1. **Game Description**

At the start of this level our player stands at the mountain artillery while no one near him. During the starting of the game, the player may change the default configurations of the control keys to rather more comfortable ones. But by default the game is shipped with keys and control configurations that more or less are common to most 3D games. These game control keys are described in the table below.

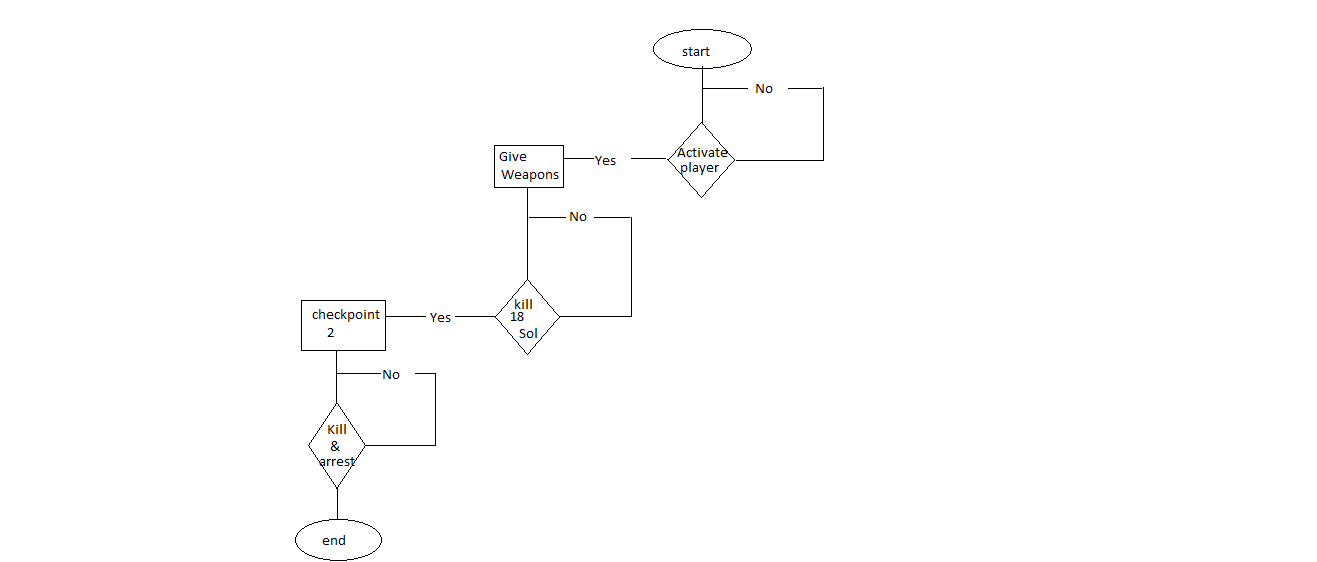
The table pairs a single key function to two different keys as primary key and secondary key. This keys have no technical difference in operation. They just provide an alternative configuration for the player.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | Name | Primary key | Secondary key | function |
| 1 | MOVE\_FORWARD | W | UP ↑ | Move the character forward (jog) |
| 2 | MOVE\_RIGHT | D | RIGHT → | Move the character right |
| 3 | MOVE\_LEFT | A | LEFT ← | Move the character left |
| 4 | MOVE\_BACK | S | DOWN ↓ | Move the character back |
| 5 | SPRINT | shift | - | Run faster when combined with MOVE\_FORWARD |
| 6 | PERFORM\_ACTION | E | - | To perform different set of actions like reading a letter, drinking water. This key can only be effective when the player is near the specific location of the task to be performed.  eg. To drink water the player needs to be near the lake or a river and a text will appear at one corner of the screen notifying the player to press the PERFORM\_ACTION to do drink water. |
| 7 | CROUCH | C | - | To duck and hide from enemy |
| 8 | SWITCH\_WEAPON | Q | SCROLL | To switch between weapons |
| 9 | FIRE1 | LCLICK | - | To attack enemy soldiers (swing sword, shoot shot gun or thrown spear) |
| 10 | JUMP | SPACE | - | Jump |
| 11 | RAISE\_SHEILD | R | - | Raise shield to defend oneself. This reduces the damage done by enemy strike to the player |

