# MicroWar

*Final Prototype Chapter*

*Team 4*

*David Niggli, Marcel Lüdi, Gaetano Paganini, Jonas Krucher*

## 1. Game description

### The survival of the fittest

Our Game tackles the very basics of Evolution: genetic mutations.
Players face each other on a map, provided each with a DNA sequence. There are different types of resources, which are used to code the DNA, flowing through the map.
The player’s genetic code is translated periodically to produce various proteins which perform different tasks: gather resources, attack enemy proteins, attack the enemy DNA etc.
Players mutate and extend their genetic code, in order to produce the desired proteins.
A player wins by destroying the enemy’s DNA.

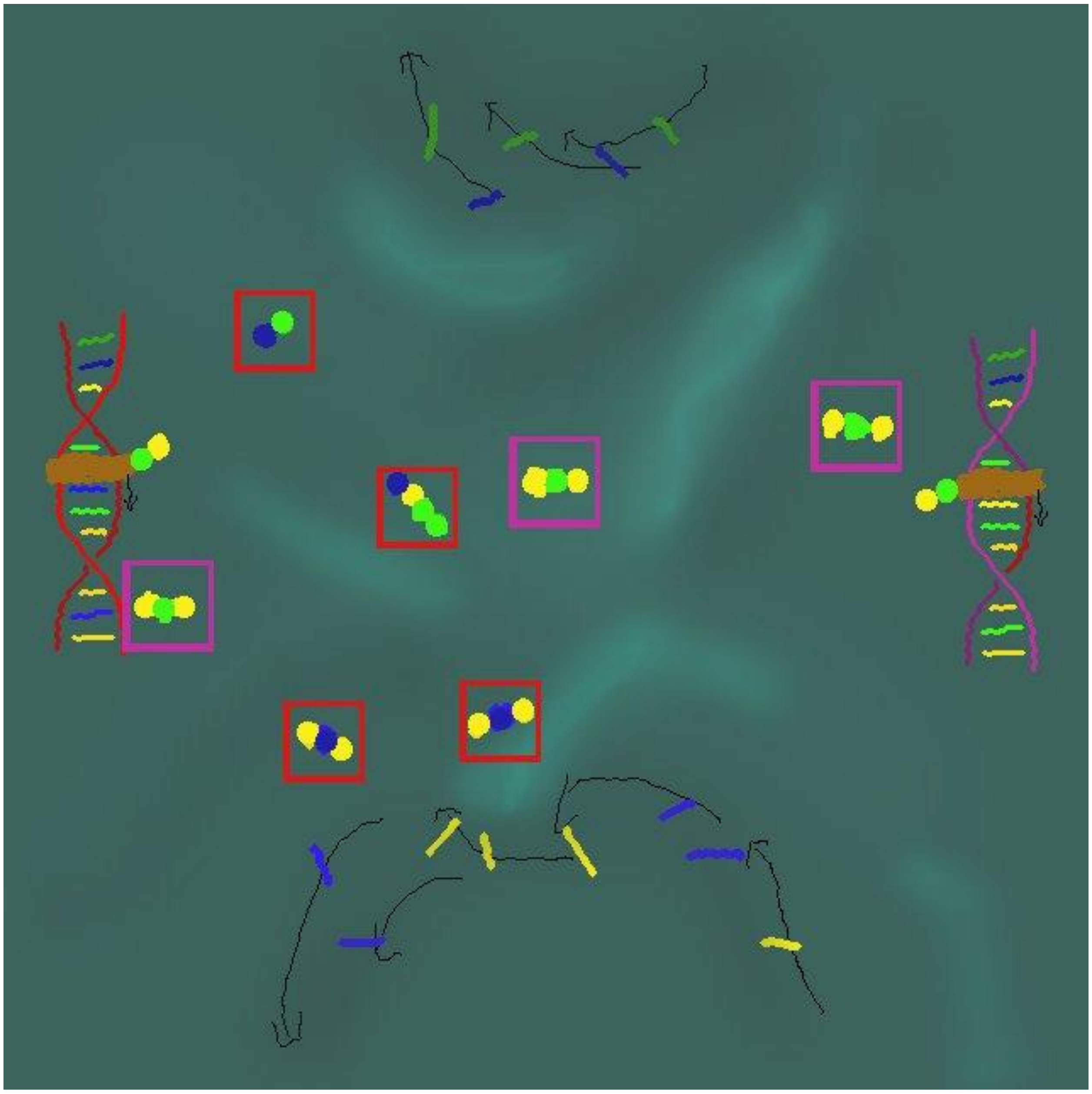


*Figure 1: The first mockup of our game, which shows a DNA on each side. Also the resources are visible at the top and the bottom. Further are proteins on the picture, they are gathering or attacking.*

