**Project Name:**

**Analysis and design phase**

**Date: 20th Febrauray 2019**

**By: Sadina jarga magar**

**Batch: 21’A’**

**NCC Id: 00170190**

* **Analysis phase**

1. Functional and non-functional requirements of supermarket management system

Functional and non-functional requirements need to be carefully selected in order to ensure that they make sense in the context of the final outcome of the project and conveyed to all the team members working on it.

The definition of a functional requirement is:

Any requirement which specifies **what** the system should do.

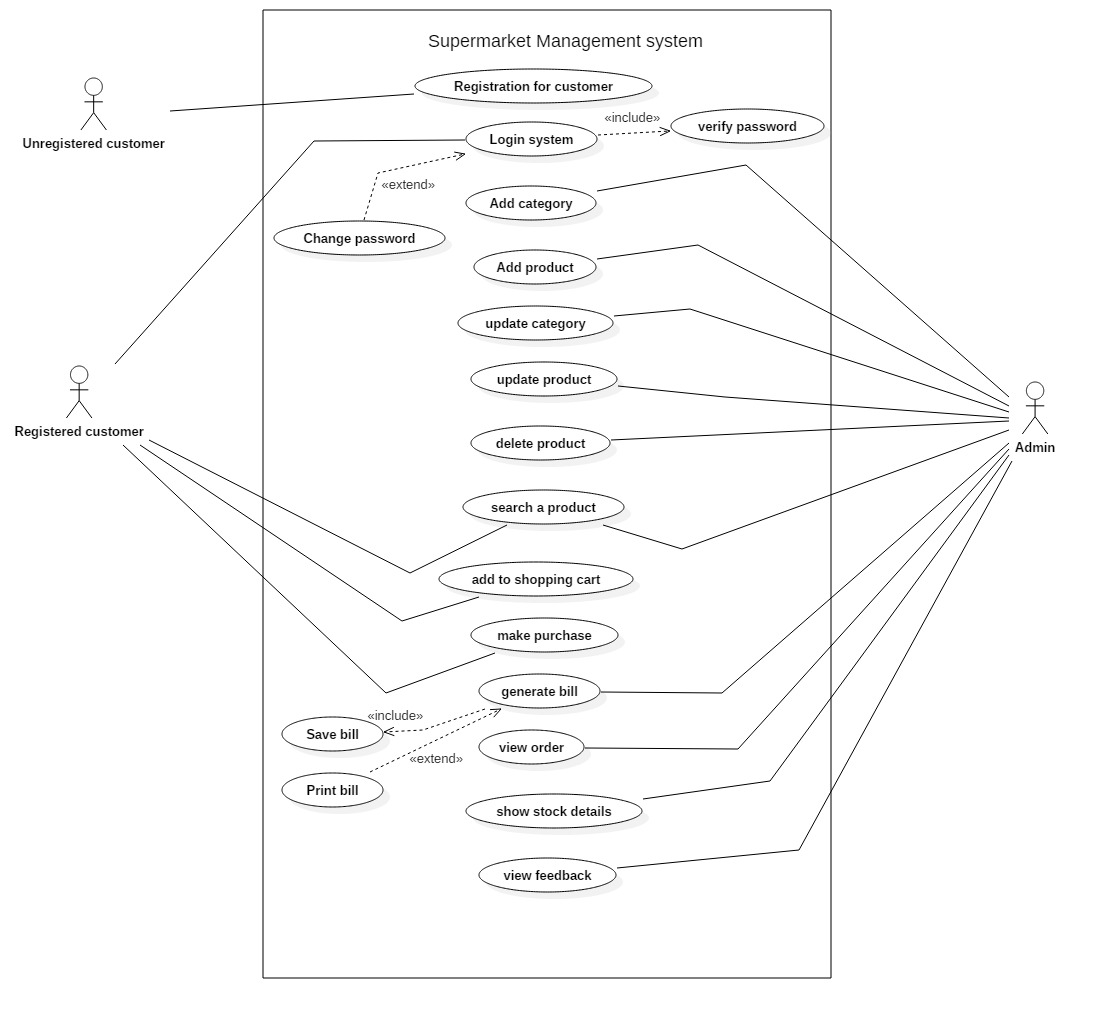
The definition of a non-functional requirement is:

Any requirement which specifies **how**the system performs a certain function.

|  |  |  |
| --- | --- | --- |
| Functional and Non-functional determination | Requirements | MoSCoW Prioritization |
| F(R1) | Registration for customer | M |
| F(R2) | Login system | M |
| F(R3) | Add category | M |
| F(R4) | Add product | M |
| F(R5) | Update category | M |
| F(R6) | Update product | M |
| F(R7) | Delete category | S |
| F(R8) | Delete product | S |
| F(R9) | View orders | M |
| F(R10) | View feedback | S |
| F(R11) | Order history | S |
| F(R12) | Search a product | M |
| F(R13) | Show stock details | M |
| F(R14) | Generate bill for customer | M |
| F(R15) | Add to shopping cart | M |
| F(R16) | Change password | S |
| NF(R17) | User-friendly interface | M |
| NF(R18) | Usability | S |
| NF(R19) | Maintainability | C |
| F(R20) | Make purchase by customer | M |
| NF(R21) | Website shall be secured from hacker | S |
| NF(R22) | Website shall be backed up | S |
| NF(R23) | Reliable | C |

1. Use-case diagram of supermarket management system

In the Unified Modeling Language (UML), a use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system. To build one, we'll use a set of specialized symbols and connectors.



Justification

* I have used this class diagram to describe a set of actions that some system can perform in collaboration with one or more external users of the system. To gather system requirements and actors. To specify the events of a system and their flows.

Advantage

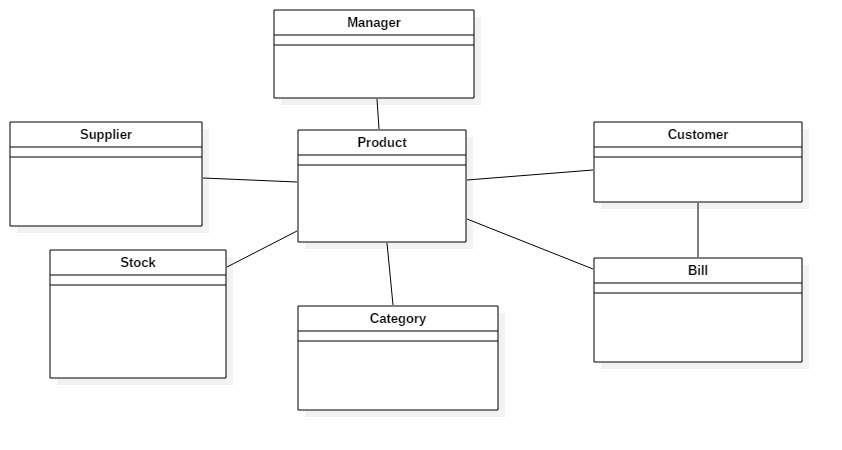
* It helps to collect the functional requirements of a system.
* It can serve as the basis for the estimating, scheduling and validating effort.
* It can also collect additional behavior that can improve system robustness.

Disadvantage

* They do not capture the non-functional requirements easily.
* Difficult to manage scenario.
* It has poor identification of structure and flow.

1. Initial class diagram of supermarket management system

* In the class diagram, classes are represented with boxes that contain three compartments:
* The top compartment contains the name of the class. It is printed in bold and centered, and the first letter is capitalized.
* The middle compartment contains the attributes of the class. They are left-aligned and the first letter is lowercase.
* The bottom compartment contains the operations the class can execute. They are also left-aligned and the first letter is lowercase



Justification

* I have made this initial class diagram to model the static view of an application. Also. Easy for forward and reverse engineering. This class diagram helps to describe the responsibilities of a system.

