

# Bachelorarbeit

Lukas Semmler



# Bachelorarbeit

im Studiengang User Experience Design

Fakultät Informatik

## **Anwenden von UX-Design-Methoden zur Konzeption eines Videospiele anhand des Beispiels einer Stromnetz-Wirtschafts-Simulation**

**Lukas Semmler**

ausgegeben am: 08.10.2019

abgegeben am: 08.03.2020

Erstprüfer: Prof. Ingrid Stahl

Zweitprüfer: Mathias Haimerl



---

# Abstract

Diese Arbeit befasst sich mit dem Konzept, Design und der prototypischen Implementierung eines Videospiele. Dabei liegt der Schwerpunkt auf der Interface-Gestaltung und Usability des Spiel-UIs. Als Orientierung für den Prototypen dient dabei das Videospiele-Genre der Stadt-Aufbau-Simulations-Spiele. Das Setting für das Spiel ist Stromversorgung und Kraftwerke. Ziel der Arbeit war es herauszufinden, wie sich das Interface eines Videospiele anpassen muss, damit das gleiche Spielerlebnis auf verschiedenen Endgeräten möglich ist.

Als Gestaltungsprozess diente der *User-Centered-Design-Ansatz* (UCD). Zunächst wurde über die Themen Stromversorgung und die Videospiele-Industrie in Deutschland recherchiert. In der Analyse-Phase wurden dann die Anforderungen für den Spiele-Prototypen definiert. Dabei wurde das *Gamer Motivation Model* verwendet, um Videospiele anhand ihrer Motivation zu unterscheiden. Zum besseren Verständnis wurde auch eine Konkurrenzanalyse zur Städte-Aufbau-Simulation *SimCity* durchgeführt. Außerdem wurden die technischen Eigenschaften der angepeilten Endgeräte – Mobile, Smart TV und Desktop – untersucht. Im nächsten Schritt wurde das Konzept des Spiele-Prototypen angefertigt. Mithilfe eines *Game Design Documents* wurden die nötigen Features des Prototypen definiert. Es wurden Skizzen und Wireframes angefertigt, darauf aufbauend ist ein Klickprototyp entstanden. Der Klickprototyp wurde mit einem Nutzertest untersucht. Darauffolgend wurde die Gestaltung des Prototypen festgelegt. Es wurde damit begonnen, den gestalteten Spiel-Prototypen in Unity zu programmieren. Im Verlauf hat sich jedoch Unity als ungeeignet erwiesen, da das UI-System zu wenige Gestaltungsmöglichkeiten bietet. Daraufhin folgte der Wechsel zu einem HTML-basierten Prototypen. Dieser ließ sich in HTML und CSS wesentlich besser gestalten. Gemäß „Mobile First“ wurde sich in dieser Arbeit auf die mobile Gestaltung konzentriert. Der zweite Prototyp wurde mit Probanden getestet, die eine hohe Erfahrung mit Videospiele hatten. Beim Vergleich der beiden Nutzertests hat sich gezeigt, dass ein voll-funktionsfähiger Spiele-Prototyp bessere Bewertungen bringt als ein Klick-Prototyp.

# Inhaltsverzeichnis

<b>Abstract</b>	<b>iii</b>
<b>1 Einleitung</b>	<b>1</b>
<b>2 Grundlagen</b>	<b>5</b>
2.1 Stromversorgung . . . . .	5
2.1.1 Stromnetz . . . . .	6
2.1.2 Netzausbau . . . . .	7
2.2 Videospiele . . . . .	8
2.2.1 Definition . . . . .	8
2.2.2 Entwicklungsprozess . . . . .	8
2.2.3 Videospiele-Genres . . . . .	9
<b>3 Analyse</b>	<b>11</b>
3.1 Zielgruppe . . . . .	11
3.2 Persona . . . . .	13
3.3 Konkurrenzanalyse: SimCity . . . . .	15
3.4 Anforderungen Spieleprototyp . . . . .	16
3.5 Anforderungen Endgeräte . . . . .	18
3.5.1 Mobile . . . . .	19
3.5.2 Smart TV . . . . .	20
3.5.3 PC / Laptop . . . . .	21
<b>4 Konzept</b>	<b>23</b>
4.1 Game Design Document . . . . .	24
4.2 Use Cases . . . . .	26
4.3 Skizzen und Wireframes . . . . .	28
4.3.1 Erstes Konzept . . . . .	28
4.3.2 Wireframes Mobile . . . . .	30
4.3.3 Wireframes Smart TV . . . . .	34
4.3.4 Wireframes PC . . . . .	40
4.4 Ingame-Assets . . . . .	42
4.5 Klick-Prototyp . . . . .	44

<b>5 Design</b>	<b>49</b>
5.1 Moodboard . . . . .	49
5.2 Farben . . . . .	50
5.3 Typografie . . . . .	52
5.4 Screendesign . . . . .	54
5.5 Icondesign . . . . .	56
5.6 Ingame-Design . . . . .	58
<b>6 Implementierung</b>	<b>61</b>
6.1 Test mit Unity-Prototyp . . . . .	62
6.1.1 Unity . . . . .	62
6.1.2 UI-Test mit XmlLayout . . . . .	63
6.1.3 Spiele-Prototyp in Unity . . . . .	65
6.2 Phaser-Prototyp . . . . .	65
6.2.1 Vue.js . . . . .	66
6.2.2 Phaser . . . . .	66
6.2.3 Kommunikation zwischen Vue.js und Phaser . . . . .	66
6.2.4 Screenshots . . . . .	67
<b>7 Evaluation</b>	<b>73</b>
7.1 Nutzertest 1 . . . . .	73
7.1.1 Aufbau . . . . .	73
7.1.2 Durchführung . . . . .	74
7.1.3 Auswertung . . . . .	76
7.2 Nutzertest 2 . . . . .	79
7.2.1 Aufbau . . . . .	79
7.2.2 Durchführung . . . . .	80
7.2.3 Auswertung . . . . .	80
7.3 Vergleich . . . . .	82
7.3.1 Think-Aloud . . . . .	82
7.3.2 UEQ und SUS . . . . .	82
<b>8 Zusammenfassung und Ausblick</b>	<b>85</b>
<b>Erklärung</b>	<b>89</b>
<b>Abbildungsverzeichnis</b>	<b>91</b>
<b>Code-Listings</b>	<b>94</b>
<b>Literatur</b>	<b>95</b>
<b>A Datenträger</b>	<b>102</b>
<b>B Nutzertest 2</b>	<b>103</b>







---

# Kapitel 1

## Einleitung

Games have an obligation to their players' sense of active enjoyment that goes beyond what UX designers normally need to deliver. That's also precisely the thing that makes game design so interesting.

---

*John Ferrara*

In seinem Buch erklärt Ferrara, dass UX- und Gamedesign die gleichen Wurzeln haben [1, S. 4f]. Beide beschäftigen sich mit Mensch-Maschine-Interaktion, beide sind auf die Gestaltung (Design) der Erfahrung fokussiert. Der Unterschied ist, dass UX-Design sich auf das Lösen von realen Problemen stützt und versucht, die Experience bei der Tätigkeit zu gestalten. *Game UX* konzentriert sich ausschließlich auf die Erfahrung. So schreibt Ferrara: „If our central concern is the quality of an interactive experience, then there's no reason why video games, which can satisfy people so richly, should be excluded.“ [1, S.12] Also warum sollte man sich als UX-Designer für Videospiele interessieren? Weil sie wie kein anderes Medium auf eine gute Experience ausgelegt sind.

Die Videospiele-Industrie ist in den letzten Jahrzehnten zu einer riesigen Branche gewachsen. In Deutschland allein hat die Industrie 2018 einen Umsatz von 4,3 Milliarden € gemacht [2]. Zum Vergleich, die deutsche Musikwirtschaft hat 2018 einen Umsatz von 1,6 Mrd. € und die Filmwirtschaft (ohne TV) 2014 einen Umsatz von 4,2 Mrd. € erwirtschaftet [3, S. 5] [4, S. 42]. Statistisch gesehen spielen 34 Millionen Menschen in Deutschland Videospiele – und das über alle Altersgruppen verteilt [2].

Das beliebteste Videospiele-Genre in Deutschland ist „Strategie“. Auch Simulations-Spiele erfreuen sich großer Beliebtheit, sie sind auf Platz 5 [5]. Eine Kombination dieser beiden Genres ist die „Aufbau-Simulation“. In dieser Arbeit soll ein Spiele-Prototyp erstellt werden. Da Aufbau-Simulations-Spiele typischerweise ein komplexes UI besitzen, wird dieses Genre für den Prototypen verwendet.

Doch welches Setting soll den Spiele-Prototyp begleiten? Untersucht man die gesellschaftlich relevanten Themen der letzten Jahre, fällt auf, dass öfter der Begriff „Energiewende“ genannt wird. Zu diesem Thema schreiben Radtke und Kersting [6, S. VII]: „In Deutschland wurde mit dem Beschluss der Bundesregierung 2011 zur Energiewende [...] eine Jahrhundertaufgabe angegangen, die zukünftige Generationen, aber auch unsere Lebenswirklichkeit

betrifft. In den kommenden Jahren wird die Energiewende nicht nur als technische, sondern gesellschaftliche Transformation unser Zusammenleben und unsere Arbeitswelt beeinflussen.“ Daher ist es das Ziel dieser Arbeit, einen Videospiele-Prototypen anhand des Beispiels einer Stromnetz-Wirtschafts-Simulation zu erstellen.

Der Spieler soll bei der Entwicklung ins Zentrum gestellt werden, also wird der *User-Centered-Design*-Prozess angewandt. Dazu schreibt Jacobsen [7, S. 83]: „Die nutzerzentrierte Entwicklung – häufig wird auch der englische Begriff User-Centered Design oder kurz UCD genutzt – sieht vor, dass der zukünftige Nutzer von Anfang an in die Konzeption und die Umsetzung einer Anwendung einbezogen wird.“

Im Verlauf dieser Arbeit werden die vier Phasen des UCD durchlaufen. Dabei wurden sie dem Thema der Videospiele-Prototypen-Entwicklung entsprechend angepasst.

Phase	Beschreibung	Methoden	Kapitel
Analyse	Definition der Zielgruppe. Welche Ziele, Erwartungen und welches Umfeld bei der Benutzung haben die Spieler?	Persona, Konkurrenzanalyse	siehe Kapitel 3
Konzeption	Beschreiben des Spiels (User Interface, Spielwelt und alle Aktionen)	Game Design Document, Wireframes	siehe Kapitel 4
Design und Umsetzung	Interaktionsspezifikationen, Gestaltung und Programmierung des Prototypen	Klick-Prototyp und Spiel-Prototyp	siehe Kapitel 5 & Kapitel 6
Evaluierung	Testen des Spiels und dessen Gestaltung anhand der Anforderungen	Thinking Aloud, Semi-Strukturiertes Interview, UEQ und SUS	siehe Kapitel 7

Tabelle 1.1: UCD-Prozess: Die vier Phasen und die dafür angewandten Methoden [7, S. 85].





---

# Kapitel 2

## Grundlagen

### 2.1 Stromversorgung

Die Stromversorgung zählt zur Grundversorgung. Es gibt in einem modernen Haushalt kaum ein System, das nicht abhängig ist vom Stromnetz: Wasserversorgung durch Pumpen, Licht, Elektroherde, Waschmaschinen, auch Fernseher und die Internetverbindung. Allein der steigende Internetverkehr verursacht mit seiner Infrastruktur einen immensen Stromverbrauch. So sagte Margrethe Vestager, Vizepräsidentin der EU-Kommission, im Dezember 2019, dass Internet-Streaming sehr energieintensiv sei. „Allein im vergangenen Jahr haben Streaming-Plattformen ungefähr so viel Strom verbraucht, wie alle Privathaushalte in Deutschland, Italien und Polen zusammen.“ [8]

Mit der in Deutschland 2011 beschlossenen Energiewende wird einerseits die Stromversorgung umgebaut: Statt fossiler Energieträger sollen mehr erneuerbare Energien genutzt werden. Andererseits muss durch die damit einhergehende Dezentralisierung das Stromnetz ausgebaut werden [6, S. VIII]. Daher sollen im Folgenden als Grundlage das Stromnetz und der Netzausbau in Deutschland untersucht werden.

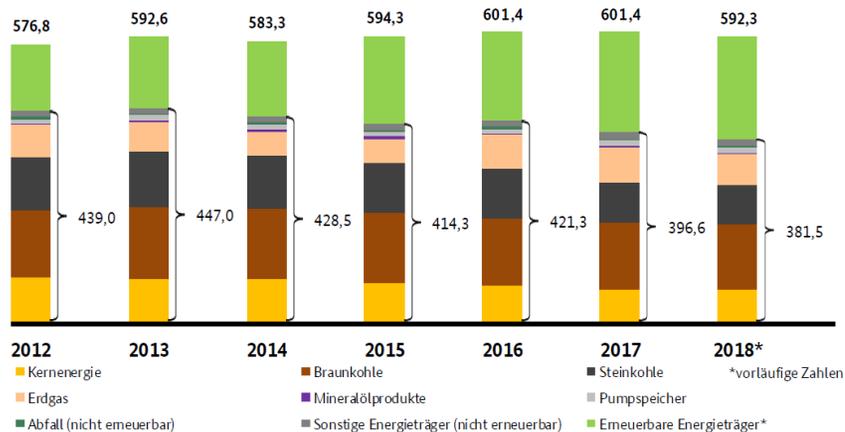


Abbildung 2.1: Entwicklung der Netto-Stromerzeugung in TWh (Stand: Okt. 2019) [9, S. 59]

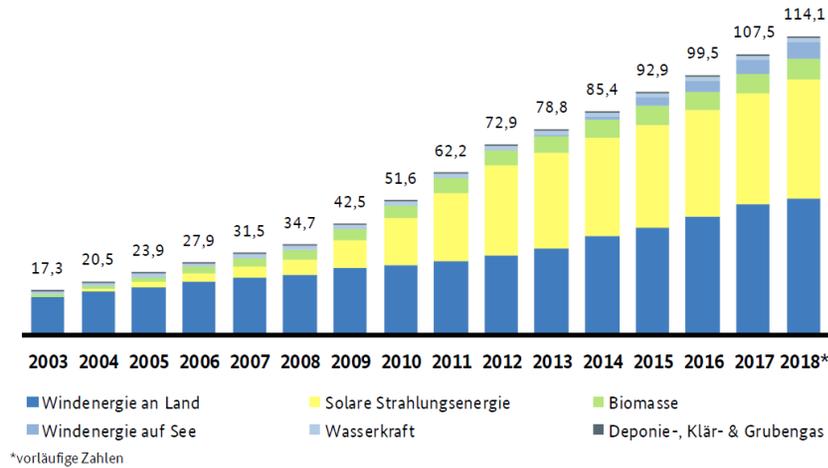


Abbildung 2.2: Entwicklung der installierten Leistung von Anlagen mit Zahlungsanspruch nach dem EEG bis 2018 [9, S. 84]

### 2.1.1 Stromnetz

Das Stromnetz in Deutschland und Österreich besteht insgesamt aus über 1.300 Kraftwerken [10]. Deutschlandweit gibt es 300 Kraftwerke mit einer Leistung von mehr als 100 MW [11]. Die fünf größten Stromnetzbetreiber (RWE, LEAG, EnBW, E.ON/Uniper und Vattenfall) haben 2018 insgesamt 259,3 TWh Strom erzeugt – das sind 73,9 % der gesamten Stromerzeugung. 1,8 Mio. Kilometer Länge hat das deutsche Stromnetz (bestehend aus Höchst-, Hoch-, Mittel- und Niederspannungsleitungen) [9, S.47, S. 39].

Kraftwerke werden von den Betreibern möglichst kosteneffizient eingesetzt. Dazu werden „neben den Brennstoffkosten auch Kriterien wie Anfahrzeiten oder ein veränderter Wirkungsgrad bei Teillast berücksichtigt.“ Der Verband der Bayerischen Energie- und Wasserwirtschaft schreibt hierzu, dass „die Kraftwerkseinsatzpläne [...] durch die Kraftwerksbetreiber am Vortag an den [N]etzbetreiber übergeben [werden].“ [12, S. 17]

Die Energiewende ist das zentrale Thema der Energiepolitik der nächsten Jahrzehnte. Neben dem Atom- und Kohleausstieg gibt es ein ambitioniertes Klimaschutzziel zur Verringerung der  $CO_2$ -Emissionen. Diese Wende soll mithilfe von erneuerbaren Energien geschehen [6, S. VII]. So wurden im Jahr 2018 bereits 37 % des Brutto-Stromverbrauches mit *Erneuerbaren Energien* (EE) gedeckt [9, S. 58].<sup>1</sup> Auch ist der Anteil der installierten Leistung von Erneuerbaren Energien die letzten Jahre stetig gewachsen (siehe Abbildung 2.2). Durch den vermehrten Einsatz dieser fluktuierenden Energieträgern ist auch der Aufwand gestiegen, mit denen das Stromnetz ausgeglichen werden muss. Wenn zu einer Uhrzeit mehr Wind weht, generieren Windkraftwerke mehr Strom. Dagegen müssen bei schwächerem Wind Reservekraftwerke angeschaltet werden, um das Stromdefizit auszugleichen. 2018 wurden insgesamt 330 Mio. € ausgegeben, um Netzreserve-Kraftwerke in Bereitschaft zu halten [9, S. 140].

<sup>1</sup>Unter [SMARD.de/ueberblick](http://SMARD.de/ueberblick) kann man Daten zu den Kraftwerken in ganz Deutschland auf einer Karte abrufen. Außerdem zeigt [SMARD.de/marktdaten](http://SMARD.de/marktdaten) in Echtzeit, wie viel Strom mit welchen Energieträgern gerade generiert wird. (Filter: „Stromerzeugung“ → „Realisierte Erzeugung“) Stand: 6. Februar 2020

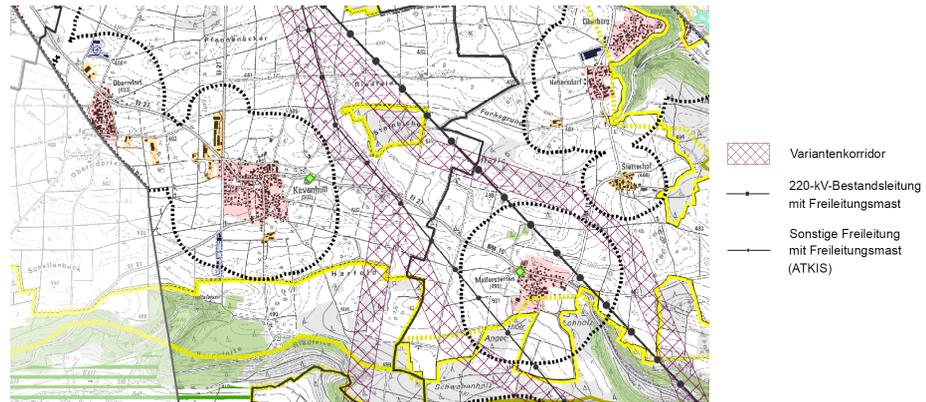


Abbildung 2.3: Ausschnitt aus dem Verlauf der geplanten *Juraleitung* [13].

### 2.1.2 Netzausbau

Durch die Änderung hin zu ökologisch neutralen Energieträgern ist es auch wichtiger geworden, dass der Netzausbau mit dem technischen Fortschritt Schritt hält. Aus diesem Grund wurde 2013 das *Bundesbedarfsplangesetz* (BBPIG) beschlossen [14]. Damit soll auch in Zukunft ein stabiles Stromnetz garantiert werden.<sup>2</sup> Es sieht vor, dass neue Leitungen gebaut werden. Da Kraftwerke abgeschaltet werden, soll dadurch vor allem Energie vom Norden, mit starken Windanlagen, in den Süden, mit hohem Energieverbrauch, geführt werden.

Gegen den Ausbau des Stromnetzes regt sich jedoch auch Widerstand in Form von Bürgerinitiativen und Vereinen (z.B. [15] und [16]). Im Zuge des BBPIG soll beispielsweise die sogenannte *Juraleitung* gebaut werden (siehe Abbildung 2.3). Die Leitung soll von Raitersaich nach Altheim führen und ist knapp 160 km lang. Unter anderem soll sie durch das Altmühltal führen – was bei den Bewohnern der umliegenden Gemeinden auf Widerstand stößt [17]. Es gibt bereits eine Freiluftleitung in dieser Region, sie soll aber rückgebaut werden. Die vorhandene 35 Meter hohe Leitung wird durch eine neue Leitung mit 75 Meter Höhe ersetzt. Nach dem letzten Stand vom 30. September 2019 wurde noch kein Antrag auf ein Raumordnungsverfahren gestellt. Bis 2028 soll die Leitung gebaut werden [18].

Da diese beiden Themen (Energiewende und einhergehender Ausbau des Stromnetzes) zu starken Interessenskonflikten führen, wurde dies als Thema für den Prototypen ausgewählt. Ziel ist es, dem Spieler eine Einführung in dieses komplexe Thema der Stromversorgung zu geben. Bei der Stromversorgung müssen Kompromisse zwischen vielen Akteuren gefunden werden. Dabei gilt es immer, ein Gleichgewicht zwischen den drei Zielen *Umweltschutz*, *Wirtschaftlichkeit* und *Versorgungssicherheit* zu finden. Entscheidungen in einer dieser energiepolitischen Dimensionen erzeugt immer Wechselwirkungen mit den anderen [19, S. 4].

<sup>2</sup>Unter [netzausbau.de/leitungsvorhaben](http://netzausbau.de/leitungsvorhaben) kann in Echtzeit der aktuelle Stand zu den Leitungsvorhaben im Stromnetz eingesehen werden. *Stand: 6. Februar 2020*

## 2.2 Videospiele

### 2.2.1 Definition

Ein Videospiele ist (allgemein gesagt) ein interaktives Medium, das zur Unterhaltung dient. John Ferrara schreibt, dass es keine einstimmige Definition für Videospiele gibt. Aber für UX-Praktiker kann angenommen werden, dass sich ein Spiel durch folgende Eigenschaften auszeichnet [1, S. 16ff]:

1. **Ziel.** Alle Spiele haben eine Art von Ziel, das erreicht werden muss. Das Ziel wird dabei vom Spiel-Schöpfer festgelegt. Es muss explizit, messbar und verlässlich sein.
2. **Umwelt-Constraints.** In jedem Spiel muss es Elemente geben, die abgrenzen, was der Spieler tun kann und was nicht. Dabei gibt es zum einen *Constraints der Umwelt*. Auf einem Schachbrett müssen sich z.B. die Figuren alle auf einem  $8 \times 8$ -Feld bewegen.
3. **Formale Constraints.** Zum anderen gibt es *formale Constraints*. Der Spieler ist in seinen Aktionen durch die Spielregeln eingeschränkt. So kann sich der Läufer in Schach nur diagonal zwischen den Zellen bewegen.

Neben diesen drei Kennzeichen haben speziell Videospiele noch ein weiteres [1, S. 21ff]:

4. **Maschinen-basierte Entscheidungen.** In Videospiele muss sich der Spieler nicht um das Einhalten der Regeln kümmern. Auch wer gewinnt, wird von der Maschine bestimmt. Der Vorteil dadurch ist, dass Videospiele viel komplexer werden können. Spiele müssen auch nicht – wie bei vielen Brettspielen üblich – rundenbasiert sein, sondern können in Echtzeit ablaufen.

### 2.2.2 Entwicklungsprozess

Videospiele folgen meist einem bestimmten Entwicklungsprozess. Er ist iterativ und wird je nach Anforderungen mehrmals durchlaufen. Dieser Prozess (nach Chandler) teilt sich auf in vier Phasen [20, S. 97], [21, S. 4ff]:

1. **Preproduction.** Während der *Preproduction*-Phase wird definiert, welches Spiel entwickelt wird. Sie macht ca. 10–25 % der gesamten Entwicklungszeit aus. Innerhalb dieser Phase soll ein *Game Plan* erstellt werden. Dieser Plan ist ein ausgeschriebener Fahrplan der Entwicklung. Er enthält alle Features, die für das fertige Spiel umgesetzt werden müssen. In der *Preproduction* wird das Konzept festgelegt. Auch sollte ein Prototyp erstellt werden, der das finale Gameplay zeigt (siehe Kapitel 6).
2. **Production.** In der *Production*-Phase werden die Assets und Code für das Spiel erstellt. Dabei folgt die Implementierung dem zuvor erstellten *Game Plan*.
3. **Testing.** Beim *Testing* wird überprüft, ob alle Features funktionieren. Es sollen z.B. keine Spielabstürze auftreten. Getestet wird dabei parallel zur *Production*.

4. **Post-Production.** In der *Post-Production* gibt es einen Archiv-Plan. Je nach Bedarf wird auch ein *Post Mortem* geschrieben. Das Post Mortem ist eine nachträgliche Analyse, was gut und schlecht bei der Entwicklung gelaufen ist.

### 2.2.3 Videospiele-Genres

Videospiele können in Genres eingeteilt werden. Die Unterteilung kann dabei nach verschiedenen Aspekten, wie Setting (*Fantasy, Sci-Fi*), Kamera (*First-Person, Isometric*) oder anderen Kriterien (*Adult, Party Game*) stattfinden [22, S. 152]. Die teils ungenaue Verwendung des Begriffes „Genre“ hilft zuallererst den Spielern, die riesige Masse an Videospiele auf ein verarbeitbares Maß herunterbrechen. In dieser Arbeit wird aber angenommen, dass das Genre eines Videospiele dessen Gameplay beschreibt. „In the game industry a genre is a common set of game mechanics and interface standards that a group of titles share.“ [22, S. 155] Laut Chris Zukowski ist das UI eines Spiele sehr wichtig, da es Hinweise gibt, um welches Genre es sich bei einem Spiel handelt [23]. So zeichnen sich z.B. Aufbau-Simulations-Spiele dadurch aus, dass es ein Baumenü gibt und eine Spielwelt, in der man die Kamera frei bewegen kann (vgl. Abbildung 3.2).

Für einen Spieler ist es sehr wichtig, dass er ein Spiel zu einem bereits bekanntem Genre zuordnen kann. Diese Zuordnung hat maßgeblichen Einfluss auf die Kaufentscheidung. So schreibt Chris Zukowski: „Your Steam<sup>3</sup>page will not convert someone who doesn't like your genre. [...] You want the message for them [the players] to be ‚My game plays just like that other game that you spent 100 hours playing and loved but this is MORE OF IT and it is PRETTIER!‘“ [23]

Das heißt, bei der Entwicklung eines Videospiele muss stark auf die Erwartungen und Anforderungen der Zielgruppe eingegangen werden (siehe Kapitel 3.1). Schätzungsweise gibt es weltweit ca. 700–800 Millionen PC-Spieler. Dazu sagt Sergey Galyonkin: „Because if you're developing a downloadable game for Steam you're not even fighting for 135M [million] of its active users, you're fighting for the attention of 1.3 million gamers that are actually buying lots of games. The 1% group.“ [25] Das liegt daran, dass der Großteil der Spieler nur wenige Spiele besitzt. Hingegen besitzen 1 % der Spieler auf Steam 100 oder mehr Spiele. Diese sind eher bereit, neue Titel auszuprobieren – solange das Spiel in ein Genre fällt, das sie mögen [25].

Somit ist es wichtig für den Spielprototypen, der zum Test in dieser Arbeit erstellt werden soll, dass er den Erwartungen eines typischen Städte-Aufbau-Simulations-Spielers gerecht wird.

---

<sup>3</sup> Steam ist eine Verkaufsplattform für den Videospiele-PC-Markt. Neben dem Steam Client, der eine Bibliothek mit allen gekauften Spiellizenzen enthält, bietet Steam auch viele Community-Features. Zum Beispiel den Workshop, mit dem Spieler für bestimmte Spiele selbst Inhalte wie Items oder Maps erstellen können [24].



---

# Kapitel 3

## Analyse

### 3.1 Zielgruppe

Videospiele haben sehr unterschiedliche Zielgruppen. Allein in Deutschland gibt es 34 Millionen Videospiele [2]. Daher ist es umso wichtiger, auch für die richtige Zielgruppe ein Spiel zu entwickeln. Ferrara schreibt hierzu: „Getting a handle on players’ motivations is extremely important because it tells you what your game needs to accomplish to fulfill their expectations, and that in turn helps to inform design.“ [1, S. 38]

Dabei ist die Zielgruppe abgeleitet worden anhand der Motivationen. Hierfür wurde von Nick Yee (von *Quantic Foundry*) das *Gamer Motivation Model* entwickelt. In diesem Modell werden Gamer anhand von sechs Motivationspaaren eingeteilt (siehe Abbildung 3.1).

Diese Motivations-Faktoren geben eine Bandbreite wieder. Spieler können in den Kategorien einen hohen oder niedrigen Score haben. Je nachdem ändern sich auch die Präferenzen. Ein niedriger Score heißt aber nicht, dass die Vorlieben weniger stark ausgeprägt sind. Zum Beispiel steht ein niedriger *Story*-Score für *Open Ended*- und ein hoher für *Scripted Drama*-Spiele [26], [27].



Action "Boom!"	Social "Let's Play Together"	Mastery "Let Me Think"	Achievement "I Want More"	Immersion "Once Upon a Time"	Creativity "What If?"
<b>Destruction</b> Guns. Explosives. Chaos. Mayhem.	<b>Competition</b> Duels. Matches. High on Ranking.	<b>Challenge</b> Practice. High Difficulty. Challenges.	<b>Completion</b> Get All Collectibles. Complete All Missions.	<b>Fantasy</b> Being someone else, somewhere else.	<b>Design</b> Expression. Customization.
<b>Excitement</b> Fast-Paced. Action. Surprises. Thrills.	<b>Community</b> Being on Team. Chatting. Interacting.	<b>Strategy</b> Thinking Ahead. Making Decisions.	<b>Power</b> Powerful Character. Powerful Equipment.	<b>Story</b> Elaborate plots. Interesting characters.	<b>Discovery</b> Explore. Tinker. Experiment.

Abbildung 3.1: Motivation von Gamern: Sechs Motivations-Paare nach dem *Gamer Motivation Model* von Quantic Foundry [26].

Mithilfe des Gamer Motivation Model lässt sich bestimmen, welche Anforderungen der Spiele-Prototyp erfüllen soll. Das heißt, Spieler mit folgenden Merkmalen zählen zur Zielgruppe:

- **Independence** (Low Community) Singleplayer. Volle Kontrolle.
- **Calm** (Low Excitement) Entspanntes Gameplay. Vorhersehbar.
- **Self-Driven** (Low Completion) Sandbox-Gameplay. Eigene Vorgehensweise.
- **Open-Ended** (Low Story) Freie Spielwelt zum Austoben.
- **Customizable** (High Design) Ausdruck von Individualität. Viele Gebäude.

Außerdem kann man anhand der Zielgruppe ableiten, welche Videospiele ähnlich sind. Im Rahmen dieser Arbeit wurden mithilfe des *Gamer Motivation Models* zehn Videospiele bestimmt (siehe Tabelle 3.1). Diese Empfehlung wurde aus folgenden drei Spielen als Input generiert: 1) *Anno 1404* (Städte-Bau- und Wirtschaftssimulation), 2) *Cities: Skyline* (Städte-Bau-Simulation) und 3) *RollerCoaster Tycoon 2* (Vergnügungspark-Bau-Simulation).

Die Wahrscheinlichkeit, dass Spieler mit gleichen Interessen diese Spiele besitzen, wird mit dem *QF Score* (QF) beschrieben.<sup>1</sup> Auffällig an den empfohlenen Spielen ist, dass alle aus dem Genre Aufbau-Simulation sind. Unterschiedlich ist jedoch das Setting, ob Städte-Aufbau, Bau eines Vergnügungsparks oder Gefängnisses [28].

	<b>Spiel</b>	<b>Spiel-Beschreibung</b>	<b>QF</b>
1	SimCity 3000	Städte-Bau-Simulationsspiel	22.8
2	SimCity 4	Städte-Bau-Simulationsspiel	20.8
3	Planet Coaster	Vergnügungspark-Bau-Simulationsspiel	20.5
4	Anno 2070	Städte-Bau- und Wirtschaftssimulations-Spiel	17.7
5	Surviving Mars	Sci-Fi-Städte-Bau-Simulationsspiel	17.5
6	SimCity (1989)	Städte-Bau-Simulationsspiel	17.3
7	RollerCoaster Tycoon 3	Vergnügungspark-Bau-Simulationsspiel	17.2
8	Banished	Städte-Bau-Strategiespiel im Mittelalter	16.2
9	Democracy 3	Regierungs-Management-Simulation	15.9
10	Prison Architect	Gefängnis-Bau-Simulationsspiel	15.9

Tabelle 3.1: Videospiele mit ähnlichen Qualitäten, die der Spiele-Prototyp aufweisen soll.

---

<sup>1</sup> Für den *QF Score* wird zuerst eine Stichprobe an Spielern genommen, die die Input-Spiele als Lieblingsspiel angegeben haben. Dann wird ermittelt, welche Spiele diese Spieler selbst als Lieblingsspiel angegeben haben. Der Wert ist relativ zur grundlegenden Beliebtheit eines Spiels – das heißt, ein *QF Score* von 2 bedeutet, dass die Spieler in der Stichprobe zweimal so wahrscheinlich das Spiel als Lieblingsspiel angeben, im Vergleich zu alle Spieler [28].

## 3.2 Persona

Anhand der Anforderungen an die Zielgruppe wurden Personas erstellt. Aber was ist ein Persona? Christian Moser schreibt hierzu, dass „Personas [...] fiktive Personen [sind], die stellvertretend für eine Gruppe von Nutzern mit ähnlichen Bedürfnissen stehen.“ [29, S. 78] Sie werden auch in der Videospiele-Industrie eingesetzt: „In some studios, they use a persona method to identify a fictional character personifying the goals, preferences, expectations, and behaviors of the target audience. This user-centered method allows the game development team as well as the marketing and publishing teams to be aligned regarding who the core player hypothetically will be.“ [30, S. 25] Ziel ist es dabei, herauszufinden, wer der typische Spieler („Core Gamer“) des Spiels sein soll. Die Personas können sogar um ein „Anti-Persona“ erweitert werden. „You might want to have several personas – for example, a primary persona for the core audience, a secondary persona for a broader reach, and an anti-persona who would be the person you are not targeting.“ [30, S. 197]

Aus diesem Grund wurden in dieser vorliegenden Arbeit drei Personas erstellt. Sie haben jeweils andere Motivationen, warum sie spielen. Jeder der drei Personas hat außerdem eine andere Lieblingskonsole.