## Map

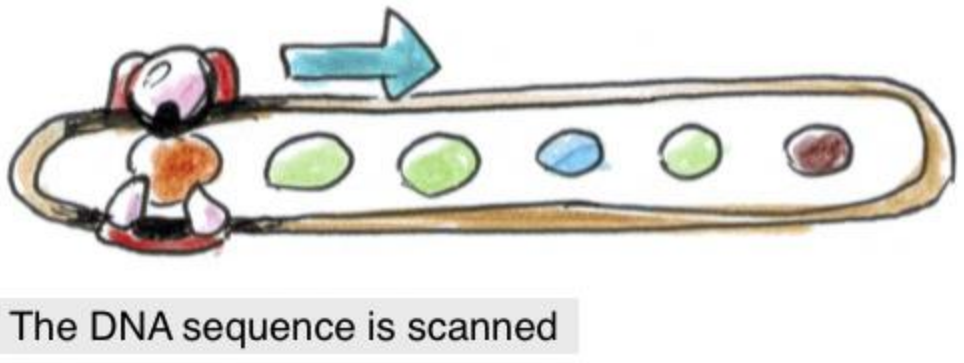
The game takes place on a map similar to the ones found in typical Real Time Strategy games. The two players each have their DNA sequence as their base, located at each side of the map. Resources flow through the map, which also contains obstacles.

## Objective

The main objective of the game is to destroy the enemy’s DNA, while defending your own.
On the way there, players want to modify the DNA sequences according to their needs.

## Mechanics

Unlike in traditional Real Time Strategy games, where players can control individual units, MicroWar only allows to indirectly control the game flow by letting the player interact with a code sequence that generates his units, which find their paths according to a deterministic algorithm.
The DNA is a sequence of slots, which can contain any of the available resources.
To insert a different resource into a slot, the player needs to spend that specific resource. The previously contained resource is lost. Extending the DNA size costs quite more resources than changing a DNA slot.
The players are provided with a legend that explains which sequences produce which type of proteins. Different protein types perform different tasks: gather resources, attack enemy proteins, attack the enemy DNA etc.
Proteins do not only have a type, but can also be enhanced in several attributes, like speed or lifepoints, through variations in the sequence that codes them (ie via pre- or suffixes).
The sequence is scanned periodically to produce the proteins. Proteins are spawned next to the base and start their behaviour according to their type.



