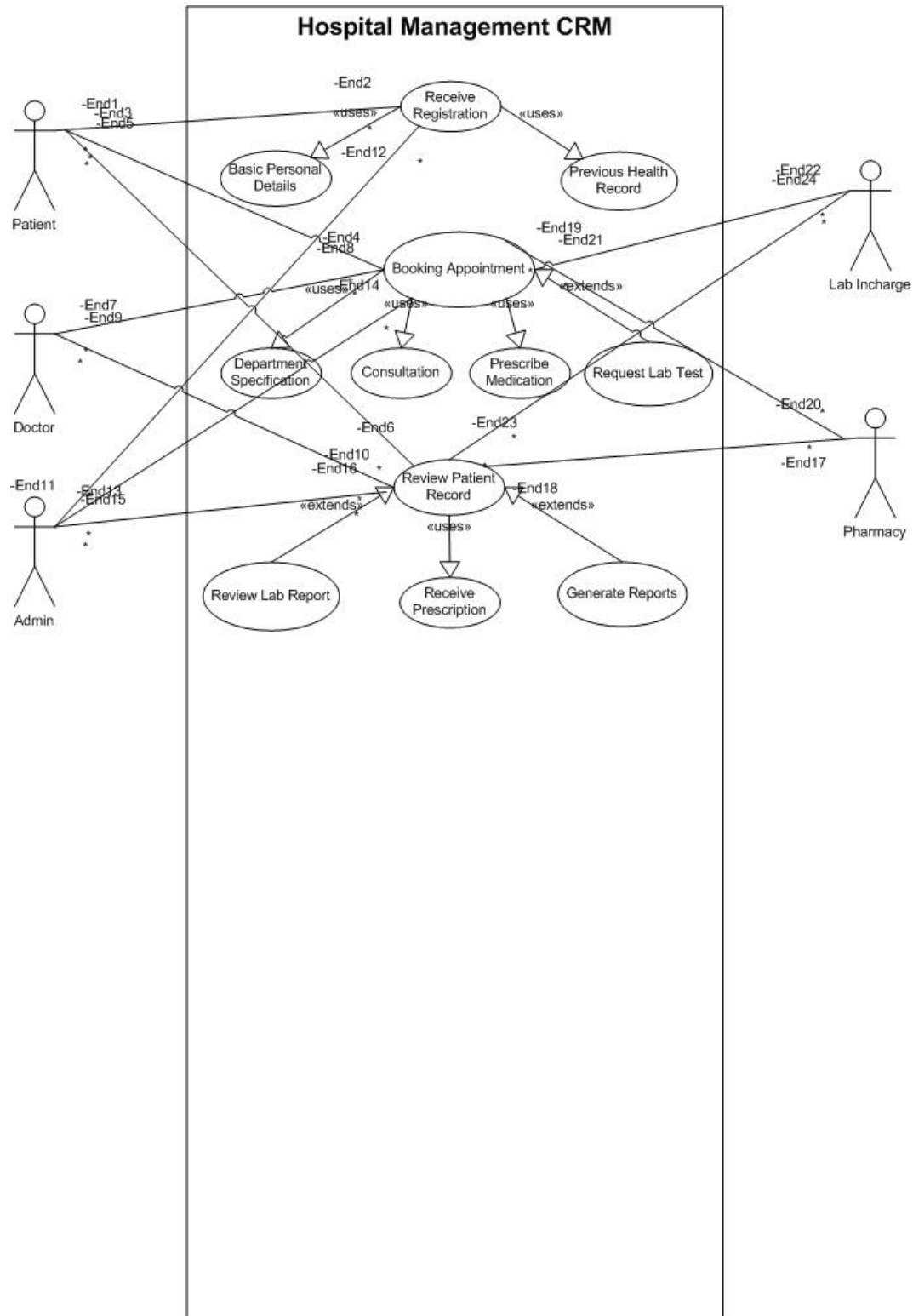
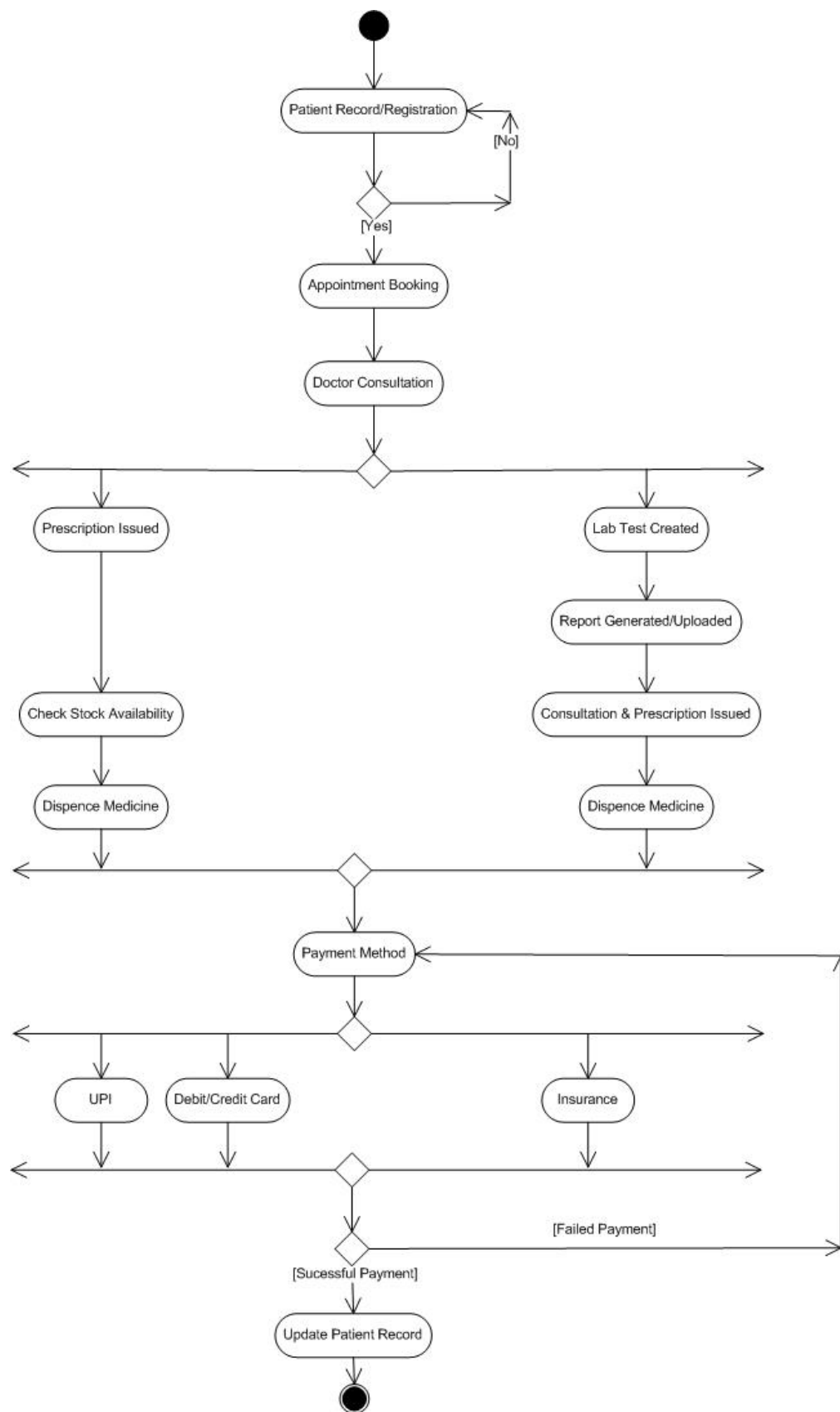


