

# Concordia University



## SOEN-6481 SOFTWARE SYSTEMS REQUIREMENTS SPECIFICATION(FALL 2019)

---

# TICKET VENDING MACHINE

---

## DELIVERABLE 1 (D1)

---

### Submitted By: (Team E)

Bhavpreet Kaur (40071697)  
Navjot Kaur (40078155)  
Mehrnaz Keshmirpour (40063320)  
Shruthi Kondapura Venkataiah (40091427)  
Sanchit Kumar (40081187 )

### Submitted To:

Prof. Pankaj Kamthan

GitHub - <https://github.com/m3hrn4z/SRS>  
October 15, 2019

# Contents

1.1	Problem 1 . . . . .	2
1.1.1	iGo Description . . . . .	2
1.1.2	Scope of Use . . . . .	2
1.2	Problem 2 . . . . .	3
1.2.1	User Requirements Elicitation . . . . .	3
1.2.2	Mind Map for TVM . . . . .	3
1.2.3	Context of Use Model: User . . . . .	5
1.2.4	Context of Use Model: Admin . . . . .	7
1.3	Problem 3 . . . . .	9
1.3.1	Domain Model . . . . .	9
1.4	Problem 4 . . . . .	12
1.4.1	Use Case Model . . . . .	12
1.4.2	Use Case: Change Language . . . . .	13
1.4.3	Use Case: View Ticket Plans . . . . .	13
1.4.4	Use Case: Buy Tickets . . . . .	15
1.4.5	Use Case: Make a payment . . . . .	19
1.4.6	Use Case: Reload Rechargeable Card . . . . .	22
1.4.7	Use Case: Update Ticket Specification . . . . .	26
1.4.8	Use Case: Customer Registration . . . . .	31
1.4.9	Use Case: False Login . . . . .	34
1.4.10	Use Case: Password Reset . . . . .	35
1.4.11	Use Case: Payment Fraud . . . . .	37
	Glossary . . . . .	37
	Acronyms . . . . .	38
	Appendices . . . . .	38
A	Appendix . . . . .	38
A.1	Interviews . . . . .	38

# Deliverable - 1

## 1.1 Problem 1

### 1.1.1 iGo Description

The product iGo is a software solution for TVM (Ticket Vending Machine) which allows user to purchase tickets, recharge their OPUS card and get information about the different tickets Société de transport de Montréal (STM)(Société de transport de Montréal)[5] offers. iGO products is designed to server people in Montreal, Quebec, Canada who use STM metros and buses. STM offers two ticketing methods: Rechargeable Card OPUS Card and Non-Rechargeable Card limited use cards. With OPUS card user can make unlimited trips in STM buses and metros for a particular period, which is the recharge cycle of the card.

User who wish recharge their OPUS card are first required to place their card in the card reader slot in TVM, they can then select the type of purchase they want to make and can select the mode of payment they wish to use. Currently, iGo supports cash and card payments. Users who want to purchase a Non-rechargeable can select from a list of options based on the type of trip they wish to make. The purchase will be considered successfull only after the payment has been authenticated(in case of card payments) and is successfull. After a sucessfull purchase the user have the option to select the mode through which they wish to receive receipt. iGo supports 3 type of receipt delivery: Paper, email and SMS receipts.

iGo will also internally generate a Transaction when the user interacts with the system. The transaction is recorded irrespective of whether the purchase is successfull or not. These transactions helps track the purchase in case of any failure in the system. At bottom bar of the TVM display, a helpline number is displayed at all times in case user needs any kind of assistance.

### 1.1.2 Scope of Use

The product iGo at this moment is restricted to server in Montreal city only. iGo is a software system and does not include the development, maintenance or deployment of the physical devices needed to develop a TVM. iGo does not provide any online platform or device application to recharge OPUS card or purchase ticket.

## **1.2 Problem 2**

### **1.2.1 User Requirements Elicitation**

We interviewed different types of TVM users to elicit contextual information and user requirements directly from TVM users by asking questions and documenting the responses as audio files and text. The results of interviews are given in Appendix A. The information gathered from these interviews was used for constructing user models, context of use[3], domain model[2] and use case diagrams[4].

### **1.2.2 Mind Map for TVM**

The mind map[1] in the figure 1.1, is for the analysis of Metro TVM design. All the concepts which are required for the development have been mapped. The possible type of TVM users will give a high level idea of how the TVM should look like, and how can a TVM software designed to be operated. The Analysis is for the elicitation of requirements, which includes analysis of problem, action items, goals to achieve and solutions to the problems. The Design includes all level of designs during the TVM software development. Application of the TVM software designed will be for Metro transportation, requirement specific information should be acquired.

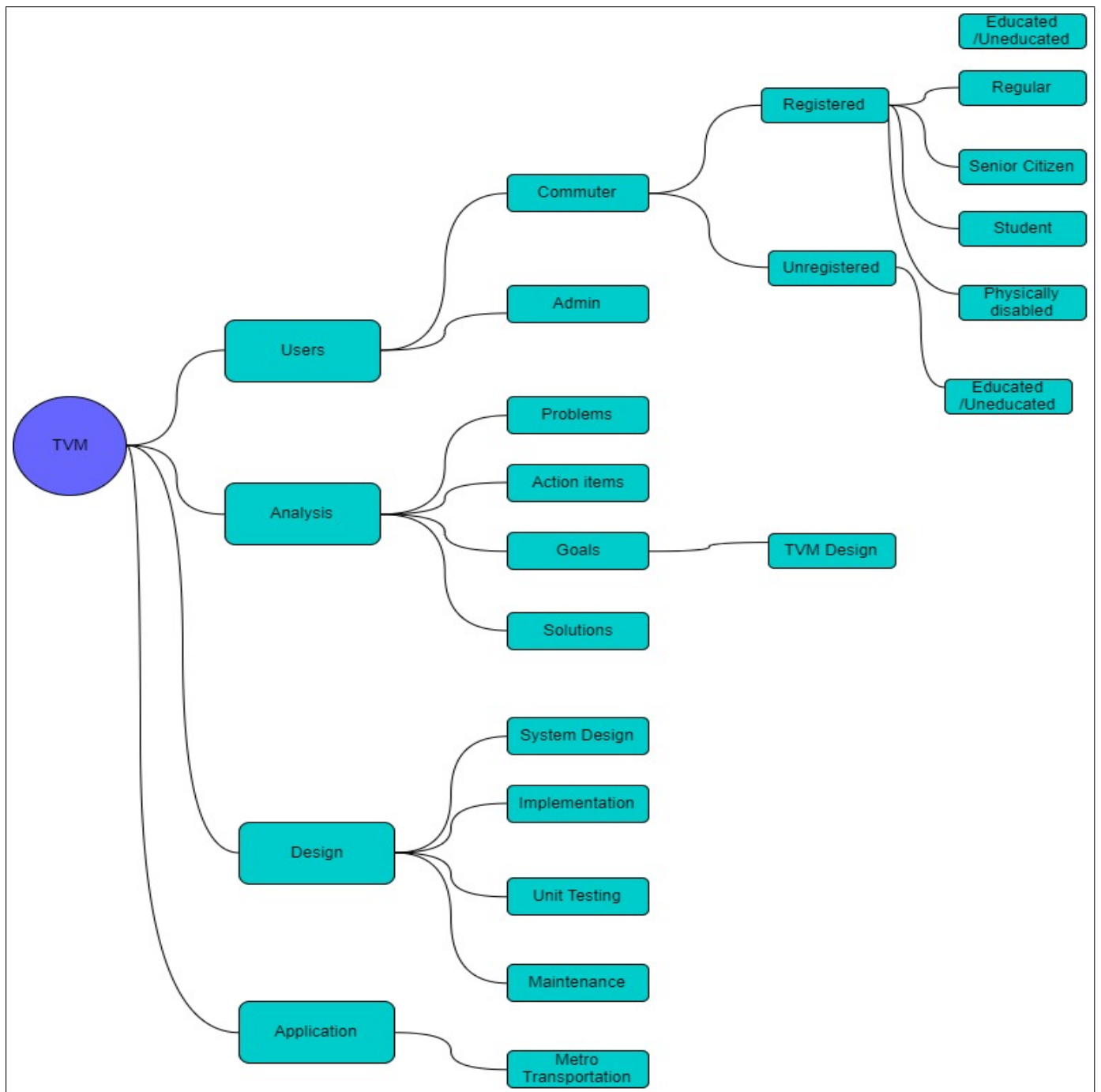


Figure 1.1: Mind Map

### 1.2.3 Context of Use Model: User

Factor	Description
User	
Role	User is the customer/passenger to purchase tickets
Experience	The experience depends on the how frequent the user is using the TVM, System is designed keeping in mind of a new user without any system knowledge
Education	System is designed assuming that customer can atleast read English. <b>constraint:</b> We wont be able to design a model for who cant read
Physical Characteristics	The physical TVM should be designed in such a way that even the physically challenged customer with wheel chair should also access the TVM.
Cognitive Characteristics	The cognitive knowledge of a user should be counted as minimal, the system is designed keeping in mind of new user
Location	The location of user must be inside Montreal city.
Age	The system should offer Reduced Fare according to the user's age.
Profession	The system should offer reduced fares for students.
Special Variant Preferences in UI	For special need people with visual impairment system should offer voice messages.
Task	
Choice in system use	Number of tasks in the system should be divided individually, so that the user can choose the options such as purchasing for a single trip, buying a monthly pass etc
Complexity	Since the users can be with minimal knowledge on the TVM, tasks should be simple and straight forward
Demands	Available tasks should be able to satisfy user requirements with context of use. For example system should not issue more tickets that the number of passengers

Workflow Con- trollability	The workflow of each task in TVM machine should be easily controllable by the user. For example user should be able to navigate back to the main menu, user should be able to cancel the ticket selection or transaction etc.
Safety	The task payment should be made safer, in such a way that the card payment authentication is kept confidential. Even the user information if stored should be maintained confidential.
Criticality	Task criticality should be divided based on the how important the task is and how often it is used/reused.
Frequency	The TVM can be used any number of times by any number of users, so the TVM should be able to guide each user based on the current information available with respect to trains or buses.
Temporal Char- acteristics	According to user requests that are dependant to date, the system should validate the request based on current date.
<b>Environment</b>	
Physical Envi- ronment	The TVM will be placed in metro stations, the passengers will be able to buy ticket/passes before there journey in train/bus. The factors like lights on the TVM machine for the visibility, space available to access the TVM, is the place allocated is too noisy should be taken care.
Technical Envi- ronment	The TVM will be connected to the data server where the user informations can be retrieved, it will be connected to the bank server to authenticate the credit/debit cards.
Social Environ- ment	The TVM must support at least two languages including English and French since the official language of Quebec is French.

### 1.2.4 Context of Use Model: Admin

Factor	Description
<b>Admin</b>	
Role	ADMIN <b>constraint:</b> Due to the data privacy, The admin of TVM real activities were not available
Experience	The experience depends on the how frequent the admin is using the TVM, System is designed keeping in mind of a new admin without any system knowledge
Education	System is designed assuming that admin can atleast read English <b>constraint:</b> We wont be able to design a model for who cant read
Physical Char-acteristics	The physical TVM should be designed in such a way that even the physically challenged admin with wheel chair should also access the TVM
Cognitive Char-acteristics	The cognitive knowledge of a admin should be counted as minimal, the system is designed keeping in mind of a new admin
Location	The location of admin must be inside Montreal city
Age	Ranging from 25-Retirement age
Profession	The Profession is Role of the admin
Special Variant Preferences in UI	For special need people with visual impairment system should offer voice messages
<b>Task</b>	
Choice in system use	Admin should be able to do all administrative activities such changing changing the ticket/monthly pass fares,changing discount offers. Or any User specific actions
Complexity	Since the admins can be with minimal knowledge on the TVM, tasks should be simple and straight forward
Demands	Should provide the TVM features in such a way, it should be easy for an admin change any relevant activity



Workflow Con- trollability	The workflow of each task in TVM machine should be easily controllable by the admin. For example admin should be able to navigate back to the main menu, admin should be able to any discount option, any membership or any user sepefic options
Safety	It should be confidential about the admin features. It should be isolated from the normal user login
Criticality	Task criticality should be divided based on the how important the task is and how often it is used/reused.
Frequency	The TVM can be used any number of times by any number of users, so the TVM should be able to guide each user based on the current information available with respect to trains or buses.
Temporal Char- acteristics	According to admin requests that are dependant to date, the system should validate the request based on current date
<b>Environment</b>	
Physical Envi- ronment	The TVM will be placed in metro stations, the passengers will be able to buy ticket/passes before there journey in train/bus. The factors like lights on the TVM machine for the visibility, space available to access the TVM, is the place allocated is too noisy should be taken care
Technical Envi- ronment	The TVM will be connected to the data server where the admin informations can be retrieved, it will be connected to the bank server to authenticate the credit/debit cards.
Social Environ- ment	The TVM must support at least two languages including English and French since the official language of Quebec is French

## 1.3 Problem 3

### 1.3.1 Domain Model

Concepts	Description
TVM	It is a ticket vending machine to issue ticket or reload rechargeable card for bus and metro
User	Someone who uses TVM as a commuter, an administrator or a negative user as a crook
Administrator	Someone who has privilege to manage ticket specification
Crook	A negative user how tries to misuse the TVM like a hacker who is trying to steal bank car information
Commuter	Someone who comes to TVM and wants to buy ticket or check different fares to travel
Registered Com-muter	Someone who has registered his/her information in the system and uses their Rechargeable Card to buy ticket
Unregistered Commuter	Someone who has not registered his/her information in the system and uses the system anonymously and buy Non-Rechargeable Ticket
Student	Someone who has the proof of being a student and is eligible to use reduced fare
Regular	Someone who has registered in the system and use Rechargeable Card, but pay Regular Fare
Senior Citizen	Someone who is 65 years old or over and is eligible to use reduced fare
Language	The display language of User Interface that can be either English or French
Printer	Printer is the device that prints the receipt or provides the non-rechargeable ticket
Receipt	It has the confirmation of the transaction and this is the superclass of different kinds of receipt

Paper Receipt	It is a subclass of receipt which is printed on paper
Email	It is subclass of receipt electronically delivered to the user
Ticket	It is either a rechargeable or non-rechargeable ticket which users pay for to use metro or bus
Ticket Type	It describes number of trips or the duration (e.g. monthly ticket) of the ticket
Non-Rechargeable Ticket	It is a temporary ticket for limited use printed on the paper by printer
Rechargeable Card	A card which has an expiry date and can be recharged and used multiple times before the expiration date
Fare	It includes the price of different plans to buy the ticket
Regular	It is regular price of a specific plan
Reduced	It is reduced price of a specific plan for special customers such as students or people with certain ages
Payment	It is superclass for different modes of payment
Bank Card	It is a payment method that uses bank card to pay for the ticket fare
Cash	It is a payment method that uses cash to pay for the ticket fare
Transaction	It debits customer bank card
Payment Authentication	It takes the details of the user's bank card and verify the payment with the bank server
Bank Server	It confirms user's bank card information

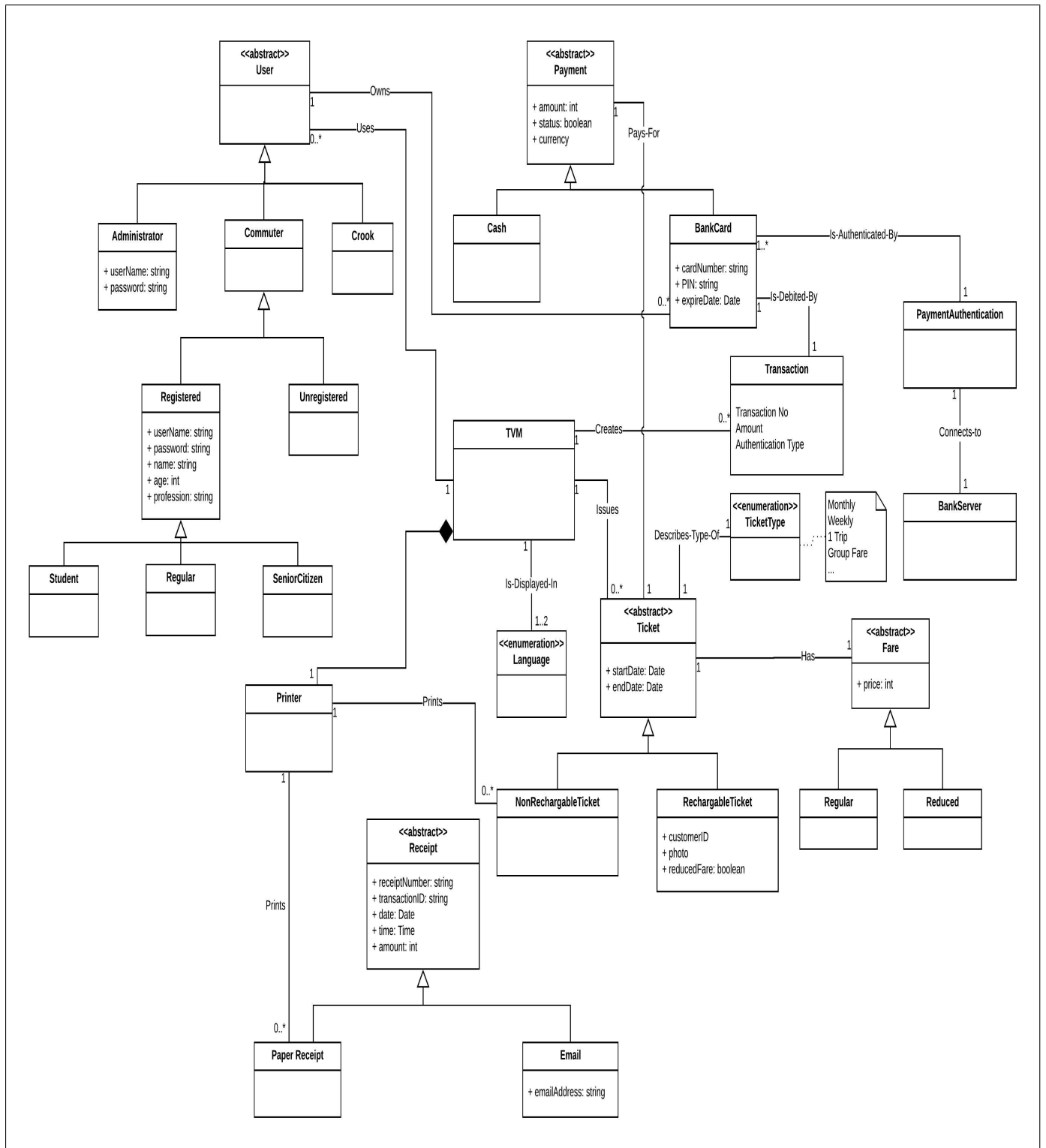


Figure 1.2: Domain Model

## 1.4 Problem 4

### 1.4.1 Use Case Model

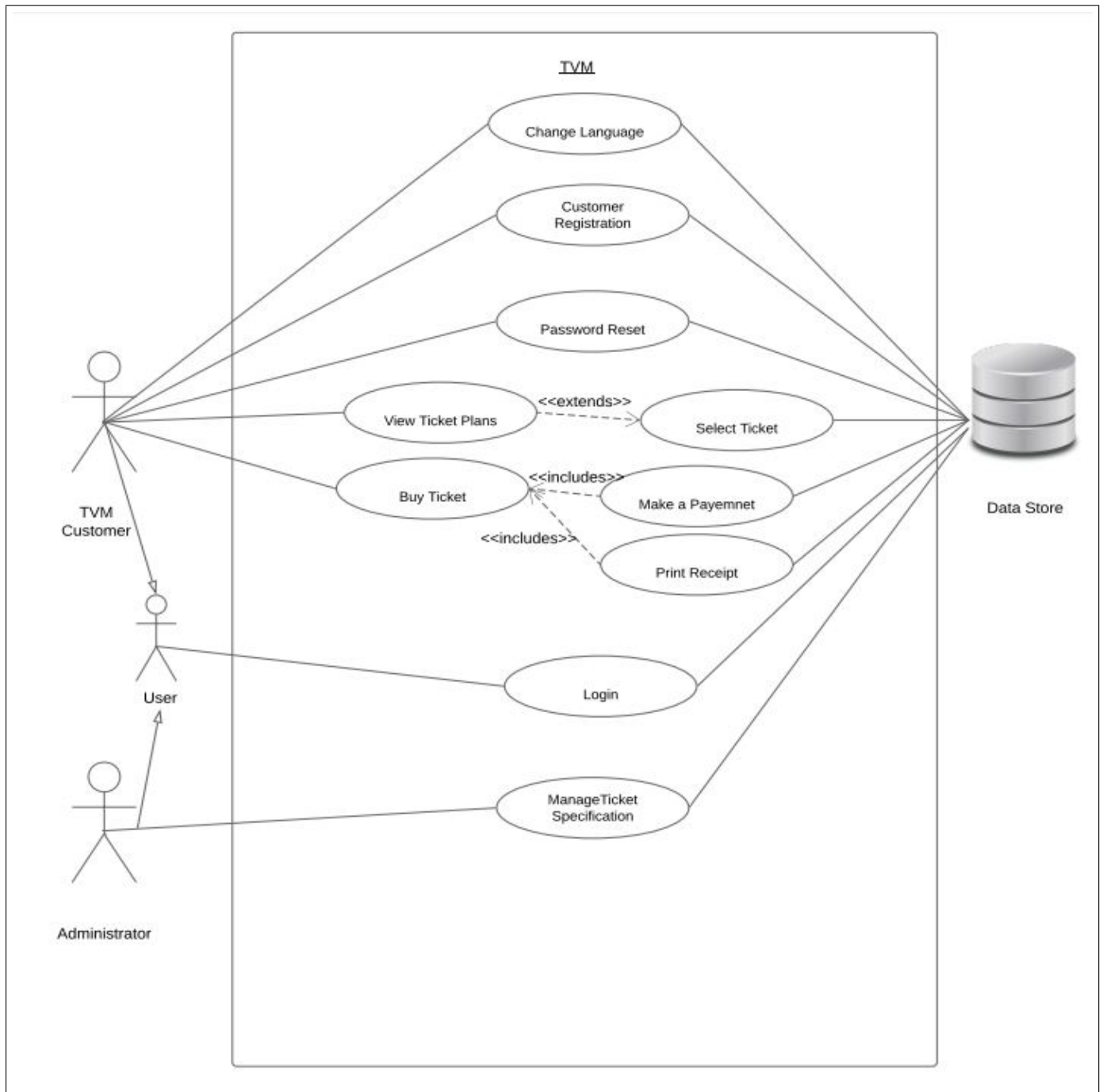


Figure 1.3: Use Case Model

### 1.4.2 Use Case: Change Language

<b>Number</b>	1	
<b>Name</b>	Change Language	
<b>Summary</b>	User wants to change to desired language	
<b>Priority</b>	4	
<b>Preconditions</b>	N/A	
<b>Postconditions</b>	User is able to change to desired language	
<b>Primary Actor(s)</b>	TVM Customer	
<b>Secondary Actors(s)</b>	TVM and Administrator	
<b>Trigger</b>	User has chosen to change language	
<b>Main Success Scenarios</b>	<b>Step</b>	<b>Action</b>
	1	User can select the desired language
	2	The application is displayed in desired language
<b>Extensions</b>	N/A	
<b>Open Issues</b>	1	Language available to be decided by the customer

### 1.4.3 Use Case: View Ticket Plans

Commuters use Ticket vending machine to view ticket plans. Commuter is the primary actor that initiate the use case by choosing to view ticket plans. Secondary actor includes Data Store that supports and responds to the requests. The result for a normal use case scenario is that the customer has viewed different ticket plans. Figure 1.4 explain this use case.

<b>Number</b>	2	
<b>Name</b>	View Ticket Plans	
<b>Summary</b>	User has viewed different ticket plans	
<b>Priority</b>	1	
<b>Preconditions</b>	N/A	
<b>Postconditions</b>	User has viewed different ticket plans	
<b>Primary Actor(s)</b>	TVM Customer	
<b>Secondary Actors(s)</b>	Data Store	
<b>Trigger</b>	User has chosen to view ticket plans	
<b>Main Success Scenarios</b>	<b>Step</b>	<b>Action</b>
	1	User selects the language
	2	System asks the ticket type
	3	User selects the ticket type
	2	System displays ticket plans
<b>Extensions</b>	N/A	
<b>Open Issues</b>		NA

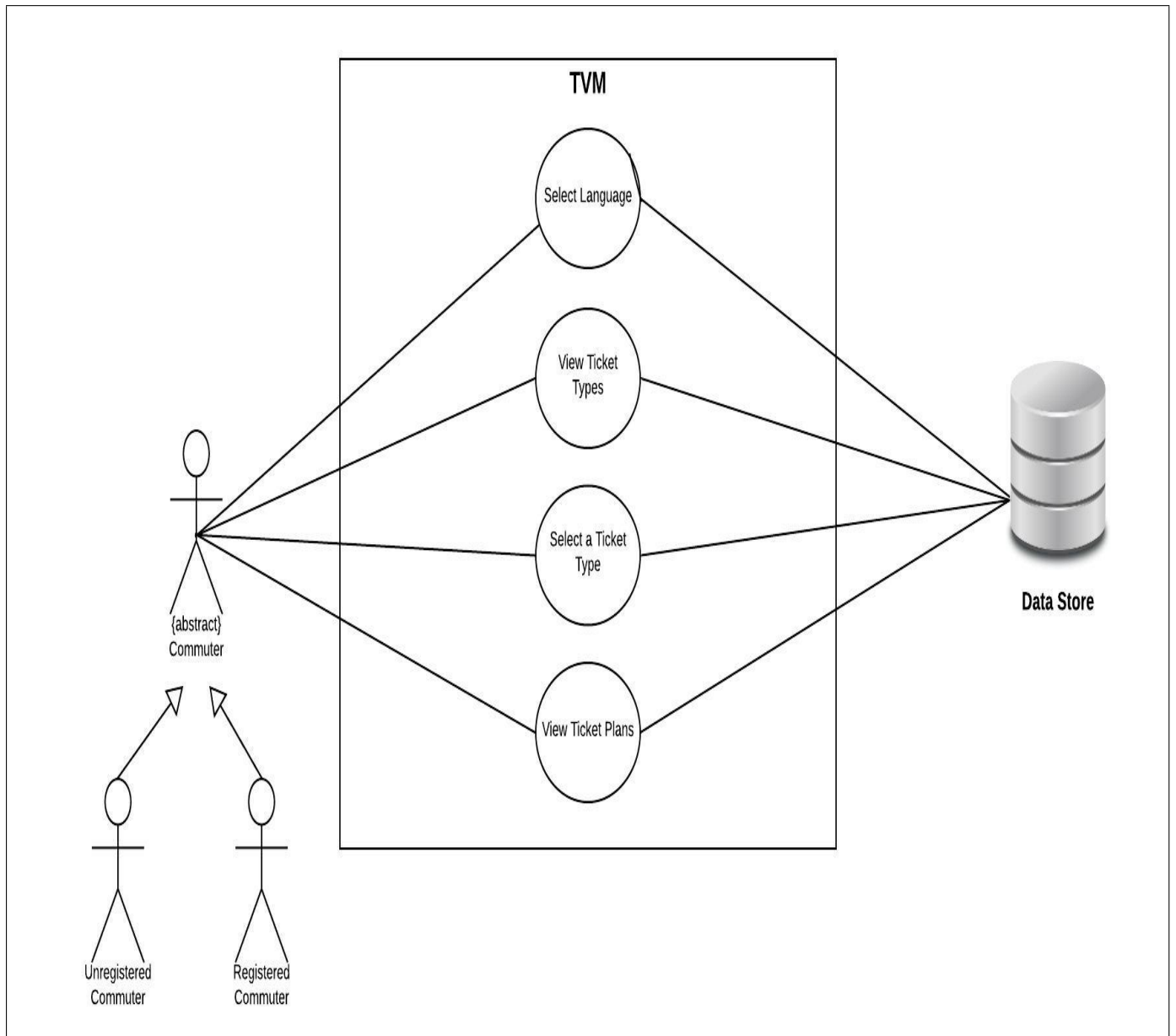


Figure 1.4: Use Case: View Ticket Plans

#### 1.4.4 Use Case: Buy Tickets

Commuters use Ticket vending machine to buy tickets. Commuter is the primary actor that initiate the use case by choosing to buy a ticket. Secondary actors include Payment Authentication and Data Store that support and respond to the requests. The result for a normal use case scenario is that the customer has received the ticket and the receipt. Figure 1.5 and 1.6 and 1.7 explain this use case.



<b>Number</b>	3	
<b>Name</b>	Buy Ticket	
<b>Summary</b>	Customer buy a ticket using TVM	
<b>Priority</b>	1	
<b>Preconditions</b>	Customer has access to a valid payment method	
<b>Postconditions</b>	Customer has received the ticket and the receipt	
<b>Primary Actor(s)</b>	TVM Customer	
<b>Secondary Actors(s)</b>	Payment Authentication, Bank Server, Data Store	
<b>Trigger</b>	Customer has chosen to buy a ticket	
<b>Main Success Scenarios</b>	<b>Step</b>	<b>Action</b>
	1	Customer selects the language
	2	System displays ticket plans
	3	Customer chooses a ticket plan
	4	System asks for payment method
	5	Customer selects a payment method
	6	Customer makes a payment
	7	System asks for type of receipt
	8	Customer selects the type of receipt
	9	System prints receipt and ticket
	10	Customer removes receipt and ticket
	11	System displays welcome message
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	6a	The payment is not successful
	6b	System displays message that payment is not successful
	6c	System exits and displays welcome message
<b>Open Issues</b>		NA



