Storage Vault

Software Specification and Requirements Document

Group1

Tilat Jahan

Rousol AI Goboori

Sri Devi Rella

CONTENTS

OVERVIEW

This document specifies the software specifications and requirements for five main components in our Storage Vault. Based on the different view perspectives of lenders and consumer's we designed our five components: 1) Storage Space Lending and Usage, 2) Peer Collaboration, 3) Market Place, 4) Ratings and Reviews, 5) Protection Plans. Each component consists of user interviews, user stories, user cases with flow diagrams that describe how each service work, class, and database diagram.

COMPONENT 1: STORAGE SPACE LENDING AND USAGE

1.1 INTERVIEWS

User Interview #1: Small Business Owner

- Looking for a system that can provide security for the data. Securely store, send, and sync data—without restrictions.
- Access and share all my files and documents in a few simple clicks.
- Should have Limitless storage, a convenient access facility and the ability to send large files—without clogging email and help accelerate productivity.
- Should be able to retrieve the data from all over the world. The system should give a single, secure point of access to all data, regardless of environment or endpoint.

User Interview #2: Teenager

- Needs to be able to store large media files (Images, Videos and Songs)
- Should have a minimal network latency or response time to retrieve and play these media files.
- Need to have a data backup in case of data loss to retrieve the lost data.

User Interview #3: Small Firm

- Should be able to use a minimum amount of storage space that is lying unused.
- Needs to have a payment process that is flexible and gives options to pay through various sources.
- Needs an application that keeps track of the utilization of your lent storage device space usage, on daily or monthly basis.

1.2 USER STORIES

User Story #1: A Small Company

- A small company spends much money on buying new servers, installing them, and maintaining them on a regular basis. While the manufacturer claims their newest
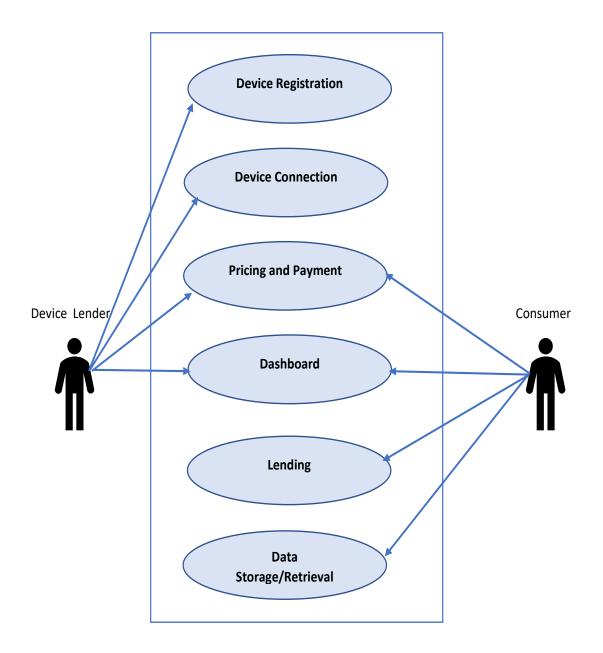
system is reliable and needs no ongoing maintenance, something always goes wrong. The company is looking at some cloud storage options that can eliminate the need for expensive in-house systems and possibly save some dollars in the long run.
- They have concern regarding the loss of valuable data due to server failures, employee mistakes, natural disasters, and more.

User Story #2: A Small Business Owner

- He wants to lend his storage device that he is not using and wants to earn some benefits from it.
- He logs into the system to check if the amount of available storage satisfies the minimum requirements for lending.
- He wants to see the monetary benefits he will gain by lending that storage space and decides upon the period to lend his device.

User Story #3: A Student

- He wants to be able to store and organize the study documents of my four years of graduation on a central storage that takes quiet an amount of space which will not be sufficient on his device and he does not intend to use many external hard drives to save the data.
- He does not want to spend a fortune for this, but with a little investment he is looking for a secure storage space.
- He also wants to be able to access these documents through web from anywhere in the world and wants to give access to multiple users for certain documents.
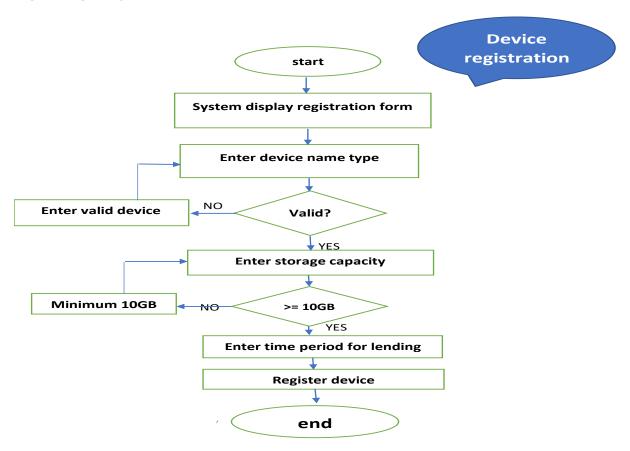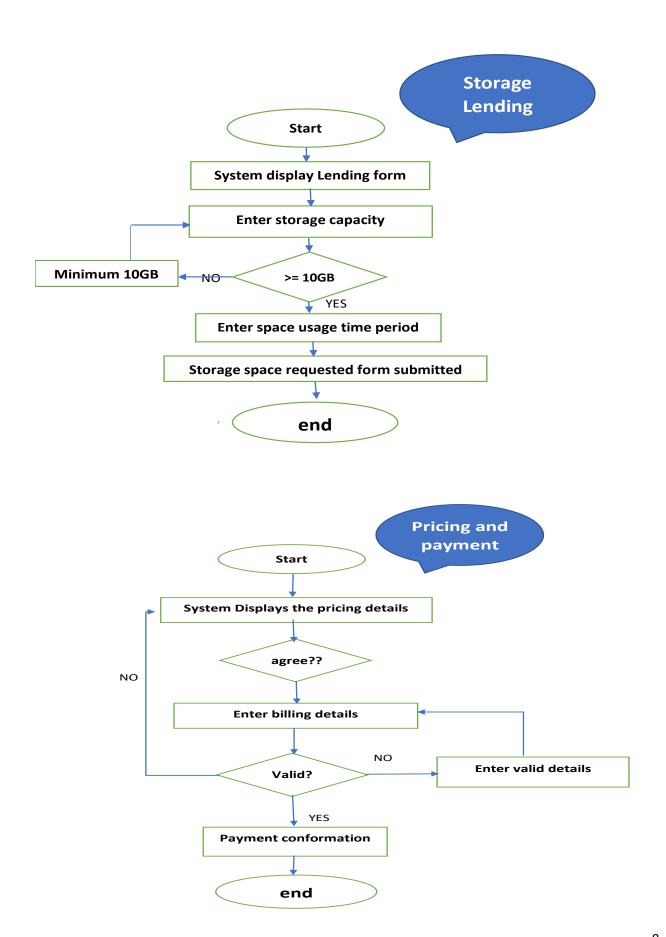
Title:	Device Registration and Storage Lending
Description:	Consumers utilizing the storage space which is lent by the lenders who have unused extra space.
Primary Actor:	Consumers, Lenders
Preconditions:	[Describe the state the system is in before the first event in this use case.] • The lender needs to register himself with his personal or business details. • The Consumer needs to register himself with his personal details.
Postconditions:	• The lender registers his device with the following details: device type, storage capacity, time period. • The consumer request for storage space with the following details: memory space required, time period.
Requirements:	• Device with network capability ex. NAS • Minimum storage capacity 10 GB • A well-set Internet connection. • Always connected to the network during the lending period.
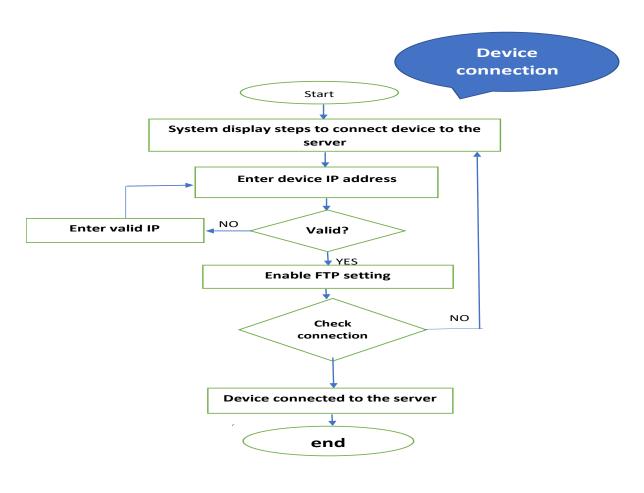
Title:	Pricing and Payment
Description:	Lenders are providing with a quote for lending their memory space and Consumers are provided with a quote for utilizing the space required. Based on the contract a billing options will be provided to both the lenders and the consumers
Primary Actor:	Consumers, Lenders
Preconditions:	• The Lenders needs their device to be registered with the vault with the device name, storage space and time for lending. • The Consumer is registered with vault with the required space request and time for the space requested.
Postconditions:	• Once the payment conditions are approved consumer, he is provided with the data storage and retrieval options through the web service. • After a mutual agreement with the lender, he is provided with the payment plan and provided with the details to connect to the remote server.
Requirements:	• Provide Bank account details. • Debit/Credit card Details.
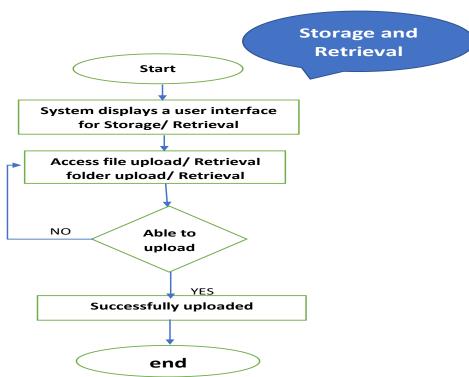
Title:	Device Connection and Data Storage/Retrieval
Description:	The lender connects his device to the remote server and the consumer accesses the storage space through the web services.
Primary Actor:	Consumers, Lenders
Preconditions:	The lender and consumer agree for the payment policies and registered with their payment account details.
Postconditions:	The lenders and consumers are provided with a Dashboard where they can access their device utilization, storage, and payment details.
Requirements	The lender Device should have an IP address and compatible with network through FTP. A registered device available in the vault to be able to provide storage space to the consumer. An internet Access with minimum mbps for access.

