2. Credit card processing:

Srs document:

(vi) online booking integration

4. interface requirements:
   (i) graphical user interface (gui) for hotel staff
   (ii) web based interface for customers
   (iii) integration with external booking platforms

5. performance requirements:
   (i) must handle atleast 1000 users
   (ii) give response in time for users
   (iii) update information on room availability

6. design constraints:
   (i) system should working in existing hardware of pc.
   (ii) software should be able to adapt future changes.

7. non functional attributes:
   (i) security: the information of customers, staff should be encrypted the payment details should not be shared
   (ii) scalability: it should be able to be used in different branches of hotel
   (iii) reliability: information should be updated in all branches

8. preliminary schedule and budget:
   (i) time required: system should be developed in 6 months

Software requirements specification for credit card processing

1. introduction:

1.1 purpose:

the purpose of this document is to outline the functional and non functional requirements for a secure credit card processing system. which facilitates transactions between merchants and card holders

1.2 scope:

the system will support multiple card types (visa, mastercard) and offer real time processing, fraud detection, and reporting tools. it is designed for integration with e-commerce platforms

1.3 overview:
the credit ~~card~~ card processing system will ensure smooth payment transaction for both online and inperson purchase

2. general description:

2.1 product perspective:
the system will work as an intermediary between the merchant and the credit card company securely processing and verifying payments

2.2 user characteristics:
user will include merchants, card holders ~~and sys~~ and system administrators

2.3 system constraints:
it must meet pcc dss and ensure secure transaction handling

3. functional requirements:
(i) card authorization
(ii) ~~paye~~
(ii) payment gateway integration
(iii) fraud detection and prevention
(iv) transaction reporting and auditing
(v) charge back and refund handling

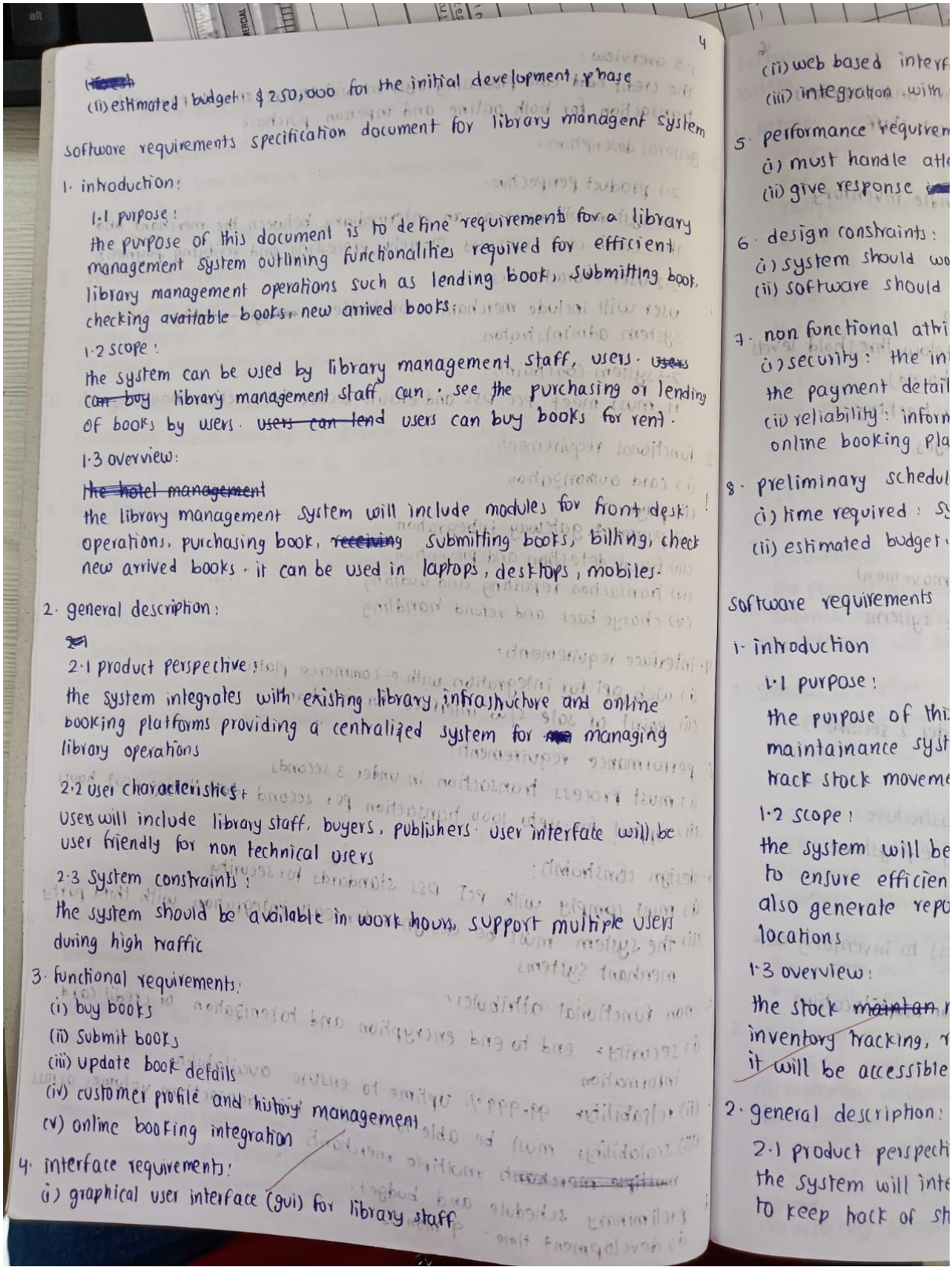
4. interface requirements:
(i) web api for integration with e-commerce platforms
(ii) point of sale s/w interface for merchants

