GARBAGE MANAGEMENT SYSTEM

Advisor's name: Mohammad Al Hajj

Course Name: Software Engineering

Course Code: CSCI317

Signature:

Instructor Notes: Date of approval:

JUNE 3, 2022

DONE BY ALI HAMED
1220024

TABLE OF CONTENTS

List of Tables

Garbage Management System

INTRODUCTION ABOUT THE SYSTEM:

Garbage Management System is a system that aims to solve the problems related to the accumulation of waste and the crises it can cause at various levels. This system was reached after several studies conducted on various economic, social, cultural and other fields.

The program that will do this process has been named Clean&Green.

The problem of waste accumulation is one of the biggest problems that any society may face. Therefore, in order to find an effective mechanism to solve this problem or even contribute to a large extent in solving it, **Clean&Green** application is the perfect project for that.

Clean&Green is an application that is downloaded to the phone by the users and it is a free application. The thing that provides users with the opportunity to benefit from it and be present with them wherever they are.

THE GENERAL OBJECTIVE OF THIS ARTICLE:

The main objective of this article is to explain the idea of the program and explain the smallest details about how it works and the things it provides.

THE GENERAL STRUCTURE OF THE PAPER:

This page is divided into several sections in order to organize the process of explaining this system and thus ease of understanding the parts of the program and its precise mechanism of action.

The sections will be displayed as follows:

1. The proposal.
2. Functional and non-Functional requirements.
3. Use case diagram.
4. ER diagram.
5. Process Modeling.

The Proposal

AIM OF THIS PROGRAM:

This program aims to raise awareness among people, mainly on solving the waste crisis in different ways.
What gives an additional advantage to this program is that it is a free and easy to use for the user, and it is also a program that achieves a financial return for the programmer by displaying some advertisements as an example.

PROBLEM OVERVIEW:

The waste crisis is deeply rooted in Lebanon, while treatment methods are of secondary importance. The year 2015 marked an important stage in the history of the environmental explosion. In that year, a large wave of popular protests erupted in the face of official failure. At that time, the government adopted an emergency plan to treat waste, by establishing temporary central dumps in several coastal areas. But this solution was not the best solution, and even the possibility of recurring the crisis or creating another crisis because of these temporary dumps is very likely to happen. (Change, n.d.)

WASTE PROBLEMS:

If we want to present the problems caused by waste, it will be in the following form:

- Accumulation of waste due to the inability to pay for its disposal in an efficient and continuous manner.
- Traffic jams that cause traffic paralysis as a result of the accumulation of waste on the roads.
- Waste is a major cause and an incubator environment for harmful insects.
- Unpleasant odors.
- Closure of sewage sewers, which leads to the formation of polluted water swamps on the roads.
- Waste is a major cause of the spread of diseases, which may sometimes lead to lifelong chronic diseases and in some cases to death.

(EARTHDAY.ORG, n.d.)

THE SOLUTIONS:

However, there are many things we can do in order to fix these problems:

- Frequent use and reduce consumption, such as bringing old empty bags to the grocery store to put what we need instead of consuming new bags.

- Recycling and treatment of organic waste.
- Sanitary landfill that meets certain standards, while keeping these landfills under control so that no leakage and thus contamination of the soil occurs.
- Burning waste to generate electricity from the heat generated by it.
- Selling waste to recycling companies abroad.

(ACTION FOR RENEWABLES, n.d.)

THE PURPOSE OF THE PROGRAM:

Why is this program important and how do this program contribute to addressing these problems?

The **Clean&Green** is in addition to what I mentioned earlier, is easy to use and free, but it depends mainly on getting people to contribute directly to solving this crisis.

This program depends on several things:

- Increasing people's awareness by explaining to them about the potential risks caused by the accumulation of waste.
- Make people do the storing process at home by explaining to them the amount of financial return that can be collected by doing this process. They will make money from nothing.
- Guiding people on how to do the sorting process correctly in order to benefit from everything and achieve a greater financial income.
- Clarify the places and ways of selling these materials after they are sorted.

Moreover, this program is not only concerned with recyclable materials, but on the contrary, it also explains how to deal with organic materials and benefit from them.

HOW TO VALIDATE THE SUCCESS:

In order to validate the success of this project, consider the following:

- Show different ways of recycling process in hopes of widening our user base to the furthest extend possible.
- Continuously update the locations and prices of the stored items.
- Put some videos that will help in the process of explaining the complex things.
- Establishing a store within this system, where the user can sell materials from his home, and companies will take care of the transportation process.
- Establishing an alert system regarding the process of creating fertilizers during the process of recycling organic materials aims to alert the user when the process is finished.
- Allow companies to post some advertisements for their services that goes with ours.

SYSTEM USERS:

- Anyone
- Small companies that want to achieve greater profit.
- Large companies and municipalities.

THE PLAN:

- General analysis and requirements gathering.
- Writing information that aims to increase people's awareness and scientific sources for this information and adding other scientific sources.
- Creating the Product Design.
- Creating the database system.
- Generating and connect the alarm system with database.
- Establishment of the store and the mechanism of communication.
- Create a space for people to communicate with each other and to present what they do and find useful.
- link the program with database.
- Testing and solving the problems.
- Deployment of the Product and Maintenance and Operations.

TIME PLANE:

The suggested timetable of the plan is as follows. (One year in total)

Table 1 ***Time Plane***

Starting date	1 May 2022
Ending date	1 May 2023

Activity	Start date	End date	Budget
1	1 May 2022	1 June 2022	450-600
2	1 June 2022	1 Jul 2022	100$
3	2 Jul 2022	1 Aug 2022	650-850
4	1 Aug 2022	1 Nov 2022	200-250
5	1 Nov 2022	1 Des 2022	300-400
6	1 Des 2022	15 Des 2022	250-400
7	16 Des 2022	1 Jan 2023	450-600
8	1 Jan 2023	1 Feb 2023	100$
9	1 Feb 2023	1 Mar 2023	650-850
10	1 Apr 2023	1 May 2023	200-250

RESOURCES:

The software required for this system is the main website and its corresponding database. No associated hardware is needed.

All in all, Clean&Green is the ideal program to solve this crisis, which is constantly recurring in Lebanon, and with the increase in economic problems and increase in the population, this program is the correct way to solve the crisis.

Functional and Non-Functional Requirements

INTRODUCTION ABOUT THE REQUIREMENTS:

One of the basic things that you must pay attention to in order to succeed in any system are the requirements.

The requirements are separated into two parts:

→Functional Requirements.

→Non-Functional Requirements.

- In software engineering and systems engineering, a functional requirement defines a function of a system or its component, where a function is described as a specification of behavior between inputs and outputs. (Wikipedia, Wikipedia, n.d.) However, a non-functional requirement (NFR) is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. They are contrasted with functional requirements that define specific behavior or functions. The plan for implementing functional requirements is detailed in the system design. The plan for implementing non-functional requirements is detailed in the system architecture, because they are usually architecturally significant requirements. (Wikipedia, Wikipedia, n.d.)

After a lengthy study was conducted with a group of experts and after a group of random opinions were taken from the people in order to achieve success for this project.

And after doing some searches on the internet. A set of basic points have been established that the system must follow. These points are under the requirements label.

In order to achieve the best success of this project, these requirements will be sorted into sections:

1- The system.
2- The admin.
3- The users.
4- The municipalities.

REQUIREMENTS GATHERING PROCESS DETAILS:

INTERVIEW QUESTIONS:

Programmer Questions:
The questions that were asked to the programmer, Muhammad Zalzalah:

Question 1: How should the program be in terms of internal form?

The program should consist of a set of independent sections that are subsequently linked in a logical manner.

Question 2: What are the sections that the program should include?

The program should include a group of sections. Each section should contain all the basic information related to it. In the case of my program, there should be a section for store, a section for conversation and another for information, and there should also be a section for the user's personal information and a section for his publications. In addition, do not forget to add a section to watch the information that is published by users on the application (like videos posts, written posts…. etc.). moreover, to what was previously mentioned, it must also take into account any additional details matter such as the admin and municipalities.

Question 3: Should the program include a set of laws and regulations that users must follow or not?

Certainly, there must be laws in order to ensure the continuity of this application and to prevent any problems that would cause inconvenience to the user and thus not to use the program by them.

People Questions:
A set of questions that were asked to some people who were randomly selected from street. After presenting and clarifying the idea of the program, questions were asked in

terms of what additions the program should include, which in their opinion are important for the success of this project. The answers were as follows:

Opinion 1: The program must clearly explain how the recycling process is carried out because the majority of people do not know how.

Opinion 2: Several methods of recycling should be presented so that people do not find it difficult to implement them.

Opinion 3: the percentage of saving and profit that the user can achieve if he uses the application must be displayed to encourage people to use it and to create a logical reason for that.

INTERNET SEARCHES I CONDUCTED:

13 things you need to consider as a part of your mobile app marketing strategy… (Mullan, n.d.)

Design (Wikipedia, Wikipedia, n.d.)

5 Important things to consider while developing a Mobile App (Abrition, n.d.)

FUNCTIONAL REQUIREMENTS:

THE SYSTEM:

1. The system should verify if the account has already been registered or not.
2. The system should provide a menu to display the program's sections.
3. The system should have the ability to access the internet.
4. The system should have a search system connected to the internet.
5. The system should have an internal search system that does not need the internet.
6. The system should provide an internal store that its users can benefit from.
7. The system should allow sales using cash.
8. The system should allow sales using credit card.
9. The system should support video playback.
10. The system should provide the ability to add publications.
11. The system should provide the ability to delete publications.
12. The system should allow its users to view some of each other's personal information.
13. The system should have the ability to decide which materials have the ability be recycled when the system users ask about it.
14. The system should have alert system.
15. The system should have a section to display information about waste problems.

16. The system should have a section to display information about how to solve the waste problems.
17. The system should have a section to display information about how to benefit from the waste to make money.
18. The system should have a section to display information about how to benefit from the waste to save money.
19. The system should implement advertisement, in order to be profitable.
20. The system should have a rating option for posts.
21. The system should provide a chat box.
22. The system should save the conversations to ensure the rights of users.
23. The system should prevent users from modifying information that is from the origin of the system and present in it when it was created.
24. The system should contain a set of rules that its users must follow.
25. The system should save the data of its users in an encrypted manner.
26. The system should prevent the user from modifying the system information.
27. The system should prevent the municipalities from modifying the system information.

THE ADMIN:

1. The system should allow the admin to sign in.
2. The system should allow the admin to sign up.
3. The system should allow the admin to see the menu of the program's sections.
4. The system should provide the admin to do search through it.
5. The system should provide the admin to have access to the store.
6. The system should allow the admin to make the sale through the store by using cash.
7. The system should allow the admin to make the sale through the store by using credit card.
8. The system should allow the admin to make a purchase through the store by using cash.
9. The system should allow the admin to make a purchase through the store by using credit card.
10. The system should allow the admin to see the videos through it.
11. The system should allow the admin to delete any video in it.
12. The system should allow the admin to add any video to it.
13. The system should allow the admin to add any written post.
14. The system should allow the admin to delete any written post.
15. The system should allow the ad admin to see some of the other's personal information.
16. The system should allow the admin to benefit from the alert system.
17. The system should allow the admin to add some options to the alert system.
18. The system should allow the admin to delete some options from the alert system.

19. The system should allow the admin to do changes to the alert system.
20. The system should allow the admin to see the information related to the waste problems.
21. the system should allow admin to add any information to the information section related to waste problems.
22. the system should allow admin to delete any information to the information section related to waste problems.
23. the system should allow admin to change any information to the information section related to waste problems.
24. The system should allow the admin to see the information related to the solutions of waste problems.
25. the system should allow admin to add any information in the information section related to the solutions of waste problems.
26. the system should allow admin to delete any information in the information section related to the solutions of waste problems.
27. the system should allow admin to change any information in the information section related to the solutions of waste problems.
28. The system should allow the admin to see the information in the system related to how waste is diverted to earn money.
29. The system should allow the admin to add information to the system related to the section of how waste is diverted to earn money.
30. The system should allow the admin to delete information to the system related to the section of how waste is diverted to earn money.
31. The system should allow the admin to change information to the system related to the section of how waste is diverted to earn money.
32. The system should allow the admin to see the information in the system related to how to benefit from the waste to save money.
33. The system should allow the admin to add information to the system related to the section of how to benefit from the waste to save money.
34. The system should allow the admin to delete information to the system related to the section of how to benefit from the waste to save money.
35. The system should allow the admin to change information to the system related to the section of how to benefit from the waste to save money.
36. The system should allow the admin to add new ads.
37. The system should allow the admin to delete ads.
38. The system should allow the admin to see the average rate for each post.
39. The system should allow the admin to add rate to any post.
40. The system should allow the admin to create a chat with users in general.
41. The system should allow the admin to add any new rule to system law.
42. The system should allow the admin to delete any rule in the system law.
43. The system should allow the admin to change any rule in the system law.
44. The system should allow the admin to modifying the system information.

THE USERS:

1. The system should allow the users to sign in.
2. The system should allow the users to sign up.
3. The system should allow the users to see the menu of the program's sections.
4. The system should provide the users to do search through it.
5. The system should provide the users to have access to the store.
6. The system should allow the users to make the sale through the store by using cash.
7. The system should allow the users to make the sale through the store by using credit card.
8. The system should allow the users to make a purchase through the store by using cash.
9. The system should allow the users to make a purchase through the store by using credit card.
10. The system should allow the users to see the videos through it.
11. The system should allow the users to delete their video in it.
12. The system should allow the users to add any video to it.
13. The system should allow the users to add any written post.
14. The system should allow the users to delete any of their written post.
15. The system should allow the ad users to see some of the other's personal information.
16. The system should allow the users to benefit from the alert system.
17. The system should allow the users to see the information related to the waste problems.
18. The system should allow the users to see the information related to the solutions of waste problems.
19. The system should allow the users to see the information in the system related to how waste is diverted to earn money.
20. The system should allow the users to see the information in the system related to how to benefit from the waste to save money.
21. The system should allow the users to see the average rate for each post.
22. The system should allow the users to add rate to any post.
23. The system should allow the users to create a chat with the other users in general.

THE MUNICIPALITIES:

1. The system should allow the municipalities to sign in.
2. The system should allow the municipalities to sign up.
3. The system should allow the municipalities to see the menu of the program's sections.
4. The system should provide the municipalities to do search through it.
5. The system should provide the municipalities to have access to the store.
6. The system should allow the municipalities to make the sale through the store by using cash.

7. The system should allow the municipalities to make the sale through the store by using credit card.
8. The system should allow the municipalities to make a purchase through the store by using cash.
9. The system should allow the municipalities to make a purchase through the store by using credit card.
10. The system should allow the municipalities to see the videos through it.
11. The system should allow the municipalities to delete their video in it.
12. The system should allow the municipalities to add any video to it.
13. The system should allow the municipalities to add any written post.
14. The system should allow the municipalities to delete any of their written post.
15. The system should allow the ad municipalities to see some of the other's personal information.
16. The system should allow the municipalities to benefit from the alert system.
17. The system should allow the municipalities to see the information related to the waste problems.
18. The system should allow the municipalities to see the information related to the solutions of waste problems.
19. The system should allow the municipalities to see the information in the system related to how waste is diverted to earn money.
20. The system should allow the municipalities to see the information in the system related to how to benefit from the waste to save money.
21. The system should allow the municipalities to see the average rate for each post.
22. The system should allow the municipalities to add rate to any post.
23. The system should allow the municipalities to create a chat with the other users in general.
24. The system should allow the municipalities to add any waste dumping sites.
25. The system should allow the municipalities to modify any waste dumping sites.
26. The system should allow the municipalities to remove any waste dumping sites.

NON-FUNCTIONAL REQUIREMENTS:

THE SYSTEM:

1. The system should operate with internet access.
2. The system should operate without internet access.
3. The system should have the ability to send verification message to the user when he edits his personal information.
4. The system should have the ability to send verification message to the admin when he edits his personal information.
5. The system should have the ability to send verification message to the municipality when it edits its information.

6. The system should have the ability to send notification message to the user every time he signs in.
7. The system should have the ability to send notification message to the admin every time he signs in.
8. The system should have the ability to send notification message to the municipality every time it signs in.
9. The system should ask the user to write his password every time he wants to edit any personal information.
10. The system should ask the admin to write his password every time he wants to edit any personal information.
11. The system should ask the municipality to write its password every time it wants to edit any information related to it.
12. The system should have the ability to display its content in several languages.
13. The system should provide the optimal answer for the questions.
14. Advertisements should be unintrusive for the user.
15. The system should allow it users to delete conversations within no more than two minutes.
16. The system should update the store every day.
17. The system should be updated every three months.
18. The system should provide a simple graphical user interface so that the user can easily deal with it.
19. The system should make sure that the sections follow each other in a logical order.
20. The system should be able to send notifications to the user for any news.
21. The system should be able to send notifications to the municipality for any news.
22. The system should have the ability to distinguish any defect made by the user against these laws.
23. The system should follow the steps related to violating laws in a systematic manner.

GANTT CHART:

WHAT IS GANTT CHART:

A Gantt chart is a type of bar chart that illustrates a project schedule, named after its popularizer, Henry Gantt, who designed such a chart around the years 1910–1915. Modern Gantt charts also show the dependency relationships between activities and the current schedule status. (Wikipedia, Wikipedia, n.d.)

THE TASKS:

(A) General analysis and requirements gathering.
(B) Writing information that aims to increase people's awareness and scientific sources for this information and adding other scientific sources.
(C) Creating the Product Design.

(D) Creating the database system.
(E) Generating and connect the alarm system with database.
(F) Establishment of the store and the mechanism of communication.
(G) Create a space for people to communicate with each other and to present what they do and find useful.
(H) link the program with database.
(I) Testing and solving the problems.
(J) Deployment of the Product and Maintenance and Operations.

The tasks will be displayed in the table based on the letters present before each of the tasks presented above.

Table 2 ***Gantt Chart***

	Gantt Chart											
	Time in month											
The task	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12
A												
B												
C												
D												
E												
F												
G												
H												
I												
J												

This project is based on a set of basic points that must be followed to ensure the creation of a program that achieves the desired goal and the needs of users in general.

Use Case Diagrams

INTRODUCTION ABOUT THE USE CASE:

WHAT IS THE USE CASE DIAGRAM?

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses. The actors are often shown as stick figures. (Wikipedia, Use Case, n.d.)

IMPORTANT POINT:

The main thing that must be constantly present during the use of the program, which is also a prerequisite for using the program is the login process.

THE DESCRIPTION OF THE USE CASE:

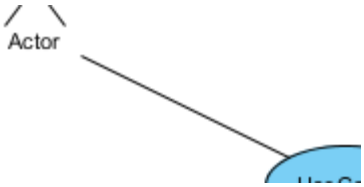
Use case:

A use case represents a user goal that can be achieved by accessing the system or software application. In Visual Paradigm, you can make use of the sub-diagram feature to describe the interaction between user and system within a use case by creating a sub-sequence diagram under a use case. You can also describe the use case scenario using the Flow of Events editor.

Actor:

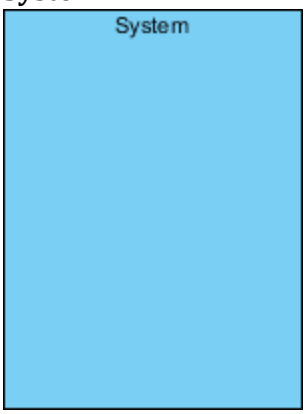
Actor and use case can be associated to indicate that the actor participates in that use case. Therefore, an association correspond to a sequence of actions between the actor and use case in achieving the use case.

Association:



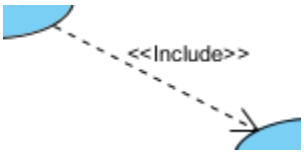
Actor and use case can be associated to indicate that the actor participates in that use case. Therefore, an association correspond to a sequence of actions between the actor and use case in achieving the use case.

System:



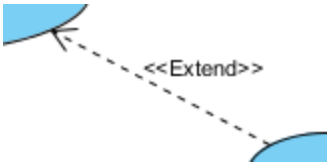
The scope of a system can be represented by a system (shape), or sometimes known as a system boundary. The use cases of the system are placed inside the system shape, while the actor who interact with the system are put outside the system. The use cases in the system make up the total requirements of the system.

Include:



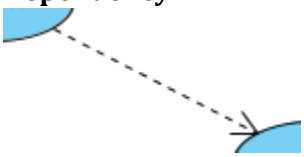
An include relationship specifies how the behavior for the inclusion use case is inserted into the behavior defined for the base use case.

Extend:



An extend relationship specifies how the behavior of the extension use case can be inserted into the behavior defined for the base use case.

Dependency:



A dependency relationship represents that a model element relies on another model element for specification and/or implementation.

Generalization:

A generalization relationship is used to represent inheritance relationship between model elements of same type. The more specific model element shares the same specification with. the more general the model element but carries more details in extra.