- **Carola Casual** mag Mobile – ihr Persona repräsentiert die Mobile-Sparte. Sie legt viel Wert auf soziale Eingebundenheit und spielt am liebsten Multiplayer-Spiele. Dennoch, sie lebt sich auch gern kreativ aus und spielt Simulationen wie *Die Sims*. Als Mobile-Expertin legt sie aber auch viel Wert auf ein gutes UI – es soll effizient und leicht bedienbar sein.
- **Henry Hardcore** legt großen Wert auf Kompetenz – er will sich mit anderen messen können. Am liebsten spielt er Action-Spiele, aber er hat Erfahrung mit einer großen Bandbreite an Videospielegenres. Henry ist gewissermaßen die „Anti-Persona“, da er mit Aufbau-Simulationen nicht so viel anfangen kann.
- **Berthold Boomer** entspricht der eigentlichen Zielgruppe. Die gleichen Videospiele-Vorlieben gelten für ihn und auch für jüngere Spieler. Denn er steht für Menschen, die Simulation und Aufbau mögen. Sie legen damit auch mehr Wert auf ein eher langsames, nicht action-geladenes Gameplay. Berthold gehört zur *Core Audience*.



**Carola Casual**

Vorlieben	Gute Freunde, Einfache Bedienung
Hobbys	Capoeira-Training, Fotografie
Spielmotivation	Soziale Eingebundenheit
Spiel-Ort	Handy-Spiele wenn unterwegs
Spielerfahrung	Casual – beherrscht Grundsteuerung
Lieblingsspiele	Die Sims, Animal Crossing: PocketCamp

26  
Jahre



„Am liebsten spiel ich zusammen mit meiner Freundin Melissa!“

Vorlieben	Explosionen, Teamspiele
Hobbys	An Konsole Zocken, 3D-Druckerei
Spielmotivation	Kompetenz
Spiel-Ort	Zuhause an der Konsole vorm TV
Spielerfahrung	Hardcore Gamer
Lieblingsspiele	Call of Duty: Black Ops 4



**Henry Hardcore**

„Ha, du Noob!  
Ich bin der Beste!“

33  
Jahre



**Berthold Boomer**

Vorlieben	Familie, Verlässlichkeit, Aufbauspiele
Hobbys	Glasbläserei, den Nachbarn anbrüllen
Spielmotivation	Autonomie
Spiel-Ort	Zuhause am PC
Spielerfahrung	Erfahrener Gamer und PC-Nutzer
Lieblingsspiele	Die Siedler 3, Anno 1800

54  
Jahre



„Seit die Kinder ausgezogen sind, hab ich viel Zeit für Städtebau.“



Abbildung 3.2: Screenshot SimCity (2013): Verwalten einer Stadt [34].

### 3.3 Konkurrenzanalyse: SimCity

Als Inspiration diente hierbei die *SimCity*-Reihe (siehe Abbildung 3.2). Die Spiele wurde von Maxis (Bestandteil von Electronic Arts) entwickelt. In SimCity hat der Spieler die Kontrolle über eine Stadt. Ziel von SimCity ist es, die größte Stadt zu bauen. Dabei liegt es an dem Spieler, die verschiedenen Bedürfnisse der Bewohner auszugleichen (die Bewohner heißen *Sims*) [34]. Das Entwicklerstudio Maxis hat hierzu 8 Thesen aufgestellt, was ein gutes Simulationsspiel („Sim game“) ausmacht [35]:

**Dilemma** Der Spieler muss mit schwierigen Entscheidungen umgehen. Die Stadt hat einen Bedarf nach Industrie, aber gleichzeitig leiden die Bewohner unter Luftverschmutzung.

**Krise** Ab und zu muss der Spieler eine Langzeit-Strategie aufgeben, um einen kurzfristigen Erfolg zu sichern. Wenn z.B. ein Feuer in der Stadt ausbricht, muss der Spieler eventuell die umliegenden Gebäude zerstören, damit sich das Feuer nicht weiter ausbreiten kann.

**Finanzplanung** Der Geldbetrag des Spielers dient gleichzeitig als Begrenzung und Punktestand. Jede Aktion im Spiel kostet Geld – der Spieler muss gut damit umgehen.

**Ästhetik** Wichtig für den Spieler ist die Gestaltung seiner ganz eigenen Stadt. Mit Gebäude-Platzierung oder Dekoration, wie Grünflächen, kann sich der Spieler kreativ ausleben.

**Regeländerung über Zeit** Die Regeln ändern sich über Zeit, um das Spiel interessant zu halten. Zu Beginn ist die Nachfrage nach Industrie groß, im späteren Spielverlauf mehr nach Gewerbe.

**Visuelles Feedback** Ein Simulator ist nichts anderes als eine Sammlung an Funktionen, die auf einem Datensatz arbeiten. Die Hauptinterpretation des Spielers der Daten ist visuell, in diesem Fall große Gebäude in der Stadt.

**Ursache und Wirkung sind nicht sofort offensichtlich** Aktionen können Nebeneffekte haben, die nicht sofort sichtbar sind. Wenn man viel Industrie baut, braucht diese ungebildete Arbeitskräfte. Dadurch wird aber das Bildungslevel der Stadt gesenkt. Den Reiz des Spiels macht es aus, diese Beziehungen zu verstehen.

**Spieler dürfen ihre eigenen Kreationen zerstören** Wenn es mal nicht so gut läuft, darf der Spieler auch gern seine eigene Stadt zerstören. Einfach nur aus Spaß.

### 3.4 Anforderungen Spieleprototyp

Die Anforderungsanalyse von Videospielen ist sehr ähnlich zu der Anforderungsanalyse von Software-Produkten. Aber der Unterschied ist, dass bei Videospielen vor allem die *Non-Functional-Requirements* wichtig sind. Callele [36, S. 249] schreibt hierzu, dass Faktoren wie *Spaß*, *Story*, *Ästhetik* und *Flow* die Anforderungen dominieren. Das Problem scheint aber momentan zu sein, dass sich Faktoren wie *Spaß* nur schwer quantifizieren lassen. Auch John Ferrara schreibt hierzu: „Fun is, of course, the most important quality of games, and the most compelling reason why people want to play. [...] Unfortunately, it’s not that easy. Instead, fun emerges from the experience when all of the elements work well together.“ [1, S. 33]

Diese *Experience* kann laut Celia Hodent auch als *Game UX* bezeichnet werden. Sie beschreibt die komplette User Experience, die ein Spieler mit einem Videospiel hat. Dazu gehören die Wahrnehmung, Verständnis, Interaktionen und Gefühle, die bei der Interaktion mit Videospielen auftreten [30, S. 105f]. Dazu ähnlich ist die von Ferrara beschriebene *Player Experience*: „Player experience is fundamentally different from the user’s experience of the nongame user interfaces that we [UX-Designer] normally design.“ [1, S. 28] Nachfolgend wird ein Modell zur Beschreibung der Player Experience verwendet – sie ist insgesamt sehr ähnlich zur normalen User Experience. Die Bestandteile sind wie folgende [1, S. 27–33]:

1. **Motivation** Siehe Abschnitt 3.1.
2. **Bedeutungsvolle Entscheidungen** Spiele müssen es dem Spieler erlauben, den Ausgang des Spiels mit Entscheidungen verändern zu können. Die Herausforderung liegt dabei, durchweg die richtigen Aktionen zu machen. Das kann sich entweder durch Langzeit-Strategien (z.B. in Schach) äußern oder bei Action-Spielen, die stark auf Reflexe und Reaktion ausgelegt sind.
3. **Balance** Balance beschreibt, wie gut die Elemente eines Spiels zusammenarbeiten. Das System soll angemessen fordernd und gleichzeitig fair sein. So sollte der Spieler beispielsweise in einem RPG nicht zu viele Erfahrungspunkte zu Beginn bekommen. Andernfalls kann er mit dem hohen Level alle Gegner sofort besiegen.
4. **Usability** Sie umfasst zum einen die Benutzbarkeit des Spiel-Interfaces. Zum anderen ist die Usability der Spiel-Erfahrung wichtig. Spieler sollten verstehen können, welche Aktionen welche Reaktionen hervorrufen.

5. **Ästhetik** Ästhetik umfasst die komplette sensorische Erfahrung des Videospiele. Dazu zählen Bilder, Geräusche, Musik und haptisches Feedback (z.B. Rumble bei Controller). Diese Gestaltungselemente haben einen starken Einfluss, wie der Ton des Spiels wahrgenommen wird.

Aus der *Player Experience* können so die Anforderungen für den Spiele-Prototypen dieser Arbeit abgeleitet werden. Die Anforderungen sind folgende:

1. Der Spiele-Prototyp soll Spaß machen.
2. Der Spiele-Prototyp soll gut bedienbar sein.
3. Der Spiele-Prototyp soll eine ästhetisch ansprechende Gestaltung haben.
4. Der Spiele-Prototyp soll die gleiche Spielerfahrung auf den drei verschiedenen Endgeräten Mobile, Smart TV und PC ermöglichen.
5. Der Spieler soll sich im Spiele-Prototyp durch Gebäudebau ausdrücken können.
6. Der Spieler soll durch den Schwierigkeitsgrad angemessen gefordert sein.
7. Der Spieler soll Grundkonzepte der Stromversorgung verstehen.

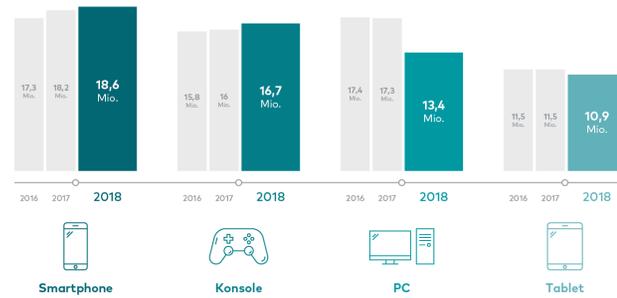


Abbildung 3.3: Die beliebtesten Spiele-Plattformen in Deutschland. Stand: Juli 2019 [37]

### 3.5 Anforderungen Endgeräte

Auf Basis der Spiele-Prototypen-Anforderungen entstehen implizite Anforderungen für die verwendeten Endgeräte. Wie muss sich die Steuerung verhalten, damit die gleiche Erfahrung auf einem Smartphone und einem Smart TV möglich ist? Dafür muss sich das Interface und die Benutzereingabe anpassen.

Je nach Endgerät ändern sich auch die dazugehörigen Input-Möglichkeiten. Diese können Maus, Tastatur, Controller, Beschleunigungssensor, Touchscreen, Spracheingabe, Gestenerkennung, usw. sein. Die Anforderungen an gängige Peripherie-Geräte sind in der ISO 9241-410 hinterlegt [38]. Daneben haben Albassam und Hassan [39, S. 7] bewiesen, dass das Verwenden einer gleichen Spiel-Programm-Basis für verschiedene Endgeräte durchaus sinnvoll ist. In ihrem Versuch haben sie ein Spiel für Desktop und Mobil programmiert und konnten ca. 85 % des Codes wiederverwenden.

Nachfolgend werden nun die Anforderungen an UI und Steuerungs-Eingabe untersucht. Die drei untersuchten Endgeräte dabei sind: 1) Mobile, 2) Smart TV und 3) Desktop.

Zunächst wurden die Eigenschaften der Bildschirme für alle drei Endgeräte gemeinsam untersucht (siehe Tabelle 3.2). Der Bildschirm hat einen starken Einfluss, wie das Screendesign gestaltet werden muss. Ein überraschendes Ergebnis war zum Beispiel, dass der Smart TV (in diesem Fall Amazon Fire TV) eine *Pixel Ratio* von 2 hat, obwohl die physikalische Auflösung dafür deutlich zu niedrig wäre. Das liegt daran, dass es sich bei Smart TVs um ein sogenanntes *10-foot UI* handelt [40].

Gerät	Dimension (Breite * Höhe)	Auflösung (angegeben)	Auflösung (physikalisch)	Bildschirm- diagonale	Betrachtungs- abstand
Smartphone <sup>2</sup>	2.560 * 1.440 px	560 dpi	554 dpi	5,3 Zoll	30 cm
Rechner <sup>2</sup>	1.920 * 1.080 px	96 dpi	91 dpi	24 Zoll	50 cm
Smart TV <sup>2</sup>	1.920 * 1.080 px	320 dpi	47,9 dpi	46 Zoll	300 cm

Tabelle 3.2: Bildschirm-Eigenschaften der drei untersuchten Endgeräte.

<sup>2</sup> Bei den hier untersuchten Geräten handelt es sich um folgende: 1. Für Mobile wurde das Nokia 8 [41], 2. für den Rechner der Monitor Asus VS248H [42] und 3. für den Smart TV der Monitor Sharp LC46LE542E [43], zusammen mit einem *Amazon Fire TV Stick 2* [44], verwendet.

### 3.5.1 Mobile

Mobilgeräte (oder Smartphones) sind kleine, portable Rechner, die mit Touch bedient werden. Mit ihnen kann eine Vielzahl von Aktionen durchgeführt werden – unter anderem Telefonieren, Nachrichten schreiben und empfangen, Fotografieren und Gaming. Besonders der Mobile Gaming-Bereich ist in den letzten Jahren rasant gewachsen. In den Vereinigten Staaten ist das Smartphone mit 60 % die am häufigsten genutzte Spieleplattform [45, S. 6]. Auch in Deutschland wurden 2015 65 % aller Spiele für Tablet/Smartphone entwickelt [46, S. 85]. Daraus haben sich vielfältige Monetarisierungs-Optionen für die Videospiele-Publisher ergeben. Im mobilen Bereich ist das Geschäftsmodell des *Free to Play* (F2P) am weitesten verbreitet. Spieler können dabei das Videospiele zunächst kostenfrei herunterladen. Aber im weiteren Verlauf sollen sie mithilfe von „Microtransactions“ Kleinstbeträge überweisen [46, S. 81]. So wurden 2019 68,5 Milliarden Dollar mit Mobile Games umgesetzt – das sind 45 % des gesamten weltweiten Spielemarktes [47].

Mobil-Geräte gelten als einfach bedienbar durch den Touchscreen. Laut Fritsch et al. [48] (zitiert nach [49]) zeichnet sich die Usability bei Mobile Games wie folgt aus:

1. Einfach und schnell erlernbar
2. Effizient zu benutzen
3. Leicht erinnerbar
4. Niedrige Fehlerwahrscheinlichkeit
5. Angenehm zu benutzen

Das heißt, damit das gleiche Spielerlebnis für Mobilgeräte möglich ist, muss die Steuerung für Touchscreen ausgelegt sein. Menü-Hierarchien sollten möglichst flach sein, um schnell ins Spiel starten zu können. Die größte Herausforderung für das UI ist der verfügbare Platz: UI-Elemente sollen platzsparend angelegt und nur die wichtigsten Elemente (wie Geldanzeige) soll jederzeit sichtbar sein.



Abbildung 3.4: Vergleich zwischen verschiedenen Spiele-Controllern. [50]–[53]

### 3.5.2 Smart TV

Smart TVs werden hauptsächlich zum Ansehen von Filmen und Serien verwendet. Jedoch kann man mit ihnen auch Videospiele spielen. Wie in Abbildung 3.3 zu sehen ist, stellen Konsolen die zweit-beliebteste Plattform für Videospiele in Deutschland dar. Die aktuelle Konsolengeneration, mit *PlayStation 4* (Sony), *Xbox One X* (Microsoft) und *Switch* (Nintendo), zeichnet sich durch sehr leistungsstarke Endgeräte aus. Vor allem die Controller tragen zur guten Spielerfahrung bei.

Der Fokus in dieser Arbeit soll aber nicht sein, wie sich ein komplexes Videospiele auf einer gewöhnlichen Spielekonsole gestalten lässt. Stattdessen ist es viel interessanter, welche Herausforderungen für das Prototypen-UI gelöst werden müssen, damit er auch mit sehr eingeschränkten Spiele-Controllern gespielt werden kann. Daher ist in diesem Fall die Wahl der Konsole auf eine eher unübliche Geräte-Familie gefallen: Smart TVs. Von den gängigen Smart TV-Modellen wurde exemplarisch *Amazon Fire TV* ausgewählt. Wie man in der Abbildung 3.4 sehen kann, unterscheidet sich der Controller stark vom klassischen Gamepad bei Konsolen-Marktführer.

Für den Spiele-Prototyp konnten nur wenige Tasten der *Amazon Fire TV Stick-Remote* verwendet werden. Das sind die Pfeilrichtungen, Bestätigen, Zurück und Menü (siehe Abbildung 3.5). Andere Smart TV-Modelle bieten mehr Input-Möglichkeiten – in diesem Fall wurde die am wenigsten variable Steuerung angenommen.

Das Display bei einem Smart TV ist ein Fernsehmonitor. Er wird aus einer großen Entfernung betrachtet – dementsprechend muss das Interface auch groß gestaltet werden. Tatsächlich ist die Gestaltung für ein Smart TV-UI sehr ähnlich zur Gestaltung eines Mobile-UIs. Ziel von solchen Geräten ist es, schnell Inhalt bereitzustellen („Deliver Content fast“ [40]).

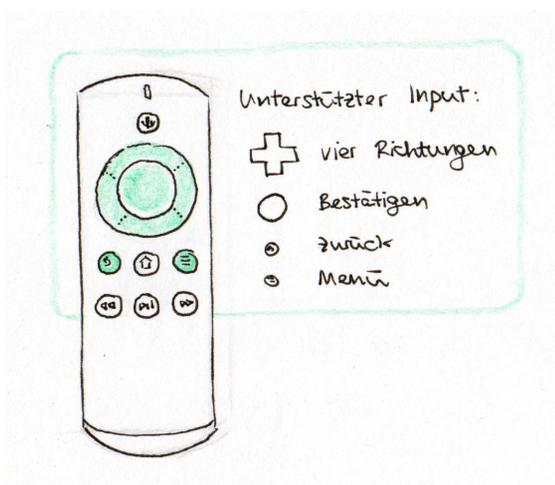


Abbildung 3.5: Unterstützter Input der *Amazon Fire TV 2-Remote*.

### 3.5.3 PC / Laptop

PCs oder Laptops zeichnen sich dadurch aus, dass der Benutzer das Gerät nach eigenen Bedürfnissen mit Software anpassen kann. Die Leistung der Spiele-Plattform ist dabei stark unterschiedlich.

Die Auflösung und Größe des Displays kann dabei stark variieren. Für gewöhnlich haben PC-Nutzer in Deutschland aber eine Bildschirmauflösung von  $1.920px \times 1.080px$  [54]. Man kann sagen, dass Spielen an Desktop unter Optimalbedingungen stattfindet: Üblicherweise werden PCs und Laptops so aufgestellt, dass das Display nicht durch eintreffende Sonnenstrahlen spiegelt. Außerdem ist die Wahrscheinlichkeit für Störungen zuhause weitaus geringer als unterwegs.

Normalerweise werden Desktop-Spiele mit Tastatur und Maus gespielt. Viele Gamer besitzen aber auch noch zusätzlich ein Gamepad (z.B. für Plattformer oder Rennspiele). Die Tastatur ist für Power-User sehr gut bedienbar, mit ihr können Tastenkürzel eingegeben werden. Ansonsten sind spezielle Tasten wie die Cursor-Tasten oder ESC für Menü/Abbrechen sehr wichtig. Die Computer-Maus selbst ist auch ein sehr präzises Eingabegerät. Sie kann viele Zustände wie Hover, Dragging oder Klicken annehmen. Darum ist in einem Aufbau-Simulations-Spiel die Maus das wichtigste Input-Gerät.



---

## Kapitel 4

# Konzept

Das Konzept definiert den Spiele-Prototypen, der in dieser Arbeit erstellt wurde, auf Basis der Analysephase. Zunächst wird der Prototyp grob mit einem *Game Design Document* beschrieben, danach folgen Use-Cases, Skizzen, Wireframes und Ingame-Assets. Abschließend folgt der Klick-Prototyp, der als Vorbereitung für die Implementierung diene.

## 4.1 Game Design Document

Der Spiel-Prototyp, der in dieser Arbeit erstellt wird, ist im Videospiel-Entwicklungsprozess der Preproduction-Phase zuzuordnen (siehe Kapitel 2.2.2). Es geht nicht darum, ein vollwertiges Spiel zu erstellen. Stattdessen soll überprüft werden, wie sich das Interface auf die Spiel-Wahrnehmung auswirkt.

Videospiele werden üblicherweise mithilfe eines *Game Design Documents* (GDD) konzeptioniert. Eine allgemeine Definition ist laut Callele [36, S. 242]: „A game design document is a preproduction artifact designed to capture a creative vision.“ Um genauer zu sein, ein GDD enthält Angaben zum Gameplay, Aussehen, Sounds, Level Design, Animationen und anderen Elementen eines Spiels [36]. Deswegen wurde in dieser Arbeit der Spiele-Prototyp mithilfe von Bausteinen eines GDD definiert.

### Thema / Setting / Genre

- Thema: Simulationsspiel eines Stromversorgers
- Setting: Nach den 2000ern
- Genre: Simulation / Kraftwerk-Aufbau

### Core Gameplay Mechaniken

- Navigation in der Spielwelt (Verschieben, Drehen, Zoomen)
- Auswählen von Gebäuden (Info und Bearbeiten)
- Platzieren von Gebäuden (Gebäude-Bau; Aufbau eines Stromnetzes)
- Erforschen von Technologien
- Ressourcen-Abbau
- Aufbau von Warenkreisläufen
- Handel mit Ressourcen und Strom
- Landschaft verändern durch heben/senken
- Feedback von Menschen lesen und auf Wünsche eingeben

### Zielplattformen

- Mobile (Android)
- Smart TV (*Amazon Fire TV*; Android)
- PC (Windows)

Der Spiele-Prototyp wurde so aufgebaut, dass das gleiche Spielerlebnis auf drei verschiedenen Plattformen möglich wäre. Dabei ist zu beachten, dass jede Plattform dabei eigene Anforderungen an die Gestaltung hat (siehe Kapitel 3.5).

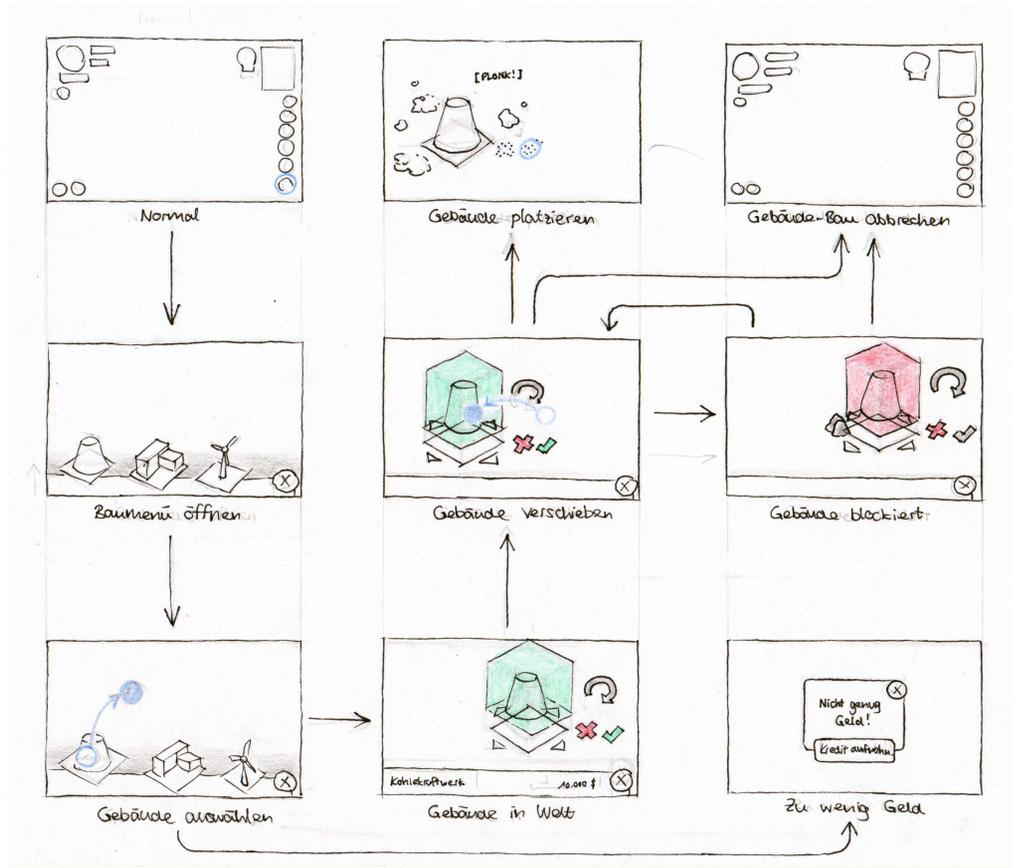


Abbildung 4.1: Use Case: Verlauf für das Platzieren eines Gebäudes.

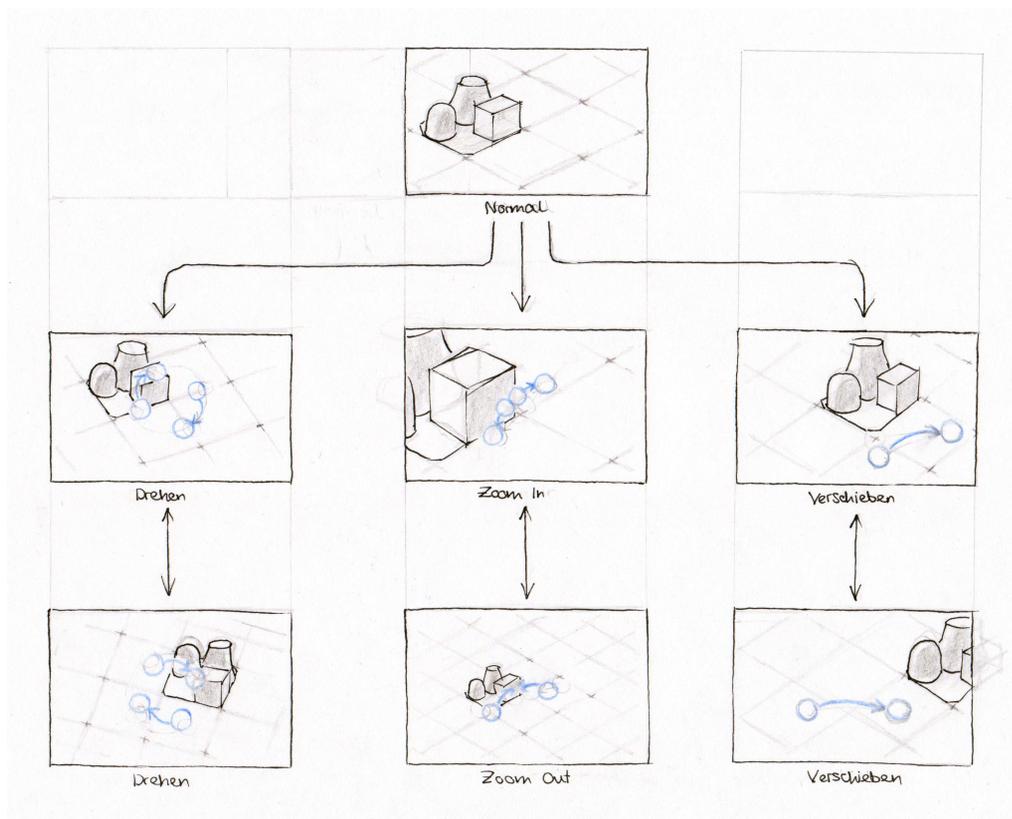


Abbildung 4.2: Use Case: Bewegen der Kamera in der Spielwelt.

## 4.2 Use Cases

Ein *Use Case* ist definiert als eine Abfolge von Aufgaben, die ein System durchführen kann. Dabei interagiert das System mit den Benutzern und gibt ein messbares Ergebnis zurück [55, S. 494]. Das heißt, mithilfe von Use Cases kann beschrieben werden, welche Aktionen ein System dem Benutzer ermöglicht. Anhand der Anforderungen wurden insgesamt sieben Use Cases ermittelt, die der Spiel-Prototyp erfüllen soll.

Use Cases werden üblicherweise in strukturierter und gut lesbarer Sprache ausgeschrieben. An dieser Stelle wird *Gherkin* verwendet, um die Use-Case-Beschreibung festen Regeln folgen zu lassen. *Gherkin* ist eine Sprache zur Beschreibung von Use-Cases, entwickelt von *Smart Bear Software*. Sie hat einen festen Aufbau und spezielle Schlüsselwörter wie *Given* und *Then* (in dieser Arbeit wird die deutsche Fassung verwendet) [56] [57].

Die insgesamt 7 Use Cases lassen sich mit *Gherkin* wie folgt beschreiben:

### Use-Case 1 – Bewegen der Kamera in der Spielwelt

**Szenario:** Spieler sieht sich in der Spielwelt um.

Angenommen der Spieler hat das Spiel gestartet

Wenn der Spieler die Spielwelt draggt

Dann bewege die Kamera mit der Drag-Bewegung

### Use-Case 2 – Baumenü das Info-Fenster anzeigt

**Szenario:** Spieler kann Gebäude-Informationen im Baumenü abrufen.

Angenommen der Spieler hat das Baumenü geöffnet

Wenn der Spieler auf eine Gebäude-Vorschau klickt

Dann öffnet sich ein Fenster mit Infos zum geklicktem Gebäude

### Use-Case 3 – Bauen eines Kraftwerkes

**Szenario:** Spieler kann ein Kohlekraftwerk platzieren.

Angenommen der Spieler hat das Baumenü geöffnet

Und die angezeigte Kategorie des Baumenüs ist "Kraftwerke"

Wenn der Spieler ein Gebäude vom Baumenü in Spielwelt draggt

Und der Spieler genug Geld für ein Kohlekraftwerk hat

Dann wird ein neues Kohlekraftwerk in der Spielwelt platziert

Und ein Platzier-Dialog wird für das Kraftwerk angezeigt

Und der Spieler bezahlt Preis für Kohlekraftwerk

Und das Baumenü schließt sich

#### Use-Case 4 – Aufnehmen eines Kredites

**Szenario:** Spieler kann Kredit aufnehmen.

Angenommen das Finanz-Info-Fenster ist geöffnet

Wenn der Spieler auf den "Geld Leihen"-Button klickt

Dann gib dem Spieler 10.000 \$

Und erhöhe die Schulden des Spielers um 10.000 \$

#### Use-Case 5 – Infos zur Stromversorgung

**Szenario:** Spieler kann Infos zur Stromversorgung abrufen.

Angenommen der Spieler hat das Spiel gestartet

Wenn der Spieler auf den Stromzähler im Spiel-UI klickt

Dann blende das Strom-Info-Fenster ein

#### Use-Case 6 – Verschieben des gebauten

**Szenario:** Spieler kann bereits gebautes Gebäude verschieben.

Angenommen ein Gebäude ist in der Spielwelt gebaut

Wenn der Spieler 500 ms lang auf ein Gebäude klickt

Dann versetze Gebäude in Platzier-Zustand

Und blende Platzier-Dialog ein

#### Use-Case 7 – Einblenden von Infos in der Spielwelt

**Szenario:** Spieler kann Informationen in der Karte einblenden.

Angenommen der Spieler hat das Spiel gestartet

Wenn der Spieler auf das Info-Symbol im Spiel-UI klickt

Dann blende Infos in der Spiel-Welt zur Stromversorgung ein.

### 4.3 Skizzen und Wireframes

Wireframes zeigen das grundlegende Layout der wichtigsten Elemente einer Benutzerschnittstelle. Sie werden meist nur gezeichnet und nicht weiter ausgestaltet [29, S. 168].

Bei dieser Arbeit wurde nach anfänglichen Skizzen auf ein sehr detaillierte Wireframes gesetzt. Sie dienten zur Erarbeitung der relevanten Features, die der Spiele-Prototyp haben soll. Außerdem waren sie die Basis für den späteren Klick-Prototypen.

#### 4.3.1 Erstes Konzept

Das erste Konzept diente als Probelauf für die Gestaltung des Prototypen. Es war ein Mockup, wie ein Kraftwerk-Simulations-Spiel aussehen könnte. Dafür wurde ein Interface gestaltet, dass sich stark durch rechteckige Formen auszeichnet. Für einen ordentlichen Eindruck wurde viel Platz zwischen den UI-Elementen gelassen.

Wichtig war, dass das UI sich je nach Endgerät anpasst. So hat z. B. die Smart TV-Ansicht einen Overscan-Bereich, wie bei Smart TV-Apps vorgeschrieben ist (siehe „Overscan and the Safe Zone“ in [40]). Auch wurde beispielsweise die Navigation für Mobile als Bottom-Leiste gestaltet.

Während des Zeichnens dieser Wireframes ist auch die Idee entstanden, dass der Spieleprototyp eine isometrische Spielwelt hat. Diese Kamera-Ansicht ist typisch für Aufbau-Simulations-Spiele.

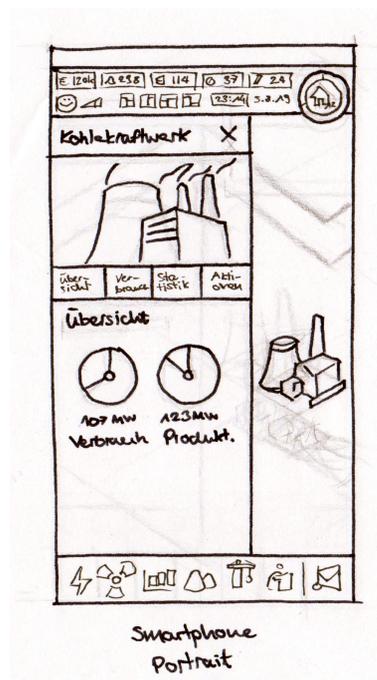


Abbildung 4.3: Wireframes: Smartphone-Ansicht des ersten Konzeptes.

