LAB 1

Q. Develop a complete IEEE standard SRS document with several requirements

HOTEL MANAGEMENT SYSTEM

1. Introduction -

a. Purpose - This document outlines the functional & non-functional requirements for the Hotel Management System. It will serve as a guide for developers, testers & stakeholders to understand the expected features & performance of the system. The goal is to automate hotel operations & improvement customer service.

b. Document Conventions -

The document follows IEEE, are headings are bold & properly numbered for easy reference. Requirements are labeled as Functional, non-function & external interface.

c. Intended Audience & Reading Suggestions -

This SRS is intended for developers, testers, project managers & hotel administrator. Developer should focus on the requirements section. Testers should refer to functional & interfaces while managers & clients may focus on scope & user roles.

d. Project Scope -

The HMS will manage hotel operations such as reservations, room availability, customer check-in/out, staff scheduling, billing & reporting.

e. References
- IEEE 830 Software Requirements
- MySQL & Java Swing documentation
- APIs for online payments & email notifications

2. Overall Description

a. Product Perspective -
The HMS is a standalone application but may be extended to integrate with payment gateways & email services. It is intended for both local use & online access via a browser or client applications.

b Product Functions -
Main functions include room reservation, check-in/check-out, billing, etc. It also allows admin-level control for staff management & room configuration.

c. User Classes & Characteristics -
- Administrator: full access to all modules & data
- Receptionist: can manage bookings & billings
- Housekeeping Staff: can view & update room cleaning st.
- Customer: can book rooms online through a portal

d. operation environment -
The system will run on Windows & Linux platforms. It uses MySQL for the backend & Java on a web interface for the frontend.

e. Design & Implementation constraints -
The system should use only open-source techno
It must be scalable, secure, & support integrat
with external services.

f. User documentation -
User manuals & admin guides will be provided
which include system navigation, features, FAQs &
troubleshooting steps to help users interact with the
system easily.

g. Assumptions & Dependencies.
Assumes users have basic computer knowledge. Online
features depend on internet availability & third-party
APIs. The system also depends on the correct setup
of the hosting environment

3. Specific Requirements -
a. Functional Requirements.
- Users can book, modify & cancel reservations
- Receptionist manages check-ins & check-outs
- System generates & print bills
- Admin can add/edit rooms & view reports
- Email confirmations are sent after booking

b. non-functional Requirements -
- System should respond within 2 secs.
- 99.9% uptime is expected
- Secure login & encrypted data storage
- Must support at least 20 users at the same time.
- Interface should be user-friendly & fast.

Date 19/8/25
Page 4

cc. External Interface Req.
- GUI for admin, staff & customers
- Integration with payment
- Email server for sending notifications
- Optional barcode/ID Scanner support at reception

d. Appendices
- Glossary of terms used in HMS
- Screenshots/mockups of interfaces
- Use Case, ER, & DFD diagrams
- Sample test cases for each feature

Meeting
19/8/25

CREDIT CARD PROCESSING (SRS)

1. Introduction

a. Purpose -

This system securely processes credit card payments, including validations, authorization, fraud checks & transaction recording. It ensures fast, reliable & compliant payment handling.

b. Document Conventions

Follows IEEE 830 format using clear section numbering & bold labels like Functional, Non-functional, External interface requirement.

c. Intended Audience & Reading Suggestions

For developers, testers, security analysts, & stakeholders. Each should focus on relevant sections like requirements, security features & performance criteria.

d. Project Scope

The system handles online & offline credit card transactions, detects fraud, validates card details & provides transaction history to banks & merchants.

e. References

- IEEE 830
- PCI DSS Guidelines
- VISA / MasterCard API Docs
- ISO Smart Card Standards

2. Overall Description -

a. Product Perspective -

Acts as middleware between merchants, banks & card networks. Integrates with payment gateways & merchant platforms for secure transactions.

b. Product functions -

Performs card validation, payment approval, fraud detection, transaction logging & reporting. Admins can review logs & manage merchant profiles.

c. User classes -

- Merchant: Initiates & monitors transactions
- Cardholder: Makes payment
- Admin: Manages users & logs
- Security Team: Monitors fraud alerts

d. Operating Environment -

Runs on secure servers with HTTPS, supports web / mobile / POS systems, & uses encrypted connections (SSL / TLS).

e. Constraints -

Can't store sensitive card data without compliance. Must integrate with API's from banks & card networks & ensure high performance.

f. User documentation -

Includes guide for merchants, admin panels, API integration & security practices for monitoring & compliance.