Document 6- Please prepare a use case diagram, activity diagram and a use case specification document.

Use Case Diagram:



Activity Diagram:



Use Case Specification:

Patient Registration

Field	Details
Use Case Name	Patient Registration
Brief Description	This use case describes how a new patient is registered in the hospital CRM system.
Actors	Primary Actor: Receptionist Secondary Actor: Patient, Database
Pre-Conditions	- Receptionist logged into CRM - Patient provides necessary details
Basic Flow	1. Receptionist selects 'Register Patient' 2. Enters personal details (Name, Age, Gender, Contact) 3. System validates mandatory fields 4. Unique Patient ID generated 5. Record saved successfully
Alternate Flow	- Patient self-registers via patient portal - Existing patient record search to avoid duplicates
Exceptional Flows	- Missing/incorrect data entered → Error message - Duplicate patient entry detected
Post-Conditions	- New patient record created - Patient ID assigned
Assumptions	- Patient provides correct details
Constraints	- Only authorized staff can register - Mandatory fields must be filled
Dependencies	- Database system availability
Inputs & Outputs	Input: Patient details Output: Patient ID, Registration Confirmation
Business Rules	- No duplicate records allowed - Age must be valid (>0)
Miscellaneous Information	Patient data securely stored in compliance with privacy laws

Appointment Booking

Field	Details
Use Case Name	Appointment Booking
Brief Description	This use case describes how a patient books an appointment with a doctor using the Hospital Management CRM.
Actors	Primary Actor: Patient Secondary Actors: Receptionist, Doctor
Pre-Conditions	- Patient must be registered in CRM - Doctor's availability schedule must exist - Active internet connection/device
Basic Flow	1. Patient logs into CRM

	2. Selects 'Book Appointment' 3. System shows available doctors & time slots 4. Patient selects doctor & time 5. System confirms availability 6. Appointment saved and confirmation sent
Alternate Flow	- Receptionist books appointment on behalf of patient - Patient modifies/cancels appointment within allowed time
Exceptional Flows	- If doctor not available → system shows alternative slots - If invalid details entered → error message displayed - If booking system is down → request queued
Post-Conditions	- Appointment created in CRM - Notifications sent to patient and doctor
Assumptions	- Patient has valid login credentials - Doctor's schedule is up to date
Constraints	- Limited slots per doctor per day - Appointment booking only within working hours
Dependencies	- Patient registration module - Doctor schedule management module
Inputs & Outputs	Input: Patient details, doctor selection, date/time Output: Appointment confirmation (email/SMS/CRM notification)
Business Rules	- Only registered patients can book - A doctor cannot be double-booked - Cancellation only allowed 24 hours before appointment
Miscellaneous Information	- Appointment data stored securely in hospital database - Links to billing module if paid appointment

Doctor Consultation

Field	Details
Use Case Name	Doctor Consultation
Brief Description	This use case describes how a doctor consults a patient and records treatment details.
Actors	Primary Actor: Doctor Secondary Actor: Patient, Receptionist
Pre-Conditions	- Patient must have a valid appointment - Doctor logged into CRM
Basic Flow	1. Patient visits consultation room 2. Doctor retrieves patient record 3. Doctor reviews medical history 4. Doctor enters diagnosis and prescription 5. Consultation record saved
Alternate Flow	- Emergency patient directly consulted without prior booking
Exceptional Flows	- Patient ID not found - System error retrieving records
Post-Conditions	- Consultation details updated in CRM - Prescription forwarded to pharmacy
Assumptions	- Patient attends on scheduled time
Constraints	- Consultation limited to available slots
Dependencies	- Appointment booking module

	- Patient registration module
Inputs & Outputs	Input: Patient symptoms, medical history Output: Consultation record, prescription
Business Rules	- Only assigned doctor can consult patient
Miscellaneous Information	Consultation data stored for future reports

Medical Records Management

Field	Details
Use Case Name	Medical Records Management
Brief Description	This use case describes how patient medical records are created, updated, and retrieved.
Actors	Primary Actor: Doctor Secondary Actors: Receptionist, Patient
Pre-Conditions	- Patient must be registered - Authorized login required
Basic Flow	1. Doctor/receptionist searches for patient 2. System retrieves records 3. New details (consultation, tests, prescriptions) updated 4. Record saved
Alternate Flow	- Patient accesses limited records via portal
Exceptional Flows	- Record not found - Unauthorized access attempt
Post-Conditions	- Updated medical record stored in CRM
Assumptions	- Records are always backed up
Constraints	- Privacy and access control rules apply
Dependencies	- Database, consultation, and lab modules
Inputs & Outputs	Input: Patient ID, new details Output: Updated record
Business Rules	- Only authorized roles can edit records
Miscellaneous Information	Data encrypted for security

Pharmacy Prescription

Field	Details
Use Case Name	Pharmacy Prescription
Brief Description	This use case describes how prescriptions are processed by the pharmacy.
Actors	Primary Actor: Pharmacist Secondary Actors: Doctor, Patient
Pre-Conditions	- Doctor has entered prescription - Patient record available
Basic Flow	1. Pharmacist views prescription in CRM 2. Medicines checked in inventory 3. Medicines issued to patient

	4. Record updated
Alternate Flow	- Patient requests home delivery
Exceptional Flows	- Medicine not in stock → Alternative suggested - Invalid prescription
Post-Conditions	- Medicines dispensed - Inventory updated
Assumptions	- Pharmacy inventory is updated
Constraints	- Only prescribed medicines can be issued
Dependencies	- Consultation and inventory modules
Inputs & Outputs	Input: Prescription ID Output: Medicines issued receipt
Business Rules	- No medicine without prescription
Miscellaneous Information	Alerts for low-stock medicines

Lab Test Request & Results

Field	Details
Use Case Name	Lab Test Request & Results
Brief Description	This use case describes how lab tests are requested and results recorded.
Actors	Primary Actor: Lab Technician Secondary Actors: Doctor, Patient
Pre-Conditions	- Doctor requests a lab test - Patient registered
Basic Flow	1. Doctor enters lab test request 2. Lab technician receives request 3. Sample collected from patient 4. Test conducted 5. Results entered into CRM
Alternate Flow	- Emergency test priority given
Exceptional Flows	- Sample not collected - Test machine failure
Post-Conditions	- Results updated in CRM - Patient/doctor notified
Assumptions	- Equipment working properly
Constraints	- Only authorized staff can enter results
Dependencies	- Consultation module - Database
Inputs & Outputs	Input: Test request Output: Test results report
Business Rules	- Test results must be verified before release
Miscellaneous Information	Test results stored securely

Billing & Payment

Field	Details
Use Case Name	Billing & Payment
Brief Description	This use case describes how hospital bills are generated and payments processed.
Actors	Primary Actor: Receptionist Secondary Actors: Patient, Accounts System
Pre-Conditions	- Patient registered - Services availed recorded
Basic Flow	1. Receptionist generates bill 2. System calculates charges (consultation, lab, pharmacy) 3. Bill shown to patient 4. Payment made 5. Receipt generated
Alternate Flow	- Online payment option used
Exceptional Flows	- Payment failure - Bill mismatch error
Post-Conditions	- Bill settled - Receipt generated
Assumptions	- Charges updated in system
Constraints	- Payment must be completed before discharge
Dependencies	- Consultation, pharmacy, lab modules
Inputs & Outputs	Input: Patient ID, services Output: Bill receipt
Business Rules	- Discounts only with approval
Miscellaneous Information	Links to insurance claim if applicable

Insurance Claim Processing

Field	Details
Use Case Name	Insurance Claim Processing
Brief Description	This use case describes how insurance claims are submitted and processed.
Actors	Primary Actor: Receptionist/Patient Secondary Actors: Insurance System
Pre-Conditions	- Patient has valid insurance - Hospital tied with insurer
Basic Flow	1. Patient submits insurance details 2. Receptionist uploads claim 3. Insurance system validates claim 4. Approval/rejection received 5. Bill updated accordingly
Alternate Flow	- Partial claim approval
Exceptional Flows	- Invalid policy - Claim rejection
Post-Conditions	- Claim settled

	- Patient pays balance if needed
Assumptions	- Insurance details valid
Constraints	- Dependent on insurer approval
Dependencies	- Billing module - Insurance APIs
Inputs & Outputs	Input: Insurance policy, bill Output: Claim approval/rejection
Business Rules	- Only valid policies accepted
Miscellaneous Information	Hospital maintains claim history

Report Generation

Field	Details
Use Case Name	Report Generation
Brief Description	This use case describes how reports are generated for hospital management and doctors.
Actors	Primary Actor: Admin Secondary Actors: Doctor, Receptionist
Pre-Conditions	- Data available in system
Basic Flow	1. Admin selects report type (daily, monthly, patient, finance) 2. System processes data 3. Report generated 4. Report viewed/downloaded
Alternate Flow	- Scheduled automatic report generation
Exceptional Flows	- No data found - Report format error
Post-Conditions	- Report saved/exported
Assumptions	- Data is up to date
Constraints	- Reports limited to authorized users
Dependencies	- All modules providing input data
Inputs & Outputs	Input: Report parameters Output: Report (PDF/Excel)
Business Rules	- Only admin can generate financial reports
Miscellaneous Information	Reports archived for audits

Document 7- Screens and pages

The following are the mock-ups created using **Balsamic**

A Web Page

http://provisionhospitals.in

Provision Hospitals

Login

Username

Password

Login

[Forgot Password](#)

Register New User

A Web Page

http://provisionhospitals.in

Provision Hospitals

Kamal Nandagopal

Logout

Dashboard

Appointments

Medical Records

Prescriptions

Billing & Payment

Lab Reports

Setting

Upcoming Appointments

Date	Time	Doctor	Status	Status
12-04-2024	10:00 AM	Dr. Smith	Confirmed	Confirmed
14-04-2024	11:30 AM	Dr. Brown	Confirmed	Confirmed