After getting in to the level when the player starts to move, change of state occurs and the player gets activated. When he gets activated and as described earlier the animated notification is invoked telling him the objective and the privileges of the level describing that he has to kill all the soldiers as they appear in front of him in pairs or three. This will clearly move him in to the level then this point is saved as the last check point. Then Albertoni’s brigades start to appear in pairs and three from random direction of the mountain to attack him. As it is described earlier at this level of the game there will be eighteen solders which come in groups of three or two and two with albertoni guarding him.

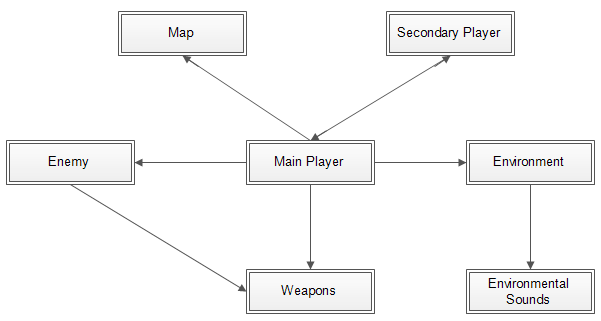
As it is being described earlier fighting the enemies continues until these eighteen soldiers are killed these point is then saved as last check point. If the player got killed before reaching this point the game will start from the previous last check point. Then at the point albertoni with his guards appear and the player should carefully attack only the guards not albertoni because the objective of this level is to arrest Albertoni.

**Game flow diagram**



1. **Architectural Design**

2.1**Graphical Representation**



2.2 **Description of Architecture Views**

* + Main player: is the main character in the game that is controlled by the user

in this level the player will find themselves set up on a mountain with a mountain artillery.

* + Enemy: are the opposing soldiers (Italian soldiers) in the game. They are activated only when the player is in a certain range from them. Then they come on to the player and start attacking.

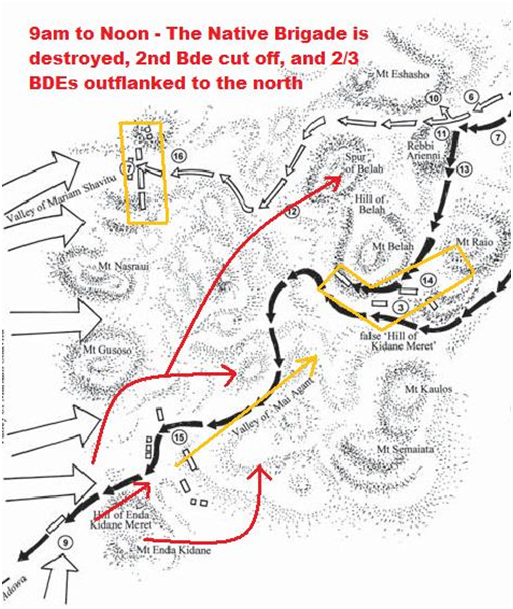
The enemy's will be the brigade that is controlled by Albertone's.

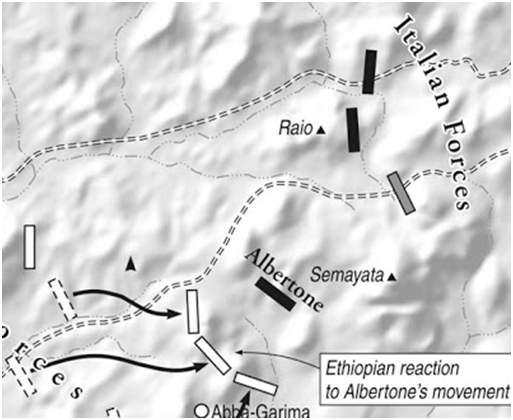
Albertone's askari brigadewill attack in different set ups like they will come up,three or two groups.

