TECHNISCHE UNIVERSITÄT MÜNCHEN

Computer Graphics and Visualization Group
Computer Games Laboratory WS 16/17

Pandemia Munich

Authors: Adrian Philipp, Clemens Kamm, Markus Siglreithmaier
Date: January 25, 2017

Contents

1. Game Proposal

1.1. Game Description

This year's topic of the Computer Games Laboratory course is "Munich", so we decided to make a game that focuses on the infrastructure of that city. The core idea is to fight a deadly virus, which starts spreading in Munich. To accomplish the goal of exterminating the virus before too much of the population has been infected, the player needs to use limited resources in real-time.

The focus of our game is to simulate the spread of the virus as accurately as possible. To achieve this we plan to implement a realistic simulation of the daily life of the inhabitants of Munich. The graphics will be rather simple so that the player can easily keep track of the simulation. We are planning to visualize the city in bird's eye view. Neutral areas are displayed in white while areas with a high infection rate are colored red.

1.1.1. Gameplay

In the beginning of the game, the player doesn't know anything about the virus, but there are already infected inhabitants. It takes some time, until symptoms appear and infected persons can be detected. Using a variety of measures like test stations at public places, this process can be accelerated. To reduce the spread of the virus, the player can take several actions. These actions can affect the whole city, e.g. improving hygiene rules, starting an information campaign or enforcing a curfew, a single block or a special place, e.g. Marienplatz or München Hauptbahnhof. It is also possible to restrict the dispersal of the virus via public transport by closing lines.

As restricting the spread of the virus is not sufficient to extinct it, we plan to integrate the possibility for the player to research medicines and vaccines. The more patients have already been examined, the faster the research progresses. Also, it will be possible to upgrade research facilities and hospitals to improve healing chances, make healing go faster, be able to accommodate more patients, as well as to increase their area of influence.

In addition to fighting the virus, the player must ensure that the popularity is sufficiently satisfied. A core concept of the game is to weigh the effectiveness of a player's actions against their effect on satisfaction. For instance, an information campaign has little consequences on satisfaction, but is less effective than a curfew, which on the other hand has a big impact on satisfaction. The players need also to get by with a fixed income of money.

To keep track with what is happening, we implement a event system, which notifies players of important incidents. Examples for such events are first identifications of infected persons

and first events of death in Munich or a specific district. Notifications are also triggered, when infection rate or death rate exceeds a threshold, satisfaction drops below a threshold, a hospital has reached its capacity and research has been completed.

The game is won, if the virus is completely exterminated or the population is immunized. It is lost, when too many people have died, the virus spread to far or the satisfaction level reached zero.

1.1.2. Inspiration

One of our main inspiration was the board game "'Pandemie"'. In the classical "'Pandemie"' game four player fight cooperatively to try to stop a global pandemic to happen. Each player has different abilities, which are needed to stop all possible viruses, which could cause a pandemic.
As already describes above, our game idea is now to transport this idea to fight a pandemic in a big city like Munich. The inspirations to do this in real time is mostly based on the game "'Plague Inc"', in which the player is evolving a virus to eliminate the world. Instead of copying this, we wanted to change the perspective and let the player control the people, which would stop such a virus.
Furthermore, it was interesting to think about what would happen if a virus like Ebola would spread in a city in Germany. What possibilities, we would have here in a developed country and what might be happening.

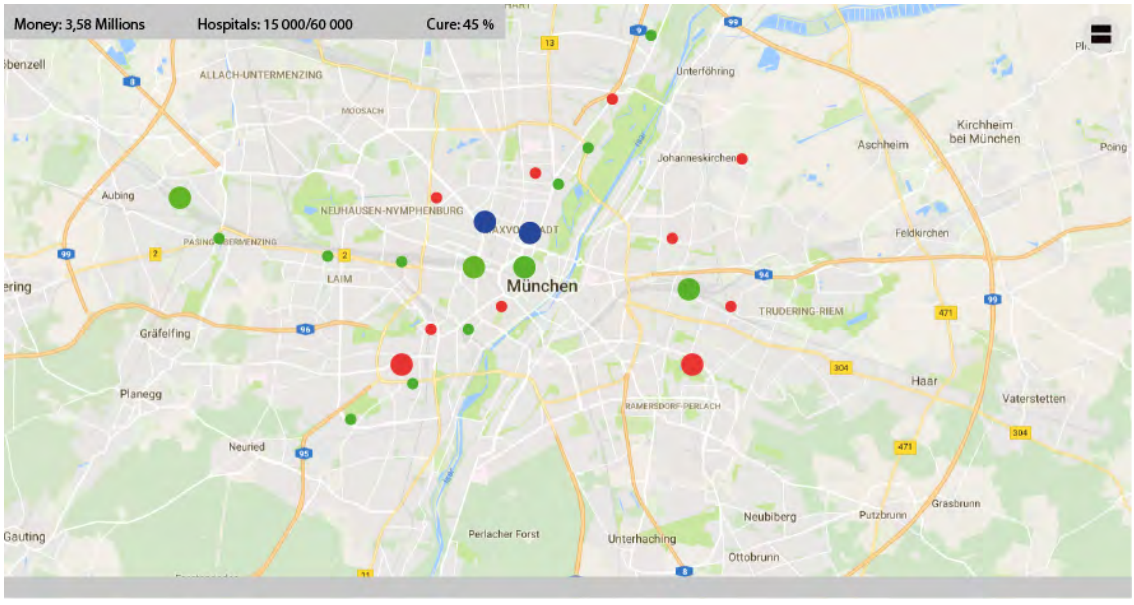


Figure 1 General game concept in 2D view

Figure 2 Visualization of special buildings in 2D view

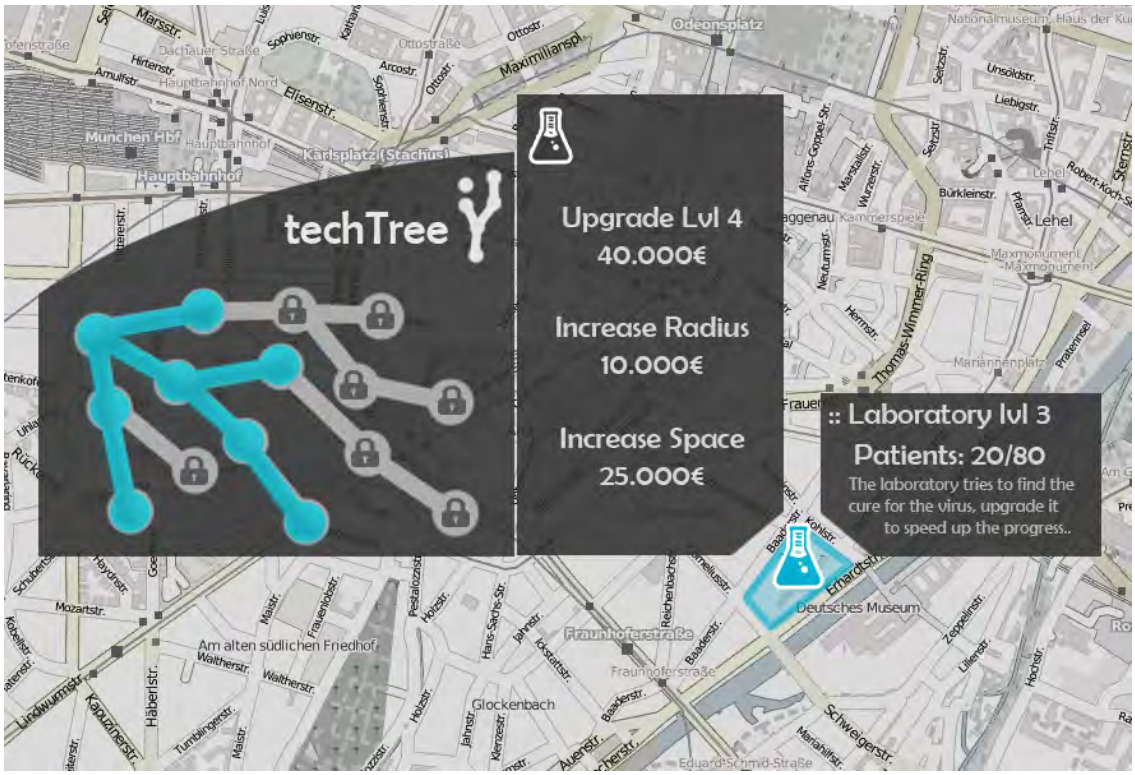


Figure 3 Concept: Upgrade options for laboratory and TechTree

1.2. Technical Achievement

For the game implementation we want the employ Unity as the underlying game engine. There are two major technical challenges for our Munich simulation we will focus on during development: a realistic crowd simulation coupled with an intelligent virus.

The crowd simulation should be scalable for several hundreds or thousands citizens and model their daily behavior. The players can take directly influence on crowd behavior (e.g quarantine areas restricting the walking areas). A large factor for the success of our game will be virus AI. Therefore we are aiming for an adaptive spreading algorithm, which has some sort of randomness to keep the players motivated and prevent finding easily winning strategies. The spreading will be further coupled with the crowd simulation to infect the citizens on contact.

Beside these we will have to deal with additional challenges like generating a virtual 3D model of Munich (or a subset) from existing data of map services and visualizing the game data (e.g infection rate) to guide the players and support them in their decision making.

Figure 4 Early Concept: View of the Marienplatz with infected citizens and buildings

1.3. Schedule

1.3.1. Development Layers

In total, five development layers are described in the following subsection. They describe the different stages of the game, starting with the functional minimum and ending with the high target. The fifth layer is composed of extras that could be added after the project ended.

The individual items are mostly described in an iterative manner. In the functional minimum we will create the first iteration of the item. In the following stages this item will be improved and enhanced with new functionalities. A good example would be the crowd simulation.

Functional Minimum

The functional minimum is the set of minimal items to make our project called a game. There are five main items that we want to finish in this layer. They are described in the following list:

- 2D Card of Munich: Street Layout of Munich including simple visualization of the spread of the virus.
- Simple Crowd Simulation: Entities are going directly from one place to another. No simulation of daily routines.
- Simple Virus Simulation: Virus is spreading more or less randomly. Mostly based on probability.

Figure 5 UI mockup: Visualizing the infection rate with colors and a few HUD elements

- Basic Player Interaction: Player can identify virus via the user interface and fight it with limited options.
- Simple Resource Management: Money is implemented and can be spend for player actions.
- User Interface: Main interfaces are done and can be used. Style is not final in this stage.

Low Target

The low target is absolute minimum we want to achieve during the semester. It contains items that should improve the fun of playing the game and the overall user experience. Five main areas are addressed in this layer:

- Simple 3D Representation of Munich: Munich should be recognizable. Some key building should be identifiable.
- Enhanced Crowd Simulation: Crowd should follow streets/transportation lines. Realistic traffic speed.
- Enhanced Virus Simulation: First steps to a realistic spread of virus should be finished. Virus should spread from one person to another with some probability if they meet.
- Enhanced Player Interaction: Style of User Interface should be done and more player actions implemented.
- Research Ladder: Player should be able to research upgrades and finally able to find a cure/vaccination.

- Upgrades: Certain buildings like hospitals or stations should be upgradeable.

Desirable Target

This layer is the target layer. This layer we want to achieve during the project time. Based on this our development schedule was created. Most of items listed here are based on earlier implementations described in the two layers before.