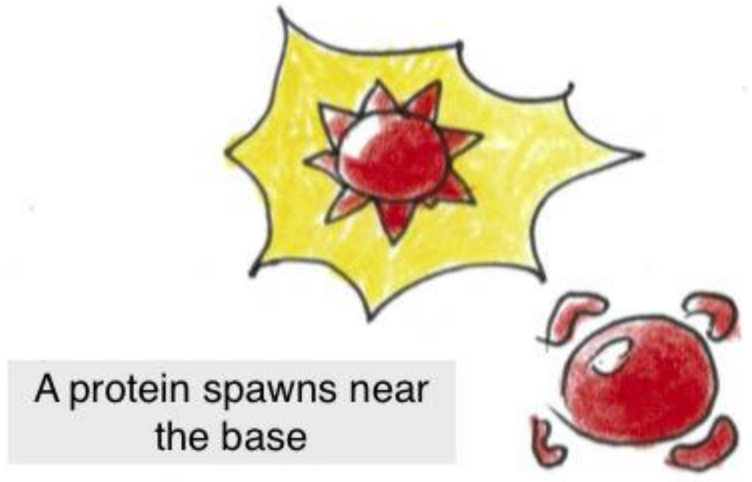


*Figure 2: This pictures shows, how some elements of the game should look like.*

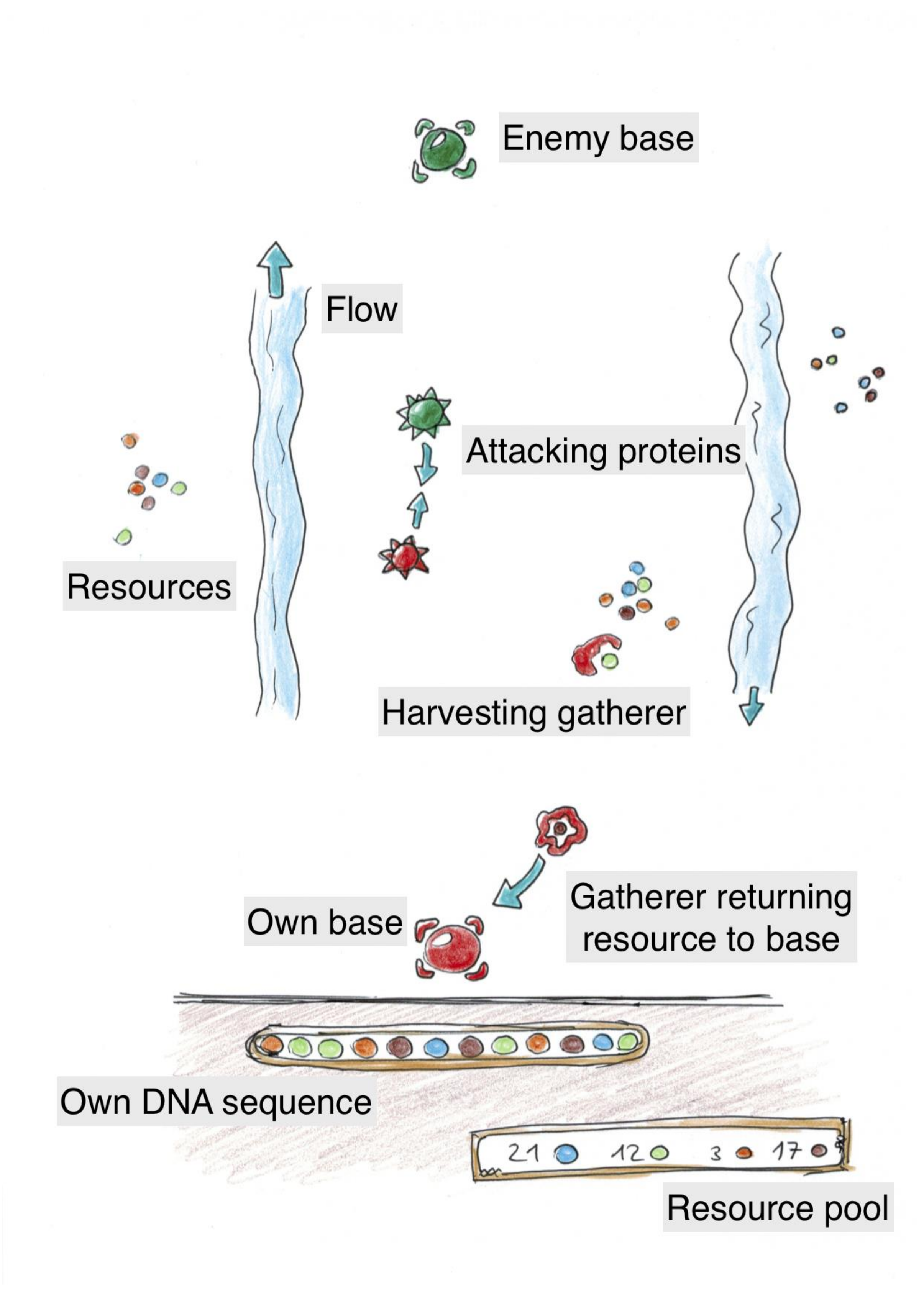


*Figure 3: This picture is a more accurate visualization and description of the whole game.*

## 2. Technical achievement

In a first step we would like to implement procedurally generated maps to diversify the playing experience. This includes a heightmap, placement of resources and placement of obstacles.

For the proteins a ‘smart’ path finding AI needs to be built.

If time allows the diffusion of the resources can be simulated using an appropriate water simulation. Otherwise we can just implement a velocity field on the map which simulates the flow of the water.

## 3. “Big idea” Bullseye

### Big idea

Mutate your DNA in order to survive as the fittest.

### Technical innovation

Smart self-controlling unit behaviour on procedurally generated maps.

## 4. Development schedule

*Functional minimum*

- 2D rectangular map
- one limited resource type on the map
- two types of proteins, a gatherer and an attacker
- a limited number of AI attacker proteins on the map
- switch button to toggle between the two types of proteins
- periodical generation of the selected protein
- naive algorithm that determines protein behavior (walk towards closest resource, when reached it is gathered; walk toward closest enemy entity, when reached it is killed)
- the player wins when all resources have been gathered
- the player loses if the AI destroys his DNA
- implemented for an easy win: a winning strategy is to make attacker proteins first, which kill and get killed by AI units, then to creates gatherer proteins to gather the

resources. A losing strategy is to generate gatherer proteins first, which get killed by the AI, which then also attacked the DNA sequence itself.

