

Planning and Implementing an Object-Oriented Software System

Pairs Practical Project

Object-Oriented Constructs / Systems Analysis & Design

GitHub Project: <https://github.com/AlissonDMoura/Movie-Rental.git>

Name:

Alisson Alves De Moura

Student number: 2019142

Marcos Vinicius Mariano Rodrigues

Student number: 2019146

Table of Content

Introduction & Objective	2
Problem definition	2
A – System	2
System Requirements	3
A – Optional	3
Wireframe	4
Use Case Diagrams	5
First successful scenario: Early return	5
Second successful scenario: Late movie return	5
Third successful scenario: Recurring User	6
First failure scenario: Bad pin or Invalid credit card	6
Class Diagram	7
Sequence Diagram	8
State Diagram	9
Activity Diagram	10
User Story for future implementations	11
Database	12
Project Management	13
Research item 1	13
Research item 2	14
References	16

Introduction & Objective

This Assessment has as objective the design and writes a movie rental software for kiosks that allows the user to rent movies and return them to different kiosks. During the assessment, the reader will go through all the phases of the creation of such a project from planning to execution and some future features.

By the end of this paper, the reader should be able to see clearly all phases of creation and implementation of an IT Project that utilises techniques of SAAD (System Analysis and Design) and OOC (Object Orientation Constructs), using Java Language, MySQL and the Design and Management tool called Visual Paradigm.

Problem definition

Consider the scenario where the movie industry is releasing a huge amount of movies every year, it turned not possible for the main public to buy every launched movie to watch on their DVD players at a low cost with easy access, therefore with a large share of the population presented growing demand for watching movies once solution presented is to have at their disposition the power of renting any movie with a simple and elegant system that provides the user choices and benefits.

A – System

Being fully flexible is vital as the growing demand for this solution brings concurrency to this market, the user would be fully satisfied if there's very little limitation on their demands, therefore convenience is the keyword for this matter.

The objective of getting over limitations as ***logistic, usability and flexibility*** define the problem we will address.

Logistic: Users have to be able to rent a movie from different kiosks and return them into different kiosks, this demands an interconnected system and stock control.

Usability: Users have to be able to use the kiosks by intuition, therefore the platform has to be user friendly and reduce the amount of information necessary for them to accomplish their objective.

Flexibility: The public, in general, have different motivations when acquiring a movie, the objective can vary from an actual purchase, multiple movies, the number of days they need to watch and the costs that imply these different scenarios should follow the necessity of the client.

System Requirements

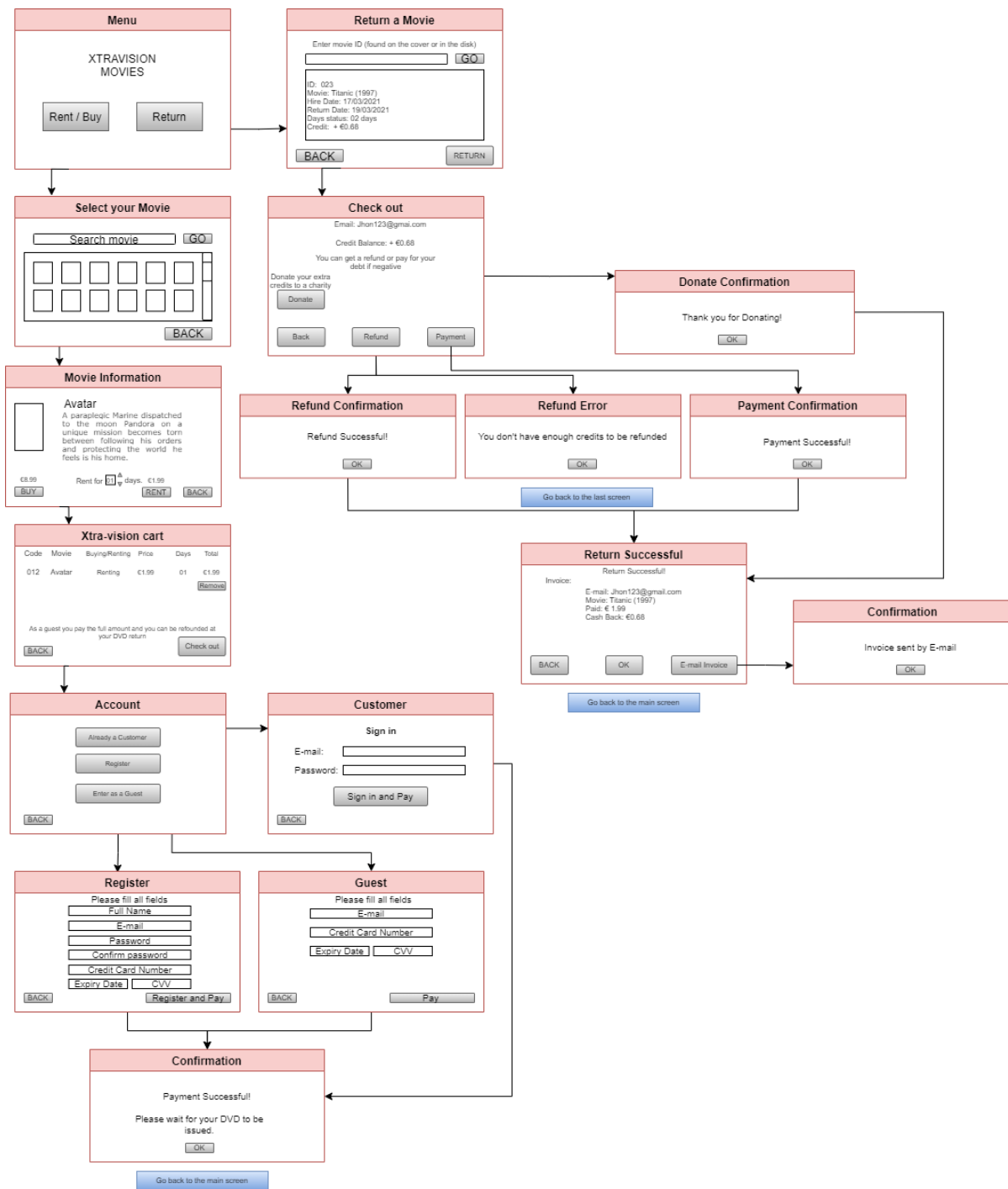
The system requires a range of movies stored in an accessible database where all kiosks can communicate with for retrieving and sharing information regarding their status and recurrent users that have enrolled in the system for more benefits.

