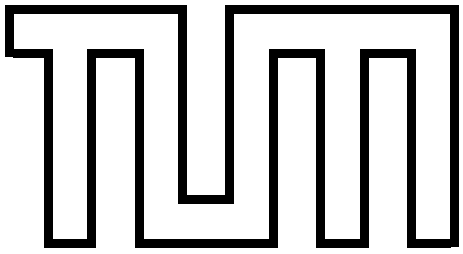


Technische Universität München

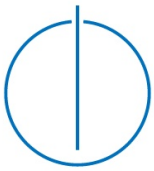
Department of Informatics

Master Practical Course Games Engineering
Augmented Reality

Rune TD

Serious Celtic Augmented Reality Tower Defense Game

Summer term 2018



Authors

Stefan Stark, Florian Hösch

Dennis-Florian Herr and Michael Felleisen

Supervisors

Prof. Gudrun Klinker, Ph.D.

Dipl.-Inf. David A. Plecher, M.A.

Contents

1 Introduction **2**

2 Game Design **3**
2.1 General Concept . . . 3
2.2 Runes and Rune Markers . . . 3
2.3 Tower Defense . . . 4
2.4 Campaign . . . 5
2.5 Historical Reference . . . 6

3 Implementation **6**
3.1 Marker Tracking - Vuforia . . . 7
3.2 Marker Generation . . . 7
3.3 UI . . . 7
3.4 Rune Combining . . . 8
3.5 Voxel Framework . . . 8

4 Encountered Problems **9**
4.1 Marker tracking . . . 9
4.2 Marker printing . . . 10
4.3 ARCore/ARKit . . . 10

5 Userstudy **10**
5.1 Study methodology . . . 10
5.2 Study evaluation . . . 11

6 Future Work **12**
6.1 Free Play Mode . . . 12
6.2 Wearables . . . 13

References **14**

Appendix **16**

A Appendix A **16**

B Appendix B **18**

C Appendix C **23**

1 Introduction

"The year is 50 B.C. Gaul is entirely occupied by the Romans. Well, not entirely... One small village of indomitable Gauls still holds out against the invadors. And life is not easy for the Roman legionaries who garrison the fortified camps of Totorum, Aquarium, Laudanum and Compendium...", Robert Steven Caron. [1]

Similar to the famous opening quote from the Asterix comics, our game is set in the middle of the last century before Christ. The Celtic culture is at its peak and shaped Central Europe and the British Isles during the last centuries. [2] But conflict arises between the Celts and the Romans when Gaius Julius Caesar is leading his military campaigns northwards to conquer Gaul, known as the *'bello Gallico'*.[1]

During his campaigns, Caesar encounters several *'oppidum'*, fortified Celtic villages or cities. [4] The player is taking control over the defenses of a small Celtic *'oppidum'* and has to defend the village using powerful Celtic runes. After the village is fortified enough to fend of wildlife, the player has to fend of wave after wave of the Romans that are trying their hardest to conquer the village. Who will come out on top? The organized roman empire or the naturalistic Celts with exceeding knowledge in ancient runes?

The game is supposed to teach players the meaning of ancient Celtic runes in a playful and engaging way. While playing the campaign players gradually unlock runes, that they must use with physical, printed markers[2] to control the augmented reality world. When a rune is unlocked the player gets educational information about the rune and an explanation of its effect in-game. The original meaning of the rune is related to their in-game effect to improve the learning experience by creating coherence between the educational information and the game experience.

We chose knowledge about Celtic runes as the educational goal, because the Celtic culture is an important part of Europe's history, but little is taught about Celts and especially about runes in the general education system. So even though most people might have heard about 'runes' through games, there is little to no knowledge about their original usage and meaning.

[1] Celtic and Gallic are for our purposes synonyms. Celtic comes from the Greek term *'Keltoi'* whereas Gallic comes from the Latin term *'Galli'* and refer to the same folk. [3] But as Gaul was also used by the Romans to describe an area that is mostly equivalent to what is now France, we stick to the terms Celts and Celtic instead of Gauls and Gallic to describe the people and culture that spread throughout Central Europe.

[2] Our system of rune markers is explained in Chapter 2.

The following chapters are meant as a documentation for our prototype implementation. We start with describing the game design choices, followed by an explanation of the mechanics used to play the game. Afterwards we discuss problems during development, especially in regard to augmented reality and present a short user study of our prototype as presented at the TUM Demo Day of the summer semester 2018. Finally we discuss future work that was out of scope of the practical course.

2 Game Design

This chapter covers our game design in regard to game play and the educational aspect.

2.1 General Concept

Our game design evolves from the idea of a serious game utilizing augmented reality techniques available on larger mobile devices, i.e. tablets or phones with a display size of at least 7 inch. The educational goal is to convey knowledge about Celtic runes and culture through an engaging and entertaining game experience. Learning is supported by a connection between the matter and the game mechanic, so the player improves his skill in the game by mastering knowledge about runes and vice versa.

The game is designed as a classical Tower Defense (TD) game. Players build towers to defend their city from increasingly difficult hordes of enemies. This is explained in more detail in Section 2.3. We use augmented reality to give the player control over the towers and buildings with printed physical markers, our 'rune markers' (cf. Section 2.2). This is meant to increase the learning success by bringing the runes into focus while providing a refreshing game experience through novel control mechanics. Knowledge about runes and Celtic history is presented gradually using a game campaign, where the player unlocks new runes and additional information after increasingly more difficult levels. This is thoroughly explained in Section 2.4.

Finally we based the style of the game on available historical data about Celtic cities, defenses and enemies, which is covered in Section 2.5.

2.2 Runes and Rune Markers

As the primary goal of playing the game is to learn about Celtic runes, the runes are the core of our game experience. Therefore we split the game into two distinct phases: A

preparation phase where the player interacts with physical representations of the runes, our rune markers, and a game phase where the player interacts with the augmented reality device.

Our rune markers are printed out representations of runes that each have a unique effect in the game and need to be properly positioned by the player. Figure 2.1 shows a rune marker for the *'Algiz'* rune, which is used to build and position towers in-game. The player can use some runes multiple times, since the rune markers have an additional unique border that enables the augmented reality tracking to distinguish the same rune on multiple rune markers.

