P.G.D.A.V. College (M), University Of Delhi

B. Sc. (H) Computer Science (II Year) SEMESTER IV

Software Engineering Project

CONTACTLESS STORE

Developers:

Pushpender (20053570026)

Esha Dahiya (20053570042)

Submitted to:

Dr. Aparna Datt

CERTIFICATE

This is to certify that the project entitled, "CONTACTLESS STORE" has been done by: PUSHPENDER AND ESHA DAHIYA of Bachelor of Science in Computer Science during semester IV from PGDAV College (Day) University of Delhi.

It embodies the work done by them under the due supervision of DR. APARNA DATT

DECLARATION

I hereby declare that this Project Report titled "CONTACTLESS STORE" submitted to the Department of Computer Science, PGDAV College (University Of Delhi) is a record of original work done by the team under the guidance of Dr. Aparna Datt. The information and data given in the report is authentic to the best of the team knowledge. This Project Report is not submitted to any other university or institution for the award of any degree, diploma or fellowship or published any time before

ACKNOWLEDGEMENT

The contentment that is achieved on the successful completion of any task is incomplete without mentioning the names of the people who made it possible with their consistent guidance, support and indispensable encouragement. The Project was jointly undertaken by PUSHPENDER AND ESHA DAHIYA as their 4th semester Software Engineering Project, under the able guidance and supervision of DR. APARNA DATT. Our primary thanks to her, who poured over every inch of our project with painstaking attention and helped us throughout the working of the project. It's our privilege to acknowledge our deepest sense of gratitude to her for her inspiration which has helped us immensely. We are extremely grateful to her for unstilted support and encouragement in the preparation of this project.

INDEX

PROBLEM STATEMENT

There are a lot of physical shopping stores available around us but the traditional way of shopping has a lot of disadvantages. For instance, we waste a lot of time standing in long queues, meeting lots of strangers and going through the fuss. In this time of pandemic, customers want less contact with the surface while shopping and want to shop in a safe and secure environment.

Hence, we came up with a **CONTACTLESS STORE** which will serve as an ideal walk-in store for customers, providing them with a safe and secure environment for hassle free shopping. We are planning for a no man-power store that will be fully automatic and functional with an Iris sensor to take eye prints which will help in fetching the details of the person entering the store and will aid in the best security and make it completely contactless. Inside the store, products are in a glass window box with a unique QR code which can be used for the payment, helping the customer escape from long billing queues. Once the payment is made the customer will get a confirmation message and the expiry date of the product and whenever the product reaches near to its date of expiry, the customer will get a reminder.

We also have a complete automatic inventory management for the store which will help in obtaining accurate operating costs, track cost of sales and helps in identifying the best and worst performing product which acts beneficial for people wanting a side-hustle with their job.

This system hence solves a lot of problems like it provides a safe,secure and contactless environment with no man-power which results in saving customer and owner's time hence providing satisfaction of shopping physically without any fear.

EVOLUTIONARY MODEL

Evolutionary model is a combination of Iterative and Incremental model of software development life cycle. Delivering it in an incremental process over time is the action done in this model. Some initial requirements and architecture envisioning need to be done.

It is better for software products that have their feature sets redefined during development because of user feedback and other factors. The Evolutionary development model divides the development cycle into smaller, incremental waterfall models in which users are able to get access to the product at the end of each cycle.

Quick Design => Construction of prototype => evolution by stakeholder

Prototyping: Prototyping is a software development method where the developer or development team first constructs a prototype. After receiving initial feedback from the customer, subsequent prototypes are produced, each with additional functionality or improvements, until the final product emerges.
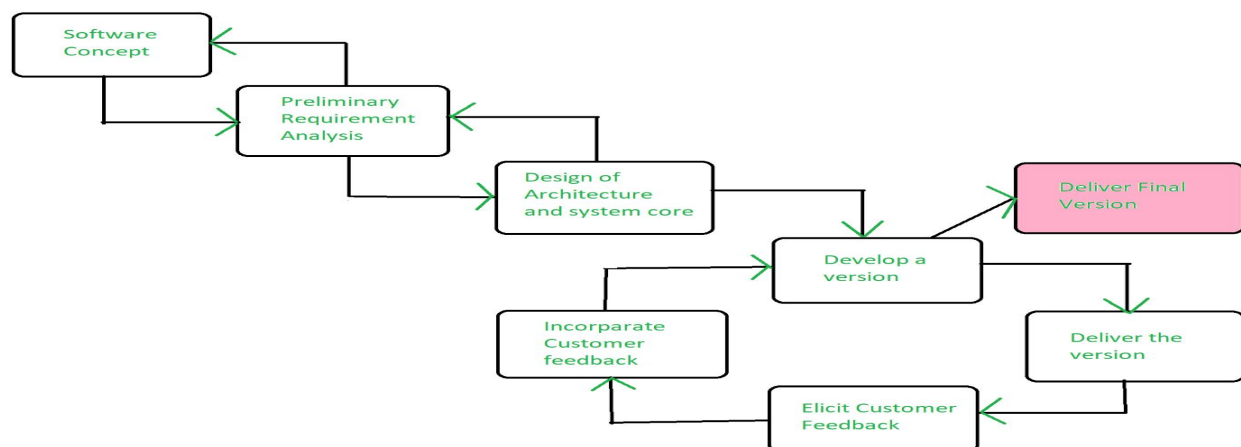

Fig 1

Reason for selecting Evolutionary Model:

1. As per the growth of the Store, owner needs to add new functionalities so for this we selected Evolutionary Model
2. Evolve the AI model according to the product's return and replacement (if there is any type of replacement/refund of a product, our AI model will check the product)
3. Evolve according to customer's feedback.

Software Requirement Specifications(SRS):-

A software requirements specification (SRS) is a document that captures complete description about how the system is expected to perform. It is usually signed off at the end of requirements engineering phase.

Qualities of SRS:

Correct, Unambiguous, Complete, Consistent,Ranked for importance and/or stability, Verifiable, Modifiable and Traceable.

The purpose of this software project is to provide a completely contactless platform for customers. Its main functionality is to provide customer a safe and secure environment for contactless shopping.

The key functionalities that the store should include are:-

- Firstly, we have to fetch details of every customer using eyeprints (using a high quality camera with iris sensor).
- Customer will choose the product from the showcase and pay using the respective QR code of every product.
- After successful payment the showcase will open for the customer so that the product is accessible and get a receipt on email/phone number with expiry date(if any) with a link for feedback.
- System will showcase new products in the place of old ones.
- Update the database for further inventory management for the owner and for reminders regarding expiry for customers.
- Check physical damages using cctvs and IR sensors.
- Owner will be able to watch the real time view of the store.
- Read customer's feedbacks.
- Inventory management.

USER CLASSES AND CHARACTERISTICS

The users of this project includes owner, developers and customers.

The users have different access according to their needs and level of privilege.

OWNER:

The owner is the authorized officer of the company which is concerned and knowledgeable with the definite functions and features of the system.In this case it may be a person with a job/business and want a side hustle with less interaction. He/she will use the system to maintain the proper working of the store and check for repairs if any. It's his responsibility to decide the functionality of each user, their privacy, experience and look after the whole system in case of any breakdown.

DEVELOPER:

The developers include the team who is responsible for ensuring user needs are met through the deployment and updates of the software.

They are also responsible for testing and modifying system to ensure that they operate reliably. Developers ensure that the interaction between the customer and system happens in a secure environment.

CUSTOMER:

These include people interested in shopping offline in a completely contactless process. People who want to save their time by escaping long billing queues and do not wish to get in contact with any object/person are the ideal customers. They will interact with the system, pay online and hence get the best contactless user experience.

Use Case:

A use case is a methodology used in system analysis to identify, clarify and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. The method creates a document that describes all the steps taken by a user to complete an activity.

Diagram:
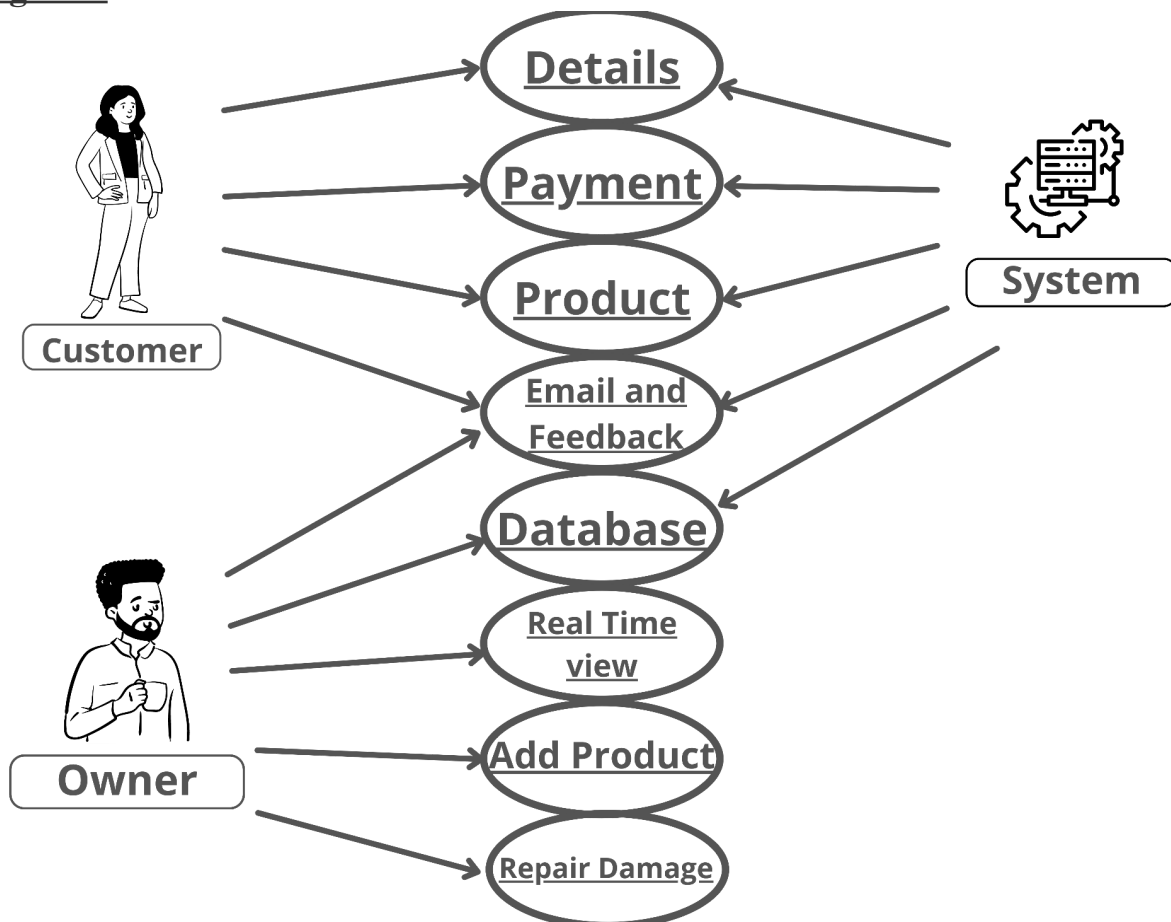
Fig 2

DFD(Data Flow Diagram):

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system.

The objective of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communication tool between a system analyst and any person who plays a part in the order that acts as a starting point for redesigning a system. The DFD is also called a data flow graph or bubble chart.

