**A**

**PROJECT REPORT(UDP)**

**ON**

**“SenseShop”**

**SUBMITTED BY:**

Chauhan Harsh (150133107005)

Gandha Bhavin(150133107008)

**TEAM ID: *4004***

***IN FULFILLMENT FOR THE AWARD OF THE DEGREE***

***OF***

**BACHELOR OF ENGINEERING**

***IN***

**COMPUTER ENGINEERING**

****

**GOVERNMENT ENGINEERING COLLEGE, GANDHINAGAR**

**GUJARAT TECHNOLOGICAL UNIVERSITY,**

**AHMEDABAD**

**GOVERNMENT ENGINEERING COLLEGE, GANDHINAGAR**

**GUJARAT TECHNOLOGICAL UNIVERSITY,**

**AHMEDABAD**



**CERTIFICATE**

This is to certify that the project entitled **SenseShop** is a bonafied report of the work carried out by **Chauhan Harsh(150133107005)** and **Gandha Bhavin(150133107008)** under the guidance of **Internal guide – Prof. M.B. Chaudhry** for the successful completion of project, Computer Engineering at **Government Engineering College, Gandhinagar, Gujarat.**

To the best of my knowledge and belief, this work embodies the work of candidate herself/himself, has duly been completed, fulfils the requirement of the ordinance relating to the completion of project and is up to the standard in respect of content, presentation and language for being referred to the examiner.

M.B. Chaudhry B.V. Buddhadev

(Internal Guide) (HOD CE-IT)

**ACKNOWLEDGEMENT**

I would like to express my deep feeling of gratitude to the under mentioned officials for their assistance, guidance and inspiration throughout the project. I am very thankful to my college “Government Engineering College, Gandhinagar”, for providing best facilities. I am highly grateful to the college to give me an opportunity to have experience in working with such an experienced Institute. I would like to express my sincere thanks to who showed trust in me and provide me with such a challenging project. I would also like to thank my internal project guide “Prof. M.B. Chaudhry” for spending their valuable time for providing me better guidance to achieve my goal, I want to express my deep appreciation for the continuous support and encouragement from my faculty members. Last but not the least, I would also thank all those who have helped me implicitly or explicitly but whose names have forgotten to mention.

**Thanking you,**

Your’s sincere Students

**Chauhan Harsh[150133107005]**

**Gandha Bhavin[150133107008]**

**ABSTRACT**

**SenseShop** is a web based application that provides a very convenient way of shopping to the customer for faster checkouts and easy payments using various type of sensors. Main goal for system is to save the valuable time of customer for purchasing of the daily needs. SenseShop system can also be used for the inventory management and the Warehouse management of the goods so that it can be efficient and helpful for the customer as well as the Retailer.

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Figure Description** | **Page No** |
| 1 | Fig 2.2.2: Risk Management | 3 |
| 2 | Fig 4.1.1: E-R Diagram | 10 |
| 3 | Fig 4.2.1: Customer Use Case Diagram | 13 |
| 4 | Fig 4.2.2: Admin Use Case Diagram | 14 |
| 5 | Fig 4.3.1: 0-Level DFD Diagram | 16 |
| 6 | Fig 4.3.2: Admin 1st Level DFD Diagram | 16 |
| 7 | Fig 4.3.3: Customer 2nd Level DFD Diagram | 17 |
| 8 | Fig 4.4.1: Customer Activity Diagram | 18 |
| 9 | Fig 4.4.2: Admin Activity Diagram | 19 |
| 10 | Fig 4.5.1: Class Diagram | 20 |
| 12 | Fig 4.6.1: Customer Sequence Diagram | 21 |
| 13 | Fig 4.6.2: Admin Sequence Diagram | 22 |
| 14 | Fig 6.1: Water Fall Model | 30 |
| 15 | Fig 6.2: TimeLine Chart | 30 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Table Description** | **Page No** |
| 1 | Table 3.2.1: Hardware Requirements | 7 |
| 2 | Table 3.2.2: Software Requirements | 7 |
| 3 | Table 4.1.1: Symbols used in E-R Diagram | 10 |
| 4 | Table 4.2.1: Symbols used in Use Case Diagram | 12 |
| 5 | Table 4.3.1 Symbols used in DFD | 15 |
| 6 | Table 5.1: Customer | 23 |
| 7 | Table 5.2: Payment | 24 |
| 8 | Table 5.3: Temp\_cart | 24 |
| 9 | Table 5.4: Order\_details | 25 |
| 10 | Table 5.5: Admin | 25 |
| 11 | Table 5.6: Product | 26 |
| 12 | Table 5.7: Stock | 26 |
| 13 | Table 5.8: Warehouse\_stock | 27 |
| 14 | Table 5.9: Retail\_stock | 27 |
| 15 | Table 5.10: Complaint | 28 |
| 16 | Table 5.11: Feedback | 28 |

**INDEX**

1. **INTRODUCTION…………………………………………………………………....1**
   1. Project Profile…...…………………………………………………….…………..1
   2. Objective of Work…………………………………………………………….…..1
   3. Scope of Work………………………………………………………………….....2
   4. Technology Used……………………………………………………………….....2
   5. Comparisons with Existing System…………………………………………….....2
2. **PROJECT MANAGEMENT……………………………………………………… .3**
   1. Milestones and Deliverables……………………………………………………...3
   2. Risk Management………………………………………………………………...3
   3. Risk Identification………………………………………………………………..4
   4. Risk Analysis…………………………………………………………………….4
   5. Risk Planning………………………………………………………………….....4
3. **SYSTEM ANALYSIS AND REQUIREMENT …………………………………...5**
   1. System Feasibility………………………………………………………………..5
   2. Requirement of System………………………………………………………….7
   3. Functional requirements…………………………………………………………8
   4. Non-Functional requirements…………………………………………………....9
4. **SYSTEM DESIGN…………………………………………………………………10**
   1. E-R Diagram……………………………………………………………………10
   2. Use Case Diagram…………………………………………………………...…12
   3. Data flow diagram……………………………………………………………...15
   4. Activity Diagram……………………………………………………………….18
   5. Class Diagram…………………………………………………………………..20
   6. Sequence Diagram……………………………………………………………...21
5. **DATA DICTIONARY……………………………………………………………..23**
6. **PLANNING AND IMPLEMENTATION………………………………………..29**
7. **FUTURE ENHANCEMENT………………………………………………………31**
8. **REFERENCES……………………………………………………………………..32**
9. **APPENDIX….……………………………………………………………………...33**
   1. Canvas………………………………….……………….…………………..…..33
   2. Periodic Progress Report……………………………………………………......36
   3. Patent Search and Analysis Report…………………………………………...…40

**CHAPTER:-1 INTRODUCTION**

* 1. **PROJECT PROFILE :-**

|  |  |
| --- | --- |
| **Project Title** | **SenseShop** |
| **Project Type** | **Web Application** |
| **Objective** | **“To Provide Easy, Fast and seamless retail shopping experience.”** |
| **Front End** | **Python/Web** |
| **Back End** | **Python/Django** |
| **Institute Name** | **Government engineering college, Gandhinagar** |
| **Internal Guide** | **Professor M.B. Chaudhry** |
| **Development Team** | **Chauhan Harsh**  **Gandha Bhavin** |

* 1. **OBJECTIVE OF WORK :-**

This solution is for the retail shops to give a generalized and best solution for the easy and cost-effective automation for many customers who don’t want to waste time for shopping everyday things and also helps the Retail merchants for maintaining the sales as well as stocks.

* 1. **SCOPE OF WORK :-**

**Customers:**

The scope of this solution for customers is any person who is involved in Purchase of any everyday things from a retail shop.

**Retail Shop Owners:**

The scope of this solution for merchants who is having a shop and want to maintain it effortlessly.

* 1. **TECHNOLOGY USED :-**

**Front-End :**

* Python/Web

**Back-End :**

* Python/Django
  1. **COMPARISONS WITH EXISTING SYSTEM :-**

In the current flow when a person wants to buy a things from a Retail Shop they need to scan and verify the purchase for billing by a merchant. Sometimes there can be huge queues for the scanning and billing which is a time consuming process. Same for the merchants they have to verify and needs to track the stock of a particular product.

In proposed system when the person wants to buy a things form a Retail shop they don’t need to scan and verify the purchase form billing by a merchant because all the process will be executed thru sensors by our system. It also tracks the current Stock of the particular product form the shop as well as Wearhouse.

**CHAPTER:-2 PROJECT MANAGEMENT**

**2.1 MILESTONES AND DELIVERABLES :-**

* Secure and Reliable payment method.
* Set triggers for low stock.
* Tracks the previous orders.
* Customers can file the complaints.
* Precision tracking of sales.

**2.2 RISK MANAGEMENT :-**

All projects involve risks. If a potential risk of the project is not identified early, then the project will be at a high risk to complete as per schedule, within budget and to meet the expected quality. One of the current difficulties faced by a new Project Manager today is not having a sample or general risk list to refer to when identifying the project risk



Fig 2.2.2: Risk Management

**2.3 RISK IDENTIFICATION :-**

After establishing the context, the next step in the process of managing [risk](http://en.wikipedia.org/wiki/Risk) is to identify potential risks. Risks are about events that, when triggered, cause problems. Hence, risk identification can start with the source of problems, or with the problem itself.

Project Risk identification is the most important process in the Risk Management Planning. Risk Identification determines which risks might affect the project and documents their characteristics. However, as recommended by [Donna Ritter], we should not spend too much time in identifying risks.