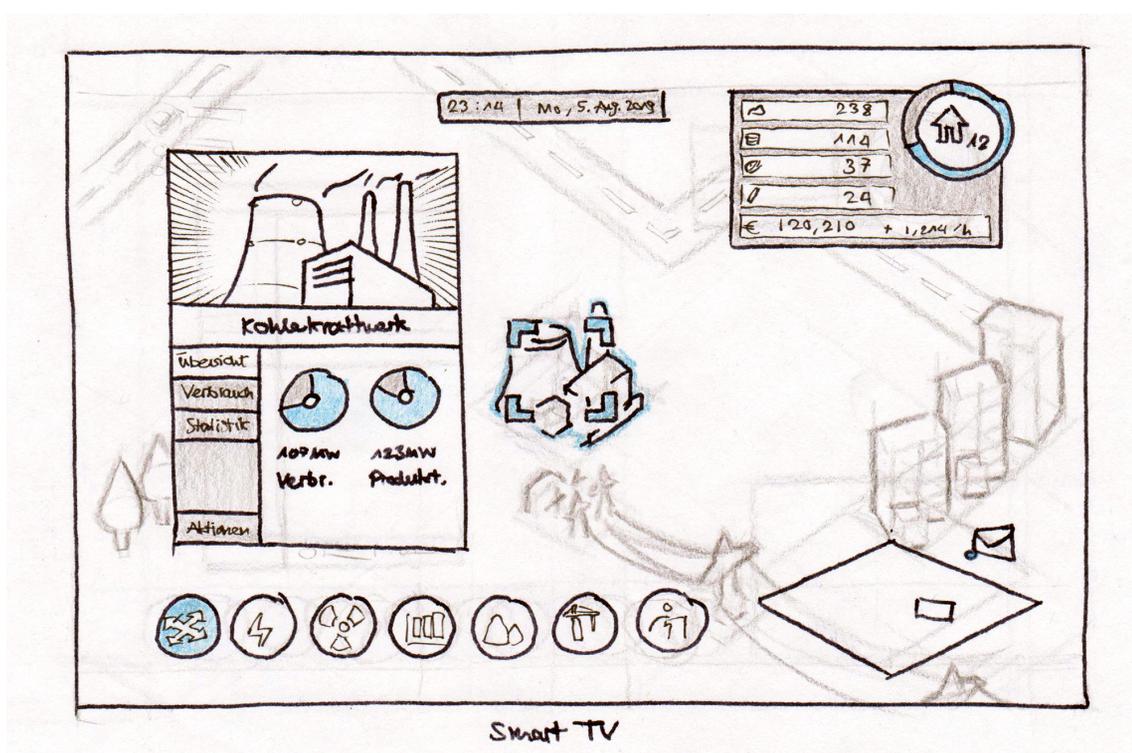


Abbildung 4.4: Wireframes: Smart TV-Ansicht des ersten Konzeptes.

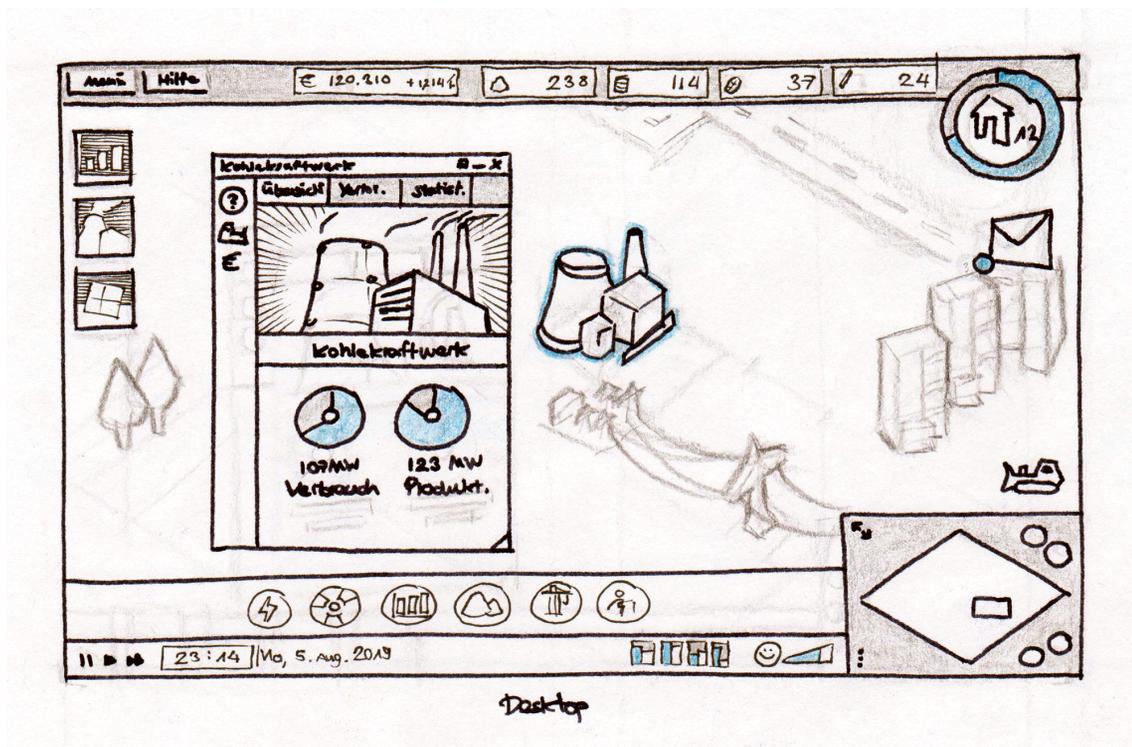


Abbildung 4.5: Wireframes: Desktop-Ansicht des ersten Konzeptes.

### 4.3.2 Wireframes Mobile

Während der zweiten Iteration wurde erarbeitet, welche Daten für den Spiele-Prototypen relevant sind. Im Zuge dessen wurden einfache Skizzen angefertigt.

In der ersten Iteration war zu sehen, dass ein Spiele-UI sich für alle drei Endgeräte anpassen lässt. Gemäß *Mobile First* ist es eine gute Praxis, erst mit dem kleinsten Interface-Layout anzufangen und dann später hoch zu skalieren.

Dabei ging die Gestaltung hin zu modularen, kleineren Fenstern. Anhand des Game Design Documents und den Spiele-Prototyp-Anforderungen wurde ermittelt, was der Spieler überhaupt im Spiel mit dem UI machen kann. Mit großen Interfaces (wie der Finanzübersicht) wurde getestet, wie komplex ein UI für Mobile werden darf. Insgesamt wurden neun Screens erarbeitet, die folgende Spiele-Features abbilden:

- Overview (Abb. 4.6)
- Detailansicht (Abb. 4.7 und 4.8)
- Finanz-Übersicht (Abb. 4.9)
- Finanz-Verlauf (Abb. 4.10)
- Stromproduktion (Abb. 4.11)
- Kraftwerksliste (Abb. 4.12)
- Forschungsbaum (Abb. 4.13)
- Ressourcenbilanz (Abb. 4.14)
- Handel (Abb. 4.15)

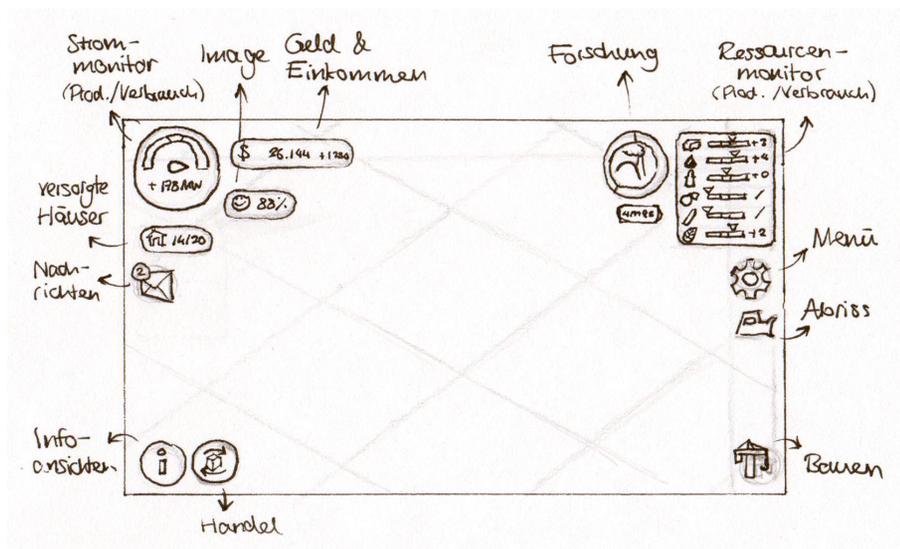


Abbildung 4.6: Wireframes Mobile: Spielübersicht.

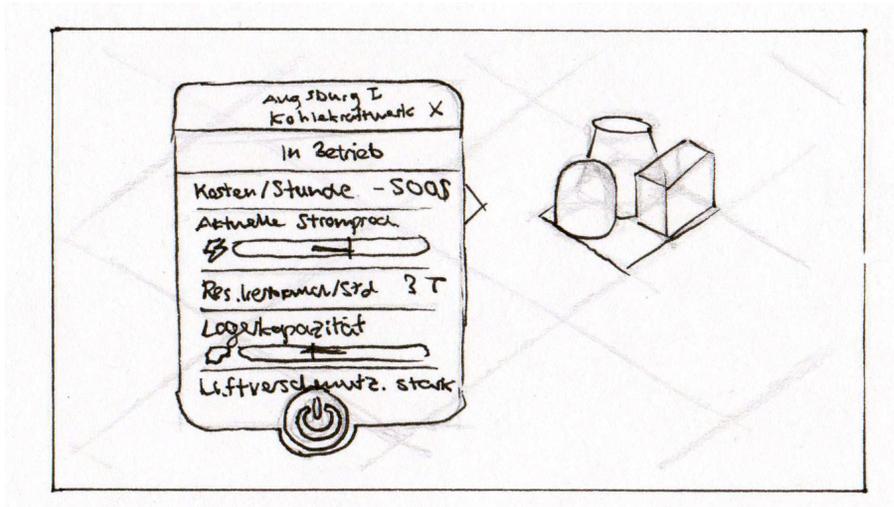


Abbildung 4.7: Wireframes Mobile: Detailansicht von Kraftwerk (Variante A).

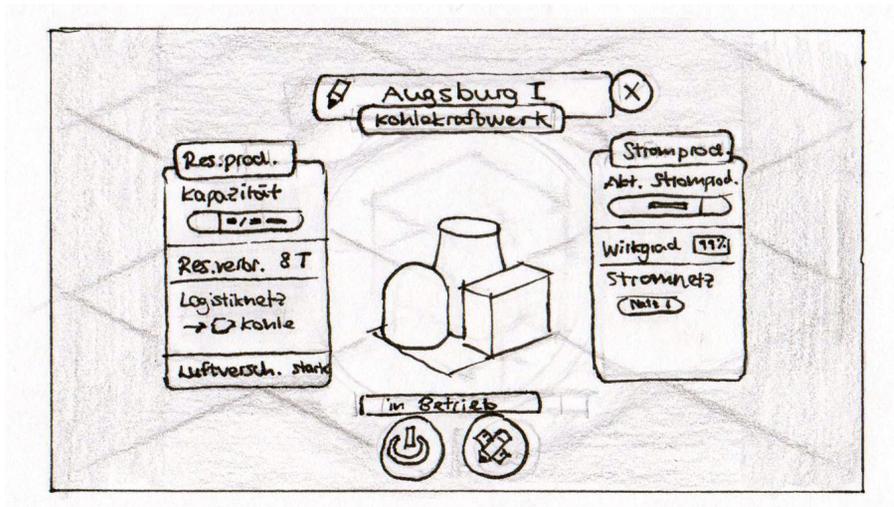


Abbildung 4.8: Wireframes Mobile: Detailansicht von Kraftwerk (Variante B).

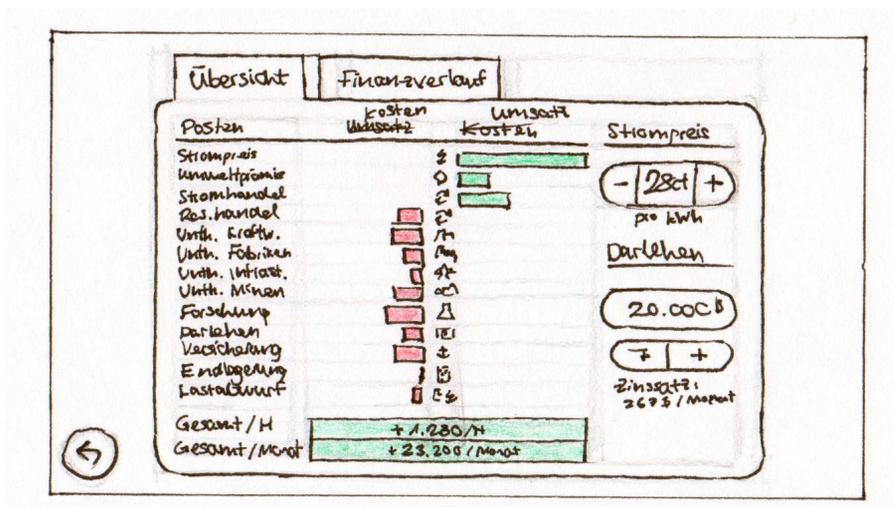


Abbildung 4.9: Wireframes Mobile: Finanzübersicht.

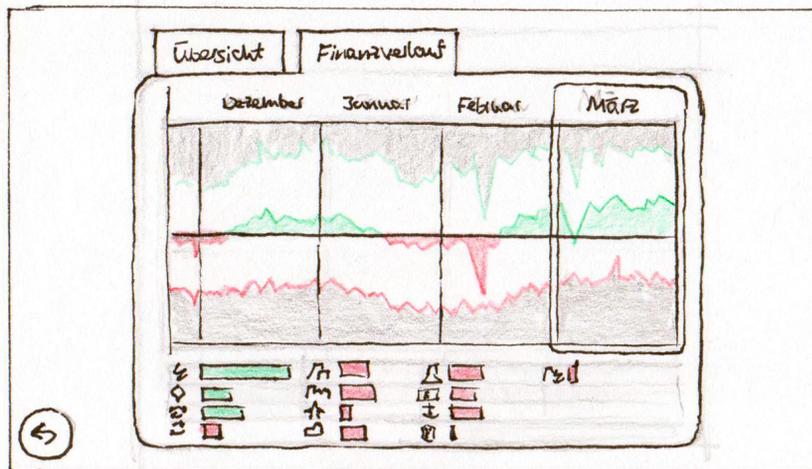


Abbildung 4.10: Wireframes Mobile: Finanzverlauf.

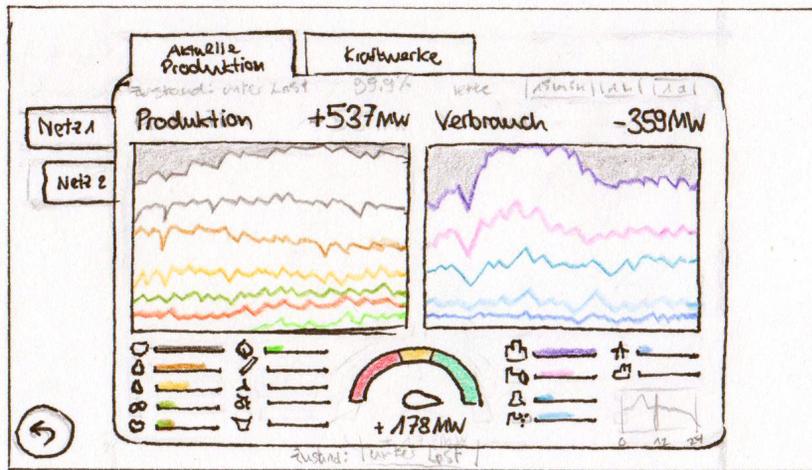


Abbildung 4.11: Wireframes Mobile: Aktueller Stromproduktion.

The wireframe shows a mobile interface for 'Kraftwerksliste'. At the top, there are two tabs: 'Aktuelle Produktion' and 'Kraftwerke'. Below the tabs is a table with the following columns: 'Kraftwerk Name', 'Stromprod.', 'Verbrauch/Nachschub', 'Ressourcenbr.', and 'Status'. The table contains four rows of data.

	Kraftwerk Name	Stromprod.	Verbrauch/Nachschub	Ressourcenbr.	Status
Netz 1	Kohle Augsburg 1	1700MW			In Betrieb
Netz 2	Kohle Augsburg 2	2000MW			in Betrieb
	Heizöl Braunschweig 1	1250MW			in Betrieb
	Heizöl Berlin 1	1200MW			Wartung

A back arrow is located in the bottom left corner.

Abbildung 4.12: Wireframes Mobile: Kraftwerksliste.

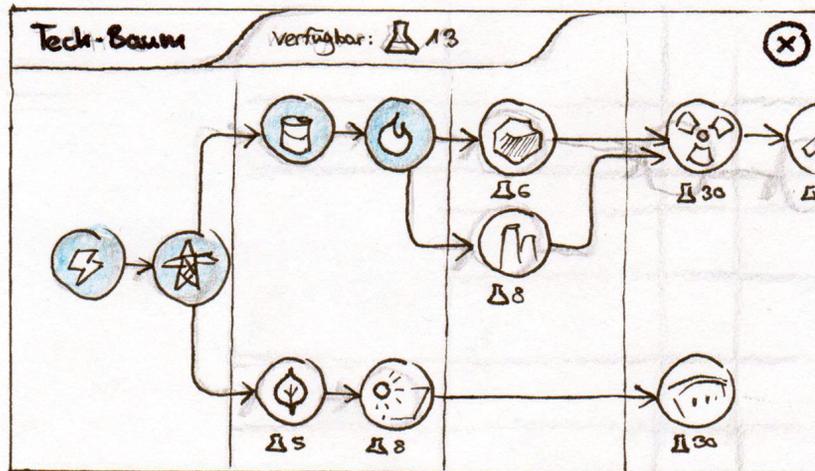


Abbildung 4.13: Wireframes Mobile: Forschungsbaum.

Resource	Prod.	Verbr.	Bilanz	Lager
Kohle	+43	-32	+11	
Erdgas	+36	-44	-8	
Rohöl	+47	-32	+15	
Müll	+40	-40	0	
Uran	+28	-40	-12	
Biomasse	+13	-12	1	
Atomabfall	+8	-10	-2	
Atomabfall	+7	-1	+6	
Heizöl	+24	-24	0	

Abbildung 4.14: Wireframes Mobile: Ressourcen-Übersicht.

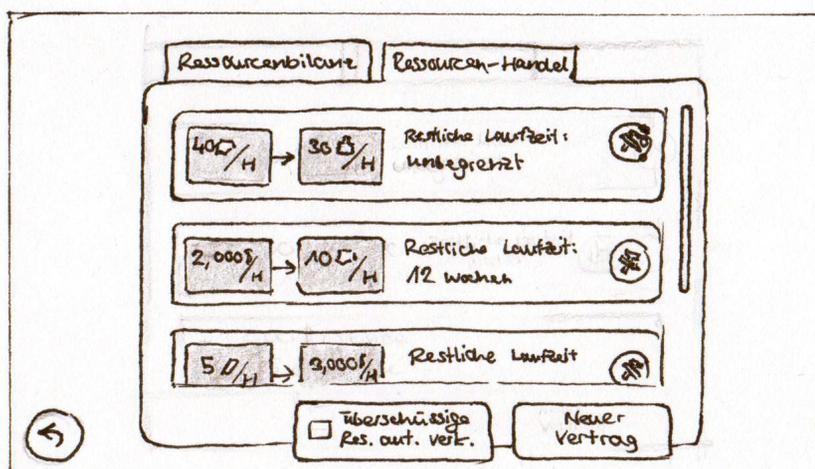


Abbildung 4.15: Wireframes Mobile: Handelsübersicht.

### 4.3.3 Wireframes Smart TV

Das Screendesign von Mobil- und Smart TV-Apps ist sich von den Anforderungen her sehr ähnlich. Daher wurde in diesem Fall aus dem untersuchten Mobile Screen (siehe Abbildung 4.18) ein Screen für den Smart TV entwickelt.

#### Smart TV-Cursor

Die Input-Möglichkeiten bei einem Smart TV sind stark begrenzt. Trotzdem muss der Spieler Objekte in der Spielwelt anwählen können. Dies ist möglich mit einem Cursor (siehe Abbildung 4.17). Aus diesem Grund wird die Interaktion in zwei Zustände aufgeteilt:

**Freies Umsehen** Der Benutzer kann die Kamera frei in der Spielwelt bewegen. Beim Klicken des Bestätigen-Buttons öffnet sich die Action Bar und der Zustand wechselt zu „Menü-Navigation“.

**Menü-Navigation** Der Benutzer kann sich durch eine vorher festgelegte Menüstruktur mithilfe der Richtungstasten Links ←, Rechts →, Oben ↑ und Unten ↓ bewegen. Um zum „Freien Umsehen“ zurückzukehren, kann der Spieler den Zurück-Button drücken.

Da die einzige Interaktion mit der Spielwelt über einen Cursor möglich ist, muss dieser sehr flexibel einsetzbar sein. Die Zustände, die der Cursor annehmen kann, sind folgende (siehe Abbildung 4.16):

<b>Normal</b>	Cursor ist frei bewegbar in der Spielwelt. Normalzustand.
<b>Pressed</b>	<i>Modifier</i> , wenn der Bestätigen-Button gedrückt wird.
<b>Hover</b>	Cursor befindet sich über ein klickbares Objekt.
<b>Not allowed</b>	Cursor darf über Objekt keine Aktion ausführen.
<b>Out of focus</b>	Cursor ist inaktiv und reagiert nicht auf Input.
<b>Scroll</b>	Am Rand des Bildschirms scrollt der Cursor automatisch in eine Richtung.

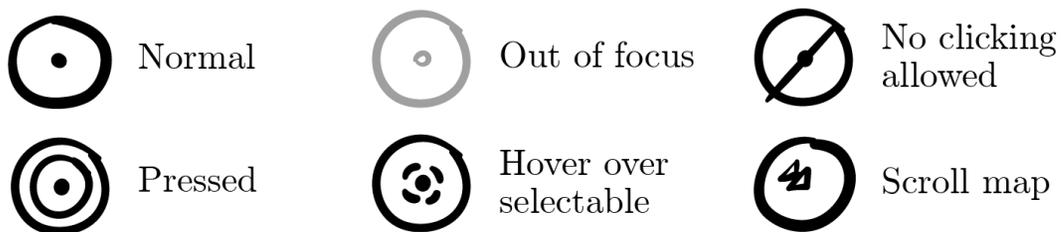


Abbildung 4.16: Skizze: Verschiedene Zustände des Smart TV-Cursors.

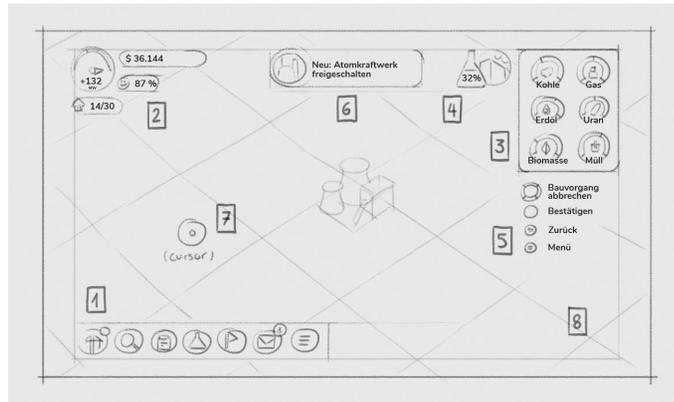


Abbildung 4.17: Wireframes Smart TV: Aufbau des UIs.

### Aufbau des UI

Wie man in Abbildung 4.17 sehen kann, ist das Interface in mehrere Blöcke aufgeteilt. Für die verschiedenen Endgeräte tauchen dabei die gleichen Blöcke auf – aber mit unterschiedlicher Gestaltung. Die Bausteine des Interfaces lassen sich wie folgt beschreiben:

1. **Action Bar** Das Interaktionsfeld bei der Smart TV-Ansicht. Man kann mit Links (←) und Rechts (→) im gleichen Level navigieren. Mit Oben (↑) kann man eine Ebene hinabsteigen und mit Unten (↓) eine Ebene hinaufsteigen. Siehe Abbildung 4.20 die Menüstruktur bei Smart TV.
2. **State Indicator** Beschreibt die Gruppe an Datenanzeigen. Der Stromzähler gibt den aktuellen Leistungsstand der Stromnetze an. Daneben sind (im Uhrzeigersinn) die Finanzen, die Zufriedenheit der Stromkunden und die Anzahl der mit Strom versorgten Gebäuden.
3. **Ressourcen-Monitor** Listet die wichtigsten Ressourcen auf und zeigt das Verhältnis zwischen aktueller Produktion und Verbrauch.
4. **Forschungsanzeige** Fortschritt des aktuellen Forschungsprojektes. Der Prozentwert steigt langsam mit verstreichender Zeit.
5. **Bedienhilfe** (Nur Smart TV) Listet alle Aktionen auf, die im aktuellen Interaktionszustand möglich sind. Zeigt außerdem für die Aktion nötige Taste an. Sie soll den Benutzer unterstützen, da die Button-Belegung kontextabhängig ist.
6. **Toast** Kleines Infofenster, das bei Events im Spiel mit einer Meldung kurz eingeblendet wird.
7. **Cursor** (Nur Smart TV) Mit Pfeiltasten (← → ↑ ↓) steuerbarer Cursor, um mit Spielwelt zu interagieren (z.B. Gebäude anklicken oder die Karte navigieren). Kann nur verwendet werden, wenn Interaktionszustand bei „Freies Umsehen“ ist.
8. **Spielwelt** Darunterliegende dreidimensionale Spielwelt. Die Kamera für die Spielwelt kann verschoben, gedreht und gezoomt werden.

## Screenshots

Wurde in der *Action Bar* ein Gebäude zum Bauen ausgewählt, befindet sich der Spieler im Baumenü (siehe Abbildung 4.21, 4.22, 4.23).

Der erste Screen (Abbildung 4.21) zeigt einen Überblick über Kraftwerke, die der Spieler bauen kann. Am unteren Ende der Leiste befindet sich ein scrollbares Menüband, auf dem alle Gebäude der Kategorie „Kraftwerke“. Beim ersten Aufrufen des Screens bekommt der Spieler ein Onboarding durch einen virtuellen Charakter. Das Onboarding [30, S. 180] dient dazu, dem Spieler die wichtige Spielmechanik des „Platzierens eines Gebäudes“ beizubringen.

Die Abbildung 4.22 zeigt das Infofenster zum aktuell ausgewählten Gebäude. Neben Titel und Preis wird auch der benötigte Rohstoff für das Kraftwerk angezeigt. Danach folgt die Kapazität zur Stromgenerierung und die verursachte Luftverschmutzung. Wenn man das Menüband mit Links (←) und Rechts (→) navigiert, wird das aktuell ausgewählte Element hervorgehoben. Mit dem Zurück-Button kann man das Baumenü verlassen. Beim Drücken auf Bestätigen folgt der nächste Schritt des Bau-Prozesses.

Als Drittes (Abbildung 4.23) folgt die Platzierung des zuvor ausgewählten Gasturbinen-Kraftwerkes. Hier kann der Spieler nun das zu bauende Gebäude in der Spielwelt platzieren. Die Vorschau des Gebäudes folgt dabei dem Cursor – das Gebäude kann aber nur in einem Raster platziert werden. Deswegen folgt der Ansicht ein transparenter Würfel, der die tatsächliche Position im Raster zeigt. Beim Platzieren darf das Gebäude auch nicht mit anderen Hindernissen, z.B. Bäumen oder anderen Kraftwerken, kollidieren. Bewegt man den Cursor, bewegt sich auch die Kamera mit. Dadurch bleibt der Cursor stets in der Mitte des Bildschirms. Der Zurück-Button beendet den Vorgang. Mit Bestätigen wird das Gebäude an der aktuellen Stelle gebaut, sofern kein Hindernis vorhanden ist und genug Geld für den Bau zur Verfügung steht.

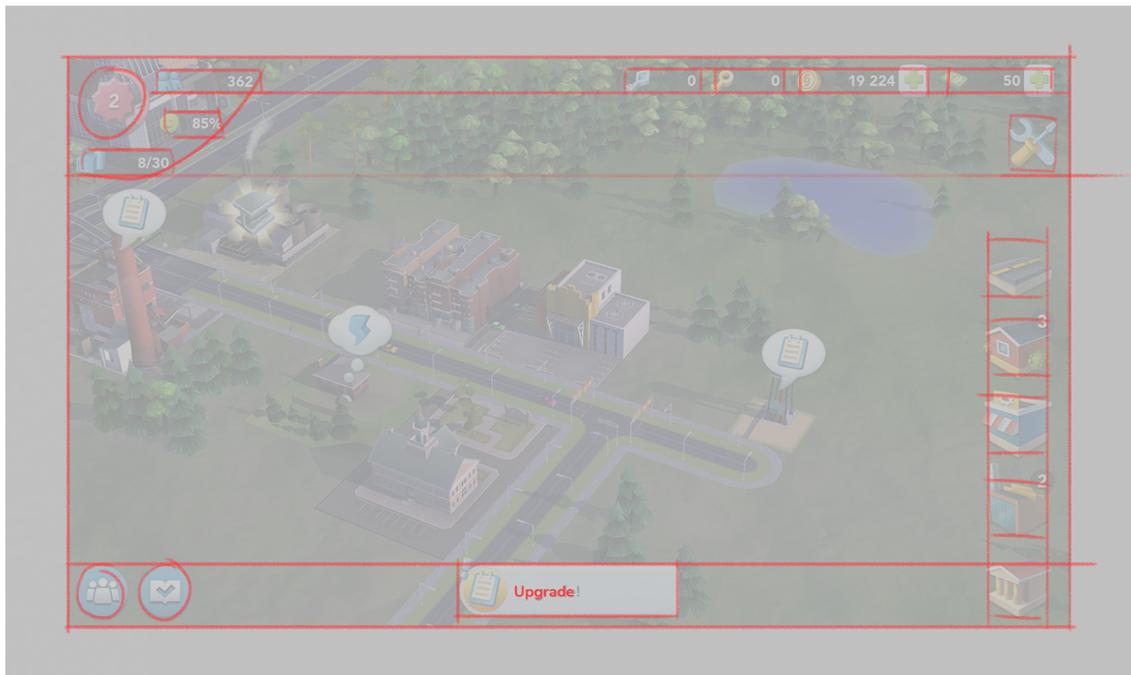


Abbildung 4.18: Wireframes Smart TV: Analyse des Mobile Screens von Sim City: Build It.

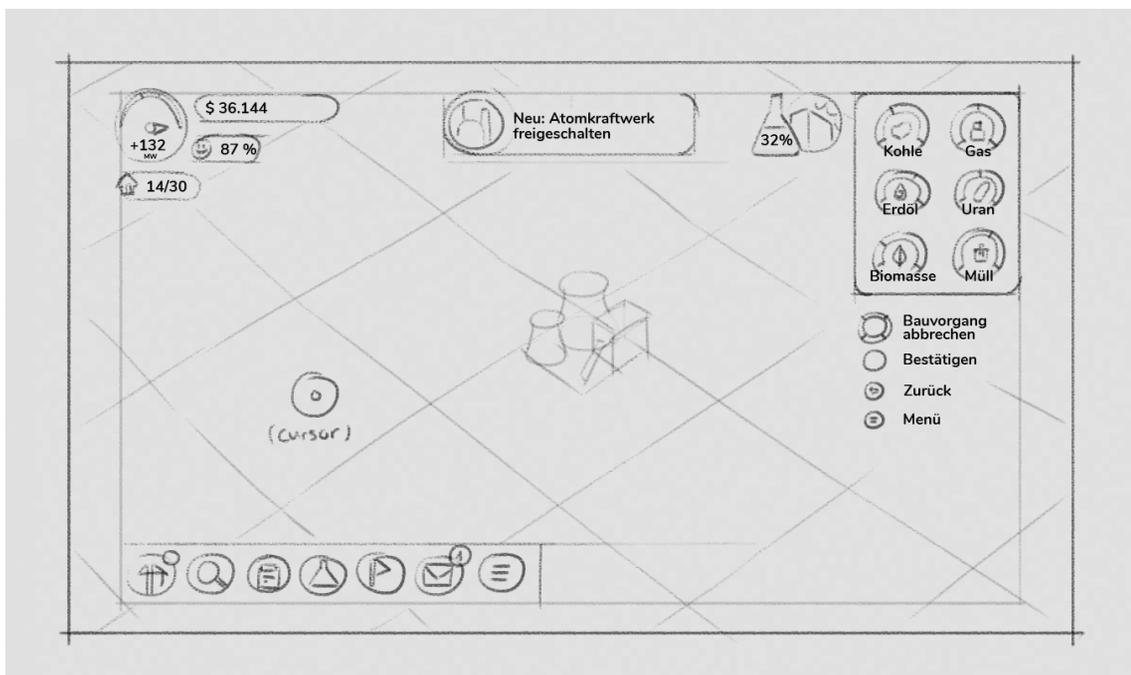


Abbildung 4.19: Wireframes Smart TV: Spielübersicht.

