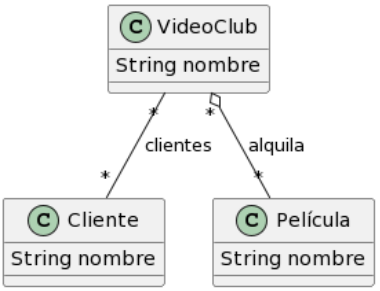
Class Modeling in UML with ChatGPT

1. Introduction

ChatGPT is a bot with which we can converse by entering prompts, to which it will provide a response. A prompt is a small text input that is given to ChatGPT to generate a conversational text response. The prompt can be any type of text input, such as a question, a statement, or even a simple word or phrase. Of course, these phrases can include requests to perform simple tasks (for example, writing a summary of a specific subject).

In our case, we will focus on prompts that lead us to generate a UML class diagram, similar to the ones we have seen in class. For example, to attempt generating a small UML class diagram for a video store like the one shown below, we could write the following prompt:

Prompt	**UML Class Diagram**	**PlantUML**

I would like to write a class diagram in PlantUML where a Video Store rents many movies. The Video Store has many customers, and both the Video Store, movies, and customers should have a name.		`@startuml`
		
		`class VideoClub {`
		`  String nombre`
		`}`
		
		`class Cliente {`
		`  String nombre`
		`}`
		
		`class Película {`
		`  String nombre`
		`}`
		
		`VideoClub "*" -- "*" Cliente : clientes`
		`VideoClub "*" o-- "*" Película : alquila`
		
		`@enduml`

In the above prompt, we can observe that we have added a specific notation called PlantUML instead of simply mentioning UML. This is done for convenience since ChatGPT doesn't generate diagram images directly, but it can generate a textual representation of the diagram, which can be translated into an image using the Plant UML website (https://www.plantuml.com/plantuml/).

It is important to note that the provided prompt can have a significant impact on the response generated by ChatGPT, as the model utilizes the context of the prompt to generate a relevant and coherent response. Therefore, the prompt should be carefully designed to obtain the desired response from the model.

Even when designing the prompt carefully, there may be instances where ChatGPT doesn't generate the expected result. To try to improve it, we can continue the conversation with the bot by providing multiple successive prompts in which we ask to modify one or another

aspect of the obtained result. For example, if it generates a Movie (“Película” in Spanish) class that doesn't include a name attribute, we can instruct it that movies should have a name.

2. Objective

The objective of this activity is to determine the usefulness of ChatGPT in generating class diagrams and to conduct a critical reflection on its capabilities, advantages, and disadvantages in the context of a software development project.

3. Procedure

To conduct this activity, we need to carry out the following steps:

1. Open an account at OpenAI/ChatGPT, following the instructions that can be found at https://chat.openai.com/
2. Go through exercises 3.1 and 3.2, which are detailed below. To do that, you will need to:
    a. Conduct the activities described in the exercises.
    b. Submit the files detailed in the “submission” section for each of the exercises.
3. Answer the questions in the following Google Form: [redacted URL]

3.1. Exercise: Generation from a domain description

In this exercise, we are going to create a class diagram for a problem we formulate, with the help of ChatGPT

Problem description

We want to develop an application to manage the plays that are performed in the theaters of a city.

- A play can be performed in multiple theaters, and multiple plays can be performed in the same theater. The theaters are located in various locations.
- The plays can be operas or musicals and can be performed during the day or at night.
- In each play, one or more performers can participate, who are identified by their name.
- A play is written by one or more authors, who are identified by their name.
- Each play can be performed in one or more sessions, which have a start time and an end time.

To solve this exercise, the steps to follow are as follows:

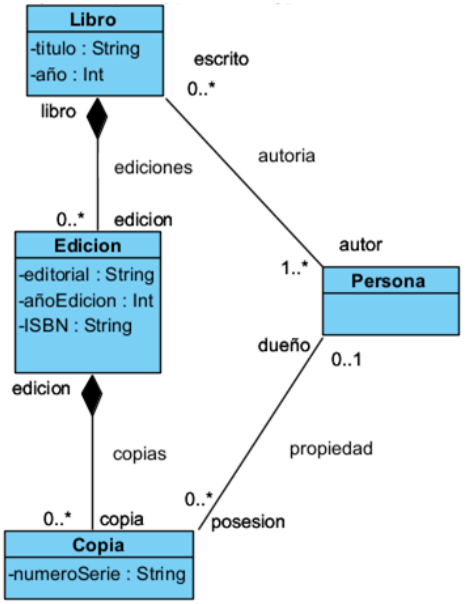
1. Generate a class diagram for the solution through a conversation with ChatGPT (we'll refer to this diagram as D31-ChatGPT). To do this, you'll need to:
    a. Write an initial prompt.
    b. If the result is not as expected, write successive prompts asking ChatGPT to modify the generated diagram to correct possible errors/omissions.
2. If necessary, manually modify the result generated by ChatGPT to correct any issues the diagram may have (we'll refer to this diagram as D31-Final). To do this, you can draw the diagram created by ChatGPT (D31-ChatGPT) in Visual Paradigm and make corrections from there.

Submission: Two PDF documents will be submitted with the following contents at the CV submission point:

1. D31-conversación.pdf: This document will include a transcription of the conversation with ChatGPT for the generation of D31-ChatGPT (simply select the entire conversation, copy it, and paste it into the document). It should also include an image of D31-ChatGPT.
2. D31-reflexión.pdf: This document will include a description of:
    a. Any problems encountered (if any) in D31-ChatGPT.
    b. How the problems were solved to arrive at D31-Final. The description should be accompanied by a reasoned justification for the changes made. It should also include an image of D31-Final.

3.2. Exercise: Generation from a class diagram

In this second exercise, we are going to attempt to generate a class diagram similar to the diagram provided below.



To solve this exercise, the following steps should be followed:

1. Attempt to generate a class diagram similar to the provided diagram through a conversation with ChatGPT. To do this, follow these steps:
    a. Write an initial prompt.
    b. If the result is not as expected, write successive prompts asking ChatGPT to modify the generated diagram to correct any possible errors/omissions in order to approximate the diagram to the one shown above.
2. If necessary (for example, if the obtained diagram differs significantly from the expected diagram), you can start a new conversation from scratch to try to achieve the desired result, using a different prompt if needed.
3. When it is determined that further improvement of the obtained diagram is not possible, consider the final diagram as D32-ChatGPT.

Submission: Two PDF documents will be submitted with the following contents at the CV submission point:

- D32-conversación.pdf: This document will include a transcription of the conversation with ChatGPT for the generation of D32-ChatGPT (simply select the entire conversation, copy it, and paste it into the document). It should also include an image of D32-ChatGPT.
- D32-reflexión.pdf: This document will include a description of the difficulties observed in getting ChatGPT to reach the expected solution. It should also include an image of D32-ChatGPT.