Diseño y arquitectura de software

Grupo 3

Profesor:

Mario Alberto González López

Proyecto Final

Entrega Oficial

Integrantes:

A01173441 Mauricio Martínez Toledo

A01234029 Jazmín Yolistli Santibáñez de la Rosa

18 de junio 2022

Express in 3 to 5 simple sentences the main functional requirements of the system.

1. The system receives 3 movie categories selected by every user when signing up.
2. The system generates a preference key based on the movies categories received.
3. The system matches the user preference key with the IMDB API and the API returns a list of matched movies.
4. The system can sort the movies by rating: ascending or descending.
5. The system returns the list of matched movies for further processing or user visualization.

Main actors of the system

The IMDB API, the system, user or services that process the system's output data.

What can we do with the system?

Obtain a preference key and generate a list of movies associated with it.

What would you ask the stakeholders to clarify your questions about functional requirements? (Write at least 3 questions)

1. Can the user preferences be modified after registration? If so, will the preference key be updated? Is there a limit in the number of preference changes?
2. Will new parameters be later added? (Example: 4 or more categories to generate the key)
3. Will there be preference when choosing one movie over another if their IMDB score is the same? (Tiebreaker)

Create a Use Cases Diagram of the intended functionality.

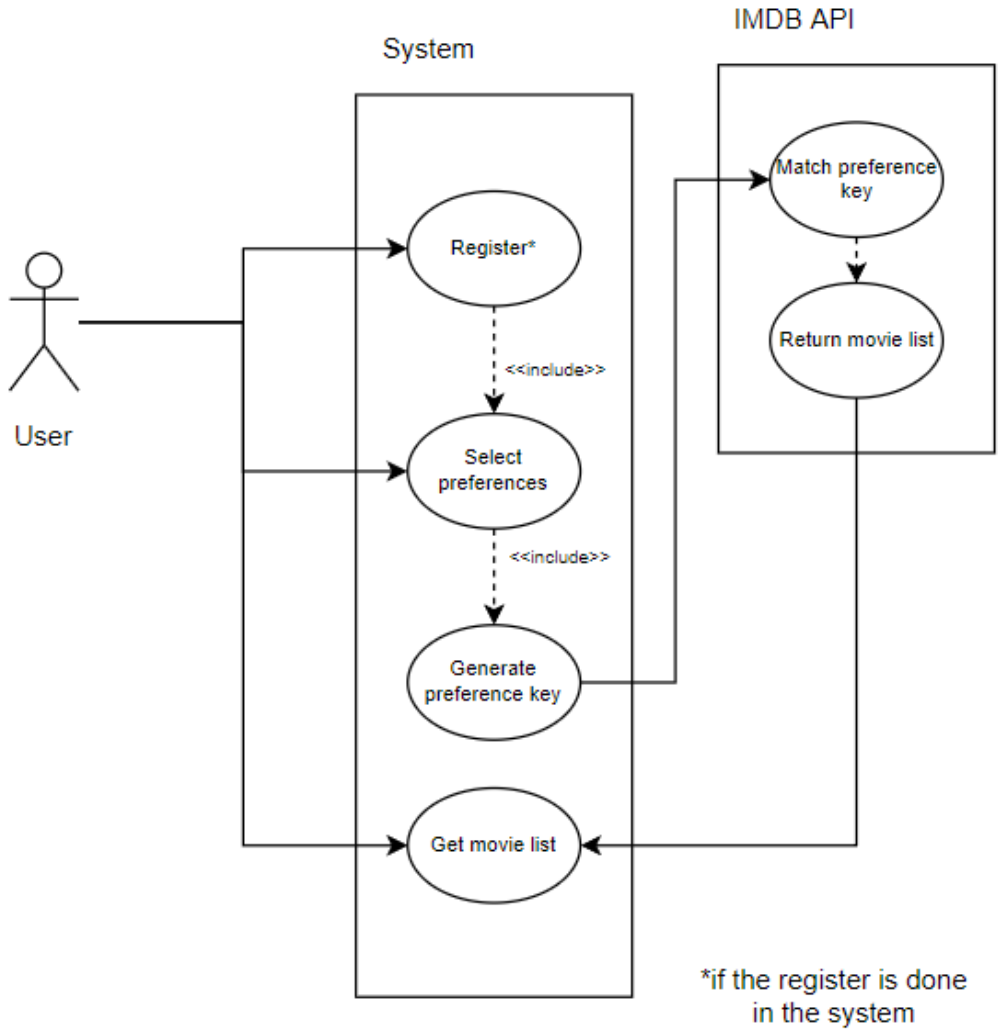

Figure 1. Use cases of the system

Create a Sequence Diagram about a user registration and another for retrieval of movie recommendations for a user.