9. Assumptions —
Requires internet, valid card network APIs, verified users, & third-party fraud detection systems for smooth operation.

3. Specific Requirements

a. Functional requirements
- Validate and details & approve/reject payments
- Record transactions & notify users
- Detect & block suspicious activity
- Provide summary reports

b. Non-functional requirements
- Response time less than 3 secs for transactions
- 24/7 System availability with 99.99% uptime
- End to end encrypted data transactions
- System must comply with PCI-DSS standards
- Scalable to handle peak load traffic

c. External Interface Requirements
- Integration with banking / payment gateway APIs
- GUI dashboards for merchants & admins
- Supports POS systems, web, & mobile interfaces
- Communication via REST APIs & Secure sockets

d. Appendices —
- Glossary
- Use case & Sequence diagrams for transactions
- API Request/response formats examples
- Sample test cases & validation criteria

Reviewed 19/8/25

Stock Maintenance system

1. Introduction

Purpose: To manage & track stock digitally with real time updates.

Convention: "Shall used for requirements

Audience: Managers, Developers, Testers, Staff

Scope: Automates stock tracking, reports & alerts

References: IEEE Std. 830, Inventory guide

2. Overall Description

Perspective - Standalone, future integration with POS / ERP

Product functions - Add / update / delete items, stock alerts, reports, search / filter, role-based access

Users: Admin (full control), staff (update, report), viewer (view only)

Environment: Windows / Linux, MySQL / SQL DB, Web / Desktop

Documentation - User manual, online help

Assumptions: Users are computer literate, data entered correctly

Date 26/8/25
Page

3. Specific Requirements -

1. Functional requirements -
The system manages stock by adding, updating, deleting items, showing real-time availability, generating low-stock alerts, reports, & keeping transaction logs with role-based access.

2. Non-functional Requirements -
It ensures fast response less than 2 sec scalability up to 10,000 items, security with authentication / encryption, user-friendly design, high uptime (99.5%) & easy maintainence.

3. External Interface Requirements -
The system provides a GUI with dashboards, connects to databases like MySQL, works on standard OS / hardware, communicates via LAN & ~~Support~~ & support future POS / ERP integration.

4. Appendices
Glossary - Stock, Report, Threshold

Future enhancement - Mobile app, barcode scan, auto supplier ordering.

Passport

1. Introduction -

Purpose: The system automates passport application, verification & issuance online to reduce delays & manual work.

Document convention: All requirements use the keyword shall; categories are functional, non functional & subparts

Intended Audience & Reading Suggestions
Applicants, government officials, developers, testers & managers each should focus on their relevant sections

Project scope: The system handles online applications, payments, tracking, verification & final passport delivery in a secure manner

References: IEEE Std 830, official govt. E-Governance policies & existing passport guidelines

2. Overall Description

Product Perspective: A web-based solution, replacing manual process; designed for integration with govt databases.

Date __/__/__
Page _____

Product Functions : Users apply online, upload documents, pay fee, schedule appointments, track status, & officials issue passports.

user classes & characteristics : Applicants submit requests, verifiers check & approve documents, & admins control the system.

operating environment - clients on windows/linux servers with MySQL / oracle DB accessible via common web browsers.

Design & Implementation constraints : must follow govt. data protection, security standard & handle large user traffic.

User documentation : Provides user manual, online help & FAQ's for applicants & officials

Assumptions & Dependices : user have internet access & valid documents, system depends on secure payment & verification services.

3. Specific Requirements.

1. Functional Requirements : Applicants can register, apply, upload, pay & track, applications, while officials verify, update, & issue passports.

Date __/__/__
Page

2. Non-Functional Requirements -
The system shall be fast (< 2 secs response), secure, reliable, scalable, & easy to use 24/7.

3. External Interface Requirements -
Provides browser-based GUI, connects to database, integrates with payment & verification system over Internet / LAN

4 Appendices
Glossary : Applicant, citizen user, Verifier, Post office, Status, Current stage of passport process

Future Enhancement - Integration with mobile app, biometric verification, & AI based fraud detection

LIBRARY MANAGEMENT SYSTEM

1. Introduction

1 Purpose - The system automates library operations including book issue, return, cataloging, & members management to save time & reduce errors.

