Level 5

5.1. Game description

The game in this level is based on the fight between the Arimondi Birgade, that the fleeing Albertone soldiers sought refuge at, and the Ethiopian mountain artillery soldiers that are later joined by the 25,000 shewans from Minilik.

5.1.1. Design goals

The design of this level aims to achieve the following goals.

1. Introduce Arimondi and the Shewan forces of Minilik.

2. Serve as a continuity to Level 4, which our player chases to Arimondi's location.

3. Develop an achievement system that rewards the player with \*\*\*\*

4. Develop a level that keeps its accountability to the actual History while stilll providing entertainment.

5. Have art and design that is in sync with the previous level as they contain the same landscape settings.

5.2. Game Mechanics (Graphics Perspective)

5.2.1. Core game play

Happenings in this level

* Player gets notified of objective for this specific level.
* Player fights with Arimondi's soldiers.
* Positional and camera change following the fleeing of Albertone soldiers.
* Player kills all the soldiers coming their way.
* The player needs to stay alive until Menelik released his reserve of 25,000 [Shewans](https://en.wikipedia.org/wiki/Shewa) and swamped the Italian defenders.
* A war against Arimondi's brigade vs. the Shewans led by Minilik.

HOW THE PLAYER PLAYS THE GAME

This is the fifth level of the game; in this level the player plays the game using various game play controls and interaction controls mentioned below

* Movement control to move around while attacking or defending.
* Sprinting control if a player needs to run.
* Attacking and defending whenever player is near enemy.
* Action key to switch between weapons if a player gets another weapon (if a player killed an enemy and acquire the weapon).
* Jump using jumping control if a player needs to jump while attacking and defending.
* Use a defending shield to protect from an enemy attack.
* Crouch using crouching control if a player needs to crouch while attacking and defending.

5.2.2. Game flow diagram

NO

Am Inotified?

Yes

Follow Albertone’s Soldiers

Keep Following

Do I see an enemy?

NO

attack

Am I dead?

YES

Any enemy

Remained?

Finish

No

Yes

Follow and attack

Am I notified?

When the game starts just like it has happened in the other levels of the game there is a notification area where the player is going to notified of the aim of the level and continue as arranged.

In this level of the game the notification is the player should be able to survive the war for at least 3 minutes until the shewans force consisting of 25000 troops swamp the area and end the level war.

Am I dead?

The player inside of the war is checked in time whether he is alive or not in order to continue the war in which case if the player fails to survive for 3 minutes he is going to be inactive unless he is going to get a better health timer and move to the next level of the game.

Any enemy Remained?

The player always follows the map which helps him move to a place where he could get his enemies and attack them.

He should always check the remained enemies before going to the next level of the game if it finds that there is no enemy it will finish the levels objectives and go to the next level of the game.

5.2.3. Game physics and statistics

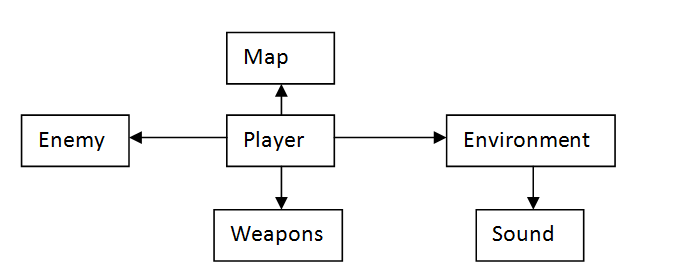
The physics and statistics on this level are described as follows

1. Movement: this is a basic action of moving around the game.
2. Jogging: this is a basic action of jogging around the arena. This enables the player to attack fast and makes sure that the player doesn’t become an easy target.
3. Jumping: this is a basic movement to jump over obstacles.
4. Sprint: this action enables the player to sprint forward fast to attack or hide. This makes the player and even harder target.
5. Swing: This is an attacking mechanism when the player has a sword as a primary weapon
6. Throwing: This is an attacking mechanism when the player has a spear as a primary weapon
7. Shooting: This is also an attacking mechanism when the player has a shot gun as a primary weapon
8. Blocking/Defending: this is a defense mechanism of raising a shield. The character may raise its shield to defend itself. This reduces the impact of being hit by an enemy
9. Crouching: This is a hiding mechanism that may be helpful when sneaking behind enemy lines or when trying to dodge enemy bullets

5.3. Architectural Design

* + Graphical Representation

As every other level, this level is divided to the following modules.



Description of Architecture Views

The modules seen on the architectural view above is the representation of the different modules in this specific level. It describes each module separately and the relations between the modules.

Let's begin with the map.

5.3.1. The Map

There are two 'maps' we are concerned about. One's the internal map that the game engine operates on and there's the external map which is the representation of this internal map on the player's screen.

The internal map will be developed based on the terrain map shown on the next page. The external map that the player sees is shown at one corner of the screen and the mockup of it is also shown on the next page.

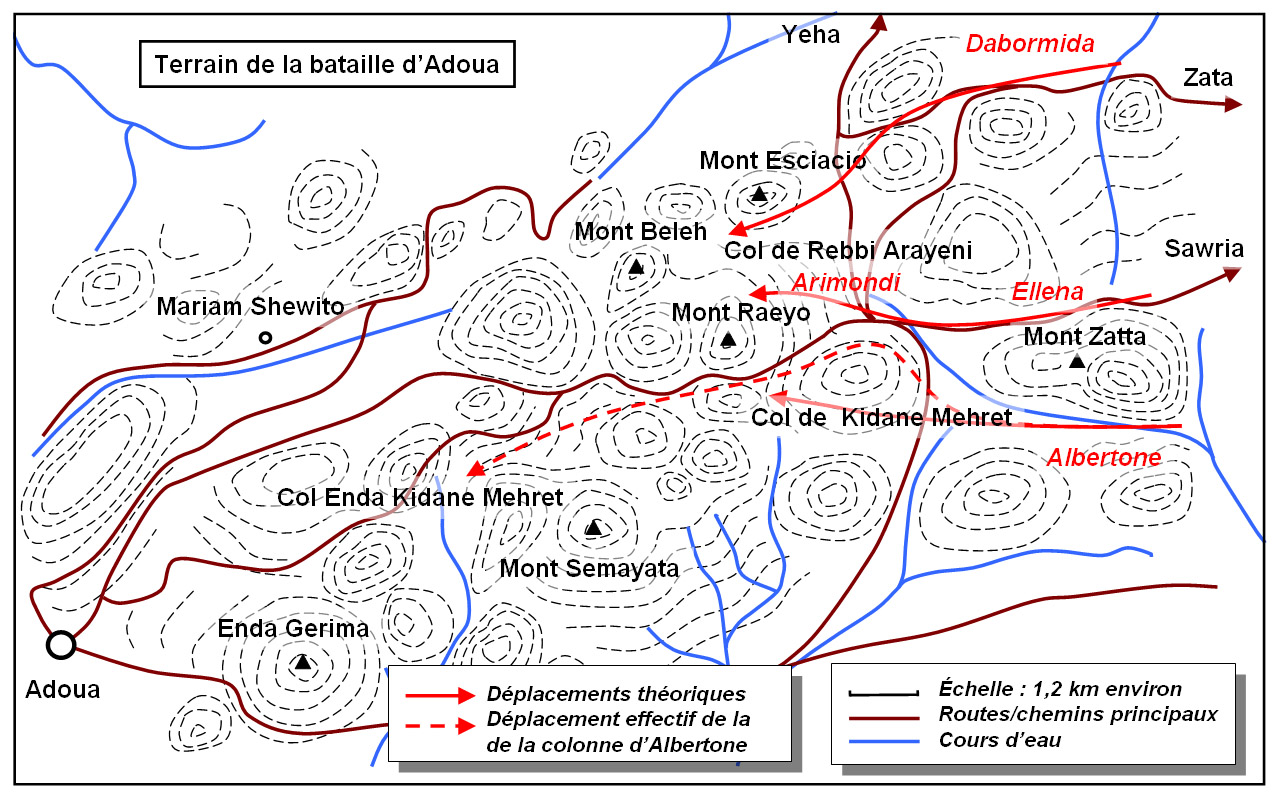