Advantage

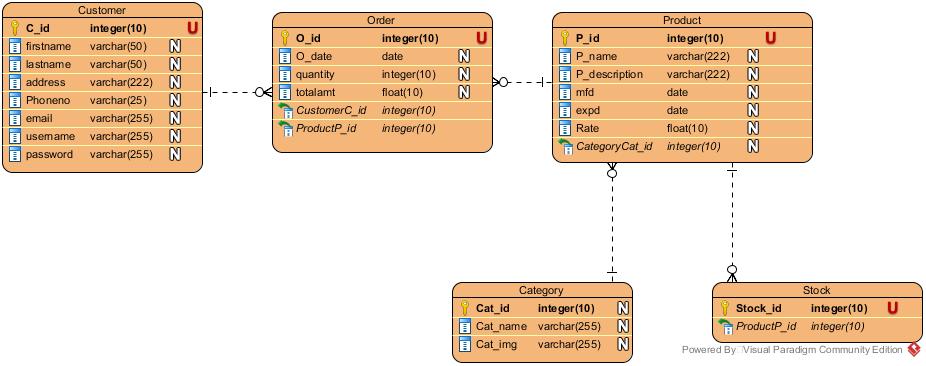
* Class diagram are simple and fast to read.
* It gives you a sense of orientation.
* They provide detailed insight into the structure of our systems.

Disadvantage

* They do not have dynamic model.
* It shows only collaboration among the elements of the static view.

1. ER-diagram of supermarket management system

The ER or (Entity Relational Model) is a high-level conceptual data model diagram. Entity-Relation model is based on the notion of real-world entities and the relationship between them.



Justification

* I have used this ER-diagram to visualize database design ideas; we have a chance to identify the mistakes and design flaws, and to make correction before executing the changes in database. By visualizing a database schema with an ERD, we have a full picture of the en tire database schema. We can easily locate entities, view their attributes and to identify the relationships they have with others.

Advantage

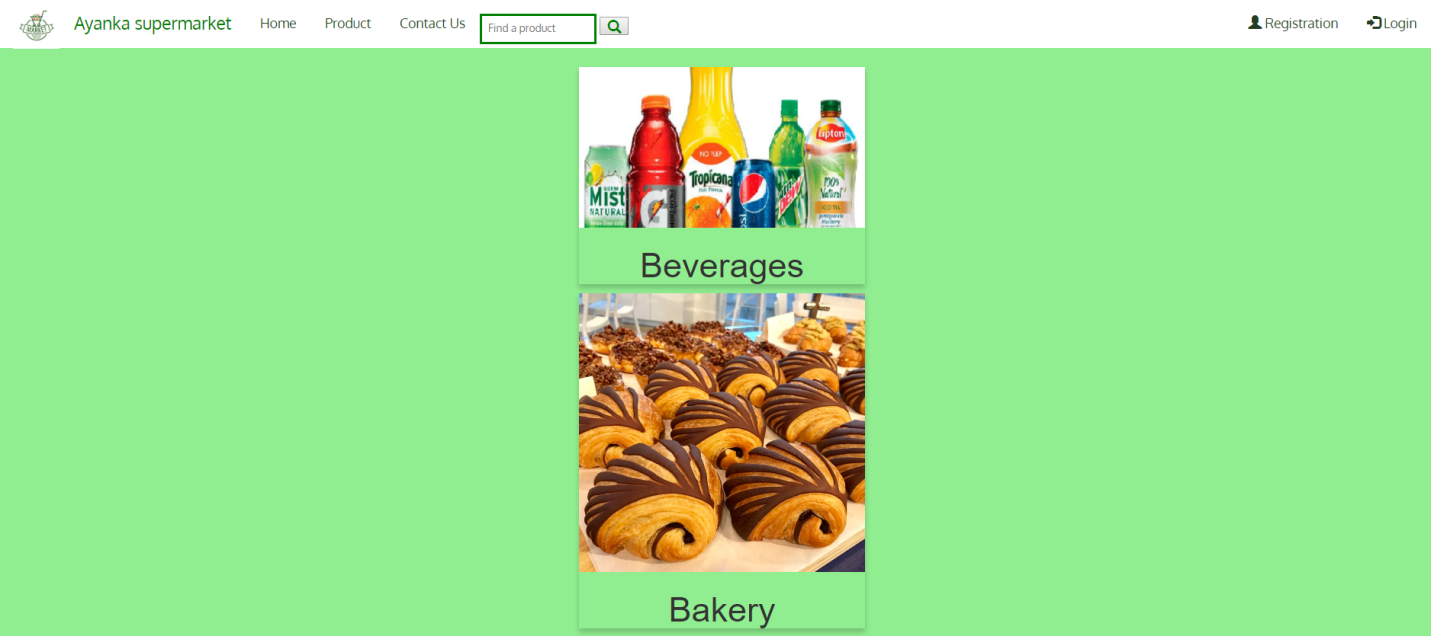
* It is very simple if we know relationship between entities and attributes.
* It is better visual representation.
* It is an effective communication tool for database designer.

