A game using CBR for the honour project

For the honour project I would like to do a single player game using Java. This game consists in two phases. In the first one, the player selects all his spells depending on their characteristics and on his preferences. In the second phase, the player is fighting against an AI in an Arena. This AI is controlled by a CBR. Using his spells, he will have to kill his opponent. In this paper, we want to describe more precisely how the game is working.

Choosing the spells

First, when you want to play to this game you will have to choose your spells. You can select only 4 spells for the match (I'm planning on creating 12 different spells). A 5th spell will be provided at the beginning of the game. This last, as a cost of 0, is doing less damage than the other ones and have no special effect.

A spell is characterized by those attributes:

- **Cost**: Using a spell cost some energy and a player has a limited energy.
- **Shape**: circle, square …
- **On Cast effects**: is applied to the players when the spell is casted on them. It contains the damage inflicted but can also have another special effect, such as:
    - A heal
    - A stun / paralysation
    - Cancellation of another effect
    - A burn (inflict damage on a long period)
    - A slow
- **Constant effects**: For all objects that remains in the arena, it may have the above effects, but they must have other characteristics such as:
    - Permeability: The ability of a player to go through them. For example, it will be 1 if the object is a wall, and 0.5 if the player is just slowed.
    - Slippery: if the object is slippery the player won't be able to stop when he wants and will slide on a given distance.
- **Type**: It can be water, fire, plant, earth, electricity or ice.

A type can be strong against another. Indeed, when a spell is casted on its opposite, it cancels all the effects of the spells. Here is a table that indicates which type is strong against the others:

Is cancelled > Cancel	Water	Fire	Plants	Earth	Electricity	Ice
Water		C		C		
Fire			C			C
Plants	C				C	
Earth		C			C	
Electricity	C					C
Ice			C	C		

C means cancellation. If there is nothing, then, the effects of the spell are applied as usual. For example, with the table above we can deduce that the water is cancelled by plants and electricity.

The player will choose his spells without seeing the opponent ones. The opponent (the IA) will have different pre-set spells and depending on the given set the AI will have a different behaviour.

In the Arena

The player will face an AI player (or more) and try to win using his spells. The player will trigger the spells with different keys and will place them using his mouse. When a player wants to move, he needs to click to the position where he wants his personage to move.

Because the arena may contain obstacles, the personage will have to take the optimal path to go to the indicated position. So, an algorithm should be able to find this path as quickly as possible. This can be realised with an A* algorithm.

The game finish when one of the two players sees his HP fall to 0 or after 1min if none of the two players are dead.

The AI player will be managed by a CBR algorithm. The CBR will implement all the phases of the R4 cycle: Retrieve, Reuse, Revise and Retain, so, it should be able to progress even when it is defeated.

For the game design, we want to keep it simple. This game will be composed by rectangles and circles. Only the colour will change depending on the type of the spell. It shouldn't be more complicated than that: