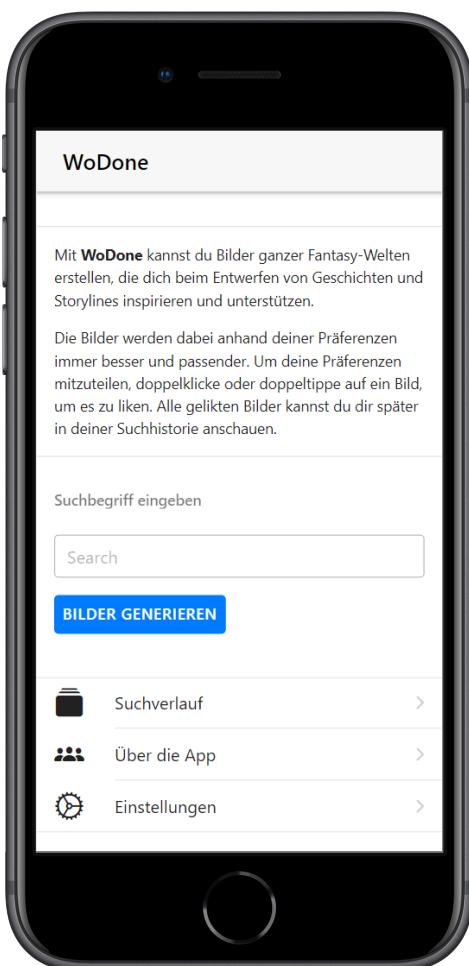


# Dokumentation Komputer und Creativität

## Gruppe 5 - WoDone



## Inhalt

1. Ideenfindung .....	2
2. Die Kreativität in WoDone .....	5
3. Arbeitsteilung im Team.....	8
4. Technische Details .....	10

# 1. Ideenfindung

Die Ideenfindung unterteilte sich wie im “Design Thinking” üblich in 3 Schritte, “Empatic Exploration”, “Problem Framing” und “Ideation”, wobei im Zuge der Zusammenarbeit im Kurs das Kolaborationstool Miro verwendet wurde.

Angefangen mit der Empathic Exploration wurden zuerst Randbegriffe definiert und ein allgemeines Verständnis geschaffen. Beispielsweise, dass der Begriff “Gamesdesign” nicht nur Videospiele enthalten muss. Dabei ging es unter anderem darum verschiedene Altersgruppen und unterschiedliche Alltagssituationen (wie z.B. Aufgabenverteilung im Haushalt), in Betracht zu ziehen. Daraufhin definierten wir als Gruppe eine “Persona”, auf die die Probleme, Ansichten und Werte der Zielgruppe projiziert wurden. Als Persona schufen wir “Samantha”, eine junge Masterstudentin im Bereich Games Design, die besonderen Wert auf kohärente und kreative Welten und Charaktere legt. Dabei fokussierten wir uns besonders auf sogenannte “Pains” und “Gains” - Probleme also, die es zu lösen gab und Vorteile die ein Produkt wie das unsere liefern sollte.