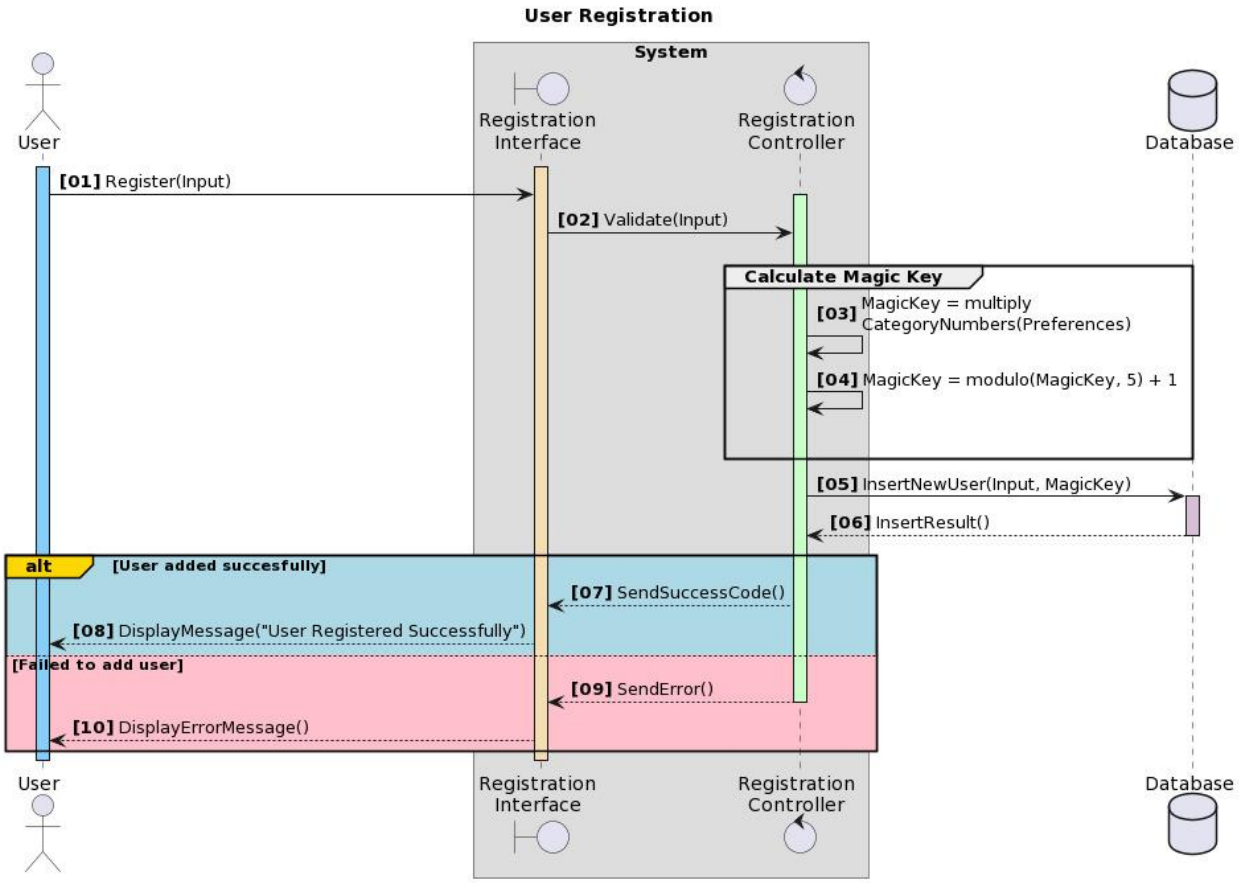

Figure 2. Sequence diagram of user registration

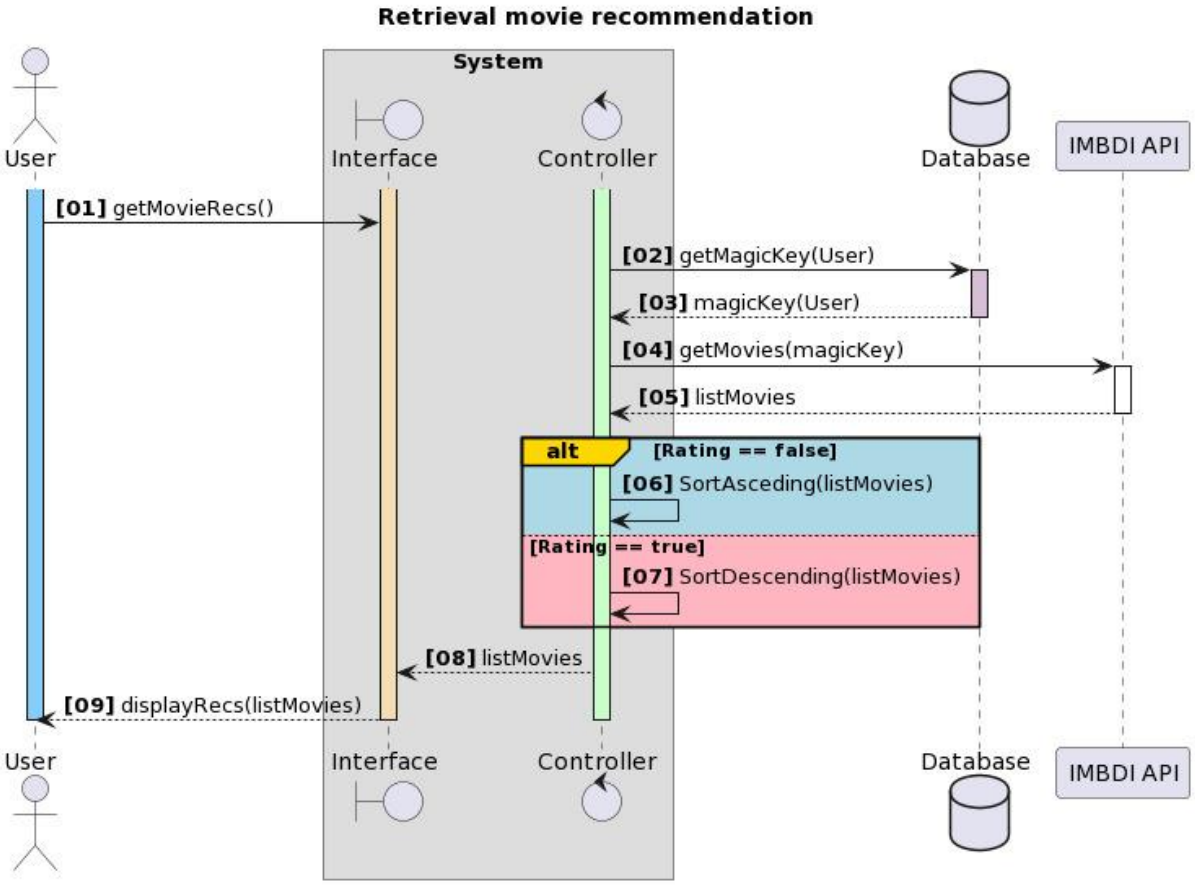

Figure 3. Sequence diagram of retrieval for movie recommendations

Consider that our system needs to be highly available and in the future we need to include new movies and different parameters to update our algorithm

- What non functional requirements should we take into consideration?
    - The processing time must not exceed a certain defined amount of seconds depending on the numbers of possible matches.
    - The database architecture and design must ensure fast and reliable access to data even when it's not the most recent.
    - The type of database used (relational or non-relational) depends on the needs of the system, because of the CAP Theorem.
    - The system must be adaptable and easy to use for integration with other systems.
    - The DB storage should allow an easy expansion