FLOW DIAGRAMS

Storage Lending

Start

↓

System display Lending form

↓

Enter storage capacity

↓

>= 10GB —NO→ Minimum 10GB

↓ YES

Enter space usage time period

↓

Storage space requested form submitted

↓

end

Pricing and payment

Start

↓

System Displays the pricing details

↓

agree??

↓ (NO loops back)

Enter billing details ←

↓

Valid? —NO→ Enter valid details

↓ YES

Payment conformation

↓

end

Device connection

Start

System display steps to connect device to the server

Enter device IP address

Valid?

Enter valid IP — NO

YES

Enable FTP setting

Check connection

NO

Device connected to the server

end

Storage and Retrieval

Start

System displays a user interface for Storage/ Retrieval

Access file upload/ Retrieval folder upload/ Retrieval

Able to upload

NO

YES

Successfully uploaded

end

Class Diagram

Device Connection
IP Address: String
Settings: List<String>
Connect Device ()

Lender
First name: String
Last name: String
Address: String
Email: String
Device Details: Device registration
Register ()
Login ()
Profile Update ()
Logout ()

Device Registration
Type: String
Model: String
Storage Capacity: int
Time Duration: Date
Registration Request ()

Payment
Request ID: String
Quote: Integer
Payment Details: Card
Approved ()
Declined ()

Registration
First name: String
Last name: String
Consumer /Lender: Boolean
Email: String
Password: String
Register ()

Login
Email: String
Password: String
Authenticate ()

Card
Type of card: Boolean
Name: String
Number: Int
Exp: Date
CVV:INT
Verity ()

Dash board
Amount of Storge: Integer
Details: Object
Pricing: Pricing
Display Details ()

Pricing
Storage capacity: integer
Time Duration: integer
Pricing Model: Pricing model
Get pricing model ()
Calculate price ()

Consumer
First name: String
Last name: String
Address: String
Email: String
Space Request
Register ()
Login ()
Profile Update ()
Logout ()

Space Request
Memory Space: integer
Time Duration: integer
Storage Space Request

Pricing Model
Price per unit: integer
Time Duration: integer
Type of customer: string
Create price Model

Storage and Retrieval
Stored Data: File
Device Details: String
Get Device Details ()
Get Data ()
Store Data ()
Display Data ()

Data Base Diagram

2.1 INTERVIEWS

User Interview #1: A Private Firm Owner

- I want to be able to contact people who are also the clients of the same software service that I am using.
- Want to have a one-one collaboration with my peers through direct messaging.
- Should be able to see all the options to join with my peers to lend our space together.

User Interview #1: A Vault Customer

- I want the application to provide a user-friendly form with no hidden details for a seamless peer collaboration.
- Want a very clear and descriptive pricing detailed with how the benefits would be distributed among the peers.
- Should be given an opportunity collaborate and lend even the minimum amount of storage space to the vault.

2.2 USER STORIES

User Story #1: Small Business Owner

- He is an active vault user, lending 20 GB of storage space to the vault receiving certain amount, but he intends to increase the percentage of the benefit he is receiving.
- He logs into the system and checks the peer collaboration to find some of his peers interested in collaborating to share the storage space together as such increase the revenue and profitability for all the peers in collaboration.
- The received benefits are shared among all the peers based on their amount of memory.