- Realistic Crowd Simulation: Crowd should have daily routines. Not just only moving between work and home, but also to other destinations like the gym or supermarket. Especially a day-night-cycle should be implemented.
- Realistic Virus Simulation: Virus should adapt to player actions and spread realistically, especially over public transportation and places.
- Enhanced Resource Management: Player should be able to control the flow of the money and manage it. Implementation of statistics.
- Difficulty Levels: Player should be able to choose between different difficulties.
- Mini Missions: Small missions to guide the player especially in early stages of the game. Show them what they can do and what would be reasonable.

High Target

The High Target contains items that might be implemented if everything goes really well. The items would be nice to have and would enhance gameplay and as well the graphical representation of the game.

- Enhanced 3D Representation: Detailed modelling of special buildings like Allianz Arena.
- Research Tree: Enhanced research possibilities with different ways to get the cure.
- Advanced Story Arc: Make the player more aware of the danger of the virus, maybe introduction trailer.
- Tutorial Mode: Implement a little tutorial for the player.
- Dynamic Sound and Graphic Adaptation: Sound and graphic should react on state of the game.
- Highscore: Online highscore to compare with other players.
- Achievements: Player can earn achievements if they master special challenges.

Extras

Some additional features that could be implemented as extras would be a cooperative gameplay. So player could work together to fight the virus. It would be also fun if another player could play the virus and therefore we would have a more competitive aspect in the game.

Another area would be to create a better atmosphere with the help of cut scenes and special events. Therefore, the player might be more engaged with the game. Furthermore, educational aspects on better hygiene and healthcare could be an interesting addition to the game.

1.3.2. Development Schedule

This schedule presents how we plan to implement the game. At the end of the implementation phase our target is to have all desirable target items to be finished. The schedule includes assignment of the team members to different tasks. This is still work-in-progress and can change during the implementation phase.

Vorgan	Vorgangsname	Dauer	Anfang	Ende
	Game Proposal	**11 Tage**	**Mit 12.10.16**	**Mit 26.10.16**
	Brainstorming	3 Tage	Mit 12.10.16	Fre 14.10.16
	Find Game Idea	4 Tage	Mit 12.10.16	Mon 17.10.16
	Finalize Game Idea	2 Tage	Die 18.10.16	Mit 19.10.16
	Write Game Proposal Chapter	**5 Tage**	**Don 20.10.16**	**Mit 26.10.16**
	Bullseye	1 Tag	Don 20.10.16	Don 20.10.16
	Technical Achievement	5 Tage	Don 20.10.16	Mit 26.10.16
	Development Schedule	5 Tage	Don 20.10.16	Mit 26.10.16
	Assesment	5 Tage	Don 20.10.16	Mit 26.10.16
	Game Description	5 Tage	Don 20.10.16	Mit 26.10.16
	Make Presentation	3 Tage	Mon 24.10.16	Mit 26.10.16
	Game Proposal Presentation	0 Tage	Mit 26.10.16	Mit 26.10.16

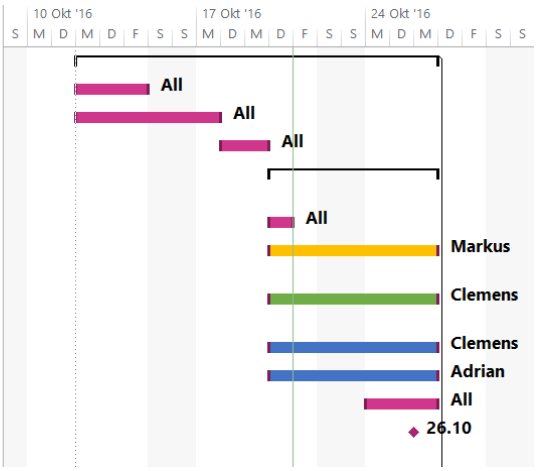


Figure 6 Schedule until the game proposal mile stone.

Vorgan	Vorgangsname	Dauer	Anfang	Ende
	Game Proposal Presentation	0 Tage	Mit 26.10.16	Mit 26.10.16
	Physical Prototype	**10 Tage**	**Don 27.10.16**	**Mit 09.11.16**
	Scope of the Prototype	3 Tage	Don 27.10.16	Mon 31.10.16
	Build Prototype	3 Tage	Mon 31.10.16	Mit 02.11.16
	Test Prototype	3 Tage	Mit 02.11.16	Fre 04.11.16
	Adapt and Review Prototype	5 Tage	Don 03.11.16	Mit 09.11.16
	Physical Prototype Chapter	**3 Tage**	**Mon 07.11.16**	**Mit 09.11.16**
	Section: Experience; What was fun?	3 Tage	Mon 07.11.16	Mit 09.11.16
	Section: What did we learn from it?	3 Tage	Mon 07.11.16	Mit 09.11.16
	Section: Design Revions	3 Tage	Mon 07.11.16	Mit 09.11.16
	Section: How did we make it?	3 Tage	Mon 07.11.16	Mit 09.11.16
	Make Presentation	3 Tage	Mon 07.11.16	Mit 09.11.16
	Physical Prototype Presentation	0 Tage	Mit 09.11.16	Mit 09.11.16

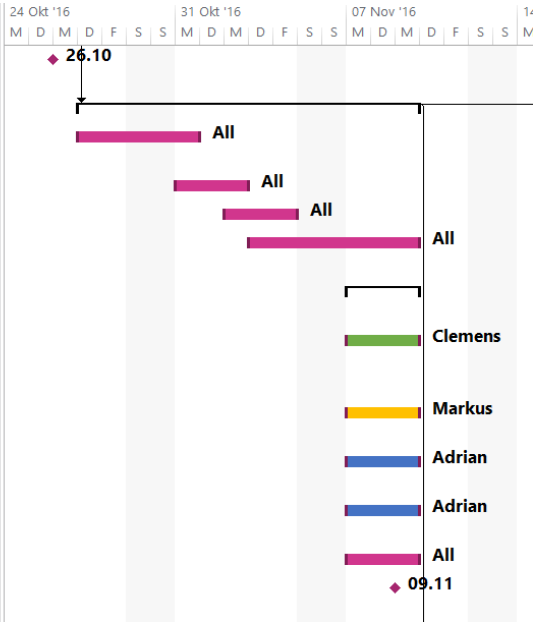


Figure 7 Schedule from the game proposal until the physical prototype mile stone.

Vorgan	Vorgangsname	Dauer	Anfang	Ende
	Physical Prototype Presentation	0 Tage	Mit 09.11.16	Mit 09.11.16
	Functional Minimum	**8 Tage**	**Don 10.11.16**	**Mon 21.11.16**
	Simple representation of Munich	8 Tage	Don 10.11.16	Mon 21.11.16
	Simple visualization of virus spread	3 Tage	Don 10.11.16	Mon 14.11.16
	First version of the crowd simulation	8 Tage	Don 10.11.16	Mon 21.11.16
	First version of virus simulation	8 Tage	Don 10.11.16	Mon 21.11.16
	Usable user interface	3 Tage	Don 10.11.16	Mon 14.11.16
	Basic Player interactions	5 Tage	Die 15.11.16	Mon 21.11.16
	Functional Resource Managment	3 Tage	Don 10.11.16	Mon 14.11.16
	Functional Minimum Milestone	0 Tage	Mon 21.11.16	Mon 21.11.16

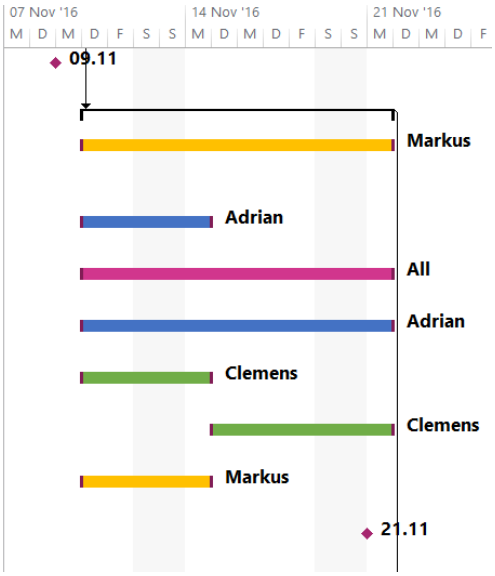


Figure 8 Schedule from the physical prototype until the functional minimum layer.

Vorgan	Vorgangsname	Dauer	Anfang	Ende
	Functional Minimum Milestone	0 Tage	Mon 21.11.16	Mon 21.11.16
	Low Target	**7 Tage**	**Die 22.11.16**	**Mit 30.11.16**
	Modelling of special buildings in Munich	7 Tage	Die 22.11.16	Mit 30.11.16
	Adapative Virus Simulation	7 Tage	Die 22.11.16	Mit 30.11.16
	Second Version of the Crowd Simulation	7 Tage	Die 22.11.16	Mit 30.11.16
	Research Ladder	4 Tage	Die 22.11.16	Fre 25.11.16
	Upgrading of buildings	3 Tage	Mon 28.11.16	Mit 30.11.16
	More player actions	3 Tage	Mon 28.11.16	Mit 30.11.16
	Low Target Milestone	0 Tage	Mit 30.11.16	Mit 30.11.16
	Interim Demo	**3 Tage**	**Mon 28.11.16**	**Mit 30.11.16**
	Make Presentation/Live Demo	3 Tage	Mon 28.11.16	Mit 30.11.16
	Interim Demo Chapter	**3 Tage**	**Mon 28.11.16**	**Mit 30.11.16**
	Section: State of the game	3 Tage	Mon 28.11.16	Mit 30.11.16
	Section: Difficulties	3 Tage	Mon 28.11.16	Mit 30.11.16
	Interim Demo Presentation	0 Tage	Mit 30.11.16	Mit 30.11.16

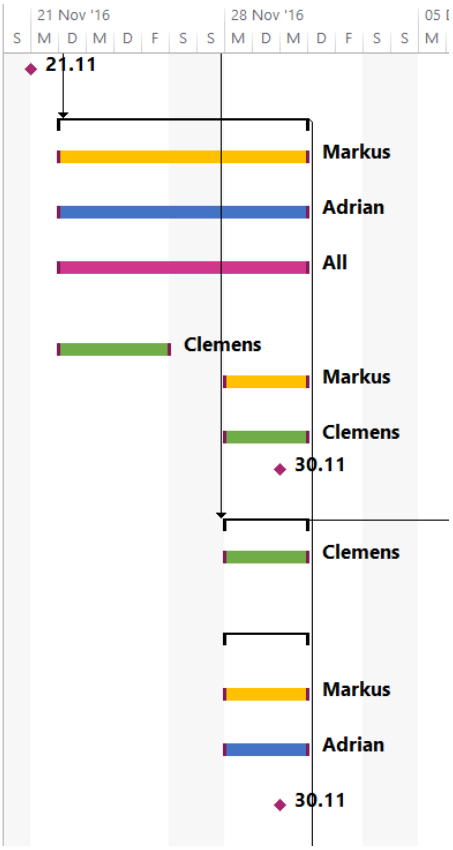


Figure 9 Schedule from the functional until the interim demo mile stone.