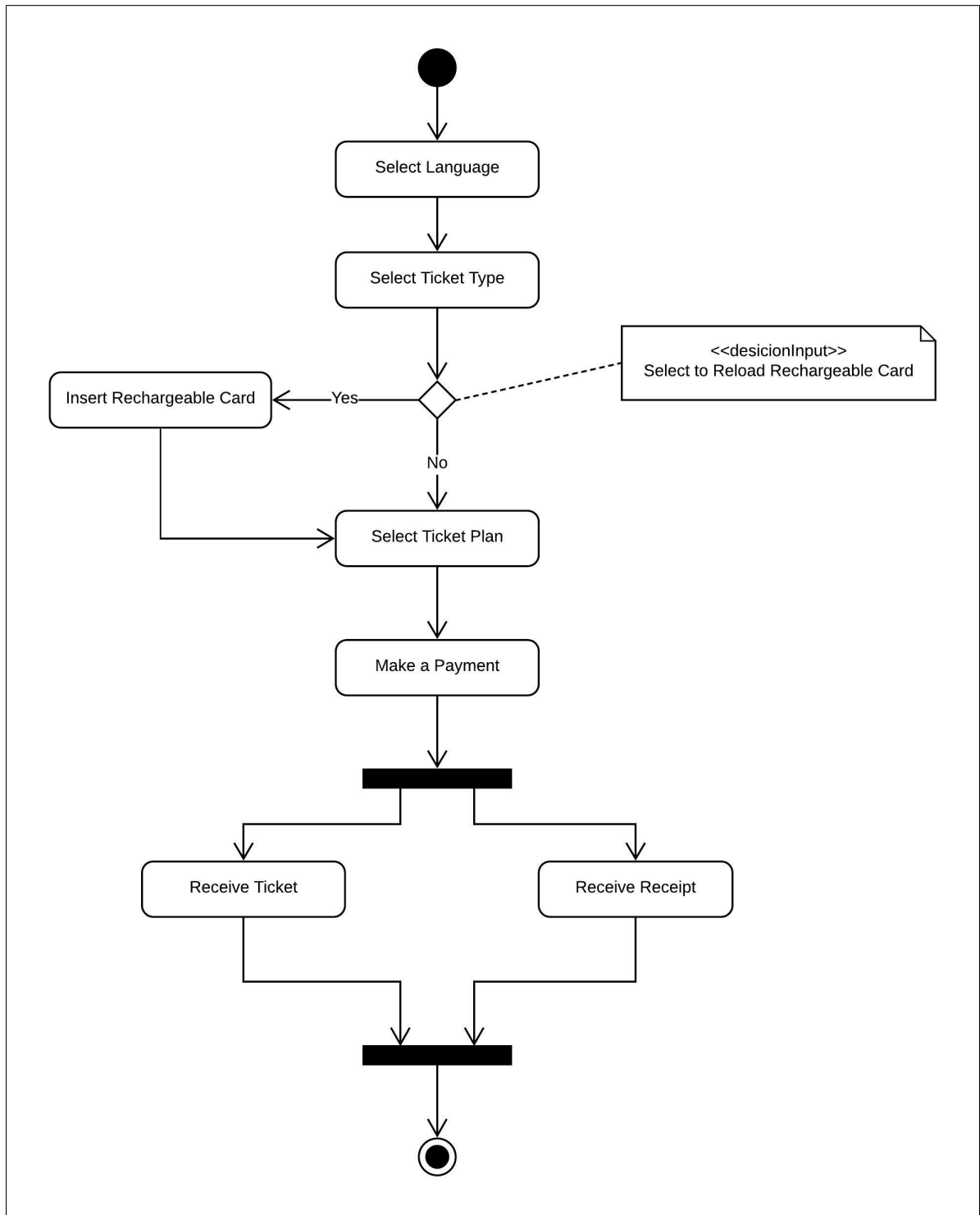


Figure 1.6: Activity Diagram: Buy Ticket

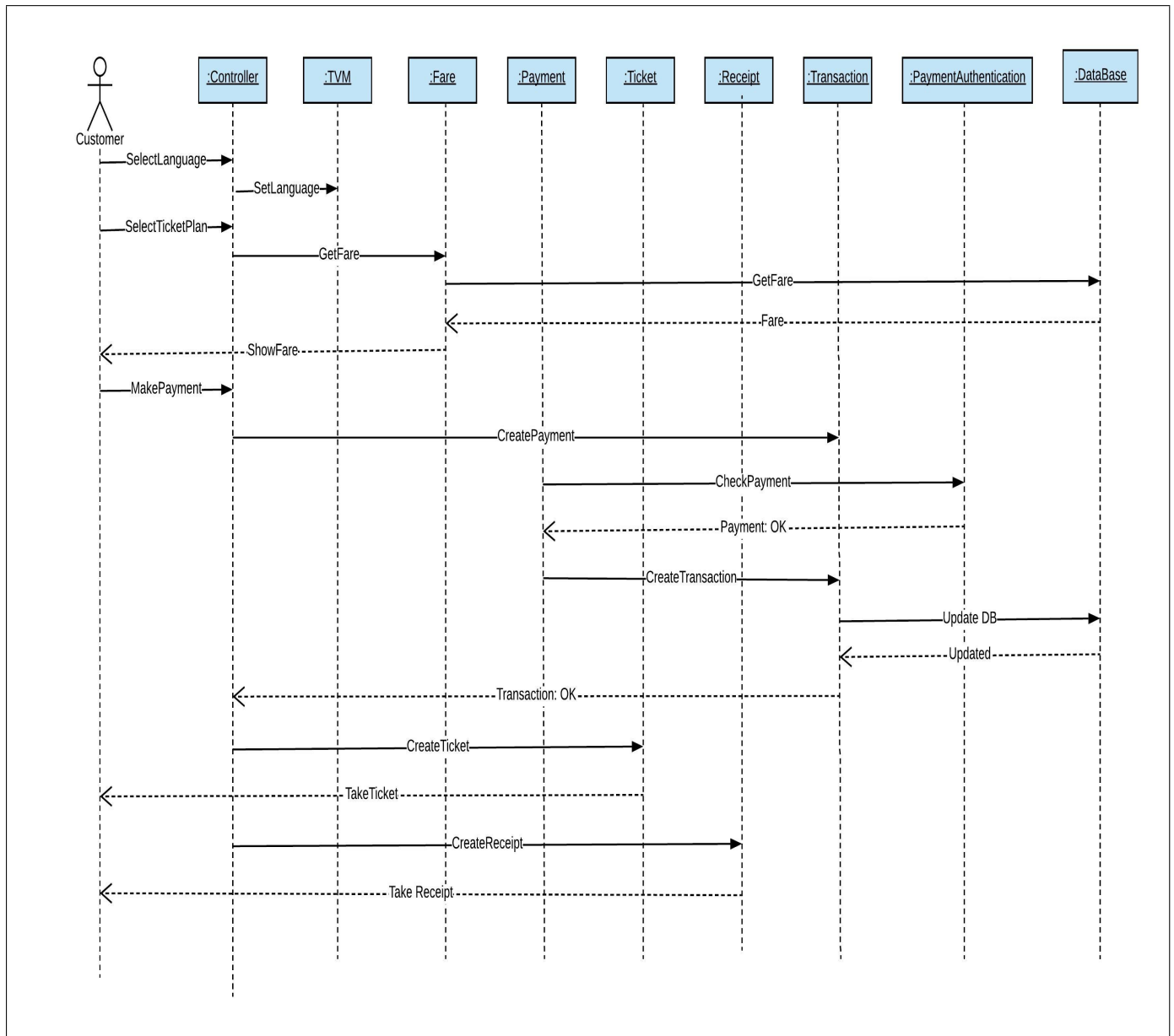


Figure 1.7: Sequence Diagram: Buy Ticket

#### 1.4.5 Use Case: Make a payment

Commuters use Ticket vending machine to make payments. Commuter is the primary actor that initiate the use case by choosing ticket plan. Secondary actor includes Data Store that supports and responds to the requests. The result for a normal use case scenario is that the customer has paid for the ticket plan and got the confirmation of payment successfull or not. Figure 1.8 explain this use case.

<b>Number</b>	4	
<b>Name</b>	Make a Credit Card Payment, Make a cash Payment	
<b>Summary</b>	Customer make payment by card or cash using TVM	
<b>Priority</b>	1	
<b>Preconditions</b>	User has a valid credit card, valid currency	
<b>Postconditions</b>	Transaction successful, user should get the ticket.	
<b>Primary Actor(s)</b>	TVM Customer / Bank Card Reader / Cash Receiver.	
<b>Secondary Actors(s)</b>	Payment Gateway.	
<b>Trigger</b>	User has chosen to buy a metro ticket	
<b>Main Success Scenarios</b>	<b>Step</b>	<b>Action</b>
	1	System displays ticket types
	2	User chooses the type of ticket
	3	System asks for payment method
	4	User select pay by cash or card
	5	System asks user to insert his card or cash (to cash receiver).
	6	User insert his cash or card.
	7	System asks user to enter the PIN (in case of card payment)
	8	User enter the PIN
	9	System connects to the payment gateway and authentication the user details.
	10	System create a transaction and records the transaction.
	11	System asks how user wants to receive the receipt
	12	User select paper receipt

	13	System prints and dispenses receipt and ticket
	14	User takes out receipt and ticket.
	15	System displays a message to remove the credit card
	16	User removes card
	17	System displays welcome message.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	6a	User insert card into card reader.
	6b	User insert cash into the cash receiver.
	7a	No pin asked in case of cash payment.
	9a	System authenticated the cash denominations in case of cash payment.
	11a	If payment is not successful user will not receive the print receipt message.
	15a	No message displayed in case of cash payment.
<b>Open Issues</b>		NA

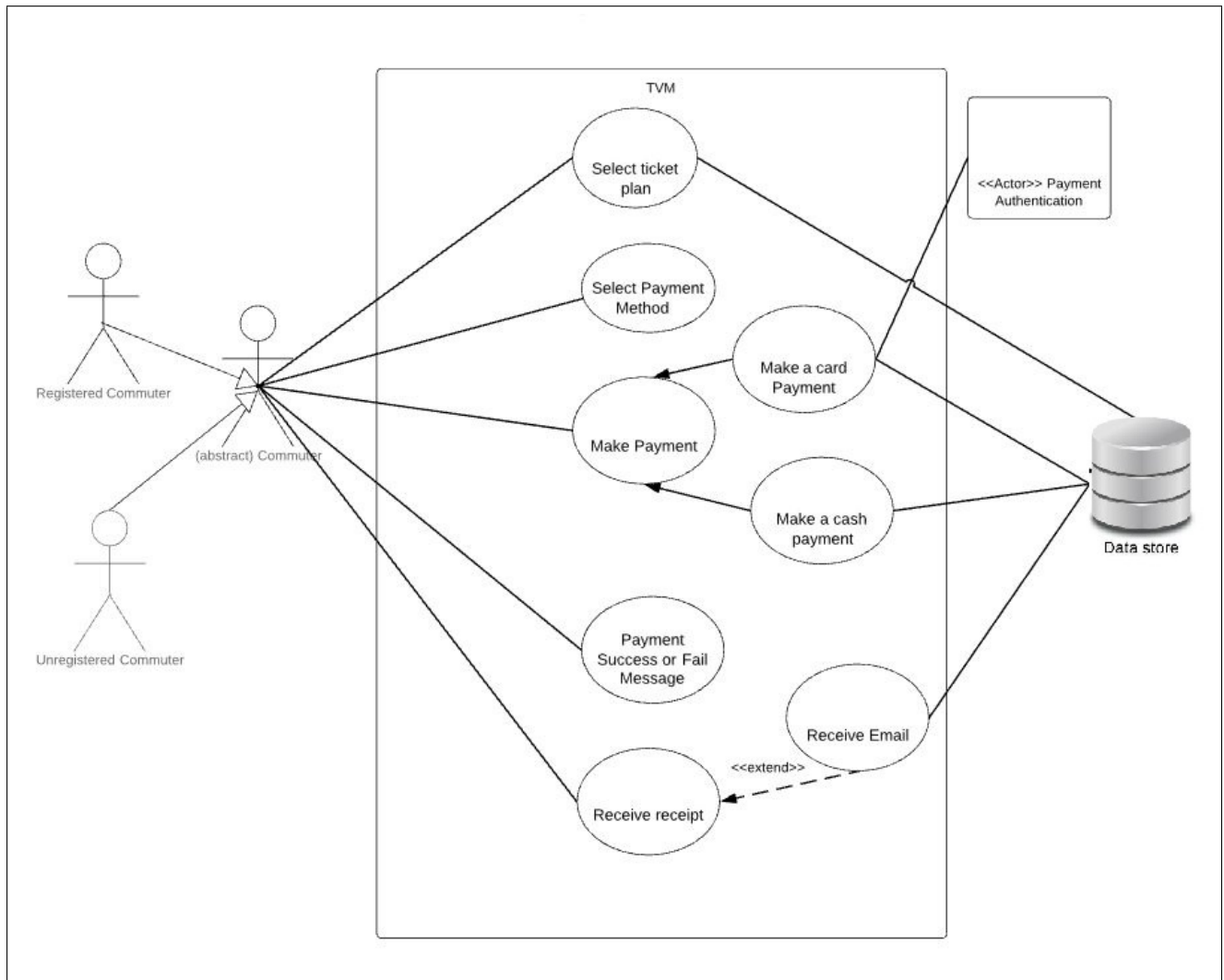


Figure 1.8: Use Case: Make Payment

#### 1.4.6 Use Case: Reload Rechargeable Card

Commuters use Ticket vending machine to recharge their rechargeable card. Commuter is the primary actor that initiate the use case by inserting his/her card. Secondary actors include Payment Authentication and Data Store that support and respond to the requests. The result for a normal use case scenario is that the customer has recharged his /her card and received the receipt. Figure 1.9 and 1.10 explain this use case.

<b>Number</b>	5	
<b>Name</b>	Reload Rechargeable Card	
<b>Summary</b>	Customer reloads the rechargeable card using TVM	
<b>Priority</b>	1	
<b>Preconditions</b>	User has a Rechargeable card	
<b>Postconditions</b>	User has recharged the card and received a receipt	
<b>Primary Actor(s)</b>	TVM Customer	
<b>Secondary Actors(s)</b>	Data Store	
<b>Trigger</b>	User has chosen to reload the card	
<b>Main Success Scenarios</b>	<b>Step</b>	<b>Action</b>
	1	User inserts the rechargeable card in the machine
	2	User selects the language
	3	User select the option to recharge the card
	4	System displays different recharge plans
	5	User chooses the recharge plan
	6	System asks for the payment method
	7	User choses the way of payment
	8	User done with the payment
	9	System displays the message to retrieve the rechargeable card
	10	User removes the card.
	11	System asks the user if he wants to print receipt
	12	User select the option from Yes or No
	13	System prints and dispenses receipt
	14	System displays welcome message
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
<b>Open Issues</b>	23	NA



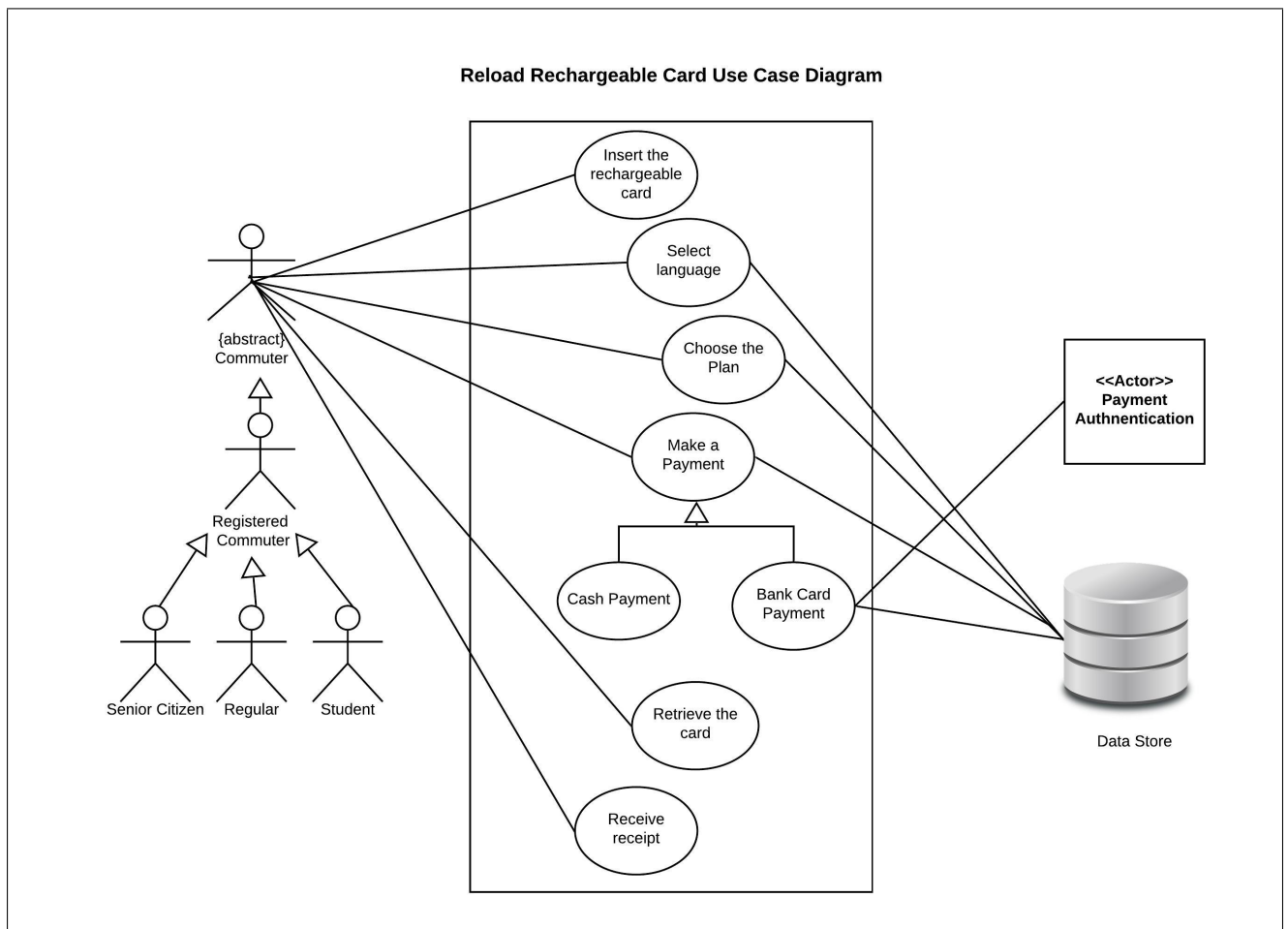


Figure 1.9: Use Case: Reload Rechargeable Card

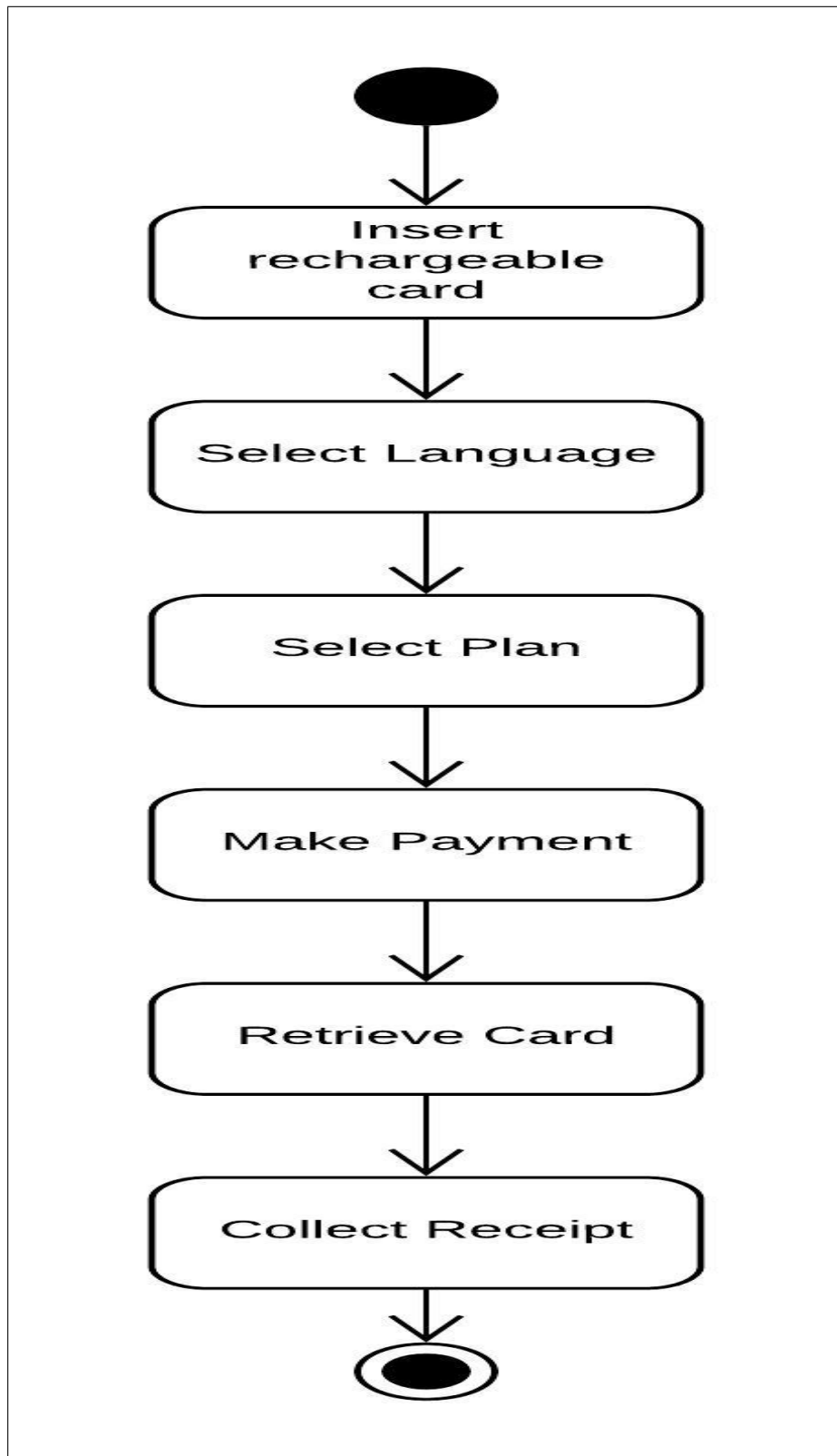


Figure 1.10: Activity Diagram: Reload Rechargeable Card

#### **1.4.7 Use Case: Update Ticket Specification**

Administrator can manage the tickets exiting in the system. He is the primary actor responsible for the Ticket Vending Machine Management. Administrator of the Ticket Vending Machine has the access to create new tickets, modify the existing tickets, view the existing tickets and delete the existing tickets.

<b>Number</b>	6.1	
<b>Name</b>	Create Ticket Specification	
<b>Summary</b>	Allows administrator to Create new ticket specifications	
<b>Priority</b>	2	
<b>Preconditions</b>	The Administrator should have a valid account with valid username and password. The Administrator should be logged into the application	
<b>Postconditions</b>	Administrator is able to add new ticket plans to the existing ticket plans.	
<b>Primary Actor(s)</b>	TVM Administrator	
<b>Secondary Actors(s)</b>	TVM and Customer	
<b>Trigger</b>	The Administrator has selected to add new ticket specifications to the systems.	
<b>Main Success Scenarios</b>	<b>Step</b>	<b>Action</b>
	1	Admin is given admin access to the system.
	2	Admin creates new ticket plans.
	3	Ticket details are specified by the admin with new ticket name, expiration date and price.
	4	The Administrator has added new ticket specifications to the system.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	3a	Admin does not have the admin access to the system.
	4a	The system gives an error.
<b>Open Issues</b>		NA

<b>Number</b>	6.2	
<b>Name</b>	Update Ticket Specification	
<b>Summary</b>	Allows administrator to update an existing ticket specification.	
<b>Priority</b>	2	
<b>Preconditions</b>	The Administrator should have a valid account with valid username and password. The Administrator should be logged into the application	
<b>Postconditions</b>	Administrator is able to edit ticket plans to the existing ticket plans.	
<b>Primary Actor(s)</b>	TVM Administrator	
<b>Secondary Actors(s)</b>	TVM and Customer	
<b>Trigger</b>	The Administrator has selected to update ticket specifications on the systems.	
<b>Main Success Scenarios</b>	<b>Step</b>	<b>Action</b>
	1	Admin selects the tickets to be modified.
	2	Admin change the ticket specification.
	3	Admin saves the new ticket specifications.
	4	Admin updates the system with updated ticket plans.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	3a	Admin does not have the admin access to the system.
	4a	The system gives an error.
<b>Open Issues</b>		NA

<b>Number</b>	6.3	
<b>Name</b>	Delete Ticket Specification	
<b>Summary</b>	Allows administrator to delete existing ticket specifications	
<b>Priority</b>	2	
<b>Preconditions</b>	The Administrator should have a valid account with valid username and password. The Administrator should be logged into the application	
<b>Postconditions</b>	Administrator is able to delete ticket plan from system.	
<b>Primary Actor(s)</b>	TVM Administrator	
<b>Secondary Actors(s)</b>	TVM and Customer	
<b>Trigger</b>	The Administrator has selected to delete ticket specifications on the systems.	
<b>Main Success Scenarios</b>	<b>Step</b>	<b>Action</b>
	1	Admin selects the tickets to be changed.
	2	Admin deletes the selected ticket specification.
	3	The system is updated after deleting.
	4	Application updates the list of new tickets available.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
	3a	Admin does not have the admin access to the system.
	4a	The system gives an error.
<b>Open Issues</b>		NA

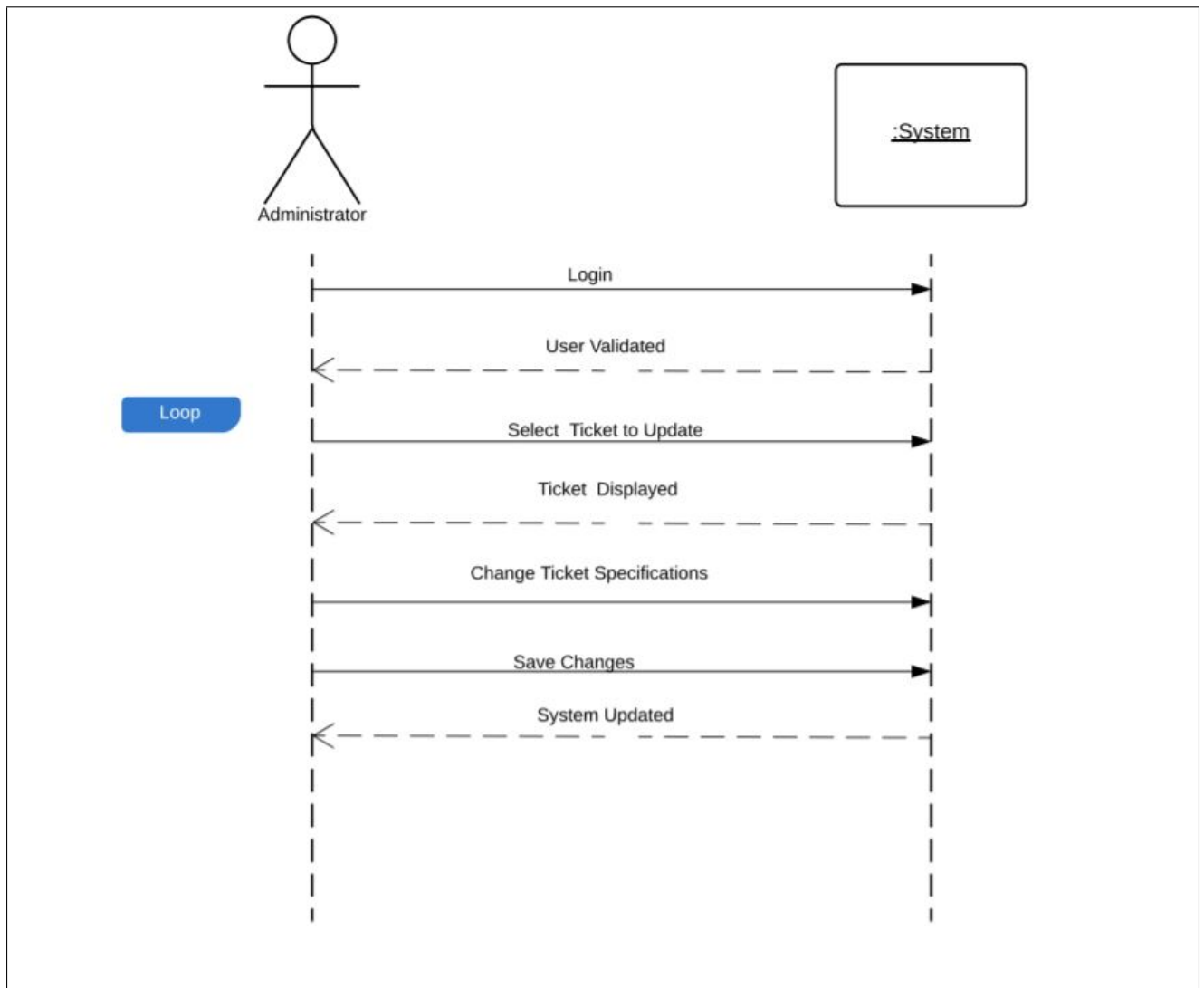


Figure 1.11: Sequence Diagram: Admin Sytem

### 1.4.8 Use Case: Customer Registration

Commuter(New User) use Ticket Vending Machine to register himself/herself with the STM. Commuter is the primary actor that initiate the use case by choosing the register option. Secondary actor includes Data Store that supports and responds to the requests. The result for a normal use case scenario is that the customer got registered with the STM. Figure 1.12 and 1.13 explain this use case.

<b>Number</b>	7	
<b>Name</b>	Customer Registration	
<b>Summary</b>	Customer register with his personal information to get the card	
<b>Priority</b>	2	
<b>Preconditions</b>	N/A	
<b>Postconditions</b>	User is registered	
<b>Primary Actor(s)</b>	TVM Customer	
<b>Secondary Actors(s)</b>	Data Store	
<b>Trigger</b>	User has chosen to register himself for the STM	
<b>Main Success Scenarios</b>	<b>Step</b>	<b>Action</b>
	1	User selects the register option
	2	System asks for his personal identification information
	3	User fills all the information and submit it
	4	System asks for the username and password
	5	User creates a username and password
	6	System checks for the validity of the user
	7	System registers the user and shows a registered user message
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
<b>Open Issues</b>		NA



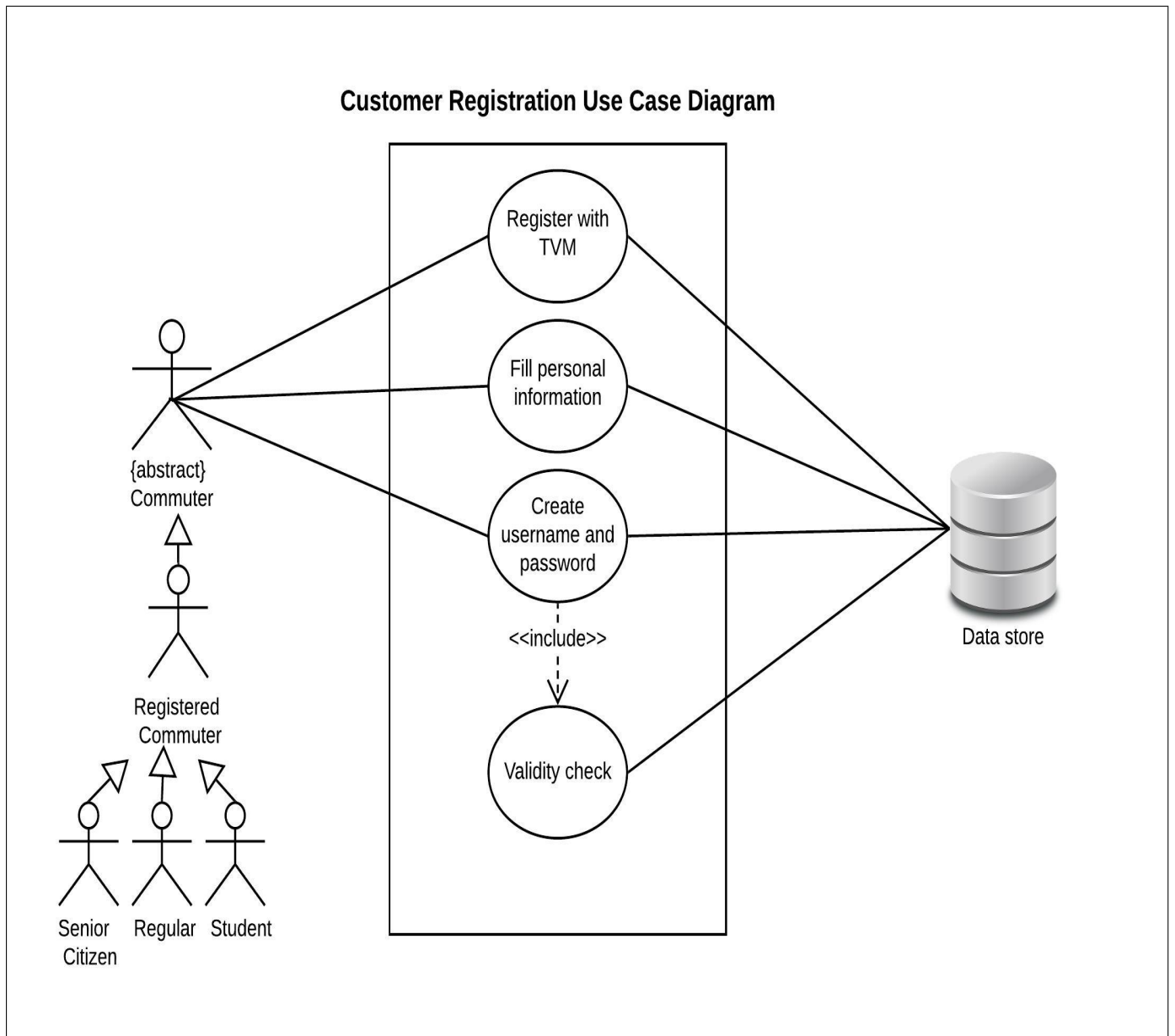


Figure 1.12: Use Case: Customer Registration

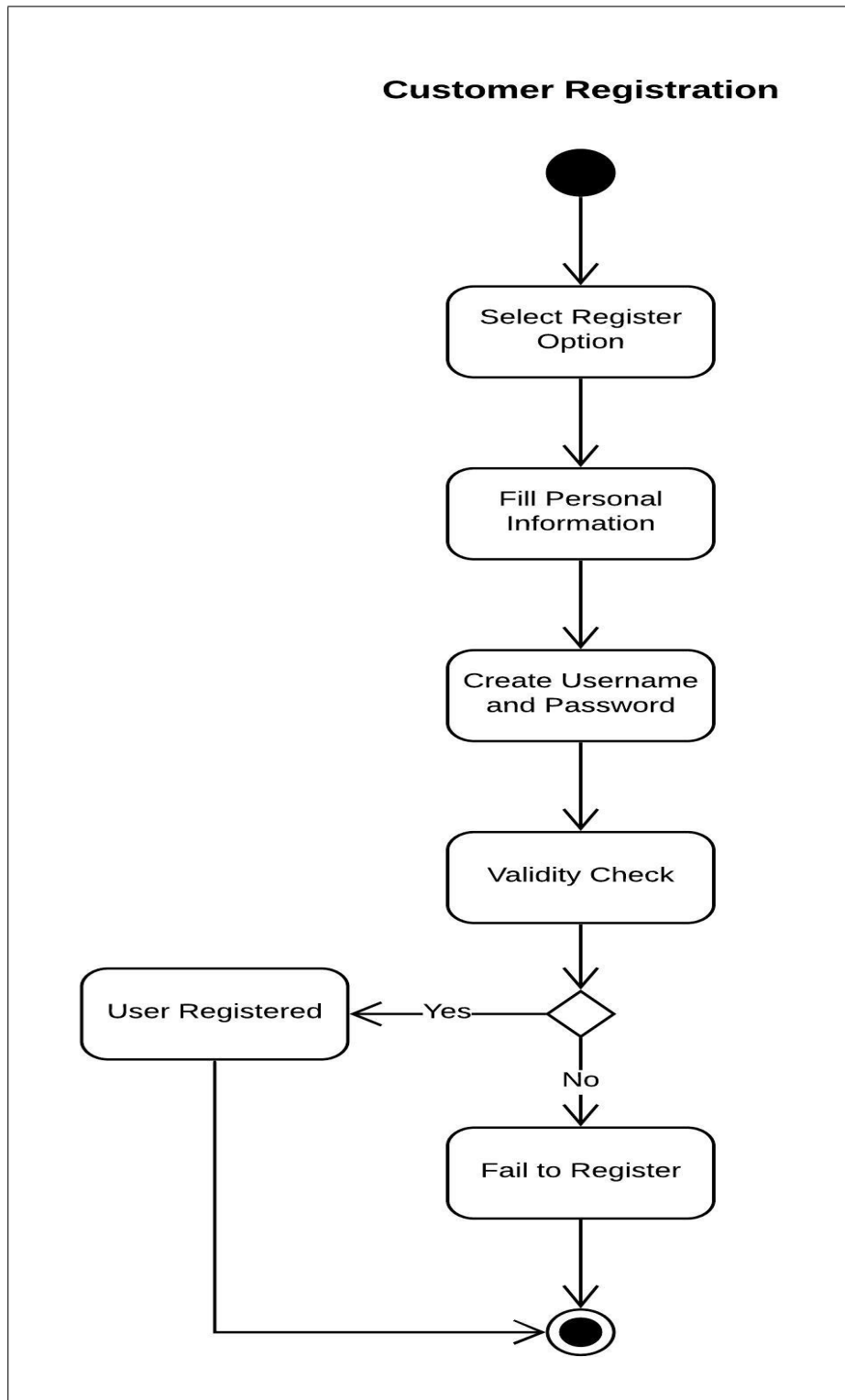


Figure 1.13: Activity Diagram: Customer Registration

#### 1.4.9 Use Case: False Login

<b>Number</b>	7	
<b>Name</b>	False Login	
<b>Summary</b>	Hacker try to login into someone else account.	
<b>Priority</b>	1	
<b>Preconditions</b>	N/A	
<b>Postconditions</b>	Login should fail with appropriate error message.	
<b>Primary Actor(s)</b>	Hacker / TVM.	
<b>Secondary Actors(s)</b>	Data Store	
<b>Trigger</b>	User account locked after 5 unsuccessful login attempts.	
<b>Main Success Scenarios</b>	<b>Step</b>	<b>Action</b>
	1	Hacker try to login into system.
	2	Hacker enters username and password.
	3	System will validate user details from database system.
	4	System decline unauthorized access. System should display appropriate error message.
	5	System should display appropriate error message.
	6	Hacker keeps on entering invalid credentials to try to login into someone account.
	7	System will lock user account after 5 unsuccessful login attempts.
	8	System should display account locked error message to user.
<b>Extensions</b>	<b>Step</b>	<b>Branching Action</b>
<b>Open Issues</b>		NA

### 1.4.10 Use Case: Password Reset

#### Use Case Diagram

The password reset use case diagram explains how a customer will be able to reset the password of his/her account. The password reset will include another use case called authenticate. This can be either by security questions or by email.