Peer 1

Peer 2

Collaboration Invite

One-one Messaging

Collaboration Request

Admin

Pricing and Payment
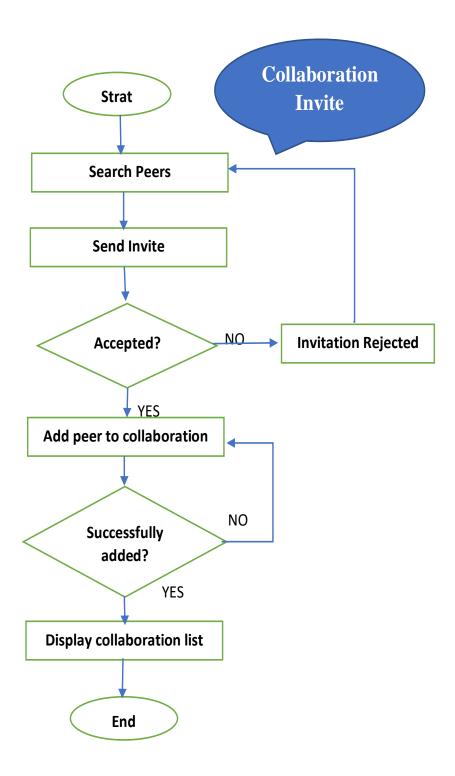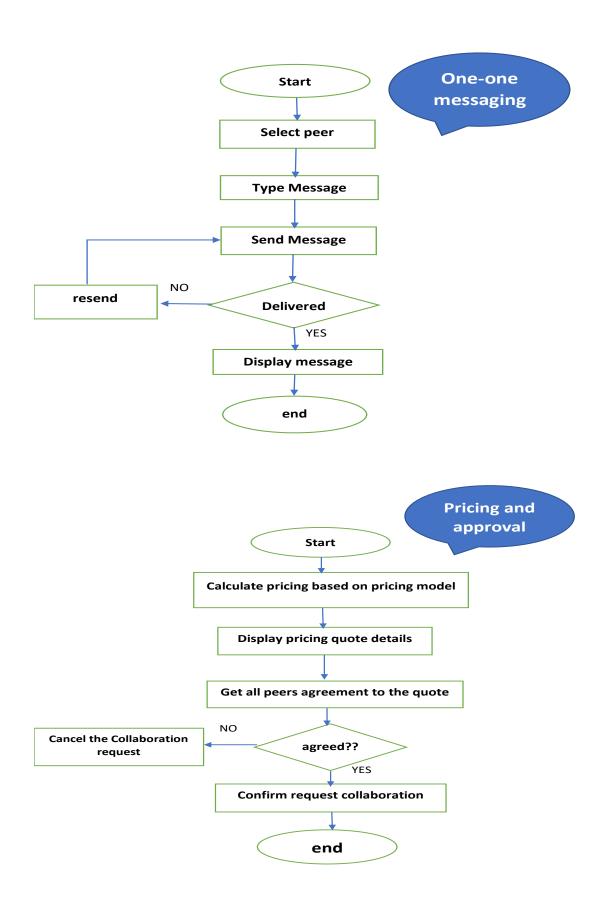
Title:	Collaboration Invite
Description:	The lenders communicate with their peers to collaborate and lend their storage space together.
Primary Actor:	Lenders
Preconditions:	The lenders should be registered users with the vault.
Postconditions:	The lenders are provided with a one-one messaging platform for discussion.
Requirements	Lenders needs to be registered with the vault in-order to collaborate. They should be having at least 10GB storage space to collaborate with each side.
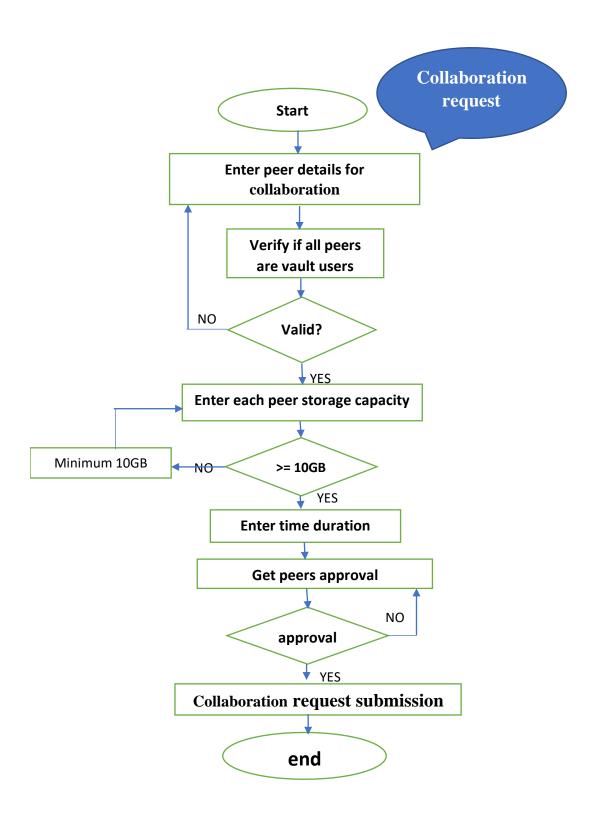
Title:	One-one Discussion Board
Description:	The lenders communicate with their peers to collaborate and lend their storage space together.
Primary Actor:	Lenders
Preconditions:	The lenders should be registered users with the vault.
Postconditions:	The lenders mutually decide to lend particular amount space from their end for a specified period of time to utilize the maximum benefits of storage lending.
Requirements	- Lenders needs to be registered with the vault in-order to collaborate. - They should be having at least 10GB storage space to collaborate with each side.
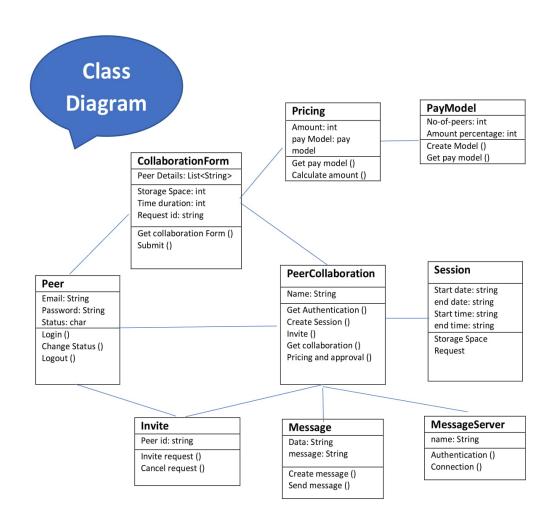
Title:	Collaboration request to admin
Description:	The lenders discuss the storage details and time for lending through the one-one discussion and send a filled collaboration request form to the admin approval.
Primary Actor:	Lenders, admin
Preconditions:	The lenders need to collaborate through a one-one discussion to discuss their peer collaboration requirements and proposal
Postconditions:	The lenders receive either approval or denial notice from the admin. If collaboration request got approved, the lenders will be provided with calculated quote for their collaborated lending.
Requirements	Should have a registered account with account details with the vault. All peers in the group have to fill the collaboration form.
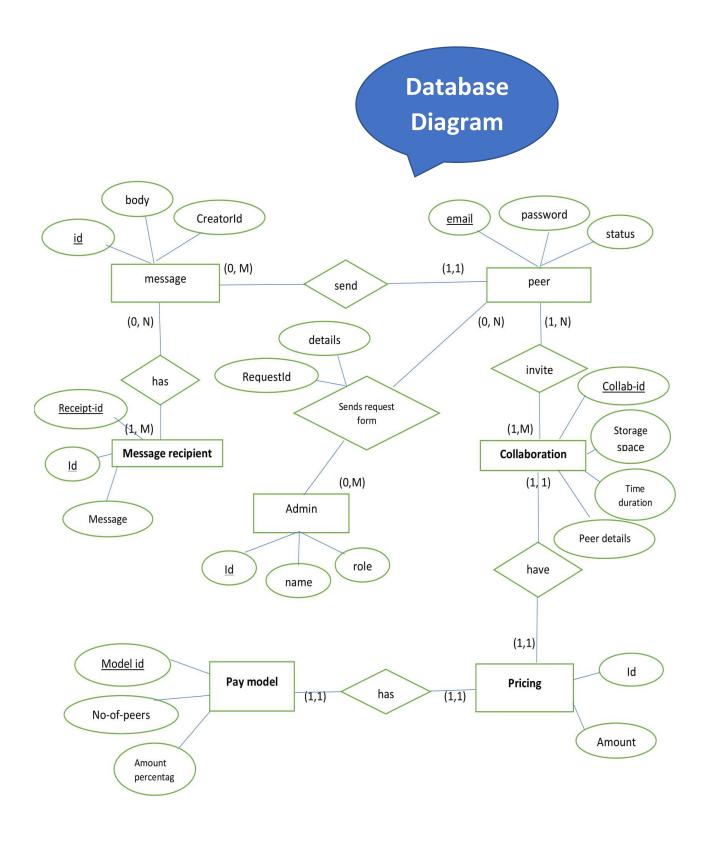
Title:	Pricing and Approval
Description:	The lenders provided with collaboration approval notice and a calculated pricing details by the admin for their collaborated lending.
Primary Actor:	Lenders, Admin
Preconditions:	The lenders in order to receive the price details, they require to fill a collaboration request form that has to be approved by the admin.
Postconditions:	The lenders are provided with a calculated price through the pricing model and based on the peer's agreement to the quote provided, the collaboration request is either confirmed or cancelled.
Requirements	Lenders need to accept the quote of the pricing model to be accepted for the peer collaboration. Should have a registered account with account details with the vault.