**2.4 RISK ANAYLSIS :-**

Regardless of the prevention techniques employed, possible threats that could arise inside or outside the organization need to be assessed. Although the exact nature of potential disasters or their resulting consequences are difficult to determine, it is beneficial to perform a comprehensive risk assessment of all threats that can realistically occur to the organization.

**2.5 RISK PLANNING :-**

To mitigate the risks, project management must develop a strategy for reducing turnover. Among the possible steps to be taken are:

Meet with current staff to determine causes for turnover (e.g. poor working conditions, low pay competitive job market).

Mitigate those causes that are under our control before the project starts.

Once the project commences, assume turnover will occur and develop techniques to ensure continuity when people leave.

Organize project teams so that information about each development activity is widely dispersed.

Define documentation standards and establish mechanisms to be sure that documents are developed in a timely manner.

**CHAPTER:-3 SYSTEM ANAYLSIS & REQUIREMENT**

**3.1 SYSTEM FEASIBILTY :-**

* Feasibility is the measure of how beneficial the development of an information system will be to an organization. A process by which we measure feasibility is called as the feasibility analysis.
* The Feasibility analysis is categorized under three different types:

1. Operational Feasibility.
2. Economical Feasibilty.
3. Technical Feasibililty.

* **Operational Feasibility**

The app should be easy to operate by the user. Proposed projects are beneficial only if they can be turned into information systems that will meet the user’s requirements. Following are some points underlying the operational feasibility of the app.

* Customers can operate the System easily.
* Customers can make transaction easily
* Customers can visit previous order for reference.
* **Economical Feasibility**

According to the concept of economical study, the system should be completed with minimum cost. And tools which we used is open source so, the proposed system is cost effectiveness. This app is in range of budget which is assumed or declared for developing it. So, it is economical feasible. In this budget the cost of hardware and software for the system is being considered.

* **Requirement Validation :**

This project contains the following validations: These validations include input validation for the members or the users while registration.

* At the time of Login, Email & Password must be required.
* The email id should contain symbols like ‘@’ and ‘.’
* Compare validation is used to compare and confirm the password.
* Some fields are compulsory to be filled by the members.
* **Technical Feasibility**

Technically the app’s GUI should be less complex. Technical feasibility means technically app should be comfortable for further versions. The proposed app has technical capacity to hold the data and provide technical support which is expected. The new app provides adequate responses to inquiries of all users. The System has technical guarantees of accuracy, reliability, ease of access and data security.

* The System is very much open by nature and can easily be expanded in the near future to satisfy newly emerging needs.
  1. **REQUIREMENTS OF SYSTEM :-**
* **Hardware Requirements:-**

|  |  |
| --- | --- |
| **Hardware** | **Recommended System Requirements** |
| **RAM** | <512 MB> |
| **INTERNAL STORAGE** | <256 MB> |

Table 3.2.1: Hardware Requirements

* **Software Requirements :-**

|  |  |
| --- | --- |
| **Front End** | <Python/Web> |
| **Back End** | <Python/Django> |
| **Operating System** | <Windows > |
| **Programming Language** | Python |

Table 3.2.2: Software Requirements

**3.3 FUNCTIONAL REQUIREMENTS :-**

* Each Module should Register the System after that they can login into the System.
* Each Module can Logout the system.
* Each Module can modify Personal Information.
* Admin can modify the products.
* Admin can manage the stocks.
* Customers view previous orders.
* Customers can file complaint.
* Customer can give feedback.

**3.4 NON-FUNCTIONAL REQUIREMENTS :-**

**Security:**

* Only authorized person can view and modify confidential details.
* Each User has his/her own access rights and to lower level user there are restrictions of not using some of the system components.
* One user can’t access other user’s account.
* User should compulsory fill right details.

**Privacy:**

* As a control, Privacy means that your actions cannot be seen or interpreted by anyone who is not supposed to know.
* Any person will be not able to change or modify information.

**Scalability:**

* Performance of the system should not be degraded when many users are logged-in and accessing the database at the same time.

**Reliability:**

* The database of various modules are maintained by system should be correct and maintained up to date.

**CHAPTER:-4 SYSTEM DESIGN**

**4.1 E-R DIAGRAM :-**

The **Entity-Relationship(ER) model** is a conceptual data model, capable of describing the data requirements for a new information system in a direct ,easy to understand graphical notation, to express their understanding of what the planned database is intended to do and how it might work, and to communicate about the database through a common language.

|  |  |
| --- | --- |
|  | Ellipse  -shows attribute . |
|  | Rectangle  -Entity class |
|  | Double ellipse  -multivalued attribute |
|  | Dotted ellipse |
|  | A Decision Node: Is used to represent a test condition to ensure that the control flow or object flow only goes down one path. |
|  | A line is used to connect attributes , decision and so on. |
|  | A line is used for one to many. |
|  | This line describe as one to many. |

Table 4.1.1: Symbols used in E-R Diagram

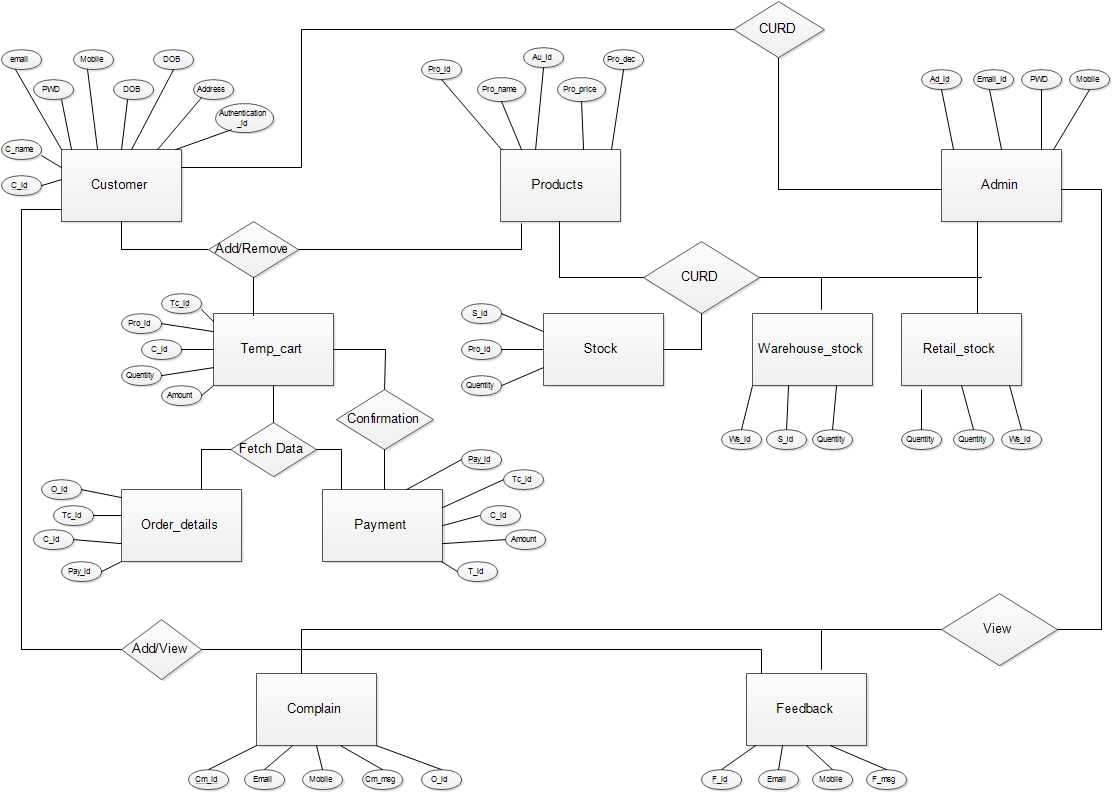


Fig 4.1.1: E-R Diagram

**4.2 USE CASE DIAGRAM :-**

|  |  |  |
| --- | --- | --- |
| Symbols | Represent as | Description |
|  | Frame or boundary | To create system environment |
|  | Use case | entity or process are  filled |
|  | Arrow | It gives the direction from one to another |
|  | Dotted arrow | Extending use case is optional, supplementary. |
|  | Actor | It shows the module of the system. |

