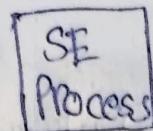


Date
13/2/24 Principles that Guide Process - I

- 1) Be agile
- 2) Focus on quality at every step
- 3) Be Ready to adapt
- 4) Build an effective team
- 5) Establish mechanisms for communication
- 6) Manage change
- 7) Assess risk
- 8) Create work products that provide value for others

Date
14/12/24 UML



RUP

Phases :- Inception, Elaboration, Construction, Transition

Architecture Models

- 1) Enterprise
- 2) System
- 3) Software

Diagrams in UML

1) Structure

- Component
- Class
- Object
- Composite

- Deployment
- Package

Hierarchical architecture chart

state models

2) Behavior diagrams / Dynamic models

- Use case
- Activity
- State Machine
- Communication
- Sequence
- Timing
- Interaction Overview

Pest chart

Architecture Modeling - Perspectives

conceptual physical

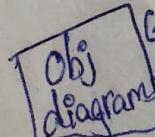
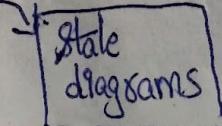
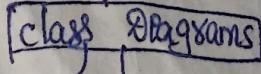
efficiency by Time and Space complexity

No. of Tangents input and output in numbers is scalability

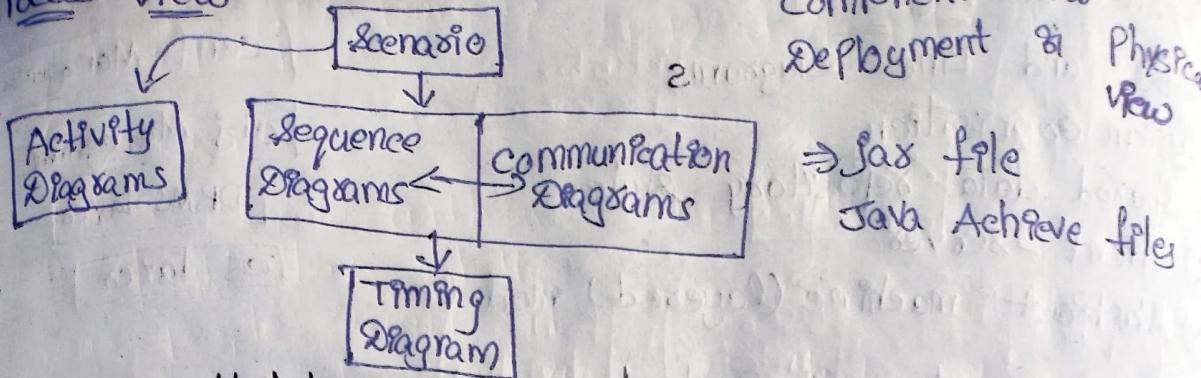
No. of times is executed is throughput.

Logical View

Package →
Diagrams



Process View



Use case Modeling

Use case Specification

name, short purpose, contact and change information
pre, post conditions

basic flow is the good day scenario

alternate scenarios

Inclusion, extension points to be identified

Activity Diagram

→ captures dynamic behavior (activity-oriented)

→ Purpose

i) Write down list of famous buildings in India?

ii) Taj Mahal, Charminar, Golconda fort, Agra fort, Lotus Temple, Golden Temple

Software architecture

Design process for identifying the subsystems making up a system and framework for sub-system control and communication is architectural design.

Q.P. is software architecture.

Characteristics

- 1) Performance
- 2) Security
- 3) Safety
- 4) Availability
- 5) Maintainability

Architectural Styles

1) Box and Line Diagrams

System organisation

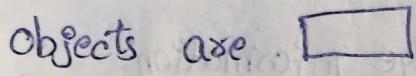
2) Shared data repository style.

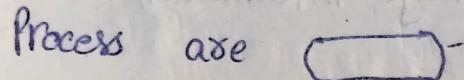
3) Client - server

4) Abstract machine (layered) model

Sub components → independent

Modules → dependent

Objects are 

Process are 

1) Identify fn req

2) non fn req

3) Special req and use case req and to traceability matrix

4) Identify set of obj, entities coming under boundary and business entities.

5) Identify set of attributes and operations of given classes.