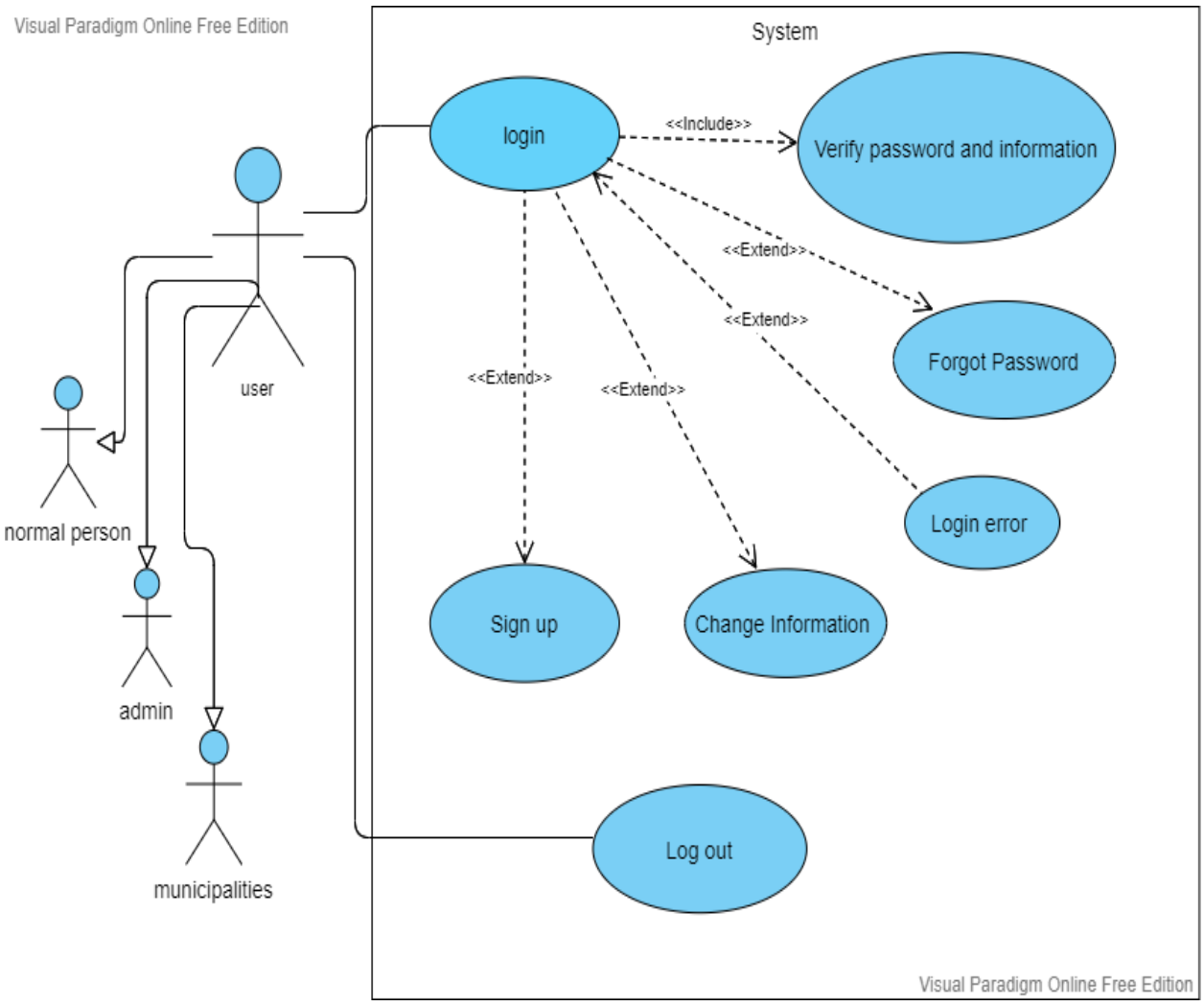
To see more details and forms of the use case, you can go to the following website: (Paraim, n.d.)

INFORMATION PRESENTATION:

THE COMMON THINGS FOR USERS:

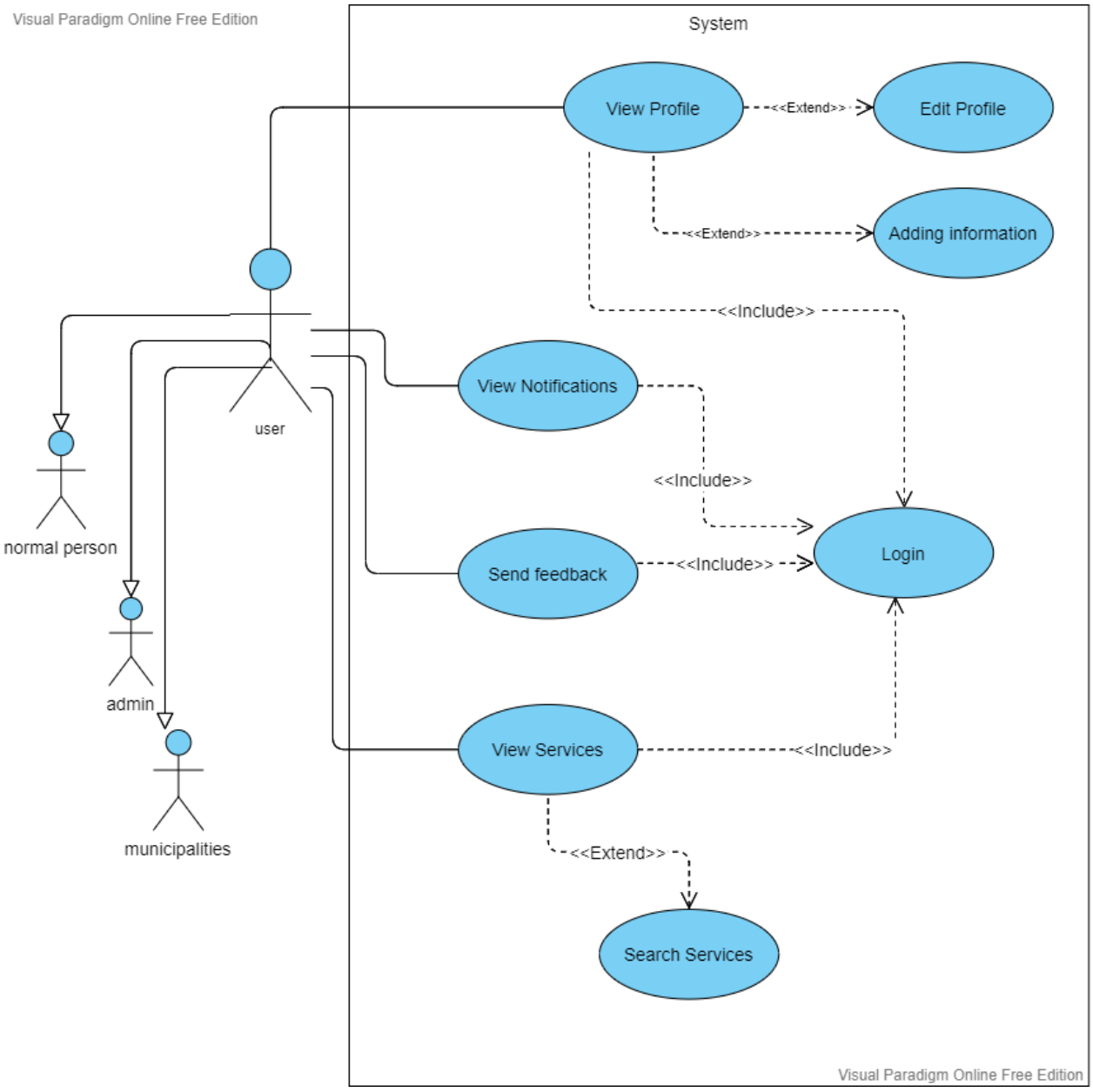
Login and Sign-up use case:

This image shows how the system provides users with the ability to submit an account registration request and how to log in and out.



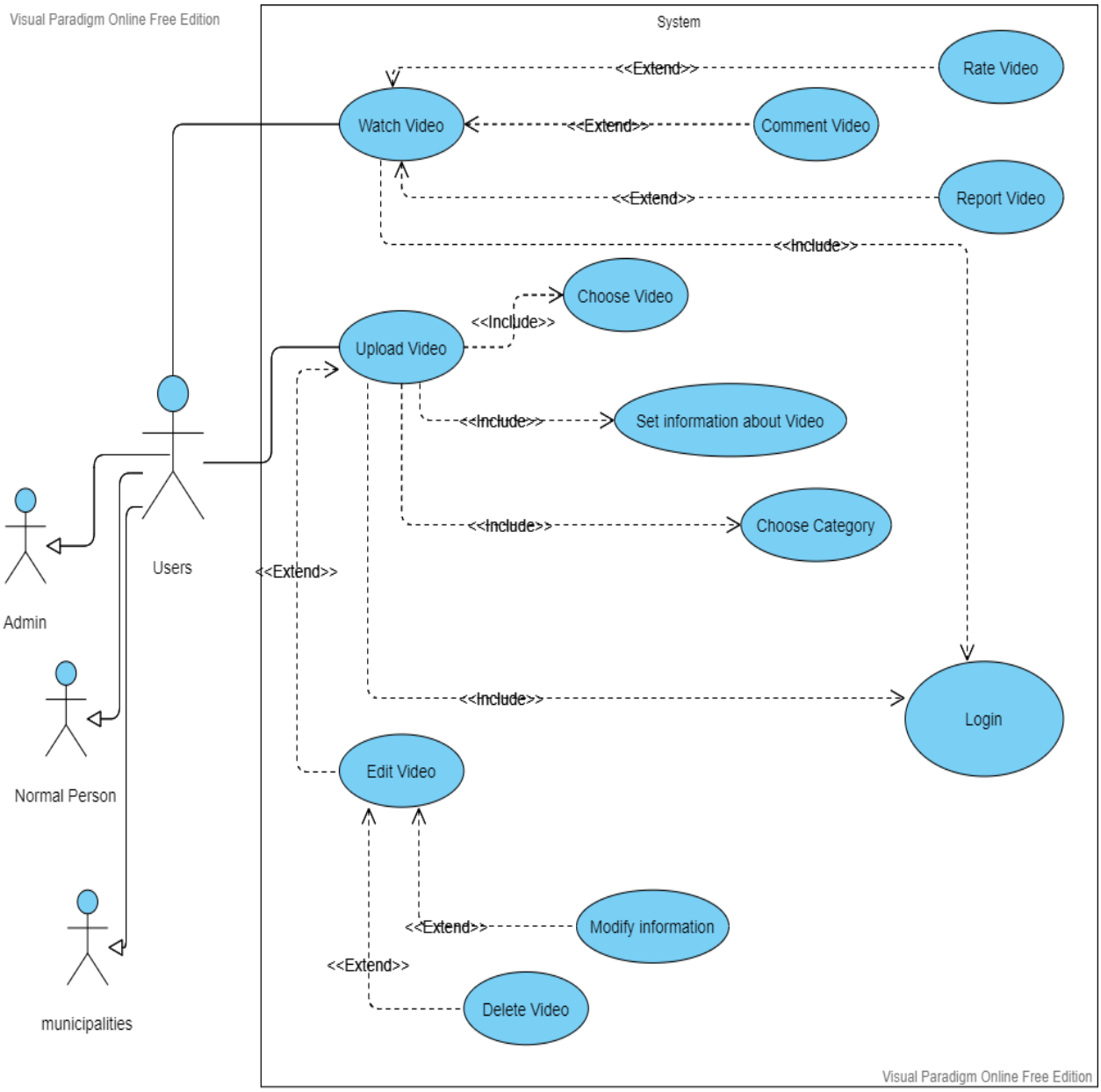
Profile use case:

This image shows how the system offers users the ability to control their profile and how to search through the system sections.



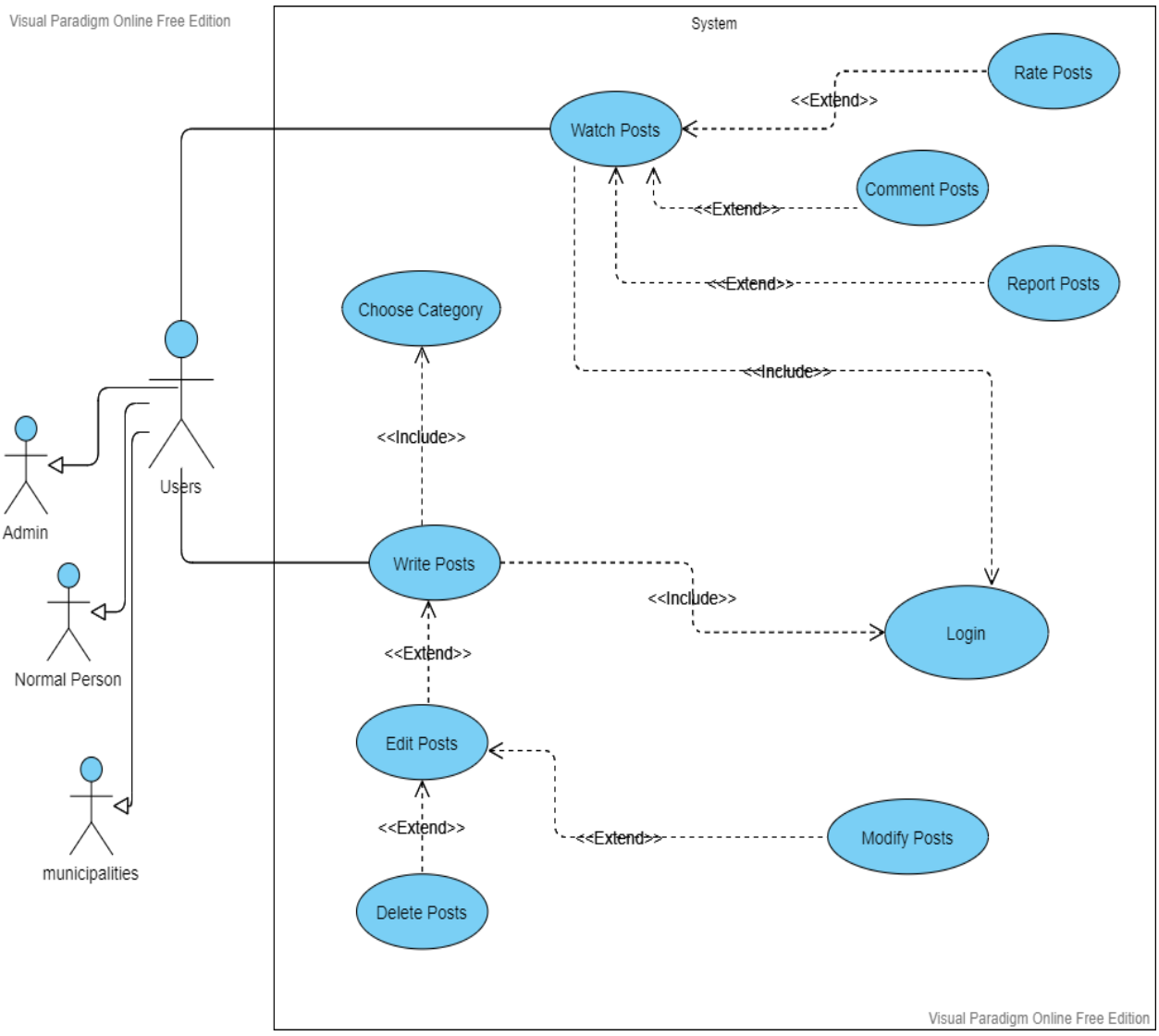
Video posts use case:

Here we can see the things that the system allows users to do with regard to the topic of videos.



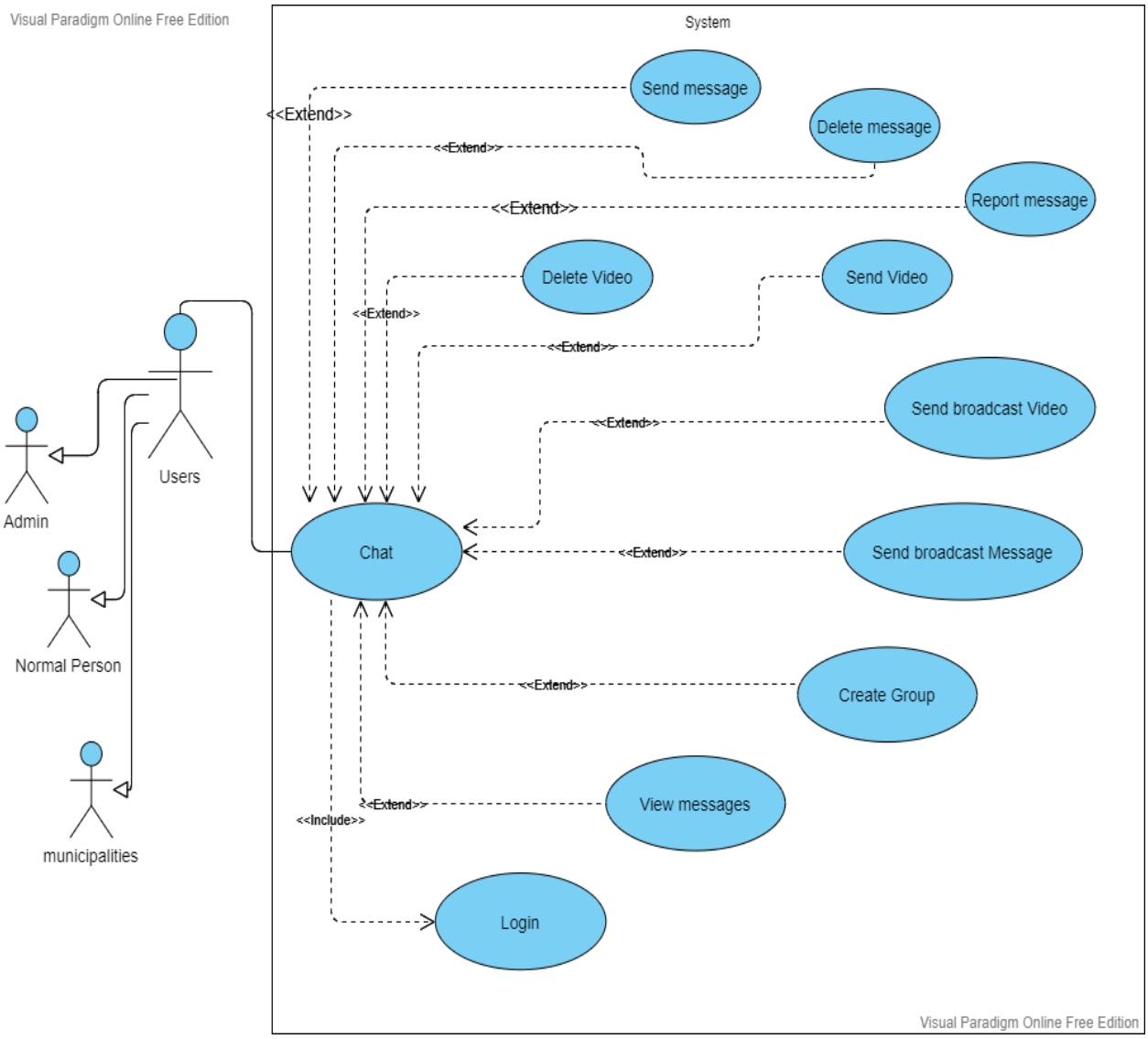
Write posts use case:

This image shows users what they can do with writing posts.



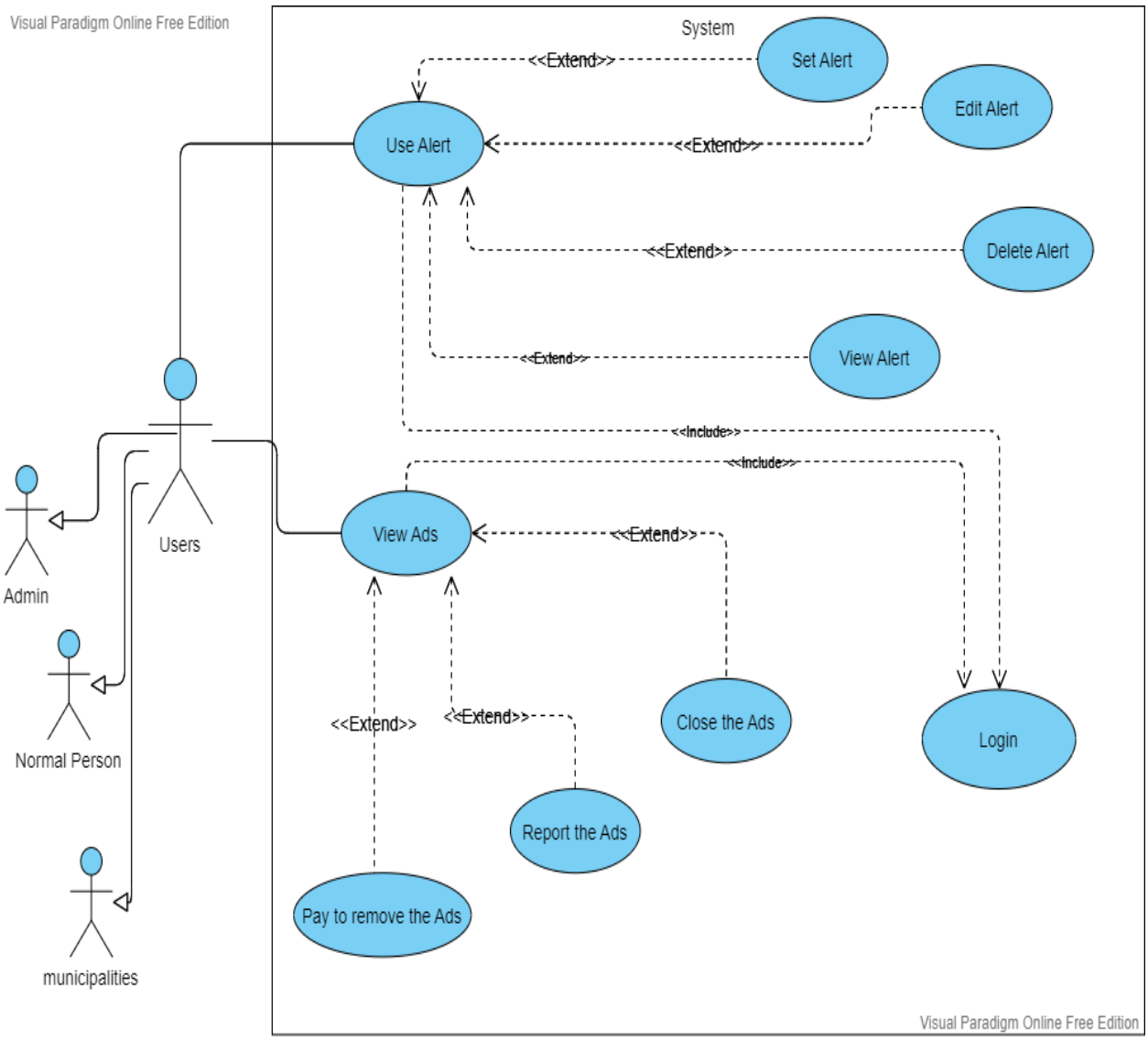
Chat use case:

This image is intended to show what users can do in the in-app chat area.