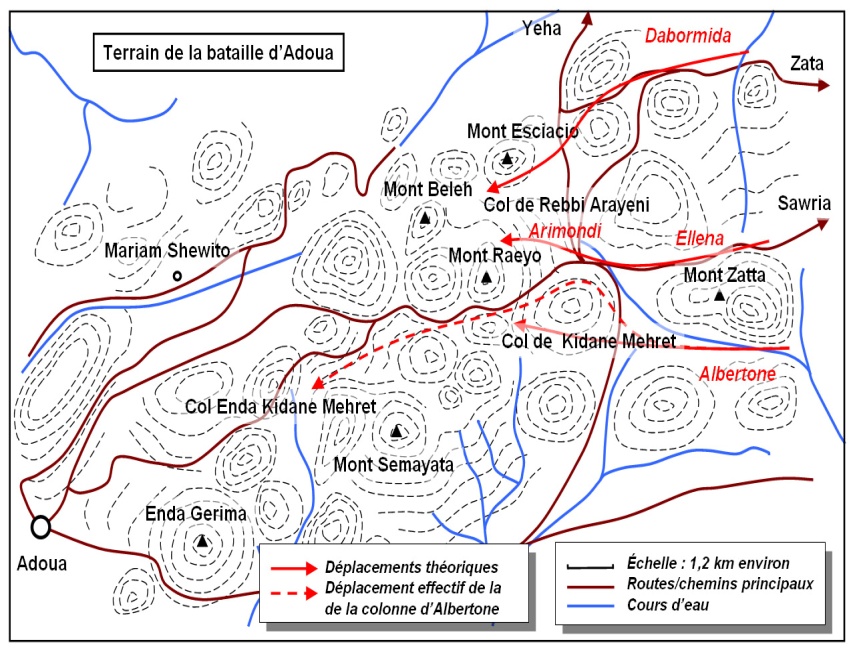
Weapons: are objects guns, **the player will use Hotchkiss**

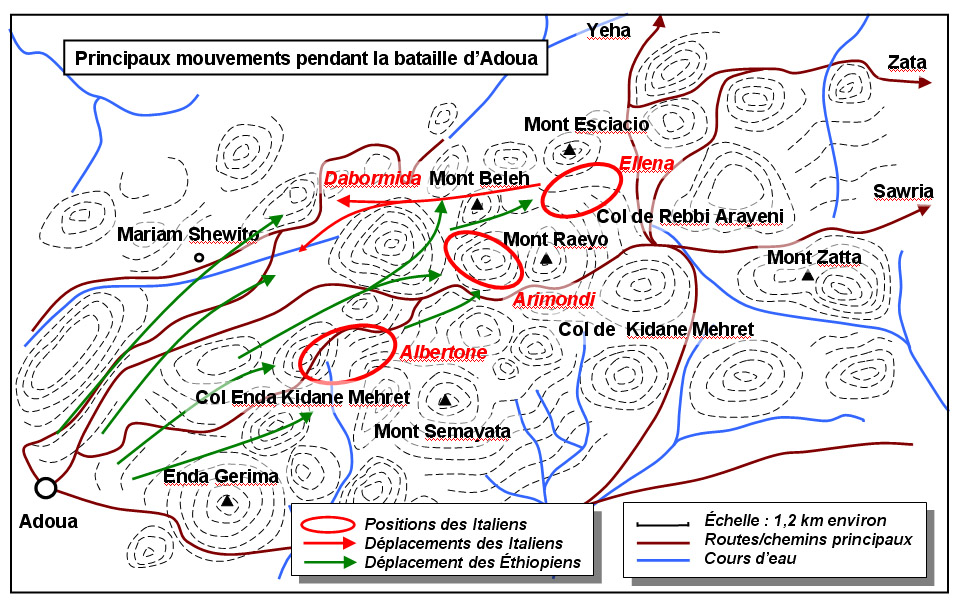


* + **Environment:** is the location and its surrounding for the level 4 action to take place. There will be mountain in, which the Ethiopian's (Ras Alula's position) one mountain for KidaneMeret,then, Italian brigades (Albertone's askari brigade) was the first to encounter the onrush of Ethiopians at 6:00, near KidaneMeret, where the Ethiopians had managed to set up their mountain artillery. The places where the fight occurred between the Ethiopian and Italian armies is designed to be fictional. The environment is going to be built using the following maps as a reference.