- A Kiosk should be able to check and update its own movie amount.
- A Kiosk should be able to inform the system's network of the status of the amount of DVD's for every different movie it has and issue an alert in case one is running low or running high.
- A Kiosk should be able to charge users for movies and hold credit for misuse of the platform before handing the movie to the user.
- A Kiosk should be able to create new users and collect users information for login to the platform for benefits, this information should be updated to the network when finished.
- A Kiosk should be able to inform users about the number of movies it has, and every option of purchase informing also the costs for every option.

A – Optional

- A Kiosk could be able to send confirmation emails.
- A Kiosk could provide a wide search for movies in other kiosks.
- A Kiosk could be able to check the DVD returned wasn't damaged or changed.
- A Kiosk could be able to give the user the options of renting a movie using a Guest Account.
- A Kiosk could have a GUI for easy user manipulation.

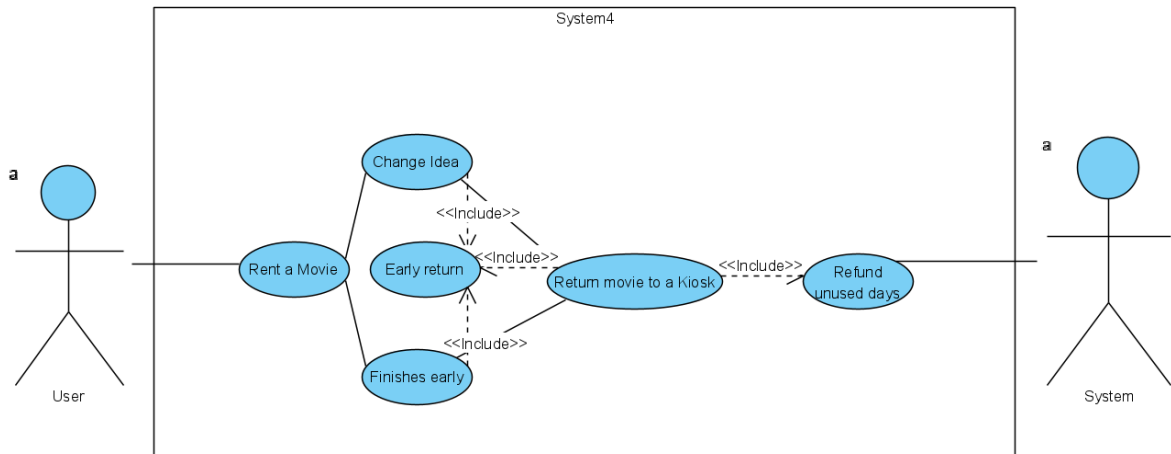
Wireframe



Use Case Diagrams

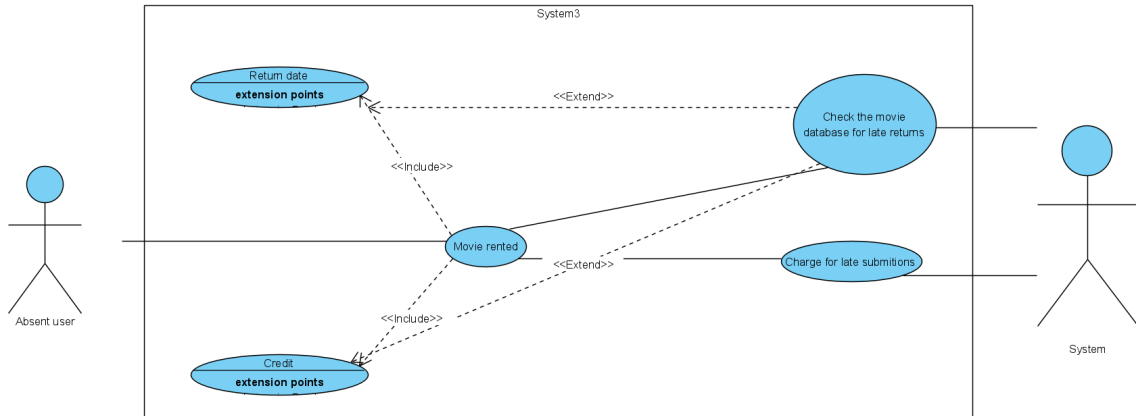
First successful scenario: Early return

The user can return a movie before the date selected, He will get a refund for unused days.