Vorgan	Vorgangsname	Dauer	Anfang	Ende
	Interim Demo Presentation	0 Tage	Mit 30.11.16	Mit 30.11.16
	Desirable Target	**15 Tage**	**Don 01.12.16**	**Mit 21.12.16**
	Mini Missions	5 Tage	Don 01.12.16	Mit 07.12.16
	Different difficulties	5 Tage	Don 08.12.16	Mit 14.12.16
	Realistic day and night-cycle	10 Tage	Don 01.12.16	Mit 14.12.16
	Final crowd simulation	5 Tage	Don 01.12.16	Mit 07.12.16
	Final virus simulation	5 Tage	Don 08.12.16	Mit 14.12.16
	Enhanced resource managment	5 Tage	Don 01.12.16	Mit 07.12.16
	Enhanced city model	15 Tage	Don 01.12.16	Mit 21.12.16
	Desirable Target Milestone	0 Tage	Mit 21.12.16	Mit 21.12.16
	Alpha Version	**3 Tage**	**Mon 19.12.16**	**Mit 21.12.16**
	Make Presentation/Live Demo	3 Tage	Mon 19.12.16	Mit 21.12.16
	Alpha Version Chapter	**3 Tage**	**Mon 19.12.16**	**Mit 21.12.16**
	Section: What is missing?	3 Tage	Mon 19.12.16	Mit 21.12.16
	Section: What is done?	3 Tage	Mon 19.12.16	Mit 21.12.16
	Alpha Release Presentation	0 Tage	Mit 21.12.16	Mit 21.12.16

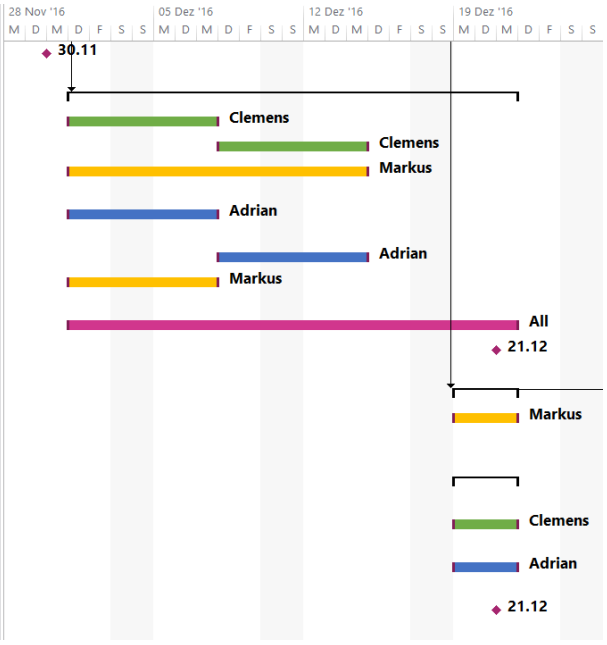


Figure 10 Schedule from the interim demo until the alpha version mile stone.

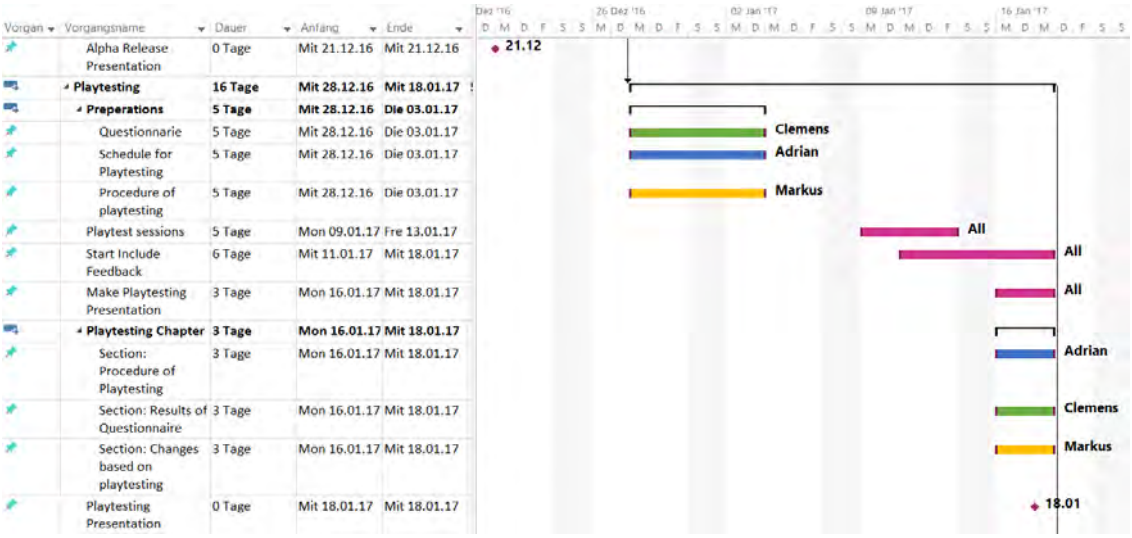


Figure 11 Schedule from the alpha version until the playtesting mile stone.

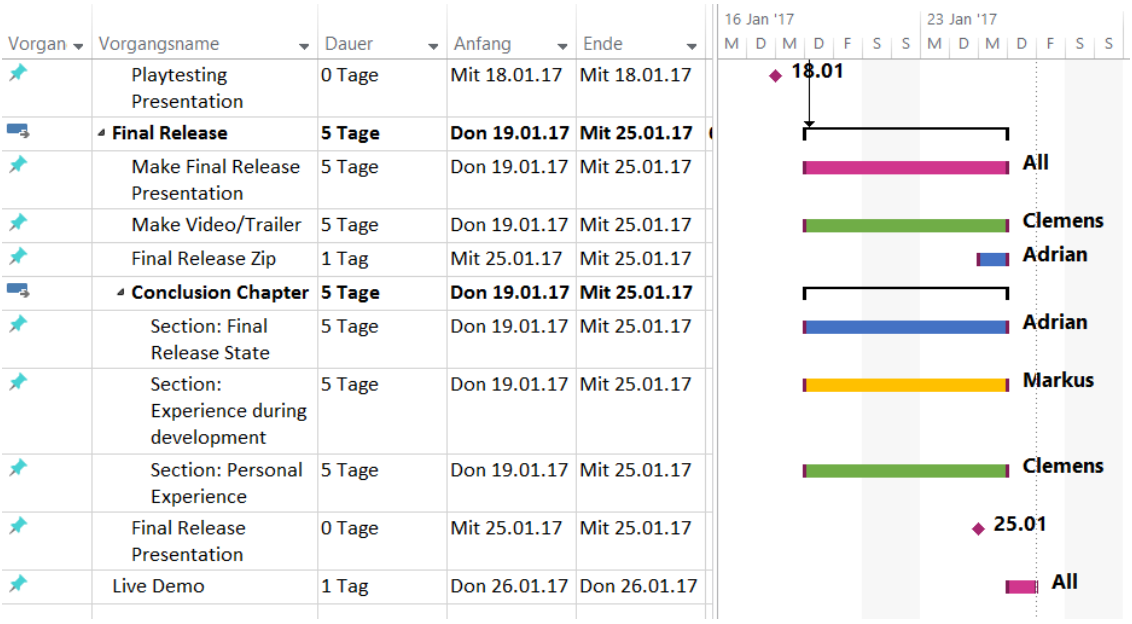


Figure 12 Schedule from the playtesting until the final version mile stone.

1.4. Assessment

The main idea of our game is to have a fast passed, but strategical gameplay. The core game mechanics should feel believable and realistic, especially the spread of the virus and the simulation of the crowd. The thrill to fight the virus and to save the city should be a main focus of the game.
Based on the strategical gameplay, our main focus group are fans of strategy and logic games. Nevertheless, we also want to reach more causal player that like fast-paced games. Additionally, we think that the topic of our game could be attractive to a wide variety of players.
For every player, the main task will be to fight the virus. Different options will be available for them, enabling different game styles. One of the key challenges will to keep the balance between fighting the virus and keeping the citizens of Munich happy.
Therefore, it is very important that Munich and the spread of the virus are simulate in a realistic manner. Citizens should follow daily routines and not only randomly go from one place to another. The virus should spread from one entity to another and not just randomly appear.
In summary, our criteria of success contains four aspects. Firstly, the player should feel the thrill to fight the virus. Secondly, the simulation should work - virus and population. Thirdly, the game should be well balanced. This means that each game feature should have a right to exist and not only be there to be a feature. As well, winning should not be too easy but the player should not lose every time. Lastly, Munich should be recognizable and buildings like the hospitals or the central station should be at the right place. And as every game it should be fun to play.

2. Physical Prototype

2.1. Prototype

The key goal of an prototype is to build something that distils out the core gameplay. This can be done in a software prototype. But often you will start with a physical one using paper, glue, and other materials like from board or card games. Even in a physical prototype, it is important to include the core game mechanics.
For our project we build such a physical prototype. In the following subsections, we will explain how we transformed our idea to a physical prototype and how it is played.

Figure 13 Shows the playfield after some turns.

2.1.1. General

As stated in the game proposal, we want to build a game in which the player fights a virus that is spreading in Munich in real time. In the game prototype we changed the gameplay to a round-based one to focus more on the actions of the player and the AI. The technical achievement for our game is to simulate the citizens and the spread of the virus realistically. The prototype cannot achieve a realistically simulation therefore we changes this to a random approach, similar to the description of our functional minimum in the game proposal chapter. Based on this changes, we created the game prototype.

(a) Detector that can be placed by the player.

(b) One of the hospitals that are on the map.

(c) Small red rods that are used for the infected.

(d) Small white rods for the research points.

Figure 14 Several items that are used in the prototype.

2.1.2. Materials

The most important material is a city-map of Munich, which is used as a play field. On the map we highlighted mayor hospitals in the area of Munich. Furthermore, we used the grid of the map for tracking the position of infected entities. Entities are represented by small red rods. Detection devices can be placed on the intersections of the map grid and are represented by blue rods. A dice is needed for different actions in the game for example which action the AI will perform. Lastly, small white rods are needed to display the amount of research points.

2.1.3. Game Rules

This subsection describes the game rules of the game. How the player can lose and how they can win the game. Furthermore, all player and AI actions are described. We will begin with the start of the game.

Game Start

The game starts with the initial positioning of 10 infected entities on the city map. Randomly each entity is allocated with a position. The position is based on the grid of the map, for

example D5 or E4. The player does not know where the infected are. They are only visible for the AI on a sheet with the positions. Now the player has in each round 3 "‘action points"’ which they can use for different actions. The goal for them is to find the infected entities, bring them to the hospital and heal them. The player can identify infected through the placing of detectors on the map and the manual searching of fields. If the player has used all there "‘action points"’, it is the turn of the AI. The AI can move, create new and kill entities.

How to lose? How to win?

The player loses when more than 10 entities are dead. They can die due to the actions of the AI or if the player tries to heal them. Another possibility to lose is when more than 20 entities are infected at the same time.
The player wins when they heal all infected entities. They can heal entities if they detect them and then heal them in the hospital. Furthermore, the player can win if they find a cure against the virus. A cure is found when the player gained 15 research points. Research points are gained if a entity is healed or dies in the hospital. Per entity one research point is gained.

Player Actions

The player has in total five different actions they can choose from. Each action costs "‘action points"’ (AP). The player has 3 AP per round.

