Components:

-Characters

-UI

-Data control

-Interactable

-Specific components

Characters:

Player:

-He must use physics for movement

-He must be able to cast spells

-He must be able to interact with the environment

-He must be able to pick up items

-He must have stats (health, mana)

-We must allow player to save his data to local file (progress, health, mana strength etc…)



Enemy:

-Depending on the enemy type there should be at least something different from the other ones…

-Must be able to detect player

-Must be able to react to player spells

-Must be able to attack

-Must be able to read data from data folder

-Must be able to drop items



NPC:

When speaking about npc’s, they need to have a different type of setup than other characters. Because their main purpose is to interact with other characters. They don’t need to have stats like health or mana and states like attacking or dying. Instead they need to have string collection component, which is used for defining what text will be displayed to character. Also, some npc’s will be used for selling or buying items (shops), so they will need to possess some kind of state that defines what will be displayed to player.

To summarize:

-Used to interact with other characters

-No need for definition of health or mana

-Usually have text that defines what they will say to other characters

-If they are used as marketer, they must have states that will respond to trigger accordingly.



Boss:

Bosses are unique creatures that are usually placed at the end of the level. They have same stats as the enemy and their purpose is the same as the enemy, but they have a different implementation of the combat capabilities.



-UI:

Main menu:

Needs to offer player few choices:

-To adjust options- set the resolution of the game manage sound and inputs.

-To start new or load game

-To exit the game.

Inventory system:

Used by player to manage his items/objects. IS should allow players:

-To swap items positions

-To stack items

-To use items

-To inspect items

Shop:

Shop should hold list of items that can be sold to player.

Items in store needs to have prices, name and description.

There should also be npc that will be used as a trigger for the opening store view.

Data control:

We will create framework that will be used for storing data and scene state. We will use attributes and reflection to detect in which file does data belong. We will also use asymmetric algorithm that will encrypt data.

Interactable objects:

-Traps

-Platforms

-Items

-Trigger cutscenes

-Trigger dialogs

-Etc…

Specific components (for now):

-Spawners