Strat

Collaboration Invite

Search Peers

Send Invite

Accepted? — NO → **Invitation Rejected**

YES

Add peer to collaboration

Successfully added? — NO

YES

Display collaboration list

End

One-one messaging

```
                    Start

                  Select peer

                  Type Message

                  Send Message

        NO
resend  ◄──────    Delivered
                      │ YES
                  Display message

                      end
```

Pricing and approval

```
                        Start

            Calculate pricing based on pricing model

                Display pricing quote details

                Get all peers agreement to the quote

            NO
Cancel the      ◄────      agreed??
Collaboration                  │ YES
request
                    Confirm request collaboration

                            end
```

```
                                        ╭─────────────────────╮
                ╭─────────╮             │  Collaboration      │
                │  Start  │             │  request            │
                ╰─────────╯             ╰─────────────────────╯
                     │
                     ▼
        ┌──────────────────────────┐
        │  Enter peer details for  │
        │      collaboration       │
        └──────────────────────────┘
                     │
                     ▼
        ┌──────────────────────────┐
        │   Verify if all peers     │
        │    are vault users        │
        └──────────────────────────┘
                     │
                     ▼
     NO          ◇ Valid? ◇
        ◄────────────┘
                     │ YES
                     ▼
        ┌──────────────────────────┐
        │ Enter each peer storage capacity │
        └──────────────────────────┘
                     │
                     ▼
  ┌────────────┐   ◇ >= 10GB ◇
  │ Minimum 10GB │ ◄── NO
  └────────────┘
                     │ YES
                     ▼
        ┌──────────────────────────┐
        │    Enter time duration    │
        └──────────────────────────┘
                     │
                     ▼
        ┌──────────────────────────┐
        │    Get peers approval     │
        └──────────────────────────┘
                     │
                     ▼
              ◇ approval ◇ ── NO
                     │ YES
                     ▼
        ┌──────────────────────────┐
        │ Collaboration request submission │
        └──────────────────────────┘
                     │
                     ▼
                ╭─────────╮
                │   end   │
                ╰─────────╯
```

Class Diagram

Pricing

Amount: int
pay Model: pay
model

Get pay model ()
Calculate amount ()

PayModel

No-of-peers: int
Amount percentage: int

Create Model ()
Get pay model ()

CollaborationForm

Peer Details: List<String>

Storage Space: int
Time duration: int
Request id: string

Get collaboration Form ()
Submit ()

Peer

Email: String
Password: String
Status: char

Login ()
Change Status ()
Logout ()

PeerCollaboration

Name: String

Get Authentication ()
Create Session ()
Invite ()
Get collaboration ()
Pricing and approval ()

Session

Start date: string
end date: string
Start time: string
end time: string

Storage Space
Request

Invite

Peer id: string

Invite request ()
Cancel request ()

Message

Data: String
message: String

Create message ()
Send message ()

MessageServer

name: String

Authentication ()
Connection ()

Database Diagram

body
CreatorId
id
message
(0, M)
send
(1,1)
peer
email
password
status

(0, N)

has

Receipt-id
(1, M)

Message recipient
Id
Message

details
RequestId
Sends request form
(0,M)

Admin
Id
name
role

(0, N)
(1, N)
invite
(1,M)

Collaboration
Collab-id
Storage space
Time duration
Peer details

(1, 1)
have
(1,1)

Model id
Pay model
No-of-peers
Amount percentag
(1,1)
has
(1,1)
Pricing
Id
Amount

3.1 INTERVIEWS

User Interview #1: Teenager

- Need to have an application that is convenient, and you do not have to set up and monitor an additional account to buy and sell my storage devices.
- I should be able to purchase items or create an advertisement.

User Interview #2: A Vault Customer

- The application should be able to give an option to connect directly with the sellers.
- Need to buy an item on marketplace by messaging the seller and negotiating a price or accepting the price as it is.
- The search filters should allow you to find a specific storage device from the available listings.
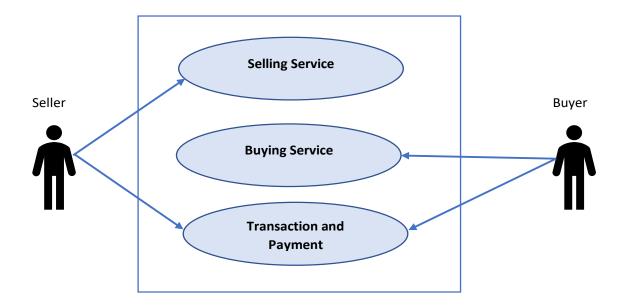
3.2 USER STORIES

User Story #1: A Vault customer who is also a dealer of Storage devices.

- This person wants to use this platform to advertise his products, since he can find more potential buyers through this application.
- He wants to check out what are the other products in the market which are in demand through the buying and selling pattern.
- After finding his potential customers he wants to connect with them directly to give a more detail description of his product.