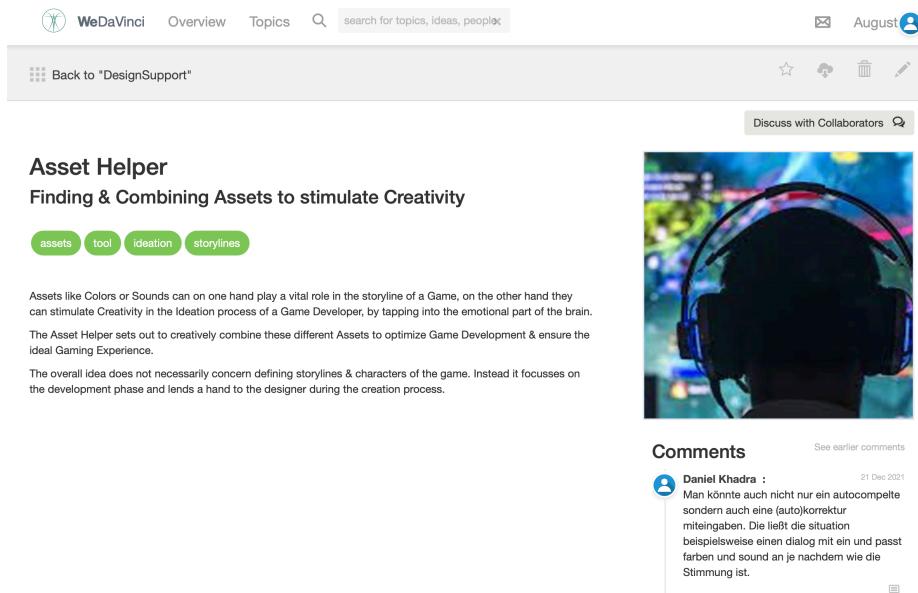
Das Ende der Empathic Exploration bestand in einer tatsächlichen Kundenanalyse. Dazu bot sich uns die Möglichkeit, Samuel Saruba, einen Games Design Studenten der TUM, zu interviewen. Da Herr Saruba bereits viel Erfahrung mit dem Entwicklungsprozess von Computerspielen hatte, konnten wir durch das Interview viele Einblicke in die tatsächlichen Bedürfnisse eines Spieletwicklers erhalten. Bereits im Laufe des Interviews entstanden verschiedenste neue Ideen und bestehende Ideen entwickelten sich weiter. Der größte Vorteil, den das Interview lieferte war ein Verständnis für die Bereiche, in denen realistisch ein Markt für ein Produkt wie das unsere entstehen könnte (z.B. “Assets” und “Testing”). Diese wurden im nächsten Schritt, dem Problem Framing, weiter durchdiskutiert und hinterfragt, woraufhin im 3. Schritt - dem Ideation-Prozess, -Produktideen entstanden. Nach erneuter Absprache mit Herrn Saruba bekamen wir seine Sicht auf unsere Ideen und bereits ein gewisses realistisches Verständnis, ob solche Tools tatsächlich einen Nutzen hätten.

Das finale Schritt im Ideation-Prozess waren schließlich die verschiedenen Sichtweise der anderen Kursteilnehmer. Hierfür benutzten wir die Design-Thinking-Plattform “WeDaVinci”, auf der alle Ideen des Kurses gesammelt, und mittels Kommentaren durchdiskutiert wurden. Wir ließen uns vom Feedback und den Ideen der anderen inspirieren und kombinierten schlussendlich Aspekte der folgenden - in der Ideation entstandenen - Ideen:

- This Scenario Doesn’t Exist: eine Generator für Storyline Szenarien
- Asset Helper (siehe Bild 1): eine Suchmaschine für verschiedenste Assets eines Videospiele wie z.B. Töne oder Themes

- Random Life Generator: ein Tool, welches durch Zufall den Hintergrund eines Spielcharakters generiert

Jede Idee bot eigene Ansichten und Lösungen für ähnliche Probleme. So war beispielsweise der Zufalls-Aspekt des “Random Life Generators” aus technischer Sicht umsetzbarer, als z.B. ein Menschen-ersetzender, algorithmischen Evaluationsprozess.



**Asset Helper**  
Finding & Combining Assets to stimulate Creativity

assets tool ideation storylines

Assets like Colors or Sounds can on one hand play a vital role in the storyline of a Game, on the other hand they can stimulate Creativity in the Ideation process of a Game Developer, by tapping into the emotional part of the brain. The Asset Helper sets out to creatively combine these different Assets to optimize Game Development & ensure the ideal Gaming Experience. The overall idea does not necessarily concern defining storylines & characters of the game. Instead it focusses on the development phase and lends a hand to the designer during the creation process.

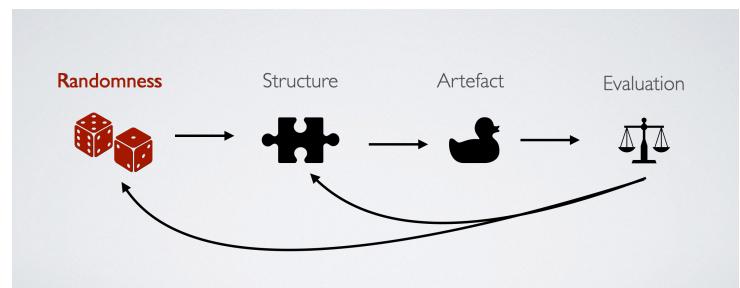
Comments

Daniel Khadra : 21 Dec 2021  
Man könnte auch nicht nur ein autocorrect sondern auch eine (auto)korrektur mit eingeben. Die liefert die situation beispielweise einen dialog mit ein und passt farben und sound an je nachdem wie die Stimmung ist.