6) Primary, 2nd and off stage actors. and find each actor use cases and communications inclusions & exclusions.

7) Find collaborators of entities using CRC card

8) Find activities of all real time entities and develop activity diagram and use case diagram

9) Prepare SRS of given problem stat.

10) Develop seq, collaboration of BCE class of scenario.

11) Perform forward and reverse engg of code.

12) Perform memory, pure coverage, performance of sample code.

conf Management

obj Management

DBMS layers

OS layers

Proverb

19) Laughter is the best medicine

Laughter reduces stress and improves immunity, leading to better health.

Ex: I think the best thing for you right now would be to spend sometime with people you can joke around with. Laughter is the best medicine, after all.

Date
2/2/24

05/08/24

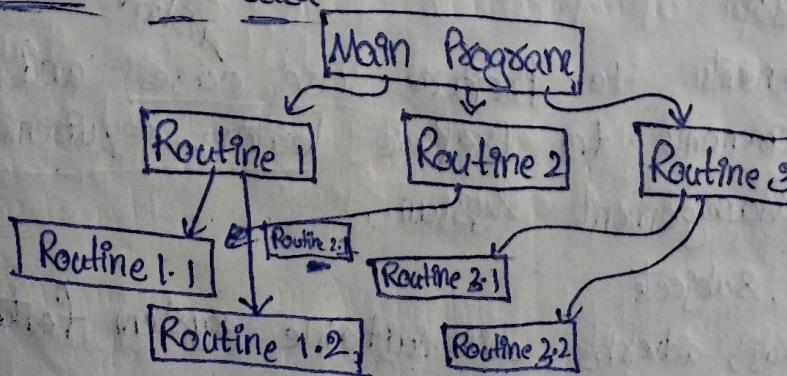
class diagram for Restaurant

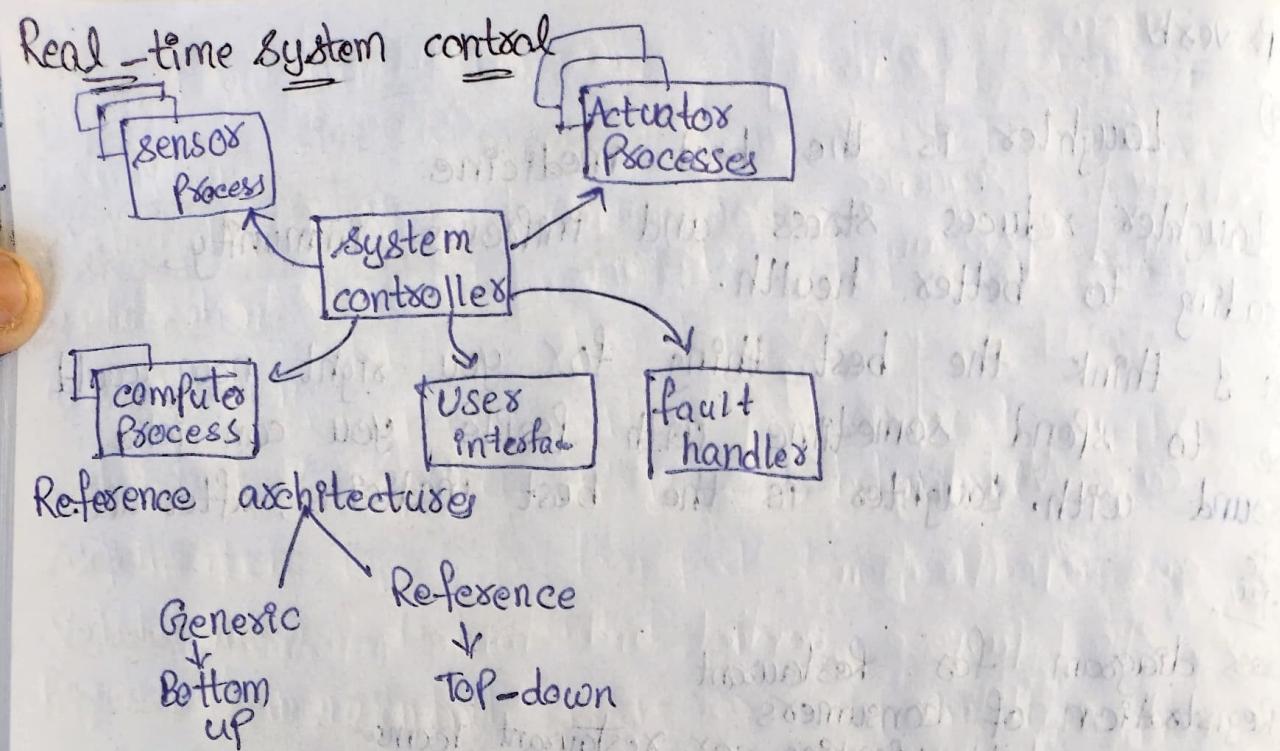
- i) Registration of consumers
- ii) Booking a table, parking car, restaurant rooms.
Party hall
- iii) Providing Multi cuisine food such as Indian, Italian, Chinese, Japanese foods.
- iv) Managing Home deliveries, Party hall bookings and event managements, through different Managers.
- v) Getting feedback from consumers ~~at~~ on each event.
- vi) complaint Register is common to everyone; a maintenance engineer will maintain it.
- vii) A Manager who takes role of discount offers at seasonal times and reporting annual income and balance sheet of credits and debits of company owner.