5. performance requirements:
(i) must process transaction in under 3 seconds
(ii) support for upto 1000 transaction per second ~~dur~~ during peak hours

6. design constraints:
(i) must comply with PCI DSS standards for security
(ii) the system must be designed for easy integration with third party merchant systems

7. non functional attributes:
(i) security: end to end encryption and tokenization of credit card information
(ii) reliability: 99.999% uptime to ensure availability
(iii) scalability: must be able to handle high transaction volumes across ~~multiple merchants~~ multiple merchants

8. preliminary schedule and budget:
(i) development time: 9 months



(ii) estimated budget: $250,000 for the initial development phase

Software requirements specification document for library managent system

1. introduction:

1.1 purpose:

the purpose of this document is to define requirements for a library management system outlining functionalities required for efficient library management operations such as lending book, submitting book, checking available books, new arrived books.

1.2 scope:

the system can be used by library management staff, users. ~~users can buy~~ library management staff can see the purchasing or lending of books by users. ~~users can lend~~ users can buy books for rent.

1.3 overview:

~~the hotel management~~

the library management system will include modules for front desk operations, purchasing book, ~~receiving~~ submitting books, billing, check new arrived books. it can be used in laptops, desktops, mobiles.

2. general description:

2.1 product perspective:

the system integrates with existing library, infrastructure and online booking platforms providing a centralized system for managing library operations

2.2 user characteristics:

users will include library staff, buyers, publishers. user interface will be user friendly for non technical users

2.3 system constraints:

the system should be available in work hours, support multiple users during high traffic

3. functional requirements:

(i) buy books

(ii) submit books

(iii) update book details

(iv) customer profile and history management

(v) online booking integration

4. interface requirements:

(i) graphical user interface (gui) for library staff

(ii) web based interf

(iii) integration with

5. performance requirem

(i) must handle atle

(ii) give response

6. design constraints:

(i) system should wo

(ii) software should

7. non functional attri

(i) security: the in

the payment detail

(ii) reliability: inform

online booking pla

8. preliminary schedul

(i) time required: sy

(ii) estimated budget

Software requirements

1. introduction

1.1 purpose:

the purpose of thi

maintainance syst

track stock moveme

1.2 scope:

the system will be

to ensure efficien

also generate repo

locations

1.3 overview:

the stock ~~maintan~~

inventory tracking,

it will be accessible

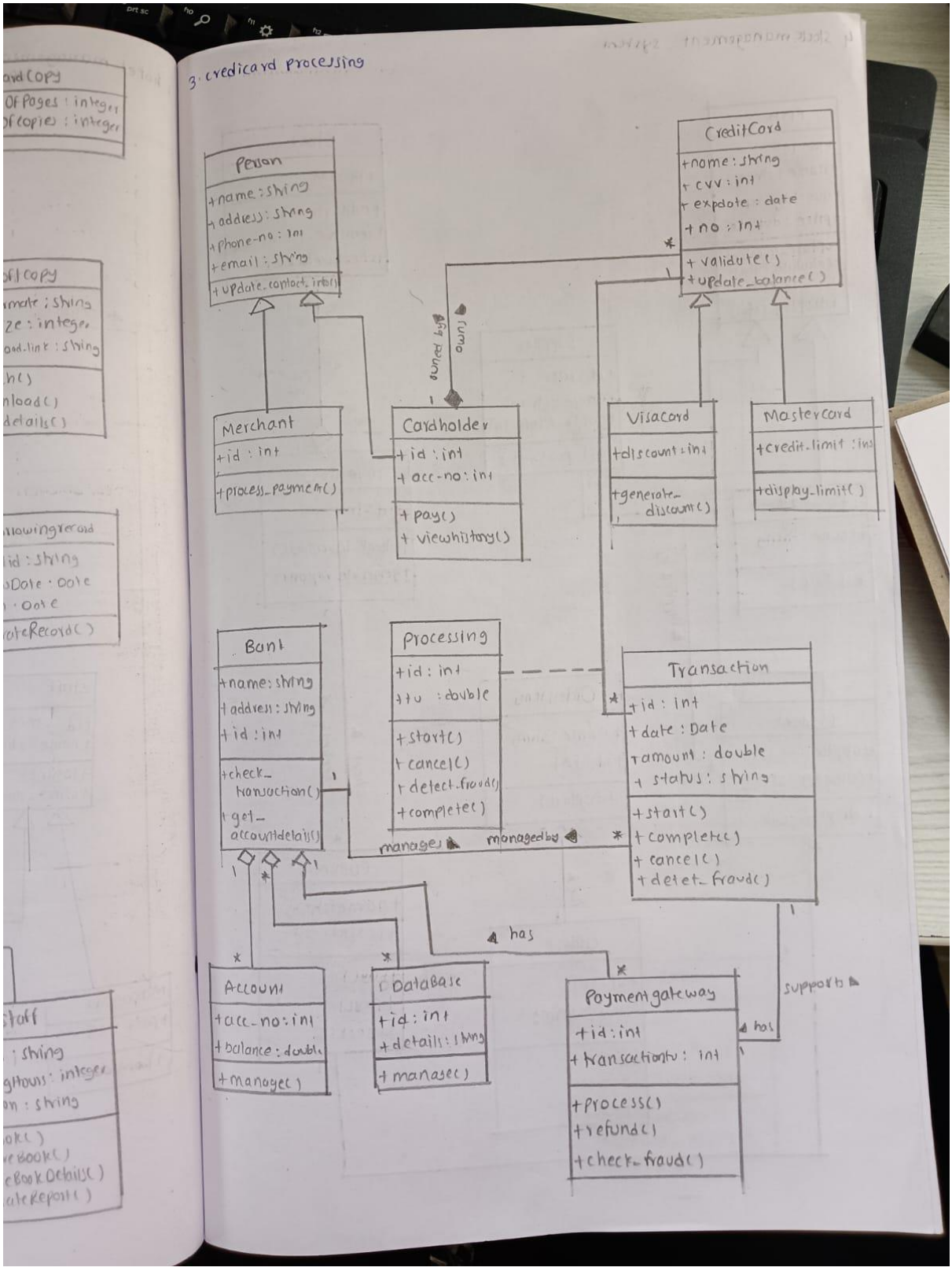
2. general description:

2.1 product perspecti

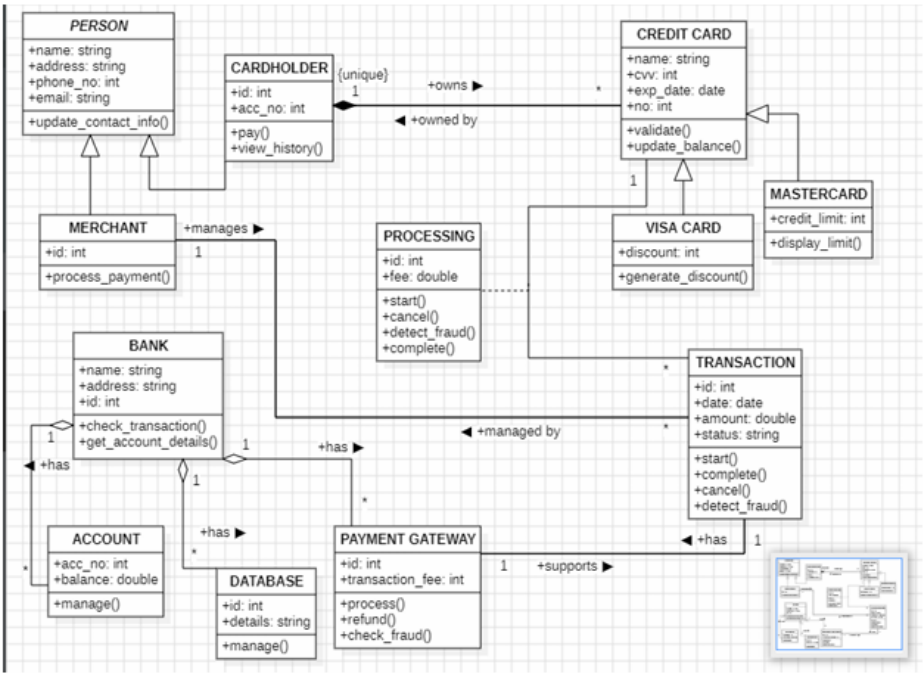
the system will inte

to keep track of st

Class diagram:



Class diagram star uml:


PERSON
+name: string
+address: string
+phone_no: int
+email: string
+update_contact_info()
CARDHOLDER
+id: int
+acc_no: int
+pay()
+view_history()
{unique}
1
+owns ▶
*
◀ +owned by
CREDIT CARD
+name: string
+cvv: int
+exp_date: date
+no: int
+validate()
+update_balance()
MASTERCARD
+credit_limit: int
+display_limit()
MERCHANT
+id: int
+process_payment()
+manages ▶
1
PROCESSING
+id: int
+fee: double
+start()
+cancel()
+detect_fraud()
+complete()
VISA CARD
+discount: int
+generate_discount()
BANK
+name: string
+address: string
+id: int
+check_transaction()
+get_account_details()
TRANSACTION
+id: int
+date: date
+amount: double
+status: string
+start()
+complete()
+cancel()
+detect_fraud()
◀ +managed by
+has ▶
◀ +has
ACCOUNT
+acc_no: int
+balance: double
+manage()
DATABASE
+id: int
+details: string
+manage()
PAYMENT GATEWAY
+id: int
+transaction_fee: int
+process()
+refund()
+check_fraud()
+supports ▶