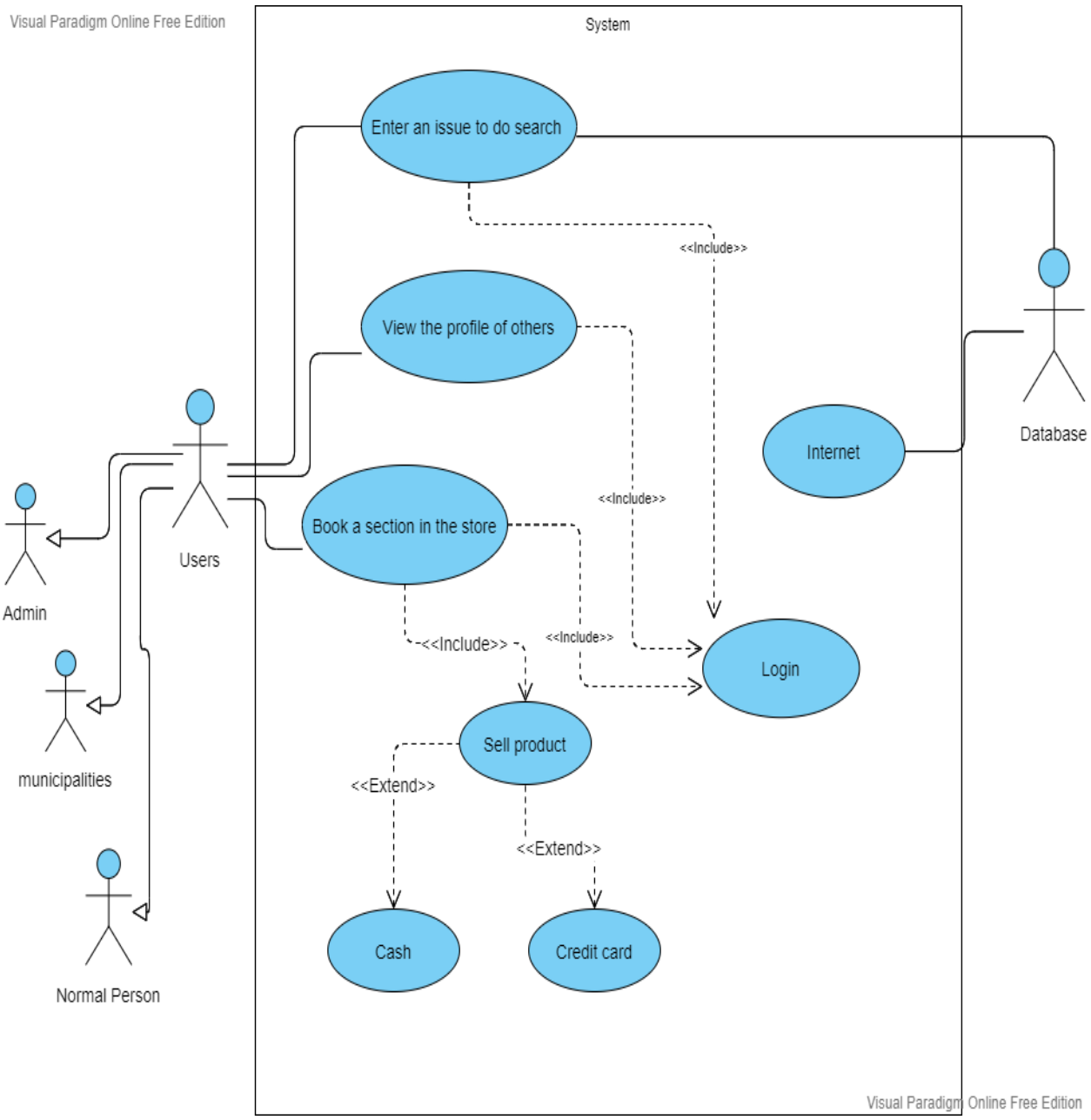
Alert and Ads use case:

It has been previously clarified that this application contains an alarm that users can benefit from, and the program also contains advertisements aimed at making profit for the programmer (the owner of the application). This image shows what users can do about these two topics.



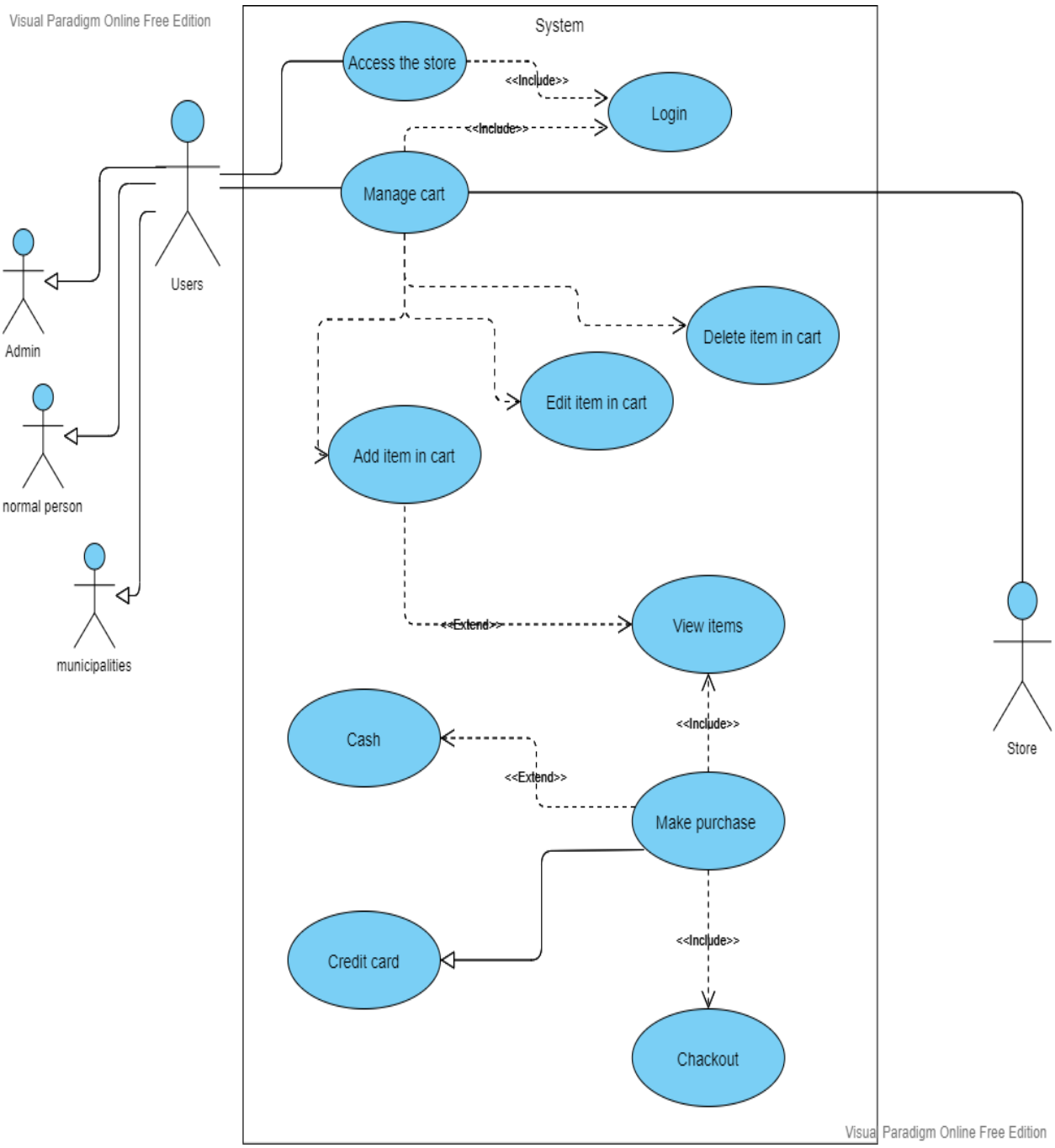
Search process, view profiles and book a section within the store:

The thing that this image shows is the process of searching within the system for any inquiry and viewing some personal information of other users, in addition to booking a section within the store to be able to sell recyclable materials for profit to users.



Store use case:

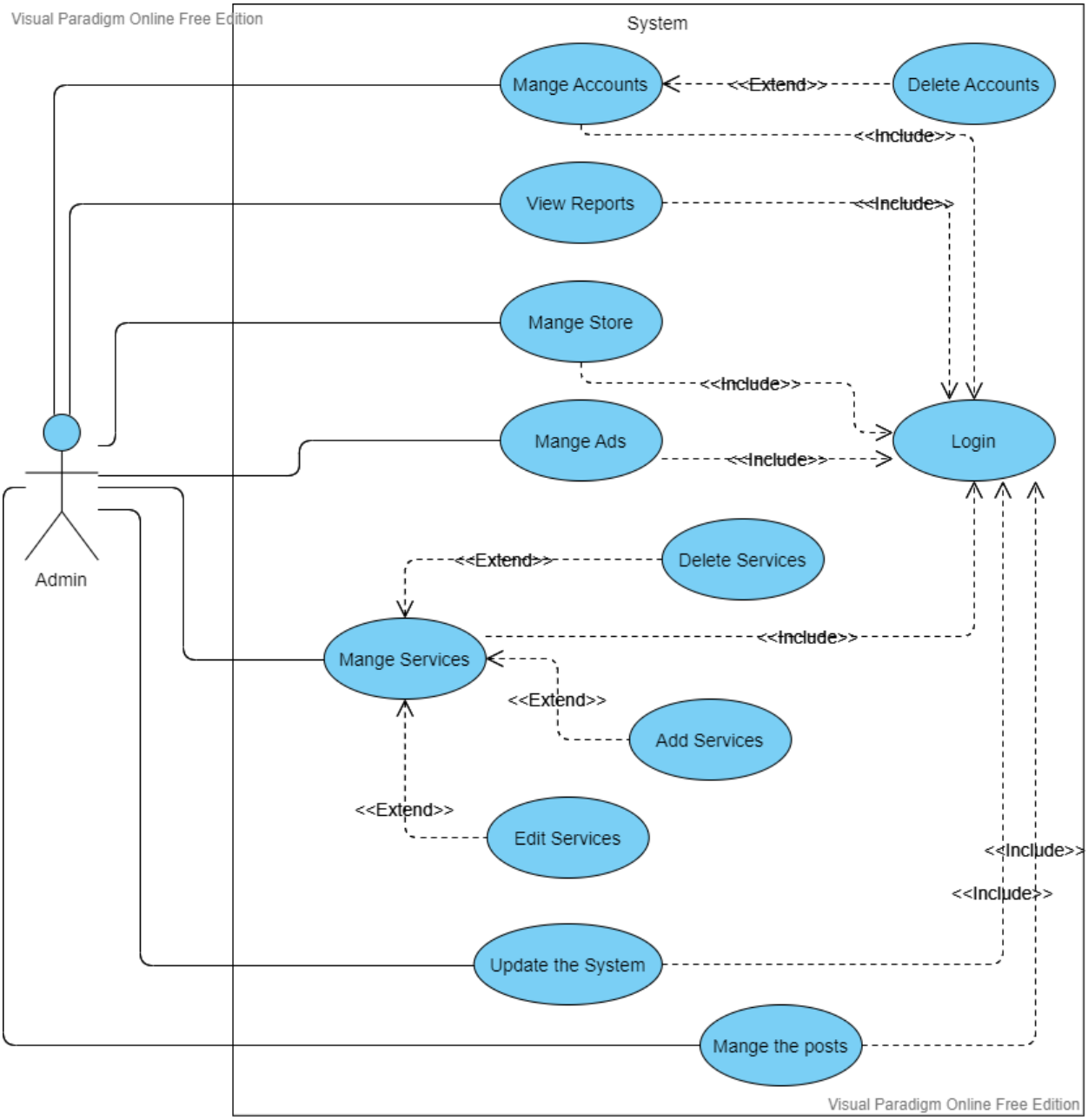
This picture is to clarify in particular matters related to the purchase process within the store.



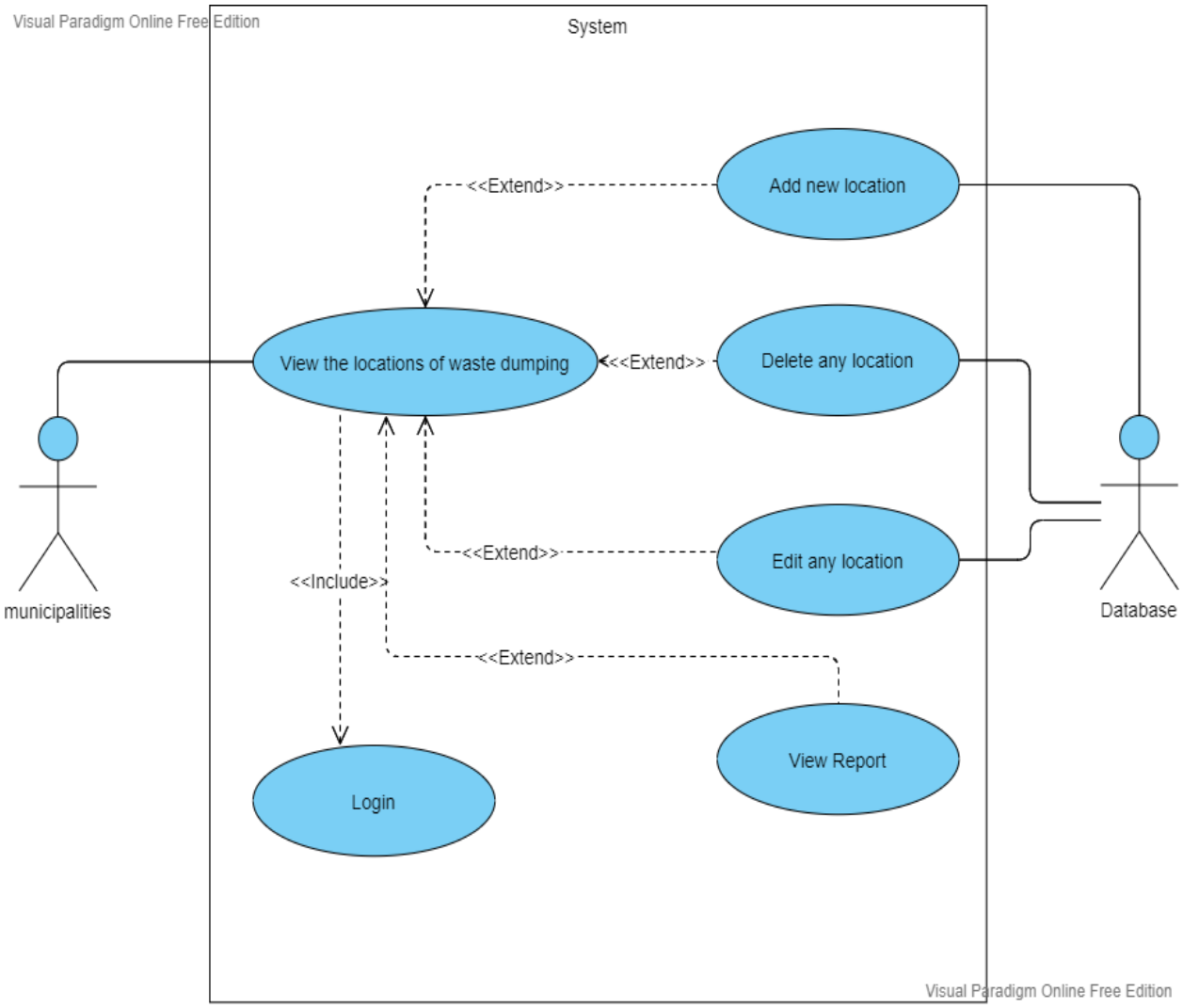
THE SPECIFIC THINGS FOR USERS:

Admin use case:

All the edit will be add to the system and to the database.



Municipalities use case:



NARRATIVE USE CASE:

WHAT IS NARRATIVE USE CASE?

A use case narrative is a largely text-based description of a use case that could be supplemented with decision trees or other easily understood notations. The description should be written in the use r's language, and thus provides an important communication tool between developers of systems and the intended use rs. (GLOBAL, n.d.)

NARRATIVE USE CASE FOR THE STORE:

main actors:

1. Admin.
2. Normal user.
3. Municipalities.

secondary actors:

1. Store.

Precondition:

1. The user has sign up.
2. The user entered his name.
3. The user entered his email address.
4. The user entered his password.

Normal flow of action:

1. The user logs in.
2. The user entered their email addresses.
3. The user entered the password.
4. The user reserved a section in the store if they want to sell items.
5. The user added their bank account number if they want to buy or sell items using credit card.

Alternative flow of action:

1. The user may enter the wrong email.
2. The user may enter the wrong password.
3. The user may ask to change the password in case he forgot the original one.
4. The system will send him a verification message.
5. The user can forget to fill in the bank account number in order to transfer the money to him.
6. The user may fill in the wrong bank account number.
7. The system will send him a notification message.

8. The user may pay for wrong item.
9. The system should allow him to cancel the operation.

Postcondition:

1. The store should remove one item from the database every time a purchase is made.
2. The store should add one item to the database every time a product is added for sale.
3. The store should transfer the money between the users.

NARRATIVE USE CASE FOR THE MUNICIPALITIES:

main actors:

1. Municipalities.

secondary actors:

1. Database.

Precondition:

1. The Municipalities has sign up.
2. The Municipalities entered his name.
3. The Municipalities entered his email address.
4. The Municipalities entered his password.

Normal flow of action:

1. The Municipalities logs in.
2. The Municipalities entered their email addresses.
3. The Municipalities entered the password.
4. The Municipalities added new waste dumped locations.
5. The Municipalities edit the previous waste dumped locations.
6. The Municipalities deleted some of the previous waste dumped locations.

Alternative flow of action:

1. The Municipalities may enter the wrong email.
2. The Municipalities may enter the wrong password.
3. The Municipalities may ask to change the password in case he forgot the original one.
4. The system will send him a verification message.

Postcondition:

1. The system will add the information modified by the Municipalities.

NARRATIVE USE CASE FOR ALERT:

main actors:

1. Admin.
2. Normal user.
3. Municipalities.

secondary actors:

1. System

Precondition:

1. The user has sign up.
2. The user entered his name.
3. The user entered his email address.
4. The user entered his password.

Normal flow of action:

1. The user added the alarm.
2. The user may edit the alarm.
3. The user may delete the alarm.

Alternative flow of action:

1. The user may enter the wrong email.
2. The user may enter the wrong password.
3. The user may ask to change the password in case he forgot the original one.
4. The system will send him a verification message.
5. The user may set wrong alarm.

Postcondition:

1. The system will add the new modified to the database.
2. The system will activate the alarm on the time set by the user.

This paper explains in a simplified way to all the people who can see it how the system works in detail. Also, some references have been added that the reader can see, which will help him understand the idea more clearly and effectively.

ER Diagram

INTRODUCTION ABOUT ER DIAGRAM:

WHAT IS DATA MODELING?

Discover how data modeling uses abstraction to represent and better understand the nature of data flow within an enterprise information system. (IBM, n.d.)

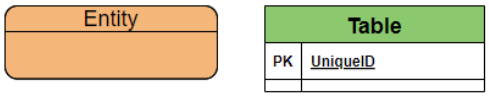
Like any design process, database and information system design begins at a high level of abstraction and becomes increasingly more concrete and specific. Data models can generally be divided into three categories, which vary according to their degree of abstraction. The process will start with a conceptual model, progress to a logical model and conclude with a physical model.

WHAT IS ER DIAGRAM?

An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how "entities" such as people, objects or concepts relate to each other within a system. ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education and research. Also known as ERDs or ER Models, they use a defined set of symbols such as rectangles, diamonds, ovals and connecting lines to depict the interconnectedness of entities, relationships and their attributes. They mirror grammatical structure, with entities as nouns and relationships as verbs. (lucidchart, n.d.)

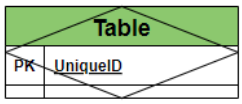
The components and features of an ER diagram:

1.1.1.1 Entity:



A definable thing—such as a person, object, concept or event—that can have data stored about it. Think of entities as nouns. Examples: a customer, student, car or product. Typically shown as a rectangle.

1.1.1.2 Associative entity:



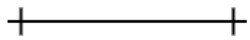
An associative entity associates entities (or elements) within an entity set.

1.1.1.3 Cardinality:

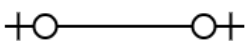
Defines the numerical attributes of the relationship between two entities or entity sets. The three main cardinal relationships are one-to-one, one-to-many, and many-many.

1.1.1.3.1 What is used in this graph is:

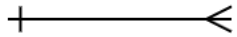
One-to-One



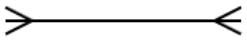
One or Zero



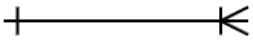
One-to-Many



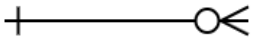
Many-to-Many



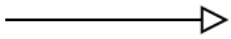
One-to-One or Many



One-to-Zero or one or Many



Generalization



1.1.1.4 Primary Key:

Primary Key is a set of attributes (or attribute) which uniquely identify the tuples in relation or table. The primary key is a minimal super key, so **there is one and only one primary key in any relationship**. It will be denoted in the table as (**PK**).

1.1.1.5 Foreign Key:

A foreign key is a set of attributes in a table that refers to the primary key of another table. The foreign key links these two tables. It will be denoted in the table as (**FK**).