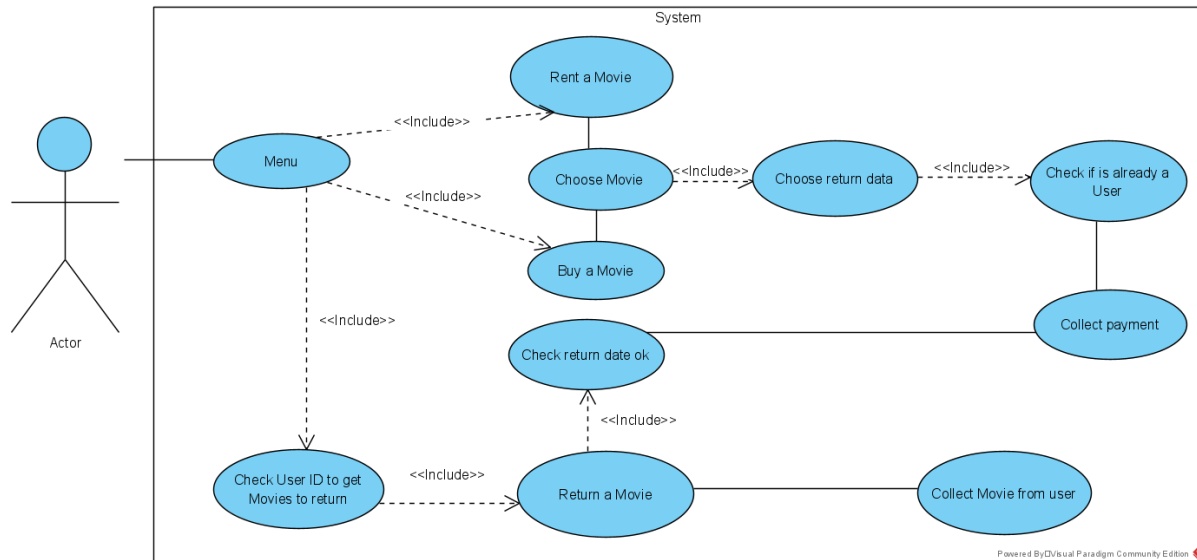
Second successful scenario: Late movie return

The user can return the movie after the date selected but he will be charged for the late return.



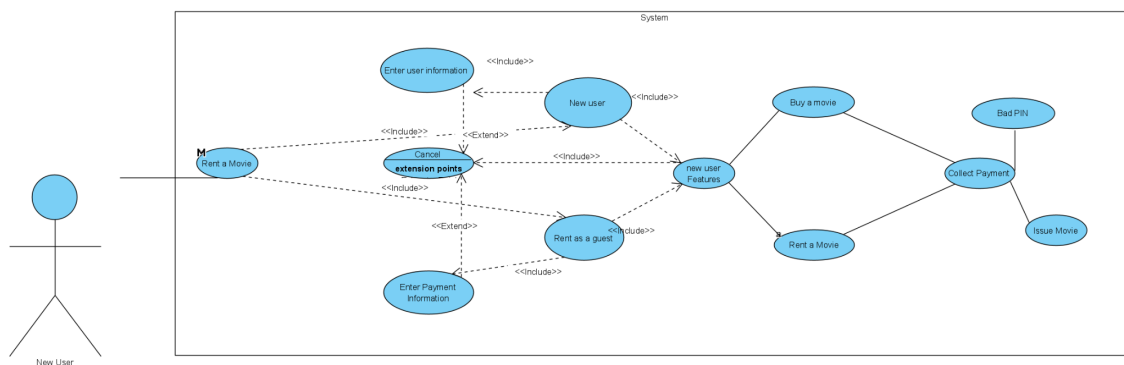
Third successful scenario: Recurring User

The scenario for a recurring user, renting and returning the movie.



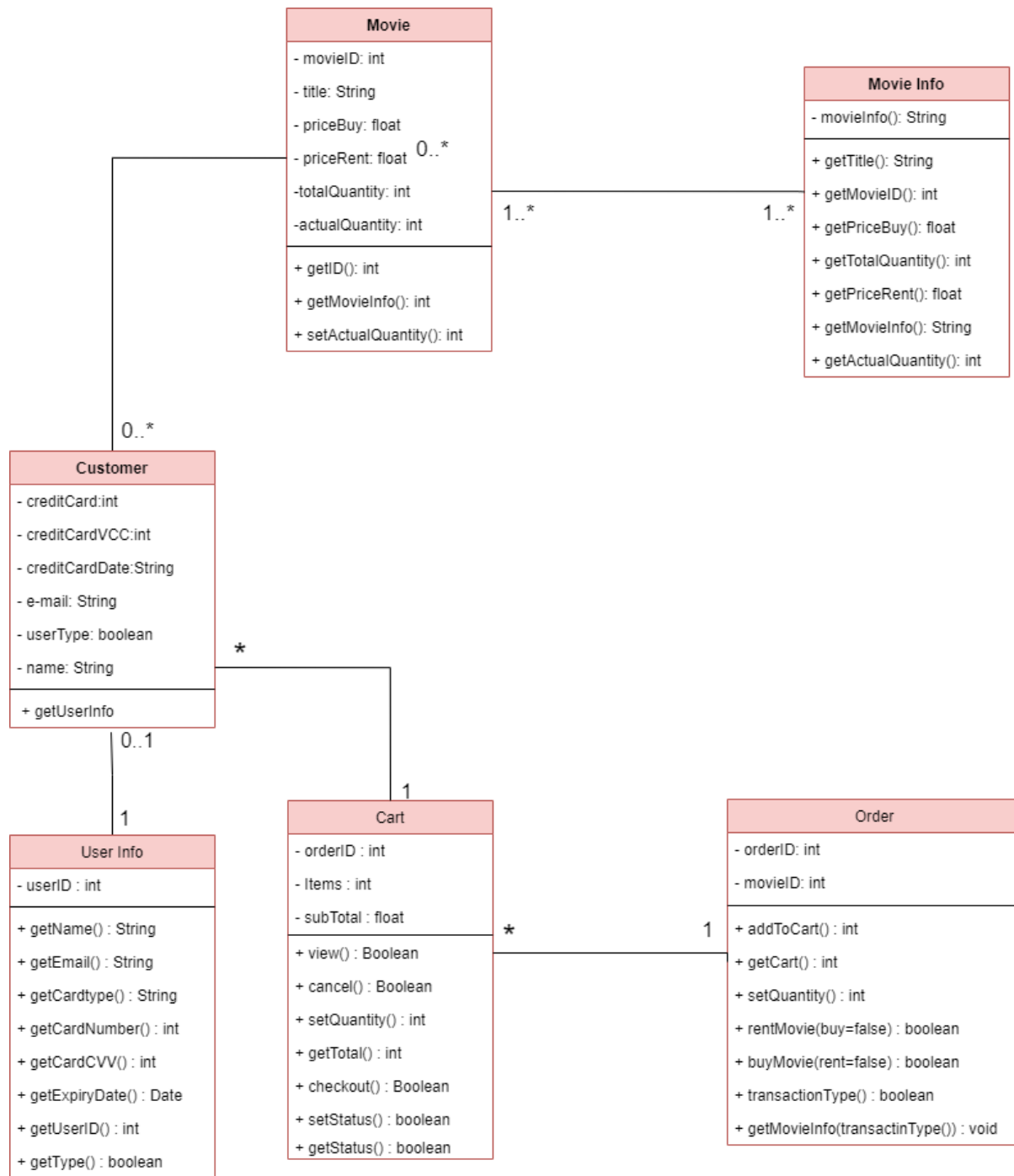
First failure scenario: Bad pin or Invalid credit card