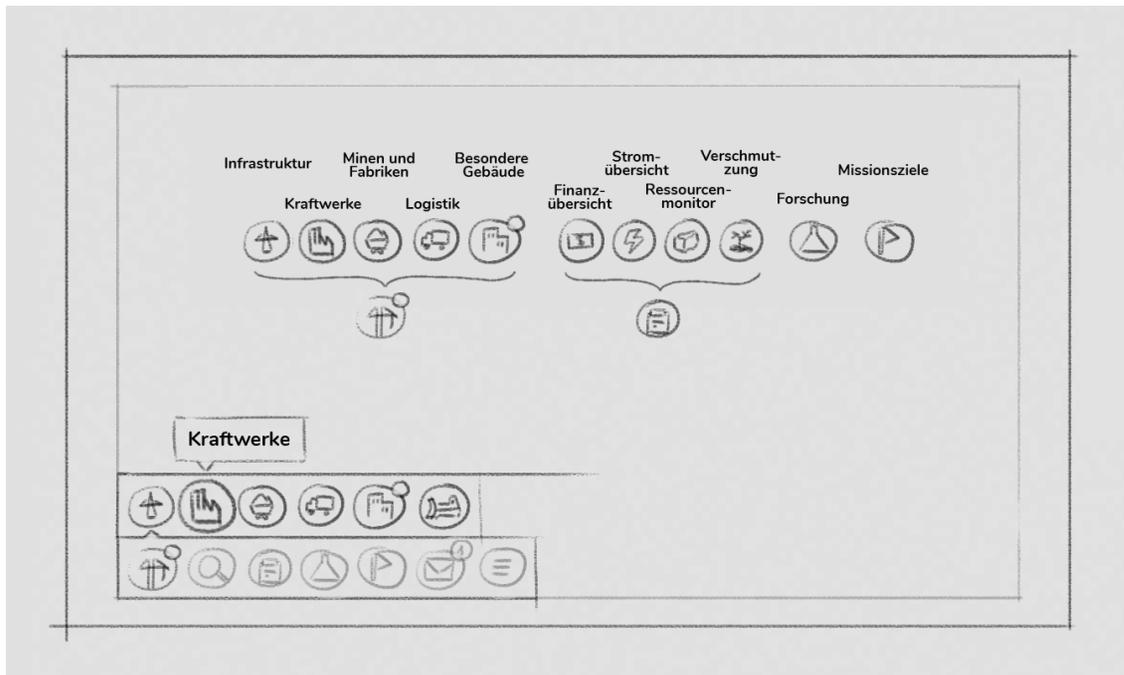


Abbildung 4.20: Wireframes Smart TV: Auflistung der Menüstruktur.

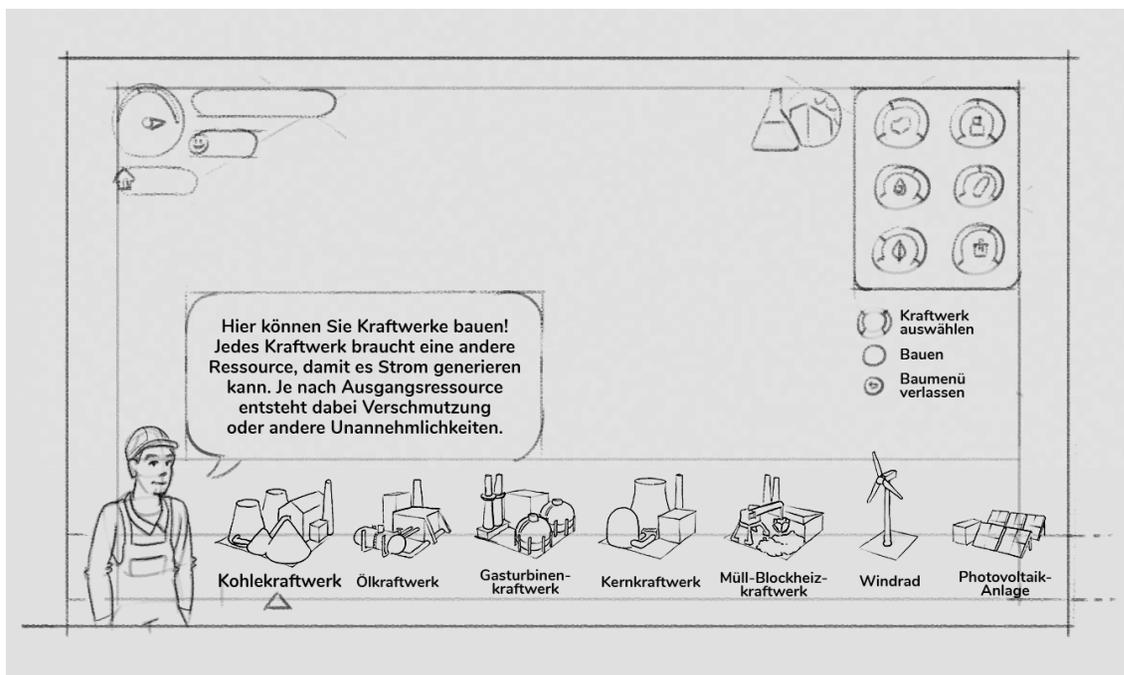


Abbildung 4.21: Wireframes Smart TV: Baumenü mit Tutorial.

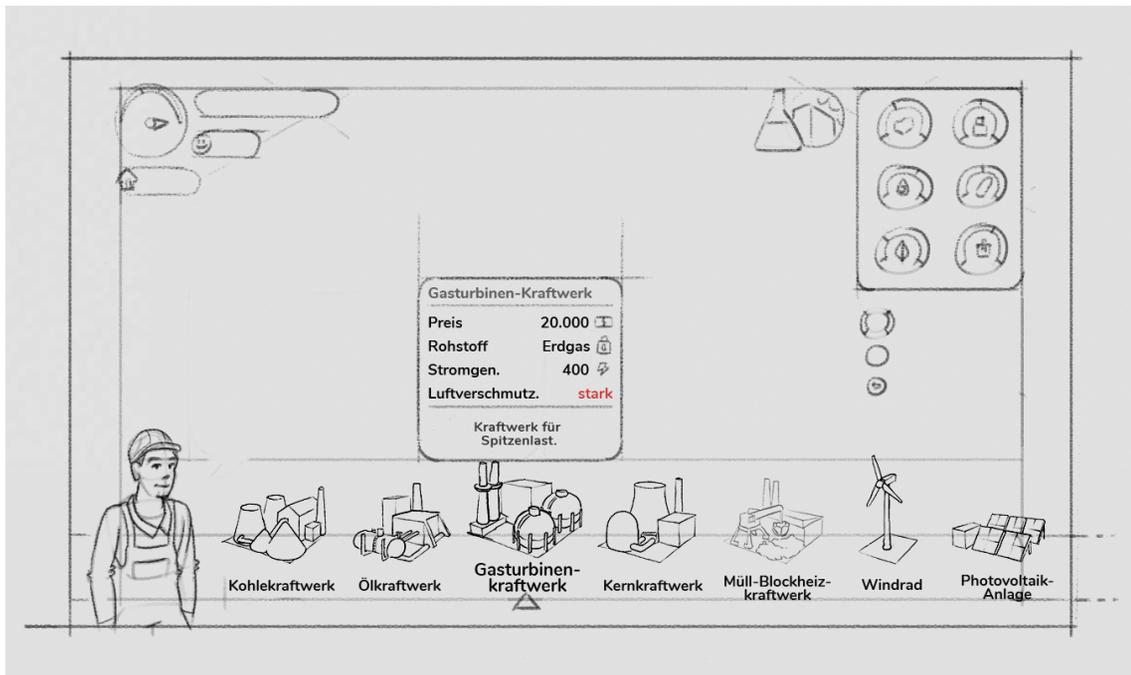


Abbildung 4.22: Wireframes Smart TV: Baumenü mit angewähltem Gebäude.

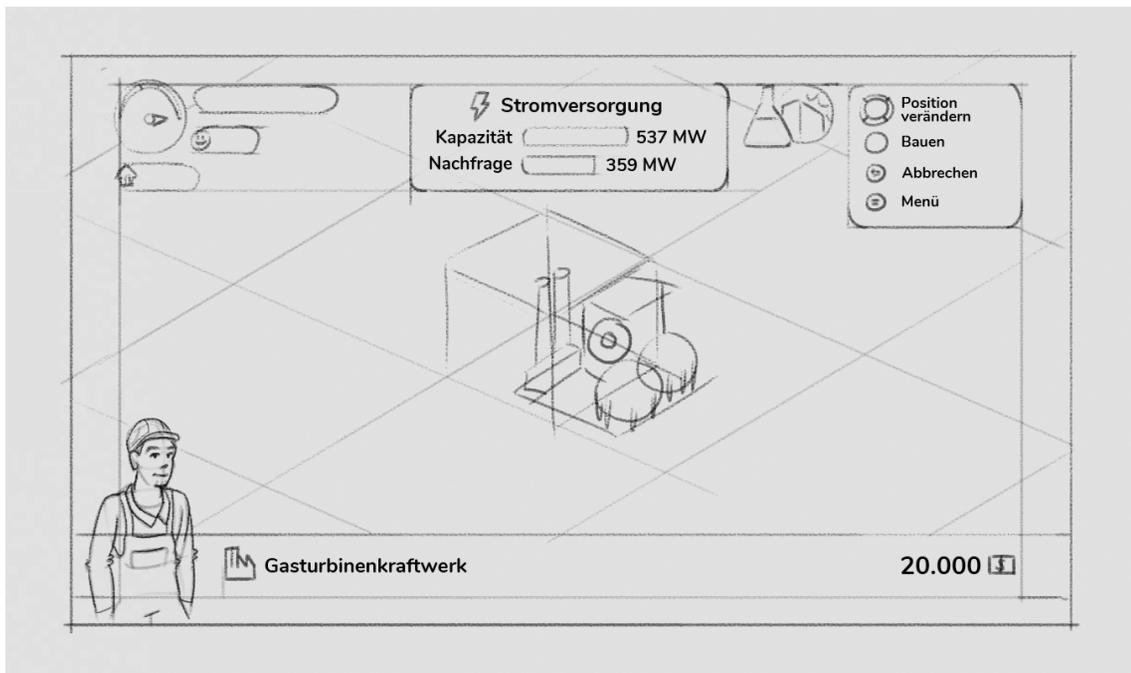


Abbildung 4.23: Wireframes Smart TV: Platzieren eines Gebäudes mit dem Cursor.

### 4.3.4 Wireframes PC

Die Abbildung 4.24 zeigt eine Analyse eines Screenshots aus Sim City (2013). Im Vergleich zu der mobilen Ansicht ist auffällig, dass in der PC-Ansicht viel mehr Screenspace dem Interface zur Verfügung steht. Der untersuchte Screenshot hat eine Auflösung von  $1.920px \times 1.080px$ , was bei deutschen Internetbenutzern der Durchschnitt ist [54].

Wenn man Abbildung 4.26 und 4.25 nun mit dem Screendesign der SmartTV-Ansicht vergleicht, erkennt man, dass das Interface ausschließlich am unteren Rand des Bildschirms angeordnet ist. Elemente wie die *Action Bar* oder der *State Indicator* sind alle als Leiste unten angefügt. Wenn der Spieler nun auf ein Icon in der *Action Bar* klickt, öffnet sich das Menüband darüber. Dadurch kann der Spieler jederzeit von einem Aktionsfeld zum anderen mit einem Klick wechseln.



Abbildung 4.24: Wireframes: Analyse des PC-Design von Sim City (2013).

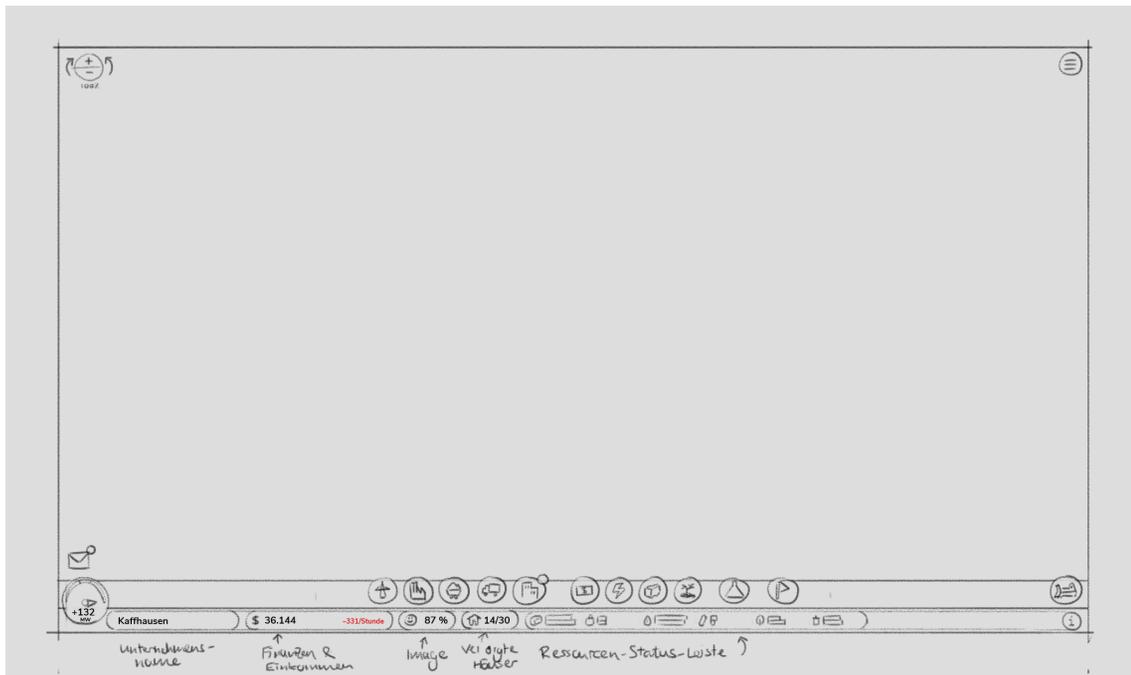


Abbildung 4.25: Wireframes: PC-Design mit Übersicht.

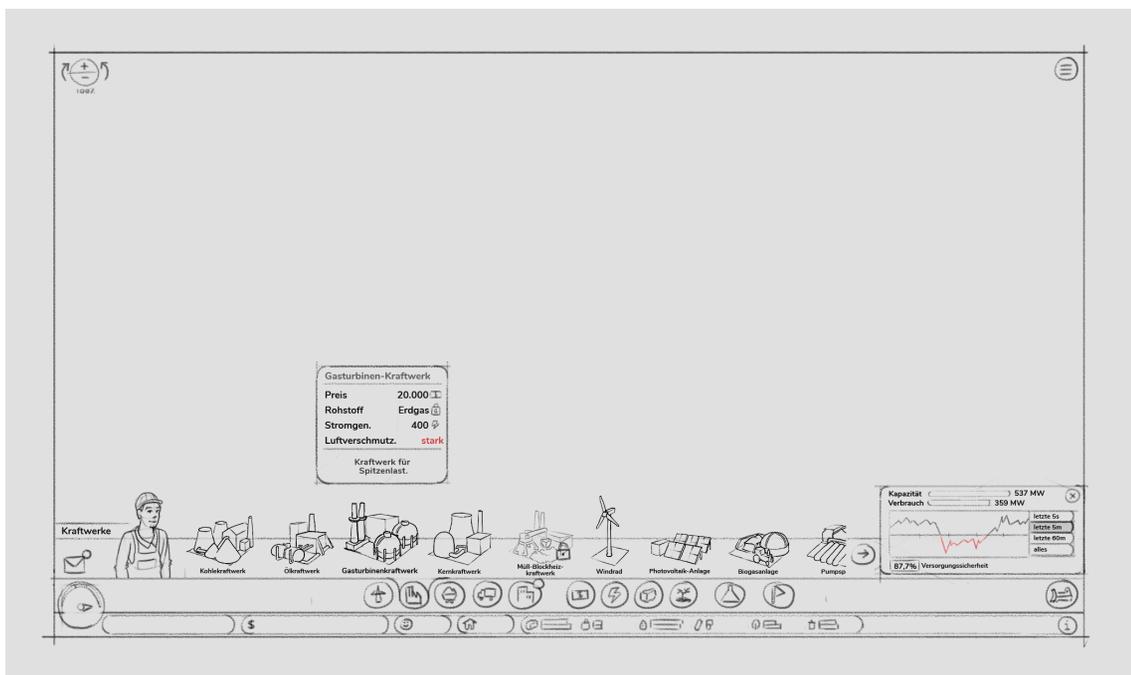


Abbildung 4.26: Wireframes: PC-Design mit geöffnetem Baumenü.

## 4.4 Ingame-Assets

Neben den Wireframes war es auch maßgeblich, grafische Assets für das Spiel zu erstellen. Denn bei diesem Spiel gibt es neben dem Screendesign auch eine navigierbare Spielwelt.

Die wichtigsten Assets für ein Kraftwerk-Simulations-Spiel waren dabei die Kraftwerke selbst. Daher wurden Porträts von dem einzelnen Gebäuden gezeichnet. Hier erfüllen die Porträts zwei Funktionen: 1) Sie zeigen im Baumenü, welche Gebäude der Spieler bauen kann. 2) Die Porträts sind die Basis für die Gestaltung der 3D-Assets – also die Repräsentation in der Spielwelt. Die dekorative Gestaltung der Spielwelt selbst war an dieser Stelle noch nicht relevant. Sie folgte erst in der Design-Phase.

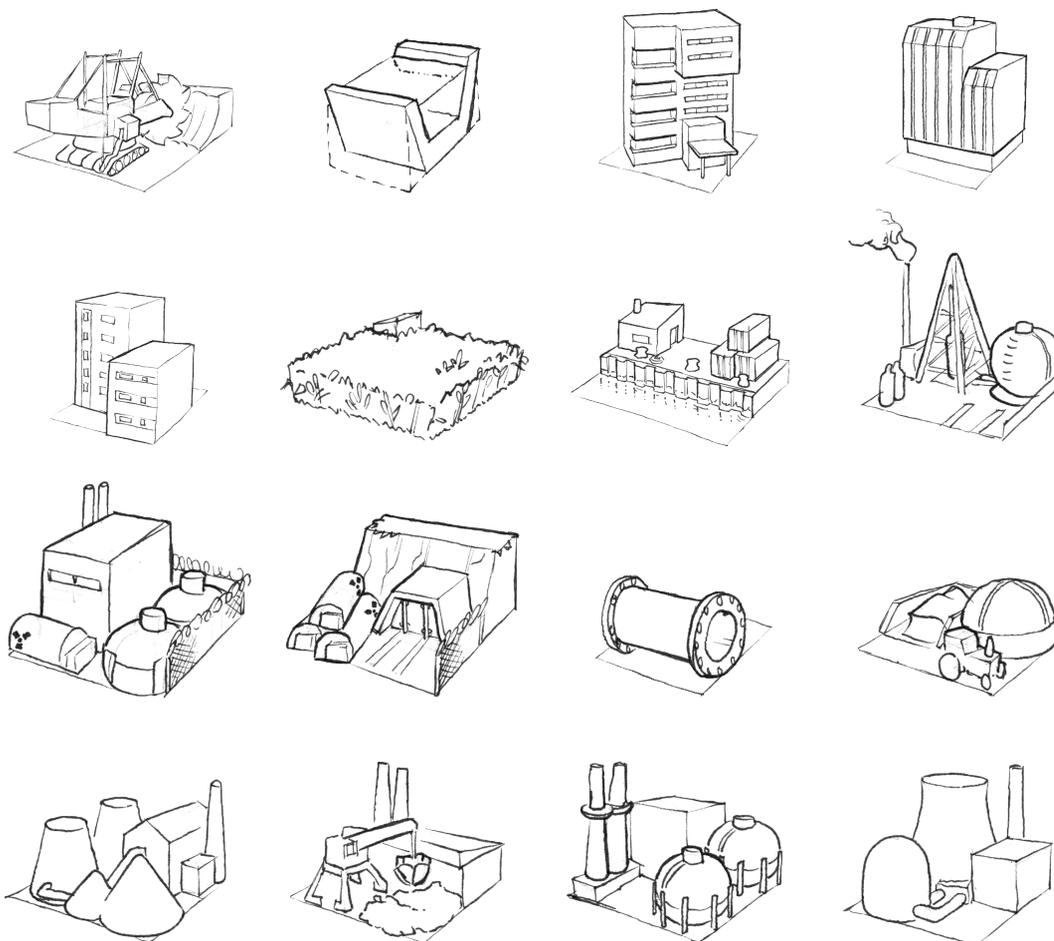


Abbildung 4.27: Skizzen: Porträts von baubaren Gebäuden, Teil 1.

Bei der Gestaltung selbst war wichtig, dass das Aussehen der Assets Hinweise auf die Funktionen des Gebäudes gibt. So hat Federoff die Heuristik ermittelt, dass Design Hinweise auf seine Funktion geben soll („Art should speak to its function“ [58, S. 42]). Beispielsweise wurde der Kohlebagger so gezeichnet, dass er beim Schürfen von Erdreich zu sehen ist. Also muss es sich um ein Gebäude handeln, mit dem Kohle abgebaut wird. Möglicherweise ergibt sich für den Spieler auch, dass die gewonnene Kohle auch weitertransportiert werden muss. Dafür sprechen z.B. die Schienen – und so weiter.

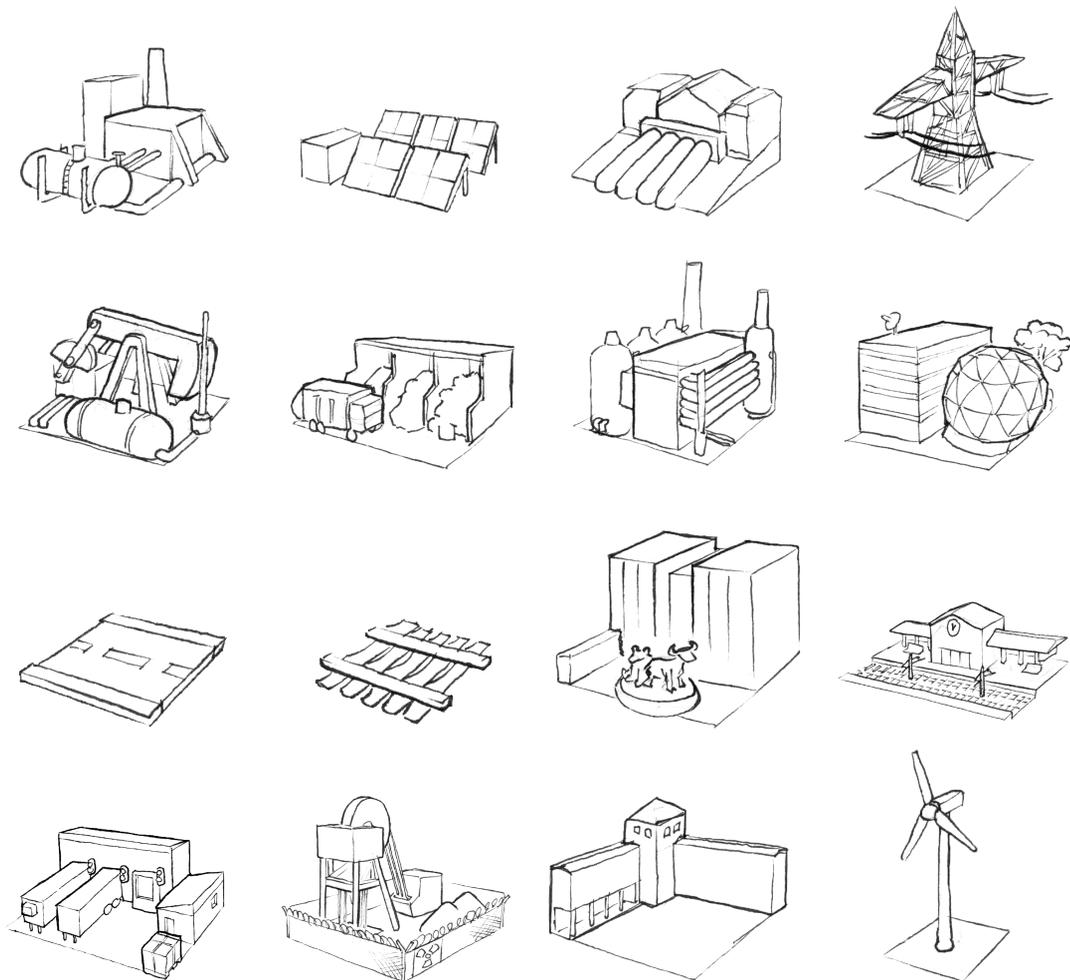


Abbildung 4.28: Skizzen: Porträts von baubaren Gebäuden, Teil 2.

## 4.5 Klick-Prototyp

Ferrara schreibt [1, S. 95], um ein Gefühl dafür zu bekommen, wie sich das Spiel überhaupt spielt, muss die Komplexität des aktiven Spiels eingeführt werden. An diesem Punkt sei es Zeit, mit dem Entwickeln eines elektronischen Spiele-Prototypen anzufangen.

Um das Spiel auf die Usability hin testen zu können, wurde ein Klickprototyp mit dem Prototyping-Tool *Azure* erstellt. Als Basis dienten die zuvor erstellten Wireframes. Ziel des Prototypen war es herauszufinden, wie sich das fertige Spiel verhalten würde. Gleichzeitig diente er auch als Basis für den später vollständig programmierten Spiele-Prototypen. Dabei wurde sich (gemäß *Mobile First*) auf die mobile Gestaltung beschränkt. Eine Anpassung für Smart TV oder PC kann dann leicht aus dem Mobil-Prototypen entwickelt werden.

Für das Erstellen eines Spiele-Prototypen gelten dabei besondere Regeln [1, S. 95–98]:

- **Start as small as you can**

Mit dem Prototypen wurden zuerst die fundamentalen Features wie die Kamera-Bewegung und Platzieren von Gebäuden umgesetzt. So war beispielsweise Zoom nicht eingebaut und es war nur das gleiche Gebäude platzierbar.

- **Work from wireframes**

Die Wireframes haben sich als sehr wertvoll bei der Umsetzung erwiesen. Mit ihnen wurde erarbeitet, welche Funktionen und Gestaltungselemente für das fertige Spiel gebraucht werden.

- **Don't Overdesign**

Die visuelle Gestaltung war für den ersten Prototypen noch nicht wichtig. Ziel war es, die grundlegende Gameplay-Funktionalität (siehe Kapitel 4.2) der Use Cases abzubilden. Diese Gameplay-Features sind auch umgesetzt worden.

- **Squeeze as Much Use Out of a Prototype as You Can**

Der Klick-Prototyp wurde nach dem Fertigstellen mit dem ersten Nutzertest überprüft (siehe Kapitel 7.1). Gleichzeitig diente der Prototyp auch als erster Ansatz für die UI-Gestaltung.

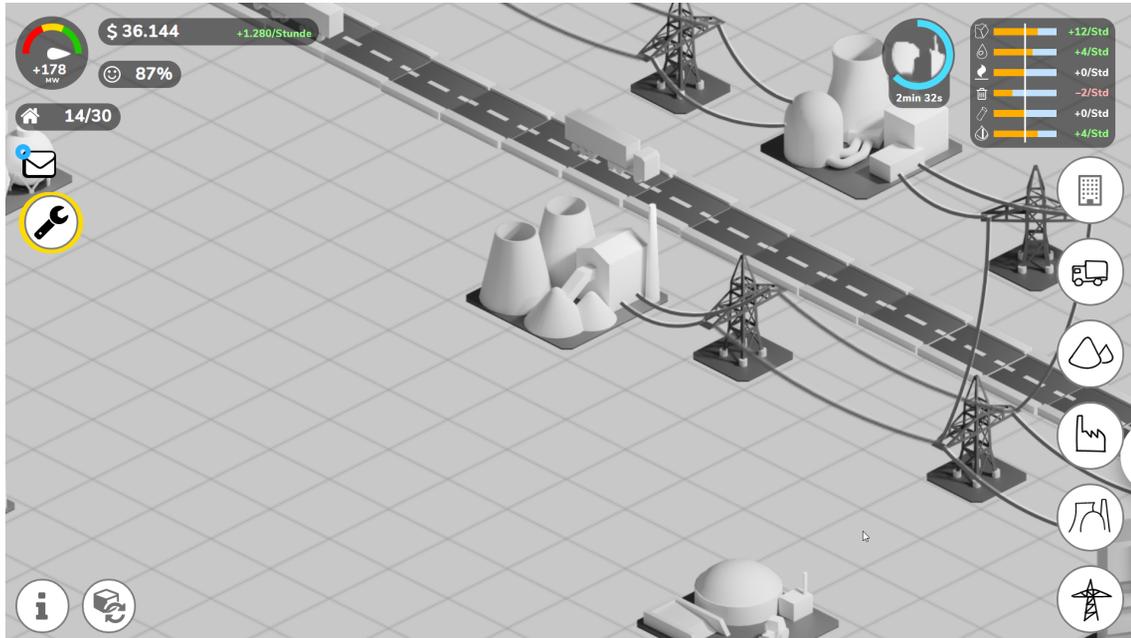


Abbildung 4.29: Axure-Prototyp: Übersicht über Spielkarte.

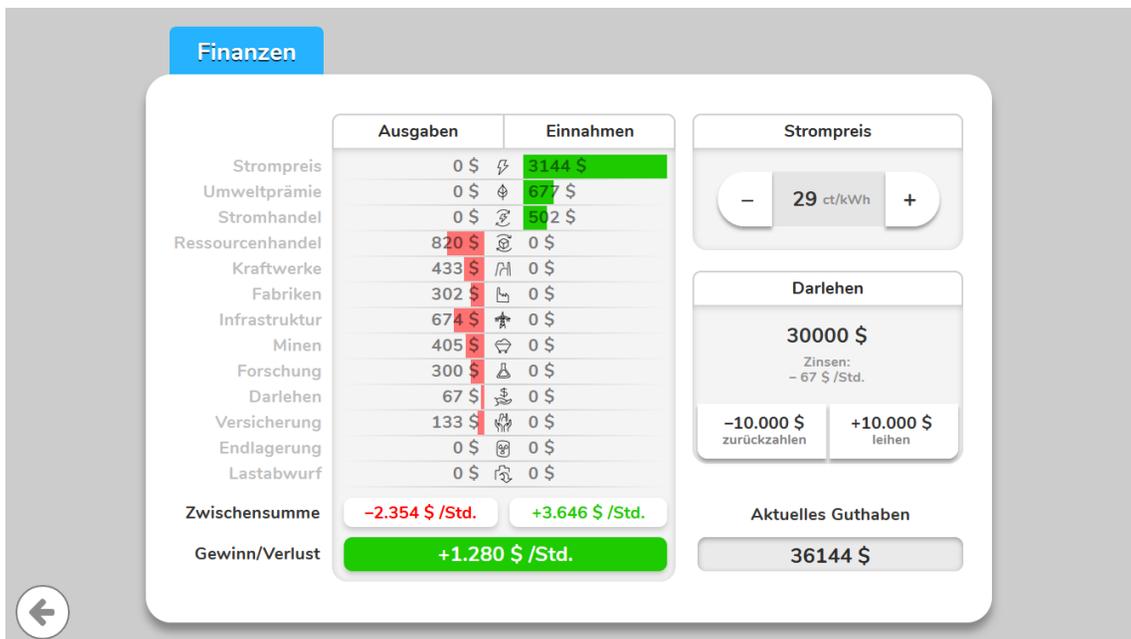


Abbildung 4.30: Axure-Prototyp: Finanzübersicht.



Abbildung 4.31: Axure-Prototyp: Modal zum Abriss eines Gebäudes.

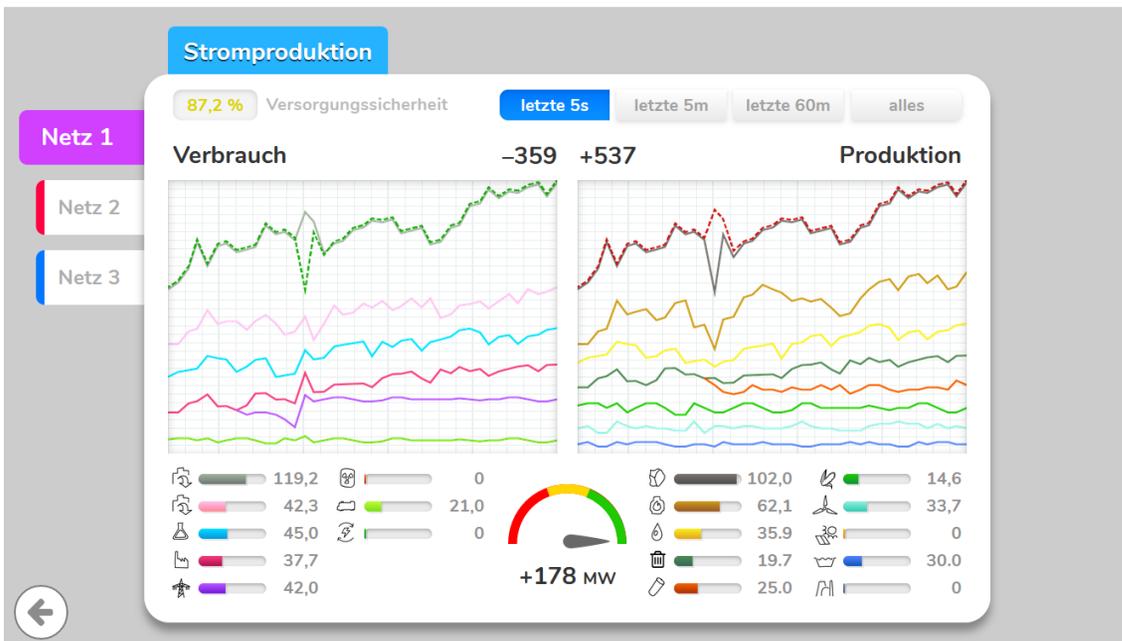


Abbildung 4.32: Axure-Prototyp: Übersicht über Stromnetz.

