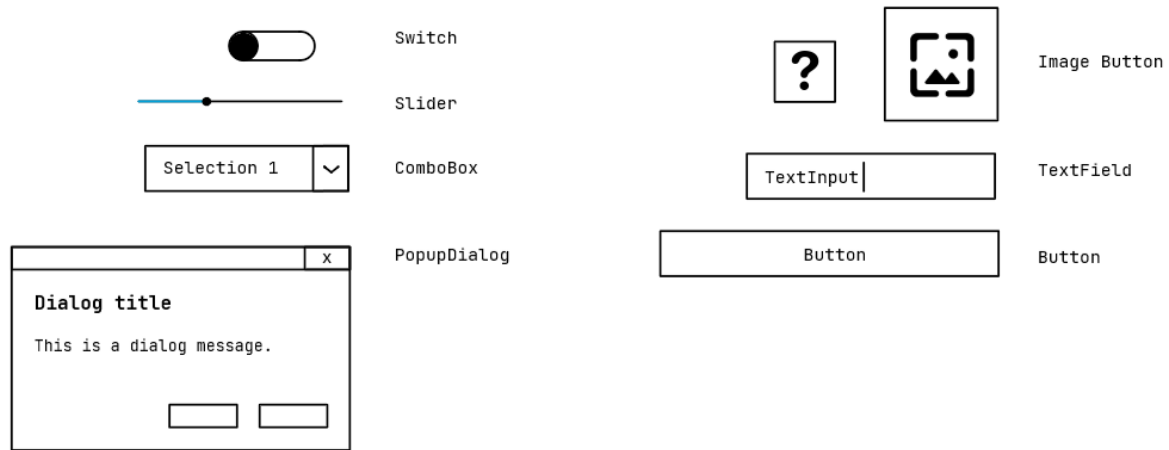


# Marvelous Mashup - Team Milestone 3

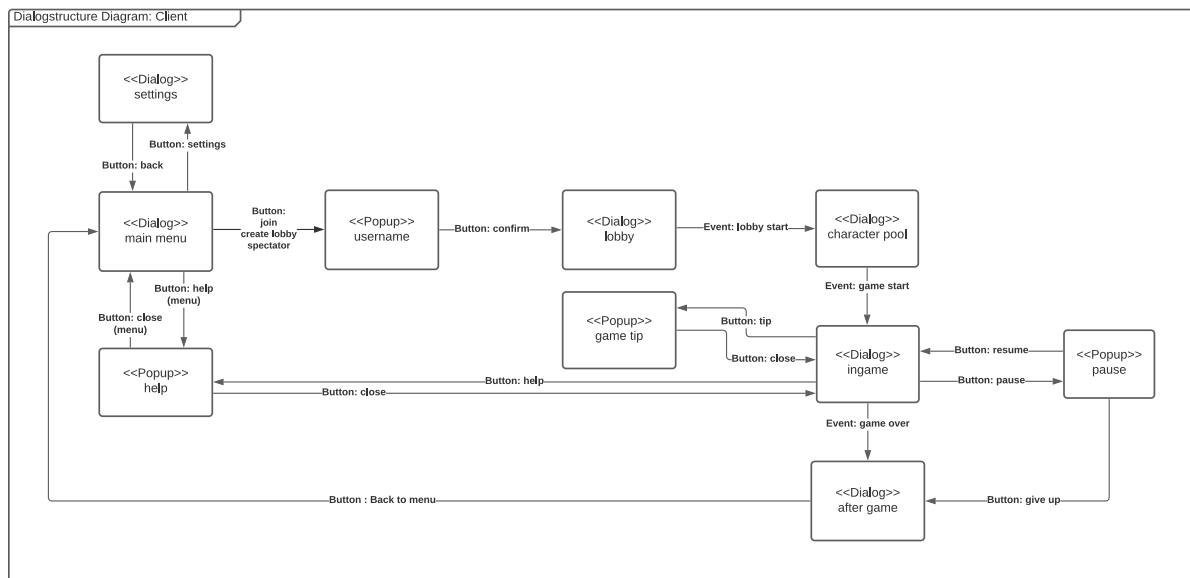
## 1. UI-Element Overview



## 2. Client

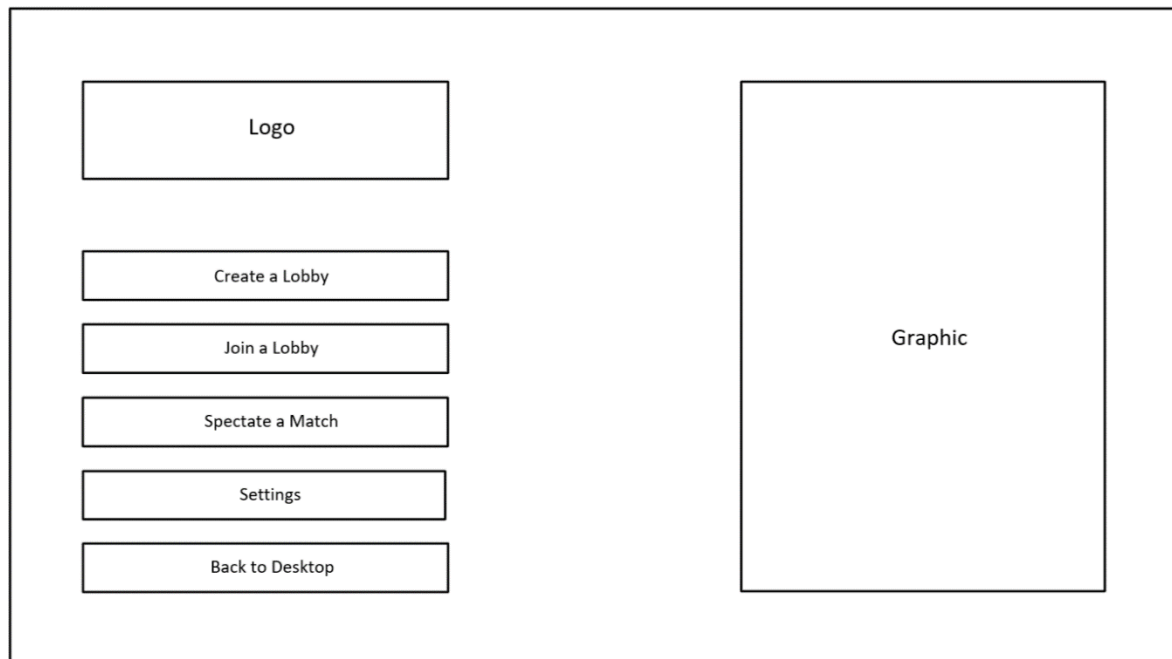
The client component is based on a graphical user interface. This decision was made due to the requirements, which need to be fulfilled. For an easy to overlook and understandable game flow, users are supposed to see the ingame state at every time. In addition to that, navigating within the application is more user friendly by clicking buttons, then console commands which needed to be entered. For the basic design of the client, we developed the following dialog structure (2.1) and mockups (2.2).

### 2.1 Dialog Structure



## 2.2 Mockups

Dialog: Main menu



Dialog Main Menu

UseCase	Manual
Registrate as player	When pressing "Join Lobby" or "Create Lobby" the Client asks for a name, if it does not have one already (Client settings), and if it is "Join Lobby" the lobby code by opening a TextInput popup. After the client contacts the server to create or join the lobby.
Registrate as Spectator	When pressing "Spectate a Match" the client opens a TextInput popup to get the lobby code. After that it connects to the sever as a spectator for this lobby and if the game has not started yet it changes to the Lobby dialog, else it goes to the Ingame dialog.
Change client settings	By pressing "Settings" the client shows the Settings Dialog, where the client's settings can be changed.

Dialog: Settings

Settings

Your username

TextInput

Sound effects

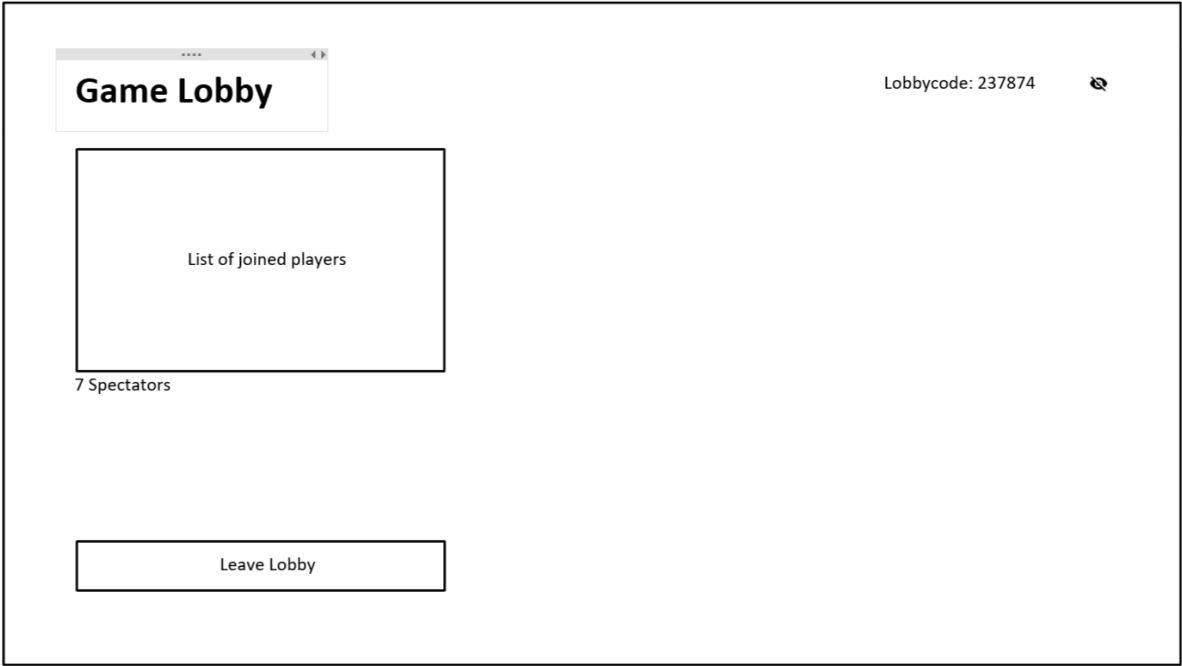
Background music

Back to Mainmenu

Dialog Settings

UseCase	Manual
Change audio settings	Audio settings can be changed by the sliders "Sound effects" und "Background music".
Enter a Username	The username can be chosen by entering it in the Text Field "Your Username"
Set Hotkeys	Hot keys can be chosen by clicking on an action in the list of actions and pressing the specific keys.