2 Document Conventions - Are the requirements use the keyword shall, categories are FR, NFR, IR

3. Intended Audience & Reading suggestions - Librarians, students, faculty, developers, testers & project managers should focus

4 Project scope - The system manages book records, member accounts, transaction & queries reports while ensuring easy search & access.

5. References : IEEE std 830, library automation standards & guidelines

2. Overall Description -

1 Product Perspective - A standalone or web-based app replacing manual registers, with options for future integration with digital libraries.

Date
Page

2. Product functions: Allows adding / removing books, issuing / returning books, managing members, generating reports, & providing search functionality

3. User classes & characteristics - Librarians manage books & members faculty & students borrow books & admin oversee the system

4. operating environment: Works on Windows / Linux with MySQL & accessible via web browser.

5. Design & implementation constraints - must ensure data consistency, prevent multiple issue of the same copy.

6. User documentation: Includes manual for staff, FAQs for students & training guides

7. Assumptions & Dependencies - Users have valid library accounts, System depends on reliable database

3. Specific Requirements -

Functional Requirement -

The system shall allow book issue

Date __/__/__
Page _____

return. Manage due dates & fines, maintain member accounts & generate reports.

2. Non-Functional Requirements - It shall provide response within 2 sec, ensure security, maintain 99.5% uptime & offer a user friendly interface.

3. External Interface Requirements - Provide a GUI with search & enroll & menus. connects to relational database & communicates over LAN/ Internet if online.

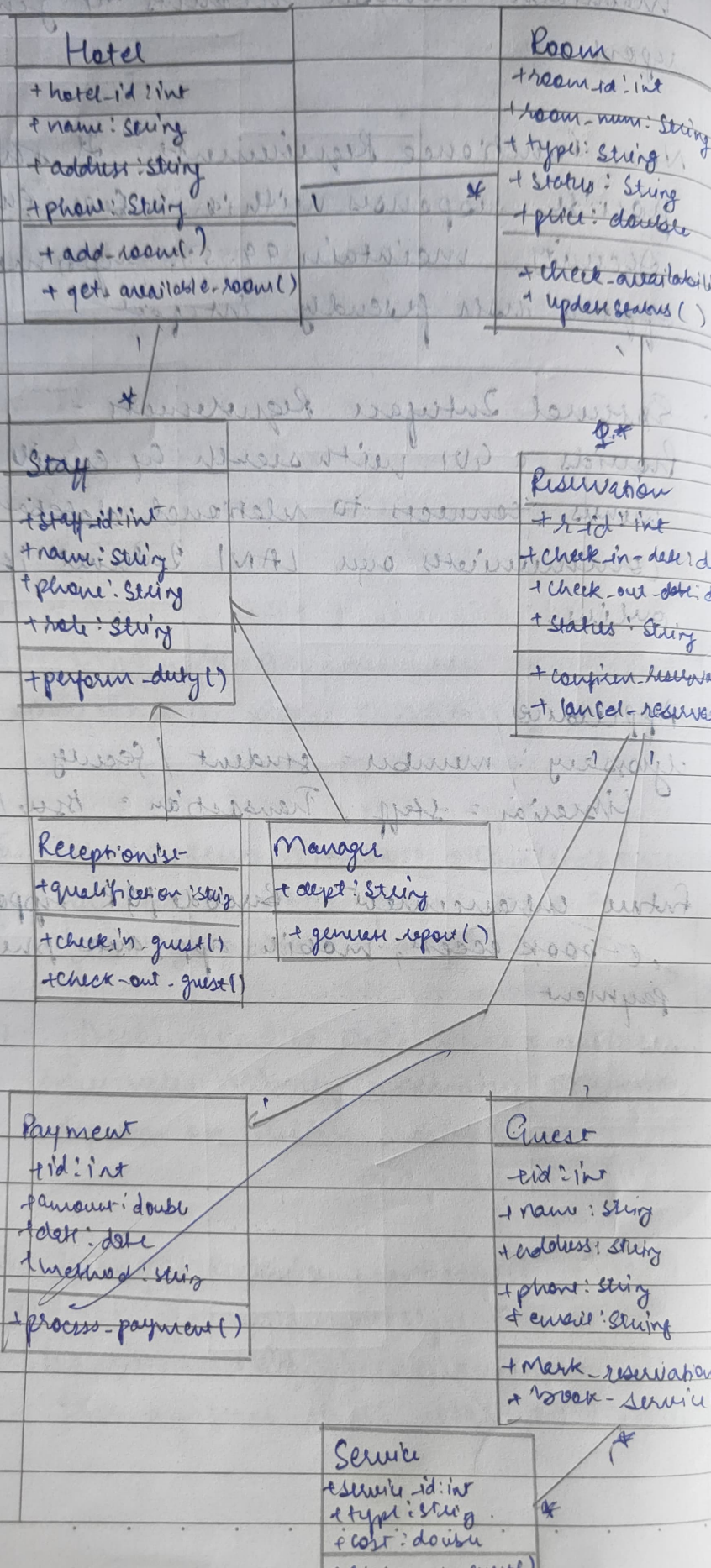
4. Appendices
Glossary: member = student / faculty, Librarian = staff, Transaction = issue / return

