functional requirements the project meets.

**5. Requirements Analysis**

* **Functional Requirements:** List all main functions DormNest provides, such as registration for students and owners, property listing, search functionality, etc.
* **Non-Functional Requirements:** Outline the performance, scalability, security, and usability requirements.

**6. System Design**

* **System Architecture:** Diagram and explain the architecture of DormNest, showing how different components (front-end, back-end, database) interact.
* **Database Design:** ER Diagram and description of the PostgreSQL database, including tables and relationships.
* **User Interface Design:** Describe the primary UI components and provide wireframes or screenshots.
* **Key Features and Modules:** Overview of major features, such as:
  + Registration forms for students and owners.
  + Owner Dashboard displaying property listings and owner information.
  + Search and filter capabilities for users seeking accommodations.

**7. Implementation**

* **Technologies Used:** List of tools, languages, frameworks, and libraries, such as Java (for backend), PostgreSQL (database), and any front-end frameworks.
* **Database Connection:** Details on how PostgreSQL is integrated and connected to the application.
* **Code Structure:** Overview of the project structure, including notable packages or modules.

**8. Testing and Evaluation**

* **Testing Approach:** Describe testing methods, such as unit testing, integration testing, and user acceptance testing.
* **Test Cases and Results:** Examples of test cases for critical functions and results.
* **Performance Evaluation:** Any metrics or benchmarks (e.g., response time, load handling) evaluated during testing.

**9. Challenges and Solutions**

* Description of challenges faced during development and solutions you implemented. Examples could include integration issues with PostgreSQL, user interface challenges, or managing authentication securely.

**11. Conclusion**

* Summarize DormNest’s contributions, the skills you developed during the project, and the overall impact of the application.