Fig: Internal map of the game for Levels 4, 5, 6 and 7

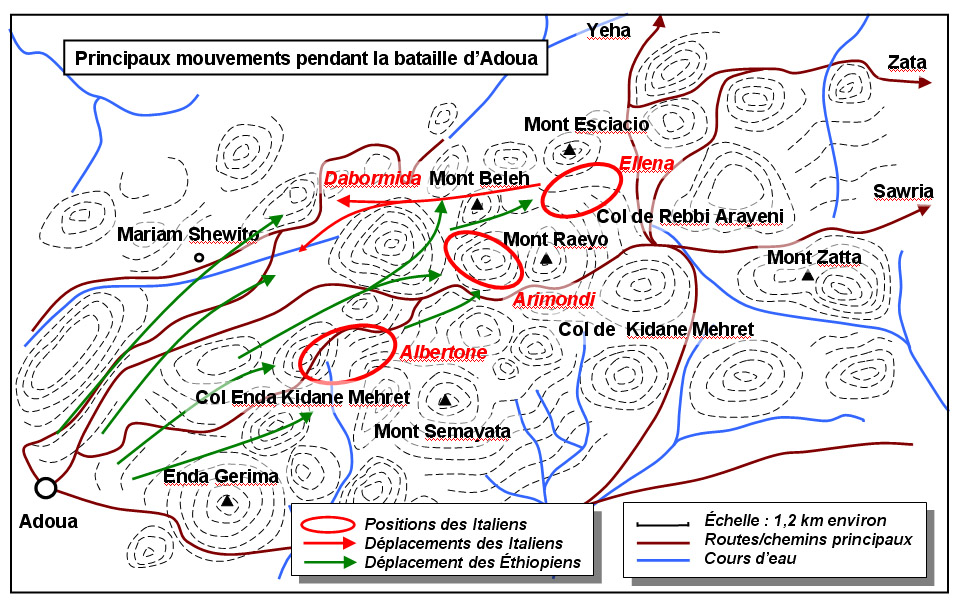


Fig: Internal map showing Ethiopian positions and Italian advances

5.3.2. Enemy

The enemy soldiers are depicted in the game in a grey uniform with a bell shaped hat. They are depicted in the picture below.



Fig: Italian soldiers

The enemy soldiers interact with the player individually. Their scripts are activated when player approaches them. This is shown in the picture on the next page.

* Player: depicted with a black star.
* Enemy soldiers: depicted in red arrows pointed in the player's direction.
* When R=D, soldiers get activated.
* When looking at an enemy, yell “Zeraf”.
* When shooting, activate rifle sound.