Figure 2.1: An Algiz rune marker for building towers

As the basis for our runes and rune markers we used the 24 runes of the *'Elder Futhark'* [5], the oldest form of the runic alphabets. Almost[3]all of the 24 runes have a special meaning in the game that closely relates to their original meaning. For example the Algiz rune has the meanings *'Elk', 'Protection', 'Defense'* [6] and is used to build towers. Though, it must be noted here though that runes can have multiple meanings and the meaning is not always absolutely clear. Rieckhoff and Biel point out that there is no Celtic historiography, literature or religious writings. [7] The absence of a large basis of usage, makes interpreting runes especially difficult. A complete table of the runes we used, the original meaning we used as a basis, and their usage in-game can be found in the Appendix A.

To properly play the game a player must recognize a rune and it's function in-game. The player has as much time as he needs to study, recognize and properly place the runes in the preparation phase.[4]In combination with the coverage of most runes in-game and a close relation of their traditional meaning to the in-game usage, this measures support the learning success of the player.

2.3 Tower Defense

We use the basic and popular tower defense concepts for our core gameplay mechanics. Towers are manually placed on the map (through the position of the rune markers relative

[3]Unfortunately, we had to skip a small number of runes as there was no reasonable way of mapping the rune to an in-game functionality.

[4]Though this can help him learn the runes, this was also a necessity due to the nature of augmented reality on a tablet computer.

to the augmented reality map) and shoot incoming waves of enemies. The enemies have to be killed before they can reach the village, which is positioned with the *'Mannaz'* rune marker and the anchor point of the augmented reality map. If enemies reach the village, players loose health. When health reaches zero, the game is lost and the player has to start over.

Players can upgrade their towers by placing buff runes close to their tower rune. For example the *'Kenaz'* rune adds fire damage to the tower it's linked to. As enemies are vulnerable to different damage types, e.g. a wooden siege weapon is vulnerable to fire damage, players have to carefully consider their elemental upgrades. Additionally players can earn gold by building farms and harvesting them during the game phase, which adds additional interactivity to the game phase. The gold can be used to buy additional lives during the preparation phase.

Although, we would've liked to implement advanced mechanics we decided to stick with this basic tower defense principles, which works with our augmented reality setup and is intuitive to use, because of the difficulties encountered (cf. Section 4).

2.4 Campaign

The game as well as the learning process is structured into levels with increasing difficulty. At each level the player unlocks new runes, gets information about the runes historical meaning and it's use in-game. E.g. in the first level the player unlocks the *'Mannaz'* and *'Algiz'* rune so he can build the village and his first tower. In the second level he unlocks another *'Algiz'* rune for a second tower and a *'Jera'* rune to build a farm he can harvest during the game phase. The player gets textual information in a dialog window and an enlarged image of the new rune. He has to immediately learn to recognize the rune to be able to find the rune marker with the rune and use it to construct or enhance buildings. For additional references the player can open a *'wiki'*, where he can find all information about runes and the game in a structured format. We chose this structured campaign approach, because we quickly realized while testing that 24 runes with different meanings is overwhelming when directly available and a slower introduction in small manageable steps yields better results.

Each level also spawns new and tougher enemies, so the player is forced to use his runes wisely to beat the level and advance in the campaign. After beating the campaign, a free-play mode is unlocked to offer a challenge with all runes available.

2.5 Historical Reference

We created a mostly historical accurate setting for the game. Because we used a self-made voxel engine (cf. Section 3.5 to render our game, we were able to create the game assets by ourselves.

Our Celtic village is based on reconstructions of the *'Oppidum of Manching'* as illustrated in [8]. The tower and stone walls were modeled after historically funded illustrations from [8][9][10][11]. A selection of these illustrations can be found in Appendix B. As enemies we created attackers that the Celts may have actually encountered. They get attacked by wolves and bears in the beginning and later by invading Roman foot soldiers and siege weapons. The attackers may also be accompanied by mythical creatures the Celts believed in like the Kelpie, a equine water spirit.

Figure 2.2 shows a comparison between our house asset and the illustrated reconstruction from Manching as found in [8].

Figure 2.2: Our house design compared with an illustration from Manching.

3 Implementation

This chapter gives details about our implementation and background information about design decision we made for technical reasons.

3.1 Marker Tracking - Vuforia

For marker tracking we are using the Vuforia Augmented Reality library for Unity. [12] It is free for non-commercial use very similar to the Unity engine. To integrate your own markers in your Unity application you have to upload your marker images to the developer portal of Vuforia and download the resulting Unitypackage. The library is fairly easy to use and offers great functionality for marker tracking and markerless tracking. The only disadvantage it has is a maximum number of uploadable markers which is at a hundred. With 96 different markers we came very close to that limit.

3.2 Marker Generation

Since we chose Vuforia for marker tracking, all markers had to be unique. The game design however requires multiple markers for each rune, as most buildings and upgrades can be in play multiple times. Furthermore the markers have to be readable and unambiguously assignable to a specific rune by the player.

Our solution was the marker design seen in Section 2.2. The rune represented by the marker is placed in the center and fills most of the space. In order to make them unique, the marker borders are seamed with small versions of all 24 runes in random order. Because many markers had to be created like this (4 per rune) we created a Java Script program to do the random placement of the rune images. This design has the additional advantage of always giving the player an overview of all runes, as well as making the correct orientation of the center rune more obvious.

3.3 UI

To create the games' user interface we used MarkLight.[13] MarkLight is a Unity framework offering a declarative design language similar in syntax to HTML. It can be used to create user interfaces with code only, bypassing the need to drag and drop UI-elements in the Unity scene view.

In order to further separate the UI from the core code, most text is handled using separate JSON files. This includes primarily the Wiki, as well as the campaign flavour text. Fortunately MarkLight supports the Unity Rich Text markup format, so we used it to make the partly huge walls of text easyer to read.[14]

At this point it should be noted, that all the content presented in the in-game wiki is taken from the wikipedia page corresponding to the rune, as well as the overview page of the elder futhark.[5]

Another important job the UI had to fulfill was showing the player if one or several of the markers has lost tracking. For this purpose every marker detected by Vuforia gets a colored sphere attached.