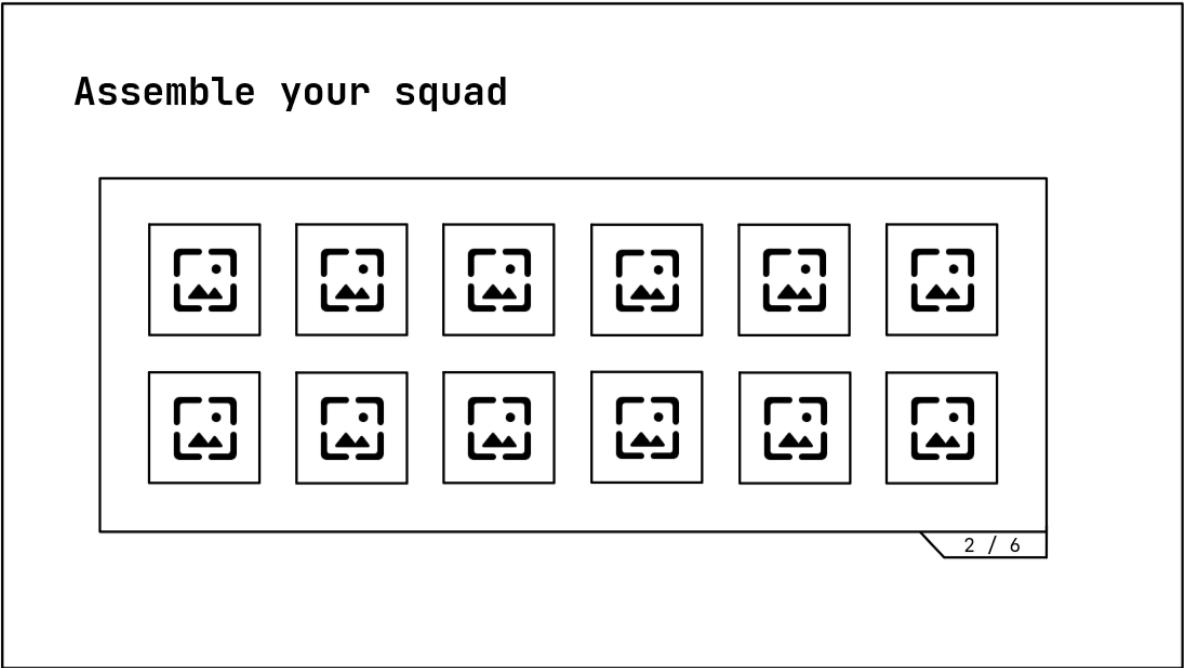
Dialog: Lobby



Dialog Lobby Screen

UseCase	Manual
Player management	Client waits for notification from the server that the game starts, which is sent, when there are two players in the game.

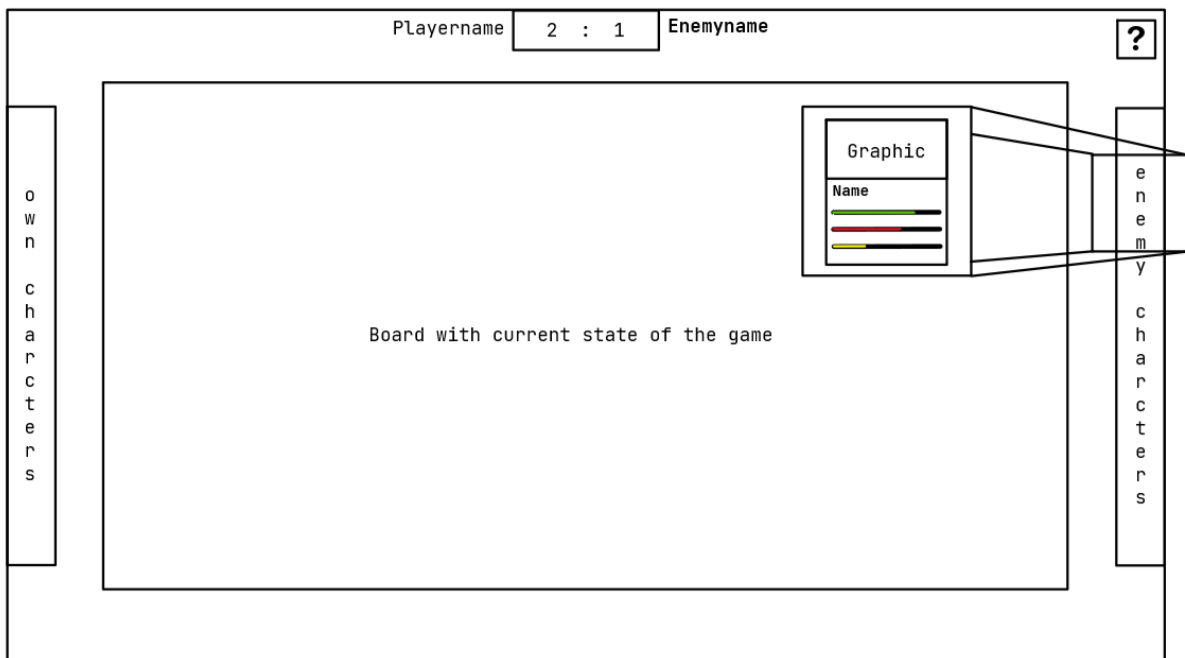
Dialog: Character pool



## Dialog Character Pool

UseCase	Manual
Assemble Superhero Squad	User can select six superheroes from the collection shown for his superhero group. When all six are selected the popup closes automatically.

