

GAME DESIGN DOKUMENT



Entwicklung eines Lernspiels, das grundlegende ***Konzepte der Rechnernetze*** vermittelt

von

Nierling, Patrick

Schäfer, Nora

Scheinflug, Sarah

Schneider, Nick

Dozent/in: Anne Hamann

Sommersemester 2023

Game Overview

Spielkonzept

In NetWorks lernen die SuS die Grundlagen der Rechnernetze kennen. Dabei kann der Spieler einzelne Computer per Toucheingabe über einen Multitouch-Table mit verschiedenen Netzwerkgeräten verbinden und lernt dabei grundlegende Probleme von Netzwerken zu analysieren und Lösungsansätze auszuprobieren. Die Handlung beginnt zunächst im Wohnzimmer eines Neuntklässlers, den neuen Familiencomputer an das Internet anschließen möchte, um an einem Schulprojekt arbeiten zu können. Nach Abschluss des ersten Levels wechselt der Handlungsort in das Computerkabinett der Schule und danach zu einer schnell wachsenden Firma. Somit kann der Spieler immer größere und komplexere Szenarien der Netzwerktechnologie bearbeiten und lernt dabei stets neue Lösungsansätze kennen. Das dritte und letzte Level des Spiels soll kein definiertes Ende haben, sondern ähnlich wie beispielsweise Mini Motorways oder Mini Metro unbegrenzt skalierbar sein, mit dem Ziel einen möglichst hohen Highscore zu erlangen. Dabei steigen die Schwierigkeit und Geschwindigkeit des Spiels stetig an, sodass hohe Scores exponentiell schwerer zu erreichen sind. Dargestellt wird alles in einer cartoonartigen Grafik mit stark simplifizierten Grafikelementen in einem einfachen Farbschema.

Genre

Das Lernspiel kann in die Genre-Kategorien Puzzle und Strategie eingeordnet werden. Es soll in erster Linie den SuS spielerisch vermitteln, wie Rechner miteinander kommunizieren und welche Komponenten dafür benötigt werden.

Als Lernspiel dient es in erster Linie dazu, den SuS Konzepte und Wissen so unterschwellig zu vermitteln, dass sie ihren Lernfortschritt möglichst nicht bemerken.

Zielgruppe

Für das Lernspiel sind keine speziellen Vorkenntnisse benötigt, diese können jedoch von Nutzen sein. So haben SuS, welche bereits Erfahrungen mit Rechnernetzen und/oder einzelnen Komponenten haben, die Möglichkeit, nach dem Tutorial-Level die Schwierigkeit anzupassen. Es wird drei verschiedene Schwierigkeitsgrade geben, die sich in der Zahl und Art der Hilfestellungen unterscheiden.

Das Lernspiel richtet sich sowohl an SuS ohne Vorkenntnisse als auch an bereits erfahrene. Das Spiel ist für SuS der neunten Klasse gedacht.

Lisa

Die 14-jährige Lisa ist Schülerin an einem Gymnasium. Sie mag shoppen und sich mit ihren Mädels treffen und benutzt ihren Laptop eigentlich nur dafür, Serien zu schauen. Auf ihrem Handy sind ihre Top 3 Apps Instagram, TikTok und Hay Day.

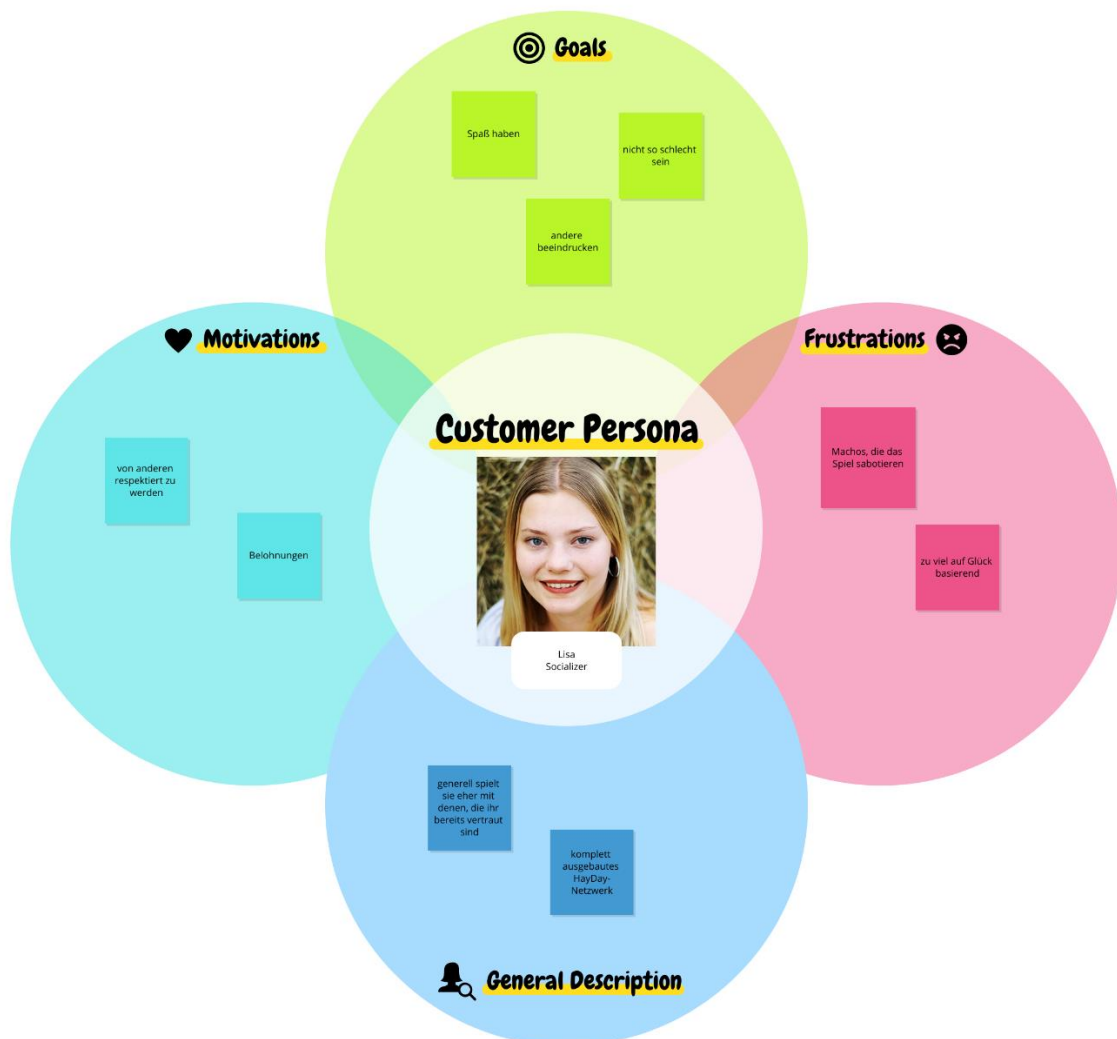


Abbildung 1: Persona für Lisa

Marie

Die ebenfalls 14-jährige Marie schleicht sich nachts immer an den Familienrechner, um IT-Videos zu schauen. Sie ist die Klassenbeste im Informatik-Unterricht und arbeitet sehr

engagiert mit. Ihr liegt viel daran, alle Aufgaben möglichst gut zu erledigen, daher stört es sie extrem, wenn andere SuS nicht ausreichend mitarbeiten.

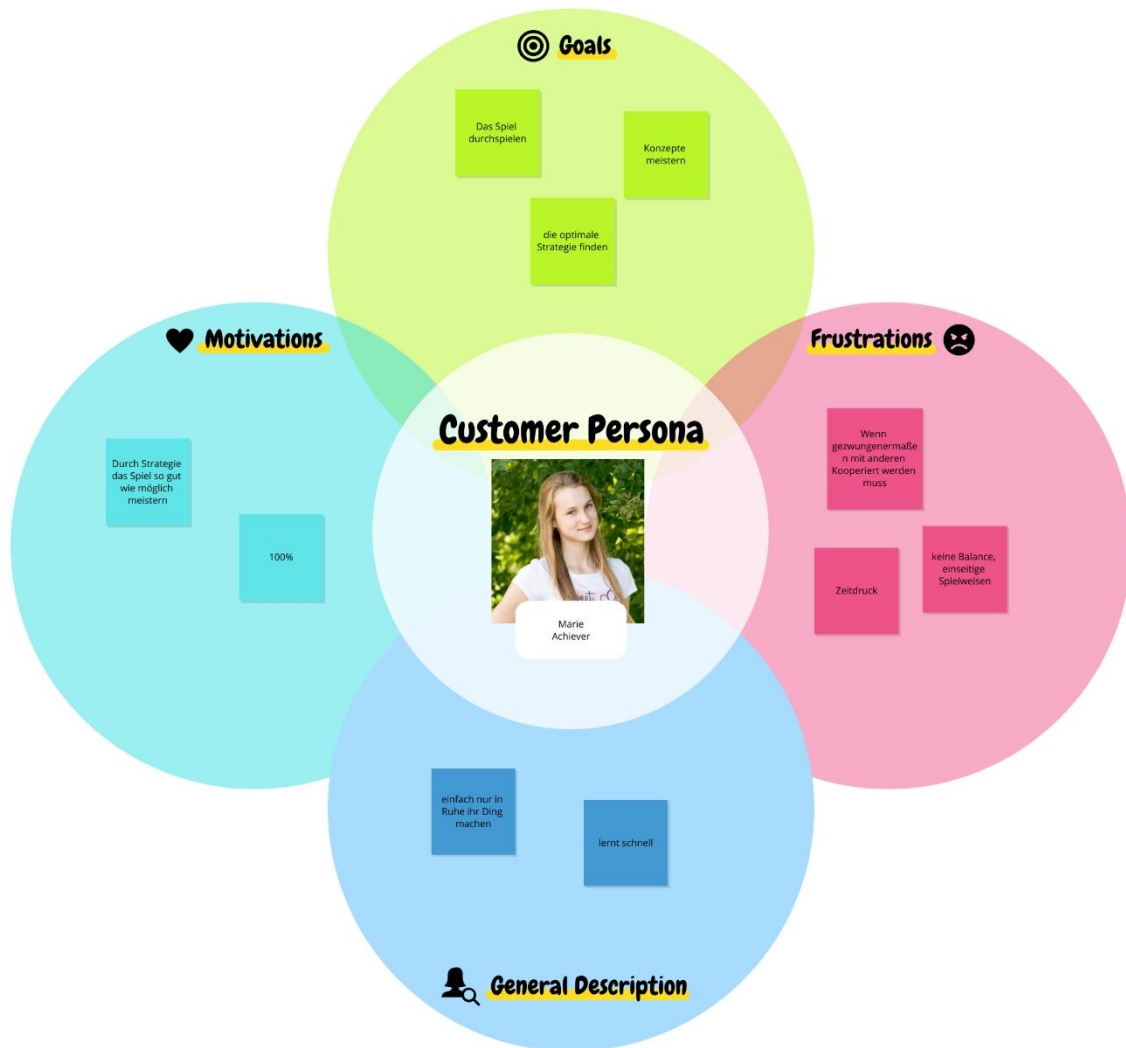


Abbildung 2: Persona für Marie

Paul

Der 15-jährige Paul ist ebenfalls Schüler in der gleichen Klasse wie Lisa. Er ist sehr technikaffin: zusammen mit seinen Kumpels haben sie schon mehrere Rechner zusammengebaut. Auf Lernen hat er keine große Lust, will aber immer der Beste sein. Nach der Schule trifft er sich gerne mit Leuten auf Discord zum Zocken.

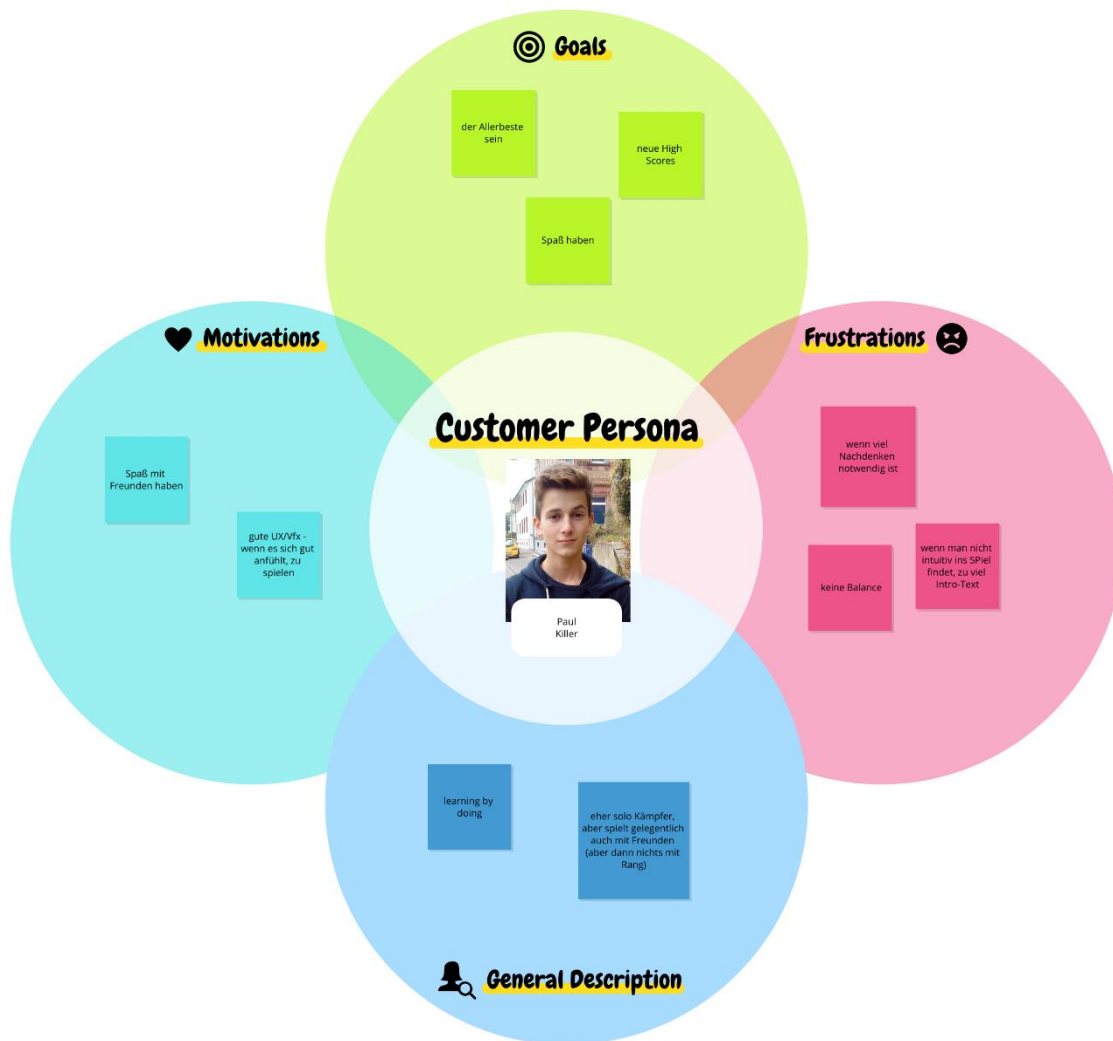


Abbildung 3: Persona für Paul

Quelle der Persona-Schemas: [Concept Board](#)

Pädagogische (Lern-) Ziele

Die Lernenden entwickeln Kompetenzen im Umgang mit der Übertragung von Daten in Netzwerken. Im Verlauf des Spiels erlangen die Lernenden Kenntnis über verschiedene Netzwerkkomponenten und Übertragungsmedien, indem diese zur Problemlösung verwendet werden. Außerdem kennen sie Konzepte wie das Client-Server-Prinzip und Peer-to-Peer - Kommunikation, indem sie diese im Spiel nachbauen können.

Grundlagen der Übertragung von Daten in Netzwerken gemäß Lehrplan Klassenstufe 9 sind wünschenswert, jedoch nicht zum Erzielen der Lernziele notwendig.

Game Flow

Bei einer neuen Spielsitzung starten die SuS in einem Menü, in dem sie zuerst ihre Spieleranzahl zwischen zwei und vier auswählen.

Anschließend erscheinen Auswahlfelder für Level, in welche sich die Benutzer begeben können. Für eine Erstsitzung wird das Tutorial empfohlen.

Im Tutorial werden grundlegende Spielbestandteile wie die zu verbindenden Rechner, Kabel und Router erklärt und Mechaniken wie z.B. unterschiedliche Kabeltypen in Text- oder Bildform erklärt, sodass dieses Level mit Hilfestellungen bewältigt werden kann und die Grundlage für den weiteren Spielfortschritt bildet. Es gibt also ein Intro in die Story, die ersten Computer werden mit vorgegebenen Kabeln verbunden und es gibt noch Hinweise für mögliche Topologien.

In den weiteren Leveln werden Hilfestellungen schrittweise reduziert, um den Spielfluss bei erfolgreich gelernten Mechaniken und Lerninhalten nicht zu unterbrechen, sondern spielerisch zu festigen. In diesen Leveln wird die Schwierigkeit von Level zu Level ein wenig höher, der größte Schwierigkeitsanstieg findet allerdings erst im letzten Level statt.

Das letzte Level ist eine Art Freimodus ohne jegliche Hilfestellungen, wo stetig mehr Bedarf aufkommt, der vernetzt werden muss. Ab da wird es also stetig schwieriger ohne ein konkretes Zeitlimit - das Limit ist hier, dass entweder das Grid voll ist oder die Datenpakete sich anstauen.