future enhancement: Barcode / QR support, e-book access, mobile app, online fine payment

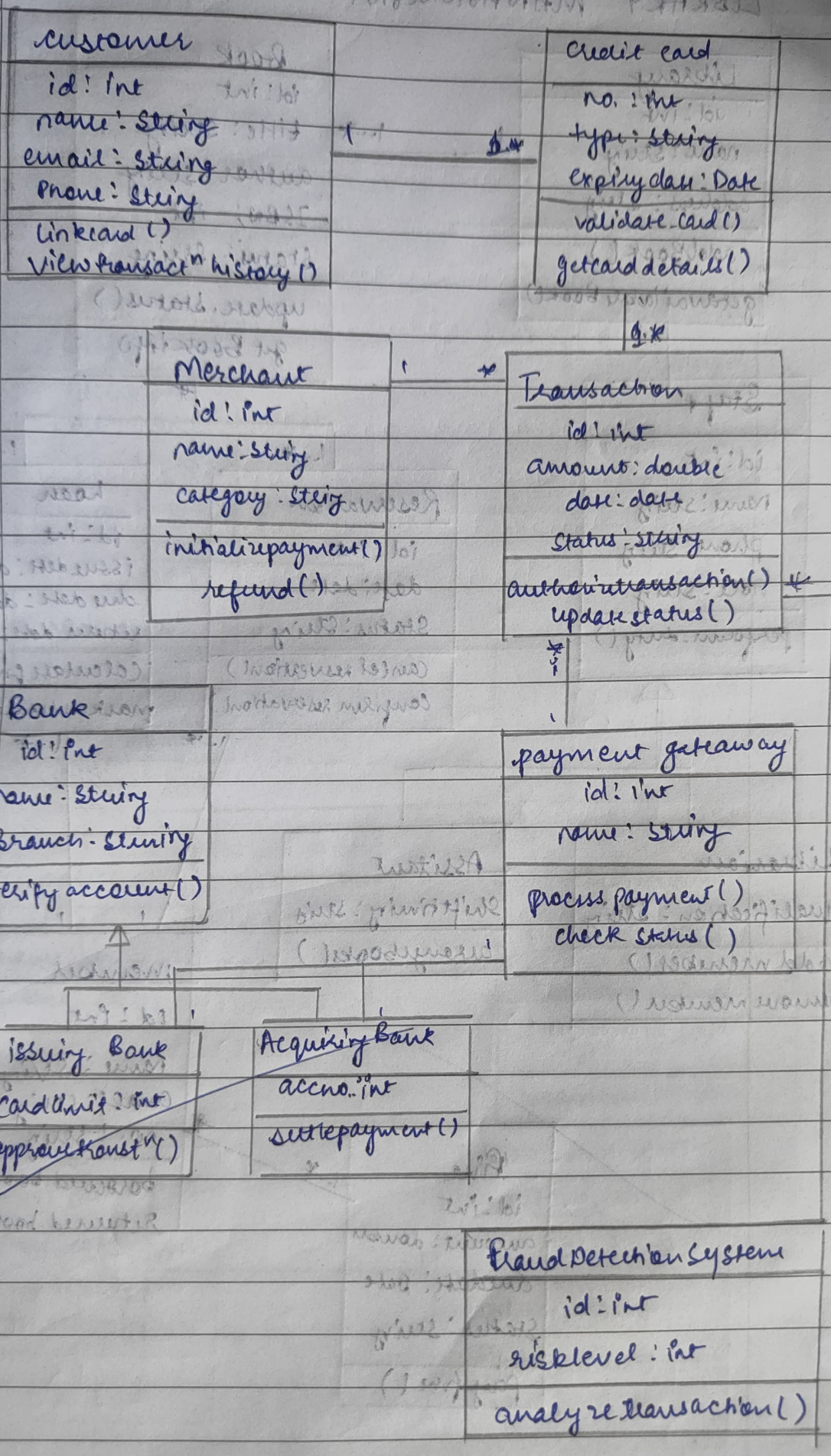