Disadvantage

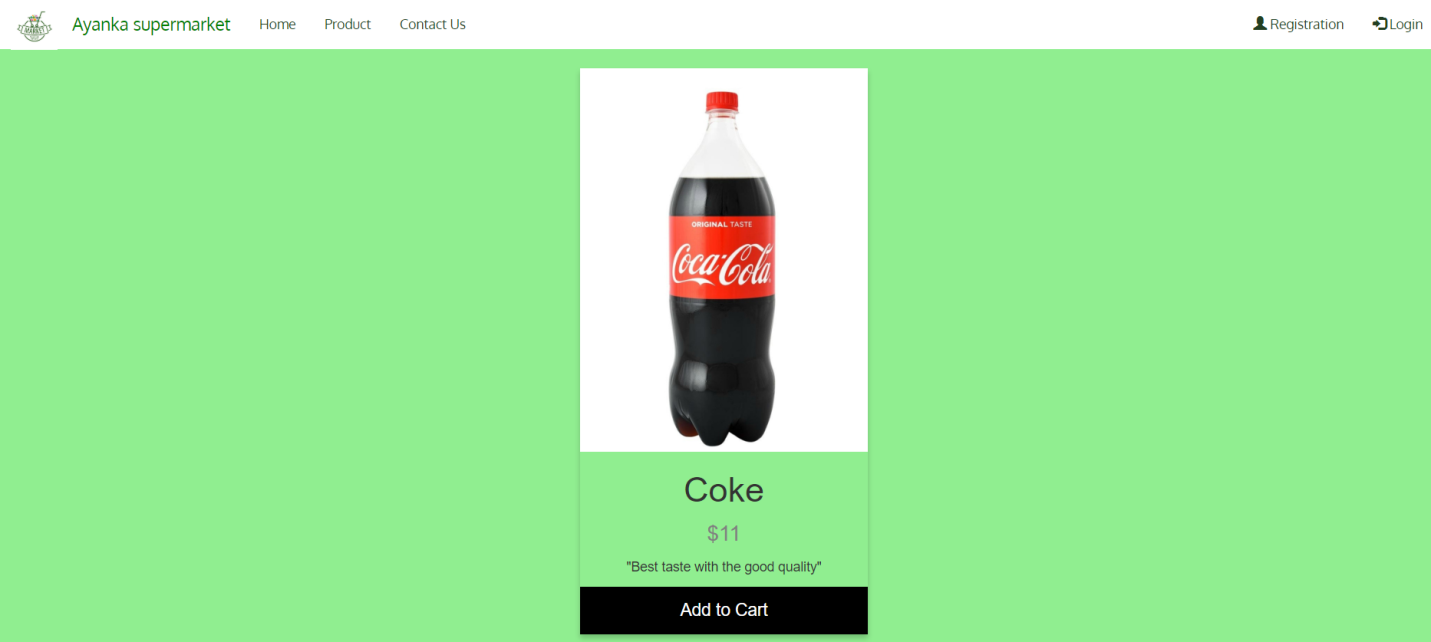
* It has limited constraints and specification.
* Information can be hidden in ER model.
* It is difficult to show data manipulation in ER model.