Konkret sieht das Ganze also so aus, dass auf einem Grid zufällig neuer Bedarf in Form von Rechnern generiert wird. Parallel dazu werden fast genauso zufällig Router und weitere Netzwerkkomponenten generiert, welche die Schüler so mithilfe von Kabeln oder anderen Übertragungsmedien so verbinden müssen, dass eine valide Verbindung entsteht. Diese hat dann je nach Eigenschaften unterschiedliche Übertragungsraten. Diese muss aber hoch genug sein, um den entstehenden Bedarf als Datenpakete zu übermitteln, ohne dass sich Stau bildet.

Um das Spiel ein wenig abwechslungsreicher im letzten Level zu halten, gibt es auch nach einer festen Zeit ein "Wochenende". Das ist eine Art Verschnaufpause für die SuS, bei der ihnen 3 mögliche Upgrades angeboten werden. Währenddessen wird das Spiel pausiert, sodass die SuS sich in Ruhe beraten können, für welches Upgrade sie sich entscheiden.

Nachdem alle abgestimmt haben und somit ein Upgrade gewählt wurde, steht Ihnen das zur Verfügung und das Spiel geht weiter bis zum nächsten Wochenende oder bis zum Spielende.

Das Spiel ist zu Ende, wenn sich zu viele Datenpakete angestaut haben oder das Grid voll ist. Je nachdem, wie gut und effizient das Netz ausgebaut wurde, erhalten die Spieler dann einen High-Score. Ähnlich wie bei Tetris wird also gespielt, bis es nicht mehr weiter geht.

Look and Feel

Das gewählte Farbschema dient der bestmöglichen Übersicht über alle Komponenten des Lernspiels. Hintergrund und weniger relevante Elemente sind in Grau-Blautönen gehalten, um optische Unruhe zu vermeiden und Fokus auf wenige wichtige Elemente zu legen. Letztere sind durch sich deutlich abhebende Farben optisch abgrenzbar.

Die gesamte Graphik ist in einem 2D-Look gehalten. Icons einzelner Komponenten werden in einem angedeuteten 3D-Stil in einen simplen Hintergrund mit Gridmuster zur Anordnung eingebettet.



Abbildung 4: Moodboard

Quelle: [Figma](#)

Integration Pädagogisches Szenario

Das Lernspiel fügt sich im Lernplan der 9. Klasse in Lernbereich 3: Netzwerke ein. Es werden die gängigsten Netzwerkkomponenten, Übertragungsmedien und das Konzept der Zerlegung in Datenpakete vermittelt. Dabei liegt der Fokus auf den Kabeltypen Kupfer und Glasfaser sowie auf den kabellosen Übertragungsmedien WLAN, Bluetooth und Funk. An Netzwerkkomponenten sollen vor allem Switches, Router und Server. Auch verschiedene Topologien sollten den Schülern bekannt sein, nachdem sie das Spiel eine Weile gespielt haben. Anschließend sollten die SuS die Grundlagen dieser Bereiche verstanden haben und können diese beispielsweise in Form eines Testes nachweisen.

Zudem vermittelt das Lernspiel eine gewisse Menge Teamwork, welches für das gemeinschaftliche Erarbeiten von möglichst guten Lösungen der einzelnen Level benötigt wird.

Gameplay und Mechanik

Gameplay

Aktionen

Elementare Aktionen:

- Verbindungen zwischen Rechnern ziehen
- Verbindungsart auswählen (verschiedene Kabeltypen mit unterschiedlicher Kapazität / Geschwindigkeit)
- 1 Verbindung = 1 Feld
- Upgrade auswählen, Platzierung erfolgt an ausgewählter Stelle (PC, Router, Switch), bzw. im Grid und am Provider

Strategische Aktionen:

- Verbindungen so arrangieren, dass kein Stau entsteht

- entsprechende Übertragungsmedien auswählen (Reichweite, Geschwindigkeit etc. in Betracht ziehen)
- passendes Upgrade wählen und platzieren

Balancing:

- Verschiedene Arten der Verbindungen über versch. Topologien gegeben
- schnellere Kabel sind selten
- Aktion 'Verbindung ziehen' hat Auswirkungen auf Datenfluss
- stetige Erhöhung des Bedarfes, dadurch bessere Kabel notwendig -> sonst Stau, was zu Spielende führen kann
- Upgrades: Internetverträge, die besser werden können, größere Switches, mehr Outputs an Routern, WLAN-Modul für mehr Reichweite (ähnlich Highways bei MiniMotorways), Repeater (selten)
- Kontrollinstanz: Zeit, nach der zwischen verschiedenen Upgrades ausgewählt werden kann, Pausiermodus für Diskussionen
- Strafe: Stau, wenn nicht schnell genug die Verbindungen (gut genug) gezogen
- Kommunikation:
 - über kleine Textfelder werden die SuS angewiesen, an besonders kommunikationsintensiven Stellen (bspw. bei der Upgrade-Wahl) dies auch miteinander zu tun
 - jeder hat seinen eigenen "Quadranten", wenn also eine Verbindung von einem Quadranten in den anderen gezogen werden soll, sollten sich die SuS absprechen; das könnte später über Checks für mehrere Inputs auf der gleichen Stelle gelöst werden

Spielmechanik

Spielraum

Der Spielraum in unserem Spiel ist zweidimensional, da Verbindungen in einem 2D-Grid in alle Richtungen gezogen werden können. Zudem handelt es sich um einen diskreten Spielraum.

Zeit

Unser Lernspiel läuft in kontinuierlicher Zeit ohne Zeitlimit. Ein Zeitlimit macht aufgrund des Spielzieles ähnlich wie bei Tetris keinen Sinn. Stattdessen

gibt es in NetWorks die Stressfaktoren "Platz auf dem Grid" und "Stau der Datenpakete". Diese halten das Spiel getaktet und herausfordernd.

Es gibt allerdings Pausenmodus, in dem diskutiert und Luft geholt werden kann. Dieser kann jederzeit einfach durch einen Knopfdruck aktiviert werden. Bei diesem bleiben Hintergrund und Grid zu sehen und werden nur von einem Layer Grau mit geringer Deckkraft überlagert, sodass der Spielflow so wenig wie möglich unterbrochen wird.

Es gibt auch kein "zu lang", da das das Ziel des Spieles ist, so weit wie möglich zu kommen. Es sollte auch nicht zu langweilig werden, da stetig neue Dinge aufpoppen und es eine Sache des Balancing ist, dass es weder zu viel noch zu wenig fordernd ist.

Wie lange das Spiel am Ende dauert, ist bestimmt auch eine Sache des Balancing und des Fine-Tunings, was während des Entwicklungsprozesses durch Testen ermittelt werden muss. Anhand der Testergebnisse können dann Faktoren wie die Anzahl an Events, aufkommenden Demands und wie schnell Datenpakete dazu kommen variiert werden.

Objekte, Attribute und Statusangaben

Objekte in unserem Spiel sind vor allem die verschiedenen Netzwerkkomponenten. Dies umfasst Computer, Switches, Server und Router. Weiterhin stehen den SuS verschiedene Kabeltypen zur Verfügung. Diese sind unterschiedlich schnell. Es gibt auch drahtlose Verbindungsmöglichkeiten wie WLAN oder BlueTooth, die unterschiedliche Reichweiten und Übertragungsgeschwindigkeiten haben. Alle Objekte sind positioniert an definierten Koordinaten im Grid entlang der x- und y-Achse.

Das sind die Hauptkomponenten unseres Lernspiels:

| Objekt | Variation? | Farbe / sonstige Details | Datenpakete |
|--------|---------------|-----------------------------|-------------------------------------|
| Kabel | langsam | Kupfer-orange | 1 Paket/Einheit |
| | mittel | Limonen-grün-gelb | 5 Pakete/Einheit |
| | schnell | Glasfaser-blau (hell) | 15 Pakete/Einheit |
| PC | universal | grau | 1-10 Pakete/Einheit |
| | Firma (Farbe) | X Bildschirm Firmenfarbe | entsprechend 1-10 Pakete/Einheit |

| | | | |
|-----------|---------------|--------------------------|---------------------------|
| Switch | 4 Ports | schwarz | Maximal-Durchsatz |
| Repeater | 0 ports | erhöht Wireless-Radius | |
| Router | 1 in, 3 out | | |
| Server | 1 in, 7 out | | |
| universal | 4 Felder groß | schnell | |
| | Firma X | Farbe entsprechend Firma | noch schneller |
| Provider | 1 out | | Durchsatz je nach Vertrag |

Side Note: Die Datenpakete-Werte sind subject-to-change und werden in der Entwicklung herausgearbeitet.