DFD (level 0):

Fig 3

6

DFD(level1):

Fig 4

7
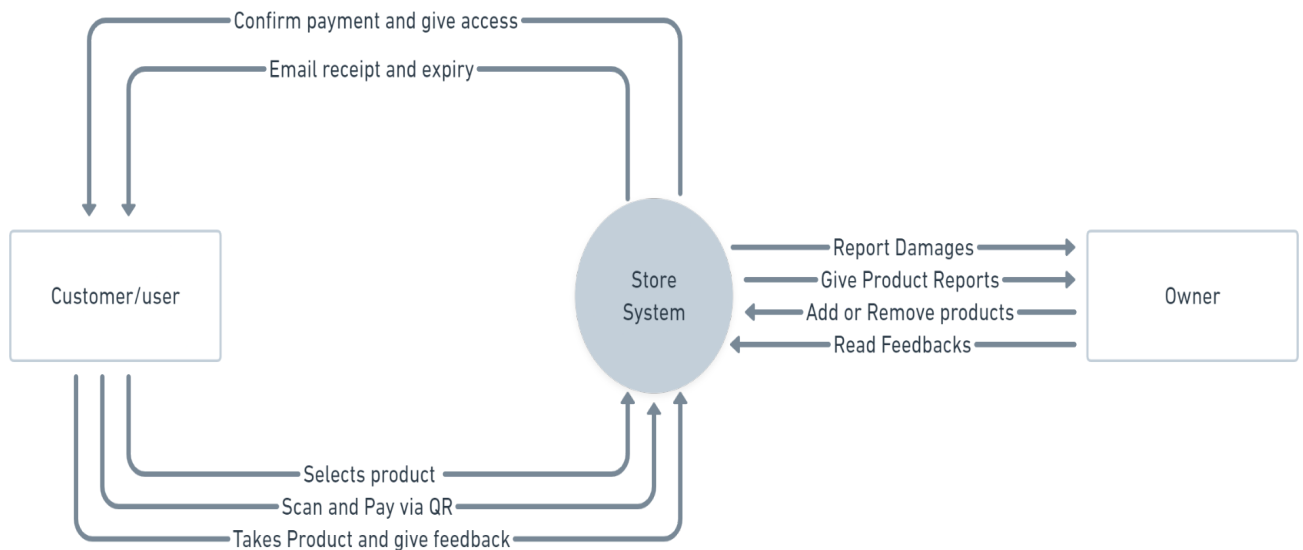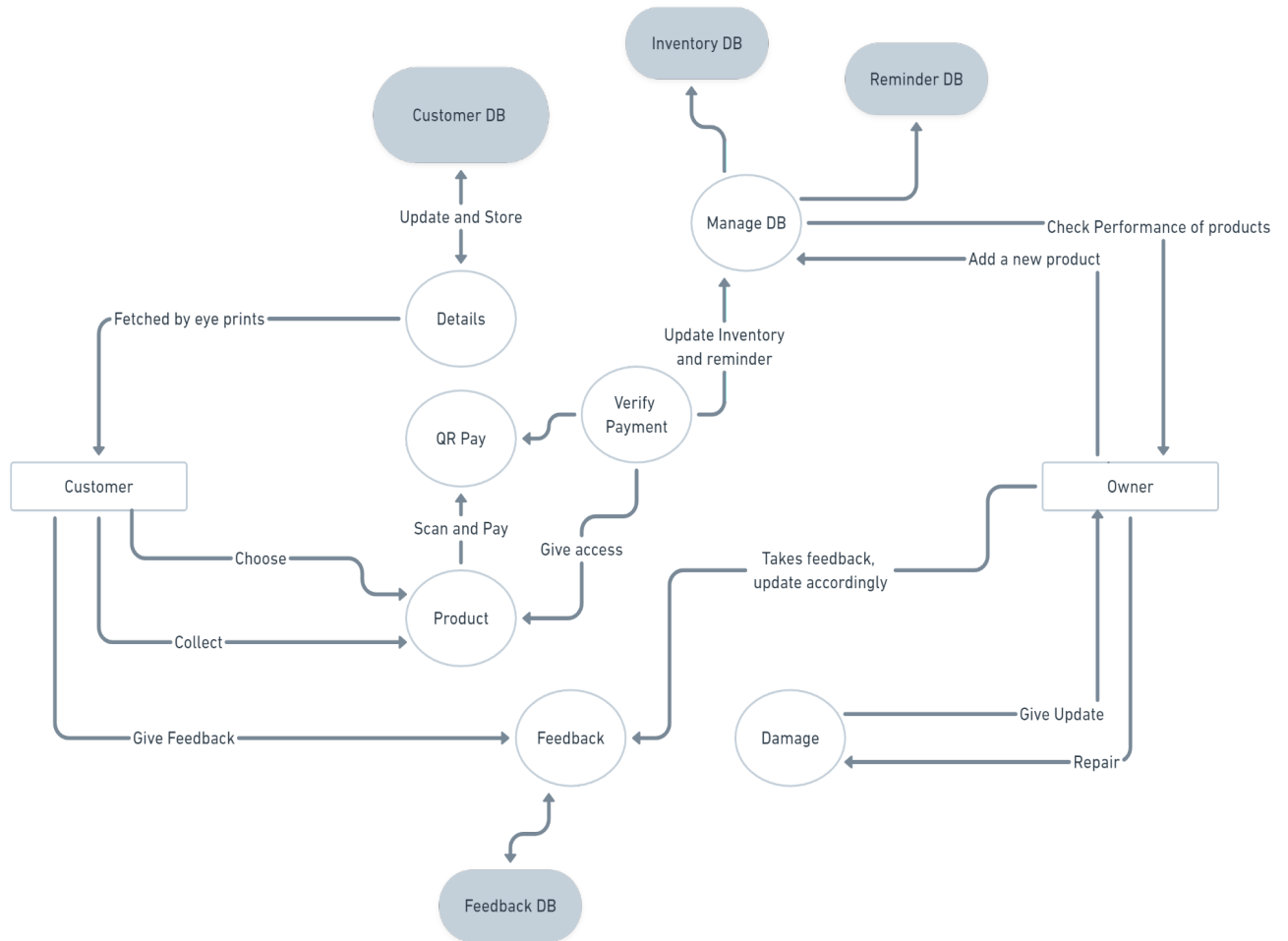
ERD:

ER-modeling is a data modeling method used in software engineering to produce a conceptual data model of an information system. Diagrams created using this ER-modeling method are called Entity-Relationship Diagrams or ER diagrams or ERDs.
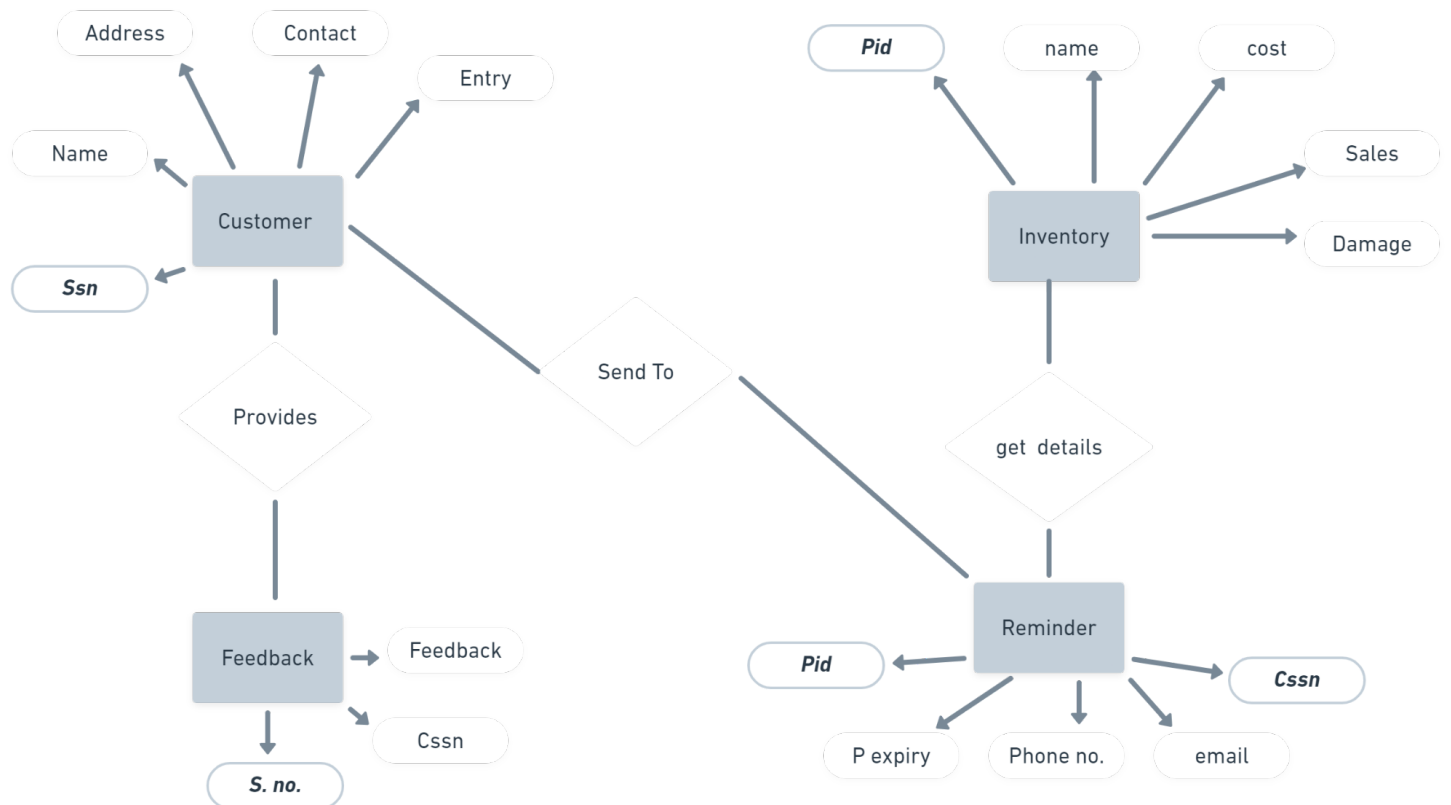
Fig 5

Relational Database Mapping

Customer:

Ssn	Name	Address	Contact	Entry

Inventory:

Pid	Name	Cost	Sales	Damage

Feedback:

S. No.	Cssn	Feedback

Reminder:

Pid	Cssn	P expiry	Phone No.	Email

Fig 6

9
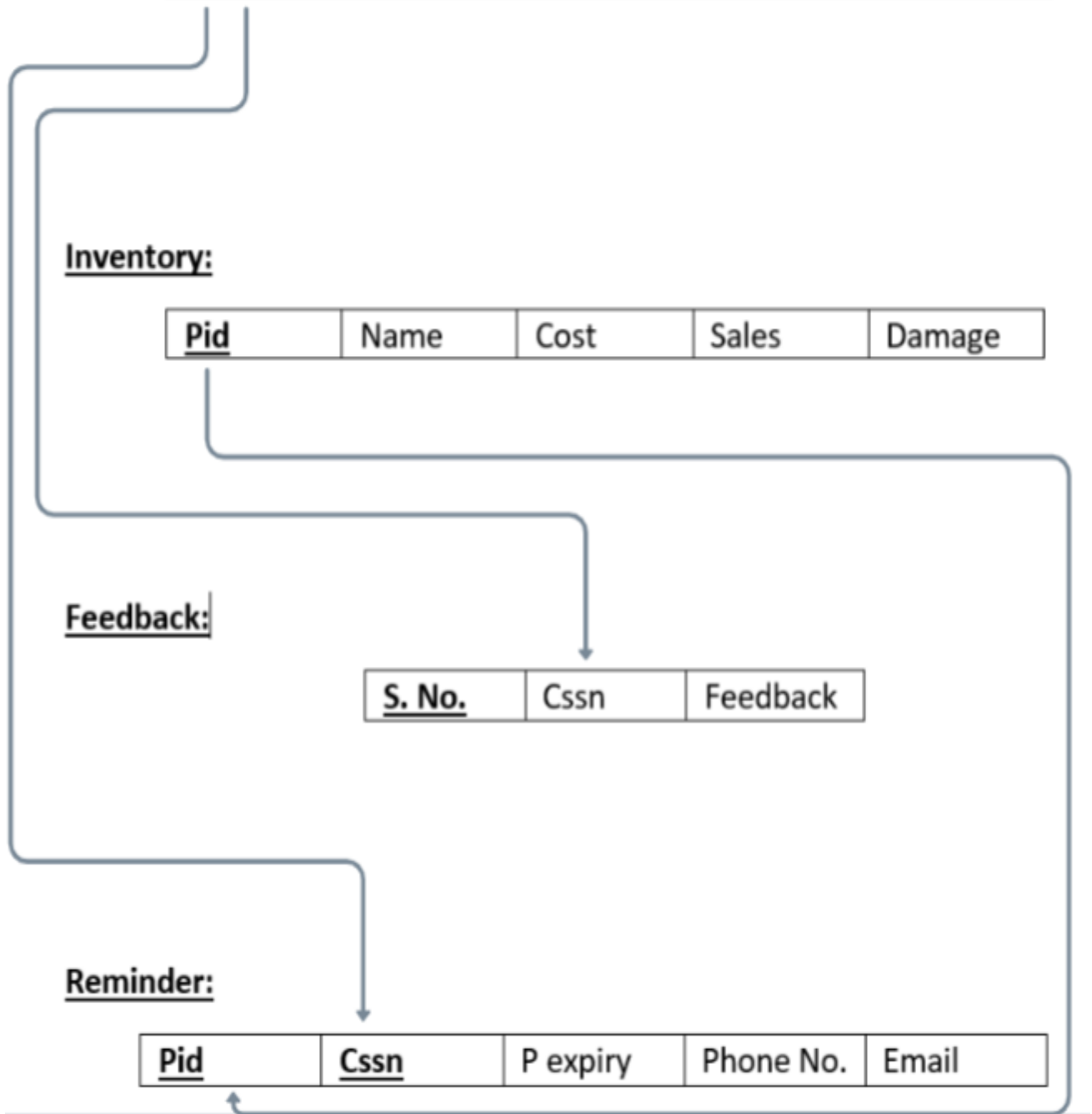
Database:

```
mysql> use contactless_store;
Database changed
mysql> show tables;
+---------------------------+
| Tables_in_contactless_store |
+---------------------------+
| customer                  |
| feedback                  |
| inventory                 |
| reminder                  |
+---------------------------+
4 rows in set (0.00 sec)

mysql> describe customer;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| Ssn     | int         | NO   | PRI | NULL    |       |
| Name    | varchar(50) | NO   |     | NULL    |       |
| Address | varchar(50) | NO   |     | NULL    |       |
| contact | bigint      | YES  |     | NULL    |       |
| Entry   | varchar(5)  | YES  |     | NULL    |       |
+---------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)

mysql> describe feedback;
+----------+--------------+------+-----+---------+-------+
| Field    | Type         | Null | Key | Default | Extra |
+----------+--------------+------+-----+---------+-------+
| Sno      | int          | NO   | PRI | NULL    |       |
| Cssn     | int          | NO   | MUL | NULL    |       |
| feedback | varchar(100) | YES  |     | NULL    |       |
+----------+--------------+------+-----+---------+-------+
3 rows in set (0.01 sec)

mysql> describe inventory;
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| Pid    | int         | NO   | PRI | NULL    |       |
| Name   | varchar(50) | NO   |     | NULL    |       |
| Cost   | float       | NO   |     | NULL    |       |
| sales  | int         | NO   |     | NULL    |       |
| damage | varchar(30) | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)

mysql> describe reminder;
+----------+-------------+------+-----+---------+-------+
| Field    | Type        | Null | Key | Default | Extra |
+----------+-------------+------+-----+---------+-------+
| Pid      | int         | NO   | PRI | NULL    |       |
| Cssn     | int         | NO   | PRI | NULL    |       |
| P_expiry | date        | YES  |     | NULL    |       |
| Phone_no | bigint      | YES  |     | NULL    |       |
| Email    | varchar(30) | YES  |     | NULL    |       |
+----------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```

Fig 7

10

```
mysql> select * from customer;
+-----------+------------------+---------+------------+-------+
| Ssn       | Name             | Address | contact    | Entry |
+-----------+------------------+---------+------------+-------+
|  31243523 | Crfre            | Delhi   | 2123371282 | NULL  |
| 312432423 | ABCSD            | Delhi   | 2198371282 | NULL  |
| 312434323 | KSHD             | Delhi   | 2123323242 | NULL  |
| 932232311 | Pushpender Kadian | Delhi  | 9896121341 | NULL  |
| 932232312 | Esha Dahiya      | Delhi   | 8721894273 | NULL  |
+-----------+------------------+---------+------------+-------+
5 rows in set (0.00 sec)

mysql> select * from feedback;
+-----+-----------+----------------------------+
| Sno | Cssn      | feedback                   |
+-----+-----------+----------------------------+
|   1 | 932232311 | Great Experiance of shopping |
|   2 | 932232312 | Amazing contactless store  |
+-----+-----------+----------------------------+
2 rows in set (0.02 sec)