- green means the marker is actively tracked
- yellow means the marker is tracked using extended tracking only
- red means tracking was lost or position is invalid

It is important, that all runes the player wishes to use have at least a yellow tracking level before starting the level.

3.4 Rune Combining

There are 3 types of upgradeable runes: towers, farms and the village rune. All other runes are either elemental (which are used to change the damage type of towers) or upgrade runes that change the behavior.

Every upgrade rune has a list of valid upgradeable runes which it can connect to. Out of that list it will select the closest one if it is not out of range. This is visualized by a line visible through the augmented reality view.

Every upgradeable rune will collect a list of all connected upgrades when the START button is pressed so they are not changed while the game is running. This list will be consulted to identify the behavior and stats of the upgradeable runes.

3.5 Voxel Framework

We decided to use a Voxel Framework as the base for our project for the following reasons:

- We had previously build a core Voxel Framework and we wanted to build upon and extend the existing framework. By using our own framework, we could also understand what's happening from the core and learn more.
- It is easier to generate nice looking assets for a voxel style game.
- Objects from the Voxel Framework can be easily and efficiently altered at runtime (e.g. creating a farm on the terrain changes the grass to dirt.)

Most objects that are used in the game, such as the towers or the terrain, are displayed by the Voxel Framework. The data for every object is stored in a 3D grid. Every data point stores the information about the material at its position. From this grid we generate

one optimized mesh that uses a Texture Atlas and UV coordinates to display the color information of the objects.

The terrain is split into multiple chunks to reduce the complexity of the mesh optimization and therefore improve the performance. Every chunk has its own mesh but all share the same materials and textures to reduce the amount of draw calls.

Besides the rendering the Voxel Framework also includes basic terrain generation which we adapted to our needs for this project. Furthermore we implemented growing plants on farms.

4 Encountered Problems

During development we encountered multiple problems, which are described in this chapter.

4.1 Marker tracking

We had issues with the performance of the Vuforia tracking library in the early stages of development. The tracking was not stable enough for the amount of markers we intended to use, so we had to change a few aspects of our game.

The first thing we scraped were enemy path markers. The original idea was to let the player decide where the enemies run. To reduce the amount of markers, we instead used the classical tower defense approach and created the enemy path for the player relative to his village. This solved stability issues as well as problems with very efficient path patterns made by the players. By removing this player variable we also had an easier time creating a balanced campaign mode.

The second thing we improved were the markers themselves. The first hand-drawn markers did not bring the image recognition performance we needed. That's why we decided to create the markers using a script (cf. Section 3.2) and let the player print them out. By doing that, we could use multiple different instances of the same rune markers and improved the effectiveness of the tracking.

4.2 Marker printing

The first time we printed all of the – close to 100- markers, we used a very simple printer and very basic printing paper. This resulted in white vertical lines across most of the markers, which was very bad for the image tracking process. Moreover, after using the markers a few times, they started to bend which made the tracking even worse. After that we used an industrial printer with very thick paper and got much better results in tracking and marker durability in general. This might be a problem, when players have to print the markers for themselves.

4.3 ARCore/ARKit

ARCore and ARKit was anounced in August 2017 and promised heavy performance boosts in the markerless tracking technology. Those advances in tracking stability could have been a great benefit for the game. Unfortunately, the support release for phones that are not flagship level (like the Galaxy S9 /iPhone 8) is still in the making. Dynamic environmental lighting and AR plane detection were the features we especially looked forward to.

5 Userstudy

To evaluate our game we conducted a user study during the Demo Day. This allowed us to get a more or less diverse group of 12 test users, that were willing to take some of their time to fill in our questionnaire.

5.1 Study methodology

Our user study was done in two separate steps. Before playing the game participants filled out some classification information (age within a range and their gender) as well as information about previous knowledge on a scale from 1 to 5. We asked about familiarity with:

- 3D Movies
- Videogames (PC)
- Videogames (Mobile)

- Augmented Reality (AR)
- Virtual Reality (VR)
- Mixed Reality (XR)
- Celts
- Celtic Runes
- TowerDefense Games

After playing the game we asked players if they liked the game and how much they thought they learned about runes on a 1 to 5 scale. Finally we let them fill out a six-question test to see how much they remembered about runes after playing the first few levels. The test consisted of remembering the name of three different runes and drawing three other runes based on their name. It's important to note that participants did not know they will be tested after playing the game, so they did play for fun not for knowledge. Unfortunately, but understandably in the Demo Day environment, not all players had the time to play long enough to unlock two runes asked for in the end of the test, so our results are a little bit flawed in that regard and learning success might actually be better. Lastly, the participants had the option to suggestion for improvement in a free-text field at the end of the questionnaire.

5.2 Study evaluation

The majority of participants were very familiar with AR/VR which is most likely due to us handing out the questionnaire during Demo Day, the most relevant event for games engineering students. On a scale from 1 (bad) to 5 (good), previous knowledge about Celts was self-rated at 1.92. After playing the game the self-rating on how much the test users learned about the Celtic culture was at 2.5. The test group was able to score 3.3 out of 6 possible points in the drawing quiz part. Overall, most of the testers liked the general tower defense concept with a score of 3.9 and liked our game with a score of 4.1. So the Celtic aspect increased the fun the testers had while playing the game.

Figure 5.1 illustrates how much participants learned based on their previous knowledge. We see that for the majority of knowledge groups there was an increase in perceived knowledge after playing the game. No participant knew less about Celtic runes than before playing the game. Figure 5.1 also shows a positive correlation between the perceived learning success and the test score of participants. Although it must be noted here, that we need a higher number of participants ideally from a more diverse group to draw sound conclusions. The raw test results can be found in Appendix C.

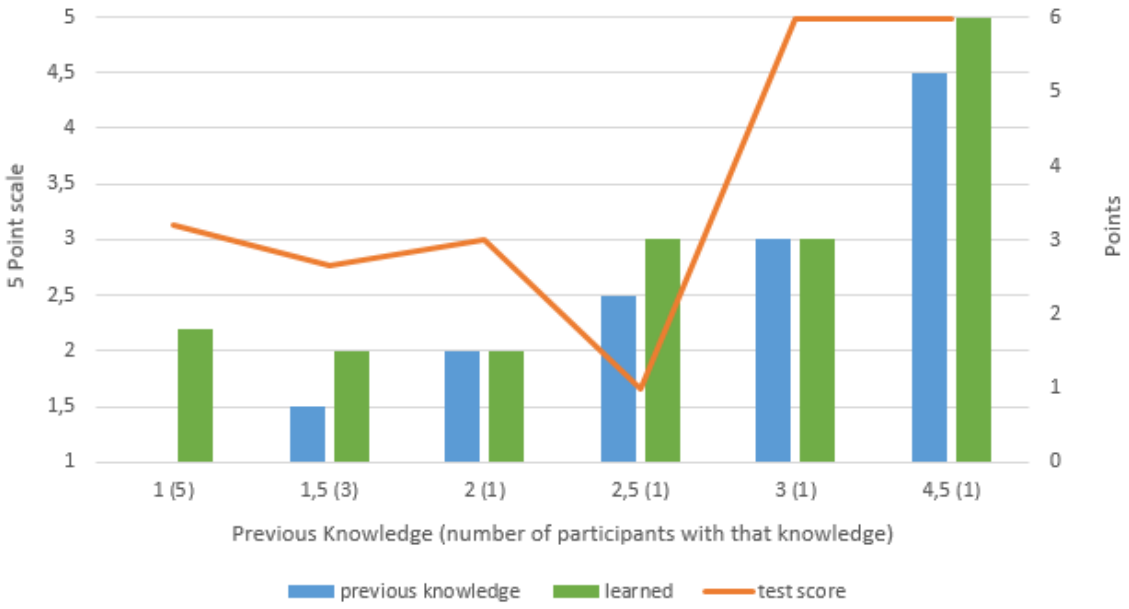


Figure 5.1: How much did participants learn based on previous knowledge.

From the free-text suggestions, the most asked for improvement was to reduce the amount of unlocked runes for every level in the campaign mode. For Demo Day we did not create a fully fleshed out campaign. We tried to keep the campaign relatively short to make it playable in a shorter amount of time, to make the game not as time consuming as it would be in the final version, but this was partly overwhelming for our players. Furthermore, the marker-tracking performance was an issue for some testers despite our efforts to make it as stable as possible. Possible ways of improving AR further would be using a better device (computing power and better camera) and/or using advanced AR-libraries (ARCore/ARKit).

6 Future Work

6.1 Free Play Mode

The Free Play Mode was supposed to be played after completing the campaign. It would be designed to test and strengthen the knowledge gained about runes, their meaning and how to use them most efficiently.

To achieve this, the player would fend off randomly generated waves of enemies with

all runes unlocked. Before each wave starts, information about the type of attackers is displayed in form of rune symbols and/or names. If the player was shown the runes "Ehwaz" (speed) and "Kenaz" (torch) for example, he would know to expect a wave of fast enemies with fire element and could adjust his strategies accordingly. This mode could easily be implemented as a two player mode as well, with one player creating the waves of attackers using markers and the other fending them off.

6.2 Wearables

For a game like this, wearable Head Mounted Displays (HMD) could enhance the player experience quite a bit. HMDs like the Microsoft HoloLens, Oculus Rift, the HTC Vive or Google Cardboard couuld be used to increase immersion. The usage of HMDs with the Unity engine is relatively easy. Player input on the other hand is a difficult task. Using a simple mouse would break the immersion a lot. Using a controller would be acceptable, but the game requires the player to use their hands to place runes so the player would have to put down and pick up the controller constantly. Gesture recognition would be the optimal solution but is very difficult and time consuming.

References

[1] Robert Steven Caron. *Beginning of most Asterix comics*. 2018. URL: http://comedix.de/lexikon/db/vorwort.php (visited on Mar. 13, 2018).

[2] Bernd Liermann. *Herkunft der Kelten und Lage des keltischen Kulturkreis*. URL: http://www.antikefan.de/kulturen/kelten.html (visited on Mar. 13, 2018).

[3] Sabine Rieckhoff and Jörg Biel. *Die Kelten in Deutschland*. German. ausgeliehen TUM TB Stammgelände GES 410f 06.2004 A 159. Stuttgart: Theiss Verlag, Sept. 2001, p. 18. ISBN: 9783806213676.

[4] John Collis. "Die Oppidazivilisation". German. In: *Das keltische Jahrtausend*. Ed. by Hermann Dannheimer and Rupert Gebhard. 2., erweiterte Auflage. Vol. 23. ausgeliehen TU München Universitätsbibliothek 94 A 484. Prähistorische Staatssammlung München und Verlag Philipp von Zabern, Mainz, 1993, p. 102. ISBN: 3-8053-1514-7.

[5] Wikipedia. *Elder Futhark*. URL: https://en.wikipedia.org/wiki/Elder_Futhark (visited on Mar. 15, 2018).

[6] Wikipedia. *Algiz*. URL: https://en.wikipedia.org/wiki/Algiz (visited on Apr. 9, 2018).

[7] Sabine Rieckhoff and Jörg Biel. *Die Kelten in Deutschland*. German. ausgeliehen TUM TB Stammgelände GES 410f 06.2004 A 159. Stuttgart: Theiss Verlag, Sept. 2001. Chap. Geld und Schrift, p. 217. ISBN: 9783806213676.

[8] Werner Krämer. "Das Oppidum von Manching. Erforschungsgeschichte". German. In: *Das keltische Jahrtausend*. Ed. by Hermann Dannheimer and Rupert Gebhard. 2., erweiterte Auflage. Vol. 23. ausgeliehen TU München Universitätsbibliothek 94 A 484. Prähistorische Staatssammlung München und Verlag Philipp von Zabern, Mainz, 1993, pp. 107–109. ISBN: 3-8053-1514-7.

[9] Sabine Rieckhoff and Jörg Biel. *Die Kelten in Deutschland*. German. ausgeliehen TUM TB Stammgelände GES 410f 06.2004 A 159. Stuttgart: Theiss Verlag, Sept. 2001. Chap. Dannenfels, KIB (RP), p. 322. ISBN: 9783806213676.