Date 2/9
Page

CLASS DIAGRAM

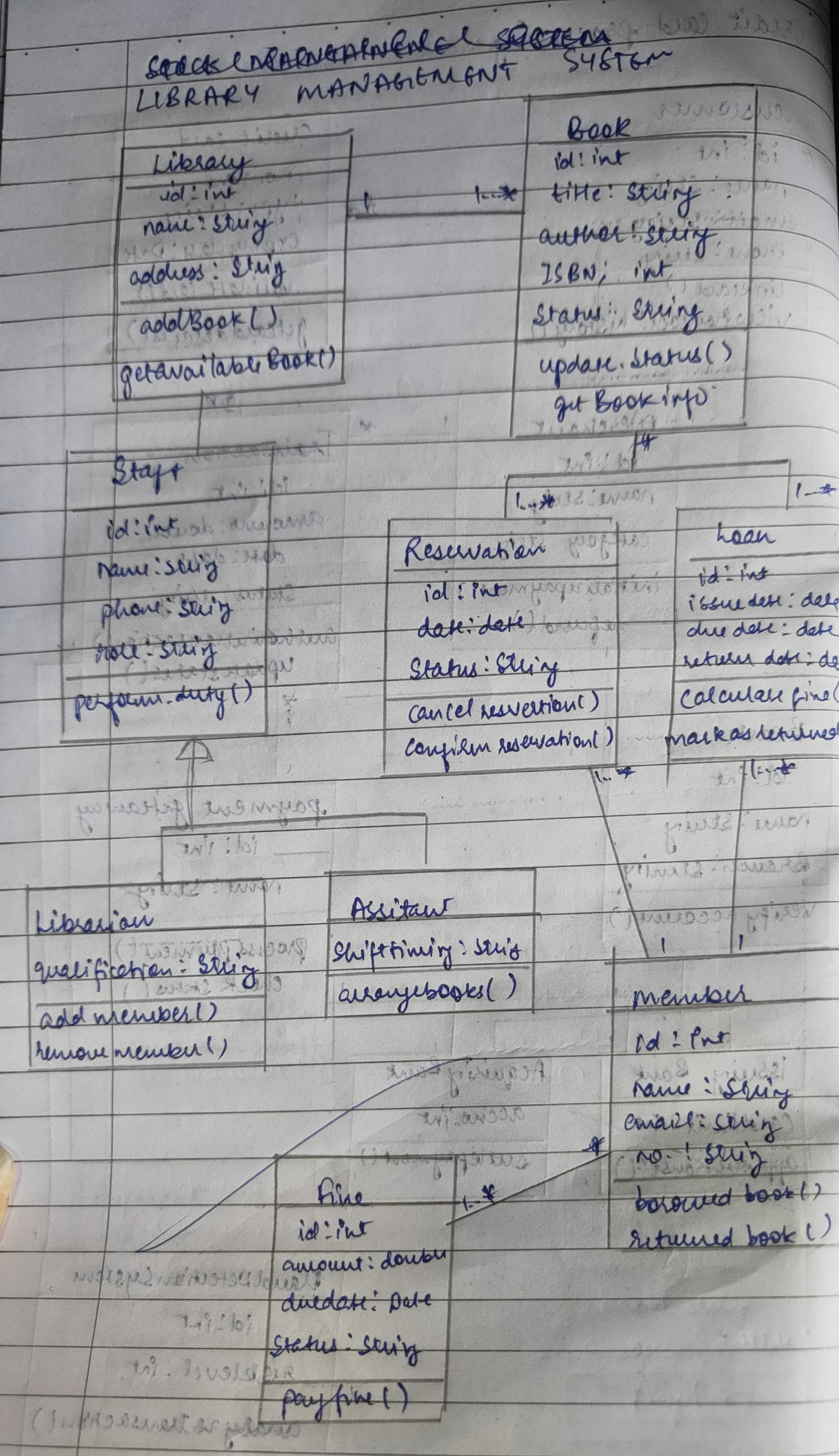
I Hotel Management system



Credit card processing