The new user or recurring user can register the credit card after making the order but if the credit card or bad pin is wrong the program will show an error message.

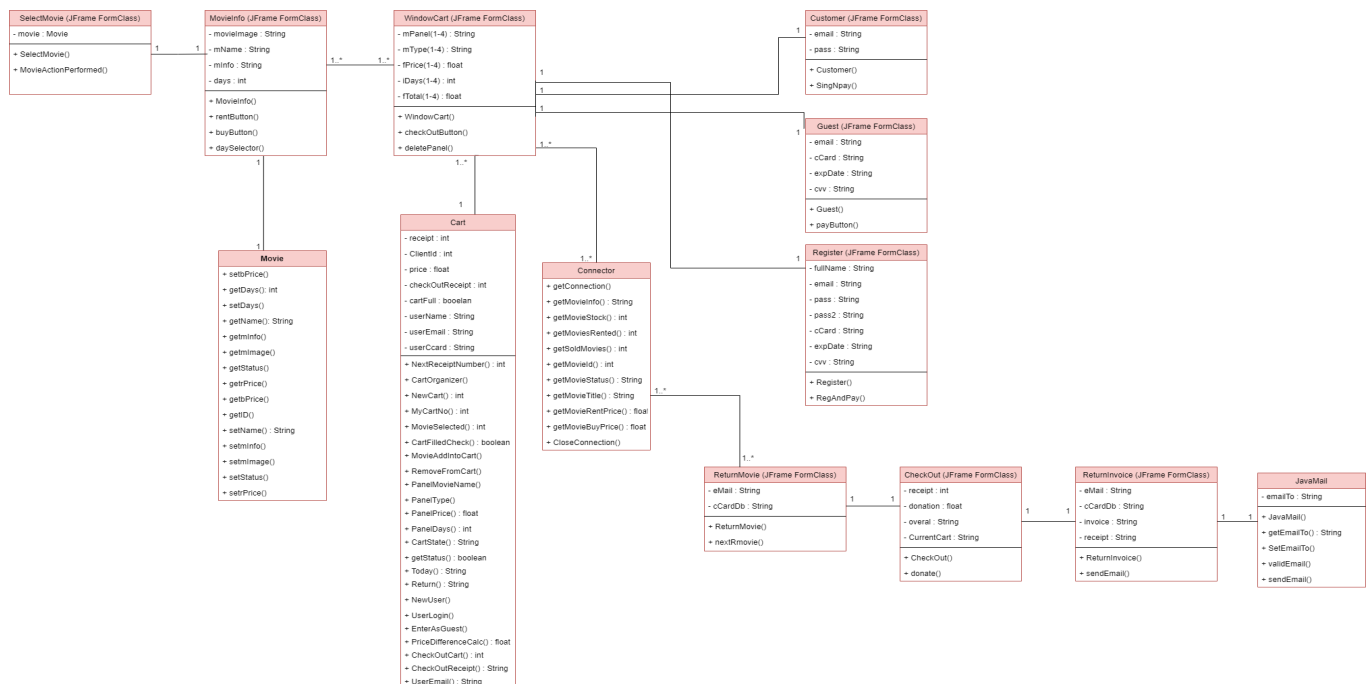


Class Diagram

The Diagram Class we planned

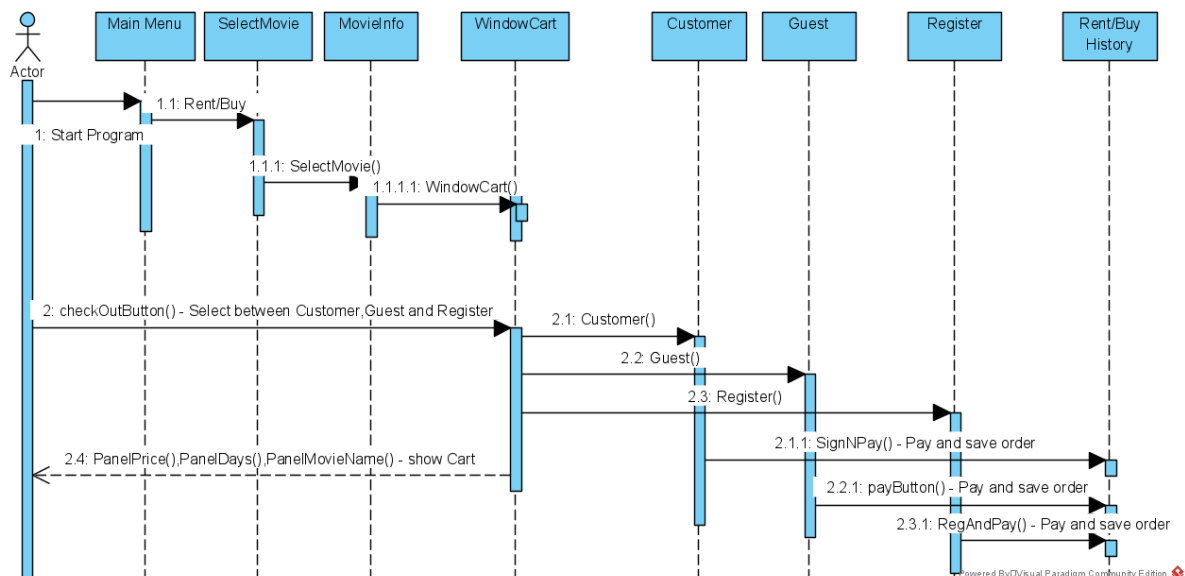