## Dialog: Ingame



## Dialog Ingame

UseCase	Manual
Request to pause game	The player can request to pause the game by clicking on the button, in the top right, displaying a pause icon. The client then sends a message to the server to request the pause.
Request to resume game	When receiving a message form the server that the game is paused, the client opens the popup. There the client can request to resume the game by pressing the button "Resume game".
Request a tip	The user can request a tip by pressing the "light bulb/hint" button, in the bottom right corner. While looking at the rules, the game is paused.
Look at the rules	The user can look at the rules by pressing the "?" / "help" button, in the bottom right corner. While looking at the rules, the game is paused.
Game round	Shows the user the current state and events of the game and gives him the option to choose his actions, when it is his turn.

Dialog: after game

Statistics

Player 1

XX  
XX  
XX  
XX  
XX  
...

MP  
AP  
Actions  
Knock-Outs  
Attacks  
...

Player 2

XX  
XX  
XX  
XX  
XX  
...

Back to main menu

Dialog After Game

UseCase	Manual
Return to menu	User can return to main menu after finishing looking at the statistics.

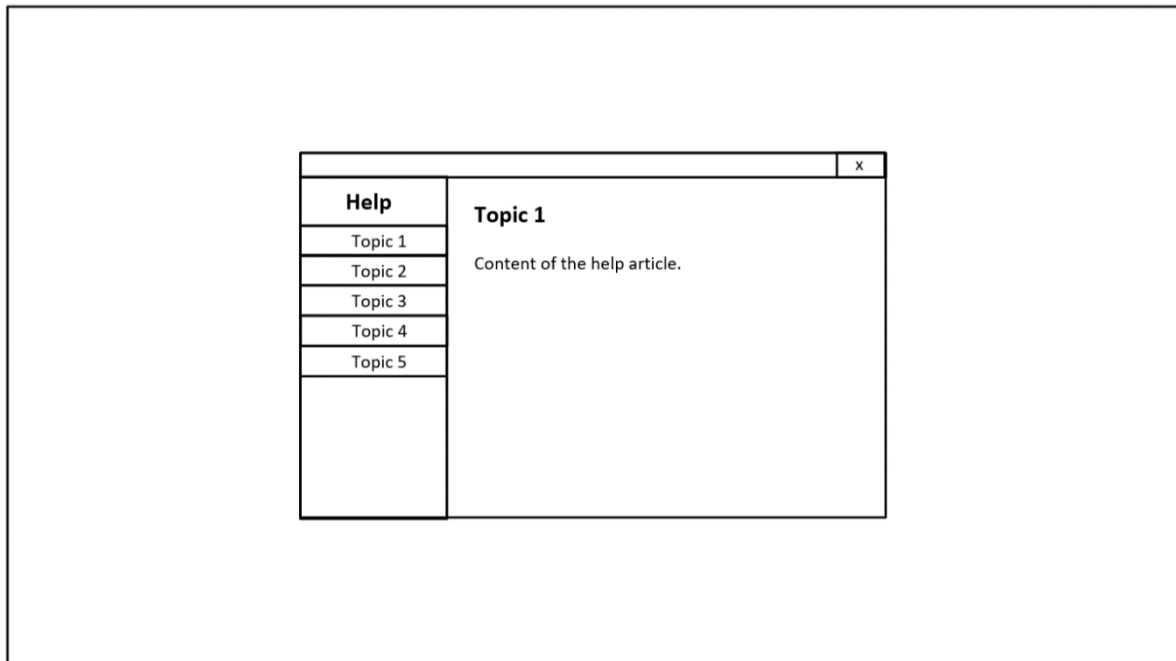
Popup: Pause

Pause

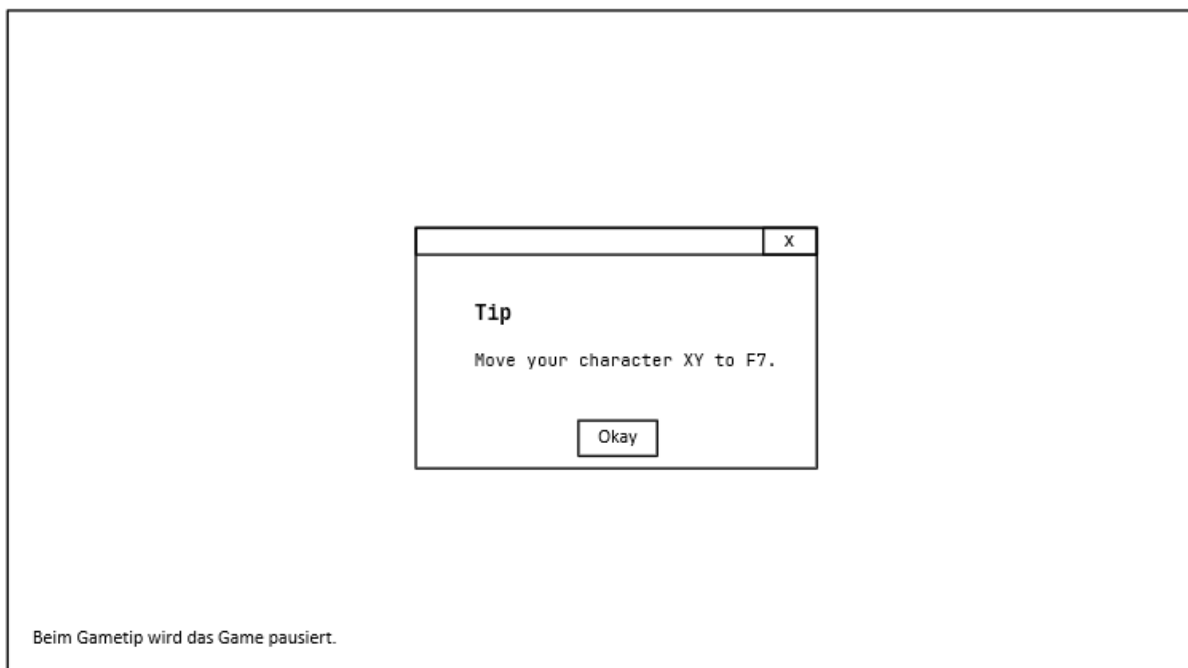
Resume

Give up and return to menu  
(Event: exit)