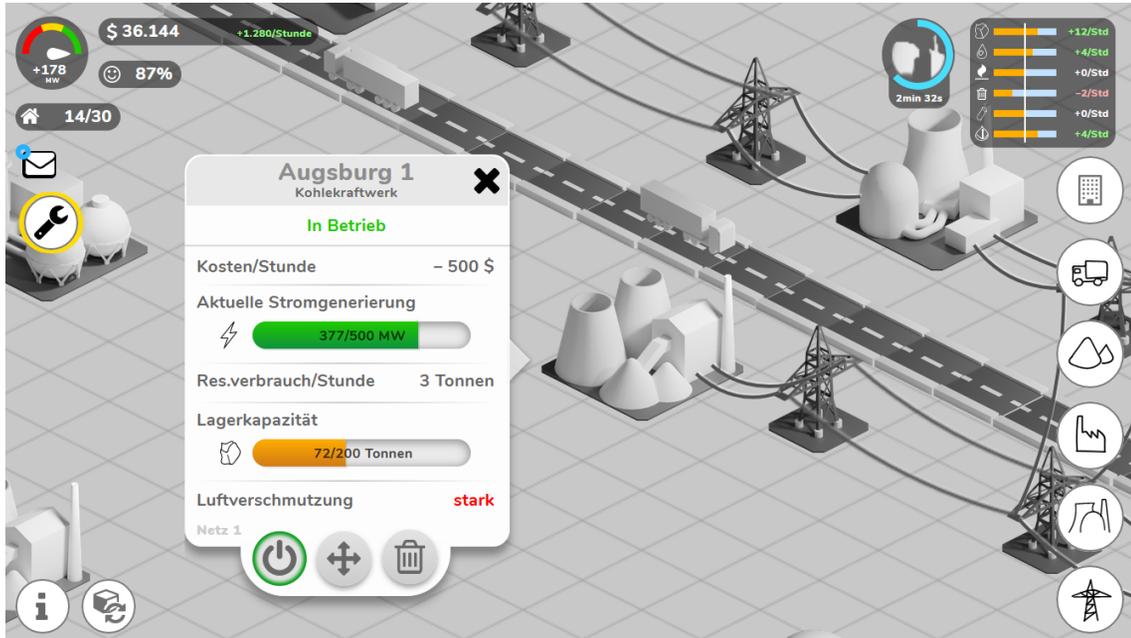


Abbildung 4.33: Axure-Prototyp: Infenster eines Kraftwerkes geöffnet.

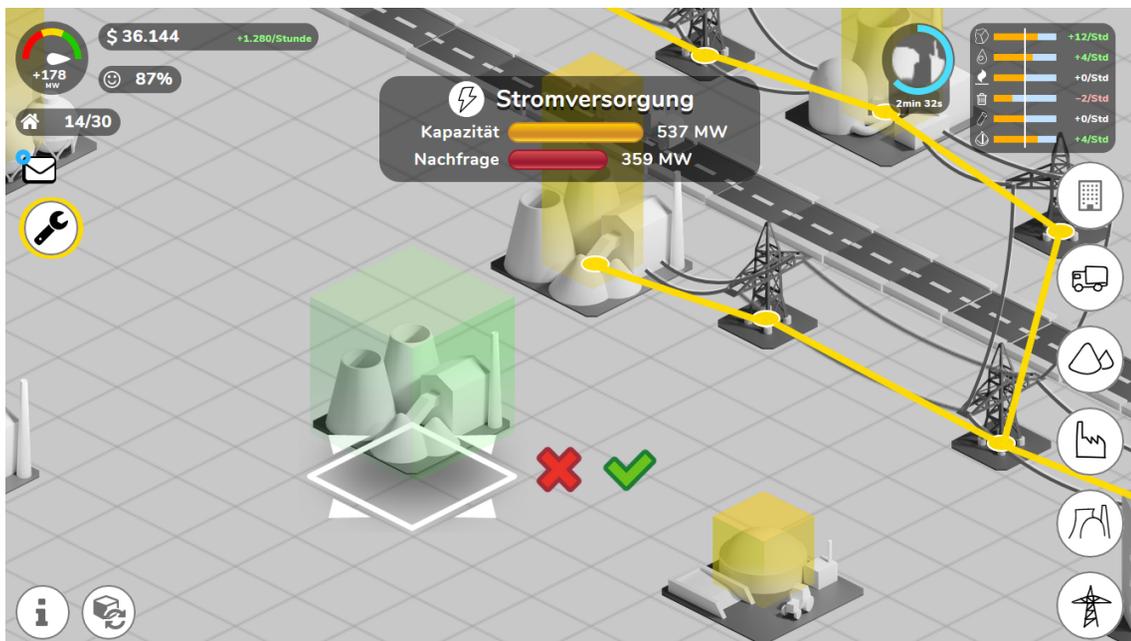


Abbildung 4.34: Axure-Prototyp: Platzieren eines Kraftwerkes und Graph des Stromnetz.



# Kapitel 5

## Design

Auf Grundlage der Analysephase und des Konzeptes wurde das Interfaces gestaltet. Dafür wurde das Design des Spiel-Interfaces und der Spiel-Welt festgelegt. Der Fokus liegt dabei auf der Gestaltung für die drei Endgeräte Mobile, Smart TV und Desktop. Die grundlegende Gestaltung ist dabei sehr ähnlich.

### 5.1 Moodboard

Um eine Idee davon zu bekommen, wie das Interface aussehen soll, wurde ein Moodboard erstellt (siehe Abbildung 5.1). Üblicherweise können so aus einem Moodboard die Gestaltungselemente abgeleitet werden. Dazu zählen Farben, Formen, Schriften und Bilder [29, S. 214]. Vor allem war interessant, wie das UI-Design in anderen Aufbau-Simulations-Spielen umgesetzt wurde.

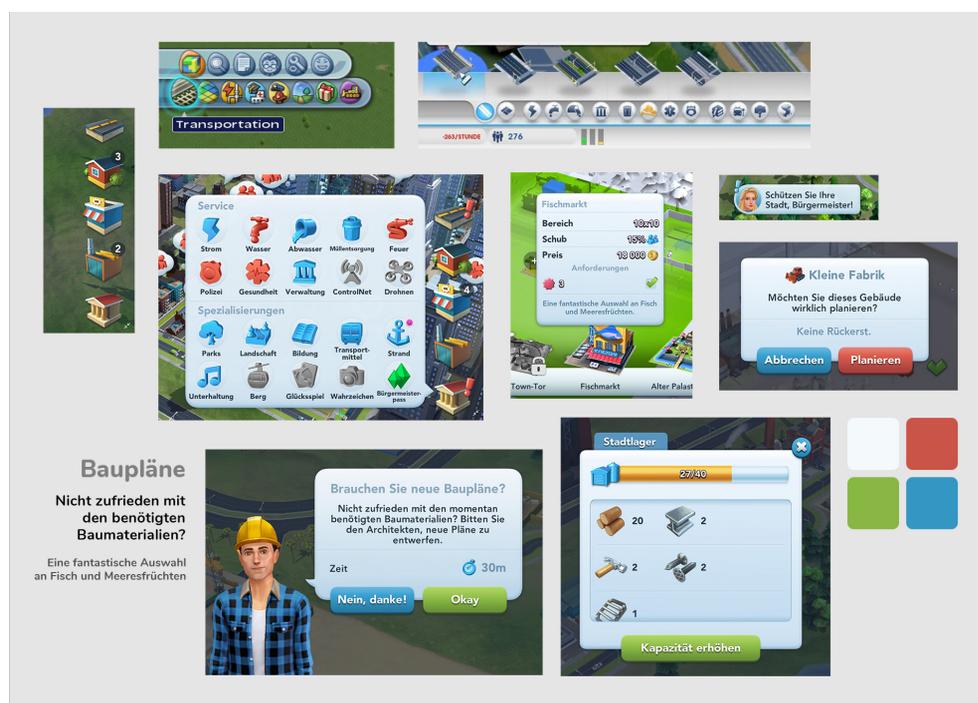


Abbildung 5.1: Moodboard: Gestaltung von Stadt-Aufbau-Simulatoren [34], [59], [60].

## 5.2 Farben

Farben sind sehr wichtig bei der Gestaltung. Wie Moser hierzu schreibt, wecken Farben Emotionen aufgrund von Assoziationen mit Eindrücken, persönlichen Erfahrungen, kulturellen Überlieferungen oder angelernten Bedeutungen [29, S. 196]. Je nach Kulturkreis haben sie auch eine andere Bedeutung [61, S. 192].

Insgesamt wurden in dieser Arbeit vier verschiedene Farbkategorien festgelegt:

- Die **Primärfarben** beschreiben die Gestaltung des Interfaces. Für das UI kommen relativ wenige Farben vor. Interaktive Elemente haben einen weißen Hintergrund. Die statischen Statusanzeigen dagegen haben einen halbtransparenten schwarzen Hintergrund.
- Die **Auszeichnungsfarben** werden bei Statusanzeigen verwendet. Will der Spieler beispielsweise ein Gebäude platzieren, dann wird eine halbtransparente Box um das schwebende Gebäude gerendert. Wenn die Box platzierbar ist, hat sie ein intensives Grün – wenn nicht, ein markantes Rot.
- **Schattierungen** sind wichtig, um das Interface gestalten zu können. Schrift ist nicht reines Schwarz sondern hat Schattierungen von Grau.
- Zur Unterscheidung der verschiedenen Informationskategorien im HUD werden die **Kategoriefarben** verwendet. Die Farben sind von der Leuchtkraft eher gedämpft, da sie nicht zu stark herausstechen sollen. Außerdem sind die Farben im HUD für den grünen Hintergrund angepasst: Das Gelb hat zum Beispiel auf weißem Hintergrund einen starken Grünstich – im Spiel liegt es jedoch auf der meist grünen Spielwelt. Dadurch ist es optisch ausgeglichen (siehe Abbildung 5.2).

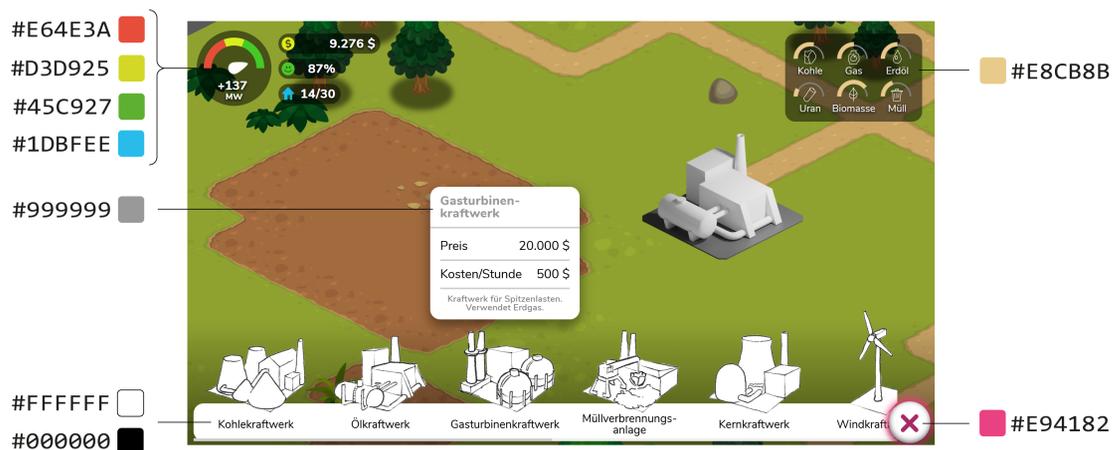


Abbildung 5.2: Farben: Verwendete Farben im Spiel.

### Primärfarben



#E94182



#FFFFFF

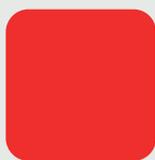


#CCCCCC



#000000

### Auszeichnungsfarben



#FF0000



#1ECA00



#317FFF

### Schattierungen



#FFFFFF



#E5E5E5



#CCCCCC



#B2B2B2



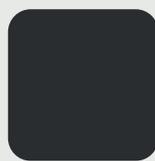
#999999



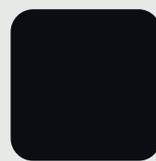
#666666



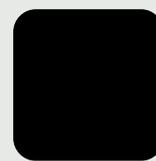
#4C4C4C



#333333

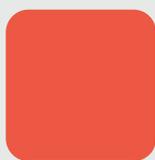


#191919



#000000

### Kategoriefarben



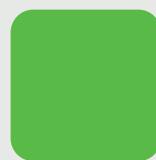
#E64E3A



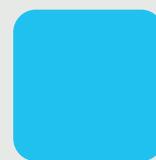
#E0902D



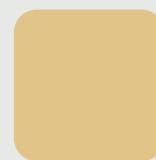
#D3D925



#45C927



#1DBFEE



#E8CB8B

### 5.3 Typografie

Als Schriftsatz für den Spiele-Prototypen wurde *Nunito* ausgewählt. Nunito ist eine serifenlose Schriftart, die von Vernon Adams entwickelt wurde. Sie wurde später von Jacques Le Bailly um verschiedene Schriftschnitte erweitert [62]. Das Besondere an der Schrift ist, dass sie relativ breit ist und die Ecken abgerundet sind. Dadurch wirkt die Schrift freundlich, offen und etwas verspielt.

Die Schrift wurde deshalb ausgewählt, da sie auch bei kleiner Darstellung gut lesbar ist. Serien-Schriften sind bei Bildschirmdarstellung eher ungeeignet [63, S. 180].

Für den Prototypen wurde überwiegend der Schriftschnitt Semi-Bold verwendet (siehe Abbildung 5.3). Überschriften und HUD-Elemente wurden Extra-Bold, Nebeninformationen mit Regular dargestellt. Text, der im HUD verwendet wurde, wurde in Weiß auf einem halbtransparenten dunklen Hintergrund verwendet.

Damit das Interface möglichst wenig Platz einnimmt, wurde eine relativ kleine Schriftgröße ausgewählt (20px). Vor allem im Mobilen Bereich und für Smart TVs ist es aber wichtig, dass die Schriftgröße noch gut lesbar ist. Um die Lesbarkeit zu verbessern, wurde deswegen die Laufweite erhöht. Der Text in den halbtransparenten Statusanzeigen hat einen leichten Textschatten für einen besseren Kontrast.

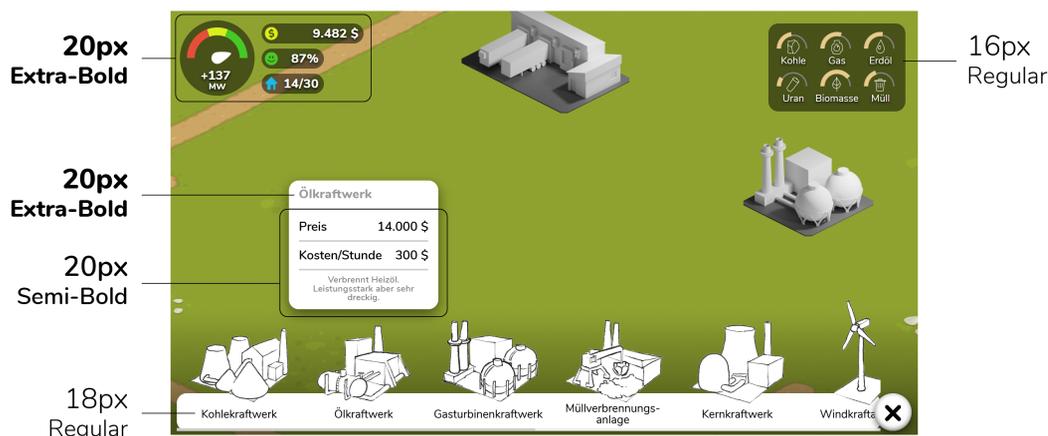


Abbildung 5.3: Typografie: Verwendete Schriftgrößen im Spiel.

# Nunito

---

Regular  
Semi Bold  
**Extra Bold**

ABCDEFGHIJKLMN  
OPQRSTUVWXYZ

abcdefghijklmn  
opqrstuvwxyz

1234567890  
( ) & % ? ! . , : ; / ~

---

Franz fährt im völlig verwahr-  
losten Taxi quer durch Bayern.

Franz fährt im völlig verwahr-  
losten Taxi quer durch Bayern.

## 5.4 Screendesign

Zunächst wurde ein grafisches Mock-up erstellt (siehe Abbildung 5.4). Der Fokus lag dabei darauf, prototypisch einen Screen der Wireframes bis ins Detail umzusetzen. Grafische Mockups werden verwendet, um beurteilen zu können, wie das fertige Design aussehen wird [29, S. 216].



Abbildung 5.4: Screendesign für Smart TV-Übersicht.

### Cursor bei Smart TV

Mobile und PC ermöglichen eine relativ genaue Eingabe mit Touchscreen, Maus und Tastatur. Dagegen ist die Steuerung beim Smart TV sehr eingeschränkt: Man kann sich nur in vier Richtungen (nicht diagonal) bewegen und häufiges Klicken mit dem *Fire TV*-Stick wird schnell anstrengend.



Abbildung 5.5: Grafische Zustände des Smart TV-Cursors.

## Kontraste und Farbverteilung

Um sicherzustellen, dass sich das Interface stark genug von der Spielwelt abhebt, wurde ein *Color Board* erstellt. Es dient dazu, alle Farben grob zu verteilen (siehe Abbildung 5.6). Die Spielwelt soll lebendig und farbenfroh wirken. Dagegen soll das User Interface kühl und sachlich sein.

Neben den Farben waren für eine bessere Accessibility vor allem gute Kontraste wichtig (siehe Abbildung 5.7): Interagierbare UI-Elemente sind hell gestaltet, die Spielwelt befindet sich im mittleren bis dunklen Bereich und statische UI-Anzeigen haben weiße Schrift auf dunklem Grund.



Abbildung 5.6: Color Board für die Farbgestaltung des Spiele-Prototypen.



Abbildung 5.7: Kontraste im Screendesign.

## 5.5 Icondesign

In dem Spiele-Interface wurden möglichst viele Icons verwendet, da sie platzsparend sind und leicht wiedererkennbar. Thorsten Stapelkamp schreibt hierzu [61, S. 150], dass eine Beschreibung in Form eines Wortes oder Textes wesentlich mehr Platz in Anspruch nehmen würde und auch nicht so schnell wahr genommen werden könnte wie ein Icon. Laut Preim sollen sie folgende Bedingungen erfüllen [64, S. 316]:

1. Einfach und gut erkennbar
2. Verständlich
3. Einprägsam
4. Guter Kontrast zu Hintergrund
5. Leicht unterscheidbar
6. Im aktiven Zustand von anderen Icons unterscheidbar

Bei der Gestaltung der Icons wurde ein einheitliches Gesamtbild angestrebt. Das heißt, Strichstärke, Radius der Ecken und Proportionen müssen ungefähr ähnlich sein. Außerdem sollen sie gut lesbar sein und freundlich wirken.

Von den Icons wurden zwei Versionen erstellt. Die erste Version der Icons (siehe Abbildung 5.9) hatte das Ziel, möglichst viele Icons mit unterschiedlichen Formen zu erstellen. In der zweiten Generation (siehe Abbildung 5.8) folgte dann der Feinschliff für das Screendesign.

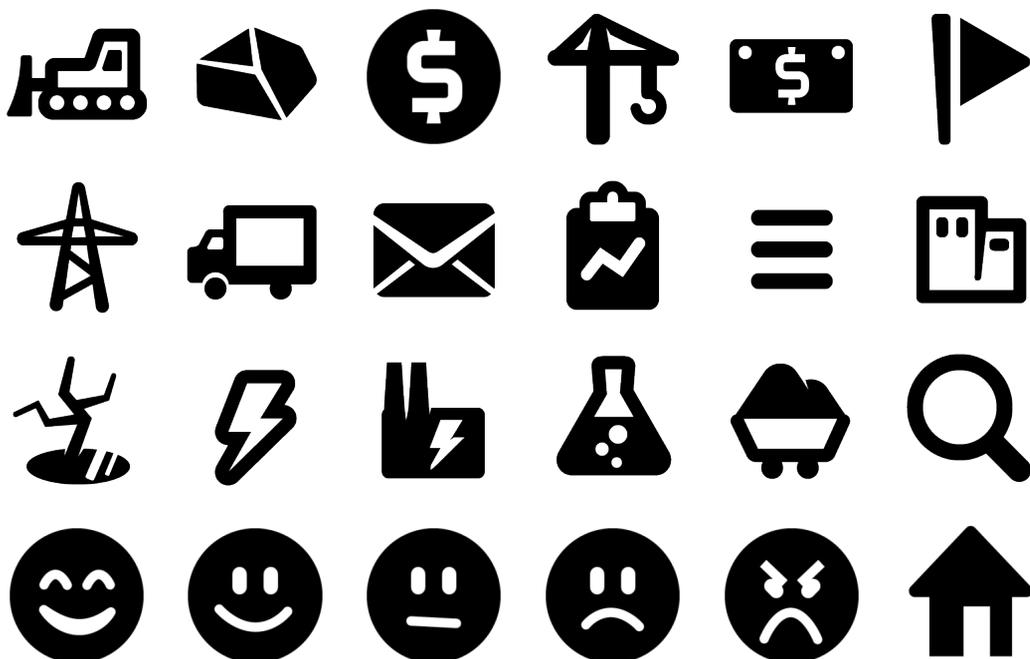


Abbildung 5.8: Icons der zweiten Generation für das Interfacedesign.

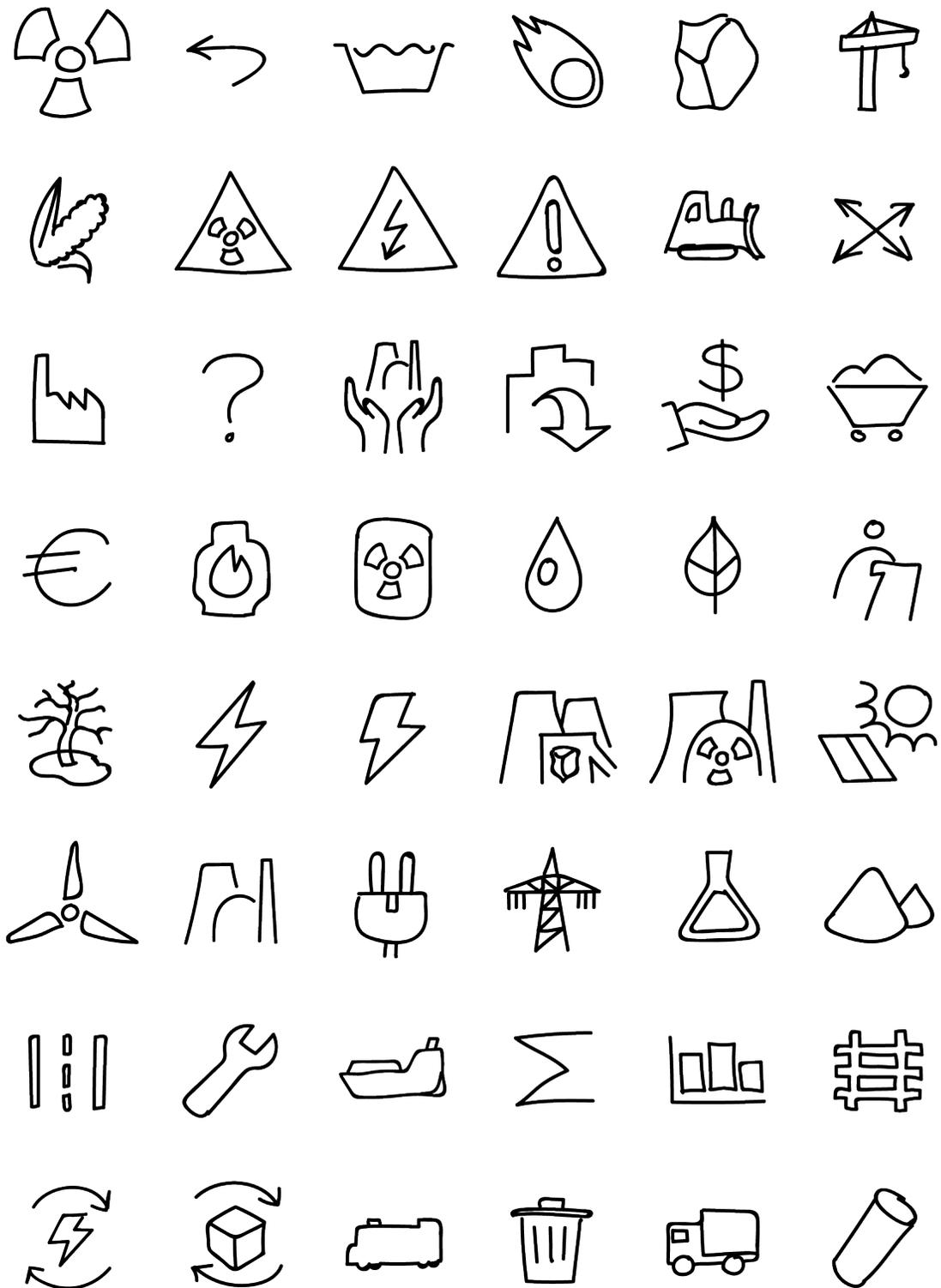


Abbildung 5.9: Icons der ersten Generation für den Klick-Prototypen.

## 5.6 Ingame-Design

Neben der Gestaltung des User Interfaces trägt bei einem Videospiel die Gestaltung der Spielwelt maßgeblich zur Erfahrung bei.

In einem Test von Caroux und Isbister [65] wurde untersucht, wie sich das HUD eines Videospieles auf die User Experience auswirkt. Dabei wurden unter anderem die Augen-Fixationen auf dem Bildschirm gemessen. Das Ergebnis war, dass bei dem untersuchten RTS-Spiel<sup>1</sup> 24,1 % der Augen-Fixationen auf das UI waren. Die restliche Aufmerksamkeit wurde der Spielwelt gewidmet.

Passend zum Moodboard wurde mit dem Tileset ein Stil gewählt, der eher verspielt und freundlich wirkt. Die Wahl fiel dabei auf ein Asset aus dem *Humble RPG Game Dev Bundle* [66], [67].

Die Gestaltung der Spielwelt selbst besteht aus drei Ebenen:

- Die Tiles, aus denen die Spielwelt aufgebaut ist.
- Dekorationselemente („Doodads“) in der Spielwelt wie Bäume oder Blumen.
- Spielelemente wie Kraftwerke oder Städte (siehe 4.4).

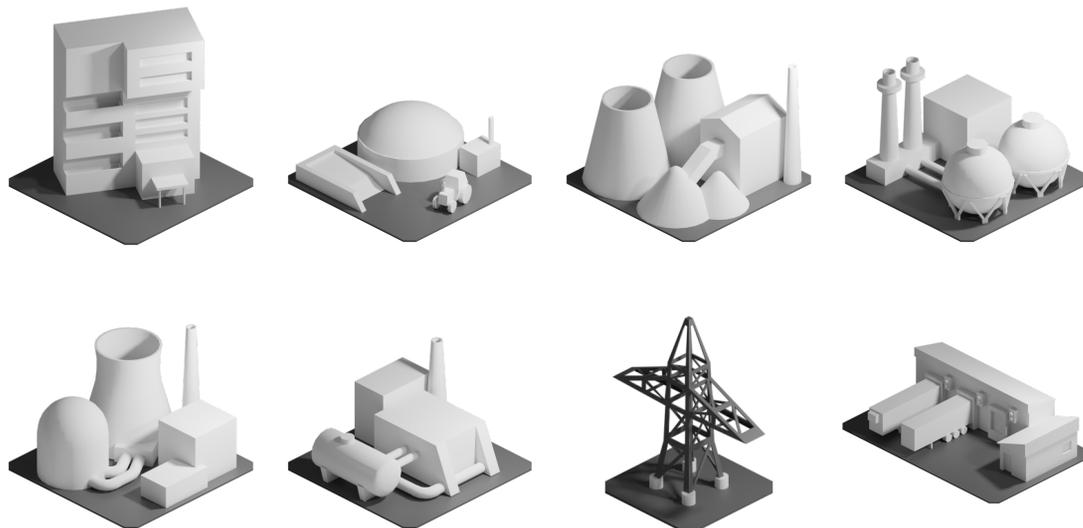
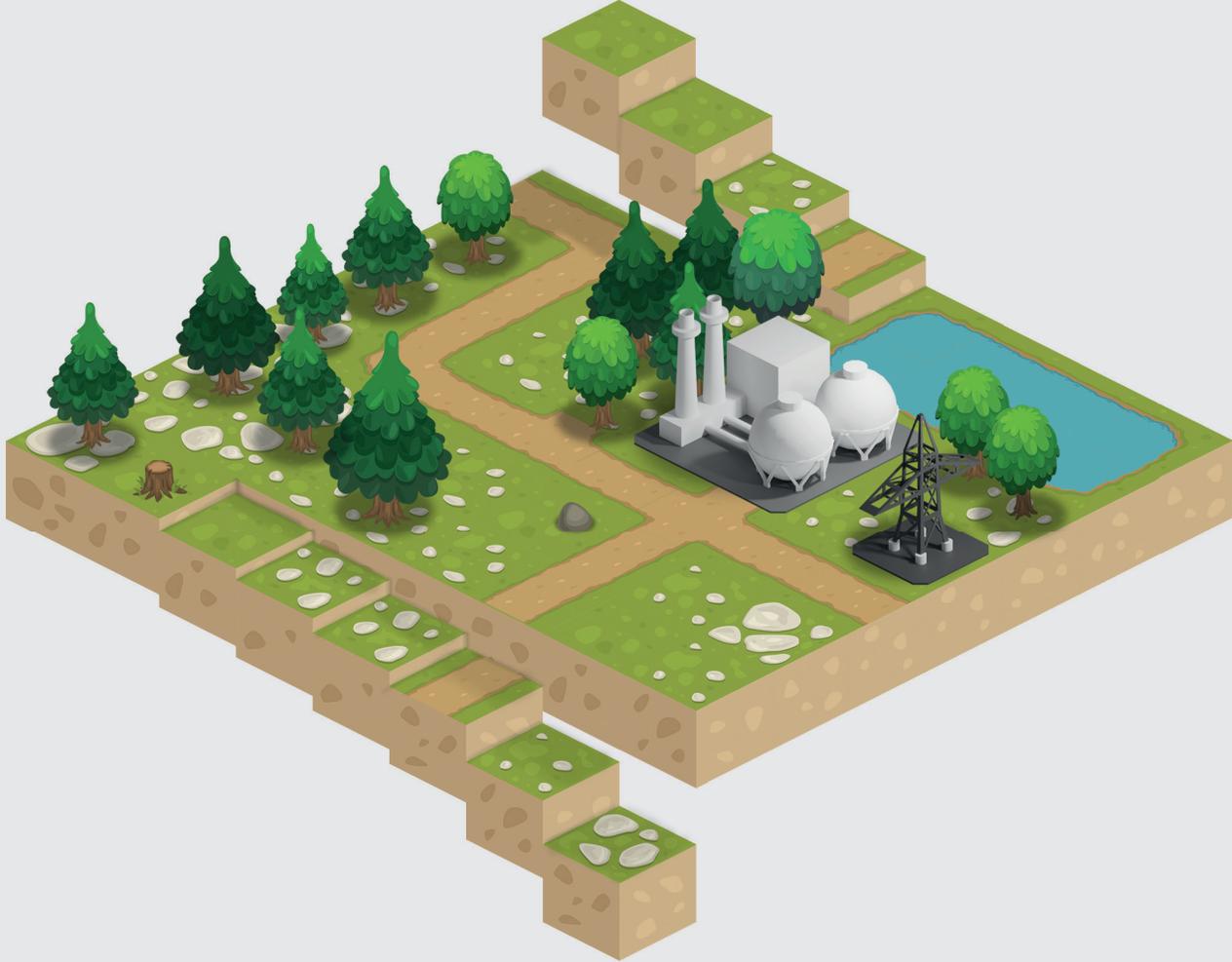


Abbildung 5.10: Renderings von ausgewählten Gebäuden.

---

<sup>1</sup> RTS, oder auch *Real Time Strategy*-Spiele, sind Strategiespiele, die ein ähnlich komplexes UI wie Aufbau-Simulations-Spiele haben. Meistens ist in RTS das Ziel, eine Basis aufzubauen und den Gegner mit den eigenen Einheiten zu besiegen.





---

## Kapitel 6

# Implementierung

Nach dem Konzept und Design wurde der Spiele-Prototyp programmiert. Als Basis diente hierfür der Klick-Prototyp (siehe Kapitel 4.5) und das Feedback der Probanden beim ersten Nutzertest (siehe Kapitel 7.1). Es wurden zwei verschiedene Ansätze durchgeführt: 1) Programmierung mit der Game Engine *Unity*. 2) Umsetzen mithilfe der HTML5-Game Engine *Phaser* und dem Frontend Framework *Vue.js*.

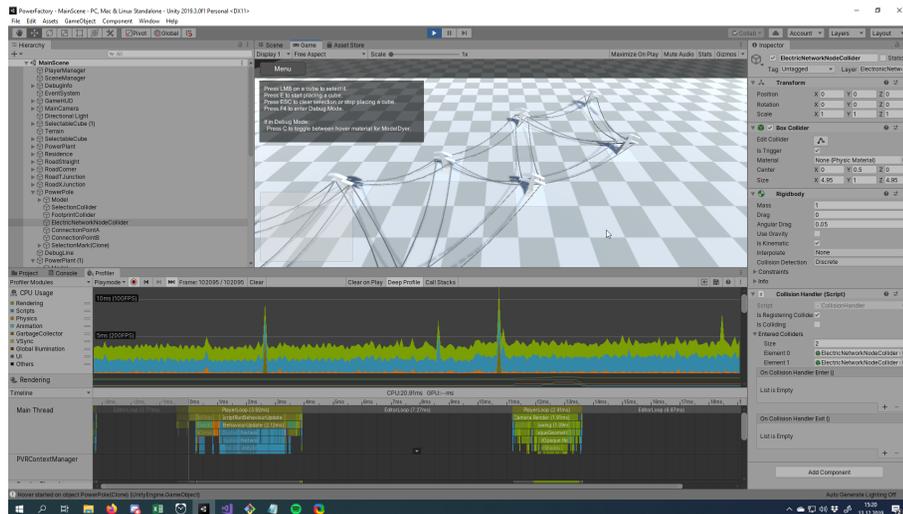


Abbildung 6.1: Unity: Die Framerate eines Spiels wird mit dem Profiler überprüft.

## 6.1 Test mit Unity-Prototyp

### 6.1.1 Unity

Unity (oder Unity3D) ist die weltweit am meisten genutzte „Real-time 3D (RT3D) development platform“. Unity-Anwendungen laufen auf 3 Milliarden Geräten in aller Welt und 50 % aller Mobile Games basieren auf der Game Engine [68]. Der Vorteil an Unity, dass Spiele für beliebige Endgeräte kompiliert werden können. In diesem Fall: Mobile (Android), PC (Windows) und Smart TV (Android).