*Low target*

- multiplayer (2 players)
- four respawning resource types on the map
- a fixed size DNA sequence which can be modified by the player (costing the gathered resources)
- better algorithm that determines protein behavior (walk towards closest resource, bring it back to the DNA. If the unit is killed while bringing it back, the resource is lost; walk toward closest enemy entity, attack it)
- the player wins when the enemy DNA is destroyed
- the player loses if his own DNA gets destroyed

*Desirable target*

- extendable DNA sequence (costs a bit more/or other resource)
- some lifepoint system (protein lifetime)
- some more units: DNA-attacker vs protein-attacker; ranged attack protein; per resource type gatherer; area damage, healer etc
- resources flow through the map along some paths
- sound
- obstacles on the map (and more)
- procedural maps

*High target*

- 3D rendering (still 2D game logic)
- DNA translation “scanning” animation
- even more units / building-like units
- resource: fluid simulation
- more modularity for units (proteins)

*Extras*

- more than 2 players
- players have cells instead of just one DNA. Those can split and multiply
- the DNA creates viruses that can dock on the enemies cells but not on their own
- animations like attacking proteins devouring enemies like macrophages
- fog of war

| Week | Task | Assigned to |
|---|---|---|
| 1<br>(3.3.15 - 10.3.15) | **Final proposal** | All |
| 2<br>(11.3.15 - 17.3.15) | **Physical prototype / adaptations of idea** | All |
| 3<br>(17.3.15 - 24.3.15) | **Functional minimum:**<br>- Simple graphic placeholders (map&unit textures)<br>- naive pathfinding algorithm<br>- randomly placed resources<br>- check for victory (all resources gathered)<br>- Button to select proteins and periodical generation | <br>Jonas<br>Dave<br>Marcel<br>Gaetano<br>Dave/Jonas |
| 4<br>(25.3.15 - 31.3.15) | **Low targets**<br>- multiplayer (2 players)<br>- refined AI<br>- four types of respawning resources<br>- winning condition: enemy DNA destroyed<br>- editable DNA and protein generation | <br>Dave/Jonas<br>Dave/Marcel<br>Marcel/Gaetano<br>Gaetano<br>Marcel/Jonas |
| 5<br>(1.4.15 - 7.4.15) | **Desirable targets**<br>- extendable DNA<br>- flowing resources<br>- obstacles<br>- sound | <br>Jonas/Marcel<br>Dave/Jonas<br>Gaetano<br>Marcel |
| 6<br>(8.4.15 - 14.4.15) | **Desirable targets**<br>- sound<br>- more proteins<br>- procedural map generation<br>**Interim Report** | <br>Marcel<br>Jonas/Gaetano<br>Marcel/Dave<br>All |
| 7<br>(15.4.15 - 21.4.15) | **High targets**<br>- DNA translation “scanning” animation<br>- graphic polishing<br>- even more units / building-like units<br>- more modularity for units (proteins) | <br>Jonas<br>Jonas<br>Marcel/Gaetano<br>Dave/Gaetano |
| 8<br>(22.4.15 - 28.4.15) | **High targets**<br>- resource: fluid simulation<br>- 3D rendering (still 2D game logic) | <br>All<br>All |
| 9<br>(29.4.15 - 5.5.15) | **Alpha Release** | All |
| 10<br>(6.5.15 - 12.5.15) | **Playtesting** | All |
| 11 | **Public Presentation + Conclusion: Preparation** | All |

| | | |
|---|---|---|
| (13.5.15 - 19.5.15) | | |
| 12<br>(20.5.15 - 26.5.15) | **Public Presentation + Conclusion** | All |

*Table 1: The schedule of the game MicroWar.*

## 5. Assessment

The strengths of the game are the indirect control and the simple but challenging gameplay. It can be learnt fast but is interesting to master.
The game can easily be expanded by new units or upgrades the players can make.
Players can think of new strategies and build orders which can make the game different, depending on whom you play against.
It also can be just fun to watch the AI control and move the units.
Since there are flows which can move the resources and an enemy which can change behaviour at any point in the game, the players have to adapt their strategies throughout the game, which is an interesting challenge and a fight to successfully lead your DNA to victory.