Switches:

- 4 Ports, davon ein zufälliger Input
- 1 Gridfeld groß
- Nimmt Farbe des Inputs an

PC:

- 1 Gridfeld groß
- 1 Port Input
- (Können später mehrere Farben haben)
- Farbige PCs nur an farbige Ports
- Graue (farblose) an alle Farben anschließbar

Provider:

- fester Output
- PC nicht direkt anschließbar

Kabel:

- Länge über Kabelgrid begrenzt
- Verschiedene Typen haben verschiedene Maximallängen

- Optisch z.B. unterschiedlich dick
- zusätzlich durch andere Farbe / Textur unterscheidbar

wireless:

- ähnlich Kabel, nur mit Wirkungsbereich
- verschiedene Typen wie WLAN, BlueTooth, ...

Kleiner Router:

- 1 Gridfeld groß
- 1 Input
- 3 Outputs

Repeater:

- erhöhen wireless Verbindungsradius

Server:

- 4 Gridfelder groß
- 1 Input
- 7 Outputs
- Outputfarben: nur gleichfarbige PCs anschließbar

Die Farben sind nur für spätere zusätzliche Schwierigkeit da.

Ein Gerät kann entweder angeschlossen sein oder nicht. Sobald ein Kabel ordnungsgemäß angeschlossen wird, ändert sich der Status des Gerätes an dem Port zu "verbunden".

Viele der Informationen werden den SuS preisgegeben, weil das dem Verständnis hilft. Sachen wie welche Geräte schon verbunden sind und welche nicht, wie viel sich anstaut, welche Felder des Grids schon belegt sind etc. Diese sind auch für alle Spielenden sichtbar.

(Port an einem) Gerät:

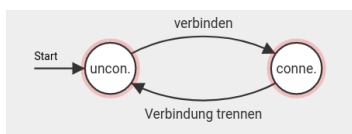


Abbildung 5: Automatendiagramm einer Komponente

Quelle: [flaci](#)

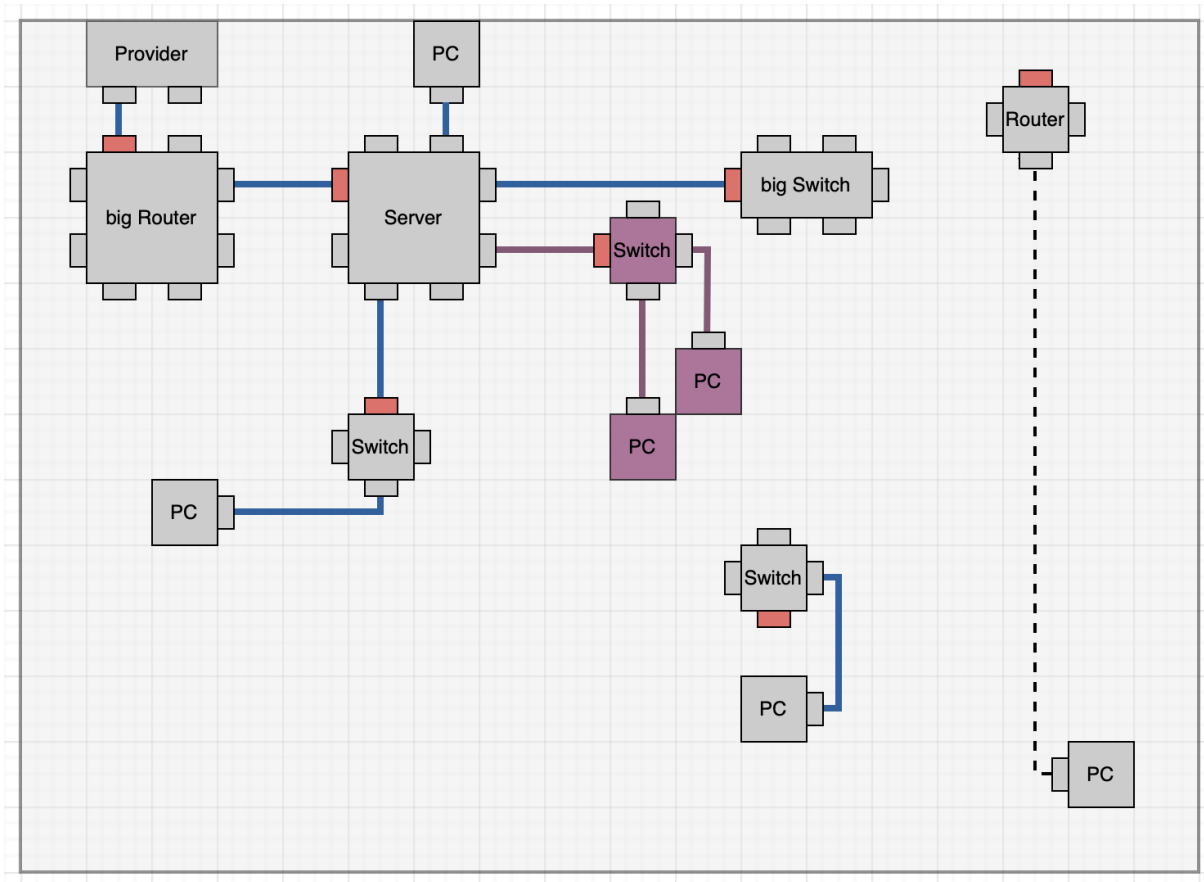


Abbildung 6: beispielhafter Aufbau des Grids

Quelle: [diagrams](#)

Kernmechanik

Im Wesentlichen darf an jeden Port eines Netzwerkgerätes nur genau ein Kabel und damit ein weiteres Netzwerkgerät angeschlossen werden. Somit müssen die SuS versuchen ihre vorhandenen Ports möglichst effizient zu verwenden. Zusätzliche Effekte und andere Arten von Übertragungsmedien helfen dabei, diese Aufgabe auf verschiedene Varianten zu bewältigen. So kann beispielsweise WLAN genutzt werden, um flächendeckend zu arbeiten, allerdings müsste man sich dafür bei den Upgrades oft für die Wireless-Ports entscheiden. Oder man entscheidet sich eher für Kabel und könnte dann über verschiedene & schnellere Kabelarten & Topologien sein Netz aufbauen.

Regeln

Operative Regeln:

- Die SuS müssen die Computer und Netzwerkkomponenten über verschiedene Übertragungsmedien miteinander verbinden.
- Dabei liegt ein Grid zu Grunde, was nur begrenzten Platz liefert.
- Effiziente Ressourcen sind rar und müssen gemanaged werden

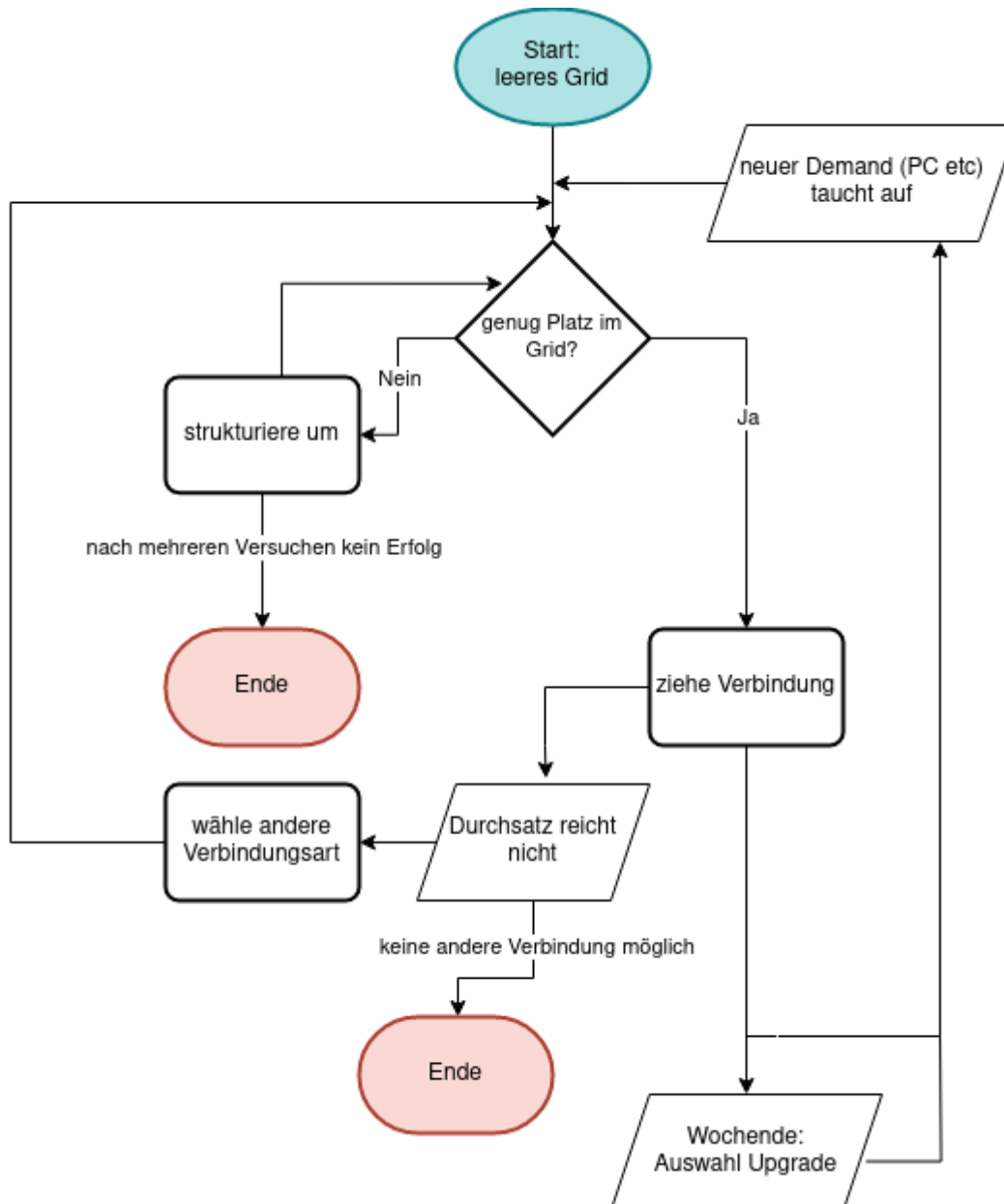


Abbildung 7: Flussdiagramm des Gameflows

Quelle: [Draw.IO](https://www.draw.io)