mysql> select * from inventory;
+-----+------------+------+-------+--------+------------+
| Pid | Name       | Cost | sales | damage | P_expiry   |
+-----+------------+------+-------+--------+------------+
|   1 | Sanatizer  |  149 |     2 | NULL   | 2022-10-01 |
|   2 | Cold Drink |   90 |     2 | NULL   | 2022-09-12 |
|   3 | Biscuit    |   49 |     2 | NULL   | 2022-08-10 |
|   4 | Chocolate  |   40 |     1 | NULL   | 2022-06-12 |
+-----+------------+------+-------+--------+------------+
4 rows in set (0.02 sec)

mysql> select * from reminder;
+-----+-----------+------------+------------+-------------+
| Pid | Cssn      | P_expiry   | Phone_no   | Email       |
+-----+-----------+------------+------------+-------------+
|   1 | 932232311 | 2022-10-01 | 9896121341 | xyz@abc.com |
|   1 | 932232312 | 2022-10-01 | 8721894273 | xyz@abc.com |
|   2 | 932232311 | 2022-09-12 | 9896121341 | xyz@abc.com |
|   2 | 932232312 | 2022-09-12 | 8721894273 | xyz@abc.com |
|   3 | 932232311 | 2022-08-10 | 9896121341 | xyz@abc.com |
|   3 | 932232312 | 2022-08-10 | 8721894273 | xyz@abc.com |
|   4 | 932232311 | 2022-06-12 | 9896121341 | xyz@abc.com |
+-----+-----------+------------+------------+-------------+
7 rows in set (0.00 sec)
```

Fig 8

Timeline:-

When creating a software project schedule, the planner begins with a set of tasks. If automated tools are used, the work breakdown is input as a task network or task outline. Effort, duration, and start date are then input for each task. In addition, tasks may be assigned to specific individuals. A timeline chart, also called a Gantt chart, is generated. A timeline chart can be developed for the entire project. All project tasks are listed in the left-hand column. The horizontal bars indicate the duration of each task. When multiple bars occur at the same time on the calendar, task concurrency is implied. The diamonds indicate milestones.
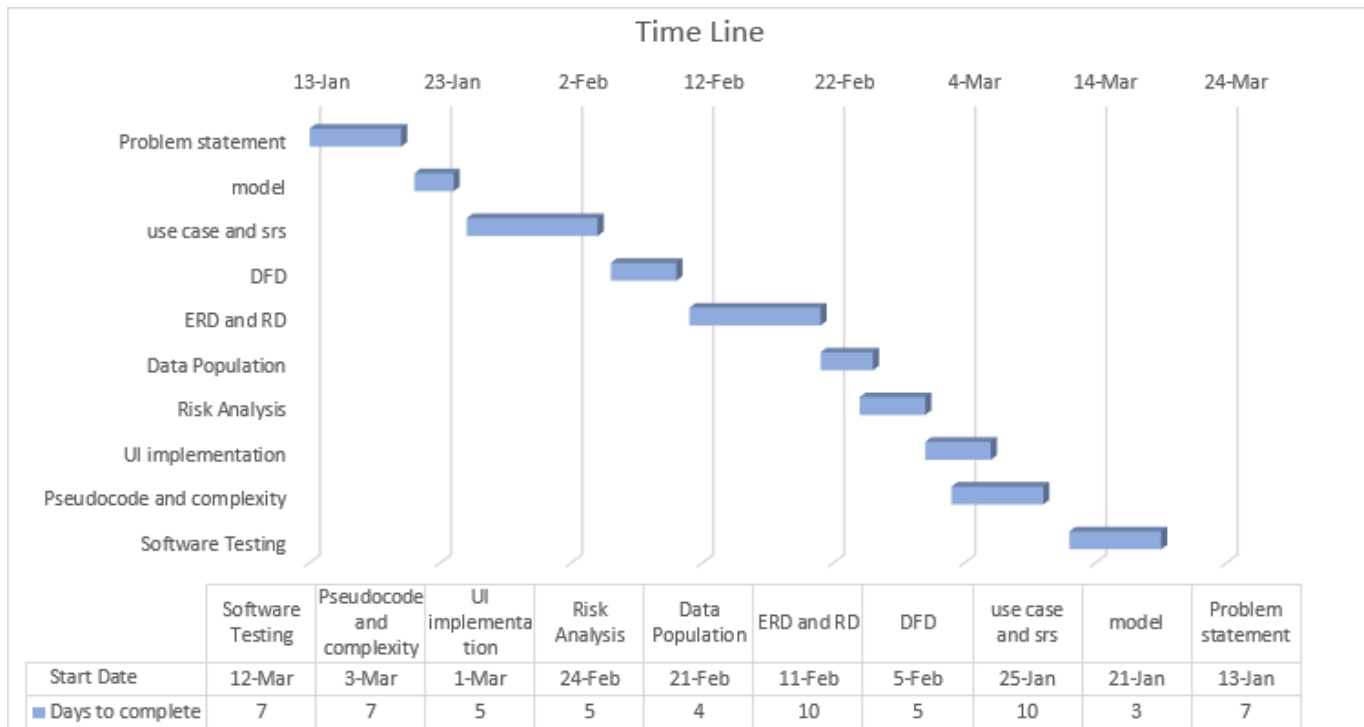

	Software Testing	Pseudocode and complexity	UI implementation	Risk Analysis	Data Population	ERD and RD	DFD	use case and srs	model	Problem statement
Start Date	12-Mar	3-Mar	1-Mar	24-Feb	21-Feb	11-Feb	5-Feb	25-Jan	21-Jan	13-Jan
■ Days to complete	7	7	5	5	4	10	5	10	3	7

Fig 9

12

GUI:-

Fig 10

Send Payment

X

INR 50.50

XYZ

Phone Number

Add More +

Product discription

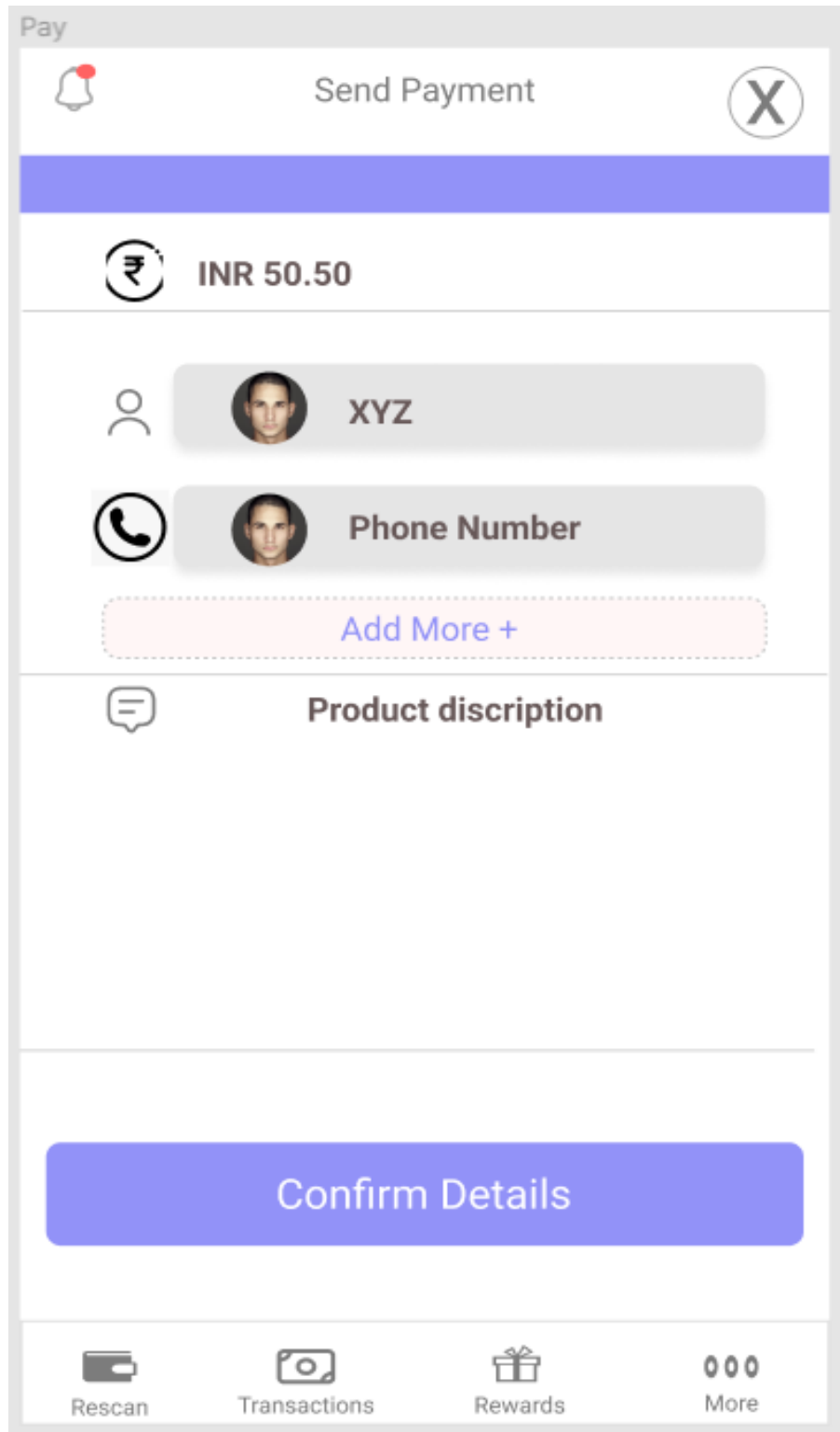Confirm Details

Rescan   Transactions   Rewards   More

Fig 11

Cyclomatic Complexity

Cyclomatic complexity is a software metric that provides a quantitative measure of the logical complexity of a program. When used in the context of the basis path testing method, the value computed of cyclomatic complexity defines the number of independent paths in the basis set of a program and provides you with an upper bound for the number of tests that must be conducted to ensure that all statements have been executed at least once.