1.1.1.6 Attribute:

A descriptive property or characteristic of an entity. Synonyms include element, property, and field.

Example:

Student entity have {First name, Last name, Address, …….}

1.1.1.7 Data type:

A property of an attribute that identifies what type of data can be stored in that attribute.

Example:

Number → Integer

Text → Varchar

EXPLANATION OF THE TABLES AND WHAT THEY INCLUDE:

Account permission:

These are the permissions allowed for each account. It constitutes an essential separation point between the user account and the admin account.

Table 3 ***Account permission***

Attribute name	Data type	Required	Description
Permission ID	int	yes	To give a number that distinguishes the permissions
Permission Name	varchar	yes	to add name for each permission
Details	varchar	yes	To give some explanation for each permission

Account Rule:

These are the terms and conditions that the user must abide by and observe during his use of this application.

Table 4 ***Account Rule***

Attribute name	Data type	Required	Description
Account rule ID	int	yes	To give a number that distinguishes the Rules
Rule Name	varchar	yes	to add name for each Rule
Details	varchar	yes	To give some explanation for each Rule

Municipality Account:

It is an account belonging to a municipality and not to a person. It is an official account as it indicates a government department concerned with helping citizens living within the scope of this municipality.

Table 5 ***Municipality Account***

Attribute name	Data type	Required	Description
Municipality account ID	int	yes	To give a number that distinguishes the Municipalities
Municipality Name	varchar	yes	to add name for each Municipality
Email	varchar	yes	To add an email to each account
Password	varchar	yes	To add a password to protect the account
Address	varchar	yes	To find out the address of the municipality
Phone Number	int	no	To add a phone number to contact the municipality
Credit card	varchar	no	Add a payment method and receive money

User Account:

In order for the general people to create an account and benefit from the application.

Table 6 ***User Account***

Attribute name	Data type	Required	Description
User account ID	int	yes	To give a number that distinguishes the Users
User Name	varchar	yes	to add name for each User
Email	varchar	yes	To add an email to each account
Password	varchar	yes	To add a password to protect the account
Address	varchar	no	To find out the address of the User
Phone Number	int	no	To add a phone number to contact the User
Credit card	varchar	no	Add a payment method and receive money

Dumps:

To know the number and location of garbage containers. And to take advantage of this information to display on the application to help people more.

Table 7 ***Dumps***

Attribute name	Data type	Required	Description
Dumps ID	int	yes	To give a number that distinguishes the Dumps
Dumps Name	varchar	no	to add name for each Dump
Address	varchar	yes	To find out the address of the Dumps
Number of garbage container	int	no	To know or estimate the volume of the landfill

Online Store:

To add a feature and the ability for all users of all levels to sell items online without the need for a fixed place such as a shop.

Table 8 ***Online Store***

Attribute name	Data type	Required	Description
Online Store ID	int	yes	To give a number that distinguishes the online stores
Store phone number	int	yes	To add phone number to the store
Address	varchar	no	to add address if there
Delivery method	int	yes	to add the way of delivery

Physical Store:

To allow people who have fixed stores and want to sell materials through these stores and do not prefer selling via the Internet.

Table 9 ***Physical Store***

Attribute name	Data type	Required	Description
Physical Store ID	int	yes	To give a number that distinguishes the physical stores
Store phone number	int	no	To add phone number to the store
Address	varchar	yes	to add address if there

Cart:

It is the place where information about all items that have been purchased or reserved to be purchased at a later time are stored.

Table 10 ***Cart***

Attribute name	Data type	Required	Description
Cart ID	int	yes	To give a number that distinguishes the physical stores
Total items price	int	yes	To add the price group of the reserved items
Reserve Date	varchar	yes	To save the date of reserve
Purchase Date	varchar	yes	To save the date of purchase

Product Category:

To determine the types of materials that can be found in stores.

Table 11 ***Product Category***

Attribute name	Data type	Required	Description
Product Category ID	int	yes	To give a number that distinguishes the Product Category
Product Category Name	varchar	yes	to add name for each Product Category
Material	varchar	yes	To set the material type of the product

Product:

To identify and distinguish the materials inside each store and add other details to it.

Table 12 ***Product***

Attribute name	Data type	Required	Description
Product ID	int	yes	To give a number that distinguishes the Product
Product Name	varchar	yes	to add name for each Product
Price	int	yes	To set a price for the product
Quantity	int	yes	To know the number of pieces

Profile:

To add additional personal details about users.

Table 13 ***Profile***

Attribute name	Data type	Required	Description
Profile ID	int	yes	To give a number that distinguishes the Profiles
Scientific Specialization	varchar	no	To set a scientific Specialization for the user
work in	varchar	no	To add where the user is working

Post:

To get information about posts that user add.

Table 14 ***Post***

Attribute name	Data type	Required	Description
Post ID	int	yes	To give a number that distinguishes the Posts
Title	varchar	no	To get the title of the post
Discerption	varchar	no	To add some details to the post

Report:

To take information about the report in case there is any problem with the publications.

Table 15 ***Report***

Attribute name	Data type	Required	Description
Report ID	int	yes	To give a number that distinguishes the Reports
Message	varchar	yes	To add some details to the report

LOGICAL MODEL DEVELOPMENT STAGES:

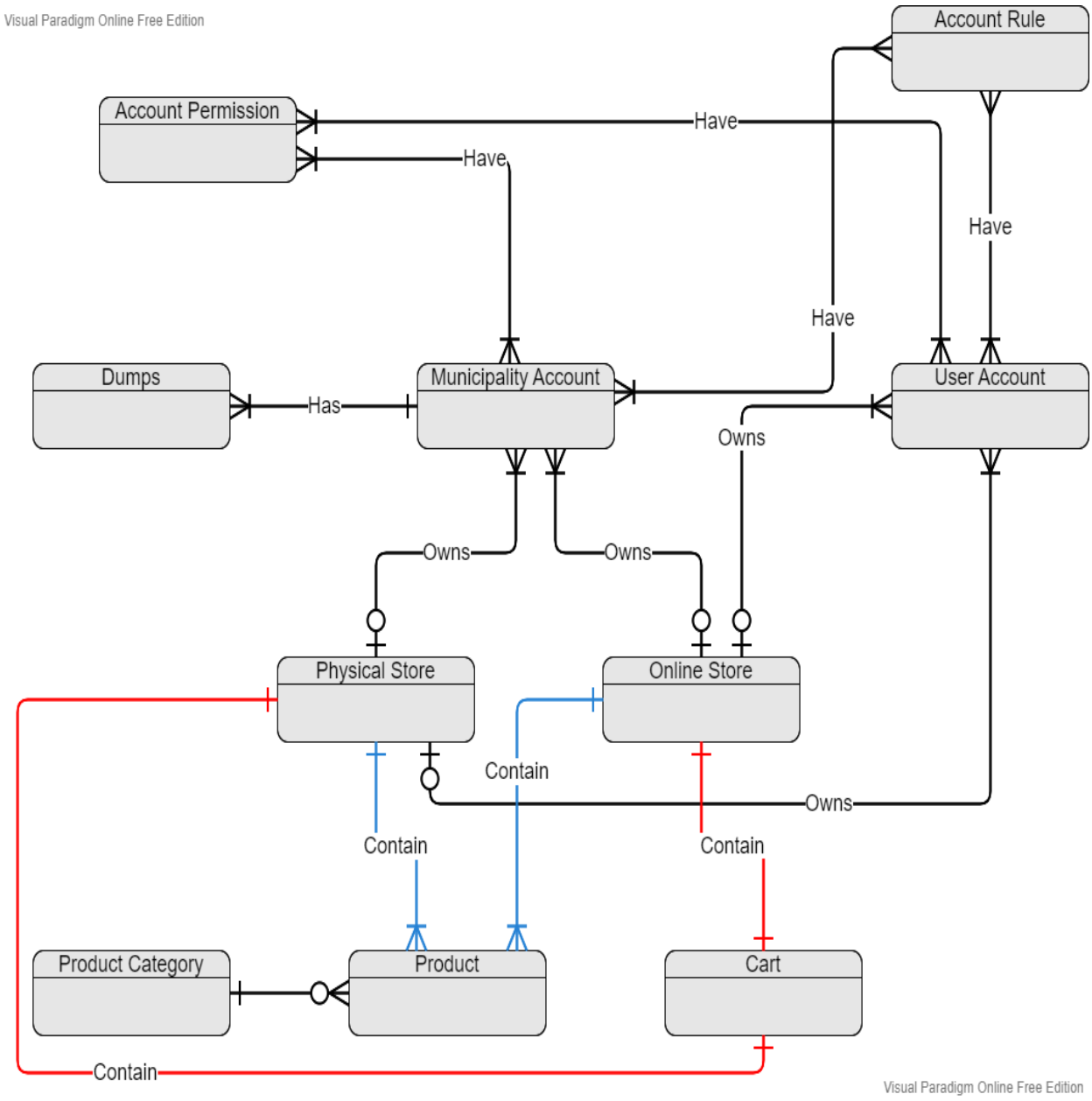
AN IMPORTANT POINT TO CLARIFY:

in each section, ER diagrams are segmented into smaller pieces for ease of reading, and at the end of each section the all the pieces are grouped into one final form to show the full scope of the project.

CONTEXT DATA MODEL:

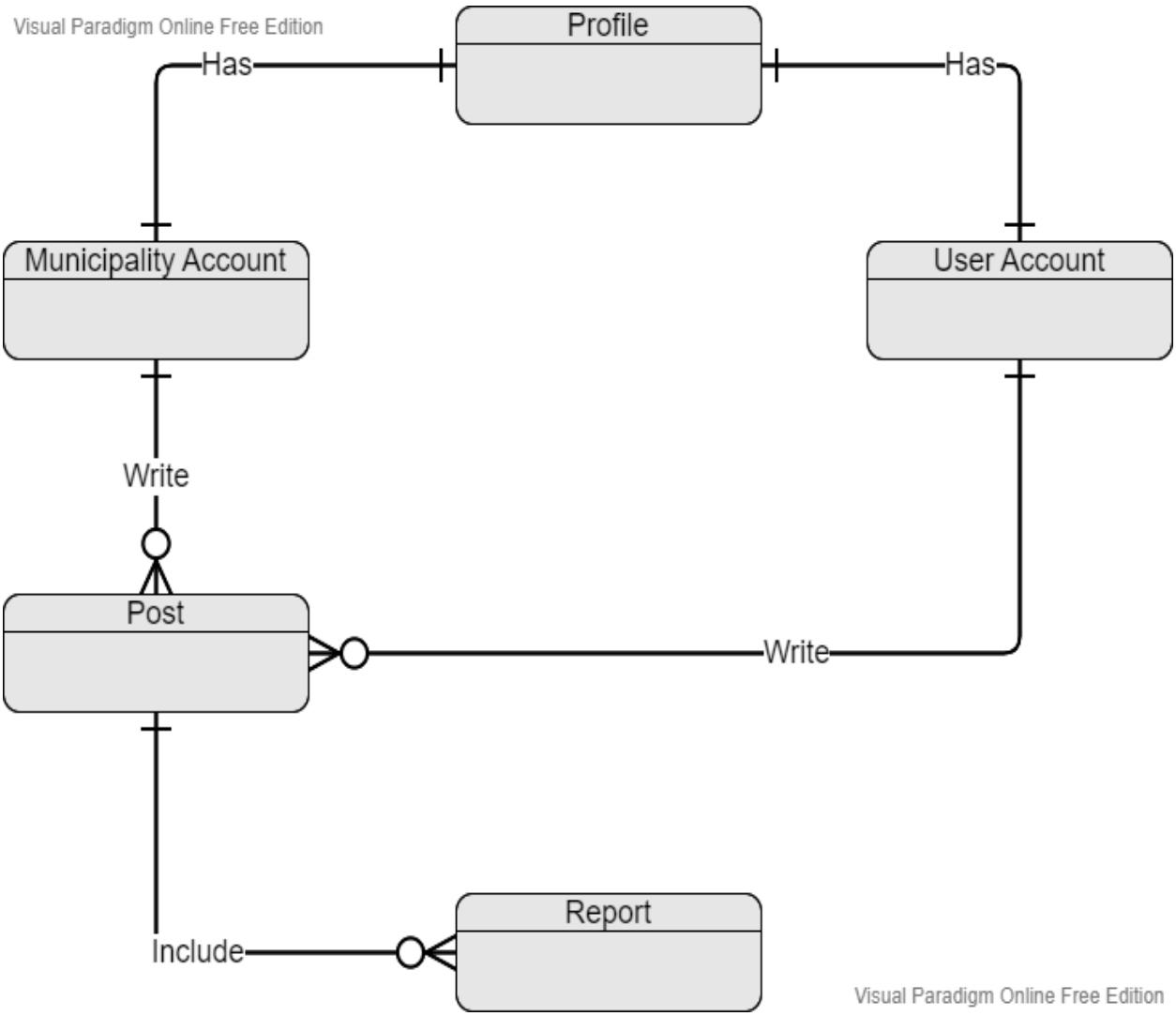
A context model (or context modeling) defines how context data are structured and maintained (It plays a key role in supporting efficient context management). It aims to produce a formal or semi-formal description of the context information that is present in a It is used to represent the reusable context information of the components (The top-level classes consist of Operating system, component container, hardware requirement and Software requirement). (wikipedia, n.d.)

Part one:


Visual Paradigm Online Free Edition
Account Rule
Account Permission
Have
Have
Have
Have
Dumps
Has
Municipality Account
User Account
Owns
Owns
Owns
Physical Store
Online Store
Contain
Owns
Contain
Contain
Product Category
Product
Cart
Contain
Visual Paradigm Online Free Edition

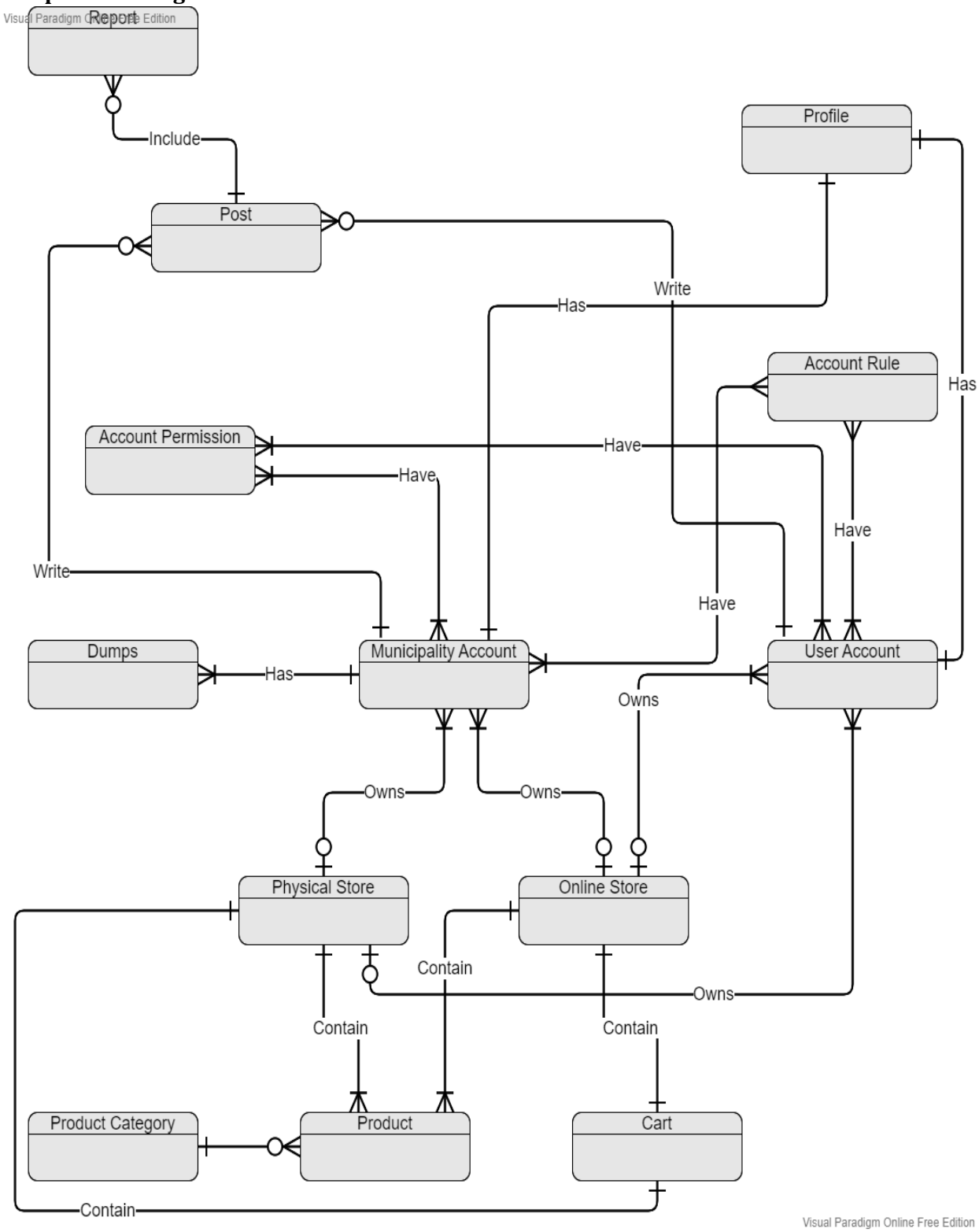
- **The relation between account permission and user account:**
  User account can **have** one or many permissions
  (The same applies to the municipality's account).
- **The relation between account rule and user account:**
  The user account **have** many rules that must be observed.
  (The same applies to the municipality's account).
- **The relation between the municipality account and the dumps:**
  The municipality **has** one or more landfills and it manage them.
- **The relation between the user account and the online store:**
  The user account **owns** one online store or does not own any online store.
  (The same applies to the municipality's account).
- **The relation between the user account and the physical store:**
  The user account **owns** one physical store or does not own any physical store.
  (The same applies to the municipality's account).
- **The relation between the online store and the cart:**
  Each online store **contains** a special shopping cart.
  (The same applies to the physical store).
- **The relation between the product category and the product:**
  Each product is **within** one of the product categories.
- **The relation between the online store and the product:**
  Each online store **contains** its own products.
  (The same applies to the physical store).

Part two:



- **The relation between the user account and the profile:**
  Each user account **has** its own profile.
  (The same applies to the municipality account).
- **The relation between the user account and the post:**
  Each user account can **write** posts if they want to do that.
  (The same applies to the municipality account).
- **The relation between the post and report:**
  Each post can **include** a report or more in case that one or more users do so.

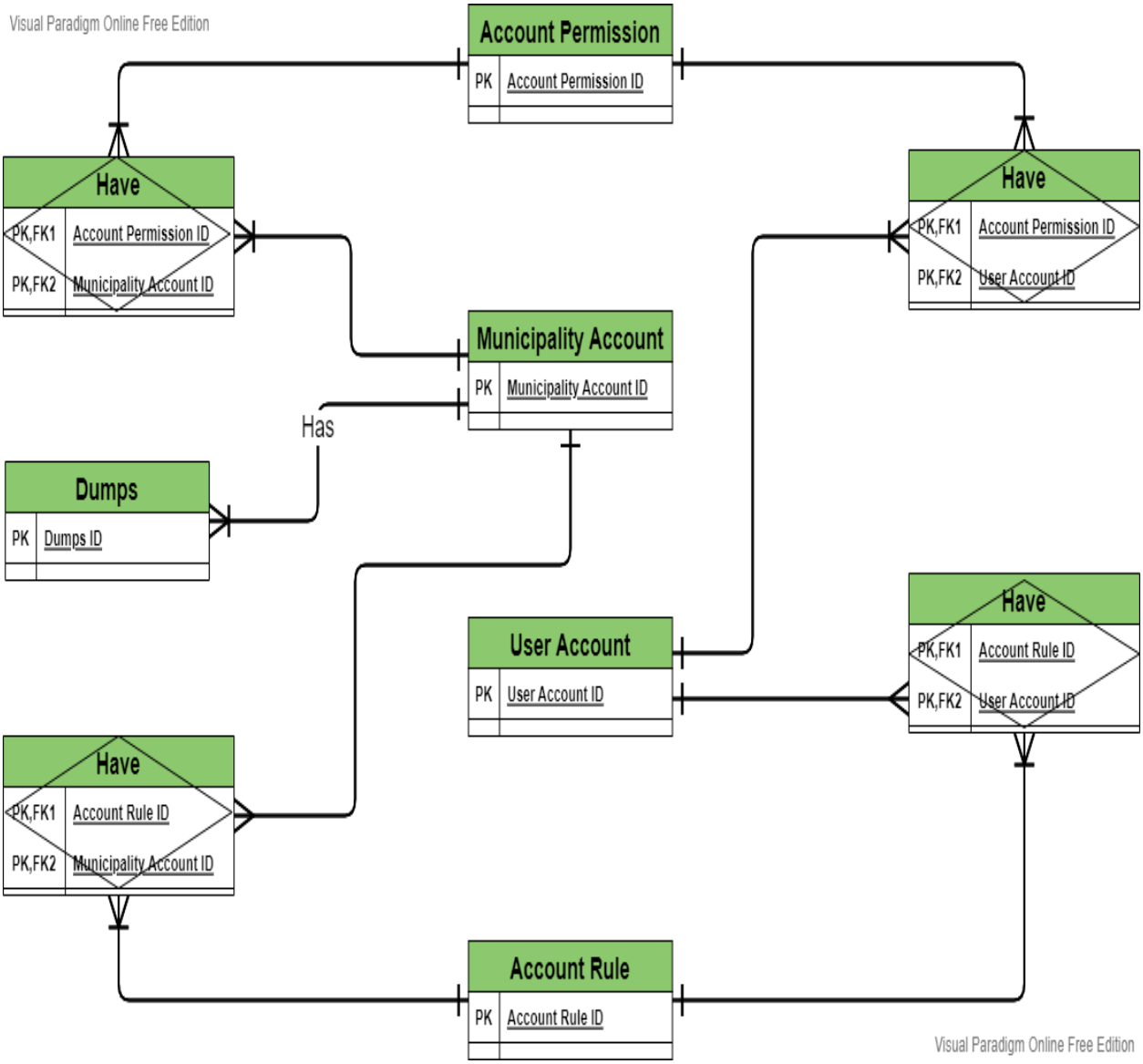
Complete ER Diagram:



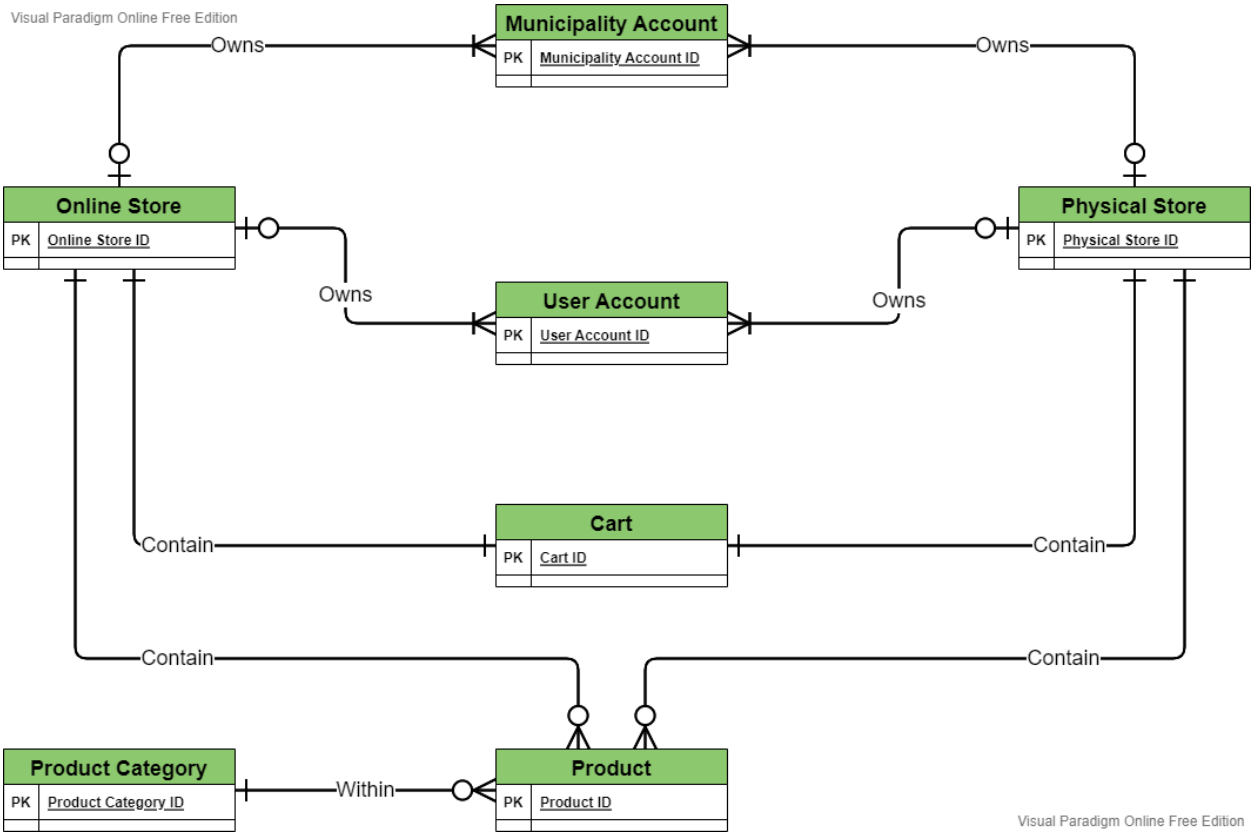
KEY-BASE DATA MODEL:

A data model (or data model) is an abstract model that organizes elements of data and standardizes how they relate to one another and to the properties of real-world entities. For instance, a data model may specify that the data element representing a car be composed of a number of other elements which, in turn, represent the color and size of the car and define its owner. (base, n.d.)

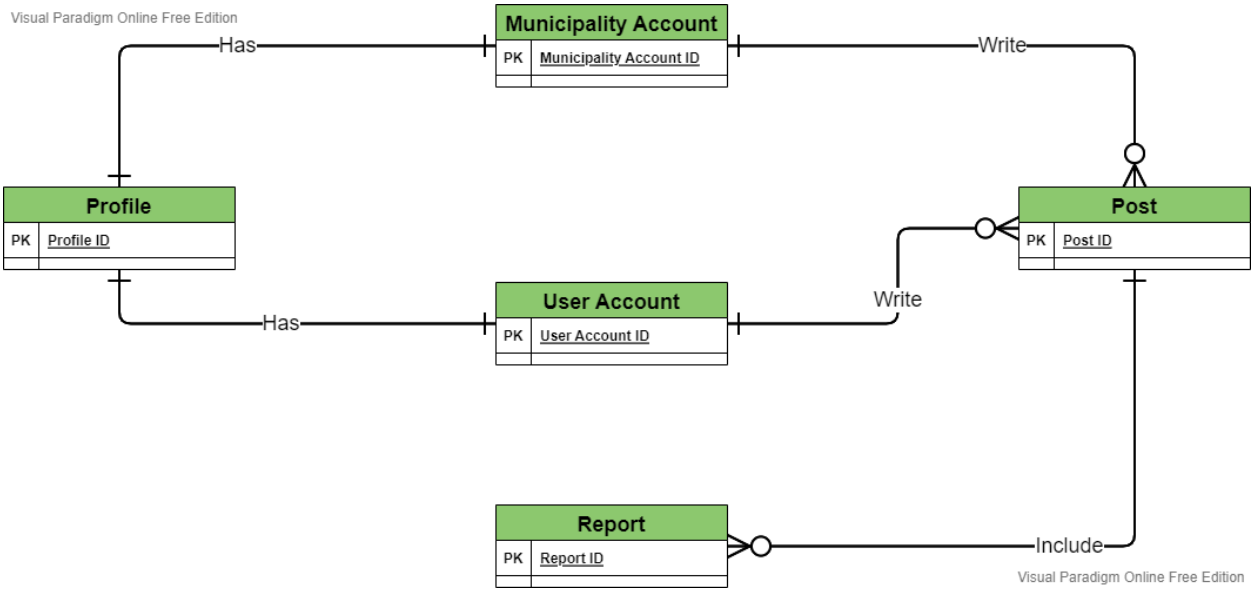
Part one:



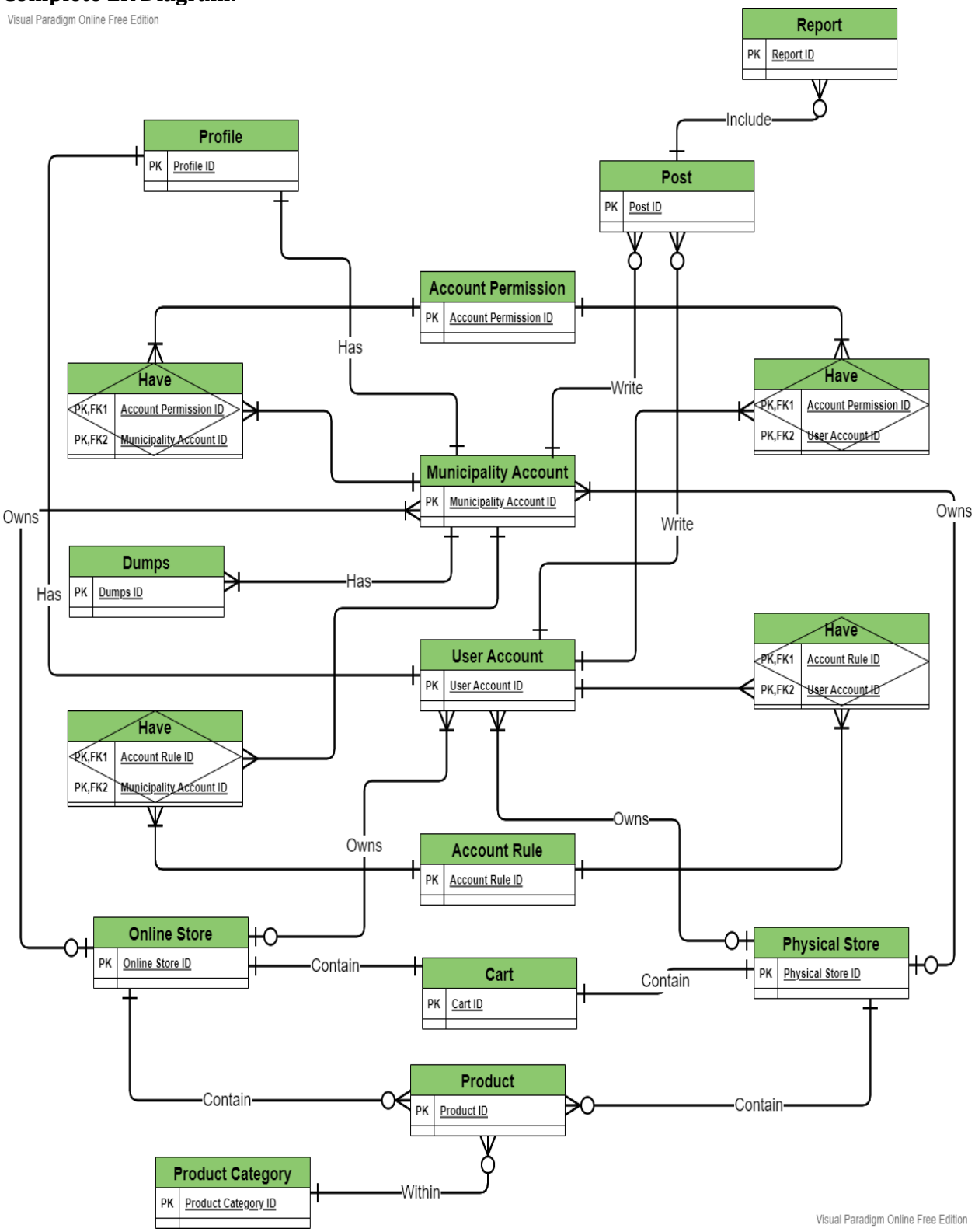
Part two:



Part three:



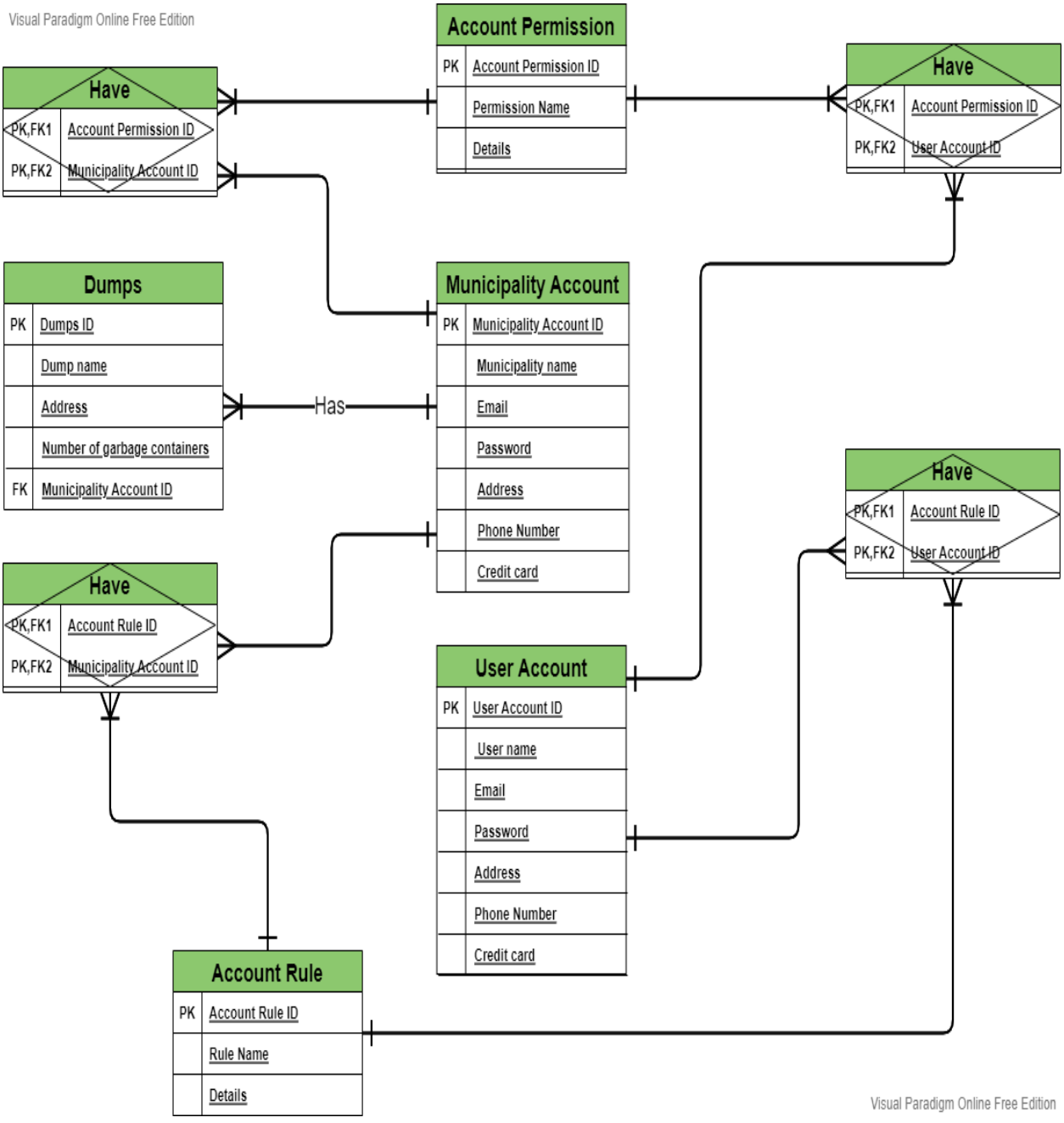
Complete ER Diagram:



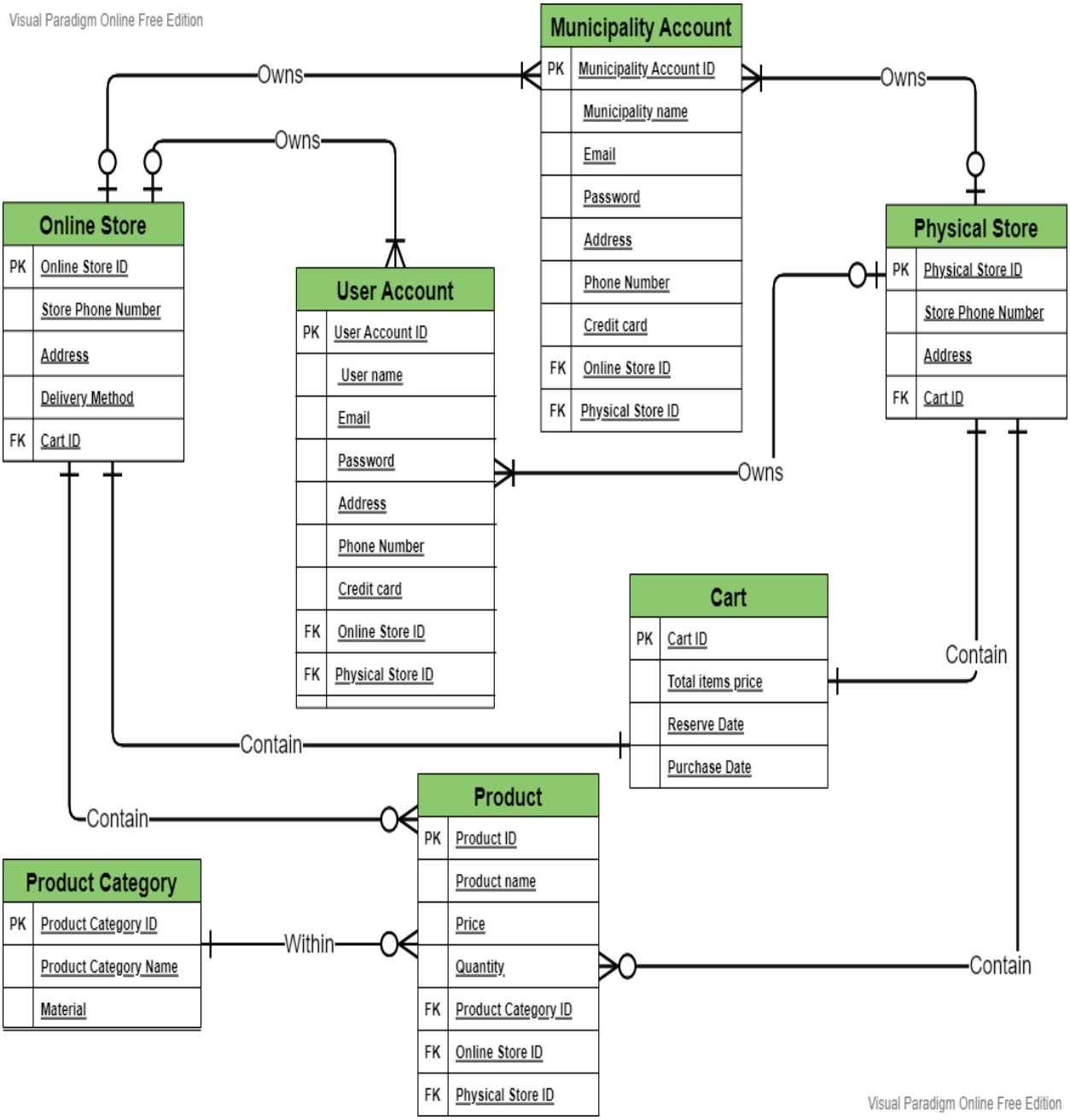
FULLY ATTRIBUTED DATA MODEL:

A fully attributed data model contains detailed attributes (descriptions) for every entity within it. The term "database design" can describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of as the logical design of the base data structures used to store the data. (attributed, n.d.)

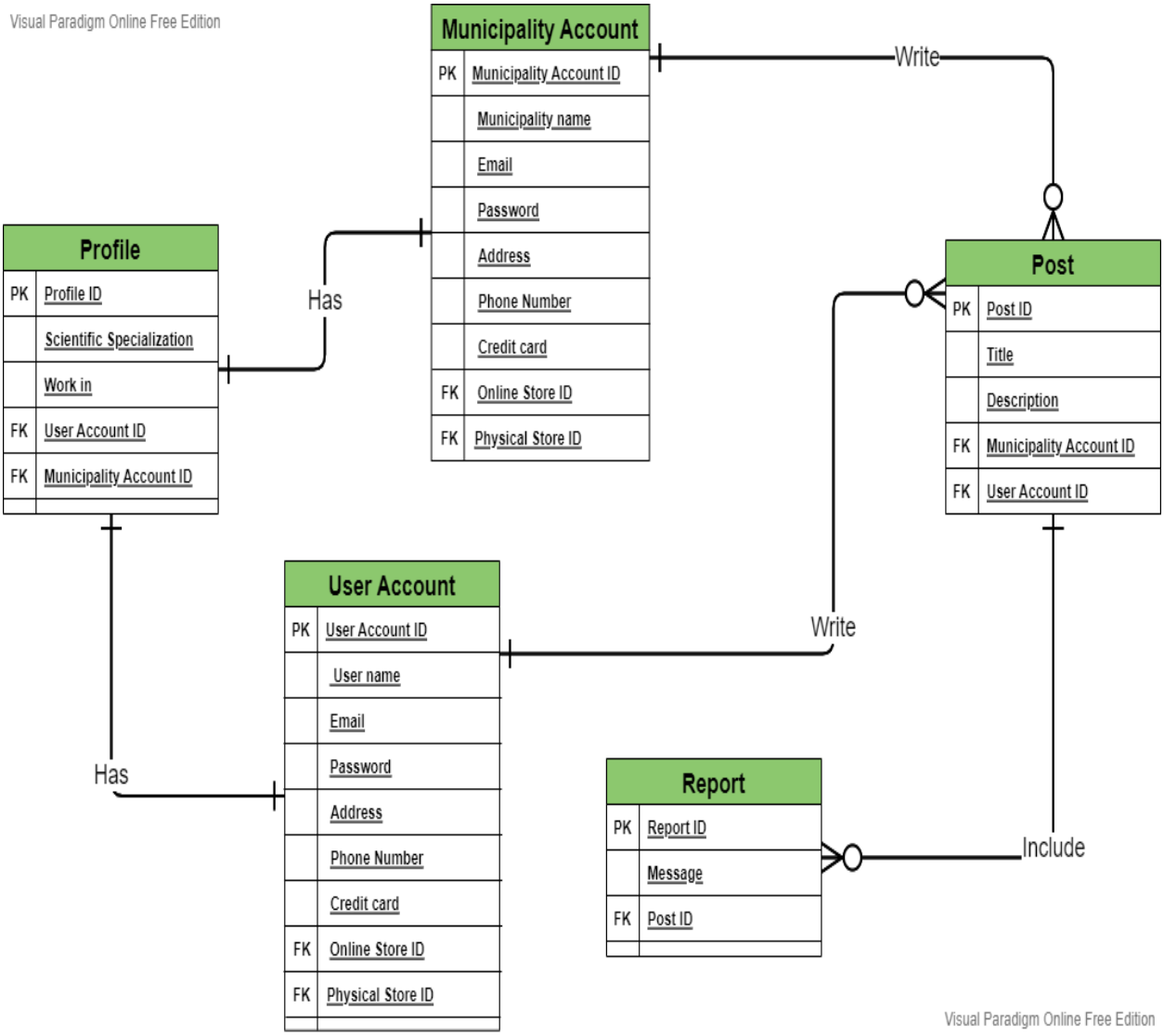
Part one:



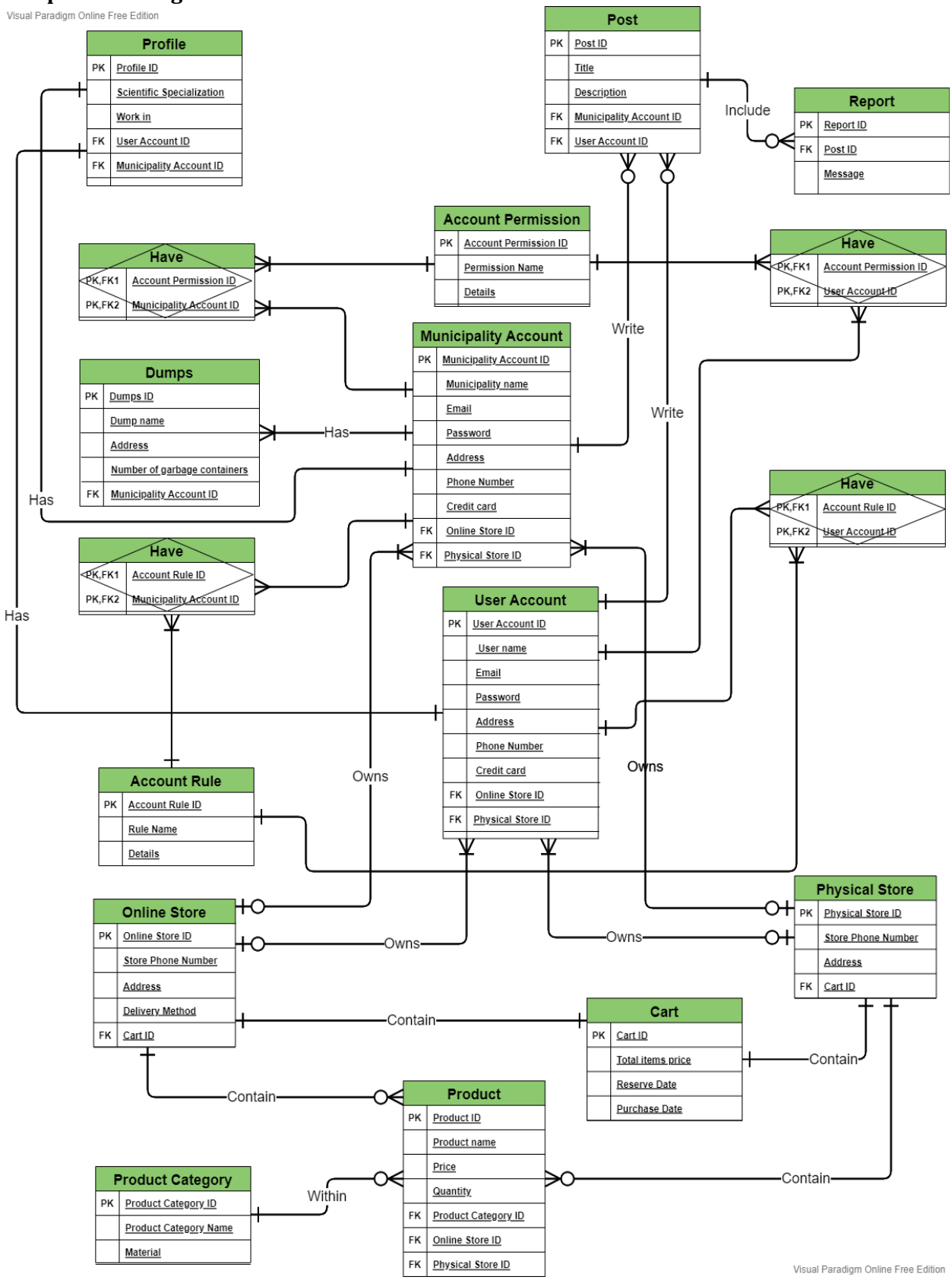
Part two:


Visual Paradigm Online Free Edition
Municipality Account
PK Municipality Account ID
Municipality name
Email
Password
Address
Phone Number
Credit card
FK Online Store ID
FK Physical Store ID
Owns
Owns
Owns
Online Store
PK Online Store ID
Store Phone Number
Address
Delivery Method
FK Cart ID
User Account
PK User Account ID
User name
Email
Password
Address
Phone Number
Credit card
FK Online Store ID
FK Physical Store ID
Owns
Physical Store
PK Physical Store ID
Store Phone Number
Address
FK Cart ID
Cart
PK Cart ID
Total items price
Reserve Date
Purchase Date
Contain
Contain
Contain
Product
PK Product ID
Product name
Price
Quantity
FK Product Category ID
FK Online Store ID
FK Physical Store ID
Product Category
PK Product Category ID
Product Category Name
Material
Within
Contain
Visual Paradigm Online Free Edition

Part three:


Visual Paradigm Online Free Edition
Municipality Account
PK Municipality Account ID
Municipality name
Email
Password
Address
Phone Number
Credit card
FK Online Store ID
FK Physical Store ID
Write
Profile
PK Profile ID
Scientific Specialization
Work in
FK User Account ID
FK Municipality Account ID
Has
Post
PK Post ID
Title
Description
FK Municipality Account ID
FK User Account ID
User Account
PK User Account ID
User name
Email
Password
Address
Phone Number
Credit card
FK Online Store ID
FK Physical Store ID
Write
Has
Report
PK Report ID
Message
FK Post ID
Include
Visual Paradigm Online Free Edition

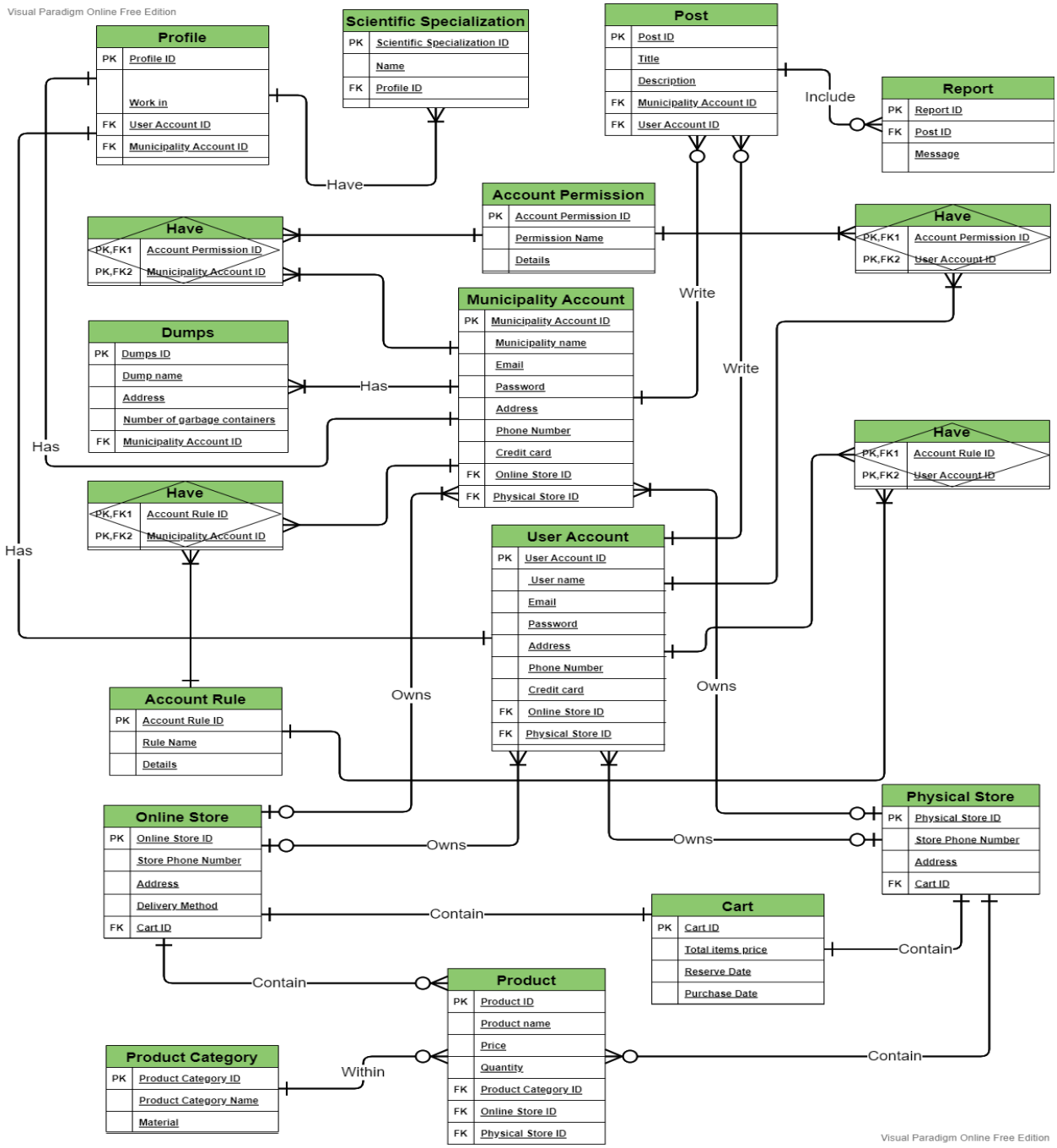
Complete ER Diagram:



NORMALIZED DATA MODEL:

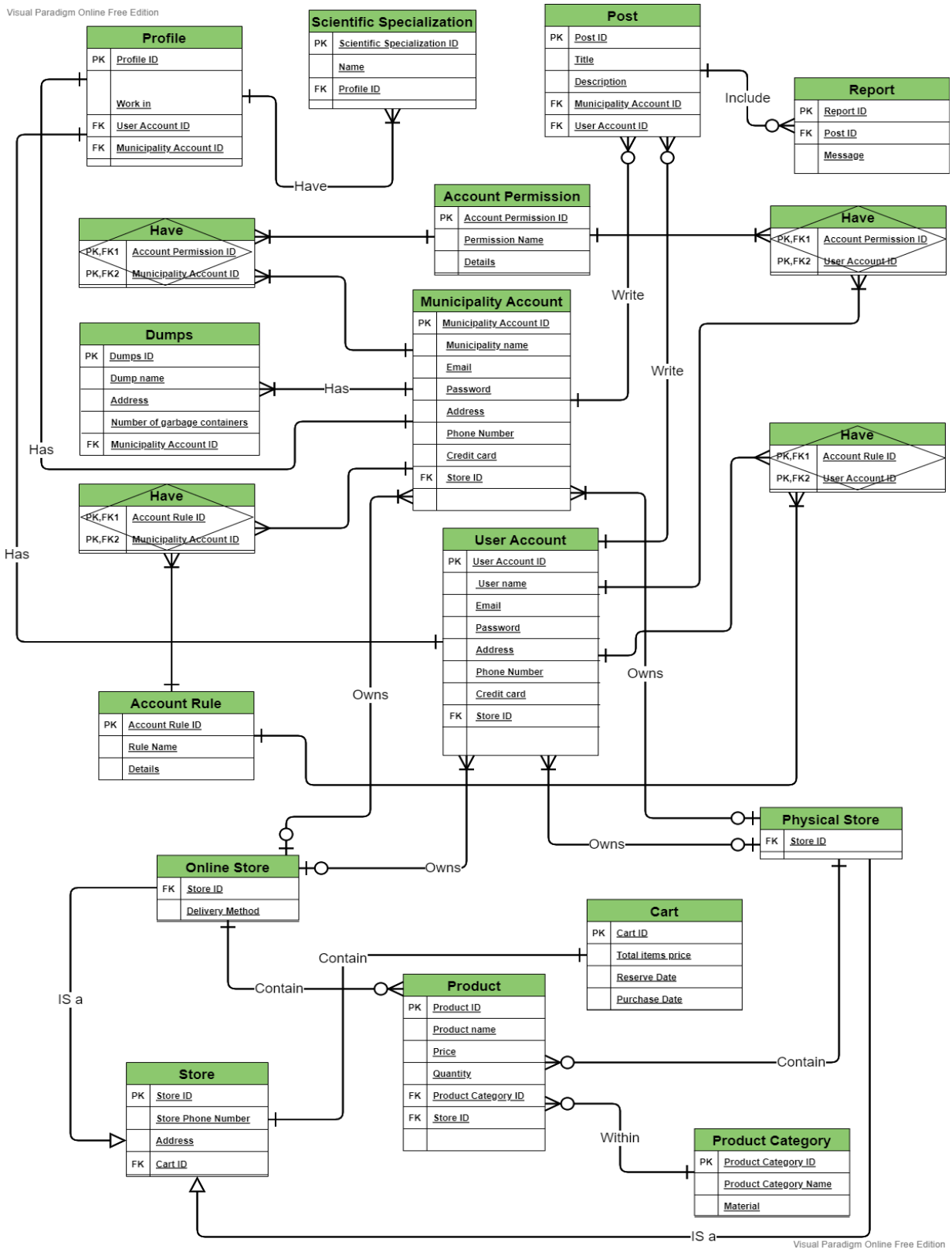
First normal form:
An entity whose attributes have no more than one value for a single instance of that entity. If there are any attributes that can have multiple values actually describe a separate entity, possibly an entity and relationship. (Newsletter, n.d.)

I notes that the scientific specialization attribute in the profile entity may have multivalued so I created a new entity for it.



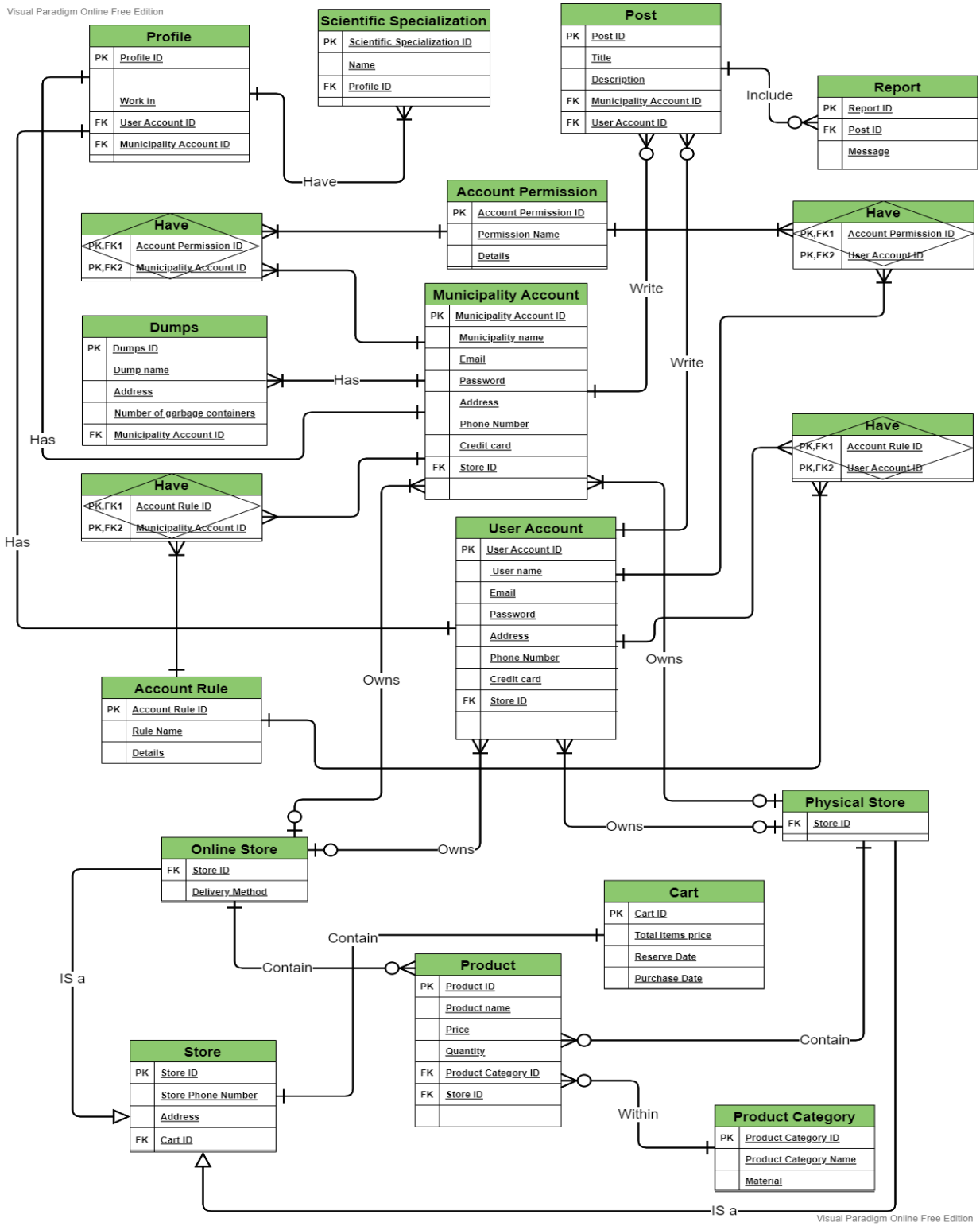
Second normal form:

an entity whose nonprimary-key attributes are dependent on the full primary key.



Third normal form:

There is a repetition of data between two entities, namely the online store entity and the physical store entity. To solve the problem, I created a new entity named store that have the common data in it with a relation to the tables.



An entity-relationship diagram, or ER diagram, is essential for modeling the data stored in a database. It is the basic design upon which a database is built. ER diagrams specify what data we will store: the entities and their attributes. They also show how entities relate to other entities. Another advantage of ERDs is that they represent the data in a graphical manner. This makes it easier for business folks to understand.

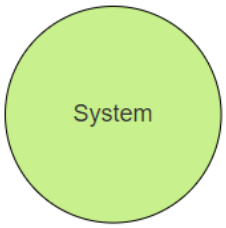
Data Flow Diagram

INTRODUCTION TO DATA FLOW DIAGRAM

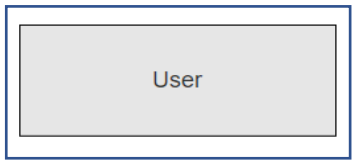
DFD is the abbreviation for **Data Flow Diagram**. The flow of data of a system or a process is represented by DFD. It also gives insight into the inputs and outputs of each entity and the process itself. DFD does not have control flow and no loops or decision rules are present. Specific operations depending on the type of data can be explained by a flowchart. Data Flow Diagram can be represented in several ways. The DFD belongs to structured-analysis modeling tools. (Flow)

THE SYMBOLS DEPICT OF DATA FLOW DIAGRAMS:

the System:

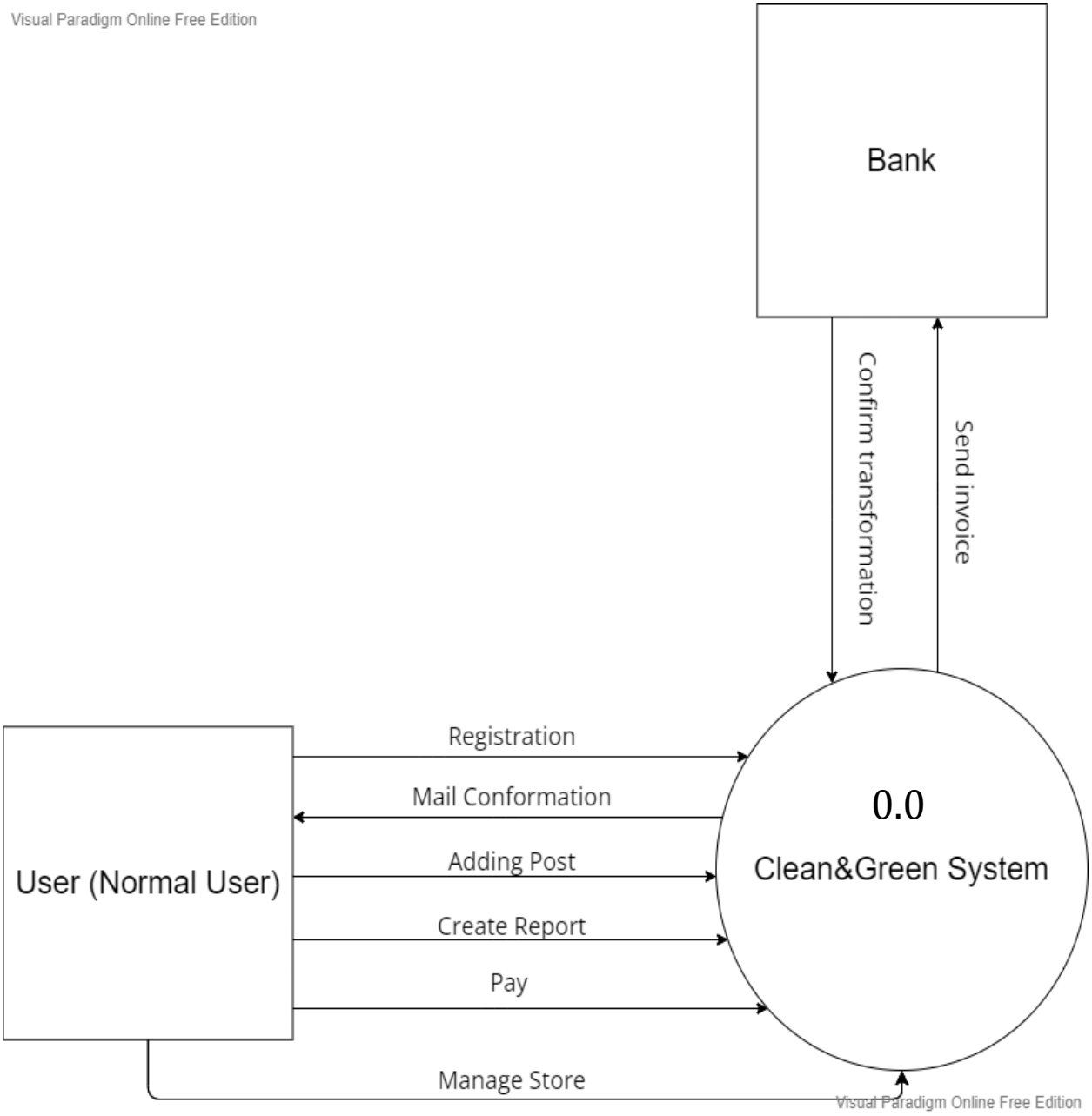


the stakeholder:

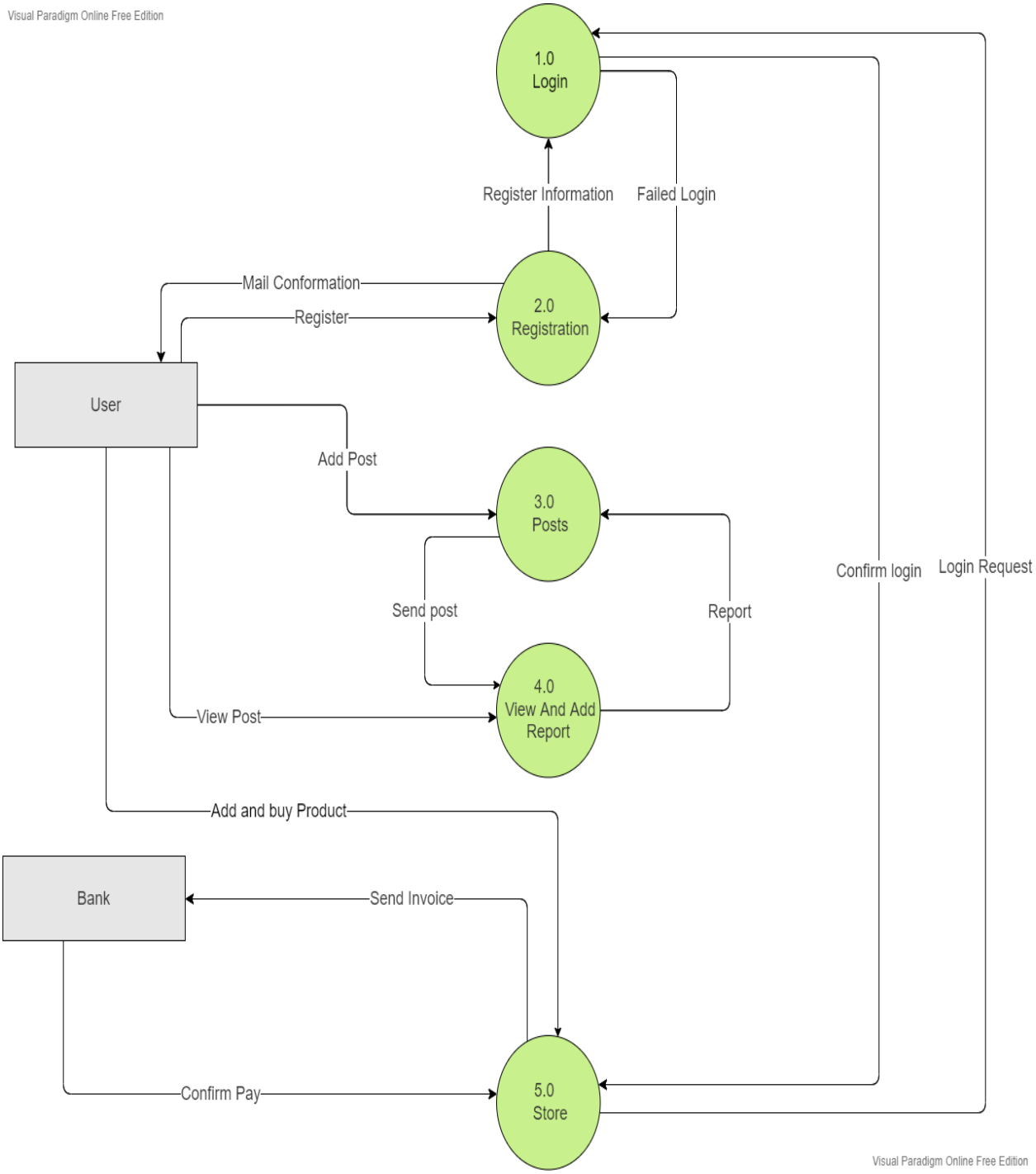


The database:

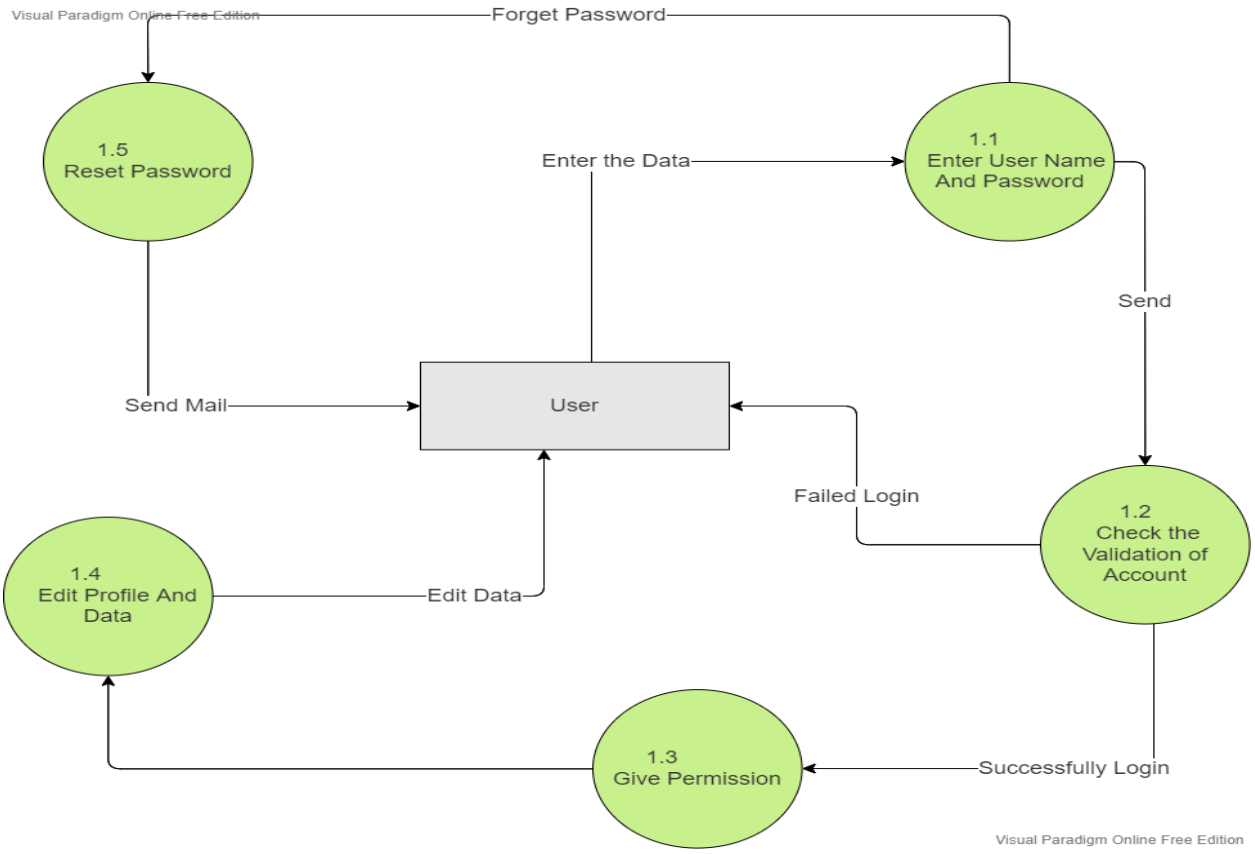
CONTEXT DIAGRAM:

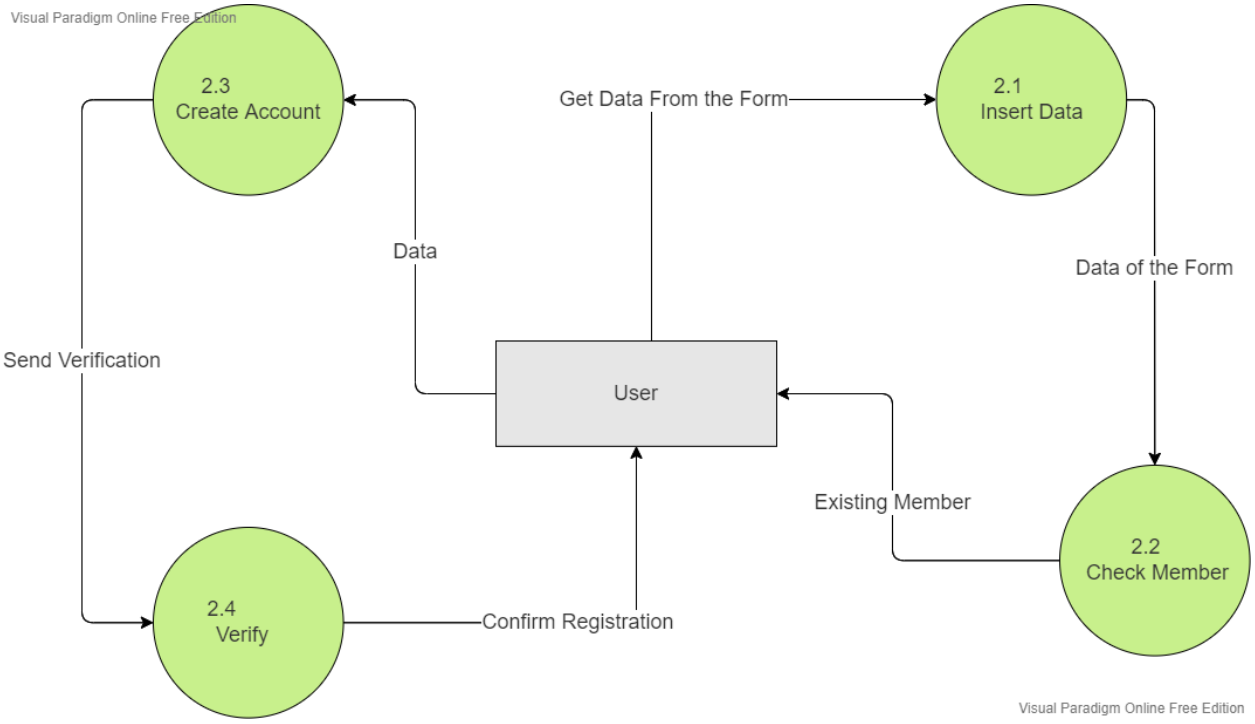


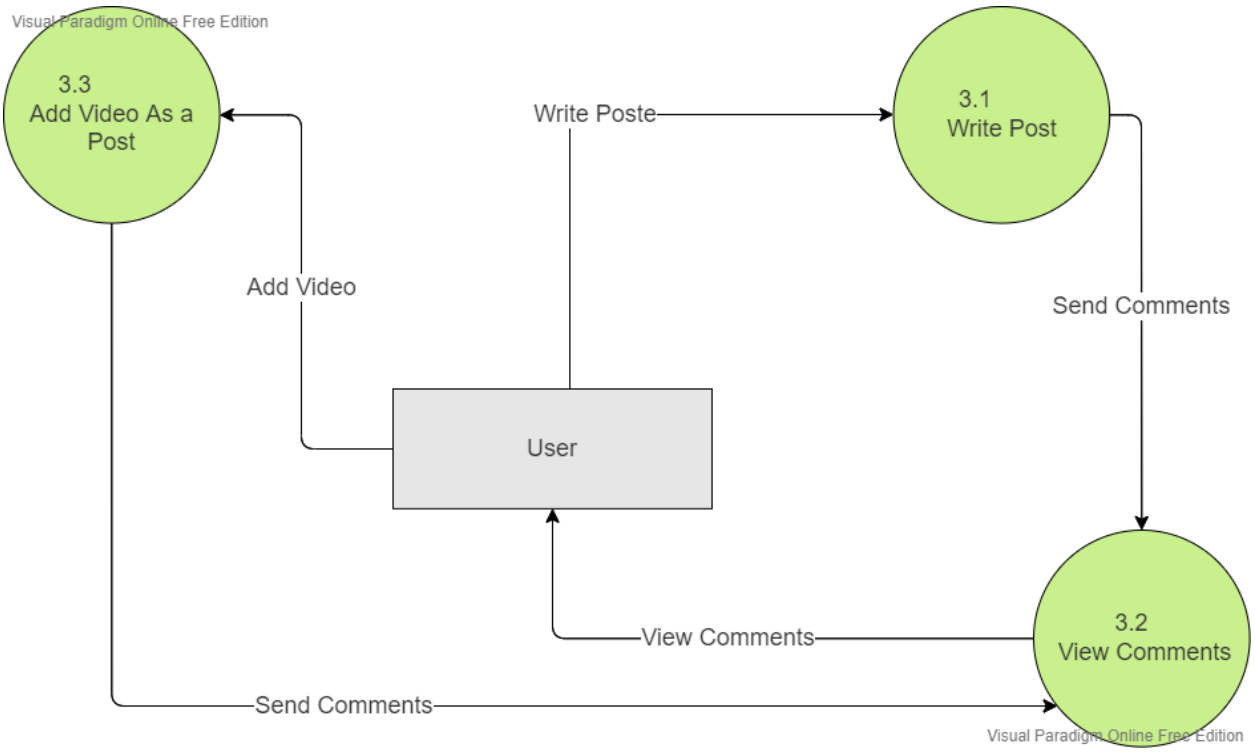
LEVEL ZERO DFD

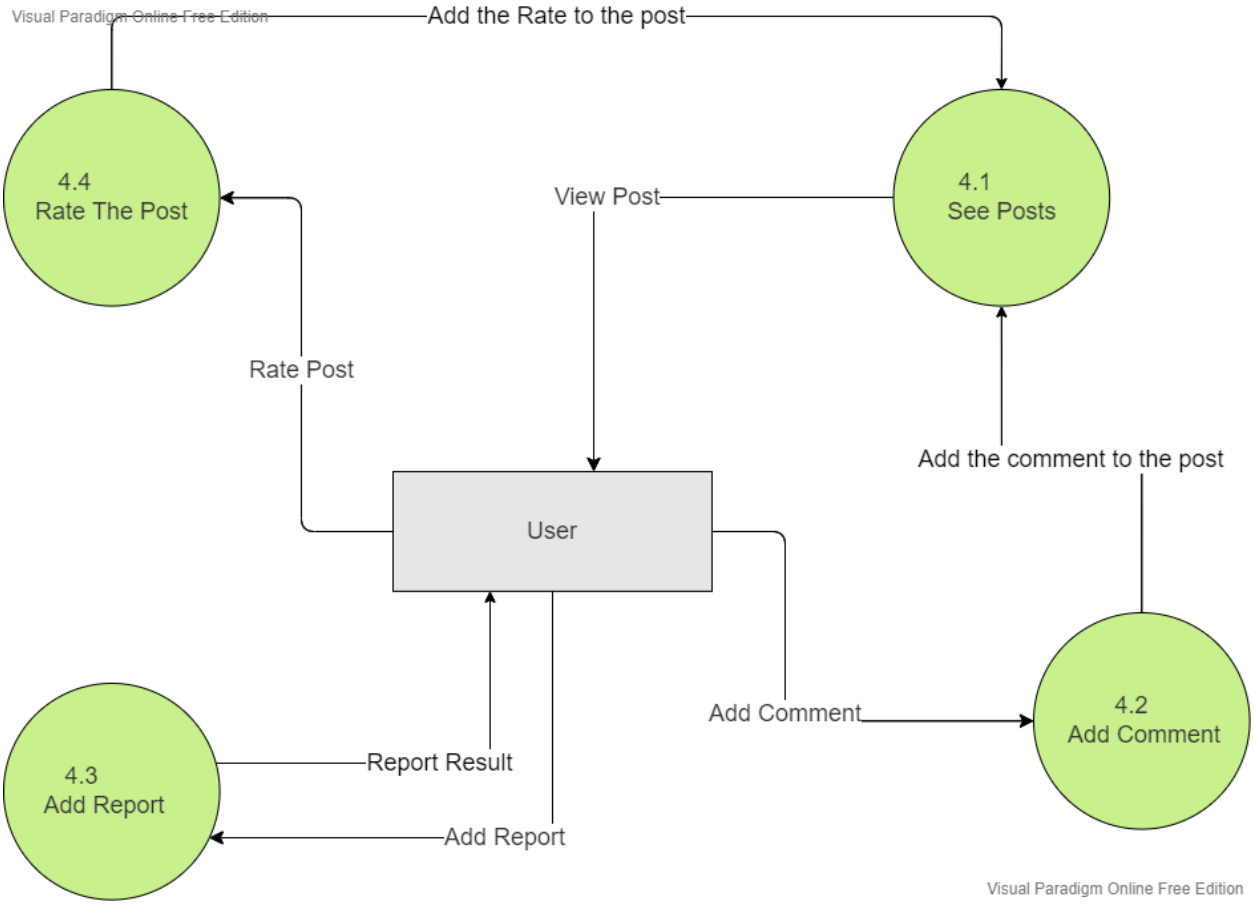


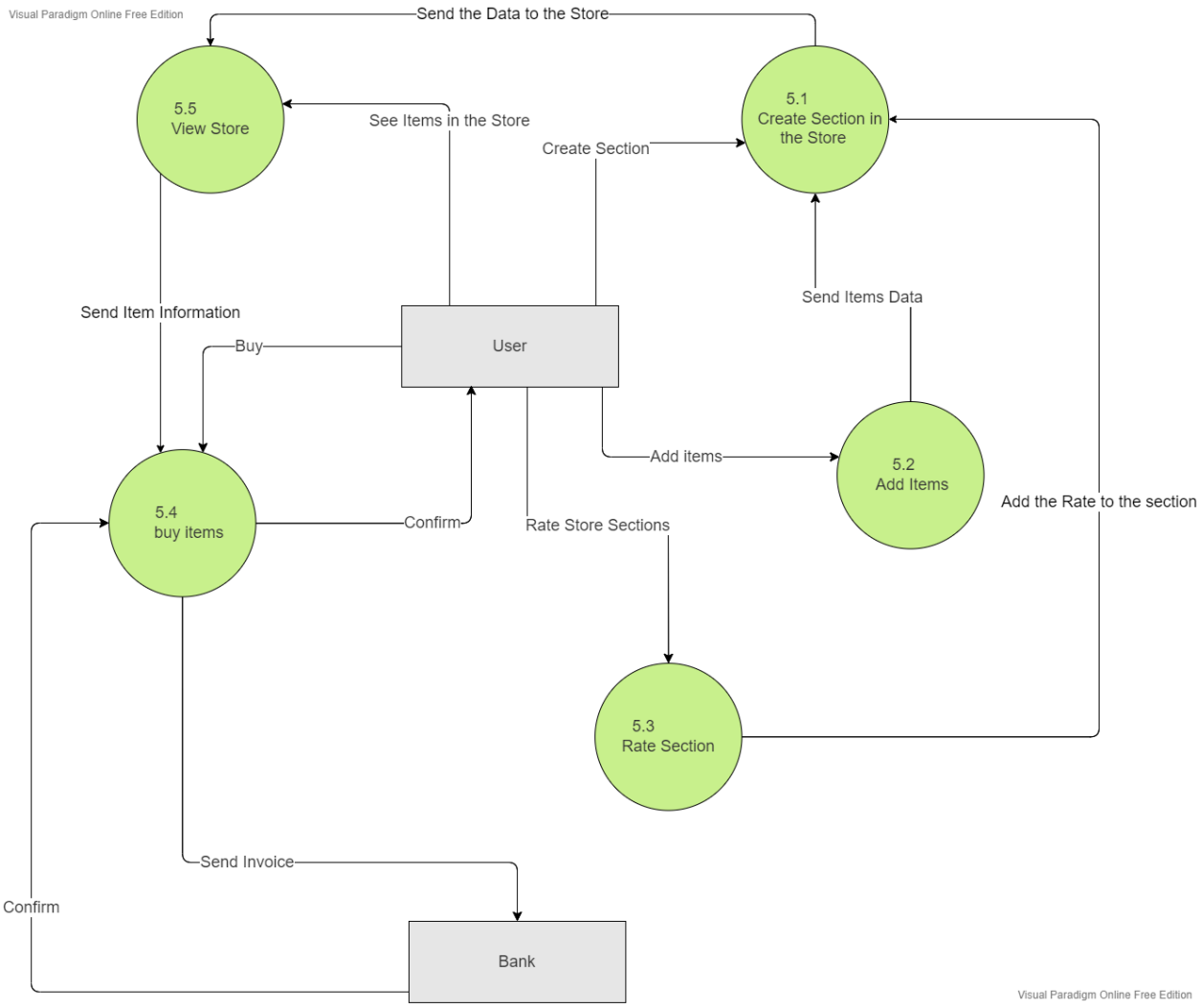
LEVEL ONE DFD



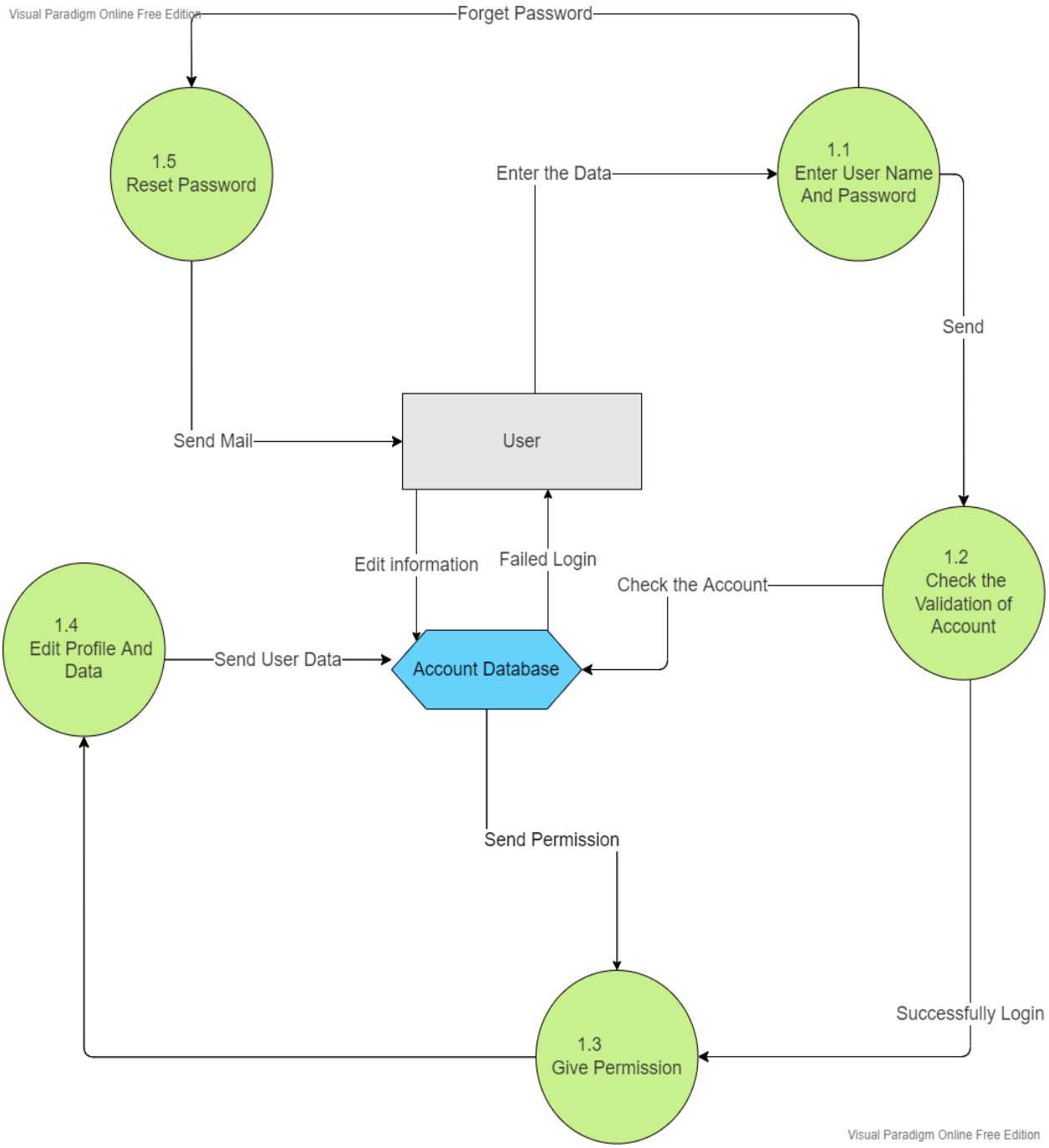


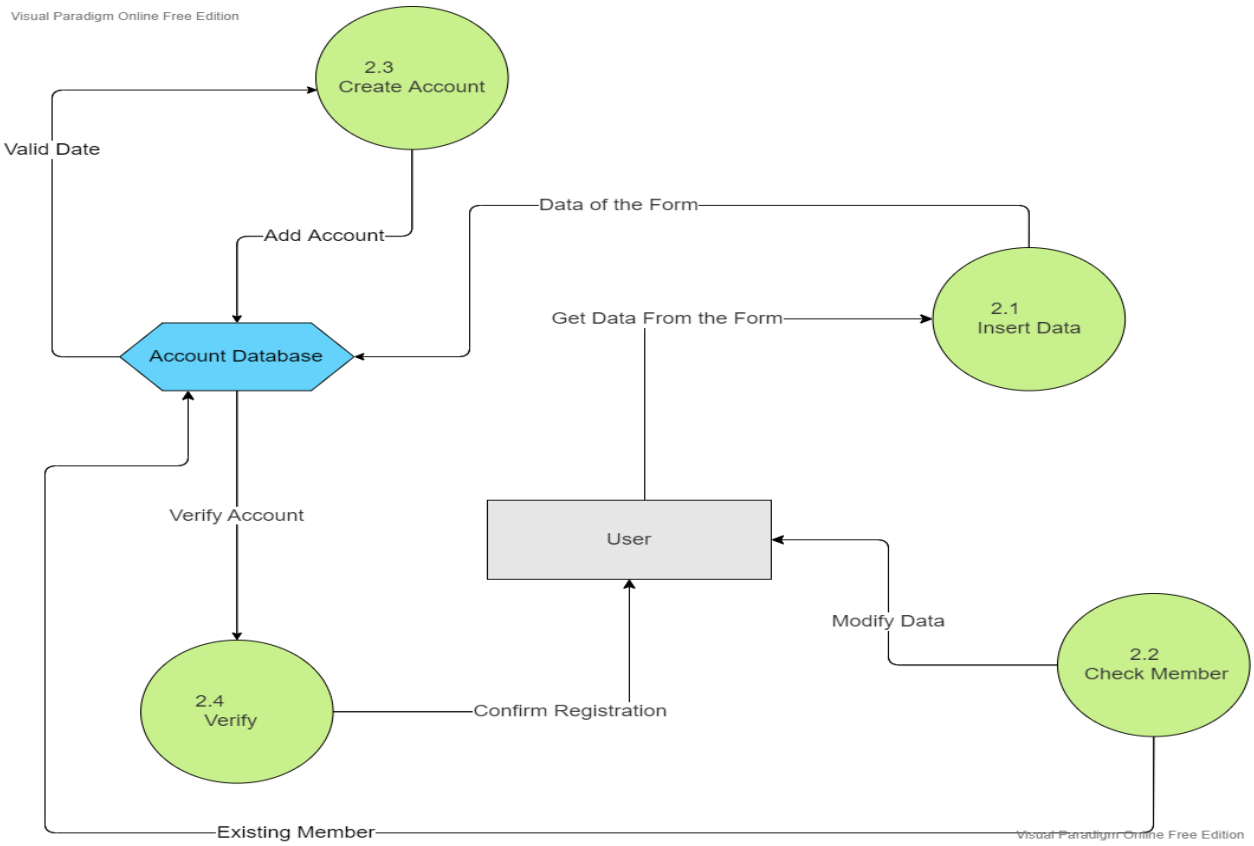

Visual Paradigm Online Free Edition
3.3
Add Video As a Post
Write Poste
3.1
Write Post
Add Video
Send Comments
User
View Comments
3.2
View Comments
Send Comments
Visual Paradigm Online Free Edition

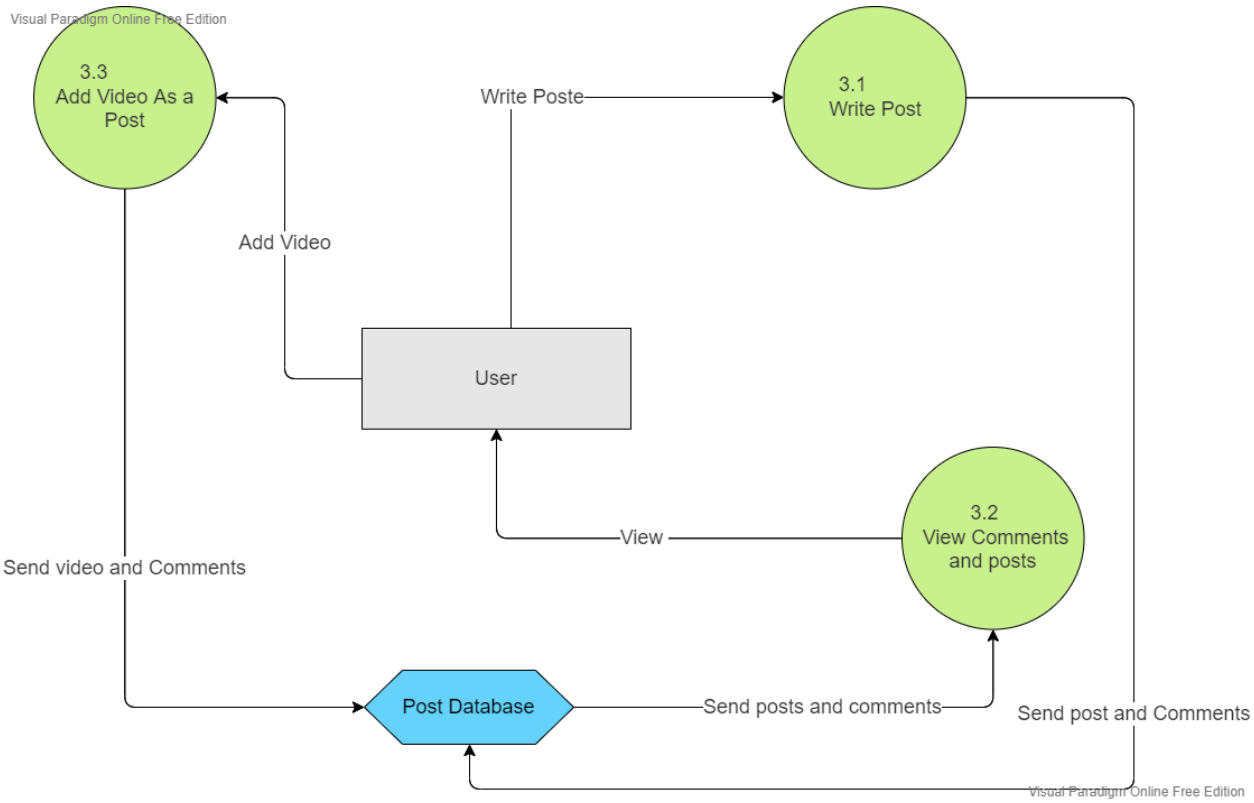

Visual Paradigm Online Free Edition
Add the Rate to the post
4.4
Rate The Post
View Post
4.1
See Posts
Rate Post
Add the comment to the post
User
Add Comment
4.2
Add Comment
4.3
Add Report
Report Result
Add Report
Visual Paradigm Online Free Edition

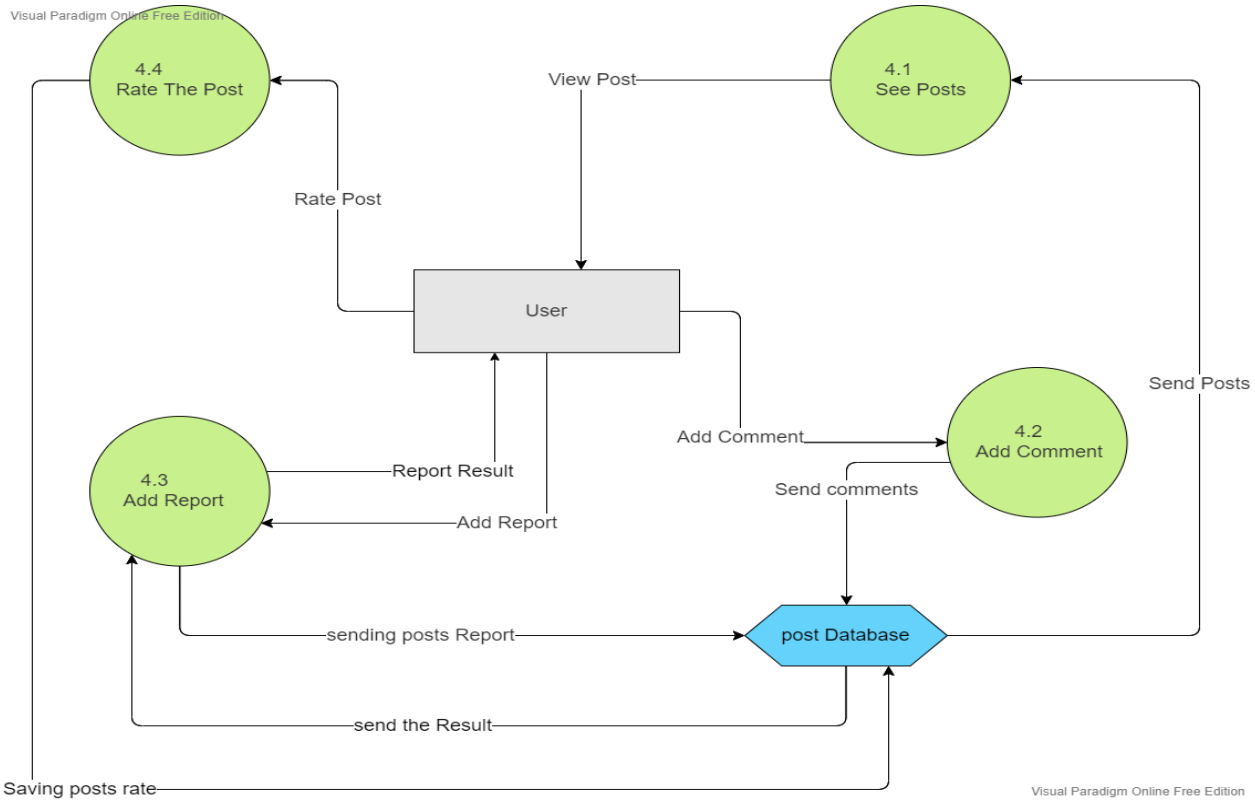

Visual Paradigm Online Free Edition
Send the Data to the Store
5.5
View Store
See Items in the Store
5.1
Create Section in
the Store
Create Section
Send Items Data
Send Item Information
Buy
User
Add items
5.2
Add Items
Add the Rate to the section
5.4
buy items
Confirm
Rate Store Sections
5.3
Rate Section
Send Invoice
Confirm
Bank
Visual Paradigm Online Free Edition

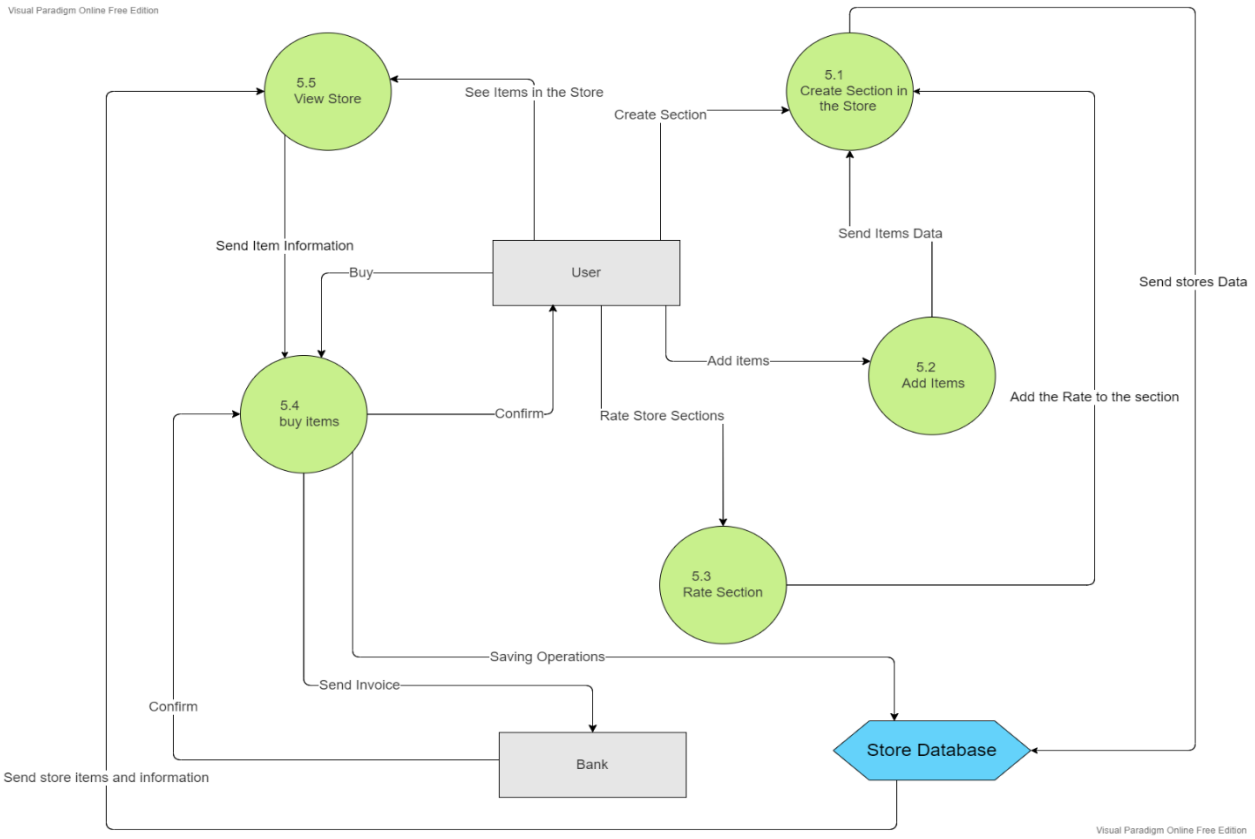
LEVEL TWO DFD




Visual Paradigm Online Free Edition
2.3
Create Account
Valid Date
Add Account
Data of the Form
2.1
Insert Data
Get Data From the Form
Account Database
Verify Account
User
Modify Data
2.2
Check Member
2.4
Verify
Confirm Registration
Existing Member
Visual Paradigm Online Free Edition


Visual Paradigm Online Free Edition
3.3
Add Video As a
Post
Write Poste
3.1
Write Post
Add Video
User
3.2
View Comments
and posts
View
Send video and Comments
Post Database
Send posts and comments
Send post and Comments
Visual Paradigm Online Free Edition


Visual Paradigm Online Free Edition
4.4
Rate The Post
4.1
See Posts
View Post
Rate Post
User
Send Posts
Add Comment
4.2
Add Comment
4.3
Add Report
Report Result
Add Report
Send comments
sending posts Report
post Database
send the Result
Saving posts rate
Visual Paradigm Online Free Edition


Visual Paradigm Online Free Edition
5.5
View Store
See Items in the Store
5.1
Create Section in the Store
Create Section
Send Item Information
Buy
User
Send Items Data
Send stores Data
Add items
5.2
Add Items
5.4
buy items
Confirm
Rate Store Sections
Add the Rate to the section
5.3
Rate Section
Saving Operations
Send Invoice
Confirm
Bank
Store Database
Send store items and information
Visual Paradigm Online Free Edition

CONCLUSION

After studying the project according to the multiple stages that were previously presented, the process of understanding and implementing the program became easier and clearer. In conclusion, this is the best solution to solve the problem of waste accumulation.

ENDING POINT

The present study is subject to limitations that should be addressed in future research. First, the study resources are limited to a few free ones, that more resources and studies must be included to achieve more information. Second, the research is missing true interviews and questionaries that would improve the requirements. Third, the system is missing an implementation and testing phases that will examine the design of the system.

REFERENCES

Abrition. (n.d.). *Abrition*. Retrieved from https://abrition.com/5-important-things-to-consider-while-developing-a-mobile-app/

ACTION FOR RENEWABLES. (n.d.). Retrieved from https://www.actionforrenewables.org/how-to-counter-waste-disposal-challenges/

attributed, w. f. (n.d.). *wikipedia fully attributed*. Retrieved from https://en.wikipedia.org/wiki/Data_modeling#:~:text=A%20fully%20attributed%20data%20model%20contains%20detailed%20attributes,base%20data%20structures%20used%20to%20store%20the%20data.

base, w. k. (n.d.). *wikipedia key base*. Retrieved from https://en.wikipedia.org/wiki/Data_model

Change, C. (n.d.). Retrieved from https://nouhadawwad.wordpress.com/2018/02/10/waste-crisis-in-lebanon/

EARTHDAY.ORG. (n.d.). Retrieved from https://www.earthday.org/how-our-trash-impacts-the-environment/

Flow, D. (n.d.). *Data Flow.* Retrieved from https://www.geeksforgeeks.org/what-is-dfddata-flow-diagram/

GLOBAL, I. (n.d.). *IGI GLOBAL*. Retrieved from https://www.igi-global.com/dictionary/use-cases-uml/37227#:~:text=A%20use%20case%20narrative%20is%20a%20largely%20text-based,developers%20of%20systems%20and%20the%20intended%20use%20rs.

IBM. (n.d.). *IBM*. Retrieved from IBM: https://www.ibm.com/cloud/learn/data-modeling

lucidchart. (n.d.). *lucidchart*. Retrieved from https://www.lucidchart.com/pages/er-diagrams

Mullan, E. (n.d.). *Hurree*. Retrieved from https://blog.hurree.co/blog/13-things-you-need-to-include-in-your-mobile-app-marketing-strategy

Newsletter. (n.d.). *Newsletter*. Retrieved from https://tdan.com/normalizing-with-entity-relationship-diagramming/4583#:~:text=First%20Normal%20Form%20%281NF%29%20The%20first%20normal%20form,of%20first%20normal%20form%20for%20the%20entity%20type.

Paraim, V. (n.d.). *Visual Paraim*. Retrieved from https://www.visual-paradigm.com/support/documents/vpuserguide/94/2575/84257_usecasediagr.html#uml-use-case

Wikipedia. (n.d.). *Use Case*. Retrieved from https://en.wikipedia.org/wiki/Use_case_diagram

wikipedia. (n.d.). *wikipedia*. Retrieved from https://en.wikipedia.org/wiki/Context_model

Wikipedia. (n.d.). *Wikipedia*. Retrieved from https://en.wikipedia.org/wiki/Functional_requirement

Wikipedia. (n.d.). *Wikipedia*. Retrieved from https://en.wikipedia.org/wiki/Non-functional_requirement

Wikipedia. (n.d.). *Wikipedia*. Retrieved from https://en.wikipedia.org/wiki/Design

Wikipedia. (n.d.). *Wikipedia*. Retrieved from https://en.wikipedia.org/wiki/Gantt_chart