- Place a detector on the intersections of the map grid: A detector can identify infected entities that stand in one of the adjoined grid fields. Entities in this area are no longer invisible and will be marked on the map via a small red rod. (2 AP)
- Search grid manually: The player can search three grids. If an infected entity is found, it will be marked on the map via a small red rod. (1AP)
- Hospital transportation: Bring a detected entity to the nearest hospital. If the hospital is full you cannot transport it to the hospital. (1AP)
- Try to heal an infected entity: If a entity is in a hospital, the player can try to heal the entity. Based on the throw of the dice different things happen: (1AP)
  - 6: Entity is healed. Player gains one research point and one additional AP. (H)
  - 3,4,5: Entity stays alive. (A)
  - 1,2: Entity dies. Player gains one research point. (D)
- Update a hospital: The player can update a hospital of there choice. (2AP)

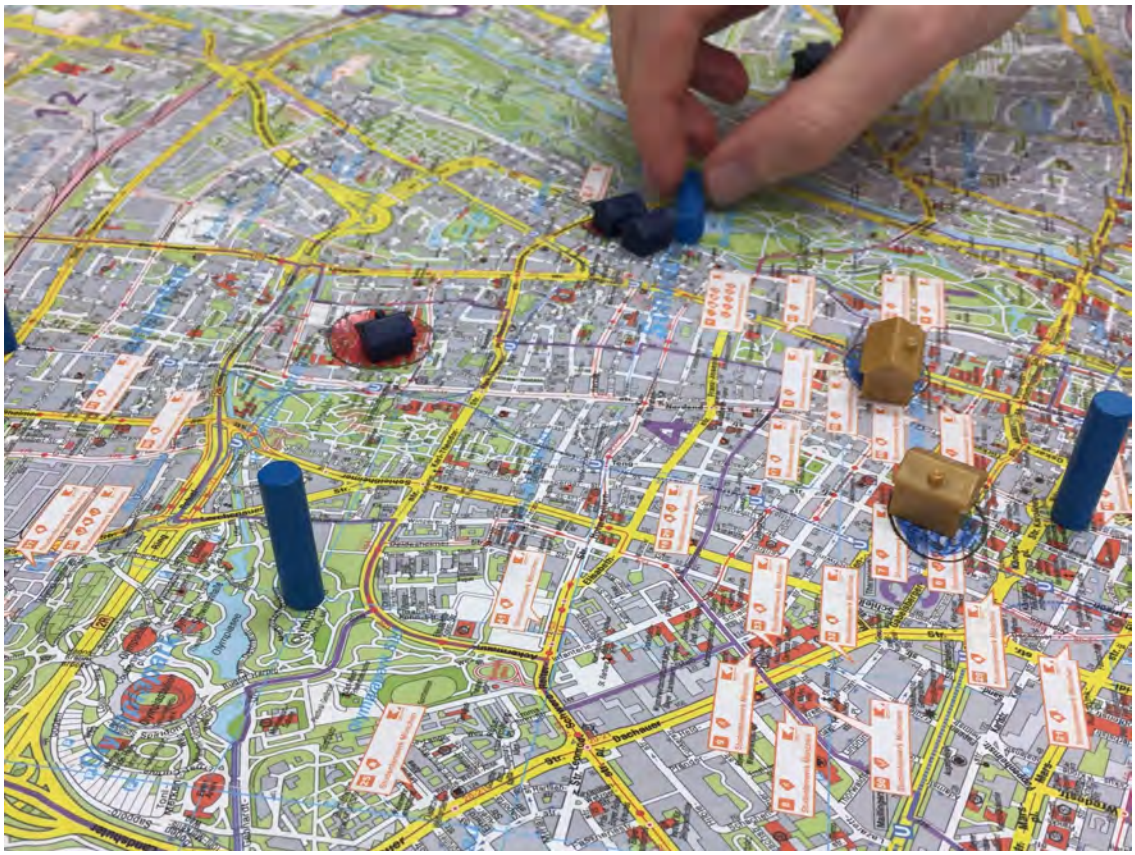


Figure 15 Player places a detector on the map. The hospital by his hand is already at stage 2.

AI Actions

The AI has three actions it can do. Which action is performed, is decided with the throw of a dice. The following listing shows the different actions with the corresponding dice number.

- 1,2: One or two infected are placed into a field in which already an infected entity is placed.
- 3,4: The AI can move three or four entities on the map. One entity can move 2 fields.
- 5,6: The AI kills randomly one infected entity.

Hospital

In the hospital infected entities can be moved and only there they can be healed by the player. Based on the stage of expansion they differ in capacity and how good they can heal infected. The list shows the attributes for the different stages.

- Stage 1: Capacity is 2. Normal statistics on healing probability.
- Stage 2: Capacity is 3. H:6; L:2,3,4,5; D:1
- Stage 3: Capacity is 4. H:5,6; L:2,3,4; D:1

2.2. Experience

Our overall experience throughout playing the prototype was quite positive, but we came across several balancing issues during playtesting, which we tried to at least partially fix in later iterations.

In our current setup we are starting with ten infected citizens to make it easier for the computer player to keep track of all of them. Unfortunately, this number turned out to be much too small for the high resolution grid of our map. Because of that, it took a long time until first infected persons were detected. Also, it was very difficult to congest hospitals, making it useless to update those.

Initially, we limited the movement of infected persons to one cell. However, this decision made the game to static and we needed to increase their mobility. In this first iteration of our prototype, it was also possible to use action points for research. Since this system could be easily be abused - the player was guaranteed to win by spending all action points on research - we changed the research system to what was mentioned in the game rules section.

We learned from these experiences, that finding the right balance between gameplay elements is very difficult. This applies in a prototype, which omits several parts of the game, so this will be a big challenge in the development of the final game.

2.3. Design Revisions

In this section we will discuss the changes of our game concept from the 1st milestone with regards to the presented physical prototype and the feedback we received for our initial concept.

2.3.1. Prototype

We tried to transfer the insights gained during playtesting and creating our physical prototype to our game idea concept. As we only modelled the core part of our game concept we can't review all gameplay features.

The virus detection turned out to play a larger factor in our gameplay and we want lay stronger focus on it (detectors). Alongside with this change comes a scale adjustment for our map as want to reduce the virtual representation of Munich to the city center. In similar fashion we transfer to more area based spreading and detection of the virus than handling it separately for each house.

2.3.2. Critiques

For our initial proposal we received a lot of feedback from our fellow students (big thanks!). The overall feedback was pretty good and we tried to extract the common aspects mentioned for improvement:

Satisfaction mechanic Our satisfaction level system for the citizens was frequently mentioned. We have to further elaborate this topic to achieve a solid part of the game concept, especially take it more into consideration for balancing to enforce a dynamic gameplay.

Learning based AI The virus AI represents a major aspect of our gameplay in terms of a balancing and has a large impact on the success of our game. The feedbacks emphasized this aspect multiple times and suggested different behavior strategies which we consider for our upcoming implementation.

Research tree We got many interesting suggestions for the improving our research tree, which could vastly improve this part of the gameplay. Having different virus types and making the research depend on it is something we already consider. An different idea, which came up, was to add minigames for advancing the research.

Crowd behavior Another mentioned point is the dynamic behavior of the crowd on the current game state (e.g grouping for riot at low satisfaction). This would be a desirable target for us as it would make the overall game more dynamic and fun.

Beside points listed above there are several good other suggestions, which probably lay out of the scope for this lab like multiplayer or a story mode due to the limited time. Even tough we would really like to implement something similar.

3. Interim Report

3.1. Progress

As stated in the project schedule we started with the implementation of our game right after finishing the physical prototype. By now, we have finished the functional minimum layer and are mostly done with the low target. In general, we are happy with our progress, especially with the user interface design and the graphical representation of Munich. Furthermore, we think based on the work we already done, we can quickly make progress in the development of the game. Nevertheless, they are still lots of things to implement and test. Especially, the part of balancing the game can become quiet a lot of work as we think.

3.1.1. Progress in Layers

In this subsection we will describe the progress of our game in more detail. This will be based on our goals for each layer as we stated in the first chapter. We will mostly focus on the first two layers, but will also mention some aspects of the desired target layer and what needs to be done.

Representation of Munich Our approach to build a realistic representation of Munich is based on the "'Open Street Map"' data, which is available for free use. The OSM data format is a form of XML, divided into three main areas: nodes (the most general structure), ways (consist out of nodes) and relations (consist out of nodes or ways). With the help of this data, we were able to rebuild Munich with real data in a procedural way. This topic includes a lot of computational effort and parsing of data, but in the end we think this will give us the best results to realistically display Munich. As you can see in Figure 16, we can now represent Munich in a simple 3d way. There are still not all tags used that OSM provides and the art-style is not yet finished. Furthermore, complex structures are not yet fully renderer (wholes in the polygon). The functional minimum and the low target can be seen as reached in this area. For the desired target there is still missing some visual highlights and especially the visualization of the spread of the virus.

Simulation: Crowd and Virus Based on the approach of using OSM and therefore the possibility of generating a graph of all roads in Munich, we first implemented a version of the A* algorithm. Quickly, we realized that this approach would be to computational heavy to do this for all citizens. Therefore, we decided to follow two approaches. Firstly, a really simple one which just calculates the next step based on a really simple heuristic, and secondly implementing a routing table for each node in the network. As by now we are finished with the simple one and are well into the routing table approach. Nevertheless, there is still a lot of work to do. For example there is the need to enhance the daily routines of the citizens and how the virus is spreading. At the moment this is all done in a really simple manner, but we think that we can make a realistic simulation with the help of the

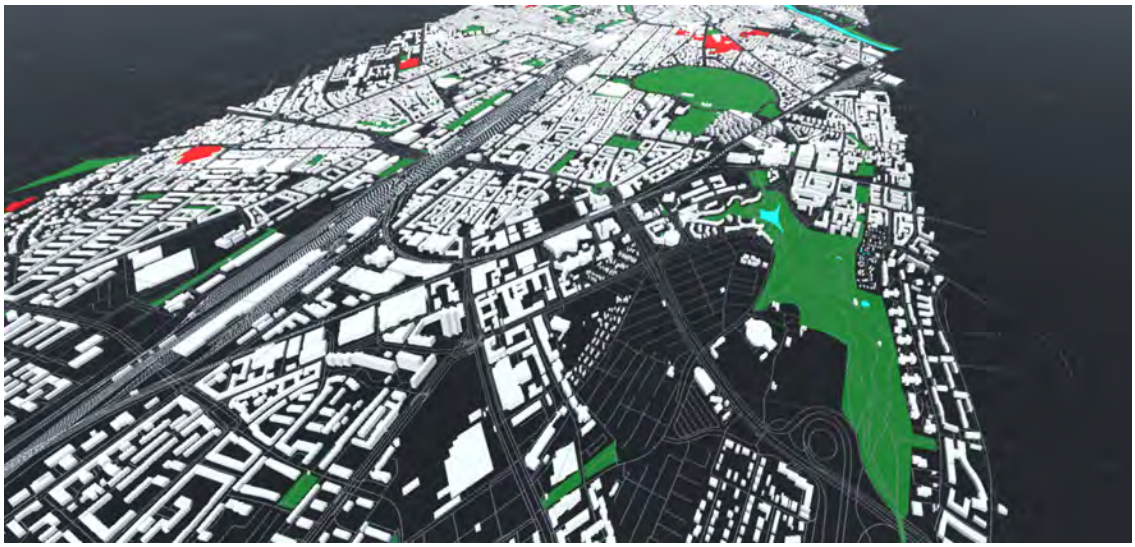


Figure 16 Current version of the graphical representation of Munich.

routing tables. In summary, the functional minimum is definitely reached and nearly the low target. To desired target there is definitely missing the implementing of realistic daily routines and a more sophisticated spread of the virus.

Player Interactions and User interface The user interface design is a very important part of the game because nearly all actions are controlled over the interface. We are very happy with the overall design of it, but there is still room for improvements, especially to give the player more feedback about the current state of the game. To this point we implemented the basic player interactions like sending medical troops to an area, placing detectors and investing research points. More details are explained in the following section. All in all, the functional minimum and the low target is reached. For the desired target, it is now necessary to improve feedback and to include more player interactions with the game.

Resource Management, Research Ladder and Upgrades This is the area where most of the work is still need to be done. In the resource management we made a design revision. Now all player actions just cost research points and there is no money any more. The player can spend it to build detectors, upgrade buildings, do research and find the cure. This is already implemented, but still needs a lot of polishing. The research ladder itself is not yet fully implemented, but the user interface is finished. In the area of upgrading buildings there is still many work to be done. In total, we finished the functional minimum. The low target is mostly done, but still needs polishing and some more work in the upgrading part. The good thing is that this can easily be enhanced once the structure is finished and should be definitely be in a good state at the end of the implementation phase.

3.1.2. Forecast

Overall, we think we should manage to finish our project plan in time. In some areas, we are a little bit behind the schedule, in other areas like the graphical representation and the user interface we are a little bit ahead. At the moment, they are two mayor areas, which could endanger our schedule. Firstly, the simulating of the citizens could become to heavy loaded,

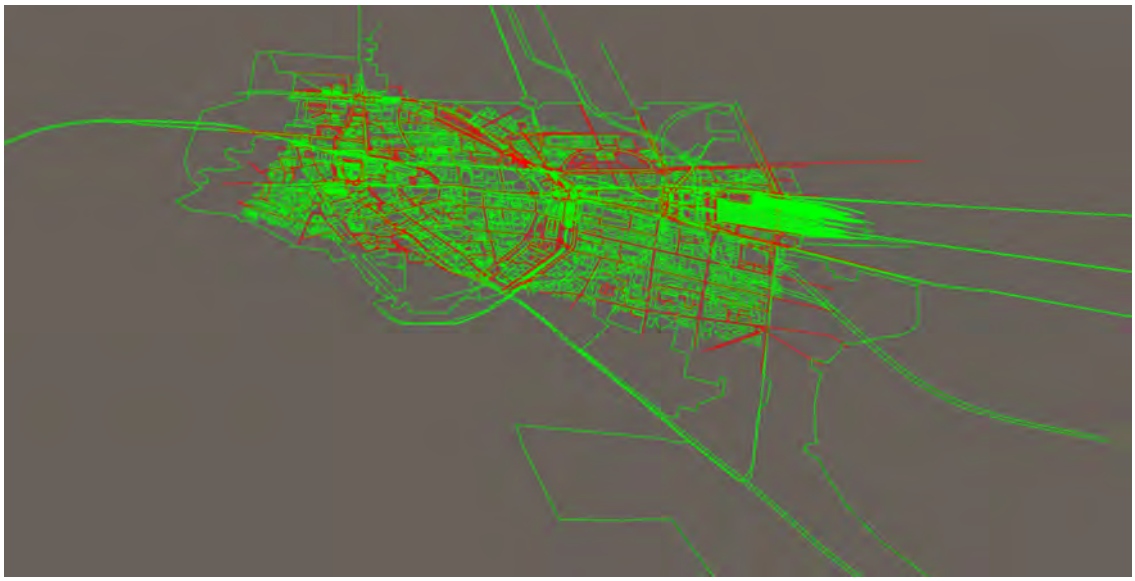


Figure 17 First trail to representing Munich.

especially with daily routines and the sophisticated spread of the virus. This could lead to an unplayable game. To stop this from happening, we are always profiling our game and the frame rates. Furthermore, we need to do a trade-off between performance and simulation in the end. Secondly, the balancing of the game could consume to much time and therefore the game might in the end not as much fun to play as it could be. As a precaution, we try to start the balancing as early as possible and give the player as many feedback as possible.

3.2. Interaction

3.2.1. Control

The player sees the city in a top-down view from a large-scale perspective. The whole game can be played by using a mouse only, but the view can also be panned with WASD or the arrow keys optionally. We implemented common interaction schemes from 3D editors, which fit quite well for our gameplay. The viewport can be adjusted by zooming in with the mouse wheel and panning via right-clicking and dragging the map. All interactable buildings and objects are clickable by hovering the mouse over the specific object in screen space.

3.2.2. User Interface

A large part of the player interaction is done via the GUI. We try to provide a clean and easily understandable interface for the user to be able to focus on the actual game and provide him good overview of the current game state. Our current progress can be seen in Fig. 18 at an early stage in the game. On top we provide show the basic condition of our city in order to allow the player to see how far the virus has already spread. The progress bar visualizes the advancement in the research towards finding the cure for winning the game. In the right panel the player can build new detectors and place it on the map or make progress in the research area using the gained Research Points (RP). Fig. 19 shows the popup menus for hospitals

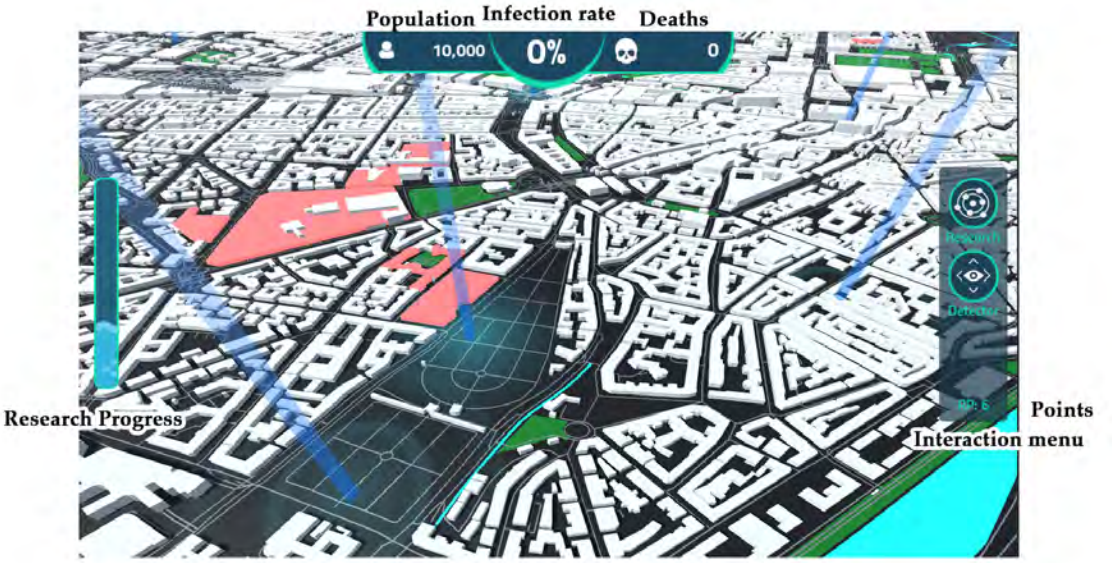


Figure 18 GUI overview: Main HUD of the game including different UI elements: Research bar (left) visualizing the current progress until finding the cure, city status on top showing the current condition of the citizens giving a rough estimation of the game state. The menu on the contains the main interaction control with detector building and upgrading the research tree

and the research tree. In the hospital menu you can control your medic troops to send them to waypoints on the map, where they can treat citizens in the specific area. For the research tree we plan to provide the player upgrade options for better virus detection, treatment, etc, but this is currently work-in-progress.

3.2.3. Gameplay

We have made several gameplay changes with respect to our initial proposal after building the physical prototype. The current game lays more focus on finding infected citizens and trying to heal them in specific places using medic troops. Over the time the player gains research points, which are our current form of currency the player can use to build new detectors, make progress in research and upgrade hospitals. Detectors will show all people in a certain radius with their current infection status. The red colored buildings in Fig. 18 are hospitals, which have associated medic troops the player can control. Each troop can be sent to a certain location on the map, where it will treat the infected citizens. For a successful treatment the player will gain additional research points which can be further invested.

3.3. Challenges

One big challenge is to make the game run smoothly. Especially the simulation of the population is a very demanding task for the CPU. Our first idea was to use the A*-algorithm once a citizen chose a goal to calculate a path towards it. As mentioned before, it turned out to be very time consuming to find a path for each citizen that way. Precomputed paths for each possible start and goal combination on the other hand takes too much memory for a network

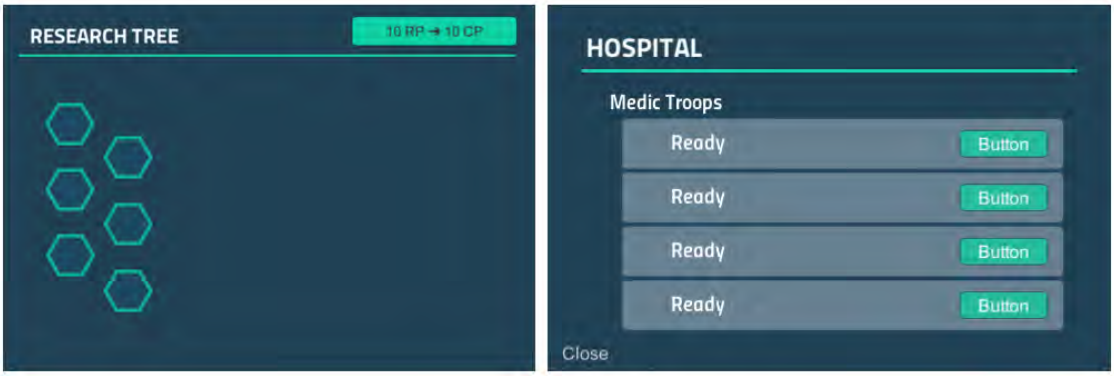


Figure 19 Popup menus: Research tree (left) for unlocking new features and advance in the development of a cure. Hospital menu controls the medic troops, which you can emit for treating citizens.

with more than 50,000 nodes. Because of this, we plan to compute a routing table at each node that stores the neighbor with the shortest remaining way for every goal node. To send medic troops to a position in the scene, it was necessary to find the nearest node to the position of the player's mouse click. Since it costs a lot of time to iterate through all nodes and calculate the distance to each of them, we do the search using some kind of k-d tree.

Another challenge was to understand how the OSM data are organized and how to process them. It turned out, that there are several inconsistencies in the OSM data set. E.g. some faces are stored in clockwise order while others are stored counterclockwise. Also, it took a long time to create the objects at the beginning of the game, so we now load models, which we precomputed beforehand, instead. Since there are many buildings in Munich, which don't change over time, we combined them to big game objects.

4. Alpha Report

4.1. Progress

At this point, most of the implementation work is done. We are happy with the progress we made since the interim report. We finished all the desired targets (besides the mini-missions) and as well some of the high targets. In retrospective, the project plane and the targets we defined was a good starting point and helped us a lot to keep track of our progress. One point that changed quite a bit from the project plan, was the assignment of the work. We think that is natural if you are not familiar with the other team members and in detail with their skills. Now we are looking forward to let other persons play the game and get their feedback. This will be especially helpful in balancing the game and maybe tweak some of the game elements. We hope that they will enjoy the game as much as we do.