[10] Sabine Rieckhoff and Jörg Biel. *Die Kelten in Deutschland*. German. ausgeliehen TUM TB Stammgelände GES 410f 06.2004 A 159. Stuttgart: Theiss Verlag, Sept. 2001. Chap. Preist, BIT (RP), p. 449. ISBN: 9783806213676.

[11] Sabine Rieckhoff and Jörg Biel. *Die Kelten in Deutschland*. German. ausgeliehen TUM TB Stammgelände GES 410f 06.2004 A 159. Stuttgart: Theiss Verlag, Sept. 2001. Chap. Pommern, COC (RP), p. 447. ISBN: 9783806213676.

[12] PTC Inc. *Vuforia*. URL: https://www.vuforia.com/ (visited on Apr. 10, 2018).

[13] Ex Makina AB. *MarkLight Homepage*. 2017. URL: http://www.marklightforunity.com/ (visited on Apr. 9, 2018).

[14] Unity Technologies. *Unity manual entry of rich text.* URL: `https://docs.unity3d.com/Manual/StyledText.html` (visited on Apr. 9, 2018).

Appendix

A Appendix A

Rune Table

Rune	Name	Meanings	In-game function
ᛉ	Algiz	Elk, Defencs, Protection	Build Tower
ᛒ	Berkanan	Birch, Birth, Earth	Element: Earth
ᛞ	Dagaz	Day, Light, Balance	Element: Light
ᛖ	Ehwaz	Horse, Movement, Speed	Tower: Attackspeed
ᚠ	Fehu	Cattle, Wealth, Dynamic Property	Farm: Cow Upgrade
ᚷ	Gebo	Gift, Trade	Tower: Money On Hit, Village: Damage costs Money
ᚺ	Haglaz	Winter, Widespread Destruction	Tower: Splash Damage
◇	Ingwaz	Wine, Growth	Farm: Wine Upgrade
ᛇ	Ihwaz	Link	Tower: Enable multiple different Elements
ᛁ	Isa	Ice, Freeze, Isolation	Tower: Slow On Hit
ᛃ	Jera	Harvest, Circle of Life	Build Farm
ᚲ	Kenaz	Fire, Torch, Ignition	Element: Fire
ᛚ	Laukaz	Water, River, Tears	Element: Water

Table A.1: Rune table for runes 1-14

Rune	Name	Meanings	In-game function
ᛗ	Mannaz	Human, Community, Intelligence	Build Village
ᚾ	Naudhiz	Distress, Hardship	Village: Villager defend Village, Tower: temp. SpeedUpgrade
ᛟ	Othanal	Heritage, Wealth, Static Property	Village: inc. Money Capacity, Tower: Money on Kill
ᛈ	Perthro	Luck, Destiny, Random	Random Upgrade Rune
ᚱ	Raido	Travel, Path	Build Path: obsolete
ᛋ	Sowilo	Sun, Warmth, targeted Power	Tower: Beam Upgrade, Farm: Sunflower Upgrade
ᚦ	Thurisaz	Giant, Thorns, Defense	Tower: AOE around Tower
ᛏ	Tiwaz	Sky, Honor, Discipline	Element: Air
ᚢ	Uruz	Healing, Health	Farm: Upgrade, Village: inc. max. HP
ᚹ	Wunjo	Happiness, Harmony	Multiple Elements, Farm: Upgrade for Sunflowerfarm

Table A.2: Rune table for runes 15-24

B Appendix B

Illustrations

Illustrations as found in [8].