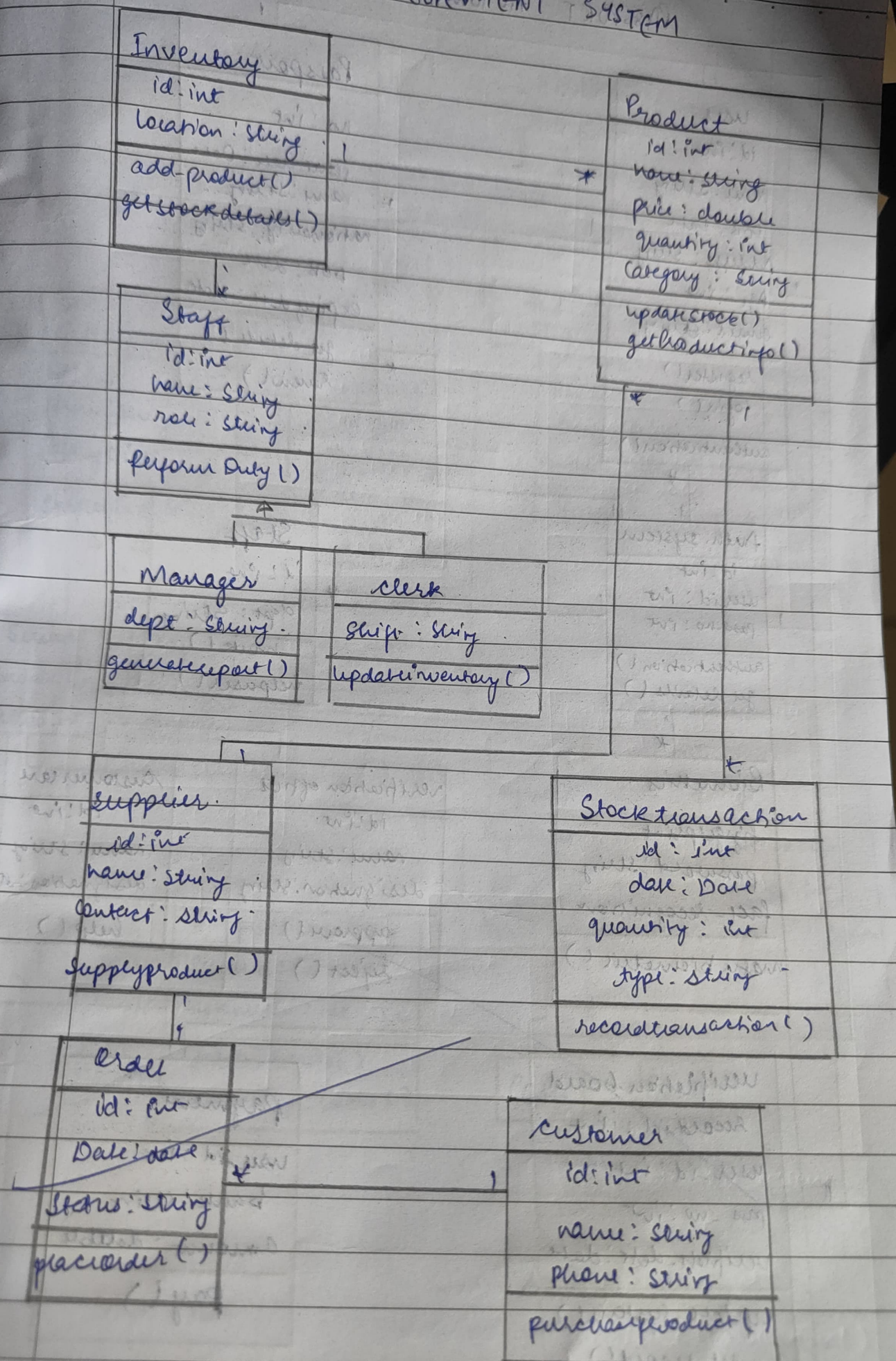


LIBRARY MANAGEMENT SYSTEM



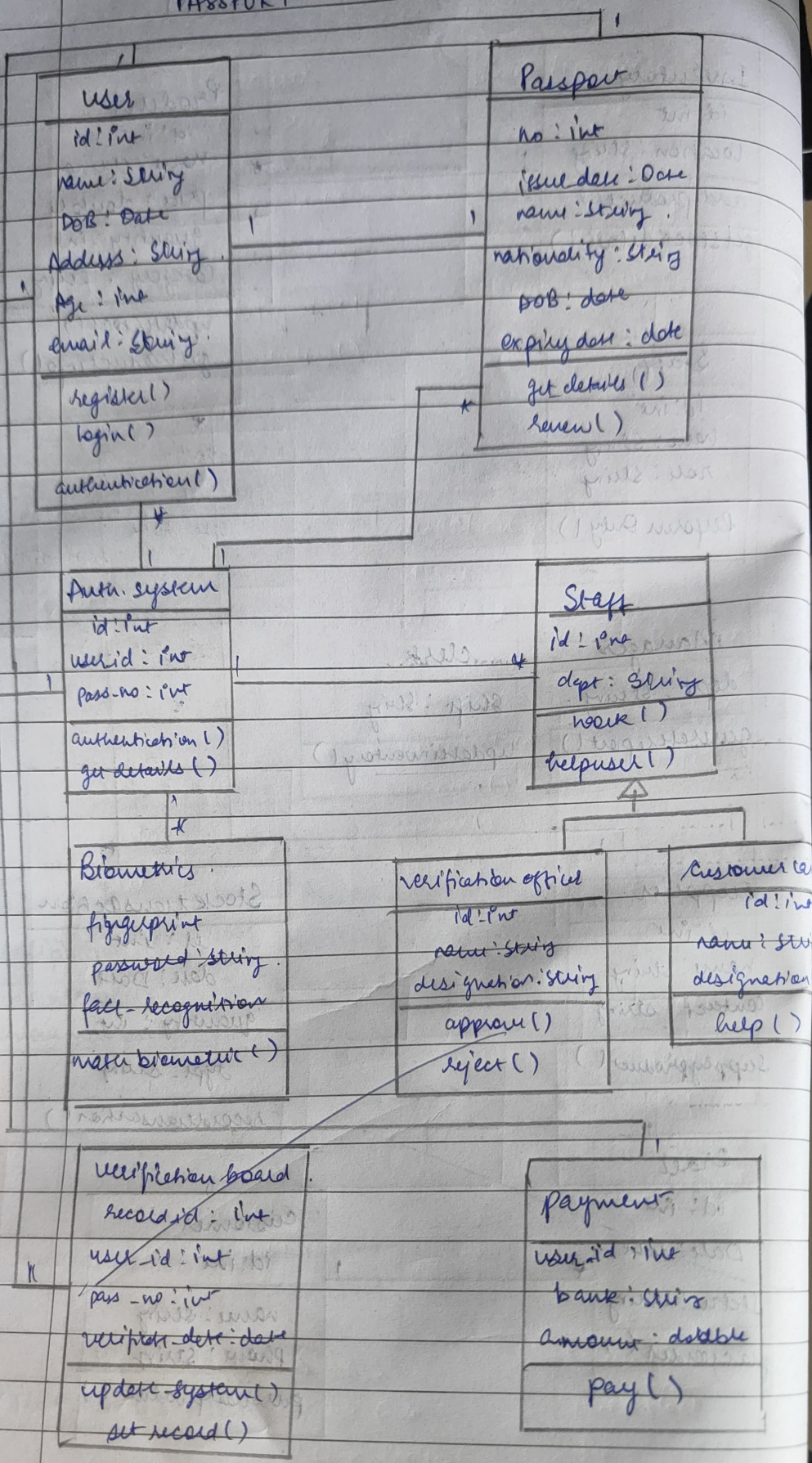