Recent Prescriptions

AmoxicillinDownload

Medical Records

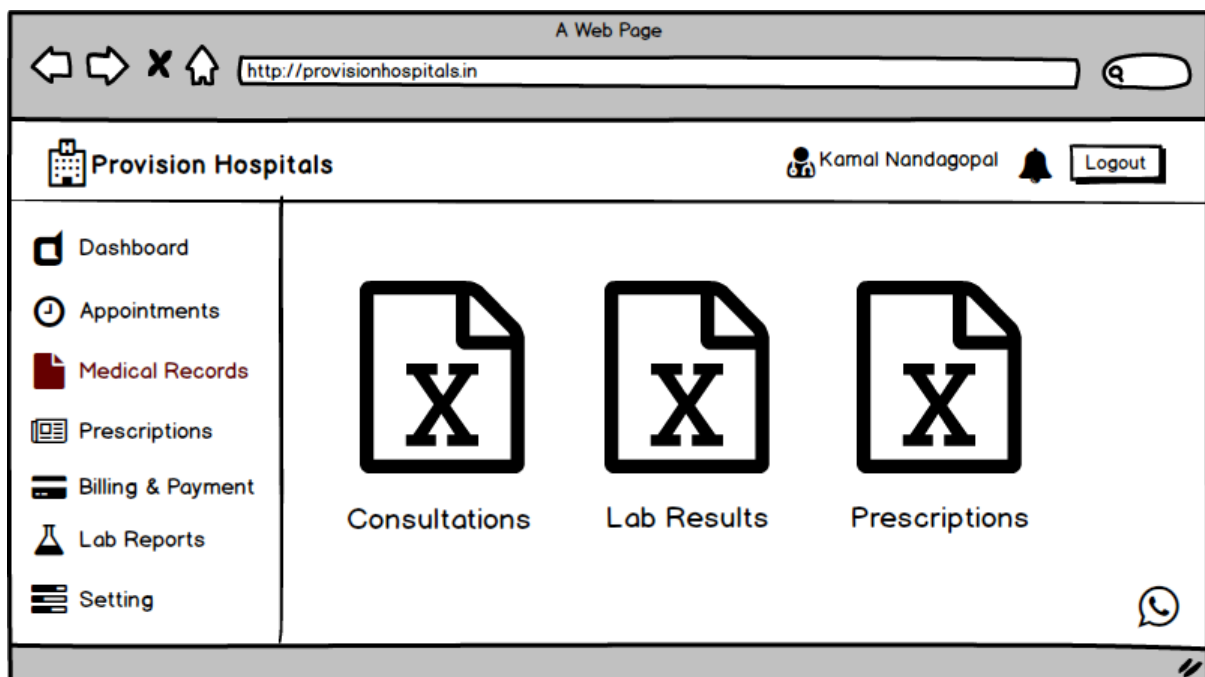
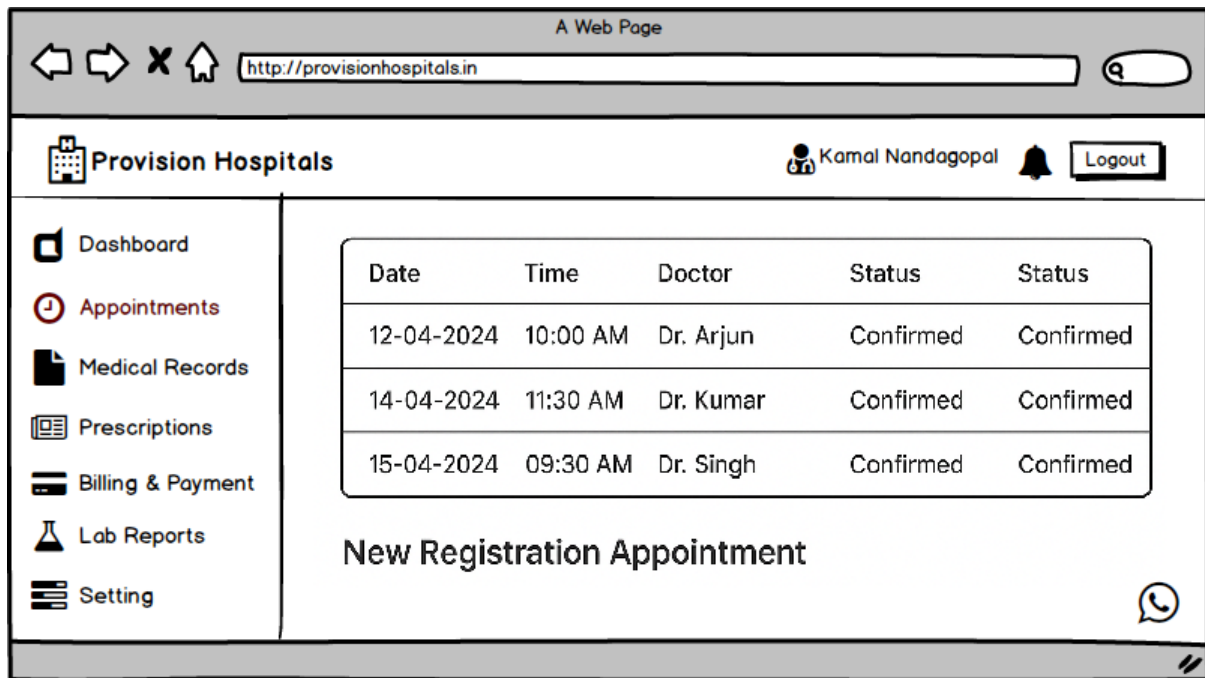
Summary: Diabetes, HypertensionLatest Visits