Grundregeln:

Zufällig auftretende Komponenten müssen über ein Grid miteinander korrekt verbunden werden. Kein Feld darf dabei "doppelt" belegt sein (außer durch besondere Komponenten wie z.B. wireless).

Datenpakete laufen automatisch über die Verbindungen von A nach B, können sich aber stauen. Zu viel Stau führt zum Ende des Spiels.

Spielziel

Fundamental:

Das Ziel des Spieles ist es, so effizient wie möglich Computer miteinander zu vernetzen.

Am Anfang kommt der Aspekt der Effizienz noch nicht durch, da in den Tutorials (Anfangslevel) erstmal die grundlegenden Konzepte vermittelt werden. Erst im letzten Level, dem "freien" Level, wird dann die Wichtigkeit der Effizienz bewusst, da dann entsprechend raffiniertere Strukturen verwendet werden müssen, um das Ende des Spiels so lange wie möglich hinauszuzögern. Dafür benötigen die SuS allerdings die Kenntnisse, die in den vorhergehenden Leveln erworben wurden.

Qualitätsmerkmale:

Die SuS müssen so lange wie möglich dem Demand standhalten, insofern gewinnen sie immer das Spiel - allerdings bietet ein High Score die Möglichkeit, dass sie sich selber wieder und wieder übertreffen wollen. Ähnlich wie bei Tetris ist es belohnend, beim nächsten Versuch wieder ein Stück weiter gekommen zu sein bzw dem wachsendem Demand standhalten zu können.

Das Spiel wird durch echte Namen und Netzwerkkomponenten sehr realitätsnah. Besonders auch in den ersten Leveln ist das Spiel sehr nah an den erlebten Realitäten der SuS, da sie sich hier im vertrauten Umfeld eines Wohnhauses bzw. einer Schule befinden.

Erwartungswerte und Wahrscheinlichkeit:

Nach einer gewissen Zeit tritt das Ereignis "Ende der Woche" ein, wo sich für eins von drei zufällig gewählten Upgrades entschieden werden kann. Diese Upgrades sollen alle gleich wahrscheinlich sein, die Balance kommt durch die Anzahl, wie viel in dem Upgrade drinsteckt. Das könnten zum Beispiel drei Einheiten Glasfaserkabel, zwei zusätzliche Ports an einem Switch oder ein WLAN-Modul sein.

Diese sind nun je nach Situation der SuS von unterschiedlichem Wert, also müssen sie dann auch abwägen, welches der Upgrades für ihre Situation am gewinnbringendsten ist.

Game Options

Am Anfang wird die Spielerzahl zwischen zwei und vier Spielenden ausgewählt. Dies wirkt sich vor allem auf das Tutorial aus, da dieses dann entsprechend oft auf dem Bildschirm verfügbar sein wird. Für den weiteren Spielverlauf wirkt es sich nur auf die Gridgröße und die Geschwindigkeit aus, in der neue Objekte auf dem Grid erscheinen, sowie den wachsenden Demand.

Die Möglichkeit der Unterbrechung des Spiels durch einen Pausenknopf hat keine weiteren Auswirkungen auf das Spiel und kann jederzeit erfolgen.

Einschränkungen

Die SuS müssen miteinander kommunizieren und **planvoll** die Verbindungen legen, da sonst das Grid sehr schnell zu voll wird. Das bedeutet das Abwägen von verschiedenen Upgrades in verschiedenen Situationen, das Auswählen der richtigen Verbindungstypen sowie das richtige Vernetzen der Netzwerkkomponenten.

Balancing

Fairness

- Tutorial ist unterschiedlich schwer, je nach Schwierigkeitsgrad gibt es mehr oder weniger Hilfestellungen
- bei mehreren Spielern könnte man bei denjenigen, die schneller mit ihren Haushalten fertig werden, noch mehr auftauchen bzw. komplexere Strukturen, was dynamisch erkannt und gehandelt wird (gilt nur für das Tutorial)
- die SuS können sich gegenseitig über Tipps helfen
- sollte das System über entsprechenden Input merken, dass ein Spieler für einen anderen Spieler die Verbindungen zieht (beispielsweise wenn die Verbindung über den eigenen Quadranten ohne einen zweiten Input weitergeht), wird er über ein entsprechendes Popup visuell zurecht gewiesen und die gezogene Verbindung wird ungültig

Herausforderung vs. Erfolgserlebnis

Die Herausforderung besteht darin, dem wachsendem Demand standzuhalten. Das funktioniert mit entsprechenden Kenntnissen besser, allerdings müssen diese auch erstmal erworben und angewendet werden.

Wenn Erfolge erzielt wurden, wie bspw. das erste Level geschafft oder einen Haushalt komplett vernetzt wurde, steigt das Herausforderungsniveau, indem farbige Kodierungen hinzukommen, wobei nur Geräte mit gleichen Farben verbunden werden können. So steigt die Schwierigkeit einer guten Planung noch einmal erheblich.

Die anspruchsvollste Herausforderung ist, der wachsenden Komplexität stand zu halten.

Relevante Spielentscheidungen

- Die relevanteste Entscheidung für die SuS ist welches Upgrade sie in ihrer Situation am Wochenende für sich wählen. Je nachdem, wie sie ihr Netzwerk aufgebaut haben, bringen unterschiedliche Upgrades unterschiedlich viel Nutzen.
- nach Ende einer Woche stehen den SuS immer drei Upgrades zur Auswahl
- die SuS entscheiden auch jedes Mal erneut mit welchem Kabel sie wie die Verbindung ziehen

Können vs. Glück

Auswahl an Upgrades zufällig generiert, dadurch Entscheidung sowohl als auch Glücksfaktor

- das Aufkommen neues "Demands" ist ebenfalls glücksbasiert dadurch, dass die Generation auf dem Grid zufällig ist
- durch einen einigermaßen smarten Algorithmus sollte der Zufall nicht unfair zufällig sein

Verstand vs. Fingerfertigkeit

Das Spiel erfordert nicht viel Fingerfertigkeit, es ist eher wie ein sich konstant weiterentwickelndes Puzzle, in das konstant mehrere Aspekte hereinspielen.

Pause vom Nachdenken gibt es über das Pause-Menü oder in der Zwischenzeit, wenn die SuS alles rechtzeitig ordentlich miteinander verbunden haben und nur auf den nächsten aufploppenden Haushalt etc. warten.

Einzelkonkurrenz vs. Teamwork

- das Lernspiel kann ohne Vorkenntnisse gespielt werden, macht aber auch mit Vorwissen Spaß
- Jene, die schon Ahnung haben, können Anfängern helfen, da alle grundsätzlich das Gleiche machen
- keine Konkurrenz, dadurch konstruktivere Atmosphäre

- Gefahr: einer macht alles und die anderen stehen nur daneben
- möglicher Lösungsansatz: einerseits physische Trennung, dass jeder an seinem Feld steht und nicht von seiner Ecke aus an die gegenüberliegende Ecke rankommt, andererseits Check für doppelte Inputs, wo Spieler für eine Verbindung seinen Quadranten verlassen muss

Belohnung vs. Strafen

Belohnungen:

- am Ende High Score, wie lange man durchgehalten hat
- wenn ein Haushalt komplett abgeschlossen, dann "verpufft" er ähnlich wie in Tetris in einer schönen Animation

Strafen:

- wenn zu viel Stau dann negatives visuelles Feedback (rotes Blinken oder ähnliches, 'Achtungs'-Straßenverkehrszeichen, rote Stelle wie bei Google Maps, ...)
 - wenn Stau aber gelöst, dann Belohnung als visuelles Feedback, was sich gut anfühlt
 - die Upgrades sind in gewisser Weise auch Belohnungen, weil sie das Gefühl vermitteln, einen "Checkpoint" zu haben (auch wenn es nicht wirklich einer ist)
-

Story, Setting und Charaktere

Story und Narrative

In unserem Lernspiel geht es um einen Schüler bzw. eine Schülerin, der/die zu Anfang im Haus der Eltern vor die Situation gestellt ist, seinen/ihren Rechner an das Internet anzuschließen, um eine Hausaufgabe bewältigen zu können. Ihm/Ihr steht dabei sein/ihr Informatiklehrer telefonisch zur Hilfe. Je nach ausgewähltem Schwierigkeitsgrad unterscheidet sich die Menge der Hilfestellungen des Lehrers, sowie die Erklärung der einzelnen Objekte. Sobald der Rechner richtig angeschlossen ist, endet das erste Level.