Date 2/9/25
Page

STOCK MANAGEMENT SYSTEM



Date 2/9/12
Page

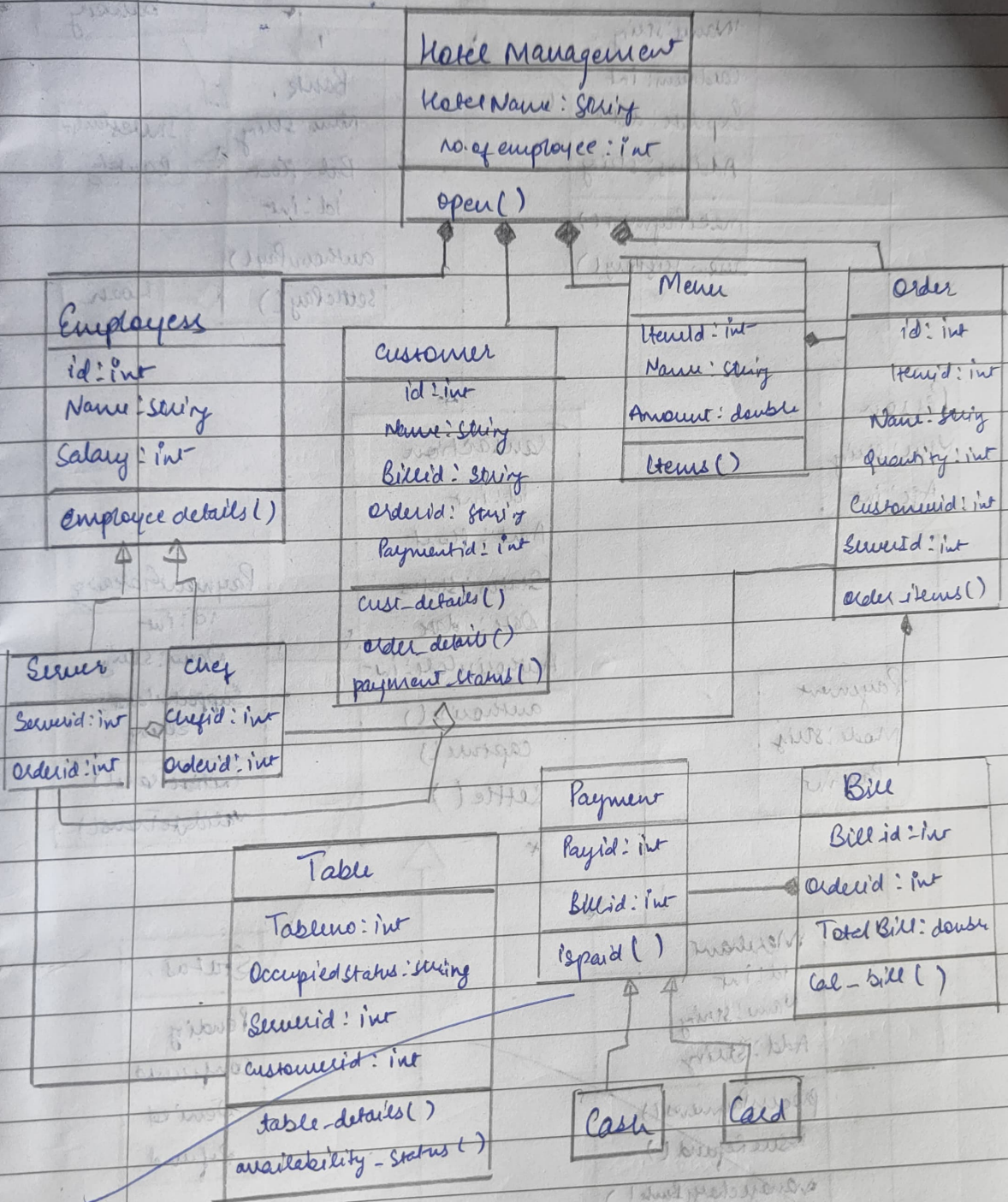
PASSPORT AUTOMATION SYSTEM:



Date __/__/__
Page:

HOTEL MANAGEMENT SYSTEM

Advanced class Diagram



Date __/__/__
Page _____

CREDIT CARD PROCESSING