Table 4.2.1: Symbols used in Use Case Diagram

**Customer:**

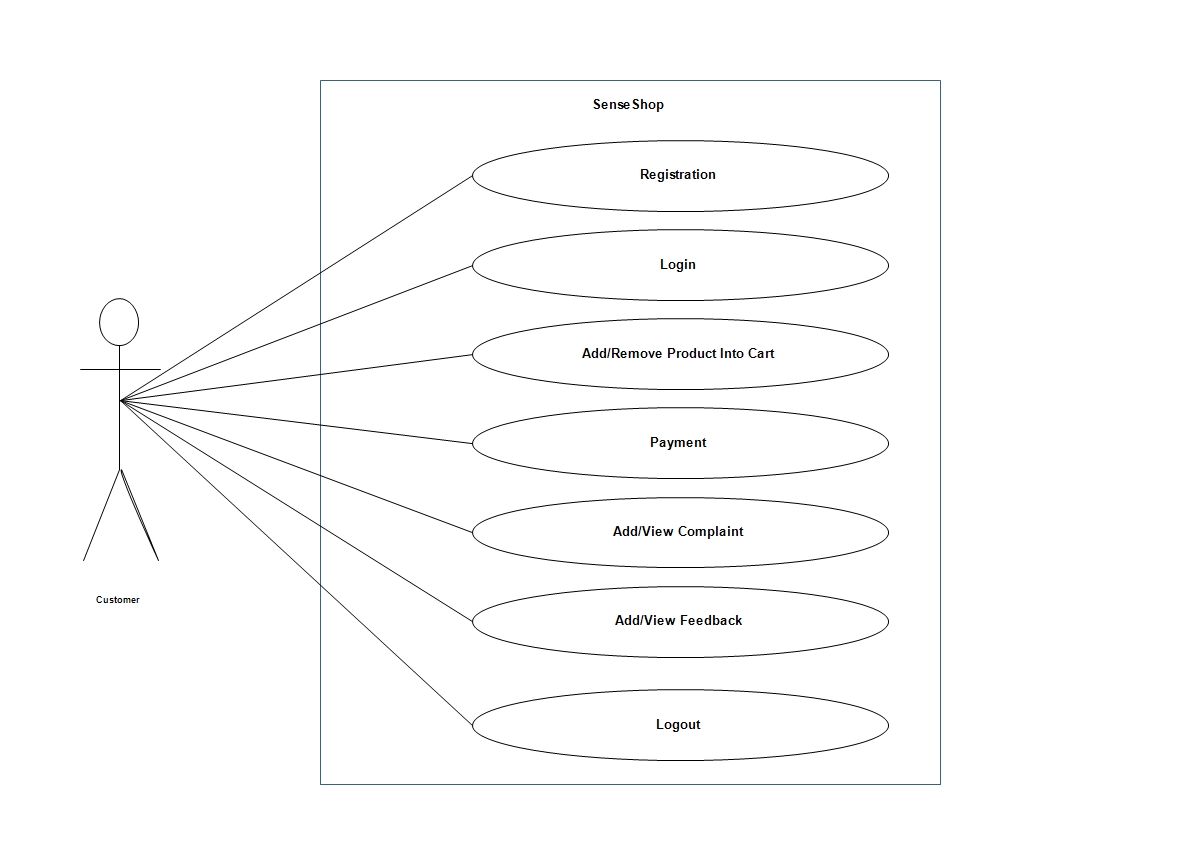
****

Fig 4.2.1: Customer Use Case Diagram

**Admin**

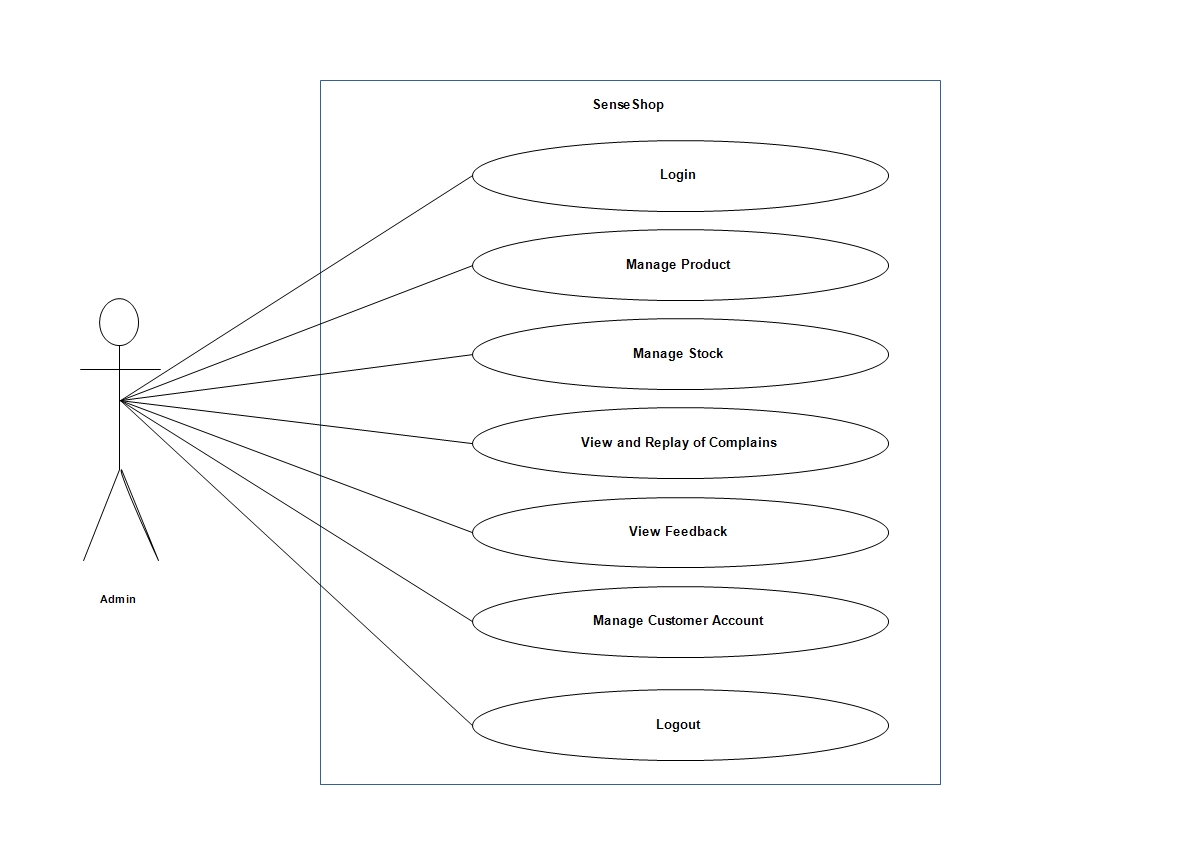
****

Fig 4.2.2: Admin Use Case Diagram

**4.3 DATA FLOW DIAGRAM :-**

A **Data-Flow Diagram (DFD)** is a graphical representation of the "flow" of data through an information system. DFDs can also be used for the visualization of data processing.

|  |  |
| --- | --- |
| Symbols | Represent as |
|  | Function |
|  | Database |
|  | Input/output |
|  | Flow |

Table 4.3.1: Symbols used in DFD

**Context Level Diagram:**

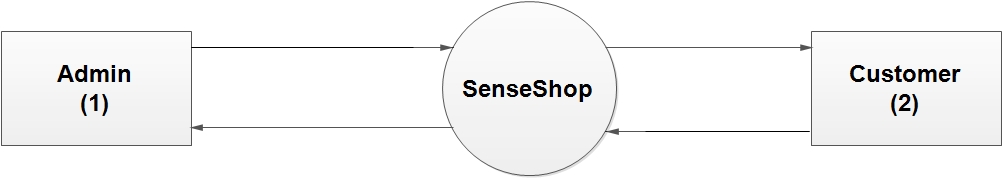


Fig 4.3.1: 0-Level DFD Diagram

**1st-Level DFD:**

****

Fig 4.3.2: Admin 1st Level DFD Diagram

**2nd-Level DFD:**

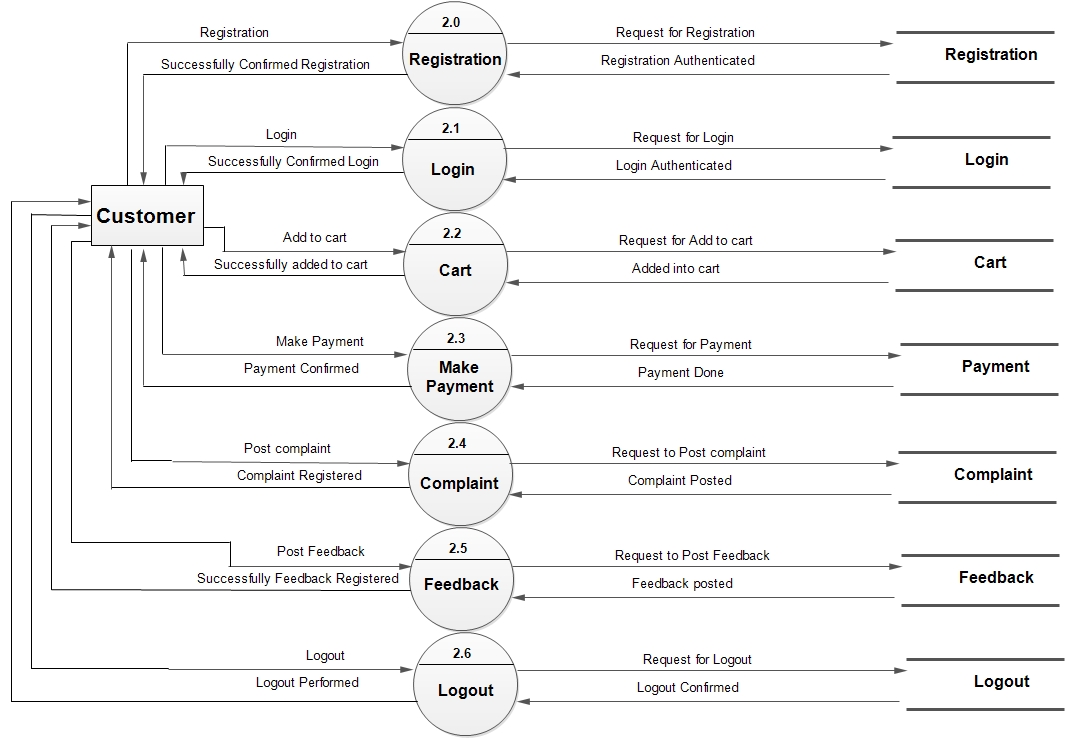
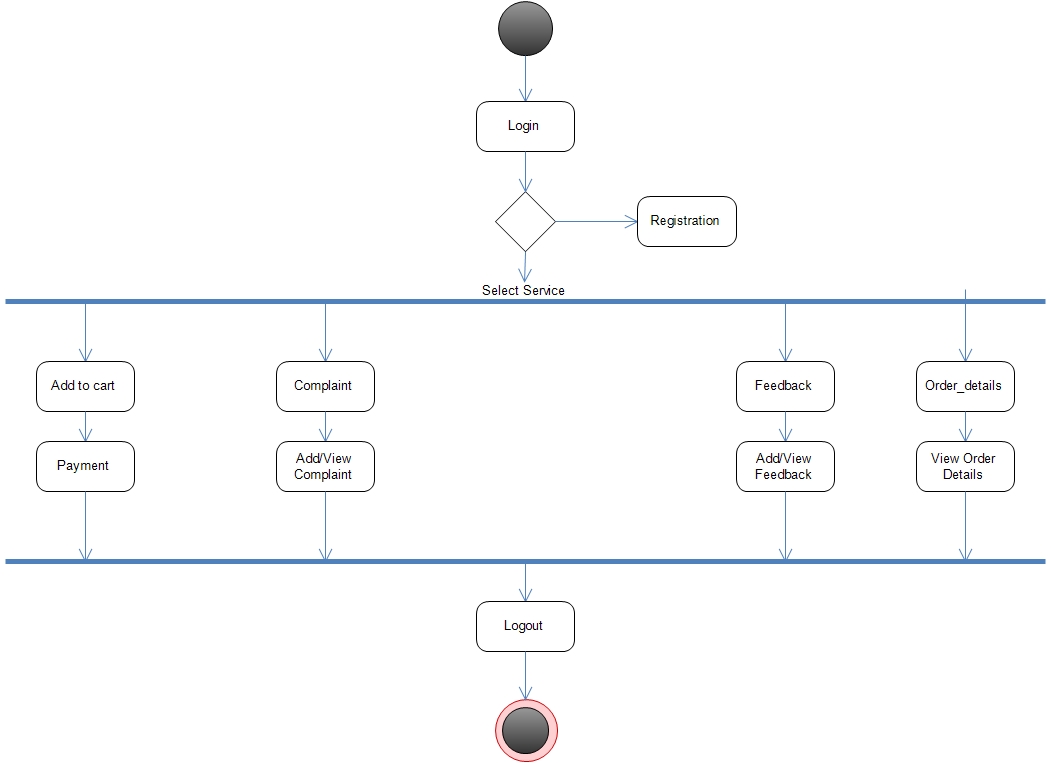


Fig 4.3.3: Customer 2nd Level DFD Diagram

**4.4 ACTIVITY DIAGRAM :-**

Activity diagrams, which are related to program flow plans (flowcharts), are used to illustrate activities. In the external view, we use activity diagrams for the description of those business processes that describe the functionality of the business system. Contrary to use case diagrams, in activity diagrams it is obvious whether actors can perform business use cases together or independently from one another.

**Customer**

****

Fig

4.4.1: Customer Activity Diagram

**Admin**

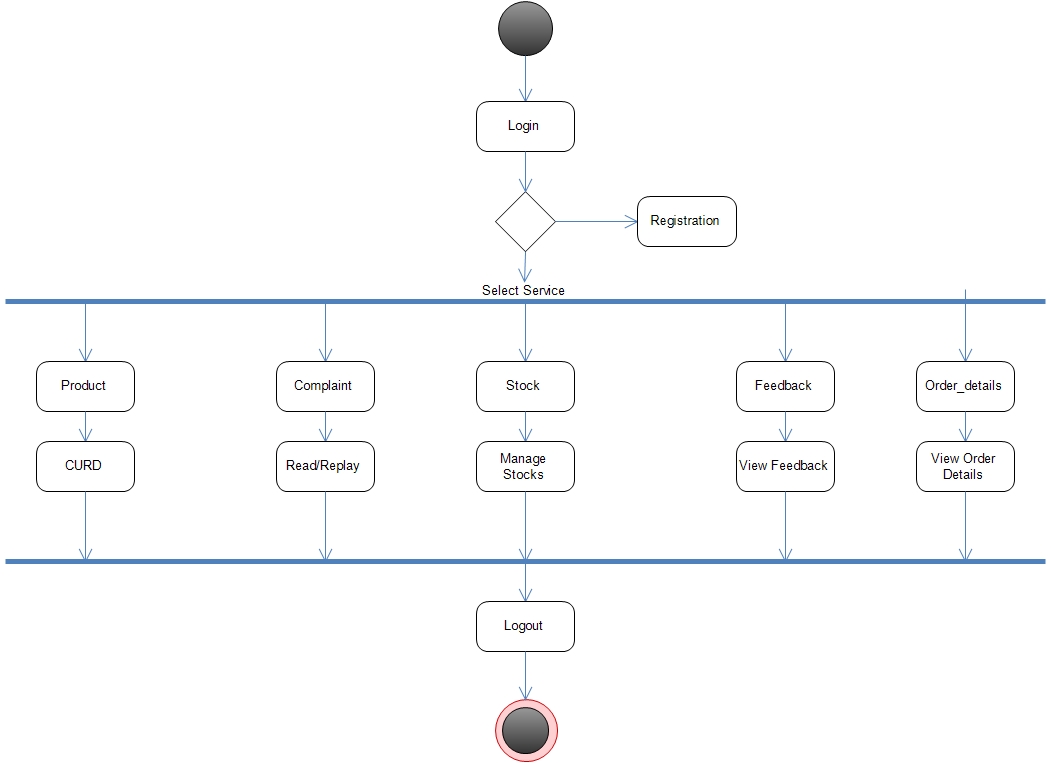


Fig 4.4.2: Admin Activity Diagram

**4.5 CLASS DIAGRAM :-**

The **Class Diagram** is a static diagram. It represents the static view of an application.A Class Diagram is an illustration of the relationships and [source code](http://searchsoa.techtarget.com/definition/source-code) dependencies among classes in the Unified Modeling Language (UML). In this context, a [class](http://whatis.techtarget.com/definition/class) defines the [method](http://searchcio-midmarket.techtarget.com/definition/method)s and [variable](http://whatis.techtarget.com/definition/variable)s in an [object](http://searchsoa.techtarget.com/definition/object), which is a specific entity in a program or the unit of code representing that entity. Class diagrams are useful in all forms of object-oriented programming (OOP).

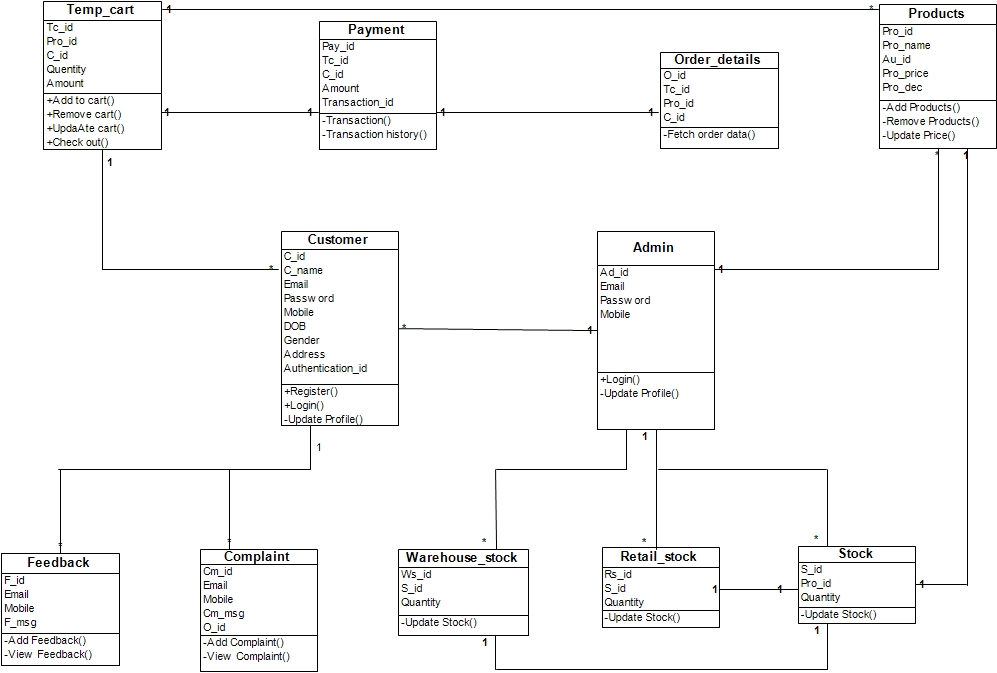
****

Fig 4.5.1: Class Diagram

**4.6 SEQUENCE DIAGRAM** :-

The Sequence Diagram models the collaboration of objects based on a time sequence. It shows how the objects interact with others in a particular scenario of a use case. With the advanced visual modeling capability, you can create complex sequence diagram in few clicks. Besides, VP-UML can generate sequence diagram from the flow of events which you have defined in the use case description.