The Final Diagram Class

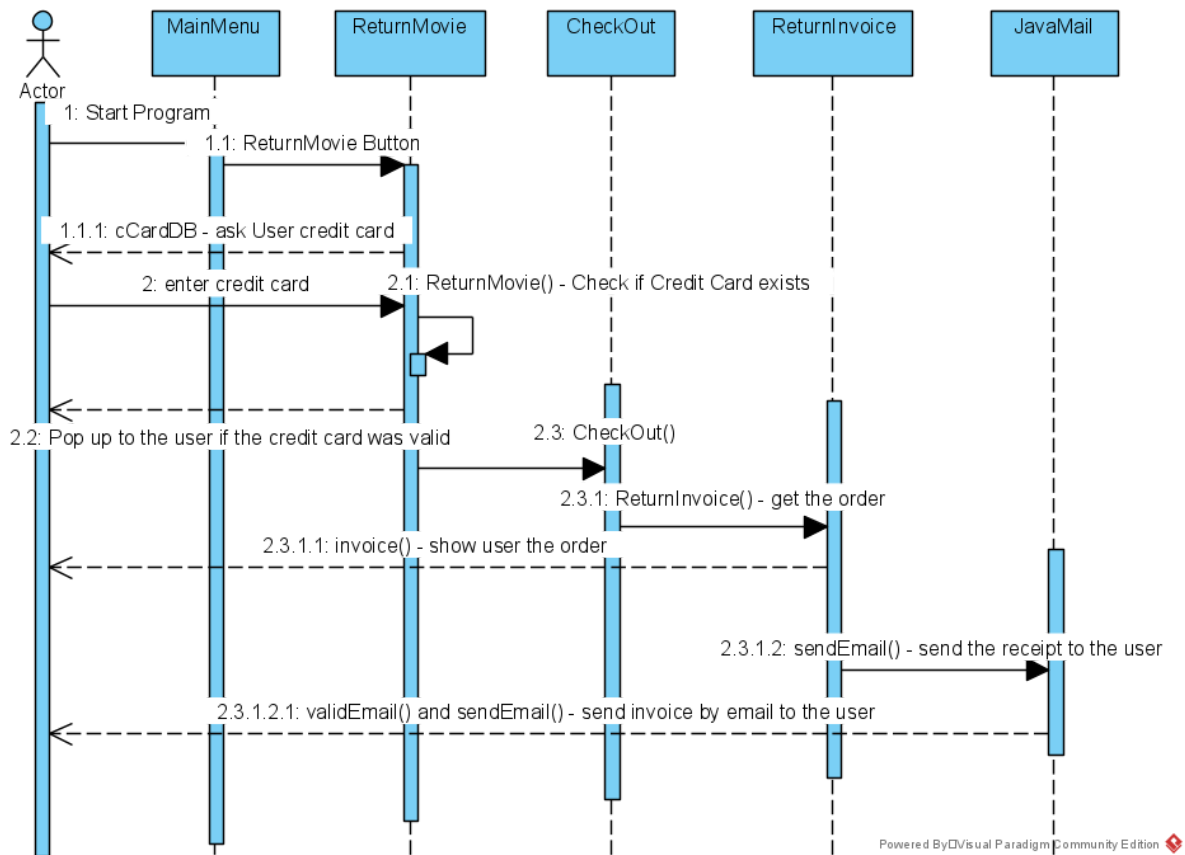


Sequence Diagram

Buy and Rent Sequence Diagram

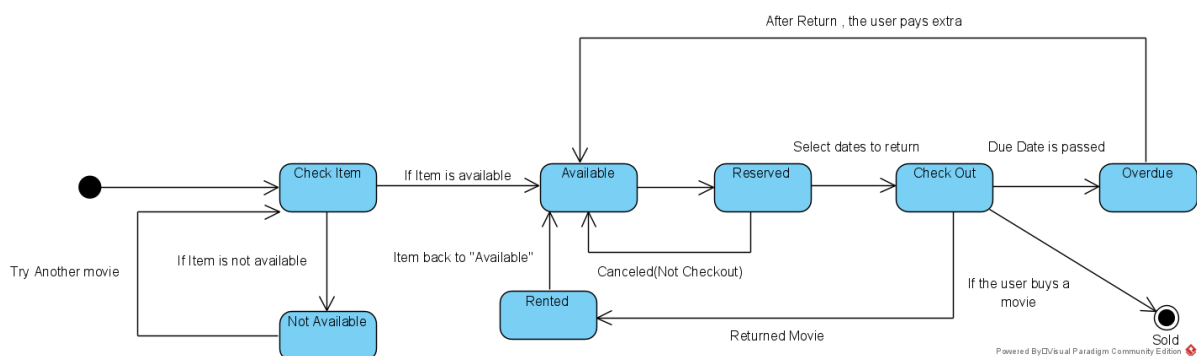


Return movie sequence diagram



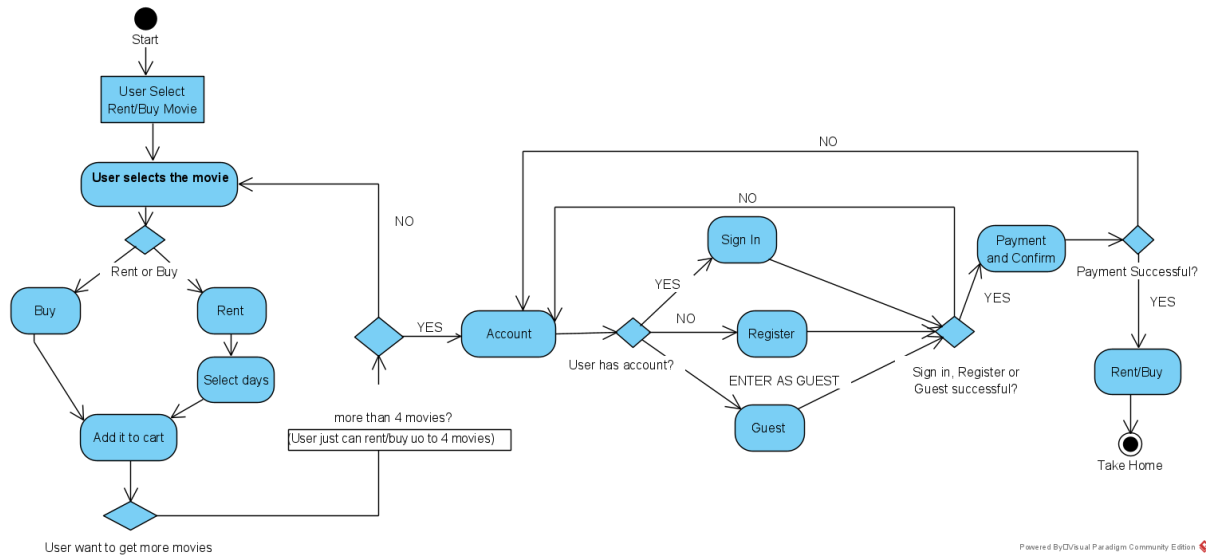
State Diagram

State of the system when the movie gets rented or sold.