User Story #2: A Frequent Traveler
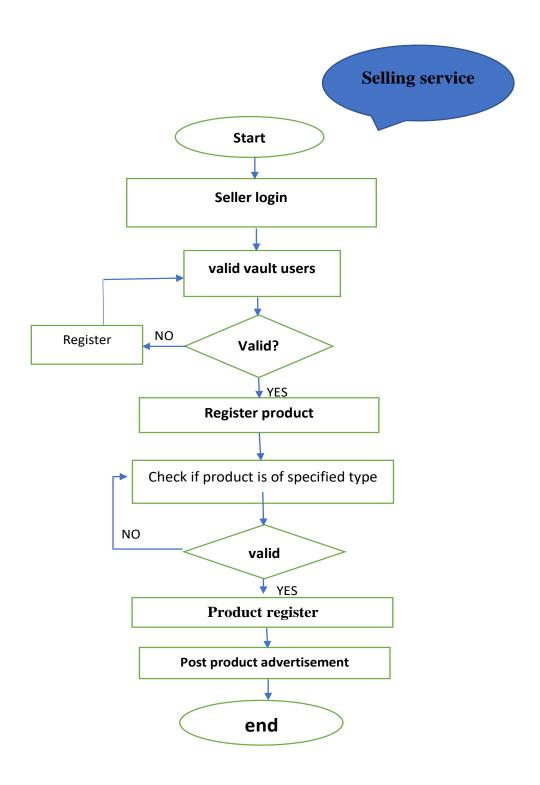
- Looking to buy a storage device with sufficient memory to carry with him during travel.
- He is not willing to pay a large amount for it, so he is searching for someone who can sell him the device on a discounted price.
- He is also willing to buy a used device if the seller has provided a detail description of his device.
- A one-one interaction with the seller to know about the device functionality and price negotiation would be an added advantage.
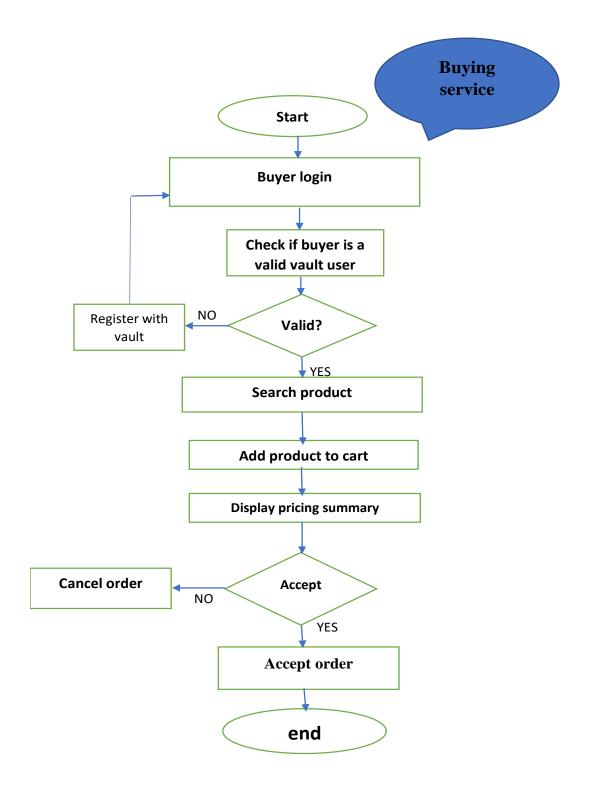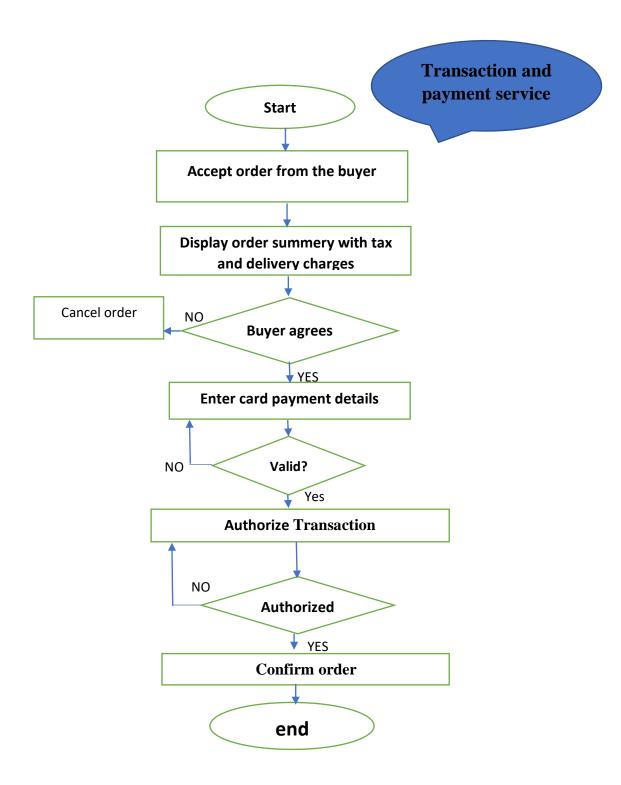
Title:	Selling Service
Description:	Registered Vault users can utilize this feature to advertise and sell their storage devices to other registered vault users.
Primary Actor:	Seller
Preconditions:	The sellers should be registered users with the vault. Give the details of the type and specifications of the product he would like to sell.
Postconditions:	The seller is registered to the marketplace if he is valid user. If the product or the device, they are trying to sell is specified in the list of type of products allowed for trade then the sellers product is registered for trade.
Requirements	• Lenders needs to be registered vault users. • The device must be of the type that is in the list of products allowed for selling and buying.
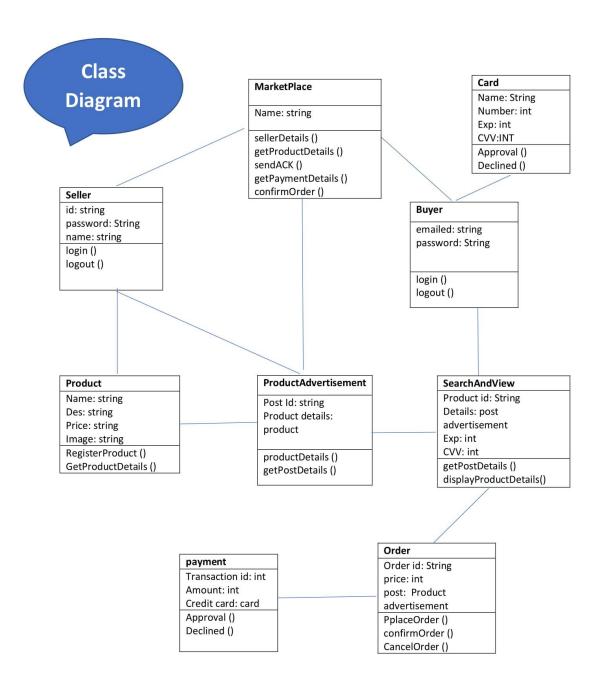
Title:	Buying Service
Description:	Registered Vault users can utilize this feature to buy the products on discounted prices from the other vault users instead of paying high prices for the same products outside.
Primary Actor:	Buyers and Sellers
Preconditions:	The buyers should be registered users with the vault.
Postconditions:	The buyers search for all the available products on the marketplace and if the products match their requirement, requests a quote from the seller.
Requirements	- Buyers and sellers need to be registered vault users. - The buyers should accept the price quote from the sellers.
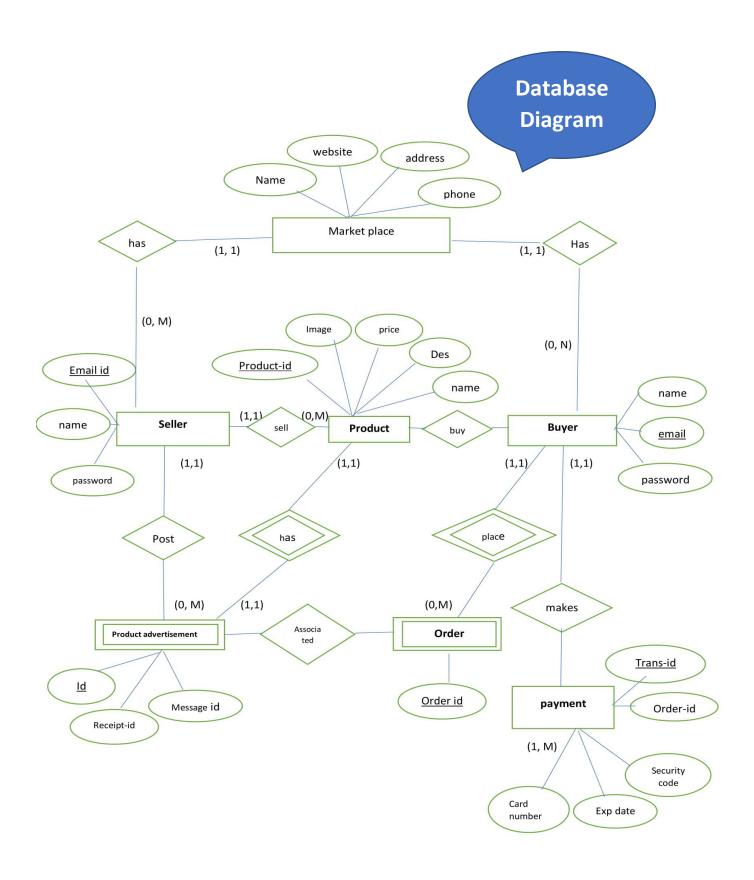
Title:	Transaction and Payment Service
Description:	The seller for the specified product provides the buyer with a quote for his product and if the seller is satisfied with that quote, they make a deal and the buyer and seller provide their transaction details for debit and credit.
Primary Actor:	Buyers and Sellers
Preconditions:	The buyer should accept the price quote provided by the seller.
Postconditions:	The order summary is generated with the details of the product and price and if the seller agrees to this order summary, he provides his account details to complete the transaction.
Requirements	- Provide Bank account details. - Debit/Credit card Details.