1. UI design of online supermarket.

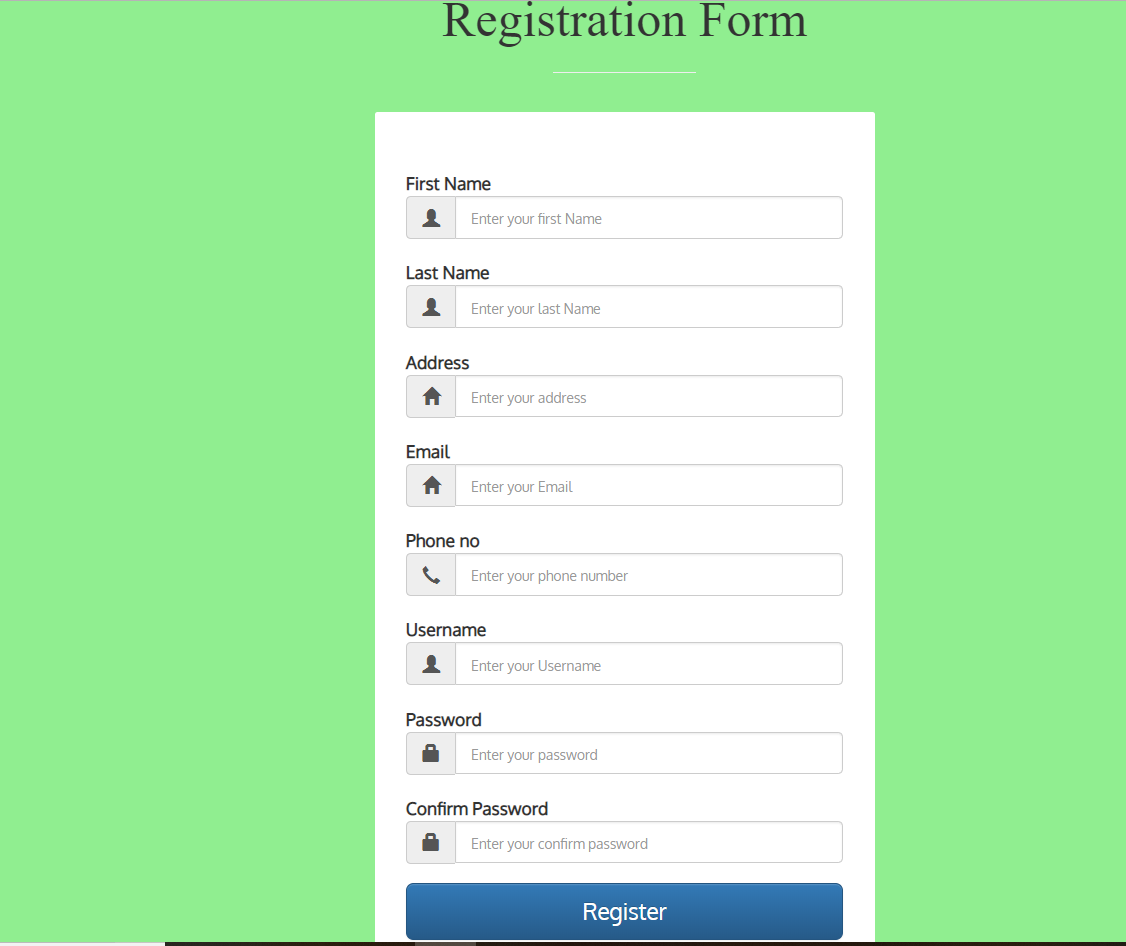
* Home page



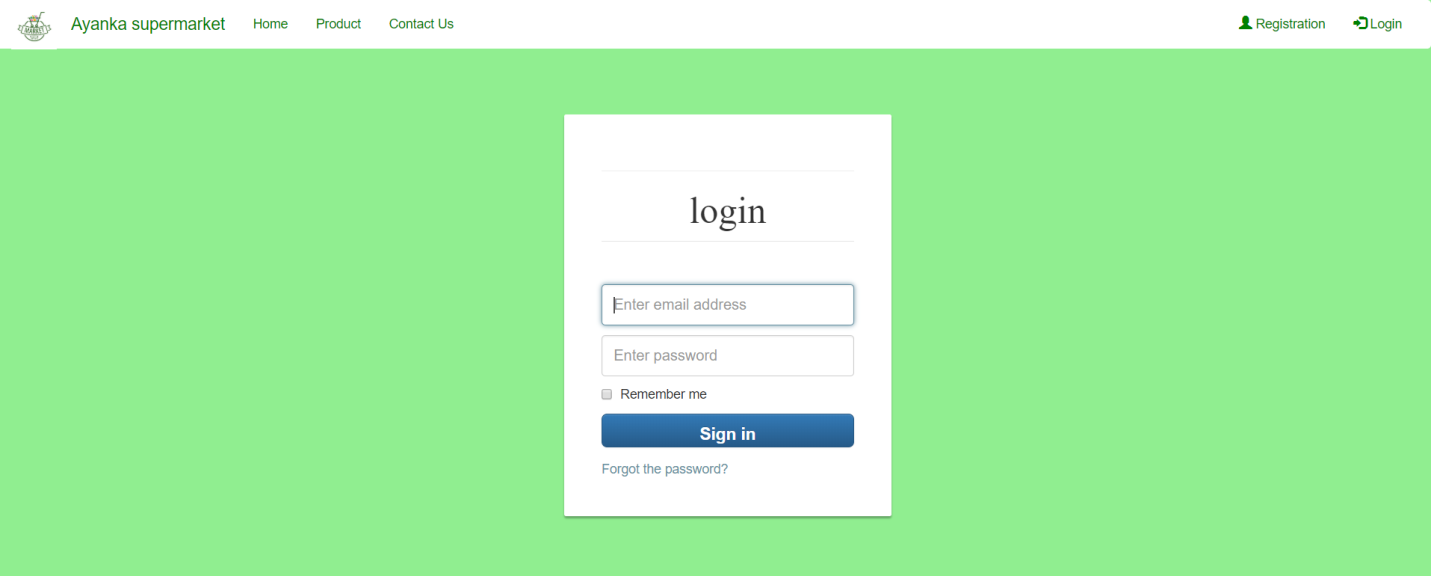
* Product page

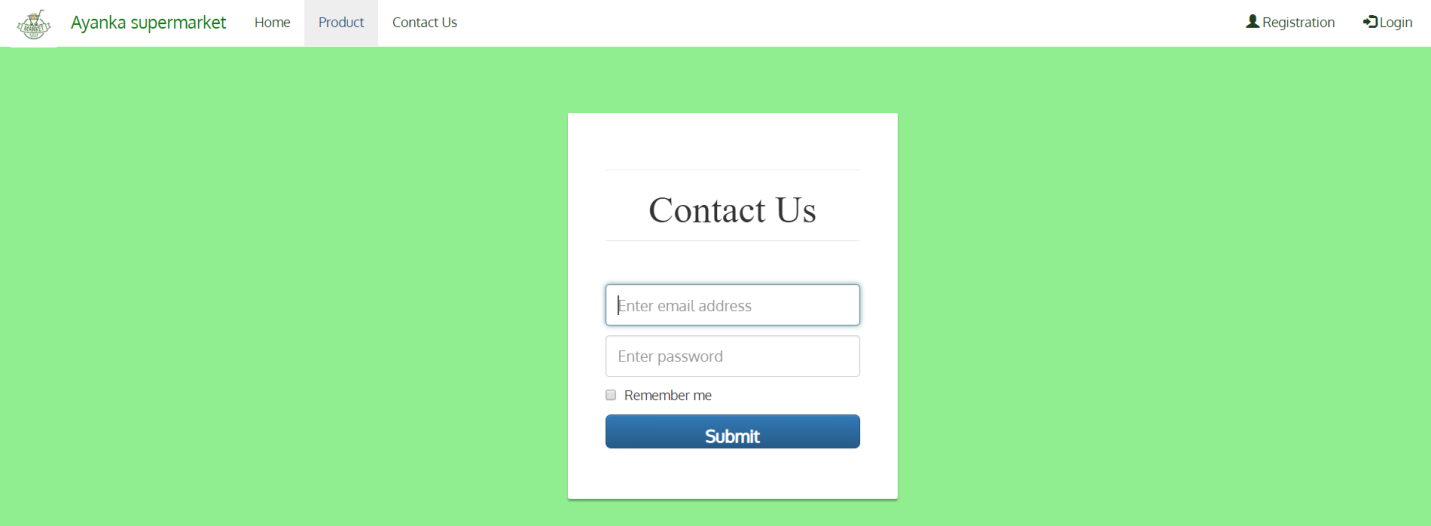


* Registration page



* Login page



* Contact page
* **Design phase**
* Class Diagram
* In the class diagram, classes are represented with boxes that contain three compartments:
* The top compartment contains the name of the class. It is printed in bold and centered, and the first letter is capitalized.
* The middle compartment contains the attributes of the class. They are left-aligned and the first letter is lowercase.