Use of Cyclomatic Complexity:

 ● Determining the independent path executions thus proven to be very helpful for Developers and Testers.

● It can make sure that every path has been tested at least once.

● Thus help to focus more on uncovered paths.

● Code coverage can be improved.

● Risk associated with the program can be evaluated.

 ● These metrics being used earlier in the program helps in reducing the risks

Advantages of Cyclomatic Complexity:

● It can be used as a quality metric, giving relative complexity of various designs.

● It is able to compute faster than Halstead's metrics.

● It is used to measure the minimum effort and best areas of concentration for testing.

● It is able to guide the testing process.

 ● It is easy to apply.

Cyclomatic Complexity can be computed in 3 ways:

1. Number of regions in the flow graph is cyclomatic complexity.

2. Cyclomatic complexity for a flow graph G is, $V(G) = E - N + 2$, where 'E' is no. of flow graph edges and 'N' is no. of flow graph nodes.

3. Cyclomatic complexity for a flow graph G is, $V(G) = P + 1$, where 'P' is no. of predicate nodes in graph G.

Pseudocode

<u>For entry checking module:</u>

1. eye_print = customer.get();
2. user = database.get(Ssn == customer.Ssn);
3. if(user.exist()){
4. database.update.set(entered=true).where(customer.Ssn==user.Ssn);

   }else{

5. Output("User Doesn't Exist");

   }

6. return 0;

Control Flow Graph

A Control Flow Graph (CFG) is the graphical representation of control flow or computation during the execution of programs or applications. Control flow graphs are mostly used in static analysis as well as compiler applications, as they can accurately represent the flow inside of a program unit.
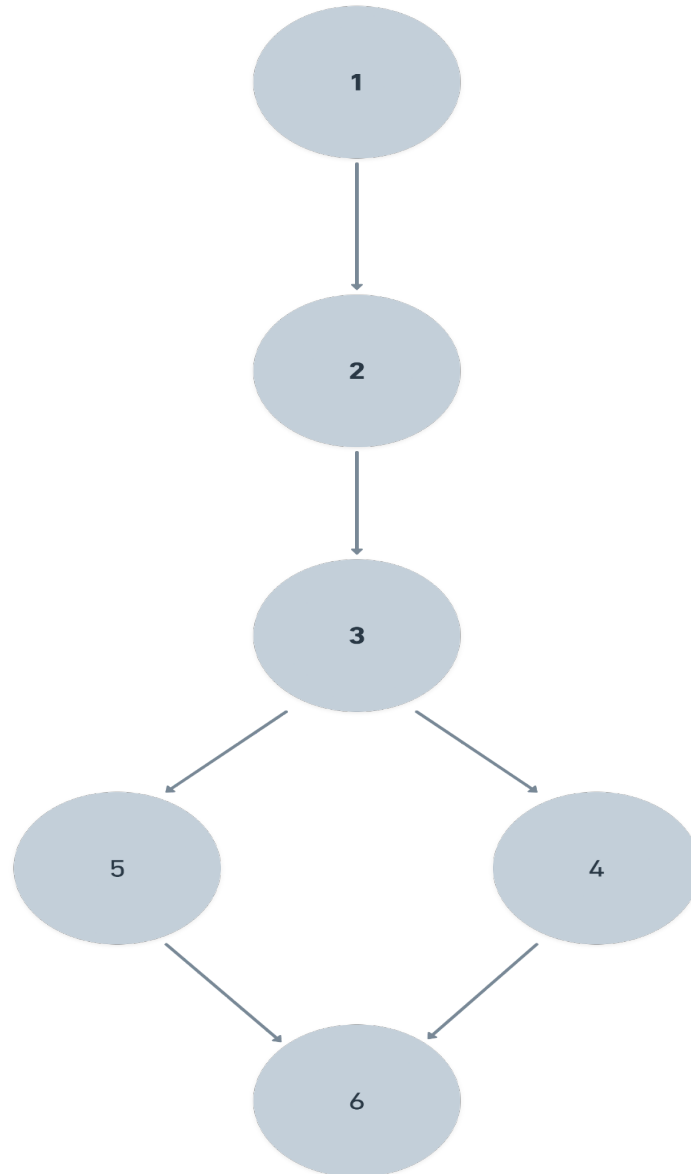

Fig 12

Cyclomatic Complexity = Number of Edges - Number of Nodes +2

Entry checking:

Number of Edges = 6

Number of Nodes = 6

Cyclomatic Complexity(SignUp) = 6-6+2 = 2

Number of closed regions = 1

Cyclomatic Complexity(SignUp) = 1+1(outside region) =2

TESTING

- Testing is the process of executing a program with the intent of finding an error. A good test case has a high probability of finding - undiscovered error.
- Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design, coding.
- The purpose of product testing is to verify and validate the various work products viz. units, integrated unit, final product to ensure that they meet their requirements.

Black Box Testing:

- Black Box testing is also called functional testing.
- Black Box Testing is a test case design method that focuses on the functional requirements of the software that enables the software engineer to derive a set of input conditions that fully exercise all functional requirements for a program.