Anschließend bittet der Informatiklehrer über einen erneuten Telefonanruf um Mithilfe beim Aufstellen eines komplexeren Rechnernetzes im neuen Computerkabinett der Schule des

Schülers/der Schülerin. Hier spielen nun auch weitere Komponenten, sowie verschiedene Kabeltypen eine Rolle. Der Informatiklehrer dient weiterhin der Konsultation bei Problemen, gibt Tipps, Hilfestellungen und Erklärungen zu den einzelnen Komponenten, abhängig vom ausgewählten Schwierigkeitsgrad.

Zwischen dem zweiten und dem dritten Level findet ein Zeitsprung statt. Der Schüler/die Schülerin ist jetzt erwachsen und baut sich in seiner/ihrer Firma ein eigenes stetig wachsendes Rechnernetz auf, welches im Verlauf des Levels zu mehreren Teilnetzen erweitert wird. In alter Freundschaft steht ihm/ihr immer noch die Hilfe des Informatiklehrers zur Verfügung.

Die Story dient dem Zweck die Thematik von Rechnernetzen erst einmal in einem vertrauten Umfeld, zuhause, anzugehen. Auch bei steigender Komplexität in Level 2 befinden sich die SuS noch in einem vertrauten, jedoch jetzt öffentlicheren Umfeld. Das hat den Hintergrund, dass sich neue Kenntnisse besser in bereits vertrautem Umfeld erlernen lassen. In Level zwei werden diese dann weiter gefestigt, immer noch in vertrautem Umfeld, um die SuS nicht zu überfordern. Sobald die gelernten Kenntnisse durch Übung mehr ins Unterbewusstsein übergegangen sind, verliert die Story an Relevanz, sie also hauptsächlich dafür da, den Prozess des Lernens zu erleichtern, indem eine Situation geschaffen wird, die tatsächlich auftreten kann.

Unsere Geschichte ist für SuS interessant, da sie real mögliche Situationen schildert und somit so nah wie möglich an den eigenen Erlebnissen der SuS ist. Zudem untermauert die Story die Ästhetik des Spiels, da sie Auswirkungen auf den Hintergrund des jeweiligen Levels und somit auf die gesamte Optik hat. Die Tatsache, dass Hilfestellungen und Erklärungen durch den Informatiklehrer gegeben werden ermöglicht eine gute und sinnvolle Einbettung von Tipps in das Spiel. Durch die erzählte Geschichte verstehen die SuS besser, wie die Komponenten verbunden werden und lernen so die Mechaniken des Spiels schneller.

Spielwelt

Die Spielwelt ist von Level zu Level unterschiedlich. Generell hat sie aber eher ein realistisches Setting, um die Lernenden an ihre vertraute Umwelt zu erinnern.

Das erste Level beginnt im eigenen Zuhause. Dafür wird ein generisches Wohnzimmer verwendet. Die Atmosphäre soll eher heimlich, vertraut sein. Die SuS wissen, was sie erwartet, und können sich voll und ganz auf die Aufgabe vor Ihnen konzentrieren. Deswegen werden nur die wichtigsten Elemente in minimalistischer Weise angedeutet, sodass kein visueller Lärm von der eigentlichen Aufgabe ablenkt.

Das zweite Level ist schon etwas größer gedacht - jetzt befinden sich die Schüler im Computerkabinett der Schule. Eine helle Atmosphäre mit freundlichen Farben, die die Schule nicht so trist aussehen lassen, sind hier angebracht. Dieses ist allerdings eher der Hintergrund.

Die immer komplexeren Level haben immer entsprechende Hintergründe, auf denen das Grid projiziert wird. Es ist also sinnvoll, eine Art Semi-Vogelperspektive dafür zu verwenden. Auf dem Grid spawnen dann die einzelnen, symbolhaften Geräte mehr oder weniger geskriptet (am Anfang mehr, am Ende weniger).

Das letzte Level ist dabei allerdings ein wenig eigen. Hier ist dann ein freies Spiel auf Stadt- oder eventuell auch Landesebene, um den Schülern zu zeigen, wie weit so etwas gesponnen werden kann. Die Hintergründe rücken hier sehr aus dem Fokus, werden also sehr minimalistisch, weil ohnehin sehr viel auf dem Grid visuell passieren wird. Da braucht man nicht noch mehr visuellen "Noise" zu erzeugen.

Generell soll die Spielwelt eher stilisiert und farbenfroh aussehen, ähnlich Mini-Motorways.

Charaktere / Pädagogische Agenten

Unser pädagogischer Agent ist der Info-Lehrer. Als lieber, älterer Mann leitet er die SuS durch ihre Aufgaben in Form von Text und visuellen Hinweisen. In Form eines Telefon-Anrufes ist er für die SuS da, wenn sie ihn brauchen - anfangs geskriptet, später evtl. dynamisch. Er steht sehr auf alte Technik, erkennt aber auch das Potential von moderner Technik und erwähnt deshalb am Rande im Tutorial immer mal wieder, wie es denn früher war und warum das jetzt besser oder anders ist (Beispiel: "Früher waren die Kabel noch 5cm breit, heutzutage haben sie einen Durchmesser wie dein kleiner Finger! Dadurch kannst du diese Kabel sogar ordentlich aufrollen, das nenne ich mal Kabel-Management!")

Zu sehen wird er nur als verschwommenes Bild auf dem Handy sein - durch dieses kommuniziert er ja mit den Schülern. Deswegen sieht man es bei allen Leveln bis auf das Letzte auch immer im GUI.



Abbildung 8: Illustration des pädagogischen Agenten

Level

Übungslevel

Fast alle Level bis auf das letzte sind in dem Sinne Übungslevel, wobei die Hilfestellungen immer weniger werden. Man kann aber auch das erste Level als 'Tutorial' sehen in dem Sinne, dass es die grundlegenden Mechaniken erklärt, wohingegen in den späteren Leveln eher Prinzipien und weitere Komponenten erklärt werden.

Im ersten Level wird den SuS die Grundaufgabe vorgestellt. Indem sie ihren Heimrechner vernetzen sollen, lernen sie die Grundprinzipien des Spiels. Dafür kriegen sie Kabel etc. gestellt und werden mit visuellen Hinweisen (alles verdunkelt sich außer das relevante Objekt, was mit einer Umrahmung nochmal deutlich gekennzeichnet wird) sowie einem kleinen erklärendem Text von unserem pädagogischen Agenten angeleitet.

Im zweiten Level werden die gelernten Prinzipien auf einen größeren Demand, nämlich den PC-Pool der Schule angewendet. Dort werden einige Topologien vorgestellt und weitere Komponenten wie Repeater.

Das dritte Level ist dann schon das freie Spiel, wo alles Gelernte gefestigt & bewiesen werden soll.

Bewertung / Wertung / Belohnung

Die in den Tutoriallevels vermittelten Mechaniken und neu eingeführten Spielkomponenten tauchen im Verlauf des weiteren Spiels mehrmalig auf. Neben den Kernkompetenzen, die das Lernspiel vermitteln soll, wird so die Grundlage für das freie Spiel im letzten Level gelegt, in dem alle vorher kennengelernten Spielbestandteile gebündelt angewandt werden müssen, um dort Fortschritte zu erzielen.

Im Verlauf der Tutoriallevels können die SuS in den Textdialogen regelmäßig für ihren Fortschritt und die neu erlangten Kenntnisse gelobt werden. Die Belohnung erfolgt in Form von lobendem Text durch den Pädagogischen Agenten, dessen Kommunikation als Sprechblase vom Telefon ausgehend erfolgen soll. Auch 1-2 gut aussehende Animationen wie bspw. Sternchen oder ähnliches stellen positives Feedback für erfolgreiche Aktionen dar.

