Institution Name

Bachelor Degree – Course Name

Project Title

Software Engineering and Databases

Author Name(s)

Supervisor: Name

December 2025

Abstract

This report presents *[project name]*, a system designed to *[short purpose]*. The main objective is to *[objective 1]* while ensuring *[objective 2]*. The solution is based on *[brief architecture]* and a relational database model that enforces integrity through constraints and normalized design. Evaluation includes functional validation and database-oriented testing, complemented by a basic performance analysis of the critical queries. Results show *[main outcome]*, with limitations related to *[key limitation]*.

Keywords: databases, software engineering, data modelling, relational design, testing

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Context and Motivation

1.2 Problem Statement

1.3 Objectives

- Objective 1: *[...]*
- Objective 2: *[...]*
- Objective 3: *[...]*

1.4 Scope and Limitations

1.5 Document Structure

Chapter 2

Background and Related Work

2.1 Core Concepts

2.2 Related Systems / Approaches

2.3 Positioning of This Work

Chapter 3

Requirements Analysis

3.1 Stakeholders and Users

3.2 Functional Requirements

FR1 [**Feature name**]: *[requirement description]*

FR2 [**Feature name**]: *[requirement description]*

FR3 [**Feature name**]: *[requirement description]*

3.3 Non-Functional Requirements

NFR1 **Security**: *[e.g., role-based access control for ...]*

NFR2 **Performance**: *[e.g., critical search queries under X ms for N rows]*

NFR3 **Usability**: *[e.g., key workflow in <= K steps]*

NFR4 **Maintainability**: *[e.g., modular structure, documentation, tests]*

3.4 Use Cases / User Stories

Table 3.1: Representative Use Cases

UC ID	Description
UC1	User logs in and accesses the dashboard.
UC2	User creates a new *[entity]* and the system validates required fields.
UC3	User searches *[entity]* using filters and sorting.

3.5 Acceptance Criteria (Optional)

Chapter 4

System Design

4.1 Architecture Overview

Figure 4.1: High-level architecture of the system.

4.2 Main Components and Responsibilities

4.3 Key Interactions (Optional)

4.4 Design Decisions and Trade-offs

Chapter 5

Database Design

5.1 Conceptual Model (ER Diagram)

Placeholder: ER diagram (entities, relationships, cardinalities)

Figure 5.1: Entity-Relationship diagram.

5.2 Logical Model (Relational Schema)

Table 5.1: Example Schema Summary (adapt to your project)

Table	**Notes (PK/FK/Constraints)**
users	PK(user_id), UNIQUE(email), NOT NULL(password_hash)
orders	PK(order_id), FK(user_id → users), CHECK(status in ...)
order_items	PK(order_item_id), FK(order_id), FK(product_id), CHECK(quantity > 0)

5.3 Normalization Discussion

5.4 Constraints and Integrity

5.5 Physical Design (Indexes and Performance Rationale)

Chapter 6

Implementation Overview

6.1 Technology Stack

6.2 Feature Mapping (Requirements to Modules)

Table 6.1: Traceability Example

Req	**Implementation**	**Evidence**
FR1	*[module/functionality]*	*[screenshot/test]*
FR2	*[module/functionality]*	*[screenshot/test]*
NFR1	*[security mechanism]*	*[test/cfg]*

6.3 Security Considerations

Chapter 7

Testing and Evaluation

7.1 Test Strategy

7.2 Test Results

Table 7.1: Testing Summary

Area	**What was tested**	**Result**
Business logic	*[rule validations, edge cases]*	Pass
Database constraints	*[FK integrity, CHECK constraints]*	Pass
API / UI flows	*[UC1, UC2, UC3]*	Pass

7.3 Performance Evaluation (Basic but Real)

Chapter 8

Results and Discussion

8.1 Key Results

8.2 Limitations

8.3 Trade-offs

Chapter 9

Conclusion and Future Work

9.1 Conclusion

9.2 Future Work

- *[feature improvement]*
- *[performance improvement]*
- *[security improvement]*

Appendix A

Appendix A: Full Schema Listing

Appendix B

Appendix B: Representative Queries

Appendix C

Appendix C: Additional Test Evidence