**Bild 1:** Eine erste Iteration unserer Idee, die wir auf WeDaVinci stellten und mit anderen Kursteilnehmern durchdiskutierten. Später integrierten wir einige zentrale Aspekte daraus in unser finales Produkt.

## Einführung in den Creative Process

Der sogenannte “Creative Process” strukturiert verschiedener Schritte im Prozess der Ideenfindung (siehe Bild 2). Als Kurs benutzten wir ihn um selbst Ideen zu entwickeln, wie die gegebene Aufgabenstellung gelöst werden könnte. Er lässt sich aber auch wunderbar innerhalb der Aufgabenstellung selbst anwenden, in unserem Fall die Entwicklung einer Welt im Rahmen eines Computerspiels. Hierbei würde unser Produkt WoDone beim ersten Punkt, “Randomness”, ansetzen, in Form von visuellen Stimuli. Den nächsten Schritt - die Strukturierung - überlassen wir jedoch einem sehr viel intelligenterem Rechenzentrum - dem menschlichem Gehirn.



**Bild 2:** Der “Creative Process” definiert allgemein 4 Schritte, welche die Entwicklung und Umsetzung von Ideen strukturieren.

## Finaler Iterationsschritt & Feedback

Regelmäßig haben wir uns während des Ideenfindungsprozesses und später während des Prototyping Feedback von Herrn Saruba geholt. Am interessantesten für uns war jedoch seine finale Sicht auf das fertige Produkt.

Bereits früh hatten wir als “Pain” von Spieleentwicklern die Ideenfindung für Storylines angenommen. Bei unserem ersten Interview erkannten wir jedoch relativ schnell, dass Herr Saruba dieser Annahme zunächst widersprach. Über mehrere Iterationsschritte und Diskussionen mit anderen Kursteilnehmern formten wir die ursprüngliche Idee jedoch um und richteten uns eher auf einen allgemeinen visuellen Input aus, frei nach dem Sprichwort “Ein Bild sagt mehr als tausend Worte”. Wir zielten also mit unserem Endprodukt viel mehr darauf ab, allgemein - mittels Bildern - die Kreativität und Emotionen des Gamesdesigners anzuregen und ihn dabei zu unterstützen, nicht nur Storylines, sondern ganze Welten zu generieren.

Normalerweise entsteht sogenannte “Concept Art” auf Basis einer Videospiel-Idee und dazugehöriger Storyline, um diese zu visualisieren. Wir drehen den Spieß jedoch um und nutzen Concept Art, um den Gamesdesigner darin zu unterstützen, sich kohärente Welt auszudenken.

Als wir Herrn Saruba unser finales Produkt vorführten und ihn selber ein bisschen ausprobieren ließen war er stark beeindruckt. Vor allem von der Einfachheit der Bedienung und der Möglichkeit mittels Input und Likes den Output immer wieder aufs Neue beeinflussen und verändern zu können, war Herr Saruba fasziniert.

## 2. Die Kreativität in WoDone

Der kreative Prozess in unserem Programm unterteilt sich in zwei verschiedene Kategorien:

Die Kreativität, die in der Bildgenerierung durch neuronale Netze erforderlich ist, sowie die Kreativität, die durch unser Produkt beim Nutzer angeregt wird.