Interface

Entwurf

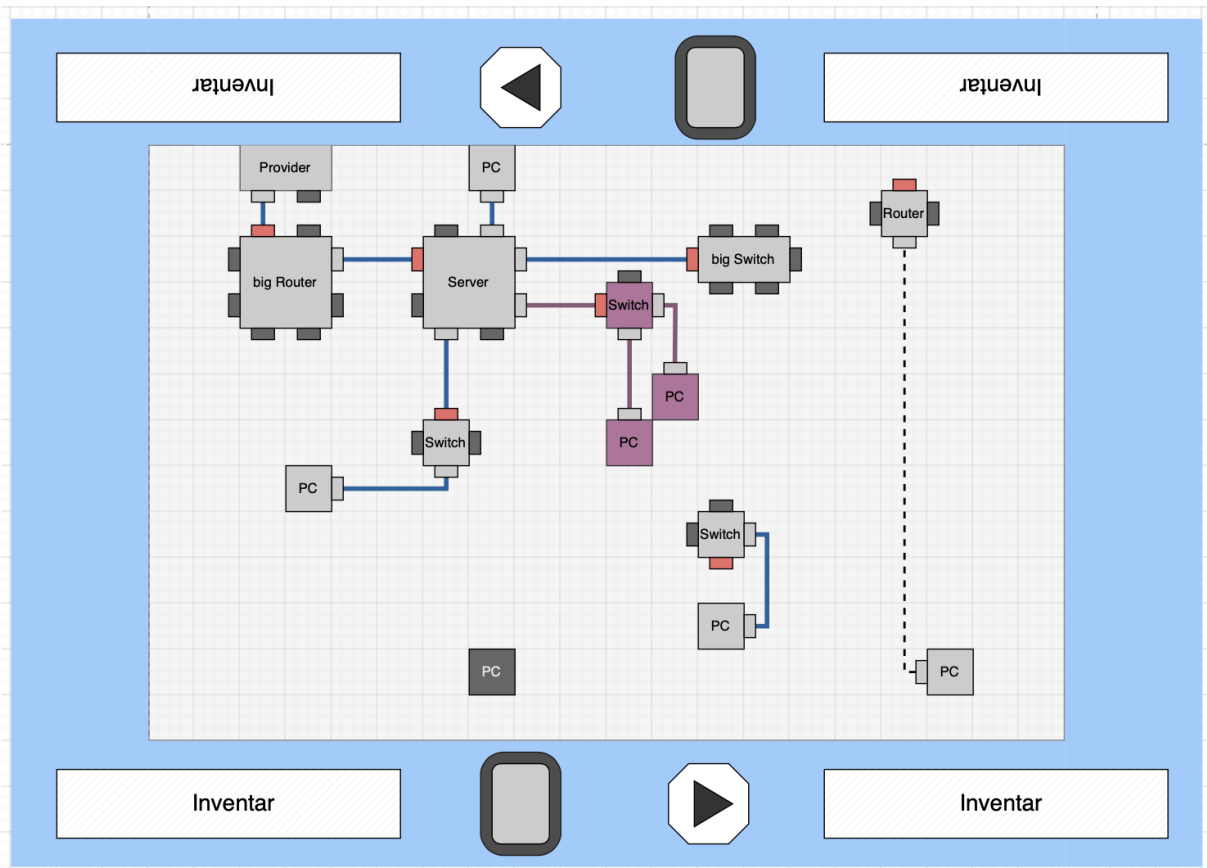


Abbildung 9: Entwurf des GUI zusammen mit dem Grid

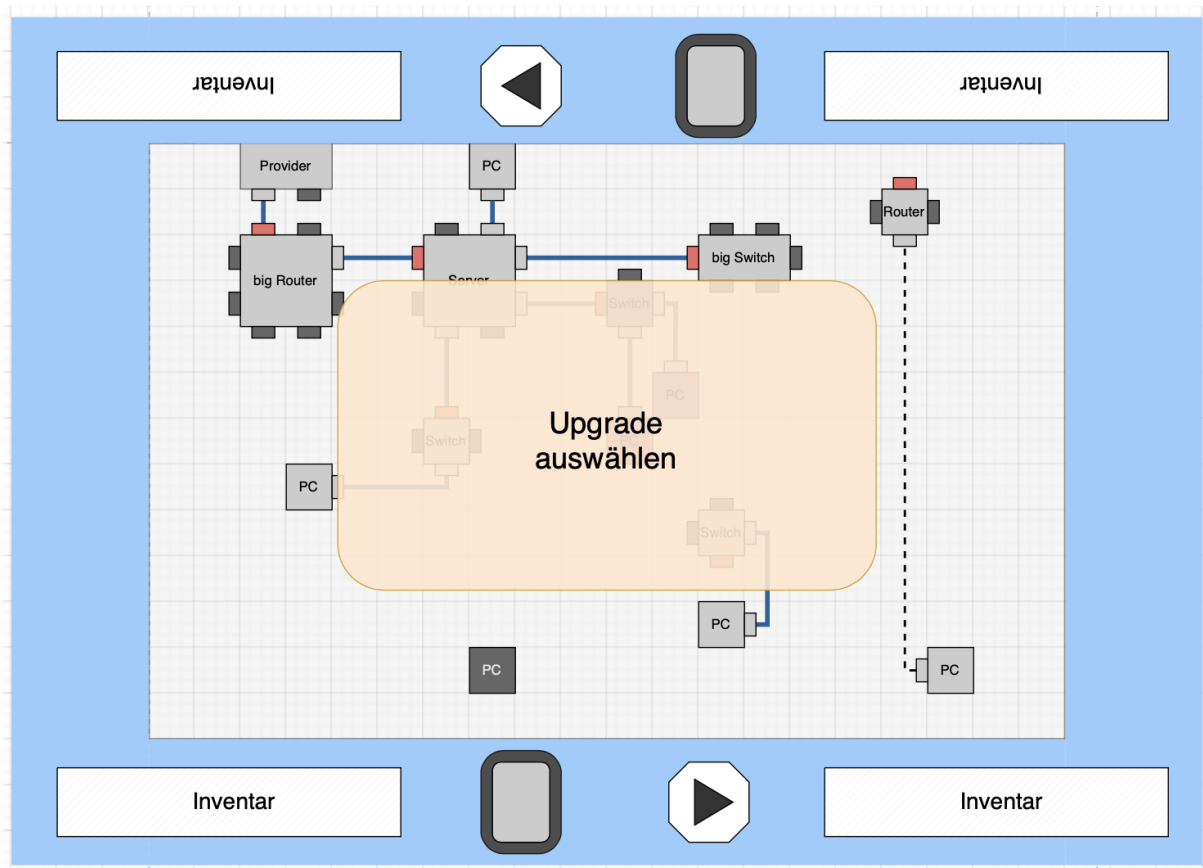


Abbildung 10: Auswahl des Upgrades am "Wochenende"

Quelle: [diagrams](#)

Side Note: Die Beschriftungen der Komponenten dienen nur zur Erleichterung des Verständnisses. Sowohl die Komponenten als auch die Grafiken sollen später als Grafiken only (mit evtl. kleinen Text zu den jeweils langen Seiten gerichtet), die idealerweise von allen Richtungen gleich einsehbar & eindeutig sind, dargestellt werden.

Generell soll der Bildschirm was das GUI angeht wie einmal längs "geteilt" werden, sodass SuS, von denen je 2 an der langen Seite stehen, jeweils in ihre Leserichtung die entsprechenden Beschriftungen richtig herumhaben. Deswegen auch die Spiegelung des Pause-Buttons sowie dem Pädagogischen Agenten.

Physical System

Die berührbaren Elemente unseres Lernspiels sind alle Einstellungen (Schwierigkeitsgrad, Spieleranzahl, Start, Weiter, Pause, Neustart), die Auswahl der Upgrades nach Ende einer Woche, das Inventar mit verschiedenen Kabeln und anderen Upgrades, die Hilfestellungen in Form des Telefonanrufes und das Grid selbst. Die als Einstellungen bezeichneten

Auswahlelemente sind als generelle Spielprinzipien bekannt und daher selbsterklärend. Zudem werden Aufforderungstexte angezeigt, wie "Spieleranzahl auswählen", was hierfür ausreicht. Die Auswahl der Kabel o.Ä. aus dem Inventar wird im ersten Level durch optischen Fokus (blinken des Objektes, ausgrauen oder verdunkeln aller anderen) auf das jeweilige Objekt gelernt. Selbiges gilt für das Telefon und die Platzierung der Objekte auf dem Grid. So wird anfangs genau deutlich, welcher Teil der Spieloberfläche interaktiv ist und welcher nicht.

Die Interaktionen mit der Spieloberfläche stimmen eindeutig mit den Aktionen im Spiel überein. Das heißt die Auswahl eines Objektes wählt ein Objekt aus, wenn die SuS eine Kabelverbindung ziehen, werden im Spiel zwei Netzwerkkomponenten miteinander verbunden und die Wahl eines Upgrades aktiviert dieses.

