Workshop 1

Henry Ricaurte Mora 20221020084

1 Introducción

Este es un documento de prueba para verificar la correcta compilación de LaTeX.

Components:

- **Mouse**: It is the first input. It is necessary for the functioning of the analyzed system.
- **Hover**, **Click**: Functions that are part of the first input.
- **X**, **Y**: Cartesian location of each of the inputs within the system.
- **Event_handler**: Group responsible for handling each of the individual events of the system.
- **Name**: Name of each event in the system.
- **Text**: Text associated with each event in the system.
- **Session_id**: Unique identifier of the session.
- **User_id**: Unique identifier of the user within the system.
- **Level**: User parameter that characterizes their skill level and/or progress.
- **Login_interface**: Interface that functions as an input to access the system.
- **Auth_system**: Authentication system to determine whether access is granted or denied.
- **Configuration**: Component responsible for managing the game's configurations.
- **Hq**: Identifier related to the visual configuration of the system.
- **Full_screen**: Identifier related to the visual configuration of the system.
- **Option_music**: Component that balances the game's music.
- **Screen**: Specifies characteristics related to the program's magnitude.
- **Audio_system**: External system component that provides the user with specific game audio.
- **Music_room**: Component responsible for the music in the room.
- **Music_game**: Component responsible for the game's music.
- **Id_notebook**: Unique identifier of the notebook.
- **Pages**: Space for making notes, paraphrasing ideas, or building a solution.
- **Event_registry**: Registry of all events in the system.
- **Data_analystic**: Component responsible for generating and managing the system's solution to reach the objective.
- **Performance**: Main output of the system (not feedback-based).

Relationships

Our first relationship is the one related to the **login_interface**, which leads us to an **auth_system** that is responsible for either returning a credential error or performing a login and modeling the system based on a **user_id**.

As a primary relationship, we have the one that generates the input: the **mouse**, which contains two main events: **hover** and **click**, where each of these is mapped using a Cartesian coordinate (**x**, **y**). This first "major" system constantly communicates by sending all information to the **event_handler**, which contains a **name** and a **text** for each of these events.

Continuing with the design of the user interface system, after entering the **user_id**, it obtains a **level**, which is responsible for setting in relation to the **user_id**. This same **user_id** component implements a **configuration** and creates a **session_id**, which is then responsible for sharing that information with the **event_handler**.

Approaching the **configuration** section, we have 3 relationships that obtain state information and share it with the corresponding components: **hq**, **full_screen**, and **option_music**. Both **hq** and **full_screen** send information to the **screen** as output.

Option_music also generates an output to the **audio_system**, and in addition, is able to receive what other components such as **music_room** and **music_game** transmit, and make changes by generating a turn **off** or **on**.

In **user_id** we also handle another relationship: this one gets to **id_notebook**'s component, which contains **pages** that are connected with an event we will manage as **event_handler**.

Event_handler is an important component since it adds everything it receives to another component called **event_registry**, which is in charge of handling all event records. These are then **sent_to** the **data_analystic**'s component, which generates a **performance** of the analyzed system.