Selling service

Start

Seller login

valid vault users

Register   NO   Valid?

YES

Register product

Check if product is of specified type

NO   valid

YES

Product register

Post product advertisement

end

Buying service

Start

Buyer login

Check if buyer is a valid vault user

Valid?

NO → Register with vault

YES

Search product

Add product to cart

Display pricing summary

Accept

NO → Cancel order

YES

Accept order

end

Start

Transaction and payment service

Accept order from the buyer

Display order summery with tax and delivery charges

Buyer agrees

NO → Cancel order

YES

Enter card payment details

Valid?

NO

Yes

Authorize Transaction

Authorized

NO

YES

Confirm order

end

Class Diagram

MarketPlace

Name: string

sellerDetails ()
getProductDetails ()
sendACK ()
getPaymentDetails ()
confirmOrder ()

Card

Name: String
Number: int
Exp: int
CVV:INT

Approval ()
Declined ()

Seller

id: string
password: String
name: string

login ()
logout ()

Buyer

emailed: string
password: String

login ()
logout ()

Product

Name: string
Des: string
Price: string
Image: string

RegisterProduct ()
GetProductDetails ()

ProductAdvertisement

Post Id: string
Product details:
product

productDetails ()
getPostDetails ()

SearchAndView

Product id: String
Details: post
advertisement
Exp: int
CVV: int

getPostDetails ()
displayProductDetails()

payment

Transaction id: int
Amount: int
Credit card: card

Approval ()
Declined ()

Order

Order id: String
price: int
post: Product
advertisement

PplaceOrder ()
confirmOrder ()
CancelOrder ()

Database Diagram

28

4.1 INTERVIEWS

User Interview #1: A Prospect Customer

- To view the Reviews, I should not have to create an account and login.
- I should be able to filter and view the reviews and ratings for every component in the application.
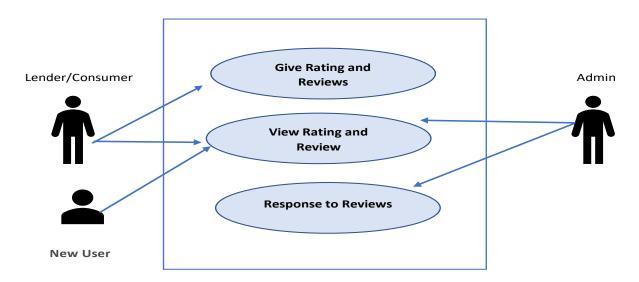
User Interview #2: A Vault User

- I should be able to review each, and every service provided for every component in the application.
- I should be able to sort the reviews and ratings based on highest, lowest, recent and relevance.

4.2 USER STORIES
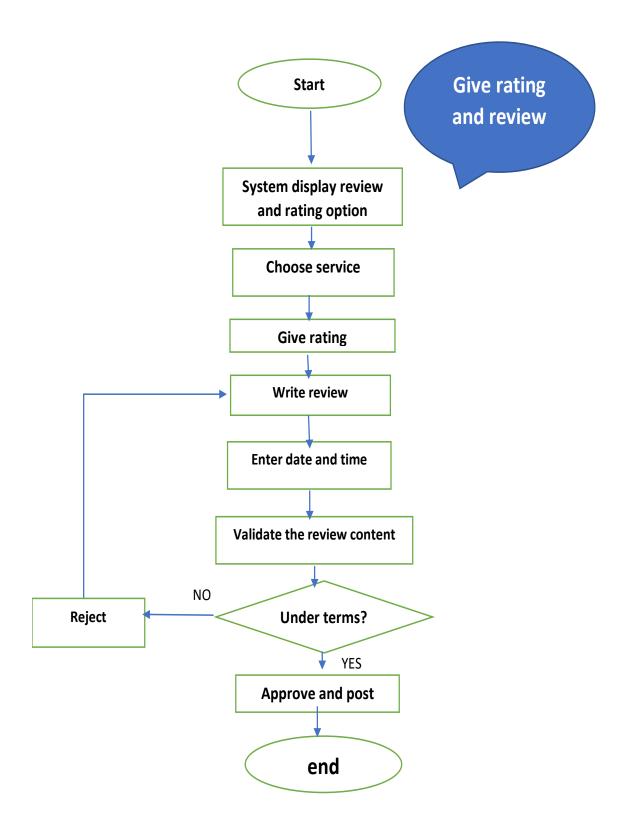
User Story #1: A Student looking for an online data storage.
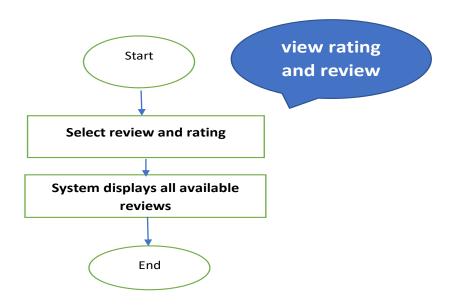
- He is using an application for which he was not able to review the services, as such he is not satisfied with the efficiency of the application.
- Reviews based on the security of data, latency will give an opportunity to lend or store data through the following application.