* The bottom compartment contains the operations the class can execute. They are also left-aligned and the first letter is lowercase

Natural language analysis (NLA) is the process of identifying nouns, verbs and adjective in a piece of descriptive text or scenario. The main objective is to identify the business models and to have a static model of the system which is represented as class diagram.

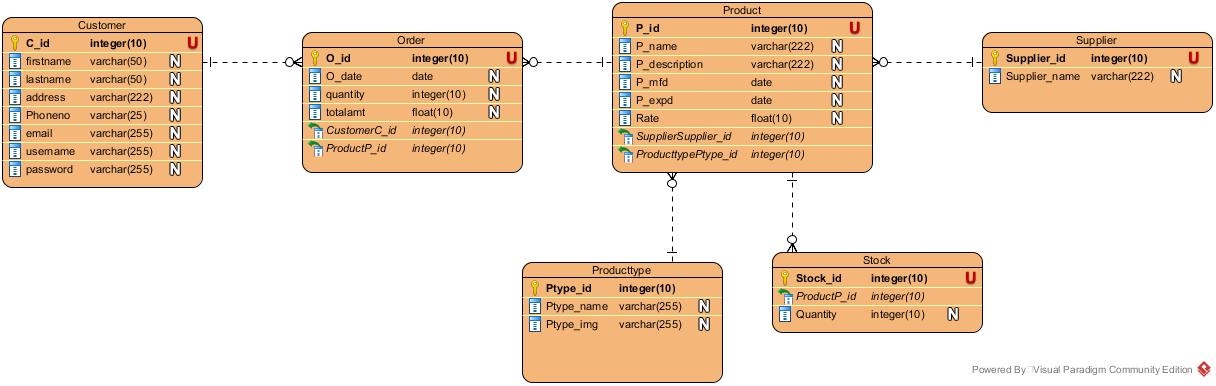
NLA proceeds filtration for nouns as candidate classes, verbs as potential methods and adjective as attributes.

1) Final class diagram of supermarket management system.

Entity-Relationship Diagram

3) ER- diagram of supermarket management system.

Entity relationship diagram displays the relationships of entity set stored in a database. In other words, we can say that ER diagrams help you to explain the logical structure of databases.



Justification

* I have used this ER-diagram to visualize database design ideas; we have a chance to identify the mistakes and design flaws, and to make correction before executing the changes in database. By visualizing a database schema with an ERD, we have a full picture of the en tire database schema. We can easily locate entities, view their attributes and to identify the relationships they have with others.