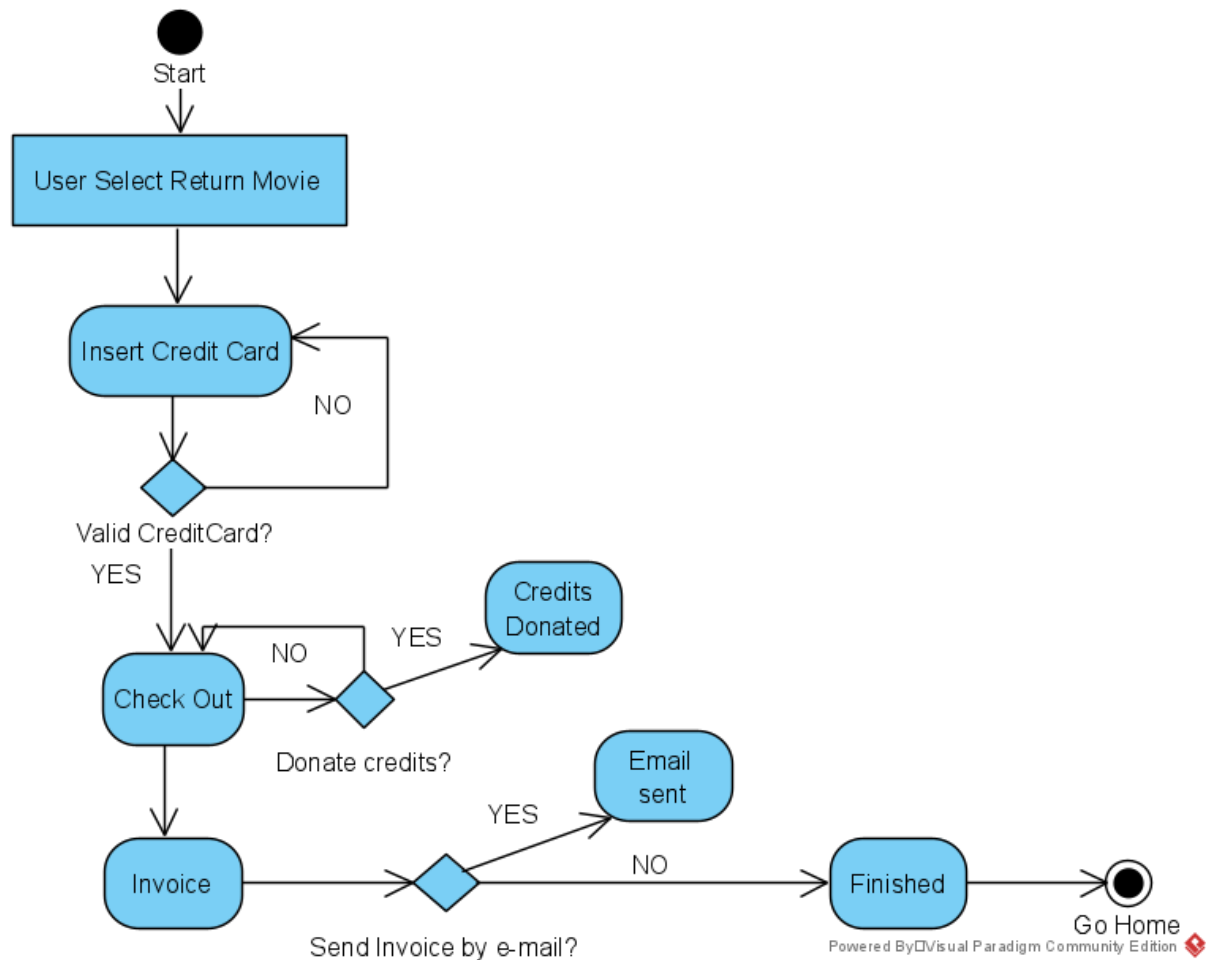


Activity Diagram

Activity Diagram for buying and renting movies



Activity Diagram for returning movies



User Story for future implementations

User Story

Title:	Priority:	Estimate:
Search Bar		
User Story: As a recurring user, I want the Xtra-vision to add a search movies bar, so that I can filter the movies by name and gender.		
Acceptance Criteria: Given that I'm logged in user, when I click on the select movie frame on the top, then I'll be able to insert a word and the system will filter the movies they have in the database.		

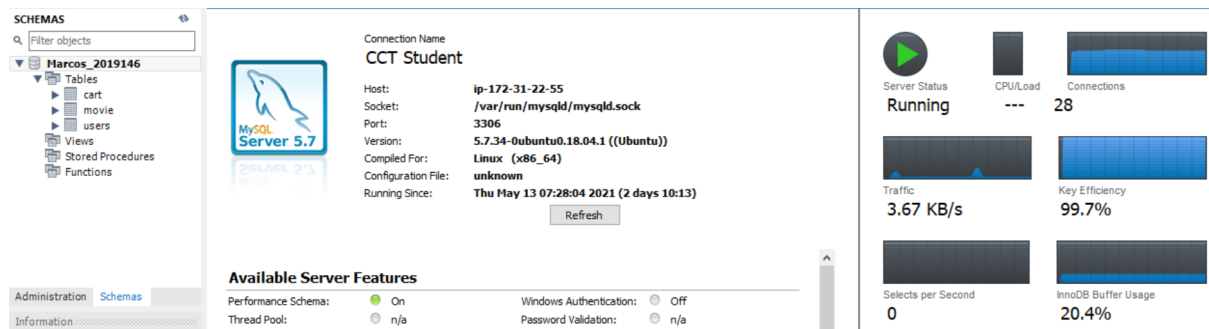
User Story

Title:	Priority:	Estimate:
Release Movies		
User Story: As a recurring user, I want the Xtra-vision to give me suggestions of the new movies, so that I can rent or buy the newest movies and chat with my friends about them.		
Acceptance Criteria: Given that I'm logged in user, when I click on the select movie, then the system will submit a message with the newest movies in the system.		

User Story

Title:	Priority:	Estimate:
Extend Rent		
User Story: As a recurring user, I want the Xtra-vision to give me a way to extend my return date online, so that I can extend for 2 more days the movie return and do it online without going to the Kiosk.		
Acceptance Criteria: Given that I'm logged in user, when I click on the return movie, then the frame will have a button on the left side that the user can click and insert the number of days you can extend the rent.		

Database



Movie Table

	idMovie	title	movieInfo	priceBuy	priceRent	status
▶	1	Titanic (1997)	A seventeen-year-old aristocrat falls in love wit...	10	2	SOLD
	2	Titanic (1997)	A seventeen-year-old aristocrat falls in love wit...	10	2	SOLD
	3	Titanic (1997)	A seventeen-year-old aristocrat falls in love wit...	10	2	In stock
	4	Titanic (1997)	A seventeen-year-old aristocrat falls in love wit...	10	2	In stock