Control Styles

- i) Centralised control
 - ii) Event-based control
- call-return
Managers

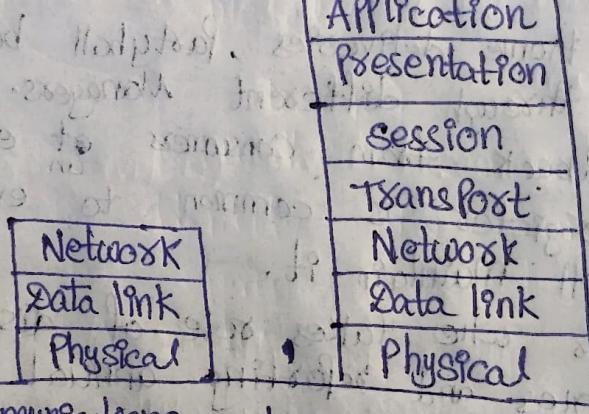
call-return model





OSI Reference Model

7	Application
6	Presentation
5	Session
4	Transport
3	Network
2	Data link
1	Physical



- 1) Palindrome
- 2) Reverse of string
- 3) Extracting substring
- 4) Area of rectangle

Find 10 fn requirements to prepare use cases and 10 non-fn requirements to prepare feature requirements of Restaurant Management System.

use case → obj, subjects
 feat req → adj, adverbs with suitable system features

Pure coverage

winxp → start → programs → virtual studio → c++ studio
run click

→ win32 application , enter project name, c++ source file and write code.

→ Build , exe file and extraction

→ Sample fn requirements to prepare use cases.

→ showcase available facilities

→ choosing their requirements

→ checking availability facilities

→ payment process

→ Generate Digital Receipt

Primary user → end user
secondary → Home Page
off stage → Bank
Intermediate

Non fn Requirements

1) Performance factors

2) Security

3) Safety

4) Scalability , adoptability , Quality , Mobility
System provides 'n' no. of users

1) Adj

2) Quick Response to Registration

3) Safe authentication methods provided such as password,
OTP, captcha

Quality
208005

Graphical errors

Bugs

Defects → is

failures → a unidentified

Risk

→ may identify when integrating

Traceability matrix traces fn & non fn requirements
either it is successfully facilitated by developer

Syntax

Schematic → Logical errors

A void analysed may identify that they throw away code.

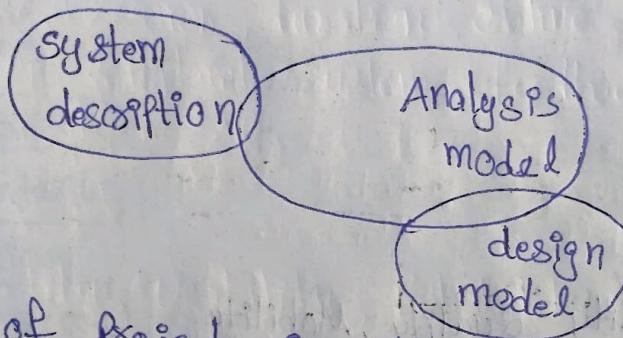
Date
24/2/24 Requirements Engineering

- 1) Inception
- 2) Elicitation
- 3) Elaboration
- 4) Negotiation

Statechart & Sequence ~~2s~~ → behavioral
⇒ Specification

Validation

Requirements management



Scope of Project is Identify b/w boundaries of starting and ending life span of Project until it move on to the next version.

