Technical Systems Document

System Name: Amandla High School Locker Booking System
Prepared by: Nishay Hira
Student Number: 69724806
Date: 10 October 2025

---

1. System Overview

The Amandla High School Locker Booking System is a database-driven web application designed to simplify the management of student locker allocations, payments, and parent communication.
It provides two main roles:

- Parent: Can log in, book a locker, and make payments.
- Administrator: Can manage lockers, monitor payments, and generate reports.

The system ensures transparency, efficiency, and accountability in the locker management process.

---

2. System Objectives

- Automate locker assignment and payment tracking.
- Provide secure login for both parents and administrators.
- Generate management information system (MIS) reports.
- Allow real-time updates on locker availability and payment status.
- Maintain a digital database of all parents, students, and lockers.

---

3. System Architecture

3.1 Overview

The system uses a three-tier architecture:

1. Presentation Layer (Frontend)
    - Developed using HTML/CSS, JavaScript, or Tkinter GUI (Python).

- Provides user interfaces for login, booking, and administration dashboards.

2. Application Layer (Backend Logic)
   - Handles login authentication, booking validation, and payment tracking logic.
   - Developed using PHP (if web-based) or Python (if desktop version).
3. Data Layer (Database)
   - Uses MySQL for data storage and management.
   - Stores all user, locker, and payment data securely.

---

4. System Components

Component	Description
Login Module	Authenticates users based on credentials and role.
Parent Dashboard	Allows parents to book lockers and make payments.
Admin Dashboard	Enables administrators to view all bookings and update payment statuses.
Locker Management	Displays all lockers, their locations, and availability.
Payment Tracking	Tracks R100 payments per student and updates statuses.
Reporting Module	Generates summaries for payments, lockers, and student records.

---

5. Database Design

5.1 Database Name: amanda_locker _system

5.2 Tables and Fields

1. parents

Field	Type	Description
parent_id	INT (PK, AUTO_INCREMENT)	Unique parent ID
title	VARCHAR(10)	Mr/Mrs/Ms/Dr
id_number	VARCHAR(20)	Parent ID number
first_name	VARCHAR(50)	Parent first name
last_name	VARCHAR(50)	Parent last name
email	VARCHAR(100)	Parent email
address	VARCHAR(255)	Home address
phone	VARCHAR(15)	Contact number
student_id	INT (FK)	Links to student table

2. students

Field	Type	Description
student_id	INT (PK, AUTO_INCREMENT)	Unique student ID
first_name	VARCHAR(50)	Student first name
last_name	VARCHAR(50)	Student last name
grade	VARCHAR(10)	Student grade level

3. lockers

Field	Type	Description
locker_id	INT (PK, AUTO_INCREMENT)	Unique locker ID
locker_number	VARCHAR(10)	Locker number
status	ENUM('Available','Booked','Unavailable')	Current locker status
location	VARCHAR(100)	Locker location in school

4. bookings

Field	Type	Description
booking_id	INT (PK, AUTO_INCREMENT)	Unique booking ID
parent_id	INT (FK)	Linked parent ID
student_id	INT (FK)	Linked student ID
locker_id	INT (FK)	Linked locker ID
booking_date	DATE	Date of locker booking
payment_status	ENUM('Paid','Unpaid')	Payment completion status

---

6. Data Flow Diagram (DFD)

Level 0 (Context Diagram)

- Parents → Login → System → Access Parent Dashboard
- Administrators → Login → System → Access Admin Dashboard

Level 1 (Expanded View)

1. Parent Process
    - Inputs: Email, password, student info.
    - Outputs: Locker booking, payment confirmation.
2. Admin Process
    - Inputs: Credentials, payment update, locker data.
    - Outputs: Reports, payment status updates.

---

7. System Process Descriptions

7.1 Login Process

1. User enters email, password, and selects role.
2. System validates credentials from database.
3. If role = Parent → redirect to Parent Dashboard.
4. If role = Admin → redirect to Admin Dashboard.

7.2 Payment Tracking Process

1. Admin logs in and opens Payment Tracking.
2. System retrieves booking and payment data.
3. Admin can update payment status for any record.
4. Changes are stored in the database and reflected in reports.

---

8. Security Features

- Passwords are hashed using MD5 or bcrypt.
- Role-based access control (Parent vs Administrator).
- Input validation to prevent SQL injection and data corruption.
- Database backups recommended weekly.

---

9. System Installation Instructions

1. Install Required Software
    - XAMPP (for PHP + MySQL) or Python 3.x (for Tkinter version).
2. Database Setup
    - Open phpMyAdmin → Create a new database `locker_booking_system`.
    - Import the SQL file provided (`locker_booking_system.sql`).
3. Application Setup
    - Copy the project folder into `htdocs` (if using XAMPP).
    - Run the system at:
      http://localhost/locker_booking_system/
4. Login Credentials
    - Admin: admin@gmail.com / admin123
    - Parent (Sample): parent@gmail.com / parent123

10. System Maintenance

- Regularly check for unpaid lockers and send reminders.
- Back up database monthly using phpMyAdmin export.
- Test login and booking functionality after updates.
- Clear old booking records annually to maintain performance.

---

11. Reporting Features

Report Type	Purpose
Payment Summary	Tracks who has paid or not paid the R100 fee.
Locker Utilization	Shows total lockers, booked vs available.
Student Directory	Lists students, grades, and linked parents.
Booking History	Displays booking trends and payment status over time.

All reports can be exported as PDF or Excel for school administration meetings.

---

12. Backup and Recovery Plan

1. Backup:
    - Weekly export of database from phpMyAdmin.
    - Store in both local and cloud storage.
2. Recovery:
    - In case of data loss, re-import .sql backup into phpMyAdmin.

---

13. System Testing Summary

Test Case	Expected Result	Status
Login with valid credentials	Redirects to correct dashboard	✓ Passed
Book locker as parent	Updates locker status to “Booked”	✓ Passed
Update payment status	Reflects instantly in database	✓ Passed
Generate report	Displays accurate data summary	✓ Passed

---

14. Future Enhancements

- Integration of online payment gateway.
- SMS or Email reminders for unpaid lockers.
- Role expansion for Teachers or Finance Officers.
- Mobile-friendly responsive design.

---

15. Conclusion

The Amandla High School Locker Booking System provides a modern, secure, and efficient method for managing student lockers. It reduces administrative burden, enhances transparency, and allows both parents and administrators to interact with the system intuitively.

Prepared by: Nishay Hira
Student Number: 69724806
Date: 10 October 2025
Signature: _____