* + **Environmental Sounds**:- Is the different background sounds heard during game play. The different sounds are played in accordance of the player location and actions.

1. Since the whole environment is in a desert area, desert wind sounds are played throughout the level.
2. When the player is in action:
   1. When the player is against the players.
   2. When the player is hit by a bullet
   3. When the player fires the enemy soldiers
   4. When the player fires a weapon
3. When other objects are struck:
   1. When the player hits the soldiers with or without a weapon
   2. When the player hits non moving objects
4. When the player and he’s fellow soldiers communicate
   1. When they discuss strategies
   2. ሽለላሲያካሂዱ！
5. And other sounds:
   1. When the player walks, jog’s or jumps
   2. Sound effect
   * Enemies: There are a total number of 20 enemies in this level. These enemies will run towards the player and his fellow warriors.
   * Map: provides information to the user where the players location is.

3.3 **Interaction between views**

In the previous section it’s discussed about the views of the architecture. Their interaction with each other is:

* + The main player: is controlled by the user it will need an interaction from the other views. Rather it will activate and control the other views. For example when the player moves from place to place, the environment will change accordingly. The environment will change unless the player moves. The main player will act as resource for the other views when to be activated or changed. This main player is the controller for the other objects in the game. Its only when the player goes to the objects that the objects are activated and interacting.
  + Enemy: is activated when the player gets close to him/her. The enemy then advances and attacks the main player. Then as shown in game mechanics, the enemy will go on and execute the tasks coded to it. Which are to check if it’s killed and deactivate, if not to keep advancing to the player and attack again.
  + Weapons: are the objects in the game play. These objects do not act by themselves. They are accessories for the player and the soldiers. If they are dropped, they are inactive unless the player or a soldier picks them up.
  + Environment: as the player moves around in the map, the environment is continuously updated to ensure the user feels that the players is moving to the desired locations.
  + Environmental Sounds: are called by the environmental view. As discussed previously when the player moves around and the environment is continuously updated, the sound's also updates to simulate the environment more accurately.
  + The secondary player: this secondary player is first awakened when the main player finds the place where he is captive and frees him. Then he will follow the main player through all the battles to reach safety grounds. This is done by continuously updating the main player’s location and tracking him. And thus will require a continuous request for the location coordinates.
  + Map: this object is interfaced with the main player that it can provide accurate information to the user. It gets the current location of the player and continuously updates throughout the game play.

**HIGH LEVEL AND DETAILED DESIGN**

**Environment**

Since the in environment holds every component in these game the duty of checking the position and activating enemies is given to it.

So it becomes

1. Check positon change in player //first move of player (for the first time only)
2. Activate notification
3. Check if notification is destroyed
4. If destroyed save this point as last checkpoint.
5. While (1) // always do this as loop
6. Check the distance between enemies’ and the player

If in range activate enemies

1. Check the distance between player and sound source

If in range activate the sound source

1. Check the number of remaining enemies
2. If none

10. Display end of level notification

11. Terminate the level.

**Enemies**

1. while (1)
2. check player position
3. Approach player position
4. If in long range
5. Fire to the player // to the player position
6. ElseIf in short range
7. Perform combat action
8. If fired to self
9. Degrade health status
10. Destroy.

**Map**

1. While **(1)**
2. Fixmap to the top wright corner
3. Track player
4. Plot the player
5. Track the enemies
6. Plot enemies