**Customer**



Fig 4.6.1: Customer Sequence Diagram

**Admin**

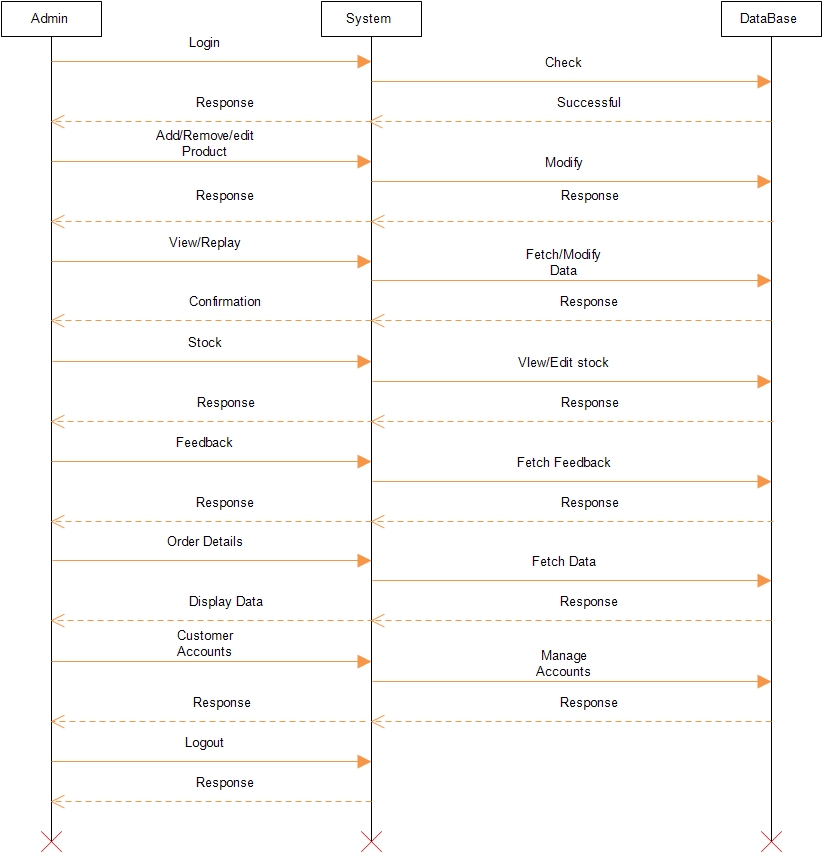
****

Fig 4.6.2: Admin Sequence Diagram

**CHAPTER:-5 DATA DICTIONARY**

A **Data Dictionary** is a catalog – a repository – of the elements in a system. As the name suggests, these elements centre on data the way they are structured to meet user requirements and organization needs. In a data dictionary you will find a list of all the elements composing the data flowing through a system. The major elements are data flows, data stores, and processes. The data dictionary stores details and descriptions of these elements.

|  |  |  |  |
| --- | --- | --- | --- |
| **Customer** | | | |
| **Field** | **Type** | **Constraints** | **Description** |
| C\_id | Int | Primary Key | Id of Customer |
| C\_name | Varchar(30) | Not Null | Name of Customer |
| Email | Varchar(20) | Not Null | Email Id of Customer |
| Password | Varchar(20) | Not Null | Password of Customer |
| Mobile\_no | Int(10) | Not Null | Mobile No of Customer |
| DOB | Varchar(30) | Not Null | Date of Birth |
| Gender | Varchar(10) | Not Null | Gender of Customer |
| Address | Varchar(20) | Not Null | Address of Customer |
| Authentication\_id | Varchar(20) | Not Null | Identity of Customer |
| Flag | Int | Not Null | For defining Flags |

Table 5.1: Customer

|  |  |  |  |
| --- | --- | --- | --- |
| **Payments** | | | |
| **Field** | **Type** | **Constraints** | **Description** |
| Pay\_id | Int | Primary Key | Id of Payment |
| Tc\_id | Int | Foreign Key | Id of Temporary Cart |
| C\_id | Int | Foreign Key | Id of Customer |
| Amount | Int | Not Null | Total amount of payment |
| Flag | Int | Not Null | For defining flags |
| Transaction\_id | Int | Unique | Id of Transaction |

Table 5.2: Payments

|  |  |  |  |
| --- | --- | --- | --- |
| **Temp\_cart** | | | |
| **Field** | **Type** | **Constraints** | **Description** |
| Tc\_id | Int | Primary Key | Id of Temporary Cart |
| Pro\_id | Int | Foreign Key | Id of Product |
| C\_id | Int | Foreign Key | Id of Customer |
| Quantity | Int | Not Null | Quantity of Product |
| Total\_amount | Int(10) | Not Null | Total Amount of Cart |
| Flags | Int | Not Null | For Defining flags |

Table 5.3: Temp\_cart

|  |  |  |  |
| --- | --- | --- | --- |
| **Order\_details** | | | |
| **Field** | **Type** | **Constraints** | **Description** |
| O\_id | Int | Primary Key | Id of Saloon |
| Tc\_id | Int | Foreign Key | Id of temporary cart |
| Pay\_id | Int | Foreign Key | Id of Payments |
| C\_id | Int | Foreign Key | Id of Customers |
| Flag | Varchar(50) | Not Null | For defining flags |

Table 5.4: Order Details

|  |  |  |  |
| --- | --- | --- | --- |
| **Admin** | | | |
| **Field** | **Type** | **Constraints** | **Description** |
| Ad\_id | Int | Primary key | Id of Admin |
| Email\_id | Varchar(30) | Not Null | Email of Admin |
| Password | Varchar(30) | Not Null | Password of Admin |
| Mobile\_no | Int(11) | Not Null | Mobile No of Admin |
| Flag | Int | Not Null | For defining flags |

Table 5.5: Admin

|  |  |  |  |
| --- | --- | --- | --- |
| **Products** | | | |
| **Field** | **Type** | **Constraints** | **Description** |
| Pro\_Id | Int | Primary Key | Id of product |
| Pro\_name | Varchar(30 | Not Null | Name of product |
| Au\_id | Varchar(30) | Not Null | For authentication purpose. |
| Pro\_price | Int | Not Null | Price of the Product |
| Pro\_dec | Varchar | Not Null | Description of product |
| Flag | Int | Not Null | For Defining flags |

Table 5.6: Products

|  |  |  |  |
| --- | --- | --- | --- |
| **Stock** | | | |
| **Field** | **Type** | **Constraints** | **Description** |
| S\_id | Int | Primary Key | Id of Total stock |
| Pro\_id | Int | Foreign Key | Id of product |
| Quantity | Int | Not Null | Quantity of Total stock |
| Flag | Int | Not Null | For Defining flags |

Table 5.7: Stock

|  |  |  |  |
| --- | --- | --- | --- |
| **Warehouse\_stock** | | | |
| **Field** | **Type** | **Constraints** | **Description** |
| Ws\_id | Int | Primary Key | Id of warehouse stock |
| S\_Id | Int | Foreign Key | Id of total stock |
| Quantity | Int | Not Null | Quantity of Stock in Wearhouse |
| Flag | Int | Not Null | For Defining flags |

Table 5.8: Warehouse\_stock

|  |  |  |  |
| --- | --- | --- | --- |
| **Retail\_stock** | | | |
| **Field** | **Type** | **Constraints** | **Description** |
| Rs\_id | Int | Primary Key | Id of Retail stock |
| S\_id | Int | Foreign Key | Id of total stock |
| Quantity | Int | Not Null | Quantity of stock in Retail |
| Flag | Int | Not Null | For Defining flags |

Table 5.9: Retail\_stock

|  |  |  |  |
| --- | --- | --- | --- |
| **Complaint** | | | |
| **Field** | **Type** | **Constraints** | **Description** |
| Cm\_id | Int | Primary Key | Id of Complaint |
| Email | Varchar(30) | Not Null | Email id |
| Monile | Int(11) | Not Null | Mobile |
| Cm\_msg | Varchar(250) | Not Null | Message of complaint |
| O\_id | Int | Foreign Key | Id of Order |
| Flag | Int | Not Null | For defining Flags |

Table 5.10: Complaint

|  |  |  |  |
| --- | --- | --- | --- |
| **Feedback** | | | |
| **Field** | **Type** | **Constraints** | **Description** |
| F\_id | Int | Primary Key | Seller Id of Seller Registration Table. |
| Email | Int | Foreign Key | Customer\_Id of Customer Registration Table. |
| Mobile | Int | Foreign Key | Saloon Id of Saloon Registration Table. |
| F\_msg | Int | Primary Key | Id of Advertisement. |
| Flag | Int | Not Null | For defining Flag. |

Table 5.11: Feedback

**CHAPTER:-6 PLANNING AND IMPLEMENTATION**

* **PROJECT PLANNING:-**

Project planning is a procedural step in project management, where required documentation is created to ensure successful project completion. Documentation includes all actions required to define, prepare, integrate and coordinate additional plans. The project plan clearly defines how the project is executed, monitored, controlled and closed.

* Identify objectives
* Determine information Requirements
* Analyse System Needs
* Designing
* Development
* Testing Deployment
* **PROJECT DEVELOPMENT APPROACH :**

The Waterfall Model was first Process Model to be introduced. It is also known as a linear-sequential life cycle model. It is simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases. The waterfall Model illustrates the software development process in a linear sequential flow; hence it is also referred to as a linear-sequential life cycle model. This means that any phase in the development process begins only if the previous phase is complete. In waterfall model phases do not overlap.



Fig 6.1: Water Fall Model

* **PROJECT SCHEDULING:-**

A project schedule provides a road map for a software project manager.

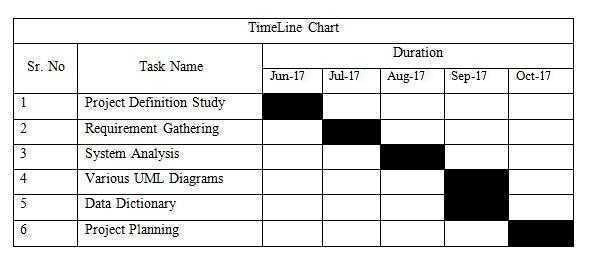


Fig 6.2: TimeLine Chart

**CHAPTER:-7 FUTURE ENHANCEMENT**

In future We are going to remove maximum limitations of the System. We will also try to reduce the size of the System And We will add new and new functionalities to our system and make it user-friendly and very attractive. We will try to make the application for the system also so that it can be more convenient.