Figure 20 Current version of the graphical representation of Munich at the beginning of the game.

4.1.1. What is missing?

Overall, we can say that nothing essential is missing. All the things that are not yet finished would just be elements that would be nice to have, like mini-missions or achievements. They would not change the core of the game. Nevertheless, they are some areas, which would definitely profit from more time. One example would be the overall atmosphere of the game. We did not have enough resources to create some background story or some sort of story arc in the game. At the moment, the player is just thrown into the game world with a short introduction, what happened and what they need to do. More time could be spend to let the player be more engaged with the game world and feel even more the thrill to fight the virus. Another example would be the simulation of the citizens and the virus, which could be

made more complex and enhanced. Beside these reasonable elements, some features were just out of scope for our project and the time we could spend, like to implement an online highscore or even some sort of cooperative game mode.

4.1.2. What is done?

After describing what is missing, this subsection is focusing on what we achieved. And as we already said we are quite satisfied with it. The structure of this subsection is similar to the one in the interim report chapter. This means that we will describe what we finished until this point of time. We will orient us on the targets we have described in the first chapter. Targets that we already described in the interim report, will just be shortly mentioned.

Representation of Munich As we stated in the interim report, we are using the data of "'Open Street Map"' to generate the game world procedurally. In comparison to the interim report, we are using much more elements/tags to generate the world. Landuse is now better visualized and we added 3d trees to the map as a visual highlight. All this adds more variety to the map and helps the player to better orient in the game world. Furthermore, the emitted light changes based on the state of the game. The art style is more darker and serious with only few coloured highlights like the hospitals or the research buildings. We think this fits very well to the game setting. In the post-processing pipeline we added a tilt shift filter to the game to get some sort of miniature effect. In summary, we finished the desirable target in this area. We think that the visual aspect of our game is quite good. It could be enhanced of course by more details, but we do not think this necessary for the game to work, maybe even hinder the player to see the core game mechanics.

Simulation: Crowd and Virus In comparison to the interim report, we made the biggest step forward in this area. The routing table algorithm was successfully implemented and is working smoothly. The tables are generated offline and at the beginning of the game the pre-calculated tables are just loaded. Furthermore, daily routines were implemented. Therefore, every citizen has a home, a working place and some leisure place, which changes randomly. The citizen is moving between these three places and stays their for a certain time. This is controlled by our time implementation, which gives us a 24 hour-cycle. It is possible to adapt the game speed in three steps from normal to very fast. We added a simple adaptation to the virus. Based on how close the player is to the cure, the virus is more randomly choosing his path. Overall, we finished all desired targets. We are satisfied on how the citizens behave and think this should be sufficient for our gameplay. Nevertheless, we could definitely improve the daily routines by adding more variation and heuristics to the citizens and as well to the virus.

Player Interactions and User interface Over the complete implementation phase, we worked continuously on the enhancement of the user interface and the feedback for the player. As we described in the interim report, we are still satisfied with the design, the art style and how we implemented the interaction. From the interim report until now, a lot of new elements were added to the user interface. Especially, a lot of feedback mechanism for the player were implemented like popups and tooltips. In more detail this is described in the

next section and as well in previous chapter. Overall, we definitely reached the targets, we had described for this area.

Resource Management, Research Ladder and Upgrades As we hoped in the interim report, we were able to finish all the work in time and even finish work from the high target. We have a fully functional research tree with different areas and interdependences between research elements. The player can upgrade the capacity of the hospital, increase the healing chances, decrease the likelihood that the virus is spreading from one person to another and much more. Therefore, the player has many different possibilities from which they can choose. Upgrades for the hospitals and research facilities are working fine. The resource management satisfies our needs and is thanks to design revision now much more easier to understand and to balance as we think. With regarding to the balance of this components, we think that the playtesting will help us a lot and let us find the right cost values for research, upgrades and other player actions.

Sound Design One area that we not really had in mind was the sound design of the game. We only had stated this as a high target and had no time allocated in the project schedule. But playing the game we realized that it would be important to have music and as well sound effects in the game. Therefore, we added this to our list with elements that we want in the game. Firstly, simple sound effects were added to player actions like sending out an ambulance or placing a detector. Then sound was added to the notifications to make the player more aware that something happens that needs the attention of the player. Lastly, we added some simple dynamic music, which changes if the infection rate reaches some threshold. This is simple, but we think that this helps a lot to get the player more engaged. In the end, we are happy that we invested the time in the sound design and reached as well the high target to some degree.

Miscellaneous They were some minor things we implemented as well. It is possible to choose from different difficulty levels. We added a very simple tutorial to the game as an introduction for the player. Furthermore, players get some statistics after finishing the game to see how well they did.

4.2. Interaction

For the alpha release we finished the GUI parts, improved UI usability and add more feedback for the player as mentioned in the prior section.

4.2.1. User Interface

We finished to research menu from the interim release, added a start and end menu including a text tutorial and match statistics. During playtesting we noticed, that emitting medic troops is a very clicking-intensive tasks, so we added a shortcut for the player to send the nearest troops to the marked waypoint. Furthermore, we implemented controls to dynamically adjust the gamespeed to slowdown or speedup the simulation to avoid stalls when waiting for resources to come up.

As the game has an initial learning curve for the players and it might be hard get an overview

Figure 21 The game after some time played. The light is shifting and is getting more and more red.

over the different menus and interactions in the first games, we tried to support them with a text tutorial and tooltips on hovering the UI elements. The tutorial covers the basic game mechanics with winning conditions, controls, explanation of the buildings and the research tree.

An important aspect is also consistent feedback for the players, which we implemented via text and audio notifications.

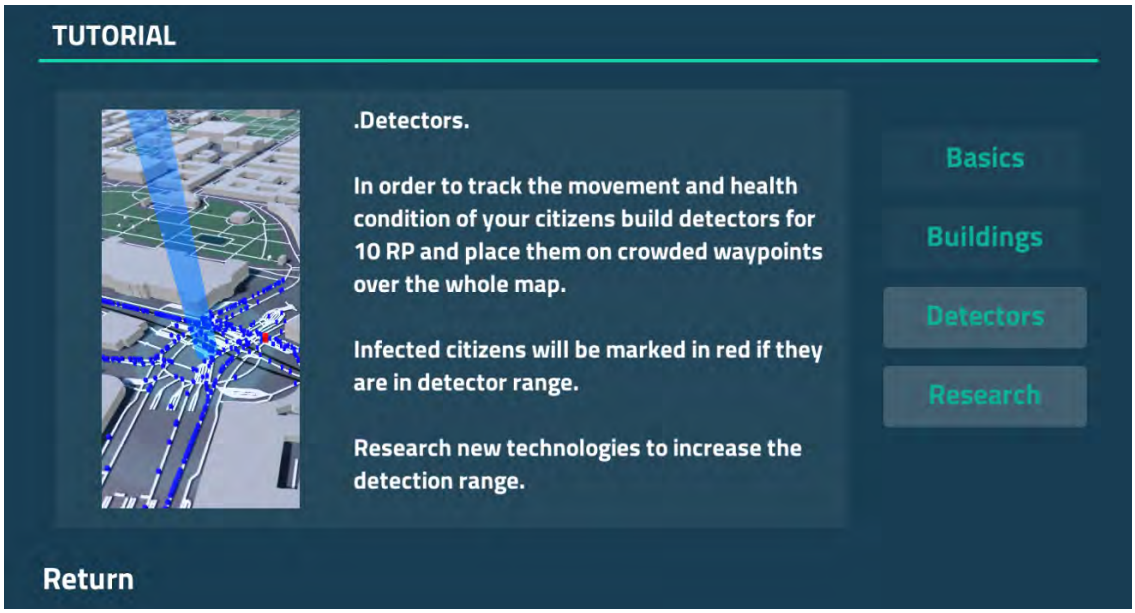


Figure 22 UI tutorial: Text based tutorial to help the players to quickly get an overview over the main game mechanics

4.2.2. Gameplay

We already settled our core game mechanics in the interim release. As a quick recap: our core game is focused on detecting and healing infected citizens by building detectors at specific waypoints in the city and emit medic troops to waypoints for bringing infected persons into hospitals, where they can be healed. We tried to enhance the game experience in the alpha release by adding more content in terms of research possibilities, hospital and university upgrades. This hopefully encourages the players to replay the game and keep them interested. Fig. 23 shows the research tree which has 3 subbranches for different research sectors allowing the player have more influence on the gameplay. The last node of each tree will slowdown the virus spreading permanently, which should be motivation for the player to take focus on spending points in research. With the different upgrade possibilities balancing becomes quite tricky and we will have to make further adjustments during the playtesting.
Beside that, we improved the visual representation of the city to reflect the current game status better and tried to catch the atmosphere of a spreading virus by adding visual and aurally effects as seen in Fig. 21.

Figure 23 Research tree: Tree based research ladder to unlock new skills with research points

4.3. Challenges and Implementation

For our game, we implemented a central control unit, which is responsible for organizing central parameters like simulation speed, research progress and the percentage of infected citizens. Several controllers, which are responsible for one aspect of the game each, can interact with these data. The UI controller displays the GUI and manages player input, as mentioned in the above section. The sound controller plays notification sounds and music

clips depending on the current state of the game.

Another controller is responsible for the simulation of the citizens. To save computing resources, they are divided in groups, of which only one is simulated in each frame. First, if a citizen is not already moving and its time for him/her to start moving, one of the predefined goals is chosen depending on the time of the day. Then, each citizen with an assigned goal moves towards the next street corner. Once such a point is reached, the next corner is chosen according to a table stored at the current one. Each time a citizen reaches a new corner, the simulation controller calculates, whether he/she is infected, depending on whether there are other infected citizens nearby. Finally, each infected citizen has a certain probability to die.

4.3.1. Challenges

The first challenge we encountered in the second phase of implementation was how to implement the routing tables. Because the map of Munich, which we use, contains more than 50.000 streets, we couldn't simply store all combinations of start and goal nodes. We solved this by combining for each node all sequences of successive nodes, whose best path leads through the same neighbor. To farther reduce the number of necessary entries, we sorted them according to their representation in the k-d-tree.

Another challenge was the missing connectivity of the data. Some streets used different nodes than others to represent the same position. Others formed sub-graphs, which were not reachable from the main street map, e.g. in parks. We approached the former problem by combining nearby nodes and the latter by removing the sub-graphs.

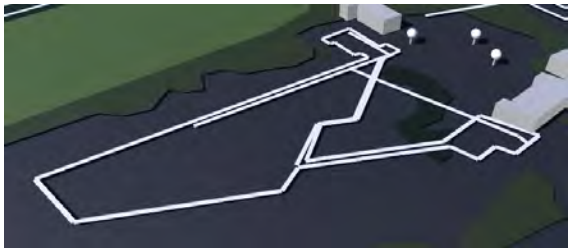


Figure 24 These streets are not connected to the main graph.