	5	Titanic (1997)	A seventeen-year-old aristocrat falls in love wit...	10	2	In stock
	6	Titanic (1997)	A seventeen-year-old aristocrat falls in love wit...	10	2	In stock
	7	Titanic (1997)	A seventeen-year-old aristocrat falls in love wit...	10	2	In stock
	8	Titanic (1997)	A seventeen-year-old aristocrat falls in love wit...	10	2	In stock
	9	Titanic (1997)	A seventeen-year-old aristocrat falls in love wit...	10	2	In stock
	10	Titanic (1997)	A seventeen-year-old aristocrat falls in love wit...	10	2	In stock

Cart Table

	receipt	creditcard	status	date	return	Item1	Type1	Days1	Item2	Type2	Days2	Item3	Type3
▶	11	1111222233334444	Due	15/05/2021	16/05/2021	31	RENTED	1	0	null	0	0	null
	12	4444888866662222	Finished	15/05/2021	null	21	SOLD	0	0	null	0	0	null
	13	4444888866662222	Finished	15/05/2021	null	1	SOLD	0	0	null	0	0	null
	14	1234567891234567	Finished	15/05/2021	null	1	SOLD	0	0	null	0	0	null
	15	2222444466668888	Finished	15/05/2021	null	11	SOLD	0	0	null	0	0	null
	16	1234567891234567	Due	15/05/2021	19/05/2021	32	RENTED	4	0	null	0	0	null
	17	2222444466668888	Finished	15/05/2021	null	2	SOLD	0	0	null	0	0	null

User Table

	cCard	email	name	password	CCV	expDate
	1111222233334444	GuestMail	NULL	guest	741	11-88
	1234567891234567	mvinimariano@hotmail.com	marcos	pass	123	11/21
	2222444466668888	alissonmaster@hotmail.com	Alisson	pass	999	77/33
	4444888866662222	jhonson@hotmail.com	jhonson	pass	852	99/41
▶*	NULL	NULL	NULL	NULL	NULL	NULL

Project Management

As the project is very complex and requires a lot of time and energy, we decided to divide the work into the points that each one has more ease. Marcos worked in the structure and helped in the building part and Alisson worked in the building and helped in the structure. As a team, we created short tasks for each week to try to do our best and complete all the parts of this project.