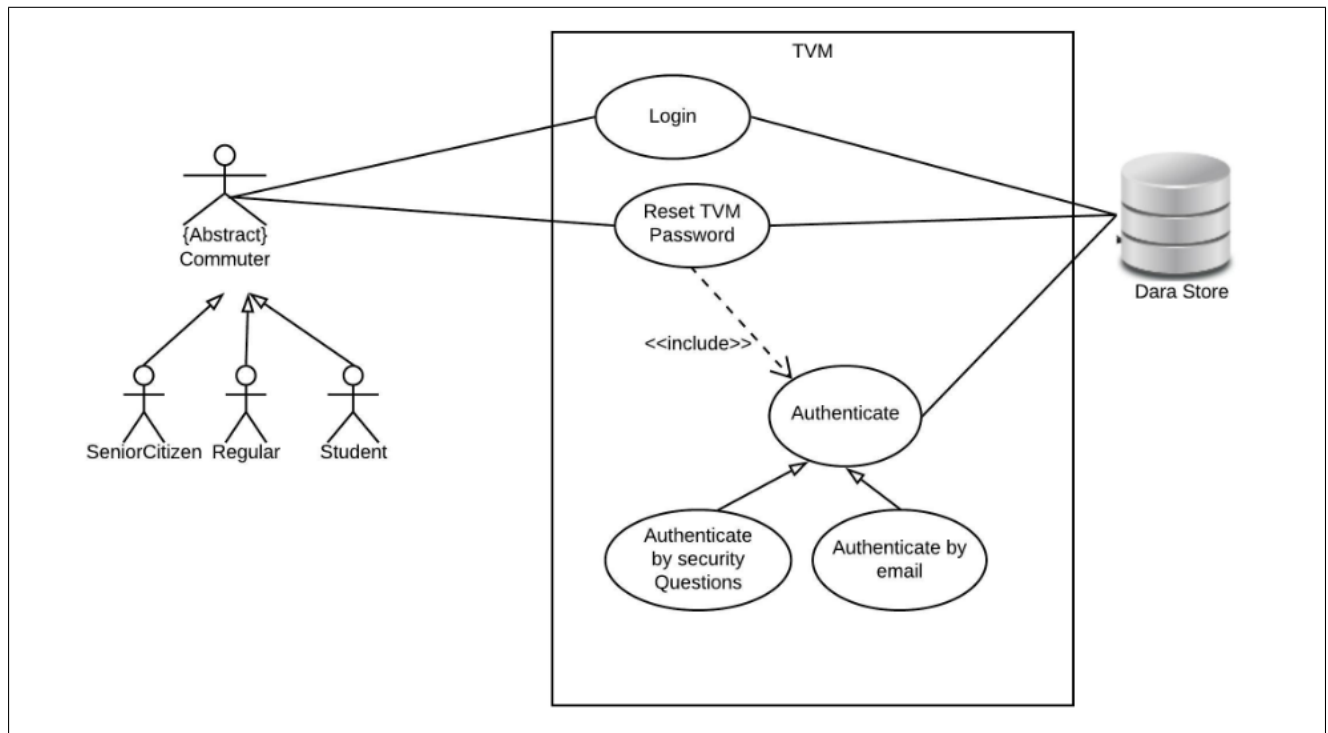


Figure 1.14: Use Case Diagram: Password Reset

#### Activity Diagram

The activity diagram is for the Password reset. The sequential steps are as shown, and if the password change authentication fails, the process will be terminated.

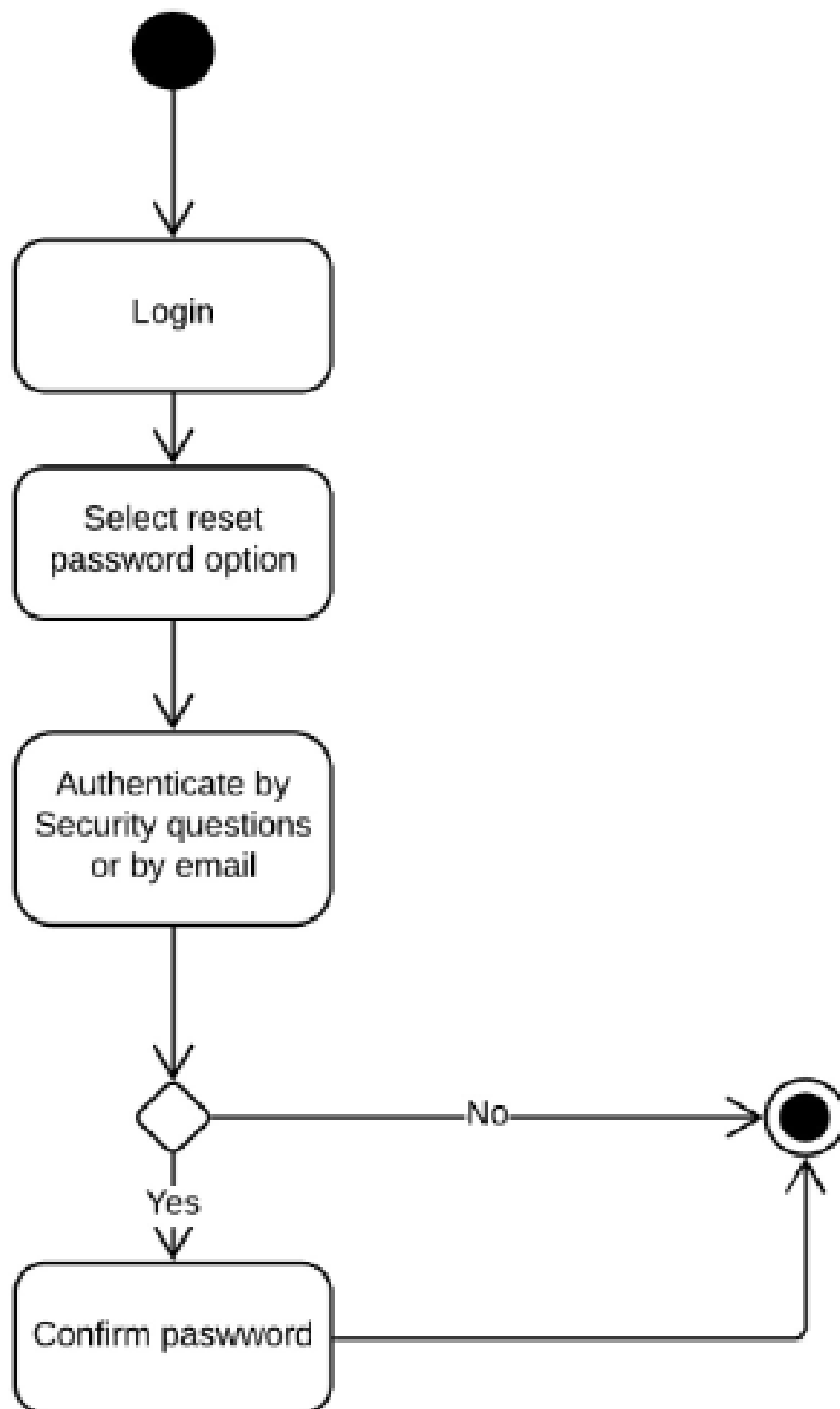


Figure 1.15: Activity Diagram: Password Reset

### 1.4.11 Use Case: Payment Fraud

The Payment Fraud use case diagram shows how the payment authentication use case can be threatened by the negative use case called Flood system and Steal card information. The name of the use cases explains the activity of negative use cases.

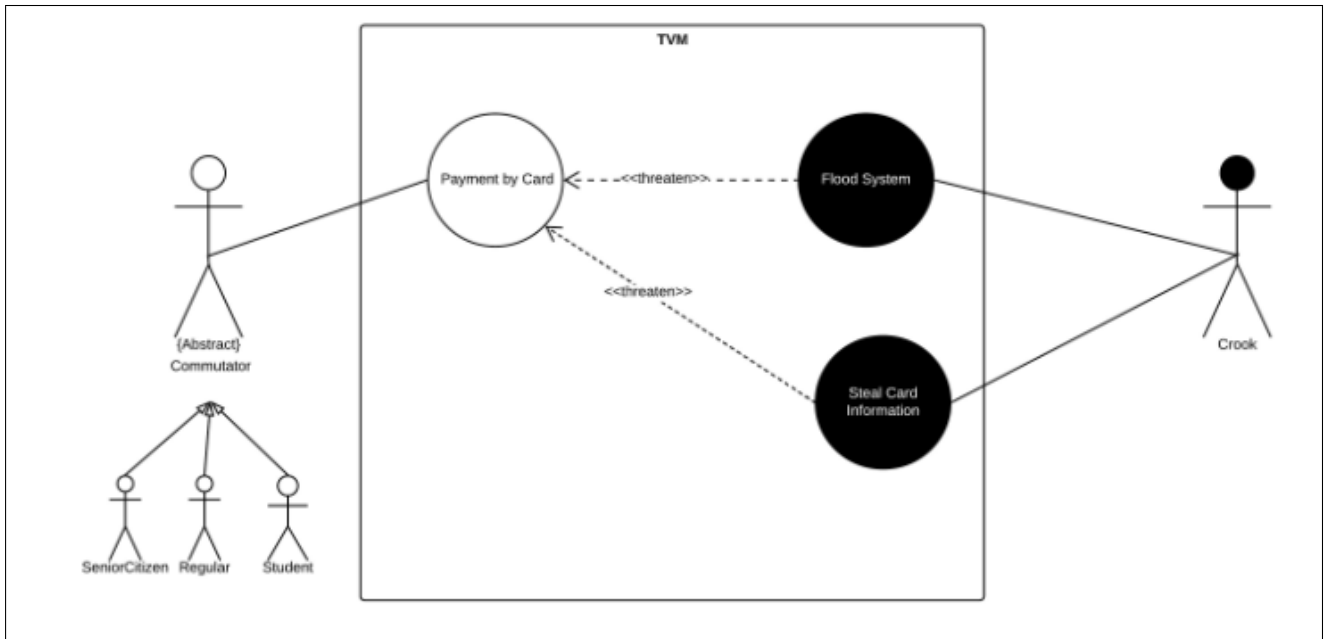


Figure 1.16: Use Case Diagram: Payment Fraud

## References

- [1] Pankaj Kamthan. *BrainStorming and Mind Map*. 2019.
- [2] Pankaj Kamthan. *Introduction to Domain Modeling*. 2019.
- [3] Pankaj Kamthan. *Understanding Context*. 2019.
- [4] Pankaj Kamthan. *Use Case Modeling*. 2019.
- [5] STM. *STM website*. 2019. URL: <http://www.stm.info/en>.

## Glossary

**iGo** The online Ticket Vending Machine Web Application integrating with STM system. 2

**Non-Rechargeable Card** One time use and throw card.. 2

**OPUS** The online Ticket Vending Machine Web Application integrating with STM system. 2

**Rechargable Card** A card that can be recharged to be used monthly, weekly.. 2

**Reduced Fare** Fare offered at reduced rates than the normal prices for certain users that qualify for it.. 5

**Regular Fare** Normal Fare Prices.. 9

**Transaction** A digital record is generated when user interacts with the system.. 2

**TVM** The online Ticket Vending Machine Web Application integrating with STM system. 2

## Acronyms

**STM** Société de transport de Montréal. 2

## A Appendix

### A.1 Interviews

Users are Interviewed and the results are accessible at: <https://github.com/m3hrn4z/SRS>