**CHAPTER:-8 REFERENCES**

**Analysis of Project Requirements :-**

* For basic requirements and to know the processes we visited our Internal guide.

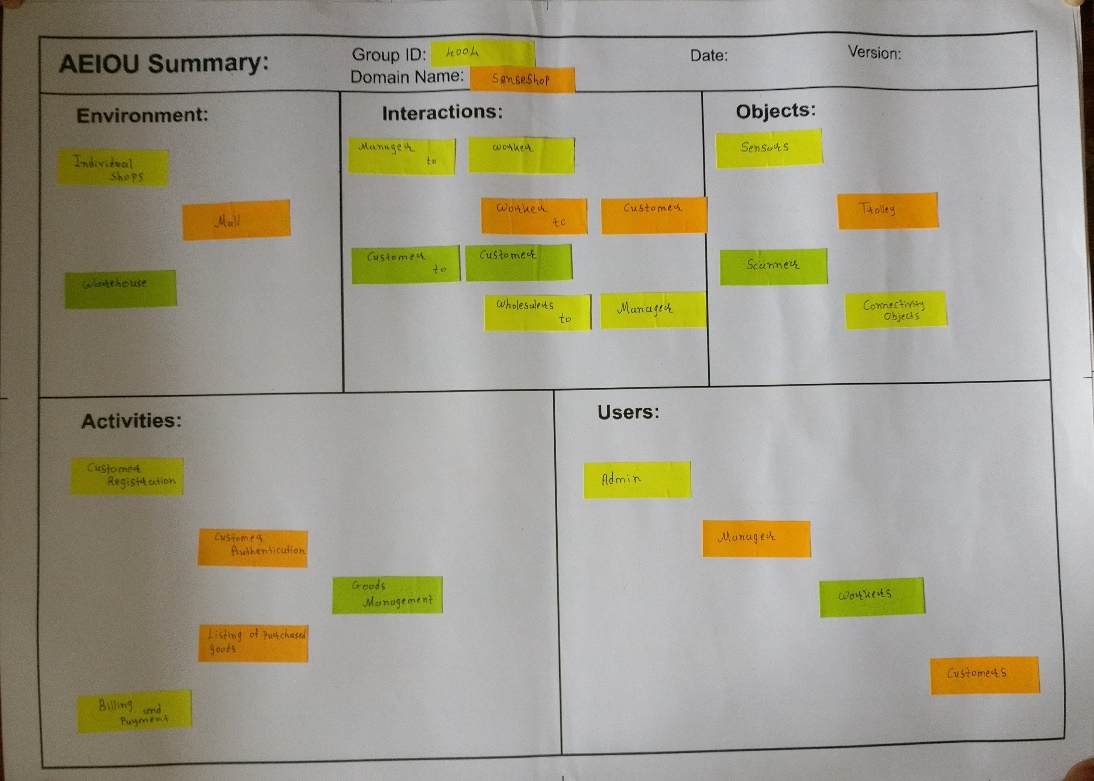
**Visited Sites :-**

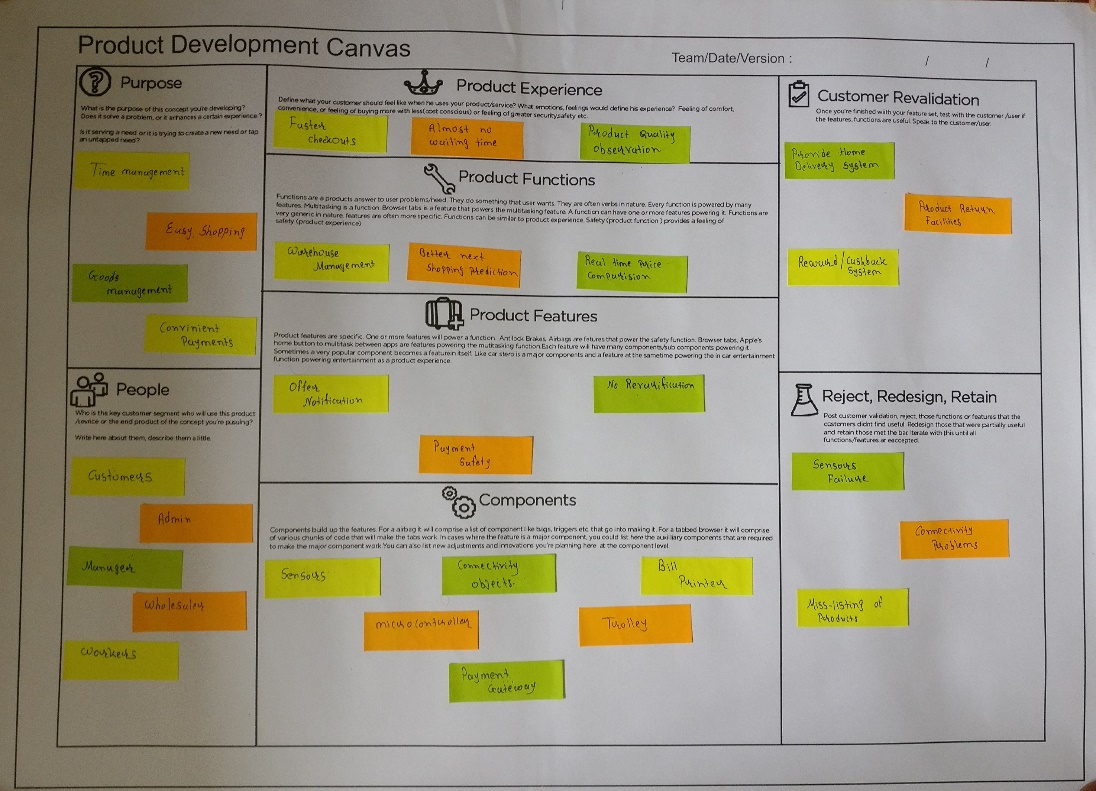
* <http://www.python.org/>
* <http://www.djangoproject.com/>
* <http://www.arduino.cc/>
* <http://www.W3C.com/>
* <http://www.qoura.com/>
* <http://www.youtube.com/>

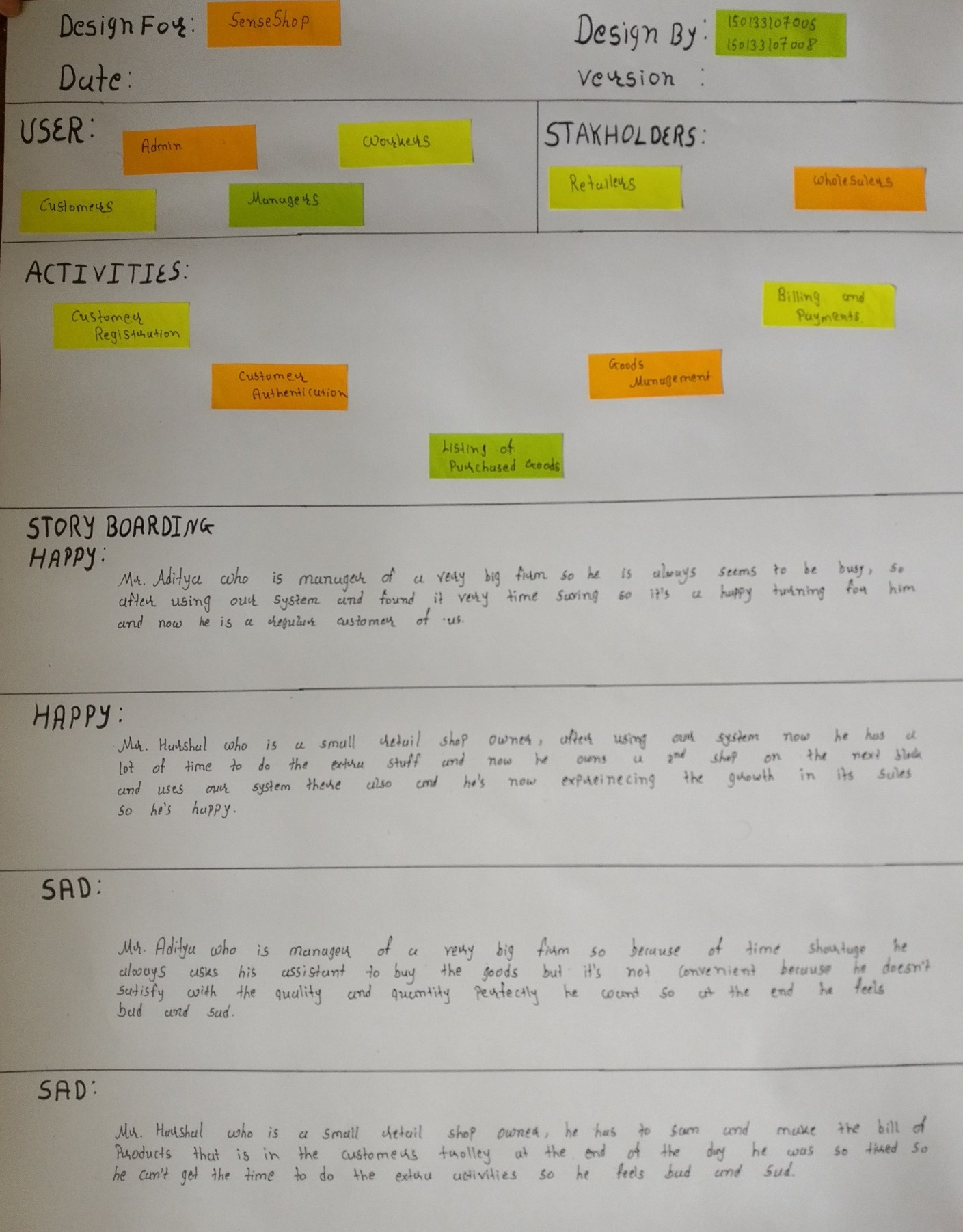
**CHAPTER:-9 APPENDIX**

**9.1 Canvas**:-

* AEIOU



* Product Development
* Empathy



* Ideation



**9.2 Periodic Progress Report(PPR):-**

**1st PPR**

|  |
| --- |
| **1.  What Progress you have made in the Project ?** |
| We are creating a system for shopping shop called as a senseshop. And we all team members are distributed our work into small modules for implementing view etc. |
| **2.  What challenge you have faced ?** |
| We faces some challenges like we need different type sensor for different implementation on our shop to automate this system. |
| **3.  What support you need ?** |
| We need a proper guidance of our faculties and new technologies to work with it. |
| **4.  Which literature you have referred ?** |
| Some books and websites are referred like Books: Iot with python Iot with Arduino Websites www.quora.com |