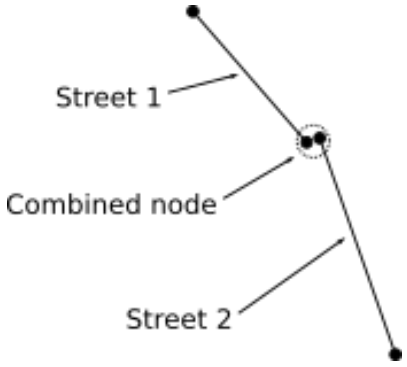


Figure 25 The nodes between the two streets are combined to a single node.

5. Playtest Report

5.1. Playtest

After finishing our alpha release, we went on to do some informal playtesting with other people to get further feedback for improving and polishing our game. 13 people participated in our playtesting in total, where the majority where male. We achieved to interview a broad spectrum of different participants in terms of age (19-31) and gaming experience as seen in the following figures. Most of them had at least some experience with playing games and also with strategy games.

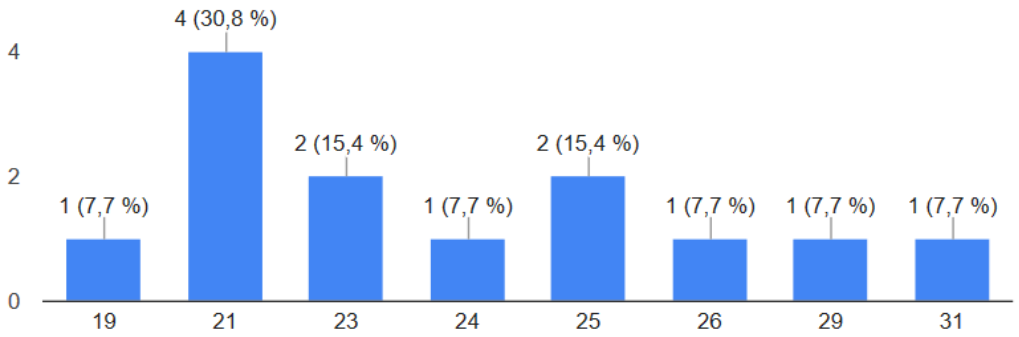


Figure 26 Age distribution of the participants

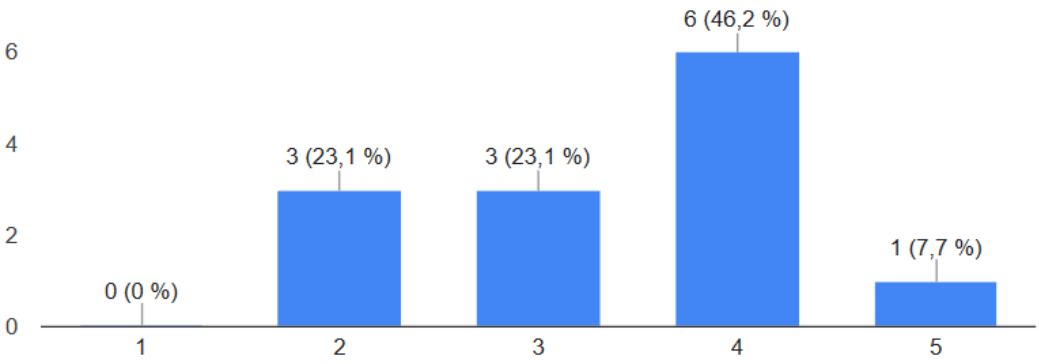


Figure 27 Gaming experience of the playtesters

The play tests are done individually as it's difficult to gather all people at the same time in the same place. Our game is supposed to be singleplayer, so groups wouldn't necessarily have a positive impact on the gameplay experience. We split the playtesting into several parts, guided with a survey to fill out for the testers:

Pre-Game Before playtesting the game we first ask them fill out the first part of our survey

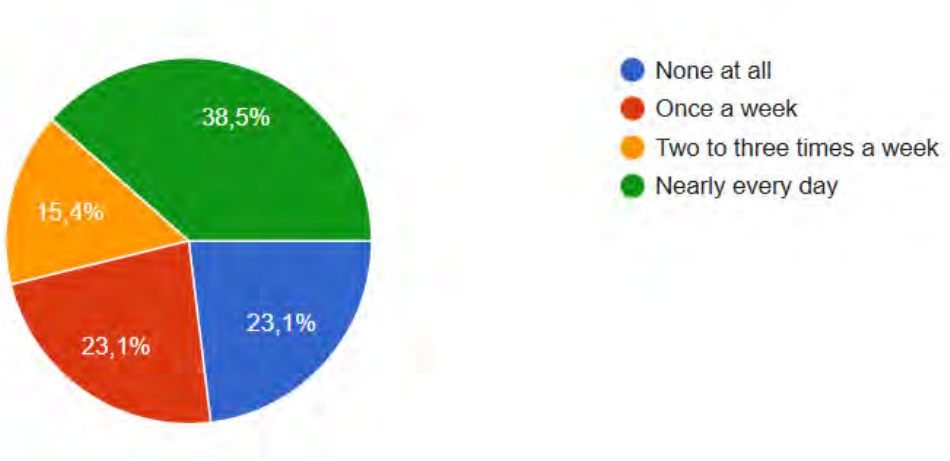


Figure 28 How often the participants are playing games

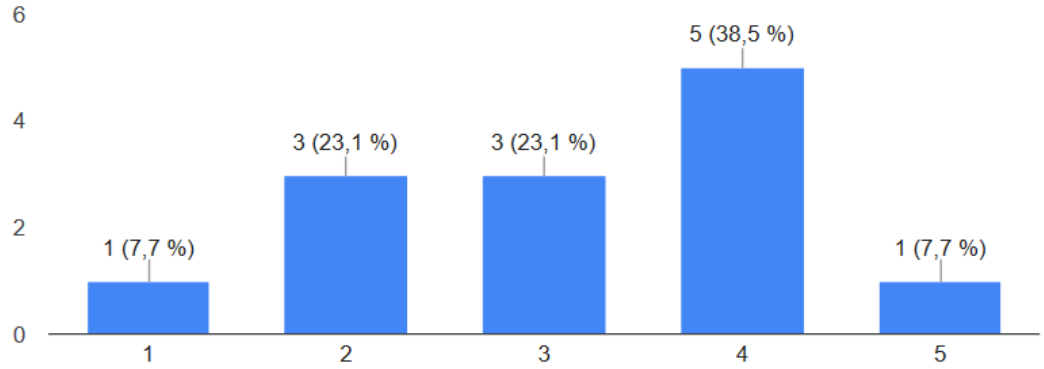


Figure 29 Experience with strategy games of the playtesters

which includes general questions concerning age, gender and experience with games. Additionally their expectations about the game based on hearing the title and the game genre.

Playing In the next step the tester are supposed to play the game for at least 5 min or longer up to their preferences (actually everyone played at least one round). During the playing phase we took notes about their behavior and decision making without helping them or giving further advices. Therefore we encouraged them to speak out loudly during playing.

Post-Game After playing the game the testers were asked to fill out the second half of the survey. In this section the participants are meant to reflect their experience and evaluate their liking of different game aspects. The aspects are divided into several categories in which the players should rate their experience with this aspects, what the liked and disliked most and suggestions for improvement. We defined the following categories:

- Gameplay
- User Interface
- Simulation

- Graphics
- Sound
- General (difficulty, general liking of the game, etc.)

5.2. Results

This section will describe the results of the playtest. Therefore, we will summarize the answers of the playtester and highlight overall trends. The structure of the this section will be analogous to the survey. A score of zero is the minimum that can be reached and corresponds to the answer "'not at all"'. A score of four is the maximum and corresponds to the answer "'very much"'.

5.2.1. Miscellaneous

The first impression of the game was overall positive. Some people mentioned that they were overwhelmed by the amount of information to begin of the game. The following enumeration shows some of the answers for this question:

- Strategy game, clear limited area, good!
- takes a moment to understand gameplay, then good impression
- Lots of stuff going on but still doable
- Interesting, kinda funny

Furthermore, most people said that they found the game length just about right or too short. This corresponded to our experience playing the game. You can see the distribution in figure 30.

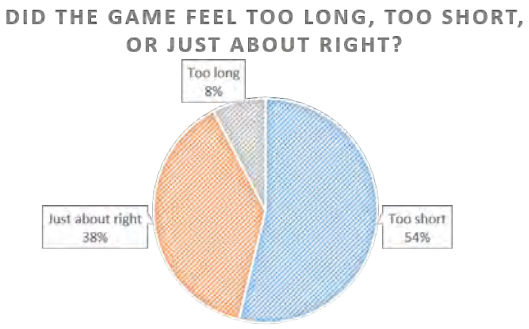


Figure 30 This figure shows how the tester answered the question about the game length.

The sound was received with mixed feelings. The music was liked by most players, the sound effects not that much. Especially the sound of the medics was criticized. The average score on the question if they liked the sound was 2.38 out of 4.

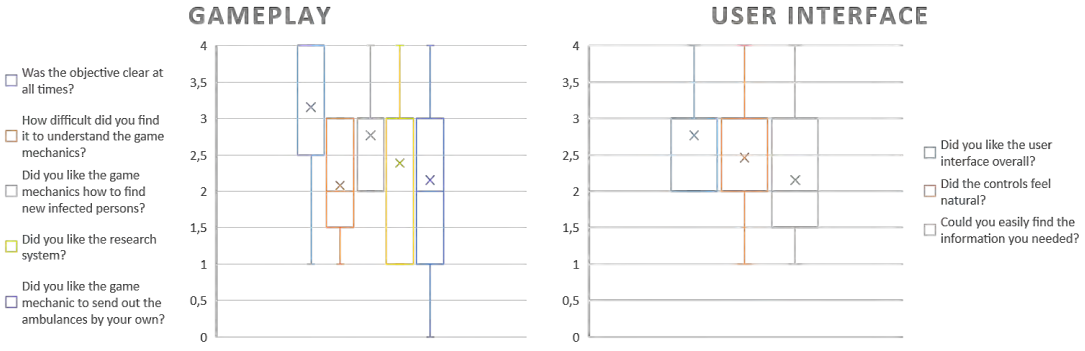
5.2.2. Gameplay

The objective of the game was clear for most of the players and the description of the objective given by the players were satisfying. The overall understanding of the game mechanics could be definitely better. Most player liked the mechanic to find new infected persons. The research system was received differently by the players. Some really liked it, though others did not like it that much. The system to send out ambulances was received quite the same way. An overview over the answer of the tester is given in figure 31.

There was a clear tendency in the game strategies to upgrade the universities to gain more research points and therefore to have more points to spend in research. Furthermore, some strategies included the use of the map to determine, where to send ambulances and place detectors. In the following enumeration some of the player strategies are listed:

- Upgrade unis to gain more RP. Then get the upgrades to slow down virus. Then getting the cure.
- Thinking about the most possible places for many inflected people (like stachus) and placing troops there
- Attack hotspots, do research
- Increasing the medic troop speeds to let them run faster and try to farm some spots.

To summarize this section, we are satisfied that most player did understand the game mechanics and what the objective of the game was. Nevertheless, we recognized that the research system could be enhanced by adding more specific attributes to it, which have greater influence on the gameplay. Also the system to send out ambulances was not received as positively as we wanted it to be.



(a) Diagram of gameplay questions

(b) Diagram of UI questions

Figure 31 This figure shows how the tester answered the question about the gameplay and the user interface. 4 points was the maximum score and zero the minimum.