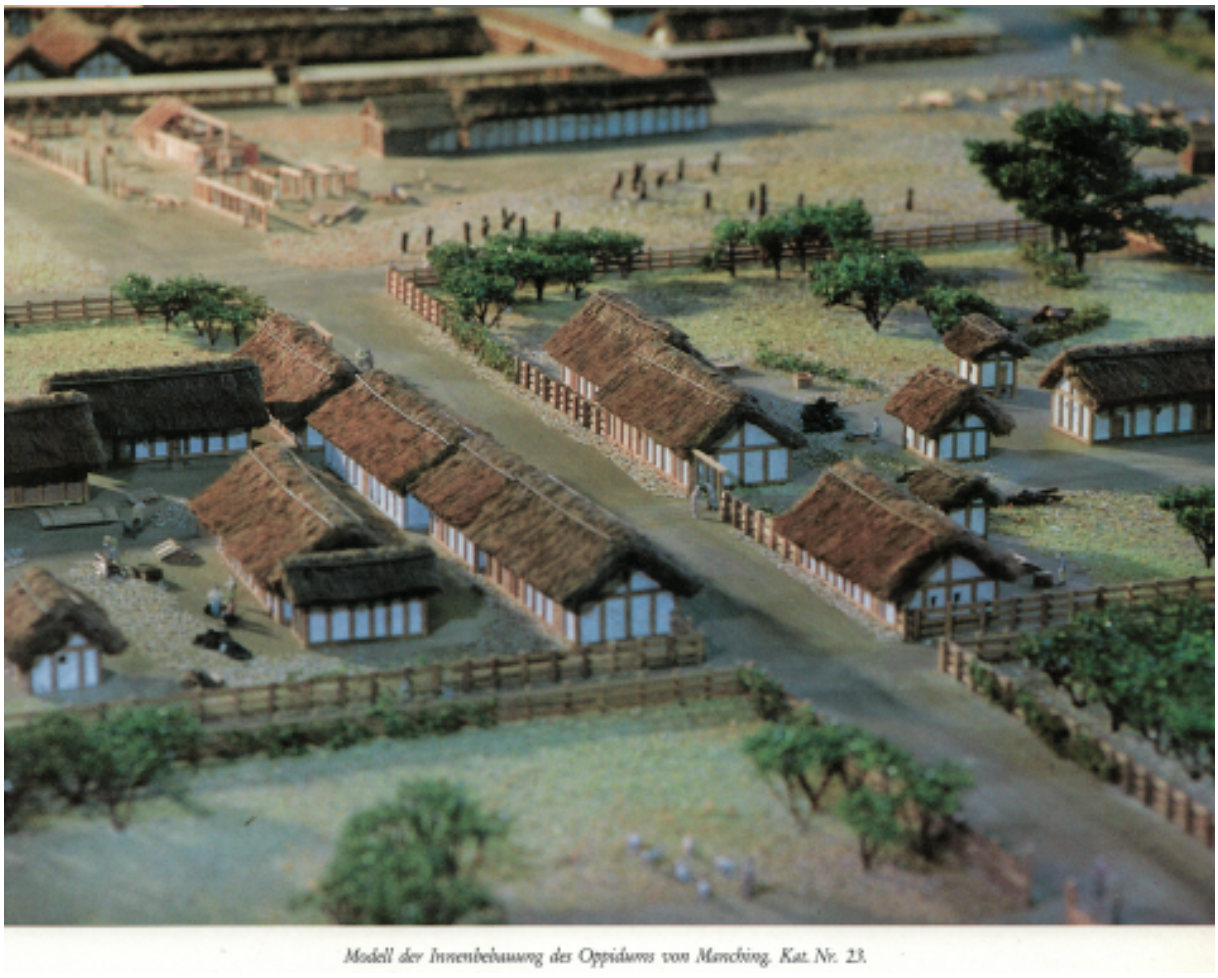


Figure B.1: Model of the inner buildings of the Oppodium of Manching.

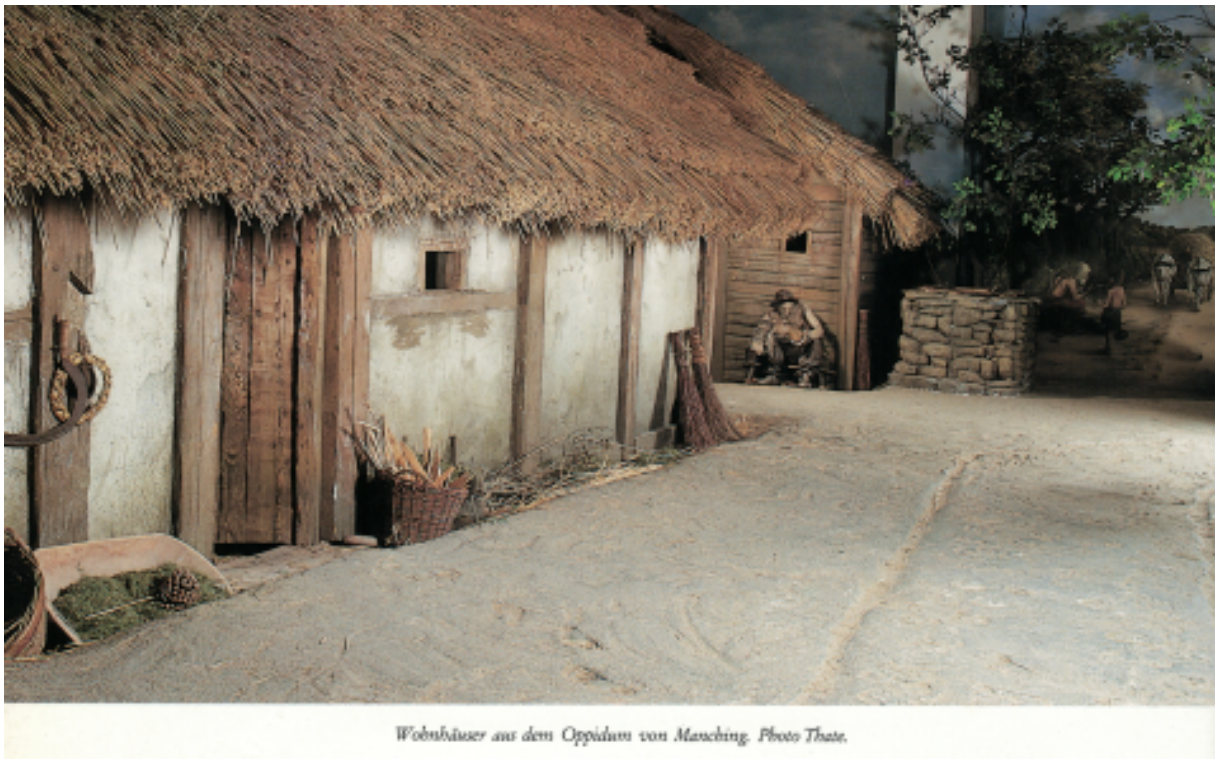


Figure B.2: A illustration of a living house.

Figure B.3: Construction of a wall.