**2nd PPR**

|  |
| --- |
| **1.  What Progress you have made in the Project ?** |
| We are thinking to make our centralized system and also thinking for development mobile app which communicate with system directly. |
| **2.  What challenge you have faced ?** |
| How to communicate mobile with python web app. |
| **3.  What support you need ?** |
| We need proper guidance of our facility to some extra knowledge for hardware & sensor |
| **4.  Which literature you have referred ?** |
| Some books and websites reffed like Books Learn python the hard way Python essential reference Websites www.python-course.eu www.python.com |

**3rd PPR**

|  |
| --- |
| **1.  What Progress you have made in the Project ?** |
| We made a list of total number of different type of user and their work while interacting with the system. |
| **2.  What challenge you have faced ?** |
| We have to validate the user |
| **3.  What support you need ?** |
| We need proper database or Id verification of the user to validate them |
| **4.  Which literature you have referred ?** |
| Some books and websites reffed like Books Database system concept Relational database management system Websites www.oracle.com www.mysql.com |
|  |

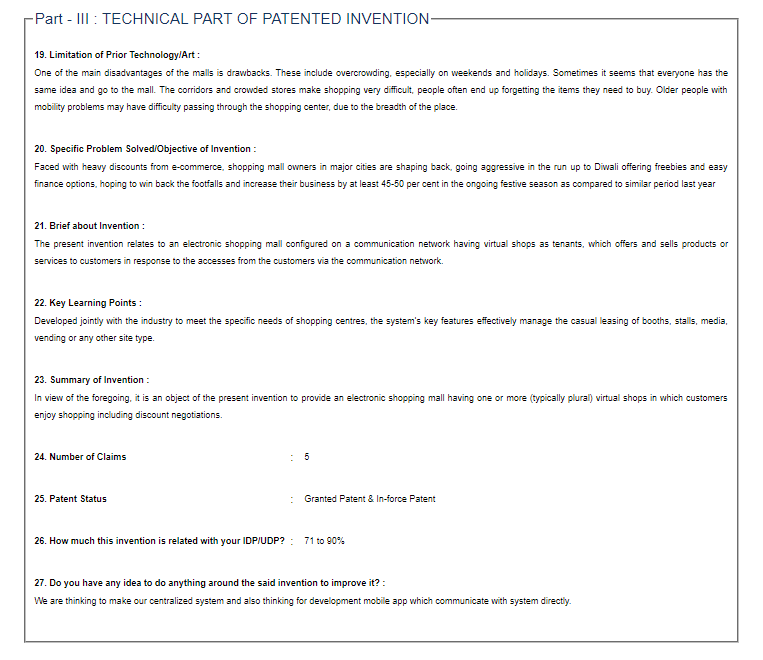
**4th PPR**

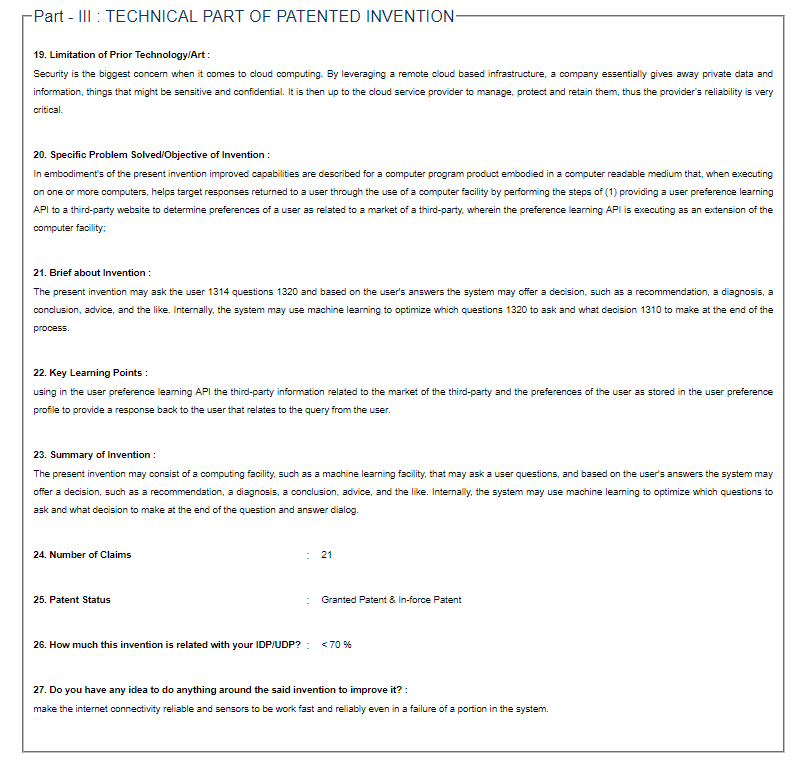
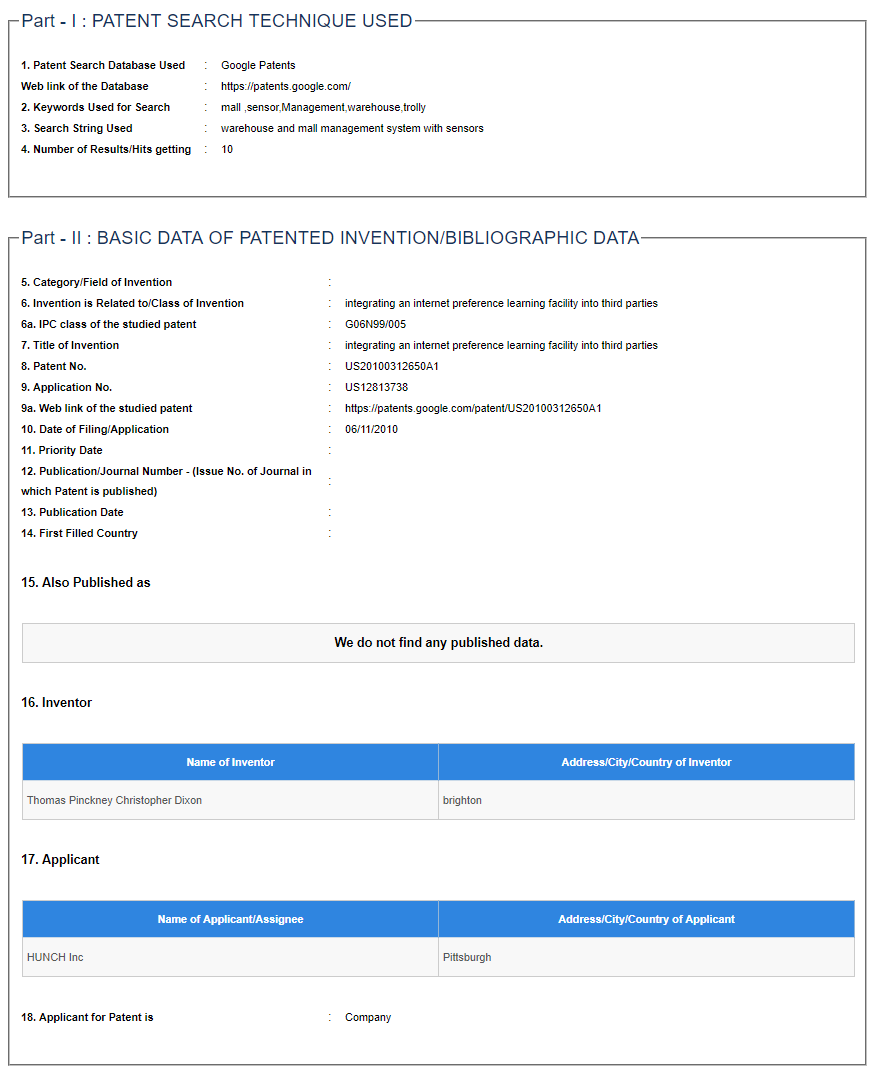
|  |
| --- |
| **1.  What Progress you have made in the Project ?** |
| We made a list of total number of different type of user and their work while interacting with the system. |
| **2.  What challenge you have faced ?** |
| We have to validate the user |
| **3.  What support you need ?** |
| We need proper database or Id verification of the user to validate them |
| **4.  Which literature you have referred ?** |
| Some books and websites reffed like Books Database system concept Relational database management system Websites www.oracle.com www.mysql.com |
|  |

**9.3 Patent Search and Analysis Report(PSAR):-**

**1st PSAR**

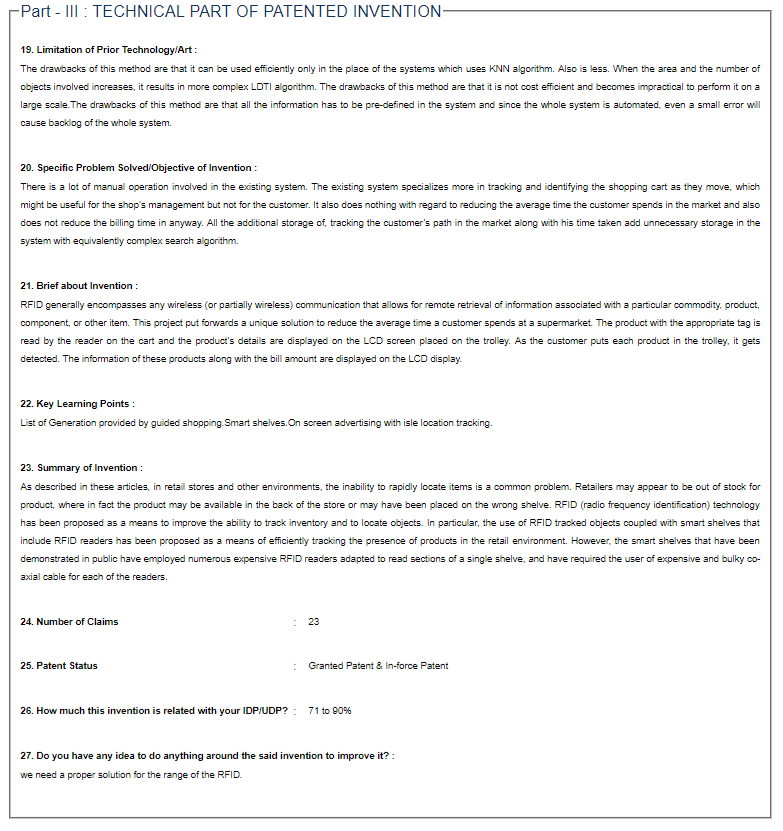
****

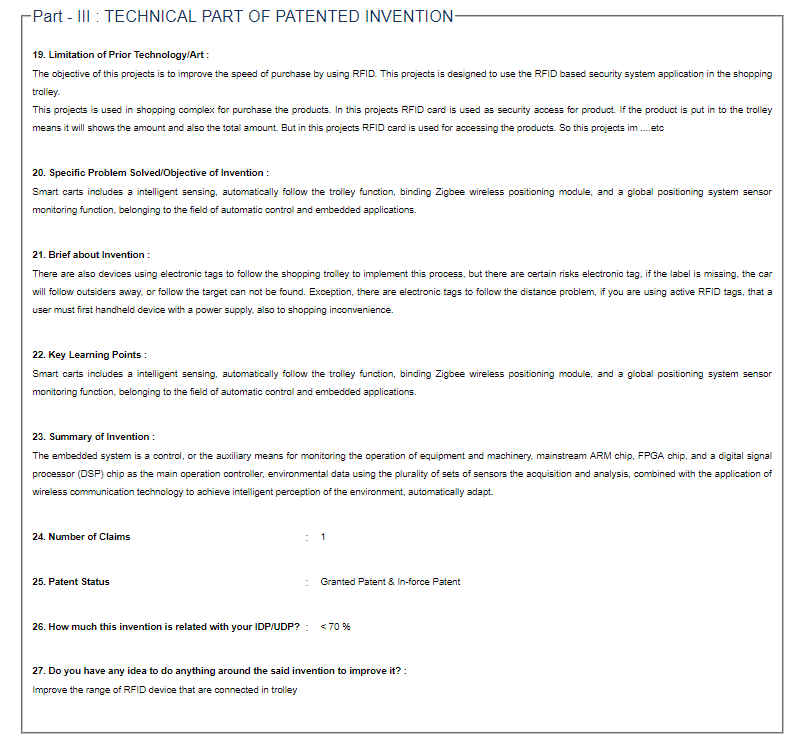
****

**2nd PSAR**

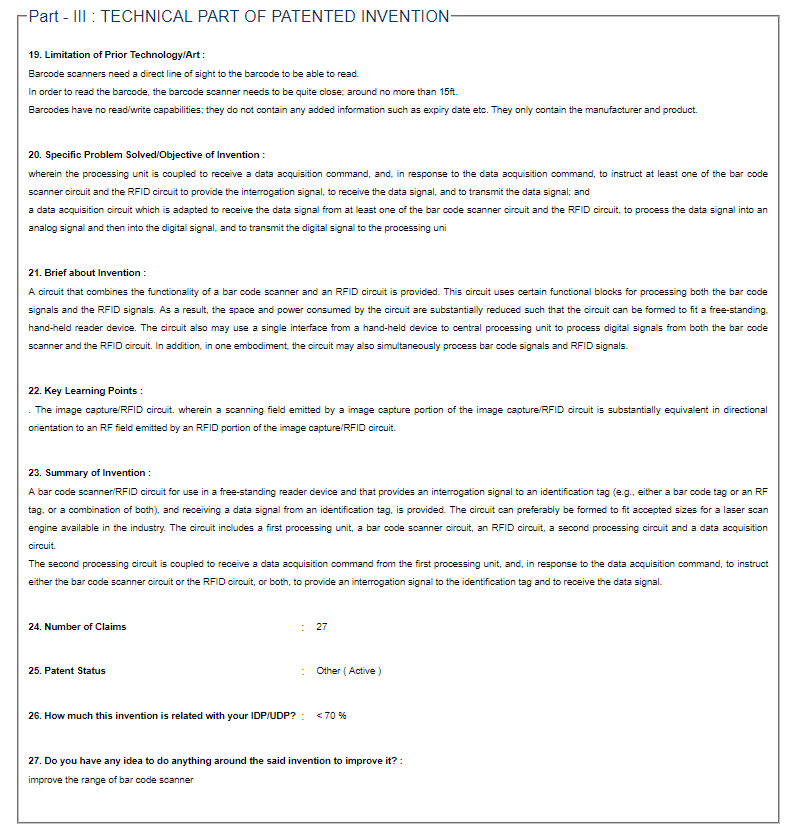
**3rd PSAR**

****

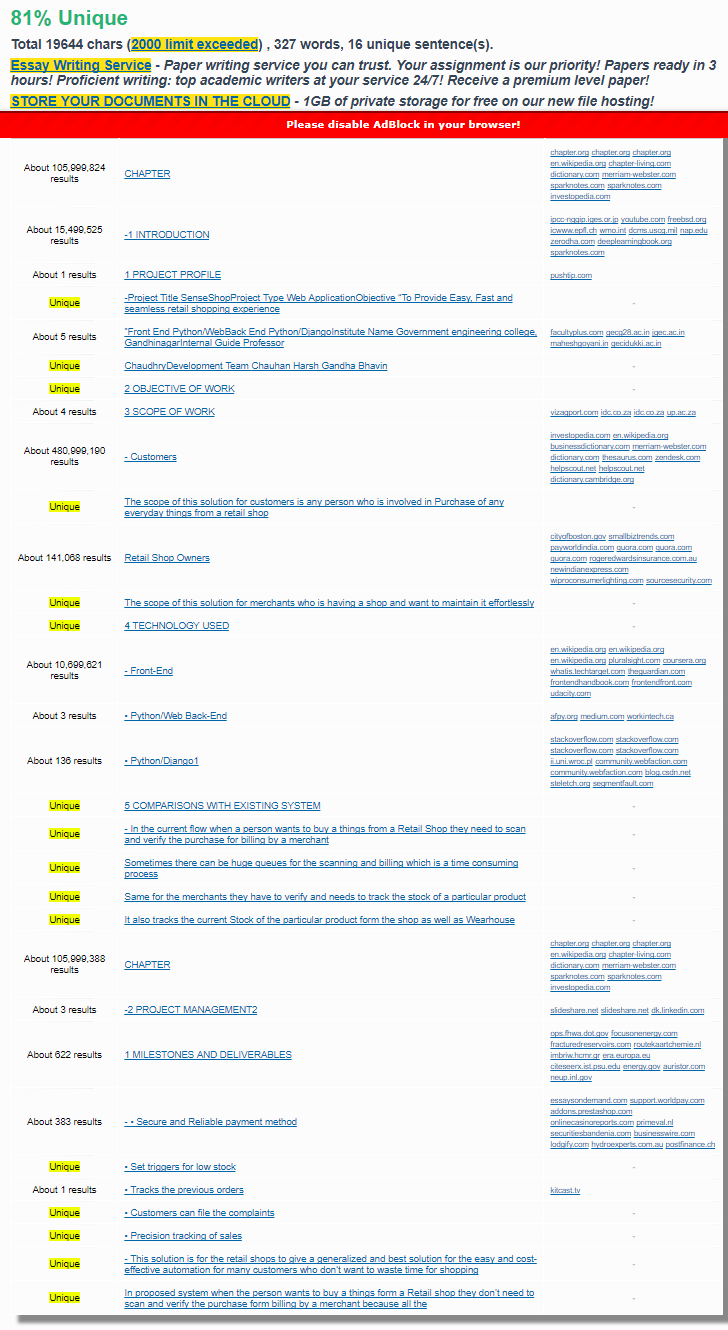
****

**4th PSAR**

**5th PSAR**

****

**Plagiarism Report:-**

****