Garage Management System - Detailed Documentation

1. Ideation Phase

The idea for the Garage Management System arises from the common issues faced in traditional garages, where most daily tasks such as customer data management, billing, service status tracking, and spare parts inventory are handled manually. Manual records lead to errors, misplaced job cards, delayed servicing, and customer dissatisfaction. Hence, the ideation phase focuses on designing a system that simplifies operations, provides real-time vehicle service tracking, maintains service history logs, and generates accurate bills efficiently.

Key Objectives

- Reduce paperwork and manual errors.
- Maintain centralized customer and vehicle records.
- Track service jobs and mechanic assignments.
- Automate billing and spare part usage tracking.

---

2. Project Planning Phase

The planning phase outlines the workflow for development, team responsibilities, timeline, and resources.

Project Scope

The system will handle: - Customer and Vehicle Registration - Service Job Creation - Mechanic Assignment - Spare Parts Management - Invoice Generation - Report Creation

Tools & Technologies

Component	Choice
Frontend	HTML, CSS, JavaScript
Backend	Python / Java / PHP
Database	MySQL
IDE	VS Code / Eclipse

Timeline (Simplified)

Stage	Duration
Requirement Gathering	1 Week
Design	2 Weeks
Development	3 Weeks
Testing	1 Week
Deployment	1 Week

---

3. Project Design Phase

The design phase involves creating the architecture, database, and interface layout.

System Architecture Diagram

```
+------------------+        +----------------------+
|   User Interface | <----> |  Application Layer   |
+------------------+        +----------------------+
                                       |
                                       v
                            +------------------+
                            |   Database Layer  |
                            +------------------+
```

ER Diagram (Text Representation)

```
Customer ----< Vehicle ----< ServiceJob ----< Invoice
                  |               |
                  |               v
                  |           SpareParts
```

UI Overview

- **Dashboard:** Overview of total customers, pending jobs, and inventory.
- **Customer Form:** Register new customer and link vehicle.
- **Service Entry Screen:** Create job card and assign mechanic.
- **Billing Screen:** Auto-calculates spare parts cost and labor.

---

4. Requirement Analysis

Functional Requirements

- System must allow secure login.
- Ability to register customers and vehicles.
- Create, update, and close service job cards.
- Track spare parts usage and update stock automatically.
- Generate invoices and print/download reports.

Non-Functional Requirements

- User-friendly interface.
- Secure handling of user login and data.
- Fast access to service records and billing operations.
- Reliable backup and recovery support.

---

5. Performance Testing

Performance testing ensures the system works efficiently under real-time workloads.

Test Parameters

Test	Expected Outcome
Database Load Test	Data retrieval should be fast even with large records
Stress Test	System should not crash on multiple job entries
Billing Speed Test	Invoice must generate instantly

Sample Flow Test Diagram

```
User -> Create Service Job -> Assign Mechanic -> Update Parts -> Generate Bill -
> Save Record
```

Conclusion

The Garage Management System offers digital transformation for traditional garages by simplifying service operations, improving customer satisfaction, and ensuring accurate and efficient workflow management.