- Test the artifacts from the external point of view.
- Specifications are used to test data that is what type of input should be given to the unit or module should be specified.
- We can check the functionality on the basis of the output generated and the input, not looking at the internal coding.
- It attempts to find errors in the following categories
  1. Incorrect or missing functions
  2. Interface errors
  3. Errors in data structure or External database access
  4. Behavior or performance error
  5. Initialisation and termination errors

White Box Testing:

- It is also called glass box testing.
- White Box testing is a test case design method that uses the control structure of the procedural design to derive test cases.
- Using White Box Testing method, the software engineer can derive test cases that
  1. Guarantee that all independent paths within a module have been exercised at least once.
  2. Exercise all logical decisions on their true and false sides.

19

3. Execute all loops at their boundaries and within their operational bounds.
4. Exercise internal data structures ensure their validity.
   - ○ Test the artifacts from the internal point of view.
   - ○ It cannot detect absence of features.
   - ○ For security purposes the Email of the user is required in case he/she forgets his/her password and wants to retrieve that.

Test Cases

Entry Module:

S.No.	Test Case	Expected Result
1	Successful Eye print scan	Access Granted
2	Unsuccessful Eye print scan	Access Denied

Payment Module:

S.No.	Test Case	Expected Result
1	Successful QR scan	Payment permission granted
1.1	Payment successful	Access to product granted
1.2	Payment unsuccessful	Access to product denied
2	Unsuccessful QR scan	Payment permission denied

CONCLUSION

The project on "Contactless Store" undertaken by us at "PGDAV College,University Of Delhi", helped us to create such a project which helps the people to shop physically in a safe environment as the process of shopping becomes completely contactless.

In these pandemic times, our store helps in preventing the spread of coronavirus infection and simultaneously enables people to get the satisfaction of shopping physically.

We have been successful in our attempt to take care of the needs of the general public. Finally, we hope that this will create a revolution in shopping physically as the process for the owner as well as the customer becomes hassle-free.

Bibliography

[1] Pressman, R. S., & Maxim, B. R. (2015). Software Engineering: A Practitioner's Approach. 8th edition. McGraw-Hill.

[2] Aggarwal, K. K., & Singh, Y. (2007). Software Engineering. 3rd edition. New Age International Publishers.