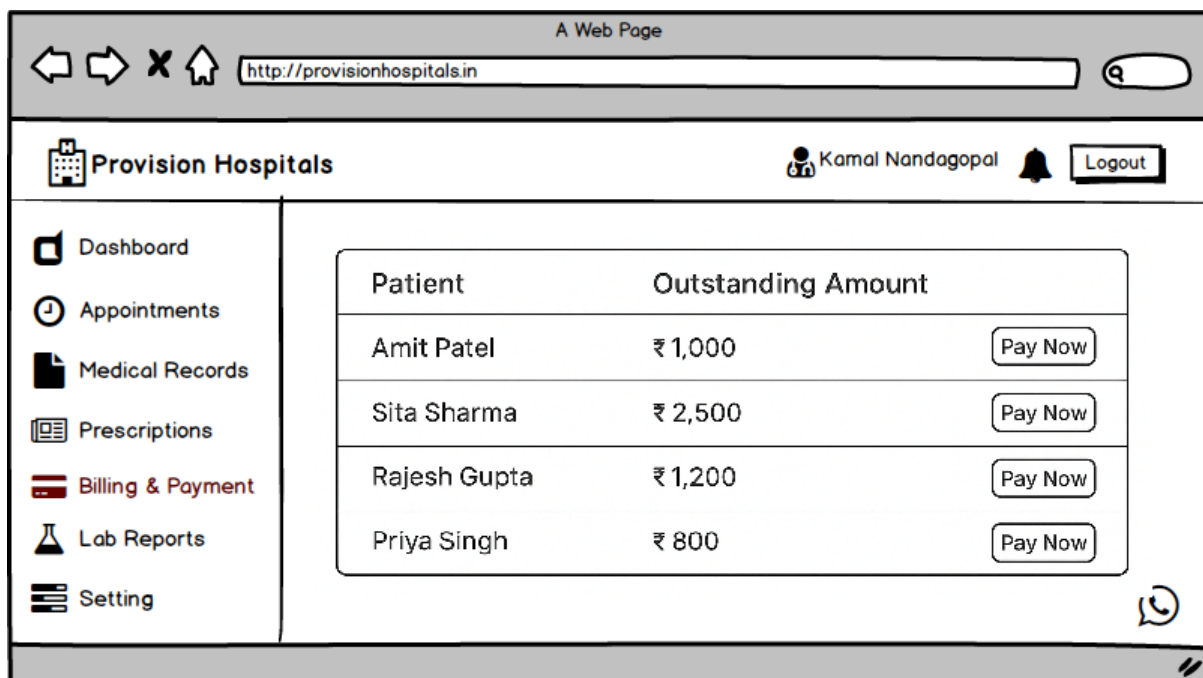
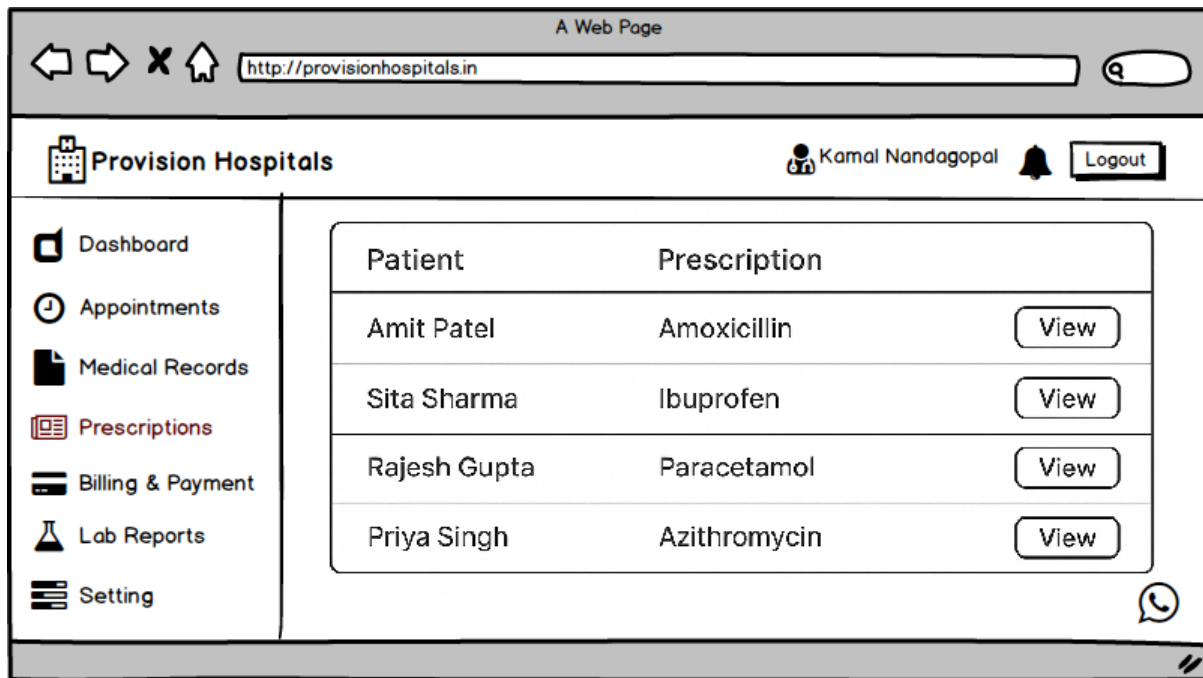
Quick Links