In dem ersten Level existiert das Spielfeld genau so oft wie die Anzahl der Spielenden gewählt ist. So ist anfangs räumlich getrennt, welches Spielfeld zu welchem Spielenden gehört, wodurch haptische Eingaben deutlich voneinander abgetrennt werden können. Der Rahmen mit allen Grundfunktionen (genauer in Visual System) ist ebenfalls getrennt vom Grid des eigentlichen Spiels, wodurch Eingaben hier ebenfalls deutlich zugeordnet und entsprechend verarbeitet werden können. Im späteren Verlauf des Spiels müssen wir die haptische Eingabe wahrscheinlich auf eine zu einer Zeit reduzieren, da sonst beispielsweise beim Ziehen eines Kabels nicht mehr erkannt werden kann, von wo bis wo es gezogen wird, sofern mehrere haptische Eingaben zur gleichen Zeit erfolgen. Das würde den SuS dann durch den Info-Lehrer am Telefon gesagt, aber ob das technisch notwendig ist, wird sich während der Entwicklung herausstellen.

Die Wahrnehmung der Spielwelt erfolgt ausschließlich durch die visuelle Ausgabe des Tabletop Displays und durch bestimmte Audiosignale (genauer in Audio, Musik, Sound). Ein weiteres physisches Ausgabegerät halten wir nicht für nötig, da durch die optische Ausgabe am Display bereits genug Output existiert und zusätzliche Ausgaben die Schüler möglicherweise ablenken oder mit zu vielen Reizen überfluten würde.

Visual System

- vor Beginn des Spiels: Startscreen mit Spieltitel, Auswahl der Spieleranzahl, Auswahl des Schwierigkeitsgrades alias Menge an vorhandenem Vorwissen (mit Erklärung der unterschiedlichen Schwierigkeitsgrade, Information zu möglichem Wechsel der Schwierigkeit auch während des Spiels)

- durch Auswahl Schwierigkeitsgrades unterschiedliche (Anzahl an) Dialoge(n) und unterschiedlich schnelles Spawnen der Komponenten auf dem Grid (insofern nicht Skript-bedingt)
- Unterschiede allerdings nicht zu gravierend, damit Fehleinschätzungen vorgebeugt und als Lernspiel darf es lieber zu viel als zu wenig lehrend sein
- einzelne Level: Grid(s) mit Objekten eingebettet in zum jeweiligen Level passenden Hintergrund (Level 1: Zimmer in einem ganz normalen Haus, Level 2: Computerkabinett einer Schule, Level 3: mehrere Räume einer Firma, bis hin zu mehreren Firmensitzen (Hintergrund verliert an Relevanz)
- um jedes Level herum existiert ein Rahmen mit allen Grundfunktionen: Pausebutton, (Score), Geschwindigkeit, Inventar, Auswahl der Upgrades nach einer Woche und Hilfestellungen bzw. dem Telefon für den Anruf des Informatiklehrers
- Pausenmenü: Grid und Hintergrund des aktuellen Levels bleibt sichtbar, nur leicht überdeckt von einem Layer Grau, um Pausenmodus sichtbar zu machen, Button zum Weiterspielen, Button für Hilfestellungen, Schwierigkeitsanpassung
- Ende von Level 1/2: großer Glückwunschtext vom Info-Lehrer, er feiert die SuS, eventuell Animationen für Konfetti o.Ä., Button um Endstand nochmal anzusehen, Button für nächstes Level
- Endstand von Level 1 und 2: Grid(s) und Hintergründe mit allen gesetzten Objekten sichtbar, Rahmen mit Grundfunktionen reduziert auf Button für nächstes Level
- Ende von Level 3: großer Glückwunschtext, Anzeige des erreichten Scores, (eventuell Liste aller gewählten Upgrades,) Button zur Anzeige des Endstandes, Button zum Neustart des Spiels
- Endstand von Level 3: Grid und Hintergrund mit allen gesetzten Objekten, Rahmen mit Grundfunktionen reduziert auf Neustarten-Button

Control System

Die SuS kontrollieren das Spiel über strategisches Ziehen der Verbindungen mithilfe von Touch-Eingaben.

Das Spawnen der Komponenten ist "smart" zufällig, also haben die SuS sehr wenig Kontrolle darüber. Sie müssen mit dem, was sie haben, eben arbeiten - aber wie sie die Verbindungen ziehen, ist ihnen komplett überlassen. Diese Macht müssen sie aber eben geschickt einsetzen, um sich nicht alles so schnell zu verbauen.

Bei der Auswahl des Upgrades am Wochenende kommt ein fast demokratisches Wahlsystem zum Einsatz. Zur Auswahl stehen jeweils 3 zufällig ausgewählte, ungefähr gleichwertige

Upgrades. Je nach Lage der SuS können diese sich für eins entscheiden. Dabei gibt jeder Spieler eine Stimme ab für sein favorisiertes Upgrade. Dabei kann sich auch abgesprochen werden. Das Upgrade mit den meisten Stimmen wird gewählt. Sollten aber mehrere Upgrades die jeweils höchste Anzahl an Stimmen haben (bspw. 2 - 2 bei 4 Spielern) wird zufällig ausgewählt. Insofern haben die SuS also die absolute Kontrolle über die Wahl des Upgrades, insofern sie sich absprechen können.

Das Spiel endet immer gleich, darüber haben die SuS also keine Kontrolle. Worüber sie Kontrolle haben, ist allerdings wie weit sie kommen. Durch den Strategie-Aspekt liegt es allein in ihrer Hand, die Verbindungen so zu organisieren, dass der Demand möglichst lange abgedeckt wird.

Das Interface ist hat sehr wenige Elemente (Pause + "Inventar" + Handy als Schnittstelle zum Info-Lehrer), insofern sollte es handlich und intuitiv sein. Das erlaubt den SuS, sich auf die Aufgabe zu konzentrieren. Allgemein ist es ein bisschen so, als wären sie Firm Chefs - sie sagen, wohin was wie schnell ausgebaut werden soll.

Transparenz

- Das Interface baut auf einfachen Touchevents wie click (Touch) & drag and drop auf. Für die gesamte Steuerung werden außer diesen grundlegenden Events keine neuen Bedienmethoden eingeführt. Somit sollte das Interface ohne Übung benutzbar sein.
- Spezielle Steuerungseinstellungen oder verwirrende Interaktionsmöglichkeiten sind nicht vorgesehen. Das wird am Ende beim Testen nochmal geprüft und ggf. nach adjustiert.
- Im Spielverlauf, insbesondere im letzten Level, könnte der darstellbare Platz mit steigender Spielkomplexität und fortschreitender Ansammlung von Spielbestandteilen etwas unübersichtlich werden. Der Reiz dieses Levels liegt unter anderem darin, trotz dessen den Überblick zu bewahren. Für bestimmte Spielergruppen kann das Interface dann in Teilen zu unübersichtlich werden.
 - Dieses Resultat sollte erst spät im Spiel erzielt werden, wenn die Lernziele bereits erreicht sind.

Feedback

- SuS sollten groben Plan haben, wie sie welche Rechner strategisch verbinden könnten es wird ihnen aber auch gezeigt, wie es am effizientesten wäre
- sobald Gerät per Kabel verbunden, fängt Datenstrom an, zu fließen & es leuchtet ein bisschen heller
- visuelles Gimmick, wenn erfolgreiche Verbindung (kleine Sternchen oder so) als kleine Belohnung
- bei wireless: wenn verbunden, dann periodisch kleine Animation (aus- & eingehende Wellen oder Pfeile)
- Visualisierung vom Bereich, den Wireless abdeckt mit entsprechenden visuellen Antworten der verbundenen Geräte, sobald etwas ankommt
- wenn Datenpakete Stau: stetig (farblich) intensiveres, rotes, langsames Blinken an Stelle wo es sich staut, ähnlich wie auf Google Maps (nur mit Blinken)
- wenn Kabel gezogen: dort, wo Feld belegt, ausgegraut; freie Flächen bleiben
- normalpädagogischer Agent liefert Hinweise wie Erinnerung an das Ziel und mögliche Wege, wie das erreicht werden kann, vor allem in Tutorials
- wenn man mehr Daten hat für Durchschnitts-Scores könnte auch bei jedem Ende eine kleine Einschätzung basierend auf dem Score und wie andere sich normalerweise schlagen kommen ("Rookie", "Könner", "Lehrling", "Meister", "Chef", ...) diese sollte aber nicht zu negativ sein, um nicht demotivierend zu wirken

Audio, Musik, Soundeffekte

- Audiodesign ist auf Soundeffekte in bestimmten Situationen beschränkt, es gibt keine Spielmusik
 - (un)erfolgreiches Verbinden von 2 Komponenten
 - Spawnen von Geräten
 - Wochenende poppt auf
 - generell Öffnen / Schließen von Menüs (Pop-Ups)
- wenn Pädagogik-Agent anruft, könnte das durch Klingelton begleitet werden

Hilfesysteme

- in Form einer Sprechblase (eckig, weil Anruf), die den Anruf des Info-Lehrers in form eines Dialog-Feldes lesbar macht
- darüber als Text und vor allem Bilder (Video-Anruf?) bspw. Topologien oder ähnliches gezeigt

- optional könnte man noch ein 'Archiv' mit Übersichten an bereits gebauten Topologien oder Definitionen von Komponenten einbauen auch mit als Option bei Schwierigkeitsgrad-Auswahl