Analysis classes

External entities, things, occurrences & events, Roles, organizational units, places, structures

1) Prepare Online Planetary using AR and VR. Identify problem stmt, list of stakeholders, purpose and beneficiary, scope of probelm & soln, operational goals, fn requirement, non fn req., at initial stage then trace feasibility matrix after negotiating and validating stakeholders requirements. Identify special req at least 4 glossary terms.

2) Find the activities of Travelling sp from home to cover all areas of sales within a given investment amount and within a day how much places visited

and how much saved as pocket money?

3) Take any famous novel and its 5 characters, develop a sequence diagram of a scenario as converted as into multimedia database for an story telling app?

Solutions

1) Pblm Stmt : We can't visit Planets directly.

sensors

camera

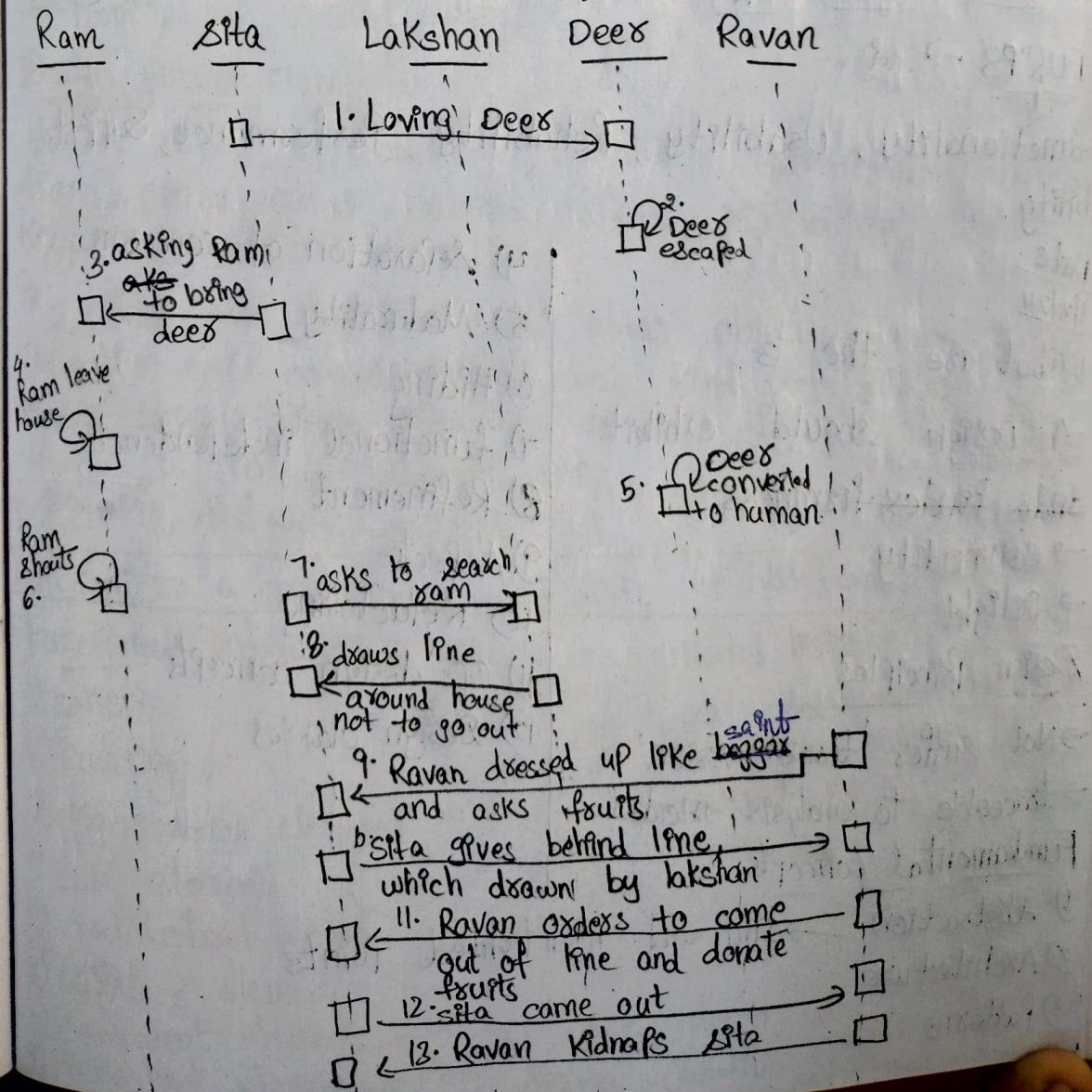
sound system

non-fn req → developers

fn-req → use case for user point of view

Hostel, Library, Play ground, Canteen, classroom.

characters :- Sita, Ram, Lakshan, Deep, Ravan



- Date
23/24
- Find online, offline collaborators of following entities:
- Library, books, journals, magazines, articles, E-book, audio-book, Musical lessons, Videos
 - Food, Serving bowls, mess hall, chef, cook, Manager, waste bins, donates, orphan home.
 - Play ground, Players, Playing rules, court, amphitheatre, referee, scoreboard, Team leaders, flag
 - Transport, types of transport, transport management, source, destination, dist, ticket, price.
 - Classroom, student, Professors, department, office, courses, session, duration, online class, mentor, monitor, complaint system, feedback.

Integrated Alaxam APP

FURPS → IMP

functionality, Usability, Reliability, Performance, Supportability.

Date

1/3/24

i) what are the 3

A Design should exhibit

And find expressiveness