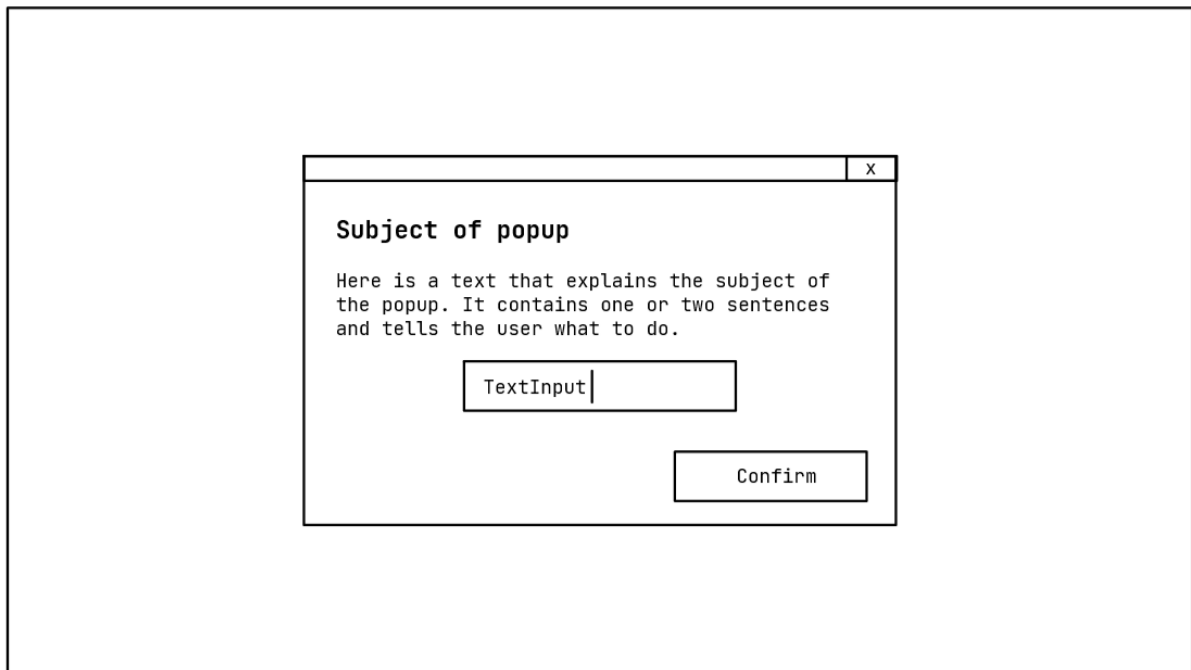
## Popup: Help



## Popup: Gametip



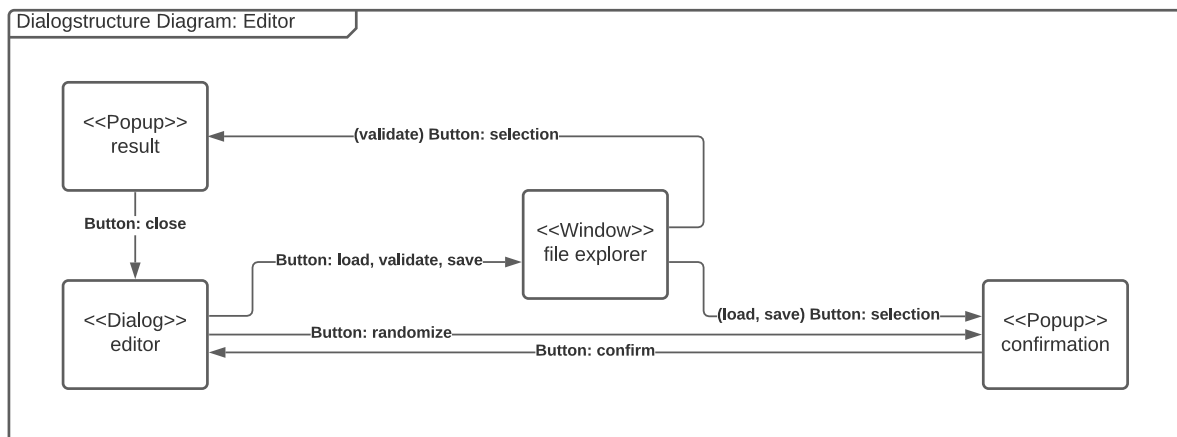
## Popup: TextInput



## 3. Editor

The configuration editor is also based on a graphical user interface. The component is meant to be a standalone app that allows users to create their own levels, characters, and match settings. The design idea is to keep the application simple and easy to understand, which lead us to the following application structure (3.1) and design concepts (3.2).

### 3.1 Dialog Structure






## 3.2 Mockups

### Dialog: Editor

## Editor

Validate a level

Load a level




Save level

---

---

---

Edit leveledata or data from selected objects



Visualisation of the gameboard

### Dialog Editor

UseCase	Manual
Create configuration	<ul style="list-style-type: none"><li>The user can load a level by clicking on the "load a level" button. The explorer will be opened in which the user can choose a level file.</li><li>In the visualization of the gameboard the user can see a preview of the final gameboard and manipulate general things about it. The edit screen to its left is used to fill in more specific details about the level in general and selected objects.</li><li>The user can save the current edited level by clicking on the "save level" button. The explorer will be opened in which he can choose a directory and a name for his file.</li></ul>
Generate random scenario	The user can generate a random level by clicking on the dice button next to "load a level". The values in the editor will be changed to random values.
Validate configuration file	The user can open an explorer window to select a level file that should be validated by clicking on the "validate a level" button. The result is shown in the result popup.

Popup: result



## 4. Server

The server component will be implemented with a console window instead of a graphical user interface. For the correct and complete functionality of the application, commands for querying and updating the various configuration files as well as basic system actions will be provided for use.