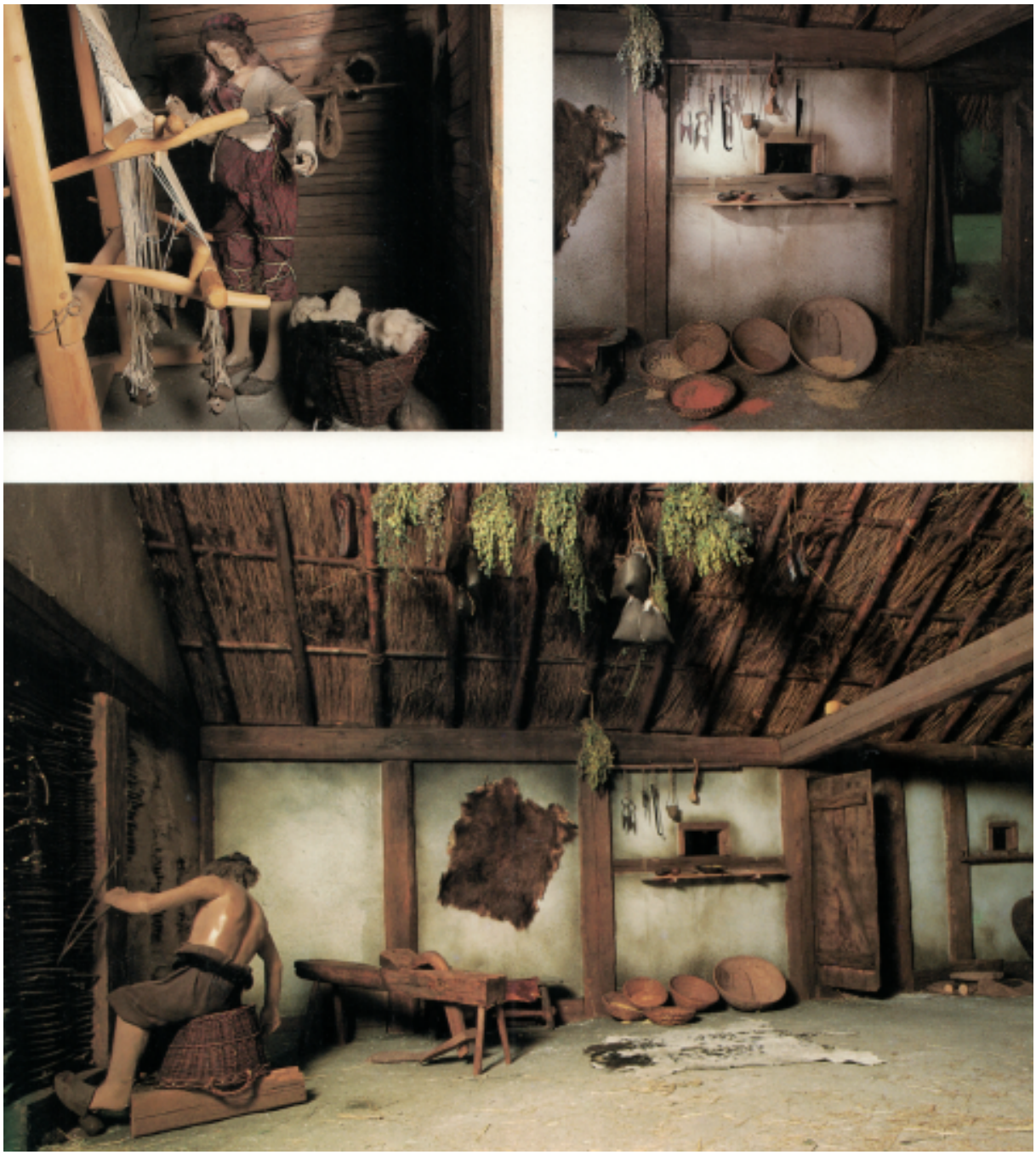
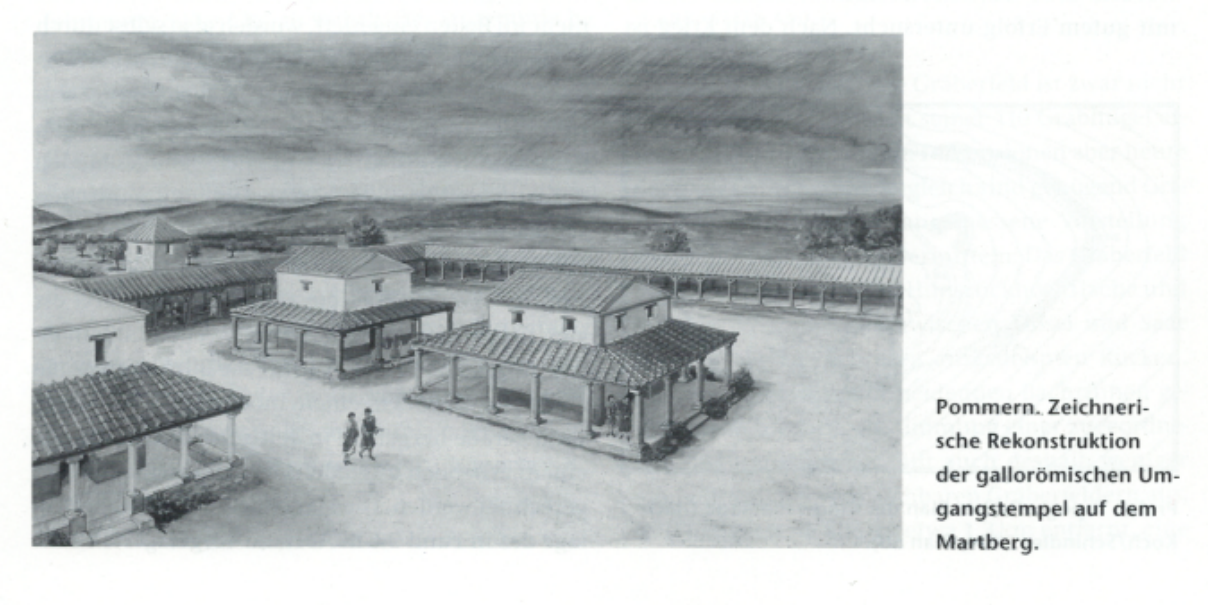


Figure B.4: Inner view of a living house.

Illustrations from Rieckhoff and Biel's *'Die Kelten in Deutschland'* [9][10][11].

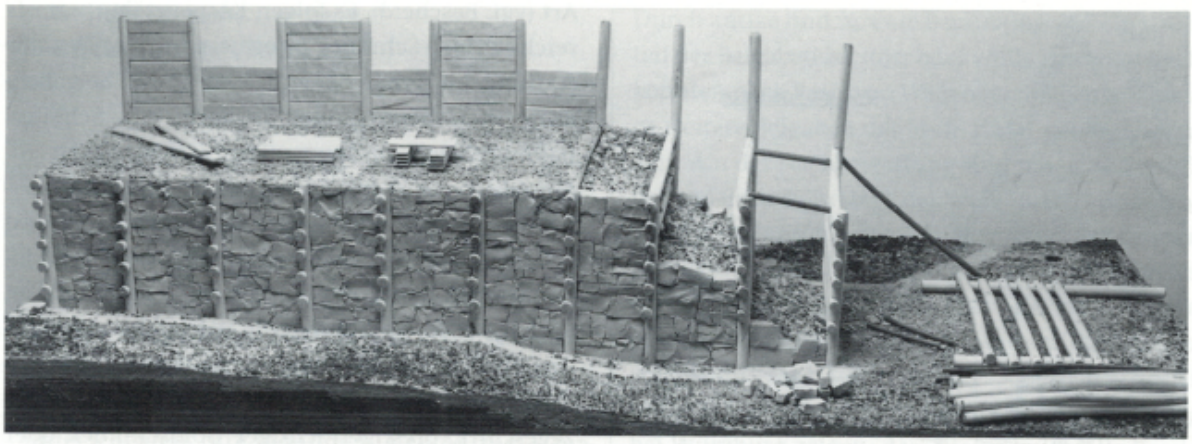


Pommern. Zeichnerische Rekonstruktion der gallorömischen Umgangstempel auf dem Martberg.