→ Commodity

→ Delight

Design Principles

→ Not suffer 'tunnel vision'

→ traceable to analysis model

Fundamental concepts

- Abstraction → IMP and highlighted points
- Architecture
- Patterns

- Separation of concerns
- Modularity
- Hiding
- Functional independence
- Refinement
- Aspects
- Refactoring
- OO design concepts
- Design classes

Date, Page no:- 1 \Rightarrow get phone no

6/2/24
2 \Rightarrow find ISP

3 \Rightarrow Show offers

4 \Rightarrow check validity

5 \Rightarrow Recharge mobile

find suitable architectural design. Find useful component
 \Rightarrow Low coupling is principle that has min. level of dependency.

Over Design \rightarrow Obj

\rightarrow class

\rightarrow encap

\rightarrow Data hiding

\rightarrow Data abstraction

\rightarrow Inheritance

\rightarrow Polymorphism

\rightarrow Abstract classes

Architecture

\rightarrow Structural Properties

\rightarrow Extra-functional properties

\rightarrow Families of related systems

Design Pattern Templates

Pattern name, Applicability, Structure, Participants, Collaborations, consequences, Related patterns, Motivation, Also known-as, Intent

* Patterns are templates ex: class, object, Polymorphism, Separation of concern concern are kind of patterns.

* Any new pattern may be decided by its responsibility if it is not available yet

e.g. Singleton, Abstract friend class

Concern is a feature of behaviour of software.

Functional Independence

1) cohesion

2) coupling

Design Model elements

1) Data elements

2) Architectural elements

3) Interface elements

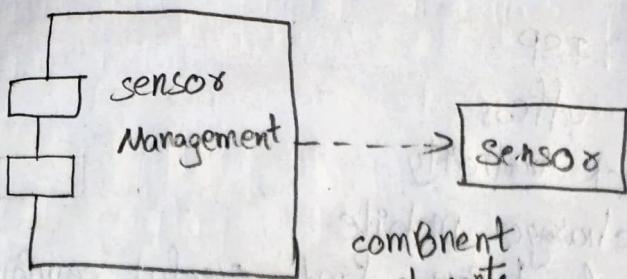
4) Component elements \rightarrow jar, .exe, .dcom, .del, activeX

5) Deployment elements \rightarrow proxy server, junction point, post no., port no., activeX

5) Nodes, links

Architectural elements sports, entertainment, games, non-profitable, embedded, AI, DL, ML, NLP, etc