4.3 USER CASES
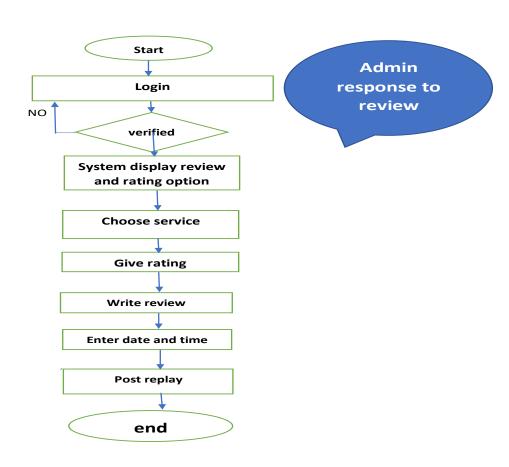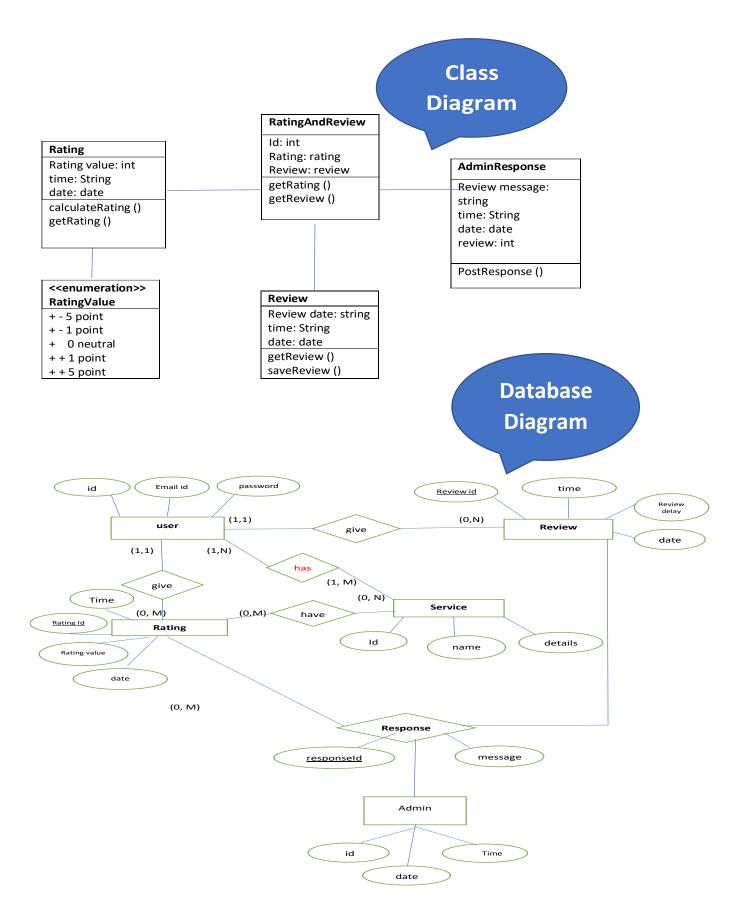
Title:	Give Rating and Review
Description:	The lenders and Consumers are given an option to provide ratings and reviews for the services provide by the storage vault application. These reviews are visible to anyone who opens this application, as such it gives a platform to attract more users to the vault.
Primary Actor:	Lenders and Consumers
Preconditions:	Ratings and reviews can be given by only the registered users of the vault. The registered users of the vault should have been utilizing the facilities of the vault application for at least a month to give a reliable review for the application.
Postconditions:	The reviews and ratings for the services is visible to the new and existing users and the admin.
Requirements	The lenders and consumers reviewing the application must be authorized users of the vault.

Title:	View Ratings and Reviews and Admin Response
Description:	The reviews and ratings for the vault services by its customers is visible to the admin and new users for view. The new users can have an idea about the functionalities of the application and the admin can reply to the reviews.
Primary Actor:	Lenders, Consumers, New Users and Admin
Preconditions:	Ratings and reviews will be given by the vault customers for view.
Postconditions:	The new users can view these comments and the admin can reply to these comments wherever necessary.
Requirements	There should be reviews and ratings given by the vault customers for the admin to respond. The new users need not register to the vault to view the review and ratings for the application.

```
                              ┌─────────────┐
                              │    Start    │
                              └─────────────┘
                                     │
                                     ▼
                         ┌───────────────────────┐
                         │  System display review│
                         │   and rating option   │
                         └───────────────────────┘
                                     │
                                     ▼
                         ┌───────────────────────┐
                         │     Choose service    │
                         └───────────────────────┘
                                     │
                                     ▼
                         ┌───────────────────────┐
                         │      Give rating      │
                         └───────────────────────┘
                                     │
                                     ▼
                         ┌───────────────────────┐
                         │     Write review      │
                         └───────────────────────┘
                                     │
                                     ▼
                         ┌───────────────────────┐
                         │   Enter date and time │
                         └───────────────────────┘
                                     │
                                     ▼
                         ┌───────────────────────┐
                         │Validate the review    │
                         │      content          │
                         └───────────────────────┘
```

Give rating and review

Start

System display review and rating option

Choose service

Give rating

Write review

Enter date and time

Validate the review content

NO

Reject

Under terms?

YES

Approve and post

end

view rating and review

```
        Start

          ↓

  Select review and rating

          ↓

  System displays all available
            reviews

          ↓

         End
```

Admin response to review

```
        Start

          ↓

        Login  ←──────┐
          ↓           │
  NO ←─ verified      │
          │           │
          ↓           │
  System display review
    and rating option

          ↓

     Choose service

          ↓

       Give rating

          ↓

      Write review

          ↓

   Enter date and time

          ↓

       Post replay

          ↓

         end
```

Rating

Rating value: int
time: String
date: date

calculateRating ()
getRating ()

RatingAndReview

Id: int
Rating: rating
Review: review

getRating ()
getReview ()

AdminResponse

Review message:
string
time: String
date: date
review: int

PostResponse ()

**<<enumeration>>
RatingValue**

+ - 5 point
+ - 1 point
+   0 neutral
+ + 1 point
+ + 5 point

Review

Review date: string
time: String
date: date

getReview ()
saveReview ()

**Database
Diagram**

id   Email id   password

Review id   time   Review delay

user   (1,1)   give   (0,N)   Review   date

(1,1)   (1,N)

has

(1, M)

give   (0, N)

Time   (0,M)   (0,M)   have   Service

Rating Id   Rating

Rating value   Id   name   details

date

(0, M)

Response

responseId   message

Admin

id   Time

date

33

5.1 INTERVIEWS

User Interview #1: A Student

- Looking for a plan which cover any old or new device, accidents, damages with greater discounts.
- An easy option to switch between the standard and premium plans.
- Continues coverage when he upgrades his device.

5.2 USER STORIES

User Story #1: A student

- He experienced some damage to his old NAS storage devices in the past year. His protection plan does not cover old devices.
- He paid a huge amount for a one-year protection plan, but the insurance company does not want to replace the faulty device even after 2 months of purchase, since the policy does not cover replacement.
- His previous insurance company took around one month to process his lost device claim request. He is looking for an easy and quick process.

5.3 USER CASES

Title:	Receipt Verification
Description:	The vault user is provided with an option where he uploads the receipt of the device purchase. With the receipt number, the admin verifies with the store with whom vault has a collaboration.
Primary Actor:	Vault user, Vault admin
Preconditions:	The vault user has to purchase the storage device from one of the vault's collaborated stores.
Postconditions:	For verified purchase, the vault user will be provided with various protection plans to select either standard or premium.
Requirements:	- The user must be a registered user with the vault. - The device purchase must be from one of the vault collaborators.

Title:	Protection plans
Description:	For a verified purchase, the vault user will be provided an option to choose from the various standard and premium protection plans.
Primary Actor:	Vault user, Vault admin
Preconditions:	The vault user has to verify his device purchase with the vault admin.
Postconditions:	For the selected protection plan, the vault user is provided with payment details.
Requirements:	The device purchase must be verified.

Title:	Payment
Description:	The Vault user is provided with various payment options to pay for his protection plan quote.
Primary Actor:	Vault user, Vault admin
Preconditions:	The vault user has to choose a suitable protection plan.
Postconditions:	In the future, he can begin claiming based on the protection plan coverage.
Requirements:	The user has to have a credit or debit card.

Title:	Claims
Description:	The vault user is provided with a claiming option for device repair, damage, or replacement service.
Primary Actor:	Vault user, Vault admin
Preconditions:	He must pay the amount of the protected plan which he has chosen.
Postconditions:	Based on the plan coverage, he will be provided with either repair or replacement of the device.
Requirements:	The user has to wait for at least three months to claim his first request after the payment.

Start

Receipt verification

Enter purchase number

Verify if purchase from vault collaborators

Valid?

Not a verified purchase

NO

YES

verified purchase

Show protection plans

end

```
                    ┌─────────────┐        ╭─────────────╮
                    │    Start    │        │ Protection  │
                    └──────┬──────┘        │   plans     │
                           │               ╰─────────────╯
                    ┌──────▼──────────────┐
                    │ System display       │
                    │ Protection plans     │
                    └──────┬──────────────┘
                           │
                    ┌──────▼──────────────┐
         ┌─────────►│ Select a Protection  │
         │          │ plans                │
         │          └──────┬──────────────┘
         │                 │
         │          ┌──────▼──────────────┐
         │          │ Select either        │
         │          │ standard or          │
         │          │ premium plan         │
         │          └──────┬──────────────┘
         │                 │
         │          ┌──────▼──────────────┐
         │          │ Create policy and    │
         │          │ get agreement        │
         │          └──────┬──────────────┘
         │                 │
    NO   │          ◄──────▼──────►
         └──────────  Agreed
                           ◄──────►
                             │  YES
                    ┌────────▼────────┐
         ┌─────────►│ Payment options │
         │          └────────┬────────┘
         │                   │
    NO   │            ◄──────▼──────►
         └──────────   authorized
                             ◄──────►
                               │  YES
                      ┌────────▼────────┐
                      │ Policy approval │
                      └────────┬────────┘
                               │
                        ┌──────▼──────┐
                        │     end     │
                        └─────────────┘
```

Start

System display Protection plans

Select a Protection plans

Select either standard or premium plan

Create policy and get agreement

Agreed

NO   YES

Payment options

authorized

NO   YES

Policy approval

end

Protection plans

Class Diagram

Admin

name: string
emailed: string
password: String
id: int
role: string

Approval ()
Deny ()

Receipt verification

Receipt no: int
Request id: int

Verify receipt

Claim

Claim id: int
Policy id: int
Amount: int

Submit claim ()

Card

Name: String
Number: int
Exp: int
CVV: int

get details ()

policy

Policy id: int
Plan: plan
Customer id: int
Start date: date

Create policy ()
get policy ()

plan

Name: String
Plan id: int
Premium: int
frequency: string

Create plan ()
get plan ()

Payment

Amount: int
 card: card

Approval ()
Declined ()

Customer

name: string
emailed: string
password: String
customer id: int

login ()
get policy ()
submit email ()

Premium

Premium id: int
Policy id: int
Amount: int

Get Premium plan ()

Database Diagram

42

All Components Class Diagram

Registration

Fname:String
Lname:String
Consumer
/Lender:Boolean
Email:String
Password:String
Register ()

Login

Email:String
Password: String

Authenticate()

Admin

name: string
emailId: string
password: String
id: int
role: string
Approval ()
Deny ()

Lender

Fname:String
Lname:String
Address:String
Email:String
Device Details: Device
registration
Register ()
Login ()
Profile Update ()
Logout ()

StorageVault

Name: String
Website: String
deviceAndStorageRequest: DeviceAndStorageRequest
peerCollaboration: PeerCollaboration
protectionPlans: ProtectionPlans
marketplace: MarketPlace
ratingAndReviews: RatingAndReviews
login: Login
getLoginDetails()
DeviceAndStorageRequest()
PeerCollabration()
ProtectionPlans()
MarketPlace()
RatingAndReviews()

Consumer

Fname:String
Lname:String
Address:String
Email:String
Space Request
Register ()
Login ()
ProfileUpdate ()
Logout ()

PeerCollaboration

Name: String
getAuthentication ()
createSession ()
Invite ()
getCollaboration ()
pricingAndApproval ()

Pricing

storageCapacity: integer
timeDuration: integer
pricingModel: PricingModel
getPricingModel ()
calculatePrice ()

ProtectionPlans

Name: String
verifyReceipt()
getPolicy()
getPaymentDetails()
submitCalim()

MarketPlace

name: string
sellerDetails ()
getProductDetails ()
sendACK ()
getPaymentDetails ()
ConfirmOrder()

RatingAndReview

Id: int
Rating: rating
Review: review
getRating ()
getReview()

DeviceAndStorageRequest

Lender: Lender
consumer: Consumer
deviceRegistration: DeviceRegistration
spaceRequest: SpaceRequest
pricing: Pricing
payment: Payment
Lender()
Consumer()
DeviceRegistration()
SpaceRequest()
Pricing()
Payment()

PricingModel

Price per unit: integer
Time Duration: integer
Type of customer: string
createPricingModel()
getModel()

Payment

requestID:String
quote: Integer
paymentDetails: Card
Approved ()
Declined ()

Card

Type of card:boolean
Name:String
Number:integer
Exp:Date
SecurityCode: int
Verity ()

All Components ER Diagram