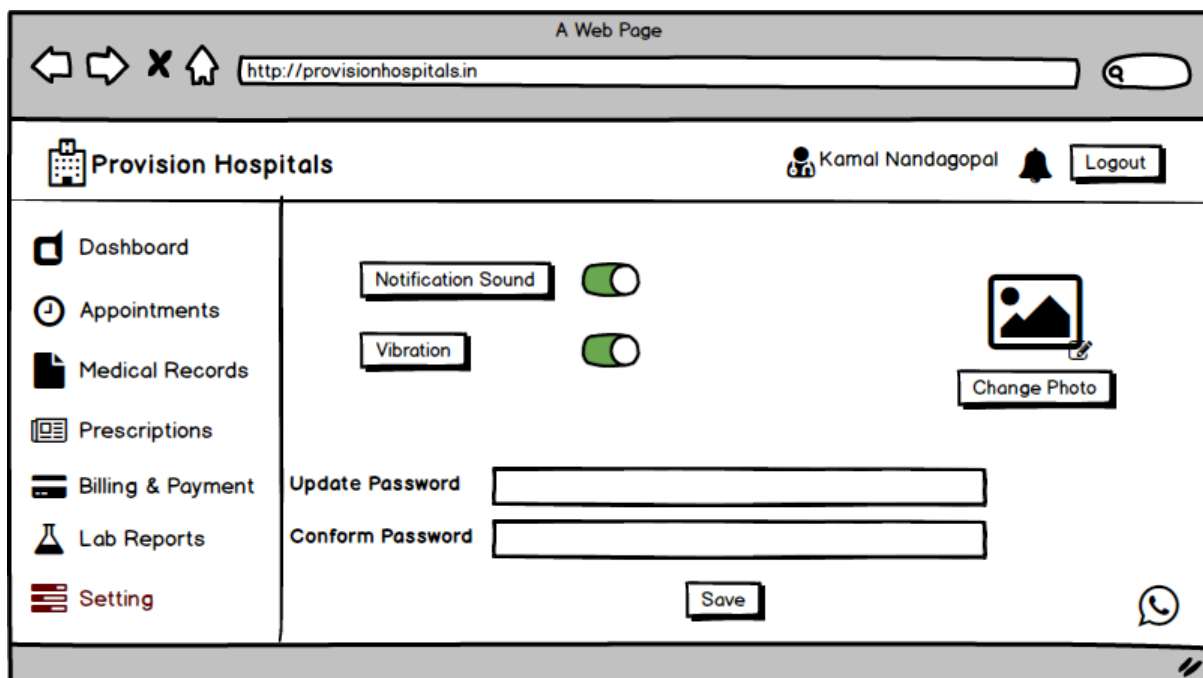
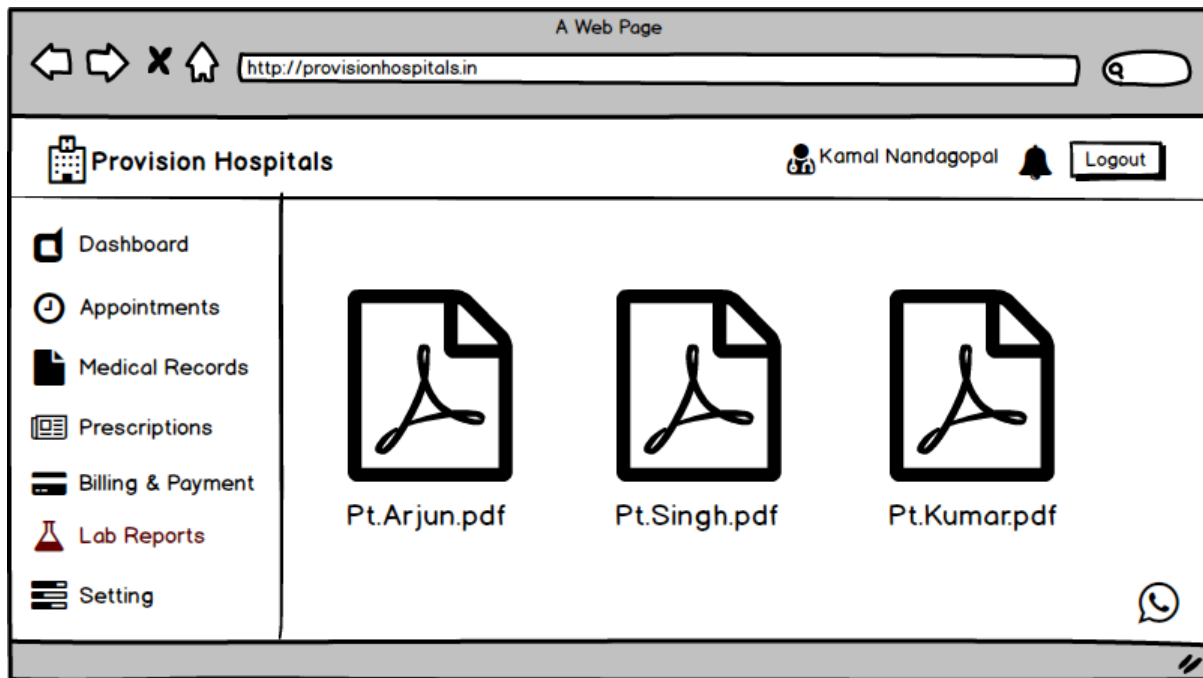
Book Appointment