Aber Unity wird nicht nur für Videospiele genutzt, sondern für verschiedenste Echtzeit-Anwendungen. Darunter fallen auch *Serious Games* oder Infotainment-Systeme von Fahrzeugen [69], [70]. Game Engines haben üblicherweise folgende Features, um die Entwicklung von Spielen (oder anderen Echtzeit-Anwendungen) zu erleichtern [69, S. 662]:

**Scripting** Code zur Kontrolle von GameObjects oder basierend auf Events. Skripte werden in Unity mit C# geschrieben.

**Rendering** Darstellung der 3D-Szene und von visuellen Effekte, wie Partikel-Systeme oder Post-Processing.

**Physics** Objekte reagieren angemessen bei einer Kollision mit anderen Objekten oder wenn Kräfte auf sie einwirken.

**Audio** Wiedergabe von Audio-Quellen. Auch *Spatial Sound* zählt dazu, sowie Anwenden von Audio-Filtern.

**Networking** Erlaubt es, mehreren Personen am selben Spiel teilzunehmen. Dazu werden über ein Netzwerk Daten ausgetauscht.

**Level-Editor** Auch *World Editor* genannt, ist ein Werkzeug, das bei der Gestaltung von 2D- oder 3D-Level unterstützt (siehe Abbildung 6.1). Das umfasst die Platzierung von GameObjects, Terrain-Gestaltung, Gestaltung der Umgebung (Skybox) und mehr.



Abbildung 6.2: Unity: Infowindow – zum Test mit *XmlLayout* erstellt.

### 6.1.2 UI-Test mit *XmlLayout*

Der Prototyp, der für das Testen des Designs entwickelt wurde, legte den Fokus auf die UI-Gestaltung. *UIElements*, das neue UI-System von Unity, wäre hierfür ideal gewesen. Durch die klare Trennung von UI und Spiellogik (vorzugsweise nach dem MVC-Pattern) gäbe es mehrere Vorteile:

1. **Versionierbarkeit**, da das Interface aus XML und USS (eine Abwandlung von CSS [71]) generiert wird. Im normalen UnityUI sind die Interface-Elemente Bestandteil der Szenen-Hierarchie und werden auch dementsprechend serialisiert. Das führt aber zu Problemen in der Versionierung, da sich interne IDs von Szenen-Objekten dynamisch ändern und nur schwer nachverfolgen lassen [72].
2. **Wiederverwendbarkeit** von UI-Gestaltung durch Styling-Angaben, die auf mehrere Objekte anwendbar sind. Ähnlich zu CSS, bei dem die Gestaltung über Klassen geschieht und Styling-Angaben kaskadieren.
3. **Entkopplung** der Gestaltung von der Szenen-Hierarchie. Die Gestaltung sollte nicht vom Szenen-Editor (Level-Editor) abhängig sein. Wenn das UI-Element als GameObject hinzugefügt ist, muss jedes Attribut einzeln an den Objekt geändert werden.

Das Problem war jedoch, dass sich *UIElements* beim Schreiben dieser Arbeit noch im Beta-Stadium befand [73]. Aus diesem Grund wurde nach einem Third-Party-Framework gesucht, um einen ähnlichen Effekt wie mit *UIElements* zu erzielen.

Die Wahl fiel auf das UI-Framework *XmlLayout* [74]. Das Framework erzeugt aus (in XML geschriebenen) Layout- und Styling-Angaben ein *UnityUI*-Interface. Attribute, die sonst über den Inspektor geändert werden, setzt *XmlLayout* automatisch. Dabei bietet das Framework folgende Features [75]:

1. **Layout** Die Komponenten des Unity UI-Systems können als XML-Tags wie `<Button>` und `<HorizontalLayout>` geschrieben werden. Aus diesen Tags werden anschließend die dazugehörigen UnityUI-Objekte erzeugt.
2. **Klassen** Elemente können keine oder mehrere Klassen zugeschrieben haben. Mithilfe von Klassen können UI-Elemente wie `<Button>` gestaltet werden. Dabei sind die verwendbaren Attribute abhängig vom Element. Bilder z.B. können einen `type` haben oder ein `<VerticalLayout>` ein `spacing`-Attribut.
3. **Defaults** Sie beschreiben die Gestaltung von Standard-UI-Elementen. Das heißt, die Default-Styling-Angaben wirken sich auf alle Elemente aus. Vergleichbar mit dem Typ-Selektor von CSS [76].
4. **Controller** Zu jedem XML-Layout kann man einen Controller zuweisen. Dieser behandelt Spieler-Input oder die Darstellung von Daten.
5. **Custom Attributes** Eigene Attribute, die die Werte oder Eigenschaften von UI-Komponenten verändern können. Beispielsweise wurde Das *Custom Attribute* `pixelsPerUnitMultiplier` geschrieben (siehe Code-Listing 1). Damit kann die Auflösung von Texturen erhöht werden; das ist z.B. bei der Darstellung auf Mobil-Geräten wichtig.

```
<XmlLayout xmlns="XmlLayout">
  <Defaults>
    <Panel
      class="debug-panel"
      color="rgba(0,0,0,0.8)"
      image="black-transparent-background"
      pixelsPerUnitMultiplier="24" />
    <Text
      class="debug-panel-text"
      color="white"
      font="Nunito-Regular" />
  </Defaults>
  <Panel class="debug-panel">
    <Text class="debug-panel-text" id="debug-panel-text">
      Debugging-Information
    </Text>
  </Panel>
</XmlLayout>
```

Listing 1: XmlLayout: Debug-Panel

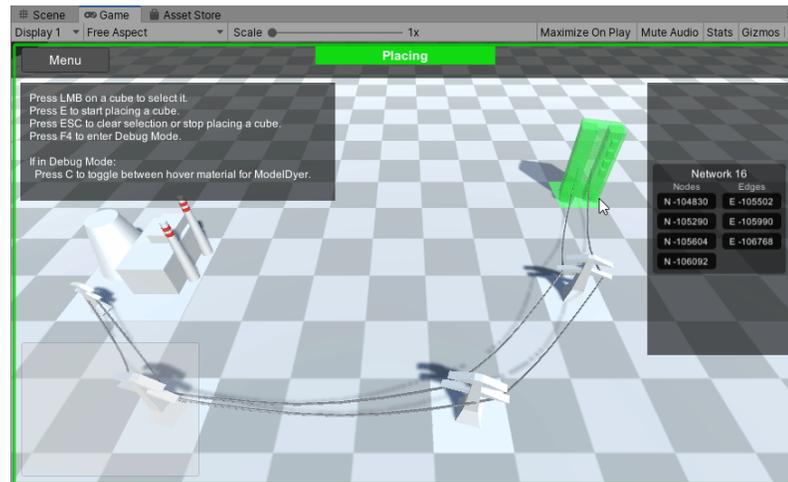


Abbildung 6.3: Unity: Spiele-Prototyp eines Kraftwerk-Simulations-Spiels.

### 6.1.3 Spiele-Prototyp in Unity

Neben dem UI eines Spiels ist vor allem die Spiellogik wichtig, sie macht erst die Spielmechaniken aus. Dazu wurde in Unity ein Prototyp eines Kraftwerk-Simulations-Spiels programmiert (siehe Abbildung 6.3). Zunächst war vorgesehen, den XmlLayout-UI-Test mit dem Spiele-Prototypen zu kombinieren. Es folgte jedoch ein Wechsel der Programmierumgebung (siehe Kapitel 6.2). Mit dem Unity-Prototypen wurden folgende Features umgesetzt:

1. Navigation der Kamera in der Spielwelt.
2. Auswählen von Gebäuden.
3. Platzieren und Abreißen von Gebäuden.
4. Aufbau eines Stromnetzes beim Platzieren von Kraftwerken und Strommasten.
5. Basic Interface und einfache 3D-Modelle.

## 6.2 Phaser-Prototyp

Nach dem ersten Test, Unity als Prototyp zu verwenden, folgte dann der Wechsel auf Phaser. Als Frontend wurde das JavaScript-Framework *Vue.js* verwendet, für die Darstellung im Spiel das HTML5-Game-Framework *Phaser*. Der Wechsel erfolgte aus mehrere Gründen:

1. Unity3D bietet alles, was man zur professionellen Programmierung von Videospiele braucht. Jedoch ist die Gestaltung von UI-Elementen bislang umständlich und zeitintensiv. Z.B. werden UI-Elemente als GameObjects zur Szene hinzugefügt. Bei Änderungen müssen in jeder Komponente die Daten einzeln angepasst werden.
2. HTML und CSS eignen sich sehr gut zur Frontend-Gestaltung. In Kombination mit einem UI-Framework kann man diese modular einsetzen.
3. Bei Unity muss man das Spiel speziell für das Ausgabe-Gerät bauen. Dagegen wenn der Spiel-Prototyp Web-basiert ist, kann man ihn mit den meisten Browser-fähigen Geräten direkt spielen.

### 6.2.1 Vue.js

*Vue.js* ist ein „progressive framework for building user interfaces“ [77]. Es basiert auf *JavaScript* und dient zum Entwickeln von Web Apps und *Single Page Applications*. Es ist auf Platz 2 der am meisten geliebten Web-Frameworks [78], [79].<sup>1</sup> *Vue.js* ist flexibel einsetzbar und gut dokumentiert. Mit dem Framework wurde das UI des Spiele-Prototypen implementiert.

### 6.2.2 Phaser

Phaser ist ein HTML5-Spiel-Framework. Es wurde speziell für moderne Browser erstellt. Das Framework erlaubt es, 2D-Spiele für Mobil und Desktop zu entwickeln. Die einzige Anforderung ist, dass der Browser das `<canvas>`-Element unterstützt [80]. Phaser wurde in *TypeScript* und *JavaScript* geschrieben.

### 6.2.3 Kommunikation zwischen Vue.js und Phaser

Die Kommunikation zwischen Phaser und *Vue.js* funktioniert relativ einfach. Wenn die Webseite mit dem HTML5-Spiel geladen wird, wird zuerst *Vue.js* (View) aufgerufen. Diese View erzeugt dann das darunterliegende Phaser-Spiel und das Interface an sich. Beim Erstellen werden View und Spiel miteinander verknüpft (siehe Code-Listing 2).

```
// Create Vue.js (view) instance
export default {
  name: 'app',
  created() {
    window.game = new Game(this); // Create game with view as argument
    this.game = window.game;     // Register game with view
  }
}

// Create Phaser (game) instance
import config from "./config";
export default class Game extends Phaser.Game {
  constructor(vue) {
    super(config); // Create game from game config
    this.vue = vue; // Register view with game
  }
}
```

Listing 2: *Vue.js* und Phaser: Setup der Kommunikation.

---

<sup>1</sup>Der *Stack Overflow Developer Survey* ist eine jährliche Umfrage, die vom *stackoverflow.com* organisiert wird. Bei der Umfrage 2019 haben knapp 90.000 professionelle und Hobby-Entwickler teilgenommen. Die Umfrage-Teilnehmer werden dabei zu verschiedenen Positionen wie Bildungsgrad, aktuelle Anstellung, beliebteste Programmiersprache oder Events in der Tech-Branche befragt.

Wenn sich nun der Zustand im Spiel ändert, kann die View entweder direkt oder über Events aktualisiert werden.

In diesem Beispiel (siehe Code-Listing 3) erhält der Spieler einen Geldbetrag. Der neue Kontostand wird als Event `FinanceEvent.FUNDS_UPDATED` gesendet. Dabei ist die View als *Listener* dieses Events registriert. Wenn sich also der Kontostand ändert, werden die Daten in der View aktualisiert. Daten in Vue.js sind *reactive* – bei einer Änderung werden die neuen Daten sofort angezeigt [81].

Events können auch von der View aus gesendet werden. Wenn in diesem Beispiel der Button „Geld leihen“ geklickt wird, sendet Vue.js das Event `'lend-money'` aus. Das Spiel wurde als *Listener* für dieses Event registriert. Klickt der Spieler den Button, wird also die Methode `financeManager.lendMoney()` ausgeführt.

```
// Local variables
let view = this.game.vue;
let finances = this.financeManager;

// Update view data if funds change
this.financeManager.eventEmitter.on(
  FinanceEvent.FUNDS_UPDATED, (funds, loan) => {
    view.$data.currentFunds = funds;
    view.$data.currentLoan = loan;
  }
);
// React to input events in view
view.$on('lend-money', (amount) => { finances.lendMoney(amount); });
view.$on('repay-money', (amount) => { finances.repayLoan(amount); });
```

Listing 3: Vue.js und Phaser: Kommunikation zwischen dem Spiel und UI über Events.

#### 6.2.4 Screenshots

Nachfolgend werden Screenshots des fertigen Spiel-Prototypen gezeigt. Die Gestaltung ist auf Mobil-Geräte ausgelegt. Interessanterweise ist der Code zur UI-Darstellung und der Code für die Spiellogik zu ca. 50/50 verteilt.

Der Spiele-Prototyp ist unter <http://power.lukassemmmler.de/> spielbar.



Abbildung 6.4: Phaser-Spiel-Prototyp. Normale Ansicht über die Spielwelt.

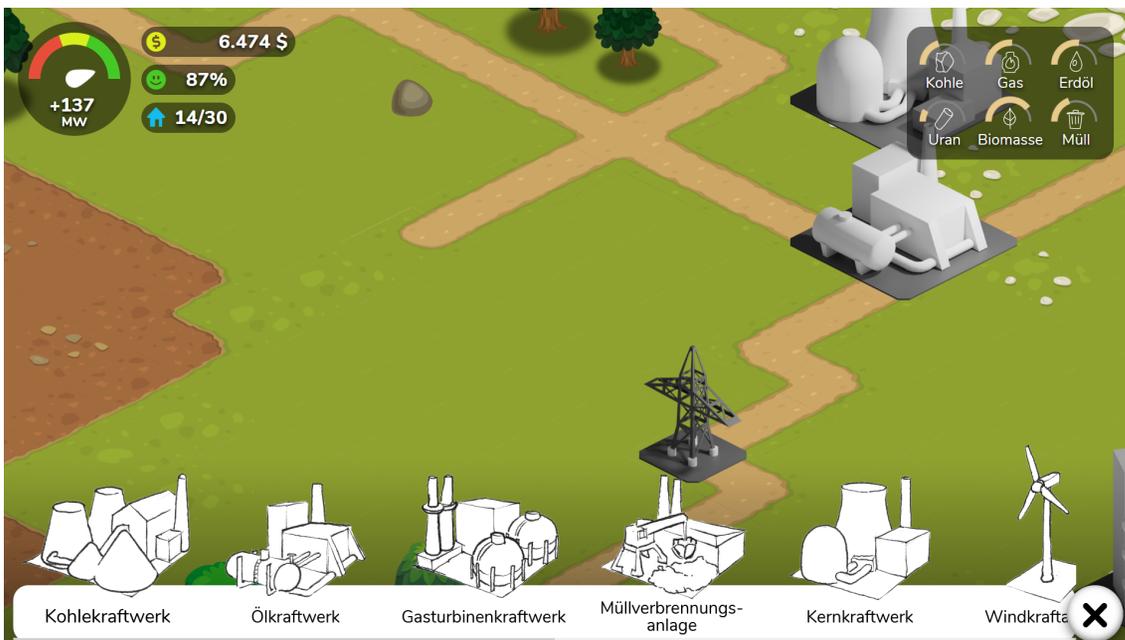


Abbildung 6.5: Phaser-Spiel-Prototyp. Geöffnetes Baumenü. Die Items in der Liste können in die Spielwelt gedraggt werden.

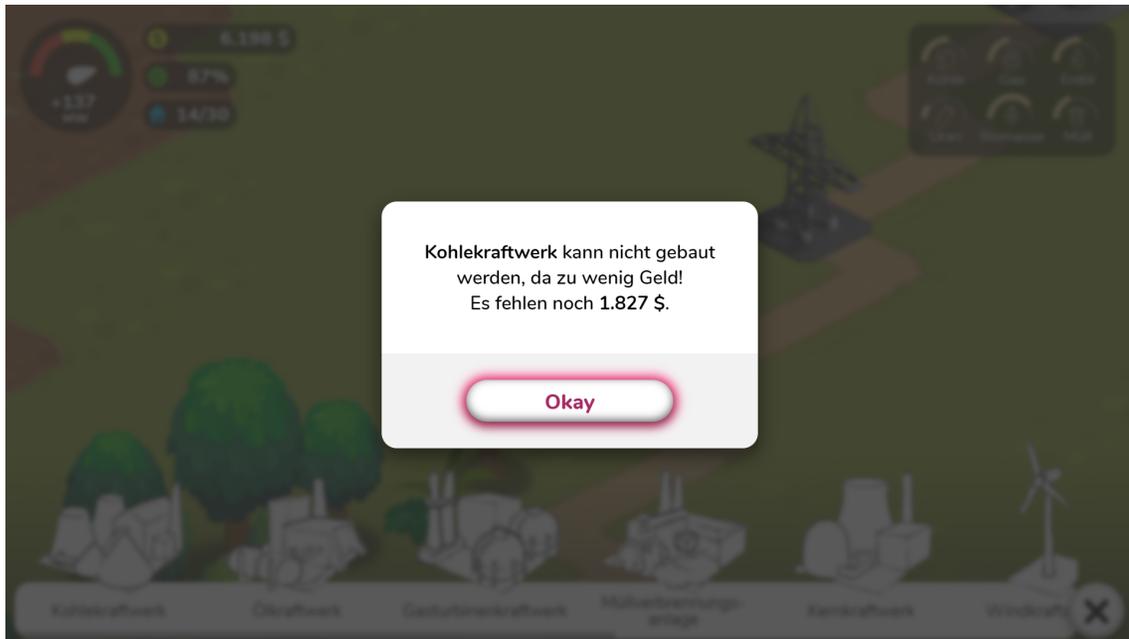


Abbildung 6.6: Phaser-Spiel-Prototyp. Man sieht ein geöffnetes Modal. Das Geld hat nicht gereicht zum Bau eines Kraftwerkes.

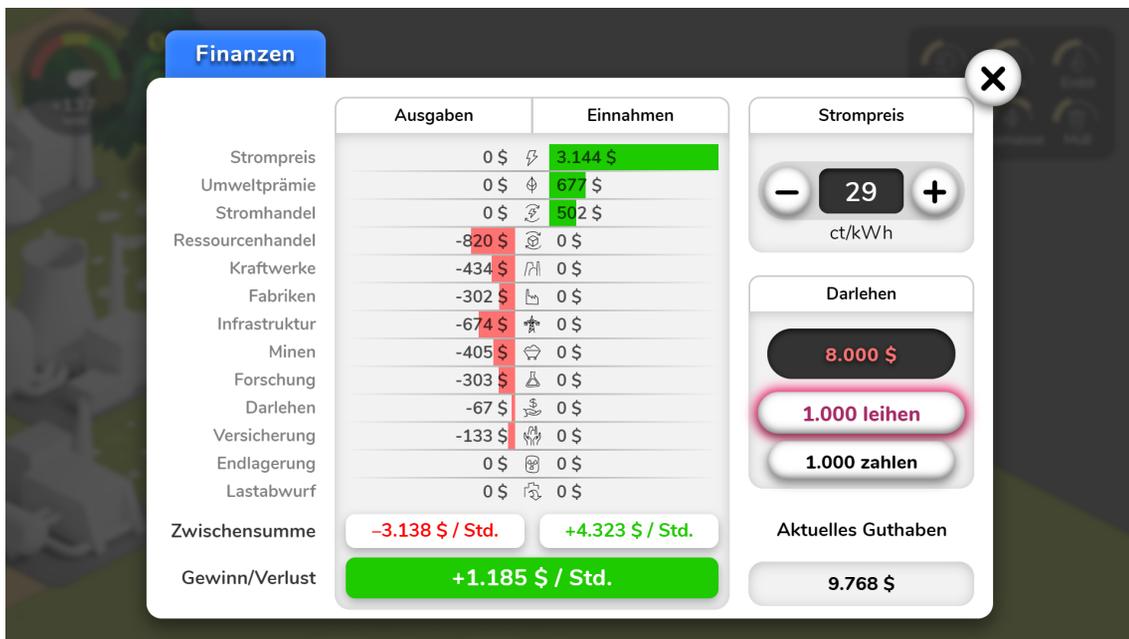


Abbildung 6.7: Phaser-Spiel-Prototyp. Finanzfenster, in dem gerade Geld geliehen wurde.



Abbildung 6.8: Phaser-Spiel-Prototyp. Gerade wird ein Kohlekraftwerk platziert.



Abbildung 6.9: Phaser-Spiel-Prototyp. Infenster mit detaillierten Informationen über ein Kraftwerk.



Abbildung 6.10: Phaser-Spiel-Prototyp. Das Infowindow des Baumenüs ist geöffnet.



Abbildung 6.11: Phaser-Spiel-Prototyp. Credits der Entwicklung.



---

# Kapitel 7

## Evaluation

Gemäß dem User-Centered-Design-Prozess wurden mithilfe von Nutzertests die Spiele-Prototypen überprüft. Dabei wurde zuerst der Klick-Prototyp (siehe Kapitel 4.5) und anschließend der voll implementierte Spiele-Prototyp (siehe Kapitel 6.2) getestet.

### 7.1 Nutzertest 1

Der erste Nutzertest diente dazu, den Stand des Klick-Prototypen zu überprüfen. Ziel war es, Usability-Probleme herauszufinden und was gut oder schlecht an dem Spiel wahrgenommen wurde. Dazu wurden insgesamt acht Probanden befragt.

#### 7.1.1 Aufbau

Der Aufbau des Nutzertestes war wie folgt:

1. **Datenschutzerklärung** Erklärung, ob der Teilnehmer bereit ist, an der Nutzerstudie teilzunehmen. Inklusive Hinweis auf Audio-Aufnahme.
2. **Personendaten und Nutzungsverhalten** Allgemeine Daten zur Person wie Alter und Geschlecht. Nutzungsverhalten bei *Mobile*, *Smart TV* und *PC/Laptop*.
3. **Präsentation** Vorstellung der Spiel-Idee (Kraftwerk-Simulations-Spiel). Beinhaltet kleines Tutorial, das den geplanten Spielablauf zeigt.
4. **Think-Aloud** Eine Usability-Test-Methode, bei der die Testperson alle spontanen Gedanken äußern soll, während er eine Anwendung benutzt. Der Proband musste hier sieben Aufgaben lösen (siehe Use-Cases in Kapitel 4.2). Mit dem Test sollte herausgefunden werden, ob das UI gut bedienbar ist.
5. **Semi-strukturiertes Interview** Ein Interview mit festen und offenen Fragen. Damit wurde untersucht, ob dem Probanden der Prototyp gefallen hat.
6. **UEQ und SUS** Der *User Experience Questionnaire* (UEQ) diente zur quantitativen Erfassung der User Experience. [82] Dabei wurde der Standard-UEQ verwendet. Als

nächstes kam der *System Usability Score* (SUS) zum Einsatz. Mit ihm wurde die Usability des Interfaces und des Spiels getestet [83].

Für den Nutzertest diente dabei ein Smartphone als Testgerät. Die Anforderungen an das Gerät waren folgende: 1) Ein Bildschirmformat von 16:9, 2) ein moderner Browser<sup>1</sup> und 3) ein leistungsfähiger Prozessor (damit der Prototyp ruckelfrei läuft).

Neben dem schriftlichen Fragebogen gab es auch eine Audioaufnahme. Die Audioaufnahme diente als Sicherheit, um nachträglich Lücken im Fragebogen zu ergänzen. Ein Laptop mit eingebautem Mikrofon wurde hierfür als Aufnahmegerät verwendet.

### 7.1.2 Durchführung

Im ersten Test war es wichtig, ein breites Spektrum an Videospiele-Erfahrung abzudecken. Wichtig war, dass die Bedienung von Smartphones für alle Probanden bekannt war. Insgesamt wurden acht Probanden befragt (siehe Tabelle 7.1).

ID	Geschlecht	Alter	Spielerfahrung	Spielhäufigkeit
P1	männlich	23	<b>Experte</b> MOBA	täglich
P2	männlich	32	<b>Fortgeschritten</b> Puzzle-Plattformer, Sidescroller	täglich
P3	männlich	26	<b>Fortgeschritten</b> Horror-Survival, MOBA, Battle Royale	3x pro Woche
P4	männlich	30	<b>Fortgeschritten</b> FPS, Strategie, MOBA	4x pro Woche
P5	männlich	27	<b>Fortgeschritten</b> FPS	4x pro Woche
P6	weiblich	28	<b>Anfänger</b> Fantasy, Simulation	selten
P7	männlich	26	<b>Fortgeschritten</b> Aufbau, Adventure	2x pro Woche
P8	männlich	26	<b>Anfänger</b> Gesellschaftsspiele, Ego-Shooter	selten

Tabelle 7.1: Probanden der ersten Nutzerstudie.

---

<sup>1</sup>In diesem Fall wurde der Prototyp mit der *Google Chrome App 80.0.3987.99* getestet.

### Vor dem Test

- Vor dem Test wurde der Prototyp auf einem Webserver hochgeladen. Als nächstes wurde das Testgerät (= Smartphone) gestartet und der Prototyp im Browser geladen. Zur Audio-Aufnahme wurde an einem Laptop das Aufnahme-Programm gestartet.

### Während dem Test

- Zu Beginn des Testes wurde dem Probanden die Datenschutzerklärung vorgelegt. Bei Einverständnis wurde der Test fortgesetzt.
- Anschließend wurde die Testperson als Proband erfasst (siehe Tabelle 7.1). Die Probanden wurden zu allgemeinen Daten, ihrer Videospiele-Vertrautheit und ihres Nutzungsverhaltens befragt. Das Nutzungsverhalten bezog sich dabei auf *Mobile*, *Smart TV* und *PC/Laptop*.
- Es folgte eine kurze Präsentation der Spielidee. Unter anderem beinhaltete die Präsentation ein Tutorial. Das Ziel des Spiels ist es, ein Gleichgewicht zwischen Stromerzeugung, Kundenzufriedenheit und Wirtschaftlichkeit zu finden. Außerdem wurde dem Probanden erklärt, dass das Spiel für verschiedene Endgeräte konzipiert ist.
- Das *Think-Aloud* fand im Anschluss statt. Dabei wurde dem Probanden vorab erklärt, dass er Aufgaben gestellt bekommt. Sie sollten gelöst werden und der Proband soll dabei laut aussprechen, was er gerade denkt.
- Mit dem semi-strukturierten Interview wurden Fragen zum Prototypen gestellt. Daneben waren auch Fragen zur Einstellung zu Free-2-Play-Spielen oder zum Icon-Design enthalten.
- Anschließend folgten der *UEQ* und *SUS*. Der Proband durfte die standardisierten Fragebögen selbstständig ausfüllen. Sie wurden anonym eingesammelt.

### Nach dem Test

- Nach dem Probandentest wurde zuerst die Audioaufnahme gestoppt. Anschließend wurde der Testperson für die Teilnahme gedankt und die Aufnahme gespeichert.

### 7.1.3 Auswertung

#### Think-Aloud

Kritik, Lob oder allgemeine Anmerkungen zum Prototypen wurden erfasst und anschließend ausgewertet. Die während des Think-Aloud gestellten Aufgaben basierten auf den zuvor definierten Use-Cases (siehe Kapitel 4.2).

Aufgabe	Bemerkungen
1. Sieh dich auf der Karte um. Gibt es jemanden, der gerade besonders glücklich oder genervt ist?	<ul style="list-style-type: none"> <li>• Maiskolben steht nicht für Biogasanlage</li> <li>• Sprechblase sieht nicht aus, als könnte man klicken</li> <li>• Pinch/Zoom-Geste wird versucht, aber gibt es nicht</li> </ul>
2. Finde heraus, wie viel es kostet, einen Bahnhof zu bauen.	<ul style="list-style-type: none"> <li>• Icon für „Other Buildings“ als Baumenü verstanden</li> <li>• Es fehlt Beschriftung der aktuellen Bau-Kategorie</li> <li>• LKW-Icon wurde nicht als „Logistik“ angesehen</li> </ul>
3. Baue ein Kohlekraftwerk.	<ul style="list-style-type: none"> <li>• Verwechslung zwischen Kraftwerk- und Fabrik-Icon</li> <li>• Swipe von Baumenü in Spielwelt hat verwirrt</li> <li>• Meist geklickt, um einen „Baumodus“ zu aktivieren</li> <li>• Verschieben des Kraftwerkes funktioniert nicht gut</li> <li>• → unter anderem war Drag-Bereich zu klein</li> </ul>
4. Für den Bau eines weiteren Kohlekraftwerkes hast du nicht genug Geld. Nimm einen Kredit auf.	<ul style="list-style-type: none"> <li>• Interface wurde manchmal als überladen empfunden</li> <li>• Aktueller Kontostand ist meist übersehen worden</li> <li>• Klickbare Geldanzeige wirkt nicht klickbar</li> </ul>
5. Finde heraus, bei welchem Prozentsatz die Strom-Versorgungssicherheit liegt.	<ul style="list-style-type: none"> <li>• Zu viele Infos auf einmal in Stromübersicht</li> <li>• Bedeutung der Diagramme nicht sofort klar</li> <li>• Prozentzahl schwierig zu sehen</li> </ul>
6. Verschiebe das gebaute Kraftwerk.	<ul style="list-style-type: none"> <li>• Anklicken für Infomenü zum Verschieben war gut</li> <li>• Ein Proband wollte länger gedrückt halten</li> </ul>
7. Blende Informationen in der Karte zur Stromversorgung ein.	<ul style="list-style-type: none"> <li>• Gestaltung des Info-Icons war zu allgemein</li> <li>• Viele wollten auf Strom-Gebäude für Infos klicken</li> </ul>

Tabelle 7.2: Nutzertest 1: Bemerkungen zu Aufgaben.

Aufgabe	1	2	3	4	5	6	7
Gelöst (absolut)	6/8	8/8	8/8	7/8	5/8	8/8	8/8
Gelöst (relativ)	75%	100%	100%	87,5%	62,5%	100%	100%

Tabelle 7.3: Nutzertest 1: Lösbarkeit der Aufgaben.

## Semi-strukturiertes Interview

Frage	Antworten
A. Welche Icons findest du gut und welche nicht?	<ul style="list-style-type: none"> <li>• Lastwagen steht nicht für Bahnhof</li> <li>• Info-Icon für Strom-Info in Karte ist zu allgemein</li> <li>• Fabrik- und Kraftwerk-Icon sind zu ähnlich</li> <li>• Forschungsanzeige ist nicht erkennbar</li> <li>• „Hügel“ nicht als Kohlehaufen für Ressourcen erkennbar</li> <li>• Haus ist entweder Haushalte oder Bewohner</li> <li>• „Andere Gebäude“-Icon wurde oft für Baumenü gehalten.</li> </ul>
B. Kannst du dir vorstellen, dieses Spiel auch in deiner Freizeit zu spielen?	<p>Antwort abhängig vom Probanden. Insgesamt war die Meinung, wenn mehr Features vorhanden sind und es eine lebendigere und schönere Gestaltung hat, dann vielleicht.</p>
C. Was ist dein Lieblingsspiel? Und warum?	<ul style="list-style-type: none"> <li>• <i>Portal 2</i> – coole Story und cooles Gameplay</li> <li>• <i>Secret of Mana</i> – Kindheitsspiel und Co-op</li> <li>• <i>DotA 2</i> – Teamspiel und flexibel spielbar mit Helden</li> <li>• <i>Call of Duty: Modern Warfare</i> – <i>competitive</i> und FPS</li> <li>• <i>ShopTitans</i> – cool, da Abenteurer-Ausrüstungs-Laden-Besitzer</li> </ul>
D. Interessierst du dich für Tycoon-Spiele?	<ul style="list-style-type: none"> <li>• Ja, gut um nebenbei mit Kumpel in Teamspeak zu reden</li> <li>• Hab mal <i>Cities: Skylines</i> ausprobiert, aber Gameplay zu ruhig</li> <li>• Ist ganz interessant</li> <li>• Ja, ich habe bereits <i>SimCity 4</i> und <i>Cities: Skylines</i> gespielt</li> </ul>
E. Was hältst du von Free-2-Play-Mobile-Spielen?	<ul style="list-style-type: none"> <li>• Relativ wenig; hab schonmal <i>Clash of Clans</i> gespielt</li> <li>• F2P ist in Ordnung; Mobile-Spiele sollten aber <math>\leq 4</math> € kosten</li> <li>• Cool ist, dass man schnell etwas ausprobieren kann</li> <li>• Werbung ist nervig</li> <li>• Viele Spiele sind ziemlich sinnlos und nur zum süchtig machen</li> <li>• Find ich gut</li> <li>• Super seltsam, wenn lauter kleine Käufe im Spiel sind</li> </ul>
F. Gibt es ein Spiel, auf dessen Release du dich freust? Wenn ja, warum?	<ul style="list-style-type: none"> <li>• <i>Cyberpunk 2077</i> – schaut ziemlich cool aus</li> <li>• <i>Final Fantasy VII</i> Remake – ich mag JRPG und das Original</li> <li>• <i>Age of Empires IV</i> – mehr Features als vorher</li> <li>• <i>Diablo IV</i> – Trailer ist cool</li> </ul>
G. Denkst du, dieses Spiel würde Spaß machen? Wenn nein, was fehlt?	<ul style="list-style-type: none"> <li>• Momentan zu wenig Inhalt</li> <li>• Ja, gut wenn man einen Aspekt (Stromversorgung) fokussiert</li> <li>• Mehr Aufgaben müssen drin sein</li> <li>• Belohnungen, wenn man Leute zufrieden macht</li> <li>• „Je komplexer, desto besser“; kann Spaß machen wer Tycoon mag</li> <li>• Wenn Gestaltung besser, könnte es Spaß machen</li> </ul>
Sonstige Bemerkungen	<ul style="list-style-type: none"> <li>• Nervig wenn Spielstand verloren geht</li> <li>• Grafik ansprechend, erinnert an <i>MiniMetro</i></li> <li>• Atompilz wäre cool</li> <li>• Zeit-Vorlauf-Modus wäre noch cool</li> <li>• Bisher uneinheitliche Gestaltung (z.B. Strichstärke bei Icons)</li> <li>• Menschlicher Einfluss fehlt noch (es wirkt öde)</li> <li>• Es fehlt noch ein Hauptmenü</li> </ul>

Tabelle 7.4: Nutzertest 1: Antworten beim Semi-strukturierten Interview.

