

Move (left/right)	
User	System
the user presses and holds the button currently associated with moving in a direction.	
	The position of the representation of the player in the GUI is updated accordingly.
	The POV is updated accordingly
User releases the button.	
	The position of the representation of the player in the GUI stops moving
	The POV remains the same
The user press and hold a button to move, but the player is impaired.	
	The player performs a not possible to move in that direction animation.
Jump/Fall (Up/down)	
User	System
the user presses the button currently associated with jumping.	
	The position of the representation of the player in the GUI is updated accordingly
	the POV is updated if necessary
	The position of the representation of the player in the GUI is reset
	the POV is reset if previously updated
input from the user causes the player to move off the ground or an object	
	The player falls to the ground.
	the POV is updated if necessary.
Respawn	
User	System
The user triggers an event that would place the player at a checkpoint.	

	The player is moved to the correct checkpoint
	The users POV is either updated or set to match the position of the player.
	the state of the player is updated
	The user gains control of the player
Attack	
User	System
the user presses the button currently associated with attacking	
	An animation is played
	an attack is performed
	the state of the player is updated if necessary
input from the user causes the players attack to hit a hostile npc	
	an animation is played
	the state of the hostile npc is updated
	if certain critera are met remove hostile npc from game
	the state of the player is updated if necessary
	the state of the level is updated if necessary
Interact	
User	System
User performs necessary action to interact with a part of the level	
	update state of player or level according to whatever interaction was performed
	update GUI accordingly

Select an item in a menu	
User	System
user selects an item using mouse or keyboard.	
	update state of the game or run function accordingly
	give feedback to the user
Pause/Unpause Game	
User	System
User performs action to pause game	
	The user loses control of the player
	The level's and the player's state are saved and made immutable.
	The GUI is updated to show the "Pause content" and it is enabled
User performs action to unpause game	
	The level's and the player's state are made mutable
	the GUI is updated to show the level and the player
	The user regains control of the player
Play an entire level	
User	System
User starts a new game on a specified level	
	Create necessary objects with correct states
	Update GUI to show Level
	Give user control of the player
User behaves according to the use case move	
	Behave according to the use case Move
User behaves according to the use case jump	

	behave according to the use case jump
User behaves according to the use case move	
	Behave according to the use case Move
User behaves according to the use case Fight	
	Behave according to the use case Fight
User behaves according to the use case move	
	Behave according to the use case Move
User behaves accordnig to the use case Complete level	
	behaves according to the use case Complete level
Complete level	
User	System
user Performs necessary action to complete level	
	An animation is played
	The game's and the player's states are updated
	The GUI is updated to show relevant information and to give the user the option of either returning to the main menu or starting the next level
the user selects an alternative	
	update the program's state and updates GUI to either show the next level or the main menu.
Collide	
User	System
input from user causes the player to touch the ground part of the level	
	Update state of player
input from user causes the player to not touch the ground part of the level	

	update state of player
	(play a falling animation after set time?)
input from user causes the player to touch a damaging object	
	an animation is played
	the player is moved depending on the type of damaging object
	the players state is updated accordingly
	the part of the GUI responsible for showing the players health is updated accordingly
	the user regains control of the player
	if the player has no more health the player respawns, if the game is over, the player Dies.
input from user causes the player to touch a damaging Character	
	an animation is played
	the player is moved depending on the type of damaging character
	the players state is updated accordingly
	the part of the GUI responsible for showing the players health is updated accordingly
	the user regains control of the player
	if the player has no more health the player respawns, if the game is over, the player Dies.
input from user causes the player to touch an object that is interacted with by touching it	
	an animation is played if necessary
	the state of the level is updated if necessary
	the state of the player is updated
input from user causes the player to touch an attack	
	an animation is played
	the player is moved depending on the type of attack
	the players state is updated accordingly

	the part of the GUI responsible for showing the players health is updated accordingly
	the user regains control of the player
	if the player has no more health the player respawns, if the game is over, the player Dies.
Death	
User	System
input from the user causes the player to lose its last health	
	The user loses control of the player
	A death animation is played
	If game is not lost, player is respawned.
Start new game	
User	System
A level is selected and the button Start game is pressed	
	The selected level is started and displayed to the user.
Change key bind	
User	System
A new keybind is set	
	The system start to listen to the new keybind and perform the specified action.
Select profile	
User	System
A profile is selected	
	The system update the currently active profile and change progress and keybinds accordingly