View Bills

Order Medicines







Document 8 – Tools: Visio and Axure

For this project, I have mainly used **Microsoft Visio** and **Balsamiq** as supporting tools.

Visio: was used for preparing UML diagrams like Use Case, Activity, and Workflow diagrams. I found it very useful for creating clear process flows and connecting each actor's role with the system. It helped me in representing the hospital CRM processes in a structured and professional way.

Axure was used for preparing mock-up screens. I could easily design wireframes for login, patient dashboard, doctor consultation, billing, and other pages. The drag-and-drop interface made it simple to visualize how the end screens will look before actual development.

Overall, using them gave me confidence in presenting the project requirements in both **diagrammatic form** and **screen design form**. It also gave a real-time feel of how analysts communicate with both business stakeholders and technical teams.

Document 9 – My Experience as a Business Analyst

As part of this live project, I worked as a Business Analyst and experienced different phases of the project lifecycle. Below is my role and learning in each phase:

1. Requirement Gathering

- We used the MoSCoW technique to prioritize requirements into Must Have, Should Have, Could Have, and Won't Have.
- During this phase, the client was not available for some period. As a BA, I sourced alternate point-of-contacts from the client's side to get information quickly.
- I validated requirements using the FURPS technique (Functionality, Usability, Reliability, Performance, and Supportability).
- Many requirements were found to be duplicated or repeated, so I immediately removed them to avoid confusion.
- We also used prototyping to help clients give more specific and clear requirements

2. Requirement Analysis

- I prepared UML diagrams to visually describe requirements.
- Activity diagrams were used to explain process flows in a step-by-step manner.
- Shared diagrams with the team, sometimes members disagreed. I considered their points and modified diagrams accordingly.
- Prepared both BRS (Business Requirement Specification) and SRS (System Requirement Specification) documents.

3. Design

- From Use Case diagrams, I prepared test cases.
- Coordinated with the client on design documents and solution documents
- Wrote both positive and negative test cases to ensure complete coverage
- Ensured no test case was missed, as missing could impact later development.
- Prepared test data required for testing activities
- Updated the RTM (Requirements Traceability Matrix) to make sure all requirements were mapped and tracked

4. Development

- Organized JAD (Joint Application Development) sessions with the technical team and client.
- Actively clarified queries raised by the technical team during coding.
- Handled situations where some team members disagreed or didn't cooperate. I had one-on-one discussions to explain the project impact and maintained a healthy environment.
- Referred to diagrams to guide development units.
- Conducted regular meetings with technical teams and the client. Since some members missed meetings, I recorded sessions and followed up with them individually.

5. Testing

- Prepared test cases directly from Use Cases.
- Performed high-level testing to validate key flows.
- Collected test data from the client.
- Updated the RTM with test coverage.
- Took sign-off from the client on testing results.
- Prepared the client for User Acceptance Testing (UAT).

6. Deployment

- Forwarded the final RTM attached with the Project Closure Document.
- Coordinated the preparation and sharing of End User Manuals.
- Planned and organized training sessions for users.
- Ensured that all users attended the training and understood the system.

Overall, my BA experience covered the end-to-end cycle from requirement gathering to deployment. I learned how important it is to be detail-oriented, flexible while managing clients, teams, and documents.