### UEQ und SUS

Für Ergebnisse des UEQ, siehe Abbildung 7.1 und 7.2.

Für SUS-Score, siehe Kapitel 7.3.2.

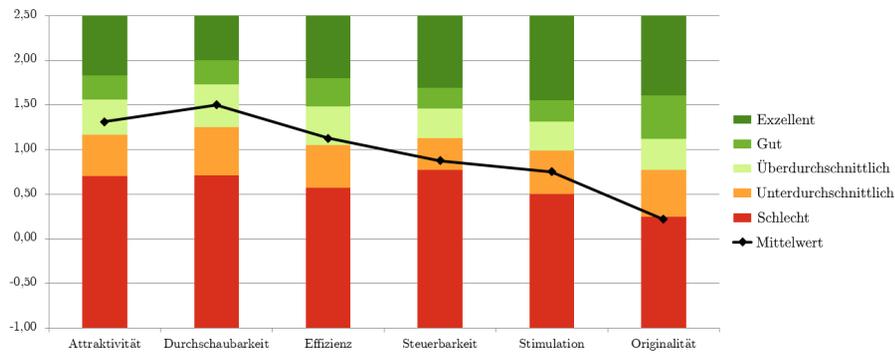


Abbildung 7.1: Nutzertest 1: UEQ Benchmark.

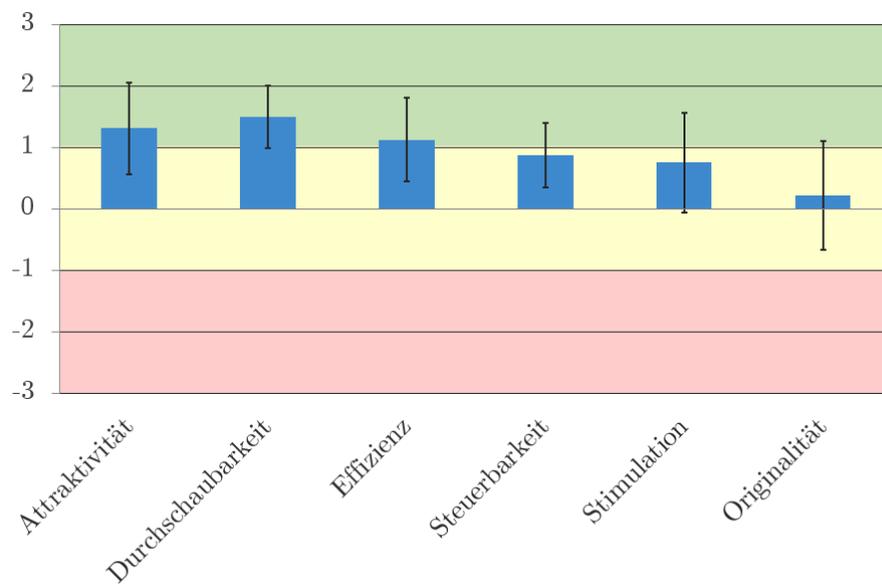


Abbildung 7.2: Nutzertest 1: Ergebnisse des UEQ nach Kategorien.

## 7.2 Nutzertest 2

Der zweite Nutzertest wurde während des *Global Game Jam 2020*<sup>2</sup> in Ansbach abgehalten. Die Probanden waren dabei vertrauter mit Videospiele als beim ersten Test.

### 7.2.1 Aufbau

Der Aufbau des zweiten Nutzertestes orientierte sich am ersten Nutzertest. Er wurde aber vereinfacht und verkürzt aus zwei Gründen: 1) Mit weniger offenen Fragen konnte der Test einfacher mit der vorherigen Runde verglichen werden. 2) Bei einer kürzeren Dauer waren die Besucher des *Global Game Jams* eher bereit, mitzumachen. Der Aufbau des Nutzertestes war wie folgt: (siehe Kapitel 7.1.1 für Aufbau des ersten Tests)

1. **Datenschutzerklärung**
2. **Personendaten und Nutzungsverhalten**
3. **Think-Aloud** (Der letzte Use-Case ist weggefallen, da nicht implementiert. )
4. **UEQ und SUS**

ID	Geschlecht	Alter	Spielerfahrung	Spielhäufigkeit
Q1	männlich	24	<b>Fortgeschritten</b> Aufbau-Simulation, Minecraft	1–2x pro Woche
Q2	männlich	28	<b>Fortgeschritten</b> RTS	3–4x pro Woche
Q3	männlich	27	<b>Fortgeschritten</b> Aufbau-Simulation, Grand Strategy, Shooter, Flug-Simulator	4x pro Woche
Q4	männlich	27	<b>Fortgeschritten</b> Action-RPG, MMORPG, MOBA, RTS	1–2x pro Woche
Q5	männlich	34	<b>Experte</b> Aufbau-Simulation, MMORPG, RTS, Survival, Simulation	3x pro Woche
Q6	weiblich	24	<b>Anfänger</b> Mobile game	3–4x pro Woche
Q7	männlich	20	<b>Fortgeschritten</b> Action-RPG	3–4x pro Woche
Q8	männlich	21	<b>Fortgeschritten</b> Aufbau-Simulation, Racing, Simulation	2–3x pro Woche
Q9	männlich	29	<b>Fortgeschritten</b> MOBA, Auto Battler, Trading Card Game	täglich
Q10	männlich	22	<b>Experte</b> Aufbau-Simulation, Grand Strategy, RTS, Shooter, Stealth	täglich
Q11	männlich	18	<b>Fortgeschritten</b> Action-RPG, MMORPG	täglich

Tabelle 7.5: Probanden der zweiten Nutzerstudie.

<sup>2</sup>Der *Global Game Jam* (GGJ) ist das weltweit größte *Game Jam*-Event. Er findet einmal jährlich und gleichzeitig an verschiedenen Orten in der Welt statt. Während des *Game Jams* bilden sich Gruppen, die innerhalb eines Wochenendes ein Spiel zu einem gegebenen Thema programmieren [84].

### 7.2.2 Durchführung

Anders als beim ersten Test stand kein extra Raum zur Verfügung, sodass für die Audioaufnahme mögliche Störgeräusche in Kauf genommen wurden. Aufgrund des kürzeren Aufbaus dauerte der Test nur ca. 20 Minuten (siehe Kapitel 7.1.2 für Durchführung des ersten Nutzertestes.)

### 7.2.3 Auswertung

#### Think-Aloud

Aufgabe	Bemerkungen
1. Sieh dich auf der Karte um. Gibt es jemanden, der gerade besonders glücklich oder genervt ist?	<ul style="list-style-type: none"> <li>• Wurde von allen gut gelöst</li> <li>• Man soll weiter hinauszoomen können</li> </ul>
2. Finde heraus, wie viel es kostet, einen Bahnhof zu bauen.	<ul style="list-style-type: none"> <li>• Mehrmals „Other Buildings“ für Baumenü gehalten</li> <li>• → Logistik dann meist richtig bei 2. Versuch</li> </ul>
3. Baue ein Kohlekraftwerk.	<ul style="list-style-type: none"> <li>• Kraftwerk folgt nicht durchgängig Swipe</li> <li>• Infofenster sollte Produktionsketten anzeigen</li> <li>• Platzier-Dialog verschwindet hinter Gebäuden</li> </ul>
4. Für den Bau eines weiteren Kohlekraftwerkes hast du nicht genug Geld. Nimm einen Kredit auf.	<ul style="list-style-type: none"> <li>• Wenn nicht genug Geld, schon in Baumenü anzeigen</li> <li>• <i>Floating Text</i> bei Platzieren, wie viel es kostete</li> <li>• Aktuelles Guthaben wird durch Finger verdeckt</li> </ul>
5. Finde heraus, bei welchem Prozentsatz die Strom-Versorgungssicherheit liegt.	<ul style="list-style-type: none"> <li>• Wurde von allen gut gelöst</li> <li>• Stromzähler sollte wieder zu Infofenster führen</li> </ul>
6. Verschiebe das gebaute Kraftwerk.	<ul style="list-style-type: none"> <li>• Option fehlte im Info-Menü zu verschieben</li> <li>• Gedrückt-Halten-Dauer teilweise zu lang (500 ms)</li> <li>• Planierdraht-Icon fehlt zum Abreißen</li> </ul>
7. Blende Informationen in der Karte zur Stromversorgung ein.	<b>(Weggefallen)</b>

Tabelle 7.6: Use Cases im zweiten Probandentest.

Aufgabe	1	2	3	4	5	6	7
<b>Gelöst (absolut)</b>	11/11	11/11	11/11	11/11	11/11	10/11	–
<b>Gelöst (relativ)</b>	100%	100%	100%	100%	100%	91%	–

Tabelle 7.7: Lösbarkeit der Use Cases im zweiten Probandentest.

## UEQ und SUS

Für Ergebnisse des UEQ, siehe Abbildung 7.3 und 7.4.

Für SUS-Score, siehe Kapitel 7.3.2.

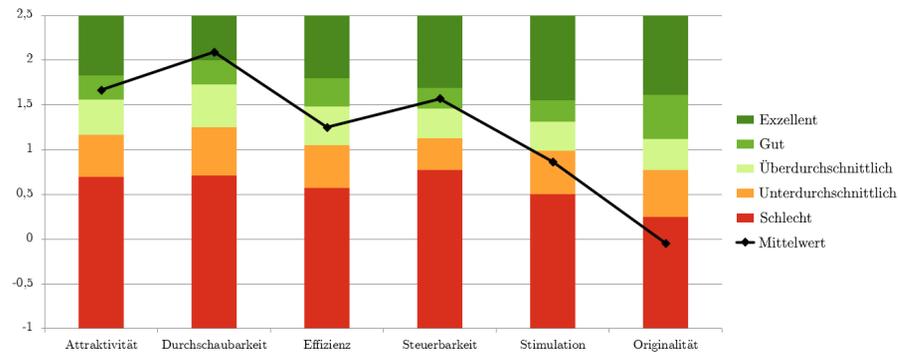


Abbildung 7.3: Nutzertest 2: UEQ Benchmark.

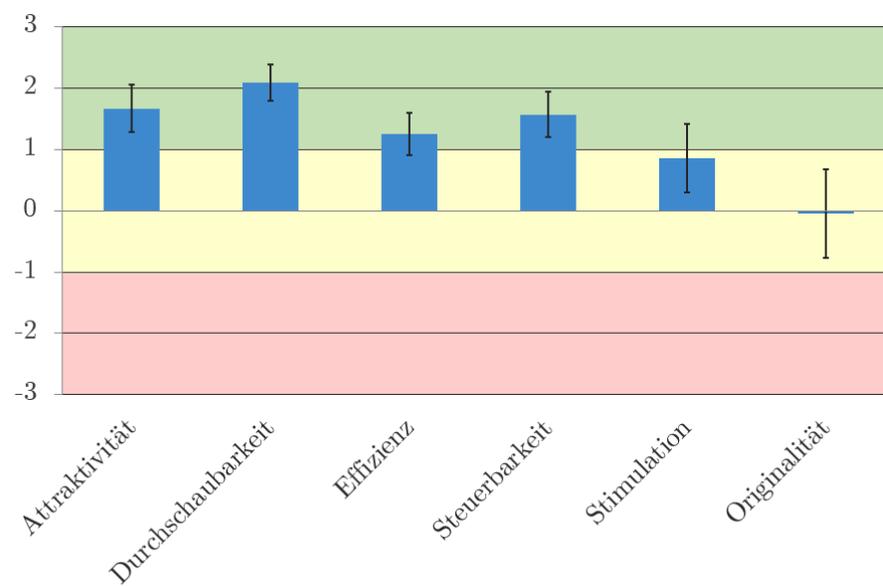


Abbildung 7.4: Nutzertest 2: Ergebnisse des UEQ nach Kategorien.

## 7.3 Vergleich

Beide Nutzertests wurden abgehalten, um einen Vorher-Nachher-Vergleich durchführen zu können.

### 7.3.1 Think-Aloud

Die Lösbarkeit der Aufgaben beim Think-Aloud lag beim ersten Test bei 89 %. Im zweiten Nutzertest ist sie auf 99 % gestiegen. Ein Grund der besseren Lösbarkeit liegt vermutlich darin, dass der Spiele-Prototyp von Grund auf mit einer Spiele-Engine geschrieben wurde. Die Aufgabe „Platziere ein Kohlekraftwerk“ beinhaltet nämlich, dass der Spieler mit *Drag and Drop* ein Gebäude aus dem Baumenü in die Spielwelt zieht. Aufgrund von technischen Einschränkungen war im ersten Prototyp keine *Swipe*-Geste möglich. Stattdessen mussten die Probanden über ein Debug-Menü den Bau-Prozess per Hand starten. Im übrigen wurden beim zweiten, programmierten Prototyp die üblichen Smartphone-Gesten wie *Pinch*, *Zoom*, *Swipe*, *Tap* und *LongClick* unterstützt.

### 7.3.2 UEQ und SUS

Mit der zweiten Testrunde haben sich insgesamt die Werte in den UEQ-Kategorien verbessert (siehe Abbildung 7.5). Jedoch sind diese Verbesserungen für ein  $\alpha$ -Level von 5% nicht signifikant. Für  $\alpha = 10\%$  gibt es einen signifikanten Unterschied bei der „Durchschaubarkeit“ und „Steuerbarkeit“. Anders gesagt: Für  $\alpha = 10\%$  ist die Durchschaubarkeit und Steuerbarkeit des zweiten Spiele-Prototypen signifikant besser als beim ersten.

Die Daten weisen insgesamt eine große Varianz auf, wobei diese beim zweiten Durchlauf etwas geringer ausfällt. Es könnte zwei mögliche Ursachen geben:

- Die Stichprobe beim ersten Test ( $N = 8$ ) war kleiner als beim zweiten ( $N = 11$ ).
- Die Probanden des ersten Testes waren, vom Videospiele-Verhalten her, recht heterogen verteilt. *Global Game Jam*-Teilnehmer waren dagegen insgesamt sehr Videospiele-affin.

Auch beim *SUS-Score* gibt es keinen signifikanten Unterschied (siehe Abbildung 7.6). Die Usability scheint etwas besser geworden zu sein mit dem zweiten Prototypen. Mit einem Score von 71 wäre der Spiele-Prototyp immerhin besser als 60 % aller anderen getesteten Produkte.<sup>3</sup>

---

<sup>3</sup> Mit „getesteten Produkten“ ist der Perzentil-Rang von <https://measuringu.com/> gemeint. Das Unternehmen *MeasuringU* hat dafür SUS-Fragebogen-Antworten von 4.664 Probanden ausgewertet [85].

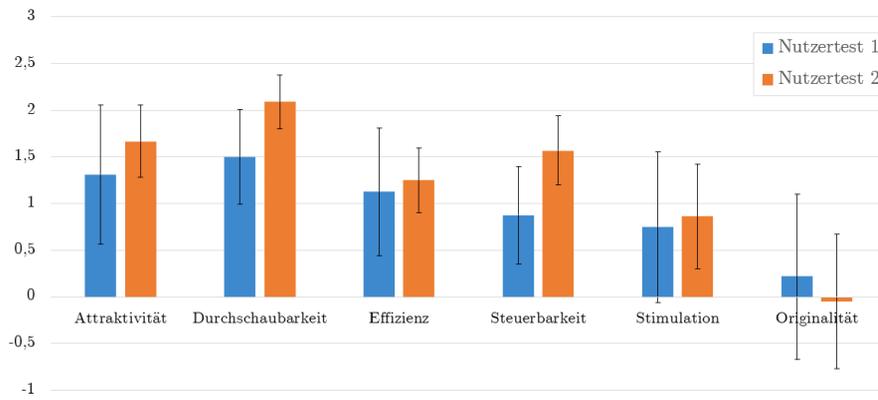


Abbildung 7.5: Vergleich Nutzertests: Werte für die Kategorien des UEQ.

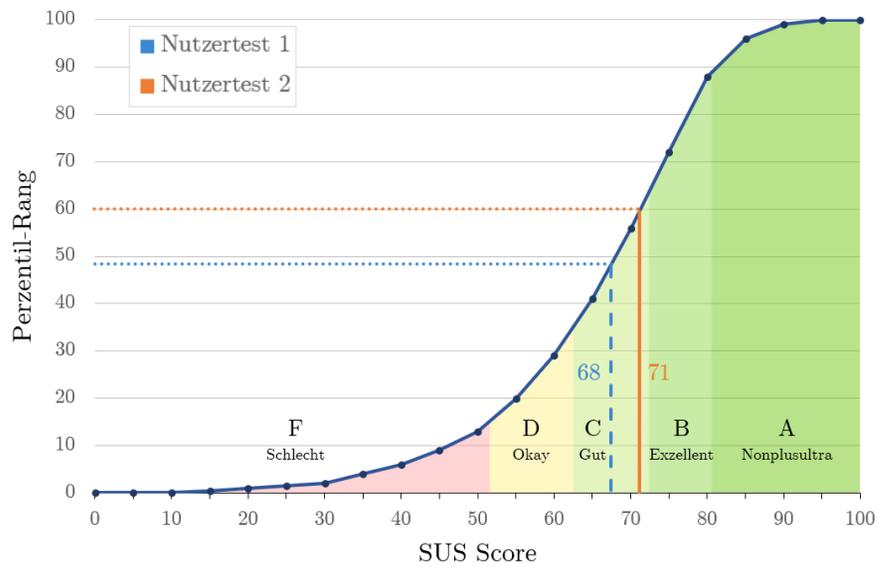


Abbildung 7.6: Vergleich Nutzertests: SUS-Score.



---

## Kapitel 8

# Zusammenfassung und Ausblick

Zusammenfassend kann man sagen, dass die Gestaltung und Implementierung eines Videospiele sehr aufwendig ist. Deshalb wurde in dieser Arbeit prototypisch ein Spiel-Konzept nach UX-Design-Methoden umgesetzt.

Zunächst wurden in der Analysephase, mithilfe des *Gamer Motivation Model*, die Anforderungen an einen Spiele-Prototypen des Genres Aufbau-Simulation aufgestellt. Mit einem *Game Design Documents* und den aus den Anforderungen abgeleiteten Use Cases wurde ein Konzept für einen Spiele-Prototypen ausgearbeitet. Auch ist in der Konzept-Phase der Klick-Prototyp entstanden, der im ersten Nutzertest getestet wurde. Das erste grobe Layout diente dazu, in der Design-Phase die angestrebte Gestaltung festzulegen. Hier wurde der Fokus auf die Anpassbarkeit des Designs für die drei verschiedenen Endgeräte gelegt. Anschließend wurde in der Implementierung ein Spiele-Prototyp umgesetzt.

Auffällig bei der Implementierung (siehe Kapitel 6) ist, dass es beim Umsetzen eines Prototypen nicht unbedingt auf den Aufwand ankommt. Im Zuge dieser Arbeit wurden drei Prototypen entwickelt: 1) Der Klickprototyp (siehe Kapitel 4.5), mit dem das erste Mal das Konzept überprüft wurde. 2) Der Unity-Prototyp (siehe Kapitel 6.1.1), als Versuch, den Spiele-Prototypen professionell aufzubauen. 3) Der HTML5-Prototyp (siehe Kapitel 6.2), aufbauend auf Vue.js und Phaser. Tatsächlich ist es empfehlenswert, Spielkonzepte zuerst mit einem Klick-Prototypen auszuprobieren. Die Schwierigkeit liegt darin, eine Balance zu finden zwischen Erstellungsaufwand und Features des Prototypen. Schon beim Entwickeln mit dem Klick-Prototypen gab es Performance-Probleme, da das verwendete Programm *Axure* nicht für solche Zwecke ausgelegt ist.

Der erste und dritte Spiele-Prototyp wurden mit Nutzertests überprüft (siehe Kapitel 7). Insgesamt 19 Probanden, mit gemischter Videospieleerfahrung, haben an den Nutzerstudien teilgenommen. Das Ergebnis der Tests ist, dass der Spiele-Prototyp als Ganzes gut bedienbar ist. Laut der Abbildung 7.4 befindet sich der UEQ-Wert für die Kategorien „Attraktivität“, „Durchschaubarkeit“, „Effizienz“ und „Steuerbarkeit“ im grünen Bereich. Der Wert der Kategorie „Stimulation“ ist knapp unter 1, die Originalität geht gegen 0 hin. Daraus kann man schließen, dass der Prototyp für die meisten Probanden attraktiv war, aber nicht als besonders stimulierend oder originell wahrgenommen wurde. Videospiele sind darauf ausgelegt, Spaß zu machen (siehe Kapitel 3.4). Ein Bestandteil von Spaß ist die

gute Usability. Mit überdurchschnittlich guten Werten in den Kategorien „Effizienz“ und „Steuerbarkeit“ wurde dieses Ziel erreicht.

Der Spiele-Prototyp, der im Zuge dieser Arbeit erstellt wurde, ist dem Genre der Aufbau-Simulations-Spiele zuzuordnen. Wie in der Einleitung bereits beschrieben, sind Zugehörigkeiten zu einem bestimmten Videospiel-Genre sehr wichtig (siehe Kapitel 2.2.3). Das Genre stellt immer implizite Anforderungen an das Gameplay und die UI-Gestaltung eines Videospiele. Die genaue Ursache für den moderat guten Stimulations-Wert konnte in dieser Arbeit nicht ermittelt werden. Vermutlich liegt die Ursache darin, dass Aufbau-Simulations-Spiele sich durch eine gewisse Komplexität ausdrücken müssen. So wurde von mehreren Probanden erwähnt, dass das Spiel noch nicht genug „Tiefgang“ bietet. Anders gesagt: Aufbau-Simulations-Spiele werden erst interessant, wenn verschiedene Systeme miteinander zusammenwirken. So wurde dies von Maxis bereits früh erkannt (siehe Kapitel 3.3).

Während dieser Arbeit sind auch viele Ideen entstanden, die aus Zeitgründen nicht mehr umgesetzt wurden. Das umfasst unter anderem:

- Aufbau von Warenkreisläufen
- Handel von Ressourcen und Strom
- Strahlenbelastung durch Kernkraftwerke und Atommüll
- Verschmutzungssystem unter Einbezug von Luftbelastung und Grundwasser
- Einbezug der Bevölkerung mit privaten Solaranlagen
- Logistiksystem, bei dem Waren mit Straße, Schiene oder Kanal transportiert werden
- Modulare Kraftwerksbauten, die nach Bedarf erweitert werden können
- Landschaft mit Höhenunterschiede, mit Land aufschütten und abtragen
- Politik und Proteste der Bevölkerung
- Forschungssystem, bei dem entweder in fossile oder erneuerbare Energieträger investiert werden kann
- Stromlast, die abhängig ist von Tageszeit
- Großabnehmer wie Stahlschmelzen zum regeln von Produktionsspitzen
- Naturkatastrophen, die zufällig oder vom Spieler ausgelöst werden können
- Verschiedene Stromnetze, die gleichzeitig betrieben werden
- Minimap (Übersichtskarte) der Spielwelt
- Spiel ist gewonnen, wenn alle Städte mit Strom versorgt sind
- Ressourcen-Filter, der Rohstoffvorkommen in Spielwelt zeigt
- Positive Auszeichnungen wie Umweltprämie für erneuerbare Energien
- Nachrichten, die bei Events oder Finanzberichten eingehen

Abschließend bleibt zu sagen, dass mit dieser Arbeit ein erster spielbarer Prototyp eines Aufbau-Simulations-Spiels über Stromnetz-Versorgung umgesetzt wurde. Die nächsten Schritte wären nun, das Feedback aus dem zweiten Nutzertest umzusetzen und mehr Features einzubauen. Oberste Priorität hat, ein Spiel zu erstellen, das dem Spieler eine gute Experience ermöglicht.





---

# Erklärung

Ich erkläre hiermit, dass ich die Arbeit selbständig verfasst und noch nicht anderweitig für Prüfungszwecke vorgelegt habe. Ich habe keine anderen als die angegebenen Quellen oder Hilfsmittel benutzt sowie wörtliche und sinngemäße Zitate als solche gekennzeichnet.

Ingolstadt, \_\_\_\_\_

\_\_\_\_\_  
Lukas Marinus Semmler

# Danksagung

An dieser Stelle möchte ich mich bei allen bedanken, die mich während des Schreibens dieser Bachelorarbeit unterstützt und motiviert haben.

Mein besonderer Dank gilt meinem Zweitbetreuer **Mathias Haimerl**. Er hat mich beim Schreiben dieser Arbeit immer mit Feedback, Ideen, Info-Materialien und Tipps bei Programmierproblemen unterstützt. Auch hat sich Mathias dafür eingesetzt, dass ich beim *Global Game Jam 2020* (GGJ) in Ansbach meine Nutzertests abhalten durfte. In wöchentlichen Meetings hatte er, auf freundliche und optimistische Art, immer eine Antwort auf meine Fragen.

Auch möchte ich mich bei meiner Erstbetreuerin **Prof. Ingrid Stahl** recht herzlich bedanken. Bei Fragen zu Design, Organisatorischem und wichtigen Entscheidungen hatte sie immer ein offenes Ohr für mich und hat mich mit Rat und Tat unterstützt.

Vielen Dank an alle Teilnehmer und Teilnehmerinnen meiner beiden Nutzertests. Ohne euer Feedback hätte diese Arbeit nicht entstehen können. Danke auch an die Organisatoren des *Global Game Jam 2020* in Ansbach, **Fabian Ihl** und **Juliane Vogt**.

Ein herzlicher Dank auch an meine beiden Brüder **Gregor** und **Jakob**, die mich immer für meine Arbeit motiviert haben und eine große Hilfe bei der Korrektur waren.

Danke an mein Team während des GGJ, das mich so bereitwillig aufgenommen hat, obwohl ich aufgrund der Nutzertests nicht immer da sein konnte. Danke **Eric**, **Fabian „akuryou“**, **Fabian „Brocc“**, **Giulio**, **Richard** und **Thin!**



1. Reihe (von links): Fabian, Fabian, Eric, Thin!; 2. Reihe: Richard, Lukas; 3. Reihe: Giulio

# Abbildungsverzeichnis

2.1	Entwicklung der Netto-Stromerzeugung in TWh (Stand: Okt. 2019) [9, S. 59]	5
2.2	Entwicklung der installierten Leitung von Anlagen mit Zahlungsanspruch nach dem EEG bis 2018 [9, S. 84]	6
2.3	Ausschnitt aus dem Verlauf der geplanten <i>Juraleitung</i> [13].	7
3.1	Motivation von Gamern: Sechs Motivations-Paare nach dem <i>Gamer Motivation Model</i> von Quantic Foundry [26].	11
3.2	Screenshot SimCity (2013): Verwalten einer Stadt [34].	15
3.3	Die beliebtesten Spiele-Plattformen in Deutschland. Stand: Juli 2019 [37]	18
3.4	Vergleich zwischen verschiedenen Spiele-Controllern. [50]–[53]	19
3.5	Unterstützer Input der <i>Amazon Fire TV 2-Remote</i> .	20
4.1	Use Case: Verlauf für das Platzieren eines Gebäudes.	25
4.2	Use Case: Bewegen der Kamera in der Spielwelt.	25
4.3	Wireframes: Smartphone-Ansicht des ersten Konzeptes.	28
4.4	Wireframes: Smart TV-Ansicht des ersten Konzeptes.	29
4.5	Wireframes: Desktop-Ansicht des ersten Konzeptes.	29
4.6	Wireframes Mobile: Spielübersicht.	30
4.7	Wireframes Mobile: Detailansicht von Kraftwerk (Variante A).	31
4.8	Wireframes Mobile: Detailansicht von Kraftwerk (Variante B).	31
4.9	Wireframes Mobile: Finanzübersicht.	31
4.10	Wireframes Mobile: Finanzverlauf.	32
4.11	Wireframes Mobile: Aktueller Stromproduktion.	32
4.12	Wireframes Mobile: Kraftwerkliste.	32
4.13	Wireframes Mobile: Forschungsbaum.	33
4.14	Wireframes Mobile: Ressourcen-Übersicht.	33
4.15	Wireframes Mobile: Handelsübersicht.	33
4.16	Skizze: Verschiedene Zustände des Smart TV-Cursors.	34
4.17	Wireframes Smart TV: Aufbau des UIs.	35
4.18	Wireframes Smart TV: Analyse des Mobile Screens von Sim City: Build It.	37
4.19	Wireframes Smart TV: Spielübersicht.	37
4.20	Wireframes Smart TV: Auflistung der Menüstruktur.	38
4.21	Wireframes Smart TV: Baumenü mit Tutorial.	38

4.22	Wireframes Smart TV: Baumenü mit angewähltem Gebäude. . . . .	39
4.23	Wireframes Smart TV: Platzieren eines Gebäudes mit dem Cursor. . . . .	39
4.24	Wireframes: Analyse des PC-Design von Sim City (2013). . . . .	40
4.25	Wireframes: PC-Design mit Übersicht. . . . .	41
4.26	Wireframes: PC-Design mit geöffnetem Baumenü. . . . .	41
4.27	Skizzen: Porträts von baubaren Gebäuden, Teil 1. . . . .	42
4.28	Skizzen: Porträts von baubaren Gebäuden, Teil 2. . . . .	43
4.29	Axure-Prototyp: Übersicht über Spielkarte. . . . .	45
4.30	Axure-Prototyp: Finanzübersicht. . . . .	45
4.31	Axure-Prototyp: Modal zum Abriss eines Gebäudes. . . . .	46
4.32	Axure-Prototyp: Übersicht über Stromnetz. . . . .	46
4.33	Axure-Prototyp: Infofenster eines Kraftwerkes geöffnet. . . . .	47
4.34	Axure-Prototyp: Platzieren eines Kraftwerkes und Graph des Stromnetz. . . . .	47
5.1	Moodboard: Gestaltung von Stadt-Aufbau-Simulatoren [34], [59], [60]. . . . .	49
5.2	Farben: Verwendete Farben im Spiel. . . . .	50
5.3	Typografie: Verwendete Schriftgrößen im Spiel. . . . .	52
5.4	Screendesign für Smart TV-Übersicht. . . . .	54
5.5	Grafische Zustände des Smart TV-Cursors. . . . .	54
5.6	Color Board für die Farbgestaltung des Spiele-Prototypen. . . . .	55
5.7	Kontraste im Screendesign. . . . .	55
5.8	Icons der zweiten Generation für das Interfacedesign. . . . .	56
5.9	Icons der ersten Generation für den Klick-Prototypen. . . . .	57
5.10	Renderings von ausgewählten Gebäuden. . . . .	58
6.1	Unity: Die Framerate eines Spiels wird mit dem Profiler überprüft. . . . .	62
6.2	Unity: Infofenster – zum Test mit <i>XmlLayout</i> erstellt. . . . .	63
6.3	Unity: Spiele-Prototyp eines Kraftwerk-Simulations-Spiels. . . . .	65
6.4	Phaser-Spiel-Prototyp. Normale Ansicht über die Spielwelt. . . . .	68
6.5	Phaser-Spiel-Prototyp. Geöffnetes Baumenü. Die Items in der Liste können in die Spielwelt gedraggt werden. . . . .	68
6.6	Phaser-Spiel-Prototyp. Man sieht ein geöffnetes Modal. Das Geld hat nicht gereicht zum Bau eines Kraftwerkes. . . . .	69
6.7	Phaser-Spiel-Prototyp. Finanzfenster, in dem gerade Geld geliehen wurde. . . . .	69
6.8	Phaser-Spiel-Prototyp. Gerade wird ein Kohlekraftwerk platziert. . . . .	70
6.9	Phaser-Spiel-Prototyp. Infofenster mit detaillierten Informationen über ein Kraftwerk. . . . .	70
6.10	Phaser-Spiel-Prototyp. Das Infofenster des Baumenüs ist geöffnet. . . . .	71
6.11	Phaser-Spiel-Prototyp. Credits der Entwicklung. . . . .	71
7.1	Nutzertest 1: <i>UEQ Benchmark</i> . . . . .	78
7.2	Nutzertest 1: Ergebnisse des UEQ nach Kategorien. . . . .	78

7.3	Nutzertest 2: <i>UEQ Benchmark</i> . . . . .	81
7.4	Nutzertest 2: Ergebnisse des UEQ nach Kategorien. . . . .	81
7.5	Vergleich Nutzertests: Werte für die Kategorien des UEQ. . . . .	83
7.6	Vergleich Nutzertests: SUS-Score. . . . .	83

# Code-Listings

1	XmlLayout: Debug-Panel . . . . .	64
2	Vue.js und Phaser: Setup der Kommunikation. . . . .	66
3	Vue.js und Phaser: Kommunikation zwischen dem Spiel und UI über Events. . . . .	67

---

# Literatur

- [1] J. Ferrara, *Playful design: Creating game experiences in everyday interfaces*. Brooklyn, N.Y: Rosenfeld Media, 2012, ISBN: 9781933820149. Adresse: <https://rosenfeldmedia.com/books/playful-design/> (besucht am 2. März 2020).
- [2] *Deutscher Games-Markt 2019*, PDF, 29.08.2019. Adresse: <https://www.game.de/marktdaten/deutscher-games-markt-2019/> (besucht am 7. Feb. 2020).
- [3] Bundesverband Musikindustrie e.V., *Musikindustrie 2018 in Zahlen*. Adresse: <https://www.musikindustrie.de/publikationen/musikindustrie-in-zahlen-im-ueberblick> (besucht am 7. Feb. 2020).
- [4] Goldmedia GmbH Strategy Consulting, HMS Hamburg Media School GmbH und DIW Econ GmbH, *Wirtschaftliche Bedeutung der Filmindustrie in Deutschland: Studie im Auftrag des Bundesministeriums für Wirtschaft und Energie*, Berlin, 2017. Adresse: [https://www.bmwi.de/Redaktion/DE/Publikationen/Studien/bedeutung-filmindustrie.pdf?\\_\\_blob=publicationFile&v=12](https://www.bmwi.de/Redaktion/DE/Publikationen/Studien/bedeutung-filmindustrie.pdf?__blob=publicationFile&v=12) (besucht am 7. Feb. 2020).
- [5] A. Kunst, *Video game preferences by genre in Germany 2019 | Statista*, 2019. Adresse: <https://www.statista.com/forecasts/998792/video-game-preferences-by-genre-in-germany> (besucht am 26. Feb. 2020).
- [6] J. Radtke und N. Kersting, Hrsg., *Energiewende: Politikwissenschaftliche Perspektiven*, Ser. Energietransformation. Wiesbaden: Springer VS, 2018, ISBN: 9783658215613.
- [7] J. Jacobsen und L. Meyer, *Praxisbuch Usability & UX*, 2. aktualisierte Auflage und erweiterte Auflage. 2019, ISBN: 3836269538.
- [8] dpa, “Warnung vor massivem Stromverbrauch durch Internetnutzung: Klimaneutralität der EU”, *Neue Osnabrücker Zeitung*, Jg. 2019, 15.12.2019. Adresse: <https://www.noz.de/deutschland-welt/politik/artikel/1958347/warnung-vor-massivem-stromverbrauch-durch-internetnutzung> (besucht am 20. Dez. 2019).
- [9] Bundesnetzagentur, *Monitoringbericht Energie 2019*. Bonn: Bundesnetzagentur, Referat 603 und Bundeskartellamt, Arbeitsgruppe Energie-Monitoring, 2019. Adresse: [https://www.bundesnetzagentur.de/SharedDocs/Pressemitteilungen/DE/2019/20191127\\_Monitoringbericht.html](https://www.bundesnetzagentur.de/SharedDocs/Pressemitteilungen/DE/2019/20191127_Monitoringbericht.html) (besucht am 2. März 2020).
- [10] —, *SMARD | Deutschland im Überblick*, 2020. Adresse: <https://smard.de/home/ueberblick/> (besucht am 7. Feb. 2020).