Advantage

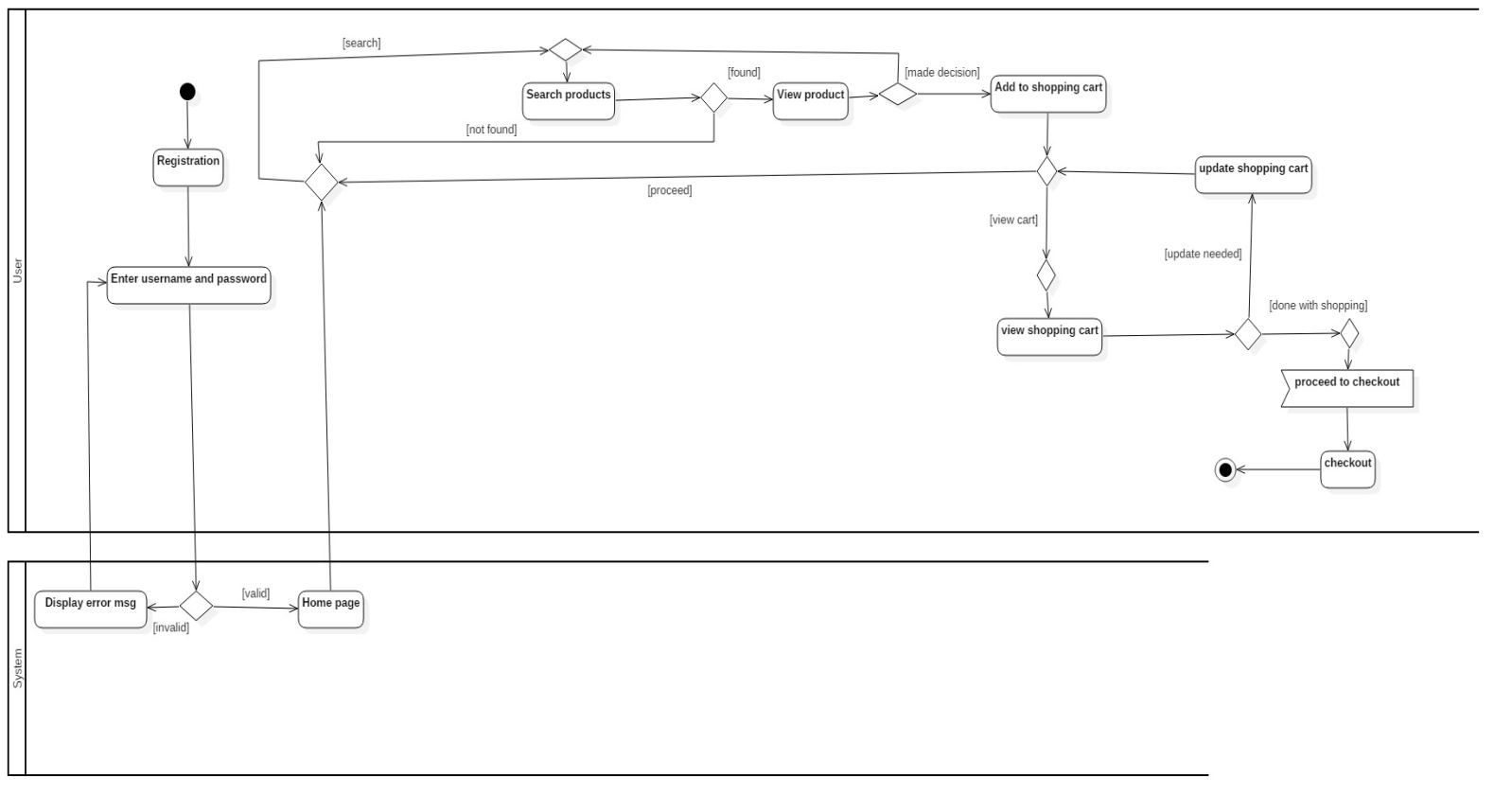
* It is very simple if we know relationship between entities and attributes.
* It is better visual representation.
* It is an effective communication tool for database designer.

Disadvantage

* It has limited constraints and specification.
* Information can be hidden in ER model.
* It is difficult to show data manipulation in ER model.

4) Activity diagram for online shopping in supermarket.

* Activity diagram is another important diagram in UML to describe dynamic aspects of the system
* Activity diagram is basically a flow chart to represent the flow from one activity to another activity
* The activity can be described as an operation of the system.
* The flow can be sequential, branched or concurrent. It deals with all type of flow control by using different elements like fork, join etc.



Justification

* I have used this activity diagram to draw the activity flow of a system and to describe the parallel, branched and concurrent flow of the system. To describe the sequence from one activity to another.

Advantage

* Since it is the most user-friendly diagram. So, generally regarded as an essential tool.
* It helps to display multiple conditions and actors within a work flow through the use of swimlanes.
* These diagrams are normally easily comprehensive for both analysts and stakeholder.

Disadvantage

* These diagrams can lead the over complex which might affect the user-friendly nature.
* These diagrams do not give the detail about how object behave or collaborate.

5) Sequence Diagram