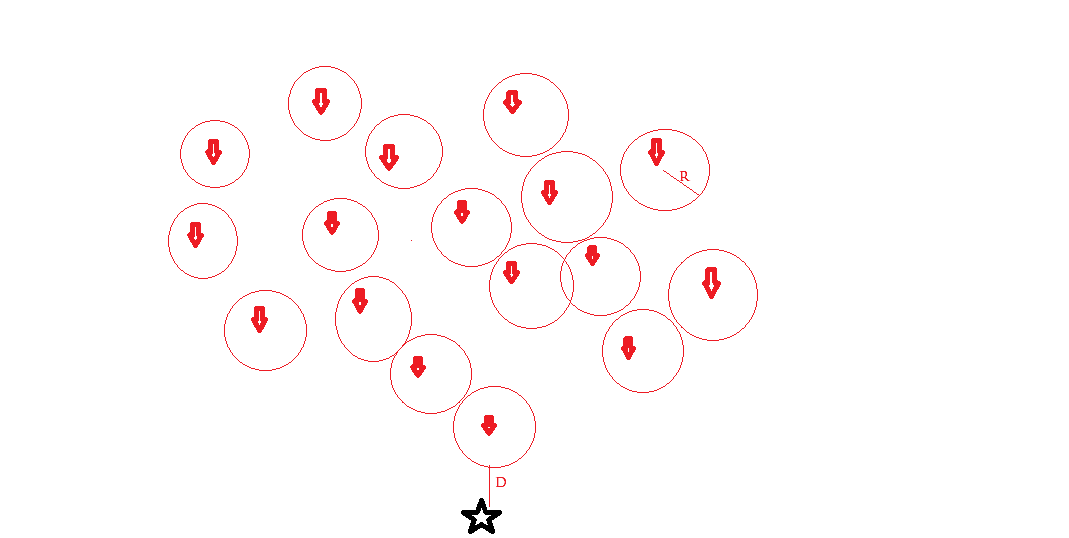


Fig: Interaction of player with enemy soldiers

5.3.3. Environment:

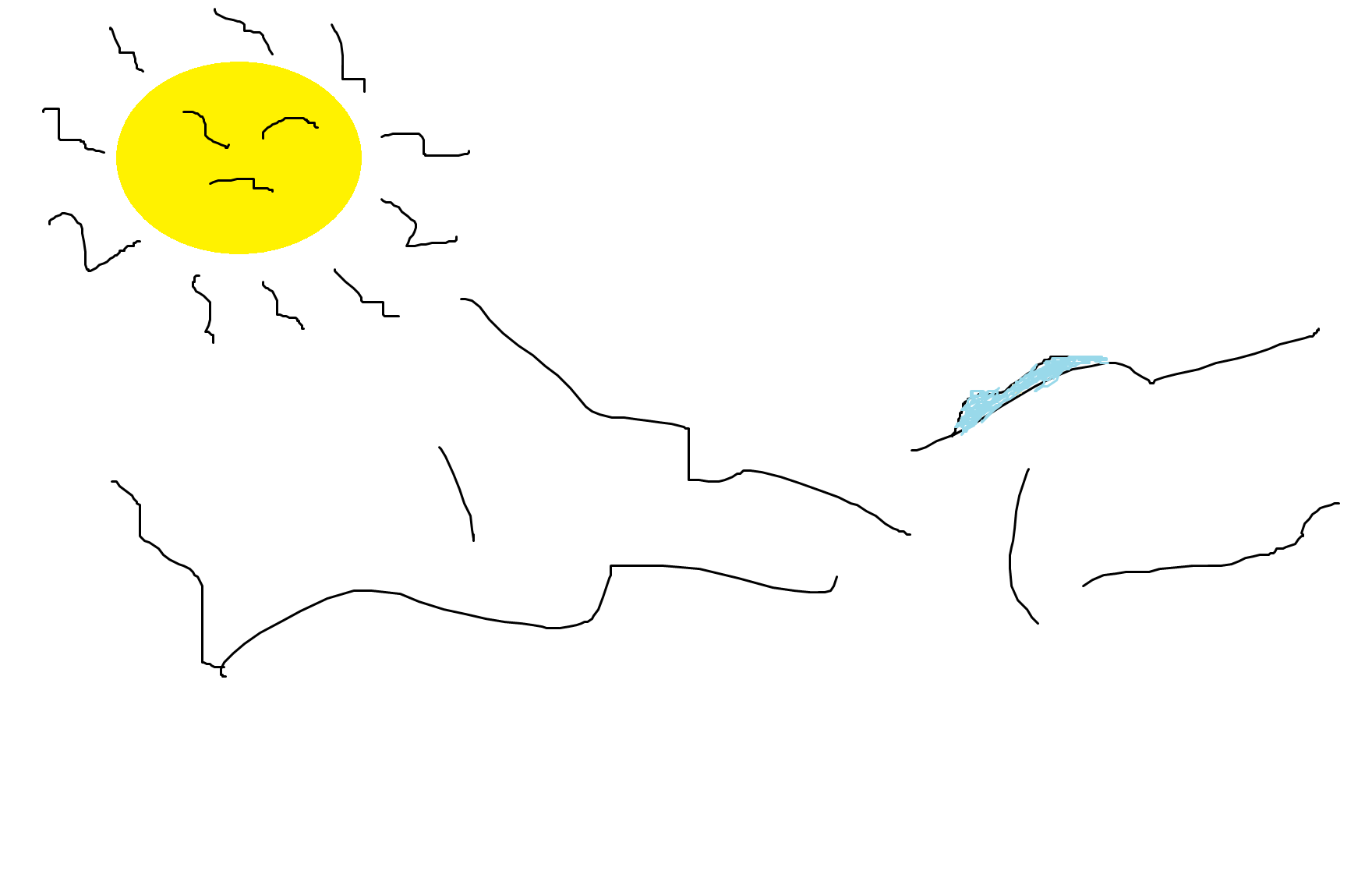


Fig 1. The environments of that time



* The Environment that we assumed is just to look like the rural areas of ancient Ethiopia and also since Albertone's askari brigade was the first to encounter the onrush of Ethiopians at 6:00 we assumed the war to happen in the middle of the day where the sun is above the head.

* The player of the game is ancient time minilik soldier who have battled so many internal and external wars and also who is also someone with so much loss of power and strength but taking a final measurement in order to free himself and his country.
* The enemies were soldiers with so much support and strength but lucking the courage and tactic to achieve the goal.

5.3.4. Weapons

The weapon being used in this level is a rifle and will be used against the soldiers as they approach. The weapons and their interaction with the player module is explained broadly in the detailed design section.

5.3.5. Interaction between views

We could see all the three views in the game but now we would see the component and connector view.

Component and Connector view:

Components: Computational elements. Or Elements that have active role in the system.

Connectors: Means of interaction between components.

Regarding this type of view our level has main components of the player and the enemy.

**Our enemy**

**Our Player**

Artillery Artillery

Our player uses supports he gets from his environment in order to survive and win the game.

The player follows the albertone’s fleeing soldiers in order to reach the arimond’s brigade which he finally helps him achieve the aim of the game and finish.

Since the environment is mountain and the enemy forces don’t have any idea about the environment our player uses this advantage to enable him survive and get the support of Minilik's soldiers which enables him finally swamp the area and kill arimondi.