Figure B.5: We modeled our tower after the tower on the left of the background.

Dannenfels. Oppidum auf dem Donnersberg. Rekonstruktion des Südwalles am Ludwigsturm.

Figure B.6: Reconstruction of a Oppidum wall.



Preist. Modell der Pfostenschlitzmauer.

Figure B.7: Another reconstruction of an Oppidum wall.

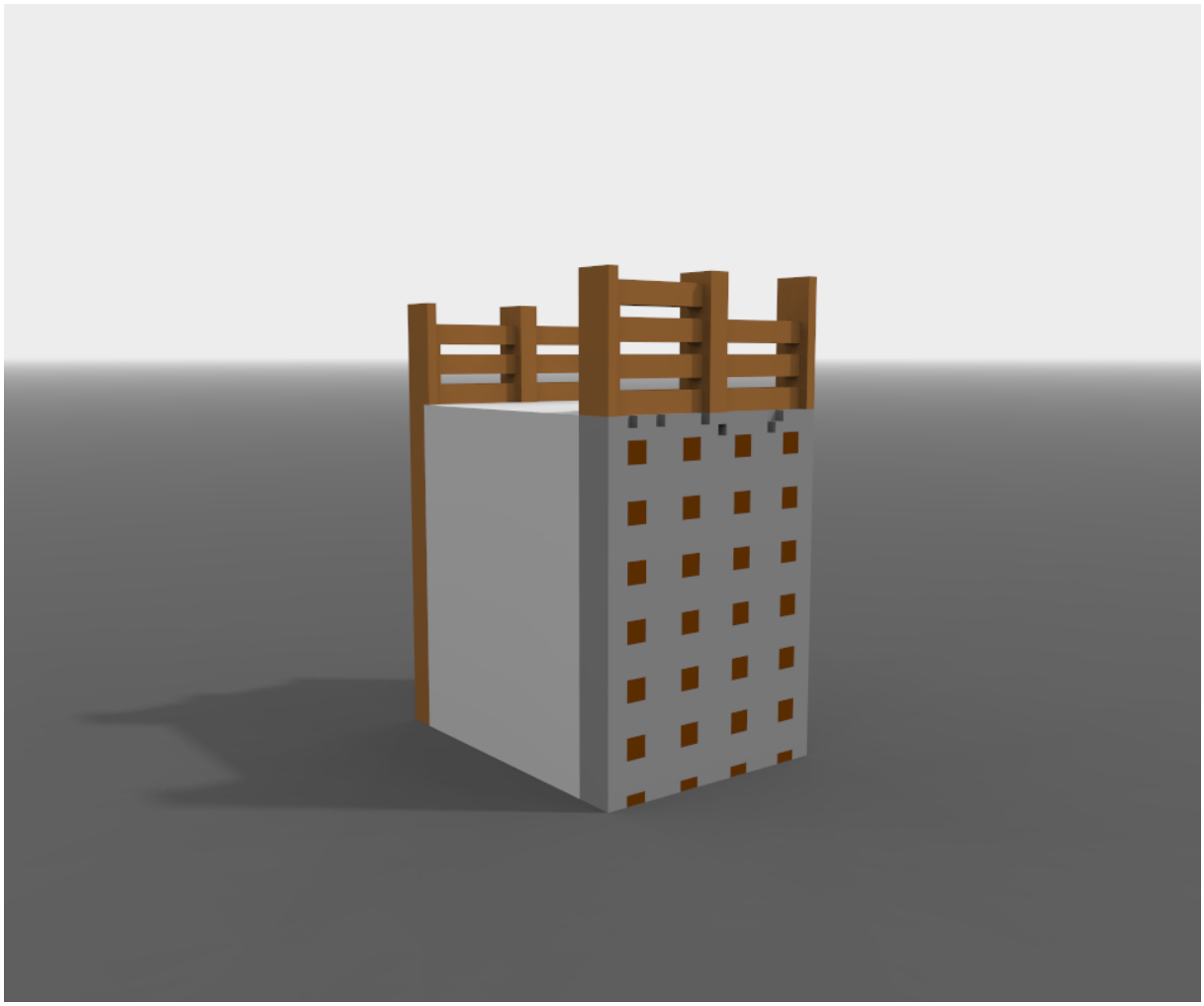


Figure B.8: Our stone wall model.

C Appendix C

Raw test results.

	Average						
Age		21-30	21-30	51-60	17-20	21-30	21-30
Gender		male	male	female	female	male	male
3dMovies	3,42	4	3	2	3	5	3
games pc	4,50	5	4	3	3	5	5
games mobile	3,83	5	4	3	5	3	2
ar	3,67	5	3	5	2	2	3
vr	3,67	5	2	5	1	5	2
xr	2,83	5	1	5	1	1	1
celts	1,67	4	1	2	1	2	1
celtic runes	1,92	5	1	3	2	1	1
td games	3,92	5	5	3	3	4	4
like game	4,17	5	5	5	5	3	3
like combination	4,58	5	5	5	5	4	5
learned	2,50	5	5	3	2	2	1
points	3,33	6	6	1	5	0	1

Figure C.1: Test results 1-6.

	Average						
Age		21-30	17-20	21-30	21-30	21-30	21-30
Gender		male	male	male	male	male	female
3dMovies	3,42	4	3	3	5	2	4
games pc	4,50	5	5	4	5	5	5
games mobile	3,83	4	3	2	5	5	5
ar	3,67	5	2	3	5	5	4
vr	3,67	5	3	3	5	4	4
xr	2,83	5	1	2	5	3	4
celts	1,67	1	1	3	1	1	2
celtic runes	1,92	1	1	3	2	1	2
td games	3,92	5	3	4	2	5	4
like game	4,17	4	3	4	4	5	4
like combination	4,58	5	4	5	2	5	5
learned	2,50	3	1	3	2	1	2
points	3,33	6	2	6	3	1	3

Figure C.2: Test results 7-12