- [11] Umweltbundesamt, *Datenbank "Kraftwerke in Deutschland"*, 2019. Adresse: <https://www.umweltbundesamt.de/dokument/datenbank-kraftwerke-in-deutschland> (besucht am 7. Feb. 2020).
- [12] Verband der Bayerischen Energie- und Wasserwirtschaft e. V, Hrsg., *Wichtige Begriffe für die Energiewende Strom*, München, 2016. Adresse: <https://www.vbew.de/energie/publikationen/> (besucht am 26. Feb. 2020).
- [13] S. Faßbender, Dr. Kübler GmbH, Baader Konzept GmbH und RaumUmwelt-Planungs-GmbH, *Juraleitung: Ersatzneubau 380-kV-Leitung: Raitersaich–Altheim*, Bayreuth, 2019. Adresse: [https://www.tennet.eu/fileadmin/user\\_upload/Our\\_Grid/Onshore\\_Germany/juraleitung/Trassenverlauf/Karten\\_Mai\\_2019/Abschnitt\\_B/GESAMT\\_Blatt5.pdf](https://www.tennet.eu/fileadmin/user_upload/Our_Grid/Onshore_Germany/juraleitung/Trassenverlauf/Karten_Mai_2019/Abschnitt_B/GESAMT_Blatt5.pdf) (besucht am 6. Feb. 2020).
- [14] Bundestag, *Gesetz über den Bundesbedarfsplan: BBPlG*, 27.07.2013. Adresse: <http://www.gesetze-im-internet.de/bbplg/> (besucht am 6. Feb. 2020).
- [15] F. Ambros, *Die Stromautobahn: Das Informationsportal gegen unnötige Stromtrassen*, 2020. Adresse: <https://www.stromautobahn.de/> (besucht am 26. Feb. 2020).
- [16] M. Quanz, V. Papenhagen-Stannick, S. Lemke, R. Bethmann und B. Ammon, *Bundesverband der Bürgerinitiativen gegen SuedLink*. Adresse: <http://bundesverband-gegen-suedlink.de/> (besucht am 26. Feb. 2020).
- [17] Donaukurier, “Gegner der Juraleitung formieren sich: Am Sonntag wird in Dietfurt eine Bürgerinitiative gegen die Monstertrasse gegründet”, *Donaukurier*, 02.10.2018. Adresse: <https://www.donaukurier.de/lokales/riedenburg/wochennl402018-Gegner-der-Juraleitung-formieren-sich;art602,3935336> (besucht am 6. Feb. 2020).
- [18] Bundesnetzagentur, *BBPlG, Vorhaben 41: Raitersaich – Ludersheim – Sittling – Altheim*, 2019. Adresse: <https://www.netzausbau.de/leitungsvorhaben/bbplg/41/de.html> (besucht am 6. Feb. 2020).
- [19] M. Faltlhauser, *Zahlen und Fakten zur Stromversorgung in Deutschland 2018*, Wirtschaftsbeirat Bayern, Hrsg., München, 2018. Adresse: <https://www.wbu.de/positionen/publikationen/> (besucht am 26. Feb. 2020).
- [20] R. Ramadan und Y. Widayani, “Game development life cycle guidelines”, in *ICACSI 2013*, S. 95–100. DOI: 10.1109/ICACSI.2013.6761558.
- [21] H. M. Chandler, *The game production handbook*, 3rd ed., Ser. Foundations of game development. Burlington, MA: Jones & Bartlett Learning, 2014, ISBN: 9781449688097. Adresse: <http://proquest.tech.safaribooksonline.de/9781449688097>.
- [22] D. Arsenault, “Video Game Genre, Evolution and Innovation”, in *Eludamos. Journal for Computer Game Culture*. Bd. 3, S. 149–176. Adresse: <https://www.eludamos.org/index.php/eludamos/article/view/65> (besucht am 26. Feb. 2020).
- [23] Chris Zukowski, *How Steam users see your game*, Gamasutra, Hrsg., 2019. Adresse: [https://www.gamasutra.com/blogs/ChrisZukowski/20190906/350248/How\\_Steam\\_users\\_see\\_your\\_game.php](https://www.gamasutra.com/blogs/ChrisZukowski/20190906/350248/How_Steam_users_see_your_game.php) (besucht am 1. Jan. 2020).

- 
- [24] Valve Corporation, *Steam: Die Ultimative Online Gaming Plattform*, 2020. Adresse: <https://store.steampowered.com/about/> (besucht am 7. Feb. 2020).
- [25] S. Galyonkin, *Your target audience doesn't exist: Why you shouldn't talk about "MOBA audience", "core gamers", "female gamers" and instead think smaller*. 2015. Adresse: <https://galyonk.in/your-target-audience-doesn-t-exist-999b78aa77ae> (besucht am 7. Feb. 2020).
- [26] N. Yee, *Gamer Motivation Model: Reference Sheets & Details (v2)*, Quantic Foundry, Hrsg., 2019. Adresse: <https://quanticfoundry.com/2019/04/11/gdc2019/> (besucht am 18. Feb. 2020).
- [27] —, *The Gamer Motivation Profile: Model and Findings: YouTube*, 2015. Adresse: <https://www.youtube.com/watch?v=gDvaBpfs11A> (besucht am 18. Feb. 2020).
- [28] Quantic Foundry, *Video Game Recommendation Engine*. Adresse: <https://apps.quanticfoundry.com/recommendations/gamerprofile/videogame/> (besucht am 19. Feb. 2020).
- [29] C. Moser, *User Experience Design: Mit erlebniszentrierter Softwareentwicklung zu Produkten, die begeistern*, Ser. X.media.press. Berlin: Springer Vieweg, 2012, ISBN: 9783642133633.
- [30] C. Hodent, *The gamer's brain: How neuroscience and UX can impact video game design*, Ser. A Focal press book. Boca Raton, FL, London und New York: CRC Press Taylor & Francis Group, 2018, ISBN: 9781138089969.
- [31] J. Vašek, *Woman Smartphone Chatting - Free photo on Pixabay*, pixabay.com, Hrsg., 2014. Adresse: <https://pixabay.com/photos/woman-smartphone-chatting-girl-410320/> (besucht am 25. Feb. 2020).
- [32] Vitabello, *Game Man Gamer - Free photo on Pixabay*, pixabay.com, Hrsg., 2019. Adresse: <https://pixabay.com/photos/game-man-gamer-playing-player-4479445/> (besucht am 25. Feb. 2020).
- [33] Pexels, *Adult Casual Caucasian - Free photo on Pixabay*, pixabay.com, Hrsg., 2016. Adresse: <https://pixabay.com/photos/adult-casual-caucasian-glasses-1851571/> (besucht am 25. Feb. 2020).
- [34] Electronic Arts, *SimCity*, Electronic Arts und Maxis, Hrsg., 2014. Adresse: <https://www.ea.com/de-de/games/simcity/simcity> (besucht am 6. Feb. 2020).
- [35] D. Hopkins, *Maxis Sim Game Requirements*, 2012. Adresse: <https://donhopkins.com/home/TheSimsDesignDocuments/> (besucht am 5. Feb. 2020).
- [36] D. Callele, E. Neufeld und K. Schneider, "Requirements Engineering and the Creative Process in the Video Game Industry", in *IEEE RE'05*, S. 240–250. DOI: 10.1109/RE.2005.58.
- [37] *Die beliebtesten Spiele-Plattformen der Deutschen: Stand 2018*, PNG, 16.07.2019. Adresse: <https://www.game.de/marktdaten/die-beliebtesten-spiele-plattformen-der-deutschen/> (besucht am 23. Feb. 2020).

- [38] DIN-Normenausschuss Ergonomie (NAErg), Ergonomics Standards Committee, *Ergonomie der Mensch-System-Interaktion - Teil 410: Gestaltungskriterien für physikalische Eingabegeräte (ISO 9241-410:2008 + Amd.1:2012); Deutsche Fassung EN ISO 9241-410:2008 + A1:2012*, Berlin, 2012-12-00.
- [39] E. Albassam und H. Gomaa, "Applying software product lines to multiplatform video games", in *2013 3rd International Workshop on Games and Software Engineering - Engineering Computer Games to Enable Positive, Progressive Change (GAS 2013)*, Piscataway, NJ: IEEE, 2013, S. 1–7, ISBN: 978-1-4673-6263-4. DOI: 10.1109/GAS.2013.6632582. Adresse: <https://ieeexplore.ieee.org/document/6632582> (besucht am 26. Feb. 2020).
- [40] Amazon.com Inc, *Device Specifications: Fire TV Stick | Amazon Fire TV*. Adresse: <https://developer.amazon.com/de/docs/fire-tv/device-specifications-fire-tv-stick.html?v=ftvstickgen2> (besucht am 27. Feb. 2020).
- [41] inside-intermedia Digital GmbH, *Nokia 8 Datenblatt | alle technischen Daten*, 2017. Adresse: <https://www.inside-digital.de/handys/nokia-8#datasheet-headline> (besucht am 29. Feb. 2020).
- [42] DisplaySpecifications.com, *24" Asus VS248H - Specifications*, 2015. Adresse: <https://www.displayspecifications.com/en/model/c00b97b> (besucht am 29. Feb. 2020).
- [43] Producto AG, *Sharp Aquos LC-46LE542E im Test | Testberichte.de*, 2012. Adresse: <https://www.testberichte.de/p/sharp-tests/aquos-lc-46le542e-testbericht.html> (besucht am 29. Feb. 2020).
- [44] Amazon.com Inc, *Getting Started with Web Apps | Amazon Fire TV*. Adresse: <https://developer.amazon.com/de/docs/fire-tv/getting-started-with-web-apps.html> (besucht am 27. Feb. 2020).
- [45] Entertainment Software Association, *Essential Facts About the Computer and Video Game Industry: 2019*, 2019. Adresse: <https://www.theesa.com/esa-research/2019-essential-facts-about-the-computer-and-video-game-industry/> (besucht am 7. Feb. 2020).
- [46] O. Castendyk und J. Müller-Lietzkow, *Die Computer- und Videospiegelindustrie in Deutschland: Daten – Fakten – Analysen*. Hamburg, 2017. Adresse: [http://www.hamburgmediaschool.com/fileadmin/user\\_upload/Dateien/Forschung/FoKo/Abschlussbericht\\_Games-Studie\\_V1.2\\_2017-12-05.pdf](http://www.hamburgmediaschool.com/fileadmin/user_upload/Dateien/Forschung/FoKo/Abschlussbericht_Games-Studie_V1.2_2017-12-05.pdf) (besucht am 27. Okt. 2019).
- [47] T. Wijam, *The Global Games Market Will Generate \$152.1 Billion in 2019 as the U.S. Overtakes China as the Biggest Market | Newzoo*, 2019. Adresse: <https://newzoo.com/insights/articles/the-global-games-market-will-generate-152-1-billion-in-2019-as-the-u-s-overtakes-china-as-the-biggest-market/> (besucht am 18. Juni 2020).

- [48] T. Fritsch, B. Voigt und J. Schiller, *Evaluation of Input Options on Mobile Gaming Devices*. 2008. Adresse: [https://www.researchgate.net/profile/Jochen\\_Schiller/publication/228460961\\_Evaluation\\_of\\_Input\\_Options\\_on\\_Mobile\\_Gaming\\_Devices/links/54fdd6760cf20700c5ec0ffd/Evaluation-of-Input-Options-on-Mobile-Gaming-Devices.pdf](https://www.researchgate.net/profile/Jochen_Schiller/publication/228460961_Evaluation_of_Input_Options_on_Mobile_Gaming_Devices/links/54fdd6760cf20700c5ec0ffd/Evaluation-of-Input-Options-on-Mobile-Gaming-Devices.pdf) (besucht am 23. Feb. 2020).
- [49] J. Biesterfeldt, *transline Deutschland: Usability – Im Zweifel für den Anwender*, 2005-03-17. Adresse: <http://web.archive.org/web/20090131020247/http://transline.de/transline-tecNews/usability-im-im-zweifel-fuer-den-anwender> (besucht am 23. Feb. 2020).
- [50] real GmbH, *PS4 Dualshock Wireless Controller Controller | real*. Adresse: <https://www.real.de/product/315381287/> (besucht am 25. Feb. 2020).
- [51] —, *Nintendo Switch Pro Controller Controller | real*. Adresse: <https://www.real.de/product/316865252/> (besucht am 25. Feb. 2020).
- [52] —, *Microsoft Xbox Controller + Wireless Adapter for | real*. Adresse: <https://www.real.de/product/323838621/> (besucht am 25. Feb. 2020).
- [53] remote-control outlet, *New DR49WK Voice Remote Control For 2nd Gen Amazon Fire TV Stick Media Player | eBay*, ebay.com, Hrsg., 2017. Adresse: <https://www.ebay.com/itm/New-DU3560-Voice-Remote-Control-For-1st-Gen-Amazon-Fire-TV-Stick-Media-Player-/222384709176> (besucht am 25. Feb. 2020).
- [54] statcounter, *Desktop Screen Resolution Stats Germany: Dec 2018 - Dec 2019*, Ronnie Simpson Consulting, Hrsg., 2020. Adresse: <https://gs.statcounter.com/screen-resolution-stats/desktop/germany> (besucht am 6. Jan. 2020).
- [55] ISO/IEC/IEEE, *International Standard - Systems & software engineering–Vocabulary*, Piscataway, NJ, USA, 2017. DOI: 10.1109/IEEESTD.2017.8016712. Adresse: <https://ieeexplore.ieee.org/document/8016712> (besucht am 26. Feb. 2020).
- [56] SmartBear Software, *Introduction - Cucumber Documentation*, 2020. Adresse: <https://cucumber.io/docs/guides/overview/> (besucht am 13. Feb. 2020).
- [57] —, *Gherkin Reference - Cucumber Documentation*, 2020. Adresse: <https://cucumber.io/docs/gherkin/reference/> (besucht am 13. Feb. 2020).
- [58] M. Federoff, *Heuristics and Usability Guidelines for the Creation and Evaluation of Fun in Video Games*. Bloomington, 2002. Adresse: <https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.89.8294> (besucht am 23. Feb. 2020).
- [59] TrackTwenty, *SimCity: BuildIt*, 2014. Adresse: [https://play.google.com/store/apps/details?id=com.ea.game.simcitymobile\\_row](https://play.google.com/store/apps/details?id=com.ea.game.simcitymobile_row) (besucht am 2. März 2020).
- [60] Hudson Soft, *SimCity Creator*, 2008.
- [61] T. Stapelkamp, *Informationsvisualisierung: Web - Print - Signaletik ; erfolgreiches Informationsdesign: Leitsysteme, Wissensvermittlung und Informationsarchitektur*, Ser. X.media.press. Berlin und Heidelberg: Springer, 2013, ISBN: 9783642020759. DOI: 10.1007/978-3-642-02076-6.

- [62] Google Fonts, *Nunito - Google Fonts*. Adresse: <https://fonts.google.com/specimen/Nunito?selection.family=Nunito> (besucht am 24. Feb. 2020).
- [63] T. Stapelkamp, *Interaction- und Interfacedesign: Web-, Game-, Produkt- und Servicedesign Usability und Interface als Corporate Identity*, Ser. X.media.press. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2010, ISBN: 9783642020735. DOI: 10.1007/978-3-642-02074-2.
- [64] B. Preim und R. Dachzelt, *Interaktive Systeme: Band 1: Grundlagen, Graphical User Interfaces, Informationsvisualisierung*, Ser. eXamen.press. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2010, ISBN: 9783642054013. DOI: 10.1007/978-3-642-05402-0. Adresse: <http://dx.doi.org/10.1007/978-3-642-05402-0>.
- [65] L. Caroux und K. Isbister, “Influence of head-up displays’ characteristics on user experience in video games”, *International Journal of Human-Computer Studies*, Jg. 87, S. 65–79, 2016. DOI: 10.1016/j.ijhcs.2015.11.001. Adresse: <https://www.sciencedirect.com/science/article/abs/pii/S1071581915001779> (besucht am 22. Feb. 2020).
- [66] Molotov85, *2d Isometric tile pack | GameDev Market*, GameDev Network Limited., Hrsg., 2016. Adresse: <https://www.gamedevmarket.net/asset/2d-isometric-tile-pack-6970/> (besucht am 25. Feb. 2020).
- [67] Humble Bundle, *Humble RPG Game Dev Bundle*, 2018. Adresse: <https://www.humblebundle.com/software/rpg-game-dev-bundle> (besucht am 25. Feb. 2020).
- [68] Unity Technologies, *Unity Technologies and NXP Collaborate to Enable Human Machine Interface Development for Immersive, High Fidelity Content in Cars*, Las Vegas, 2020. Adresse: <https://unity3d.com/company/public-relations/news/unity-technologies-and-nxp-collaborate-enable-human-machine-interface> (besucht am 8. Feb. 2020).
- [69] B. Cowan und B. Kapralos, “A Survey of Frameworks and Game Engines for Serious Game Development”, in *2014 IEEE 14th International Conference on Advanced Learning Technologies*, S. 662–664. DOI: 10.1109/ICALT.2014.194.
- [70] Unity Technologies, *Unity Public Relations*, 2020. Adresse: <https://unity3d.com/public-relations> (besucht am 8. Feb. 2020).
- [71] —, *Unity - Manual: UIElements Developer Guide*, 2019. Adresse: <https://docs.unity3d.com/Manual/UIElements.html> (besucht am 9. Feb. 2020).
- [72] Unity Forum, *Scene Merging - Unity Forum*, 2017. Adresse: <https://forum.unity.com/threads/scene-merging.453901/> (besucht am 9. Feb. 2020).
- [73] Unity, *Building UI for games with the new UI Builder - Unite Copenhagen: YouTube*, 2019. Adresse: <https://www.youtube.com/watch?v=t4tfgI1XvGs> (besucht am 9. Feb. 2020).

- 
- [74] Digital Legacy Games, *Digital Legacy - XmlLayout Documentation*, 2020. Adresse: <http://www.digital-legacy.co.za/XmlLayout/Documentation> (besucht am 9. Feb. 2020).
- [75] —, *XmlLayout: Xml-driven UI Framework - Asset Store*, 2020. Adresse: <https://assetstore.unity.com/packages/tools/gui/xmllayout-xml-driven-ui-framework-61090> (besucht am 9. Feb. 2020).
- [76] Mozilla Developer Network, *Type selectors - CSS: Cascading Style Sheets | MDN*, 2019. Adresse: [https://developer.mozilla.org/en-US/docs/Web/CSS/Type\\_selectors](https://developer.mozilla.org/en-US/docs/Web/CSS/Type_selectors) (besucht am 10. Feb. 2020).
- [77] E. You, *Introduction - Vue.js*, 2020. Adresse: <https://vuejs.org/v2/guide/> (besucht am 11. Feb. 2020).
- [78] Stack Overflow, *Stack Overflow Developer Survey 2019*, 2019. Adresse: [https://insights.stackoverflow.com/survey/2019#technology-\\_\\_-most-loved-dreaded-and-wanted-web-frameworks](https://insights.stackoverflow.com/survey/2019#technology-__-most-loved-dreaded-and-wanted-web-frameworks) (besucht am 11. Feb. 2020).
- [79] J. Silge, *The 2019 Stack Overflow Developer Survey Results Are In*, Stack Overflow, Hrsg., 2019. Adresse: <https://stackoverflow.blog/2019/04/09/the-2019-stack-overflow-developer-survey-results-are-in/> (besucht am 11. Feb. 2020).
- [80] R. Davey, *Making your first Phaser 3 game: Part 1 - Introduction - Learn - Phaser*, Photon Storm Ltd., Hrsg., 2018. Adresse: <http://phaser.io/tutorials/making-your-first-phaser-3-game/part1> (besucht am 11. Feb. 2020).
- [81] E. You, *Reactivity in Depth - Vue.js*, 2020. Adresse: <https://vuejs.org/v2/guide/reactivity.html> (besucht am 11. Feb. 2020).
- [82] M. Schrepp, *User Experience Questionnaire Handbook*, 2019. Adresse: <https://www.ueq-online.org/Material/Handbook.pdf> (besucht am 14. Feb. 2020).
- [83] U.S. Department of Health & Human Services, *System Usability Scale (SUS) | Usability.gov*, 2018. Adresse: <https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html> (besucht am 14. Feb. 2020).
- [84] Global Game Jam Inc., *About | Global Game Jam*. Adresse: <https://globalgamejam.org/about> (besucht am 4. März 2020).
- [85] J. Sauro, *5 Ways to Interpret a SUS Score*, MeasuringU, Hrsg., Denver, Colorado, 2018. Adresse: <https://measuringu.com/interpret-sus-score/> (besucht am 16. Feb. 2020).

## Anhang A

# Datenträger

Im Rahmen der vorliegenden Bachelorarbeit sind eine Reihe von Dateien entstanden. Sie wurden auf dem beiliegenden USB-Stick gespeichert. Dabei handelt es sich um folgende Dateien:

<b>Ordner/Datei</b>	<b>Beschreibung</b>
• Schriftliche Arbeit.pdf	Vorliegende schriftliche Arbeit
• Icondesign/	Icons v1 und v2
• Probandentests/	Testbogen 2 und Sprachaufnahmen aus Test 1 und 2
• Recherche-Screenshots/	Screenshots zu drei SimCity-Titeln
• Screendesign/	Smart TV-Screendesign und Cursor-Gestaltung
• Spieleprototypen/	Rohdaten von Axure-, Unity- & Vue/Phaser-Prototyp
• Spieleprototypen-Assets/	Gebäude-Porträts, Gebäude-Render und Tilesets
• Spieleprototypen-Screenshots/	Screenshots zu den Spiele-Prototypen
• Wireframes/	Wireframes für Mobil, Smart TV, PC und Vorskizzen



## Probandentest 2 Power-Prototyp

Probandentest zur Untersuchung der Usability des  
Prototypen für ein Kraftwerk-Simulationsspiel.

<b>Test-Utensilien</b>	<b>Ablauf</b>
Laptop (Sprachaufzeichnung)	Vorstellen der Bachelorarbeit
Smartphone (Testgerät)	Hinweis auf Sprachaufzeichnung
Tablet (Testgerät)	Datenschutzerklärung
Fragebogen	Probandendaten und Nutzungsverhalten
Prototyp	Use Cases an Testgerät
	Standardisierte Fragebögen SUS & UEQ

Getestet während des  
**Global Game Jam 31.01.–02.02. in Ansbach**

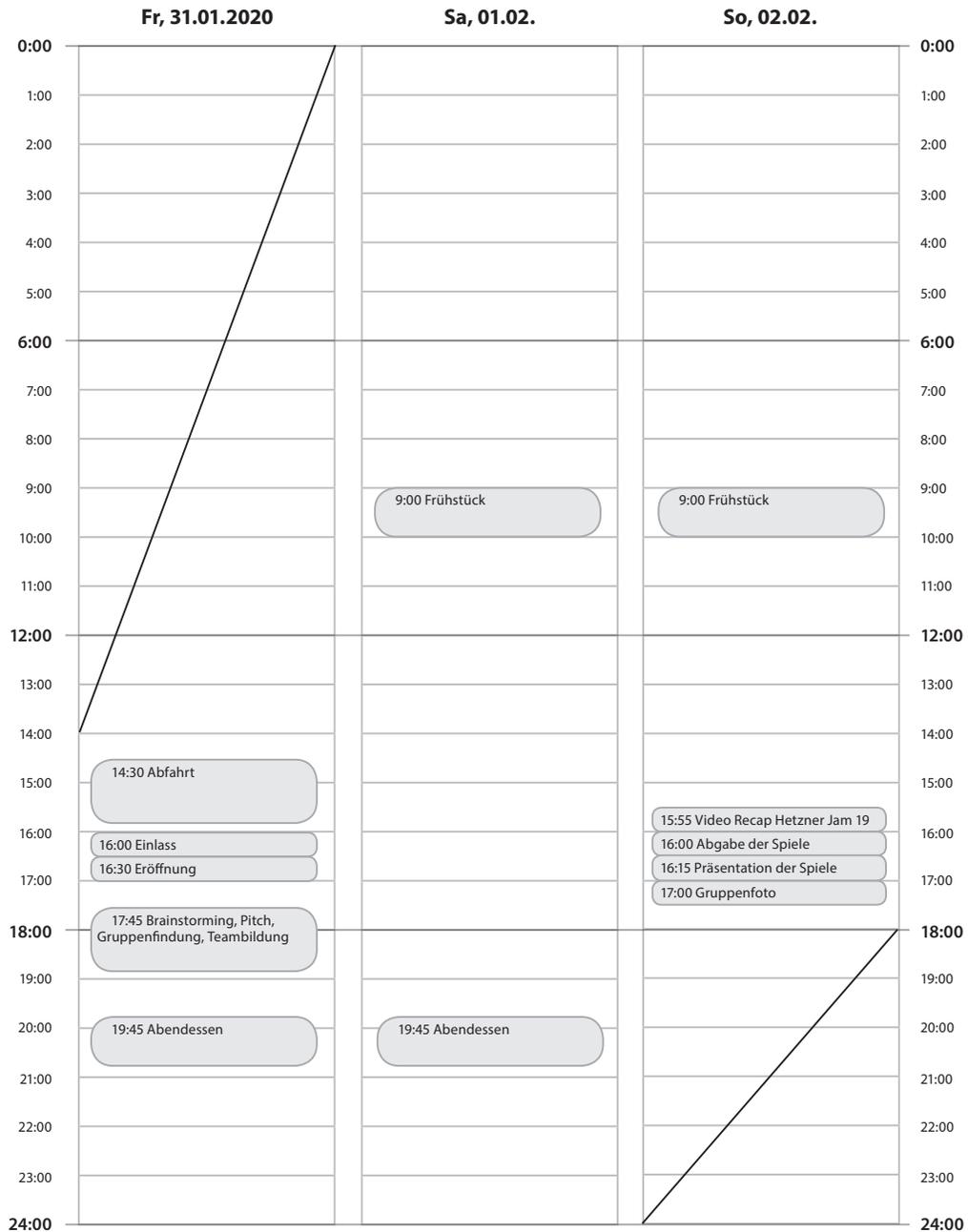


Technische Hochschule Ingolstadt  
User Experience Design  
31. Januar 2020

**Lukas Semmler**

  
A7

# Zeitplan



⚡  
A2

## Datenschutzerklärung

Die Daten zu dieser Nutzerstudie werden im Rahmen der Bachelorarbeit zum Thema

*„Anwenden von UX-Design-Methoden zur Konzeption eines Videospiele  
anhand des Beispiels einer Stromnetz-Wirtschafts-Simulation“*

von Lukas Semmler (Technische Hochschule Ingolstadt / Studiengang User Experience Design B. Sc.) erhoben.

Mit Hilfe dieser Studie wird untersucht, inwieweit der vorliegende Entwurf eines Spiele-UIs benutzbar ist. Dabei werden die Daten anonym und nicht personenbezogen mit Hilfe von Fragebogen und Sprachaufzeichnung erhoben. Die Löschung kann jederzeit bei Lukas Semmler beantragt werden.

Die von Ihnen gemachten Angaben werden ausschließlich im Rahmen der Bachelorarbeit von Lukas Semmler verwendet.

Betreuende Professorin: **Prof. Ingrid Stahl**,

Co-Betreuung: **Mathias Haimerl**

Ich habe diesen Hinweis gelesen und bin mit der anonymisierten Erhebung meiner Daten einverstanden.

\_\_\_\_\_  
Ort, Datum

\_\_\_\_\_  
Unterschrift



## Probandendaten und Nutzungsverhalten

Proband-ID

Alter

Aktueller Beruf /  
Ausbildung

Geschlecht

Spielerfahrung

Anfänger / Fortgeschritten / Pro-Gamer

Spielhäufigkeit

Wie oft pro Woche

Lieblingsspielgenre

Action-RPG

Aufbau-  
Simulation

Fighting

Grand Strategy

Idle game

MMORPG

MOBA

Mobile game

Party Game

Platformer

Racing

Rhythm

Roguelikes

RTS

Shooter

Sports

Stealth

Survival

Survival-Horror

Tower Defense

Visual Novel

Nutzung Smartphone

Wie oft pro Woche /

Daheim, Unterwegs, in Arbeit

Nutzung Smart TV

Wie oft pro Woche /

Daheim, Unterwegs, in Arbeit

Nutzung Laptop/PC

Wie oft pro Woche /

Daheim, Unterwegs, in Arbeit



2

## Use Cases

1. Sieh dich auf der Karte um.  
Gibt es jemanden, der gerade  
besonders glücklich oder  
genervt ist?

2. Finde heraus, wie viel es  
kostet, einen Bahnhof zu  
bauen.

3. Baue ein Kohlekraftwerk.

4. Für den Bau eines weiteren  
Kohlekraftwerkes hast du  
nicht genug Geld. Nimm  
einen Kredit auf.

5. Finde heraus, bei welchem  
Prozentsatz die Strom-  
Versorgungssicherheit liegt.

6. Verschiebe das gebaute  
Kraftwerk.

Sonstige Notizen



## UEQ

Bitte gib nun Deine Einschätzung des Spiels ab.  
Kreuze bitte nur ein Kästchen pro Zeile an.

▼

unerfreulich	<input type="checkbox"/>	erfreulich	1						
unverständlich	<input type="checkbox"/>	verständlich	2						
kreativ	<input type="checkbox"/>	phantasielos	3						
leicht zu lernen	<input type="checkbox"/>	schwer zu lernen	4						
wertvoll	<input type="checkbox"/>	minderwertig	5						
langweilig	<input type="checkbox"/>	spannend	6						
uninteressant	<input type="checkbox"/>	interessant	7						
unberechenbar	<input type="checkbox"/>	voraussagbar	8						
schnell	<input type="checkbox"/>	langsam	9						
originell	<input type="checkbox"/>	konventionell	10						
behindernd	<input type="checkbox"/>	unterstützend	11						
gut	<input type="checkbox"/>	schlecht	12						
kompliziert	<input type="checkbox"/>	einfach	13						
abstoßend	<input type="checkbox"/>	anziehend	14						
herkömmlich	<input type="checkbox"/>	neuartig	15						
unangenehm	<input type="checkbox"/>	angenehm	16						
sicher	<input type="checkbox"/>	unsicher	17						
aktivierend	<input type="checkbox"/>	einschläfernd	18						
erwartungskonform	<input type="checkbox"/>	nicht erwartungskonform	19						
ineffizient	<input type="checkbox"/>	effizient	20						
übersichtlich	<input type="checkbox"/>	verwirrend	21						
unpragmatisch	<input type="checkbox"/>	pragmatisch	22						
aufgeräumt	<input type="checkbox"/>	überladen	23						
attraktiv	<input type="checkbox"/>	unattraktiv	24						
sympathisch	<input type="checkbox"/>	unsympathisch	25						
konservativ	<input type="checkbox"/>	innovativ	26						



## SUS

Aussage	Stimme gar nicht zu	Stimme voll zu
Ich kann mir sehr gut vorstellen, das System regelmäßig zu nutzen.	<input type="radio"/>	<input type="radio"/>
Ich empfinde das System als unnötig komplex.	<input type="radio"/>	<input type="radio"/>
Ich empfinde das System als einfach zu nutzen.	<input type="radio"/>	<input type="radio"/>
Ich denke, dass ich technischen Support brauchen würde, um das System zu nutzen.	<input type="radio"/>	<input type="radio"/>
Ich finde, dass die verschiedenen Funktionen des Systems gut integriert sind.	<input type="radio"/>	<input type="radio"/>
Ich finde, dass es im System zu viele Inkonsistenzen gibt.	<input type="radio"/>	<input type="radio"/>
Ich kann mir vorstellen, dass die meisten Leute das System schnell zu beherrschen lernen.	<input type="radio"/>	<input type="radio"/>
Ich empfinde die Bedienung als sehr umständlich.	<input type="radio"/>	<input type="radio"/>
Ich habe mich bei der Nutzung des Systems sehr sicher gefühlt.	<input type="radio"/>	<input type="radio"/>
Ich musste eine Menge Dinge lernen, bevor ich mit dem System arbeiten konnte.	<input type="radio"/>	<input type="radio"/>