Timer, calendar



Date
6/3/24 Interface Design
→ GUI

command based, Alexa, Siri

Golden rules

- 1) Place the user in control
- 2) Reduce the user's memory load
- 3) Make the interface consistent

User Interface Design Models

- 1) User Model
- 2) Design Model
- 3) Mental Model
- 4) Implementation model

Interface Design Steps

- Design interface obj's
- Define events
- Depict each interface state
- Indicate how the user interprets the state of the system

Testing

components :- Sample app, a code, Module

- 2) Test Plan
- 3) Test case
- 4) Testing Tools
- 5) Test Script
- 6) Test Input
- 7) Test results
- 8) Test Report

Design Issues

- Response time
- Help facilities
- Error Handling
- Menu & command labeling
- Application accessibility
- Internationalization

Testing

- 1) Black box → Testing on external environment how screen appears about input date
- 2) white box
 - ↳ clear box testing related bugs
 - ↳ logical flow of system (BVA)

Two ways of conducting testing

- Manual testing
- Automated testing

Date P13T24 Component

→ Multiple modules, deployable & replaceable part of a system.

Views

- object oriented
- Conventional view
- Process - Related

Design Principles

- Open - closed Principle (OCP)
- LSP → Liskov Substitution Principle
- DIP → Dependency Inversion Principle
- ISP → Interface Segregation Principle

Packaging Principles

- REP → Release Reuse Equivalent Principle
- CCP → Common closure principle
- CRP → Common Reuse Principle

Cohesion → unity

CCSDRTIE

- 3 CSDRTIE → coupling, content, Common, control, stamp, Data, Routine call, Type use, Inclusion & Import, External

coupling and cohesion measures quality of software system design.

Types of coupling

- Data coupling → good
- Stamp coupling
- Control coupling
- External coupling
- Common coupling → global data structures
- Content coupling → worst
- Sequential
- Temporal
- Communicational
- functional
- Data structures (Block chain structure)
- Interaction (Recursive calls)
- Component (Inheritance)

Artificial Intelligence

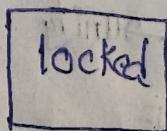
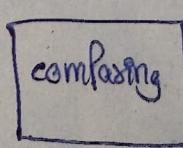
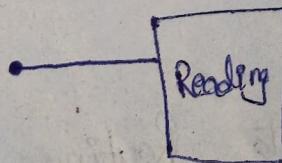
Date
13/3/24
Identify domain, analysis, potential, design, classes of story telling app?

Component levels

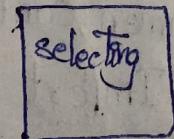
- Identify application & infrastructure of domain classes.
- Statechart is rounded rectangle box
- It has 2 compartments

Constraints

- Invariant
- Pre condition
- Post-condition
- Guard



cohesion (degree of collaboration).



Types of cohesion

- 1) Function → High calculating Total amount of transaction
 - 2) Sequence → purchase() → stock() → sell() → deliver()
 - 3) Communicational → update record in db and send to printer
 - 4) Procedural → calculate GPA, print student record, not change order
 - 5) Temporal → check OTP()
 - 6) Logical → logically related not functionally
 - 7) Coincidental → low, print next line and reverse the characters of a string
- Informational cohesion → Linked list, Blockchain
- Layer cohesion :- Uses, view / DB

Components of Software Process

Testing

Objective : Finding uncovered errors.

2 types of Testing : blackbox testing, white, box testing

two ways / approaches of testing : Manual, automated testing

Testing components are :-

- 1) Testing tools
 - 2) Sample applications
 - 3) Test Plan
 - 4) Test cases from use cases
 - 5) Test inputs
 - 6) Test results
 - 7) Test log
- exit → end

Date
15/3/24

System features

Gimp Pack Module classes

UI - View

even triggered - controller

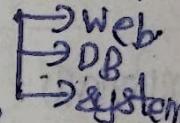
DB - Model

Online food Ordering System

Hotels, foodie, user, delivery boy, APP developer, Admin

Non-living

food types, order, menu, cancel, reserve, Reschedule



A class may be a noun, things - place & process,

Hospital Management System

doctor, nurse, patient, ward boy's, APP developer