5.2.3. User Interface

The results for the user interface were quite good. Overall most of the player liked the user interface. Also the controls felt natural for most of them. Only in the area of information finding

the results were not as positive as in the other two areas, but still not bad. You can see more details in figure 31

Some of the positive aspects that were mentioned by the players:

- simple and small
- Colour, Form and clearity of text
- Very intuitive, no learning curve to get started doing all the things you need to do to win
- easy access to all game mechanics

There were as well some negative aspects. Especially the sound of the medics were criticized, but as well that some buttons were not totally clear at the beginning. Also one person mentioned that it was not satisfying to control every medic individually. Some people would have liked it to have a better overview of the game world.

Furthermore, the playtester had a lot of suggestion what things could be added to the user interface. For example it was suggested to include a minimap, a list of hospitals and medics, display all cost directly by the element, add more feedback and make the closing of the windows easier.

Overall, we were thankful for the feedback and happy that the user interface was received quite positively. In the area of player feedback and information we recognized that there were still some things that should be improved for the final version.

5.2.4. Simulation

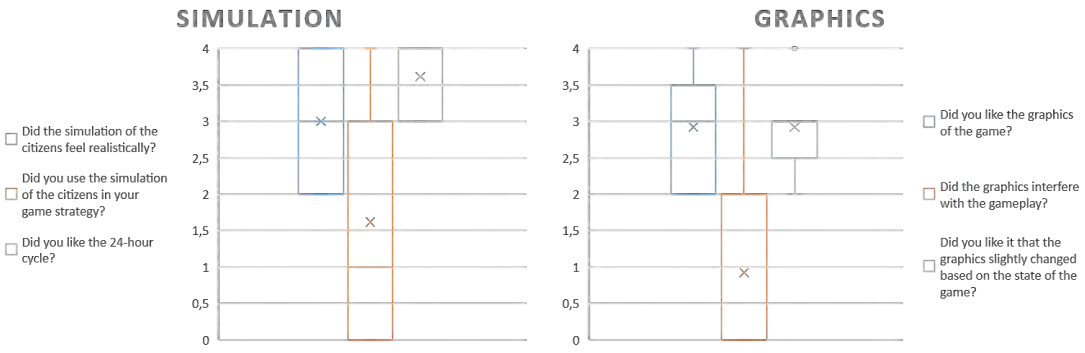
This part of the survey was quite interesting for us. How would the player like the simulation part of the game? All in all, the simulation of the citizens felt quite realistically for most of the player. All player liked the 24-hour cycle, that we had implemented in the game. The average score was the highest over all questions with 3.62 out of 4. The only downer was that many players did not use the simulation of the citizens in their game strategy, which we think was also a result that the game was a little bit too easy and that the people played the game for the first time. In figure 32 you can see in more detail, how the tester have answered.

The players mentioned positively that the set up felt realistic and alive, and that over night less people were on the streets.

Some of the negative aspects mentioned were that it was difficult to find infected at the beginning of the game and that too many people were displayed, which made it sometimes confusing. Also one tester mentioned that it seemed there was no way to avoid a widespread outbreak and that it would be nice to have more options to really stop the pandemic.

They were also some suggestion what we could improve in the simulation. For example to add residential and living areas, to reduce the map size, to simulate traffic more realistic, and maybe to change the colour of the citizens.

In the end, we are really happy how the simulation turned out. Most people liked it and especially the decision to have a 24-hour cycle was validate by the good score. Still we think that with more time we could easily enhance the simulation to make it even better and therefore to increase its importance in the game strategy of the player.



(a) Diagram of simulation questions

(b) Diagram of graphics questions

Figure 32 This figure shows how the tester answered the question about the simulation and the graphics. 4 points was the maximum score and zero the minimum.

5.2.5. Graphics

Overall, most people liked the graphics of the game. Most tester said that the graphics did not interfere with the gameplay, which was quite important for us. They also liked that the graphics slightly changed based on the state of the game. The results of the questions are shown in figure 32 in more detail.

Some of the favourite aspects of the players are shown in the following list:

- it was a simplistic image of munich... wich is nice
- Munich was clearly identifiable
- Night phase looked pretty cool
- No difficulties for me to recognize all the relevant places

Some negative aspects mentioned by the playtester were that they not liked the colours and the design of the people, that medics were not clearly identifiable as medics and that the game was sometimes too grey. Suggestion for improvements were to add more details, have stronger contrasts and add information on transportation system.

In summary, the feedback for the graphics was very positive and most of the playtester liked the style of the graphics that we have chosen. Only the models of the ambulance and the

citizens were criticized to some degree.

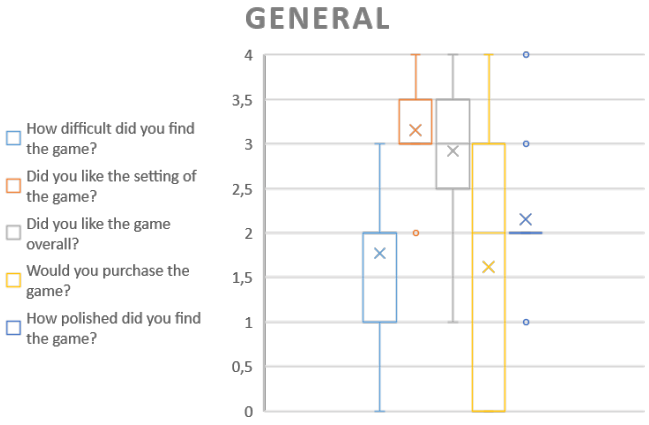


Figure 33 This chart shows how the tester answered the general questions about the game. 4 points was the maximum score and zero the minimum.

5.2.6. General

In total, nobody found the game too difficulty. For some it was too easy, for some just about right. The setting of the game was liked by all players. A very good score of 3.15 out of 4 was reached. Some player answered that they would very likely purchase the game, where others would purchase the game not at all. Overall, most people found that the game was polished to some degree.

Most important, nearly all tester liked the game, which made us quite happy and satisfied. They gave us an average score of 2.92 out of 4. Most liked the core game idea and how the game worked. They definitely saw space for improvements and things that could be added, but mostly in the area of usability and not that much in the area of game mechanics. More details can be seen in figure 33.

Furthermore, the tester gave a lot of general ideas for improvements. Very often it was mentioned to improve the tutorial for the player, to give them more information and feedback. Also to add more content to the game and give the player more options what they can do in the game, like build new hospitals or add a scouting unit.

5.3. Changes

Based on the aforementioned critiques provided by the testers, we decided to change several things in our game.

First of all, many players noted, that it took some time to figure out many important aspects of the game, so we tried to improve our tutorial. Especially the system of the cure points required further explanation. To make this system easier to understand, we also added a small notification on top of the progress bar, once a new cure point was gathered.

We also changed some aspects of the user interface. The percentage, which is shown in the top center of the screen, now starts at 100% and reaches zero once the game is lost. Also, we modified the display of the count of currently healthy citizens so that it shows the count of currently infected citizens instead. Since the sound effect, which was played when sending a medic troop, confused some players, we replaced it.

Another point that caused confusion among the players, was the fact, that a message showed up, once a hospital was full, but it didn't say which one. Therefore, we gave a number to each hospital, which is displayed on top of them in the map view and mentioned in the warning message. Also, we now show the load of a medic on top of it. To make the game more comfortable to play, medics now return to the place where they stopped gathering infected citizens after they finished taking them to a hospital.

Figure 34 The number four is shown on top of this hospital.

Figure 35 This medic has already gathered one patient, it is able to load another two.

6. Final Release

6.1. Summary

The topic of the Computer Game Laboratory WS 16/17 was 'Munich' and we had the idea to develope a real-time strategy game, which is based on fighting a deadly pandemia in Munich City. As technical achievement we aimed for a realistic crowd simulation coupled with virus spreading. Overall we managed to translate our inital concept into a final game. Core features and some high level features (e.g. highscore, dynamic graphics, tutorial, etc.) have been successfully implemented according to our development schedule.

In the last week only did minor changes including bugfixes, adding an event backlog, polished some parts of the UI and gameplay, and added a few image effects. Most aspects, which have been mentioned during the playtesting sessions, haven been already addressed in the playtesting chapter, where a list of the adjustments is provided, which we refer to for further insight concerning the changes from the alpha release.

You can see several screenshots from our final game below, showing different states throughout the game.

Figure 36 Early stage of the game, where the virus hasn't spread very far. Showing the 'Hospital' page of the Research tree

Figure 37 The pandemia is slowly taking over the city. Trying to search with detectors and medics for infected citizens to treat them in the hospitals

Figure 38 Endgame scenario: Almost reaching the threshold at which virus will spread unstoppable over the whole city. The cure needs to be found before we are reaching 0% (Hint: Looks like we are losing the game this time)

6.2. Experience

In the first project stage we developed the main game concept, which we kept over the whole course. Even though our concept has been settled the main game mechanics weren't clear and heavily discussed. Building the physical prototype helped us to test the different approaches and lead us to the core gameplay mechanics we implemented in the final game. Getting valuable feedback from the course participants allowed us to polish the game concept furthermore.

Writing a development schedule in advance to plan the progress for the whole course was pretty challenging, but we were able to follow it in the most parts. In reality, certain aspects required longer time than expected and multiple iterations to reach a certain quality level and performance. Having incremental milestones definitely helped us to keep everything structured and not planning too much. During the course we were able to reach each milestone in time, which is quite good experience. Doing weekly meetings in team also helped us to keep on track and focusing on the progress.

Playtesting the game by external persons, who wasn't involved in the development or the course, was a very important step for polishing and improving the game. Especially aspects like usability and learning curve are difficult to estimate if you are already familiar with the game due to the development.

Overall, the structure supported us in developing the game quite well without being too compressed or strict.

6.3. Personal impression

We are happy with the game we made in this course. While there wasn't that much time, it was enough to create a small game. On the one hand, the compact schedule is a challenge, which helps to tackle such problems in future projects. On the other hand, it helps to avoid to get stuck into some aspects for too long and forces one to go on. The biggest difficulties we had were to extract the relevant parts of the open street map data, to get the simulation of the citizens to work so that they behave relatively natural and to optimize the performance, which required a deep understanding of how the Unity game engine works. Even though this year's theme "Munich" leaves very much freedom and is at the same time challenging to use as theme of the game instead of just as a label, it functioned as an impulse for finding a game idea. Without such an impulse, it would have been much more difficult to come up with an idea.

We consider the project as a big success. Not only are we happy with the result of our project, but more importantly, we learned a lot about how to plan a game and what must

be considered in the process of creating it. Also, we learned much about how Unity and game engines in general work. We are confident, that the numerous technical challenges we faced during the project will help us in future projects. The greatest success for us, that we overcame all technical challenges, so we could finish in time and the play testers liked the game. We mostly met our project plan and milestones, but due to design changes we laid more focus on some aspects than on others. Also, most tasks were done by other group members than specified in the original plan. In our next game project, we could apply the experience we made with this game to make a better plan. Also, we would start executing play-testing sessions at an earlier point in time.

For the course organization, we suggest to make the project structure document clearer in some aspects. A clearer grading scheme would help to indicate what one should focus on in the development of the game. Also, the wiki could be used more intensely. It could also be helpful to receive more feedback about whether the project progresses in the right direction. This could be direct feedback from the instructors or additional feedback rounds throughout the development phase.

List of Figures