Research item 1

Explore the testing needs of the finished Xtra-Vision software system and explain how automatic and manual tests are created to address the testing needs of a Java-based object-oriented software project (research and references required). Discuss any testing that you performed while implementing this system.

Manual tests are performed by humans through test cases that have step by step to obtain the expected result. In manual testing, no tool or even an automated script is used to perform the tests.

The Automated tests are performed with the help of tools and frameworks that generate a script with input and output elements that, when within the scope of the test, defines whether it passed or failed.

Unfortunately, there is no right or wrong test to define the best type of test for your programme, but we can see the differences between them to decide which one is better for each case. I will give you some comparison.

The Manual test may not be accurate due to human error while the Automated are more reliable because it is run by tools.

The Automated test is the best choice for repetitive and long-term testing and the Manual is practical when it is necessary to test a few times.

The Manual test is also better for usability tests while the Automated is a better option for stress and load test for user experience.

We performed a Manual test in our programme as the Xtra-Vision is a small scale prototype and it will not have loads of users accessing it at the same time (i.e 100 users at the same

time in different kiosks around the country), so there is no need of really strong stress and load test.

One of the tests that we performed was the connection tests, making sure that the programme has a reliable connection to the database (MySQL), without the database we can't get access to the movies information, users information and transactions. Another test that we made was the usability test, making sure that the user will be able to easily use the programme. The design was created to be as clean and simple as possible because the niche market is really wide and it has to be accessible to all potential customers. All ages, gender, occupation and income can use the Xtra-Vision products.

We also performed Unity and Functional Tests, comparing and confirming that the behaviour and logic of each component are working as planned during the diagram process (Class diagram, Sequence diagram, State diagram and activity diagram) and double-checking if the programme does what it should do.

Research item 2

Dealing with design complexity: Software systems can start off simple and quickly grow to be quite complex and therefore when developing a computer system, the developer will often be dealing with a certain amount of complexity. Explain (in your own words) how modelling techniques manage complexity in the development of computer-based systems. Your answer should include a description of how models can support software developers who are striving to understand the complexity of the systems that they are developing. Be sure to include factors that are typically used to deal with complexity when analysing a system during the UML modelling process (research and references required).

To build a house it is essential to make a plan, with the purpose of thinking about the forms of construction and making estimates of time, the development of software is similar to this process.

To make a good project, a modeling language that has diagrams should be used. These diagrams allow the representation of simple or complex systems under different views, as this facilitates the understanding and standardizes the communication and organization of the problem.

The UML methodology combines different types of models to describe a system: a class diagram that the team can use for checking the set of classes and relationship between classes, a sequence diagram which the developers can see how the different parts of the system interact with each other, a state diagram that represents the behaviour of the system after an event and an activity diagram that presents visually the flow of actions in a system. They are different points of view that can be seen separately but they work together to provide a better understanding of the requirements.

Each of them covers important aspects of the system and they are all necessary for a complete system description as they have an intuitive graphic representation and are useful for communicating with the team and customer.

Sometimes the programs are unsuitable for the company, they don't meet the needs of users or need to solve some recurring problems such as security, load balancing, fault tolerance and distribution. The visual programming languages make the project more productive and easier to implement because the more complex and bigger the system, the greater the probability of errors occurring in case it has been done without any type of modelling, that is why modelling is so important, if the project has no documentation with modeling, the work will be much greater and with a risk of having an unsuccessful system.

The models are the main key for the developers because without a plan it can further complicate the project causing confusion and diverting the attention of the developers' team to details that are not important to the project. Actually, the diagrams help the team analyze the project from different perspectives having a more comprehensive view of the system and develop it more quickly and correctly.

References

- Guru99.com. 2021. *Automation Testing Tutorial: What is Automated Testing?*. [online] Available at: <<https://www.guru99.com/automation-testing.html>> [Accessed 10 May 2021].
- Guru99.com. 2021. *Manual Testing Tutorial: What is, Concepts, Types & Tool*. [online] Available at: <<https://www.guru99.com/manual-testing.html>> [Accessed 10 May 2021].
- Leitner, A., Ciupa, I., Meyer, B. and Howard, M., 2021. [online] Se.inf.ethz.ch. Available at: <http://se.inf.ethz.ch/people/leitner/publications/auto_test_hicss.pdf> [Accessed 10 May 2021].
- OpenLearn. 2021. *An introduction to software development*. [online] Available at: <<https://www.open.edu/openlearn/science-maths-technology/introduction-software-development/content-section-6>> [Accessed 13 May 2021].
- SEI Blog. 2021. *Managing Model Complexity*. [online] Available at: <<https://insights.sei.cmu.edu/blog/managing-model-complexity/>> [Accessed 13 May 2021].
- Visual-paradigm.com. 2021. *Why UML Modeling?*. [online] Available at: <<https://www.visual-paradigm.com/guide/uml-unified-modeling-language/why-uml-modeling/>> [Accessed 13 May 2021].