5.4. High level and Detailed design

LOGIC FUNCTIONS

Player

Most of the player’s script is implemented as an event triggered actions and this events are taken as key stroke from keyboard and what the player wants to do is given to the player with this functionality.

1. While (1)
2. If action from keyboard == “w”
3. Player moves forward;
4. else If action from keyboard == “s”
5. Player moves backward;
6. else If action from keyboard == “a”
7. Player moves left;
8. else If action from keyboard == “d”
9. Player moves right;
10. else If action from keyboard == “space”
11. Player jumps;
12. else If action from keyboard == “LShift”
13. Player advances faster in the direction the player is facing;
14. While (1) /// for the attack combat
15. If action from keyboard == “LClick”
16. Player attacks nearby enemy;
17. If weapon selected is shoot able
18. Shoot the weapon in a direction a player is facing
19. Else if swing the weapon selected in a direction player is facing.
20. ElseIf action from keyboard == “Capslock&& LClick”
21. Zoom in at the direction player is facing and fire;
22. If weapon selected is shoot able
23. Shoot the weapon in a direction a player is facing
24. Else//weapon is not shoot able
25. weapon cannot zoom
26. If action from keyboard == “ x”
27. Player throws spear if the selected weapon is spear;
28. Else selected weapon cannot be thrown; select throw able weapon
29. If action from keyboard == “r”
30. Player bocks an attack by an enemy;
31. If action from keyboard == “c”
32. Player crouch irrespective of the direction faced;

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 position 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. Else If in short range
7. Perform combat action
8. If fired to self
9. Degrade health status
10. Destroy.

Map

1. While (1)
2. Fix map to the top right corner
3. Track player
4. Plot the player
5. Track the enemies
6. Plot enemies