### Kreativität des neuronalen Netzes VQGAN + CLIP

Der erste Aspekt, in dem Kreativität in unserem Programm ist die Erstellung eines Bildes, welches möglichst genau die Suchanfrage des Nutzers berücksichtigt. Für die eigentliche Bildgeneration benutzen wir ein bereits vetrainiertes Netz mit der sogenannten VQGAN + CLIP Architektur, welches im Januar 2021 von OpenAI veröffentlicht wurde. Die generativen Fähigkeiten dieses Netzes sind überaus beeindruckend. Die eigentliche Kreativität in solchen Netzen wird an sich nicht explizit programmiert, sondern entsteht aus dem komplexen Zusammenspiel von Millionen von Neuronen. Das von uns verwendete Netzwerk ist eine Kombination zwischen einem VQGAN (Bild 3 GAN) und einem CLIP (Bild 4) Netz. Die Funktionsweise besteht im Wesentlichen darin, dass VQGAN die Bilder erzeugt, während CLIP beurteilt, wie gut ein Bild zu unserer Texteingabe passt. Diese Interaktion und den daraus folgenden Anpassungen der Gewichte im Netz, führt dazu, dass dieses immer realistischere und zusammenhängendere Bilder zu den Texteingaben generiert.

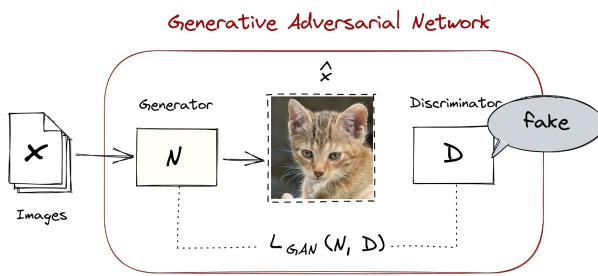


Bild 3: Ein Diagramm des grundlegenden Aufbaus eines generative adversarial network

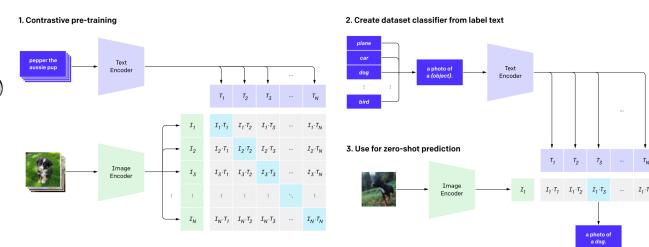


Bild 4: Ein Diagramm, das zwei Encoder zeigt, die ein contrastive language-image pre-training Netz verwenden

### Kreativität des Benutzers

WoDone unterstützt den Benutzer im kreativen Prozess, wie oben erwähnt, in der "Randomness"-Phase. Der Benutzer gibt einen Text ein und es werden verschiedene Bilder generiert, die dazu passen. Prinzipiell spielt es dabei keine Rolle, ob der User bereits eine konkrete Idee für die Story im Kopf hatte oder nicht. In jedem Fall sind die Bilder visuell so interessant, dass er die Strukturen darin fantasievoll interpretieren kann.

Die verschiedenen Bilder sind aufgrund der Umsetzung des Like-Algorithmus (siehe auch Readme.md) stilistisch sehr variiert. Auch einfache Ideen des Nutzers werden durch das Anhängen von Adjektiven (z.B. "futuristic", "abandoned", "japanese", ...) schrittweise weitergesponnen.

Wenn der Nutzer z.B. lediglich nach "castle" sucht, dann sieht die gewünschte Burg beispielsweise im einen Bild futuristisch, im nächsten Bild verlassen und heruntergekommen, und im nächsten Bild japanisch aus.

WoDone begleitet den Nutzer somit auf der gesamten Reise seiner Ideenfindung, und bringt ihn durch stetig variierende Vorschläge immer ein Stück weiter. Aus dem simplen Grundgedanken "research facility" kann nach ein paar Minuten schon der Generations-Prompt "burning underground overgrown research facility" geworden sein (siehe Bild 5.). Dieser Text erzählt schon eine deutlich konkretere, lebhafte Geschichte.

Dadurch, dass man den Bildern live bei der Generation zuschauen kann, entsteht regelrecht Vorfreude beim Benutzer, herauszufinden wie das Ergebnis aussehen wird. Gleichzeitig erzählt der Generationsprozess stets auch selbst eine Geschichte, wie sich einzelne Elemente im Bild nach und nach durchsetzen, während andere verschwinden. Durch die Vorfreude und das Erzählen ständig neuer Geschichten wird die Benutzung von WoDone gamifiziert, was die Kreativität weiter voranbringt.

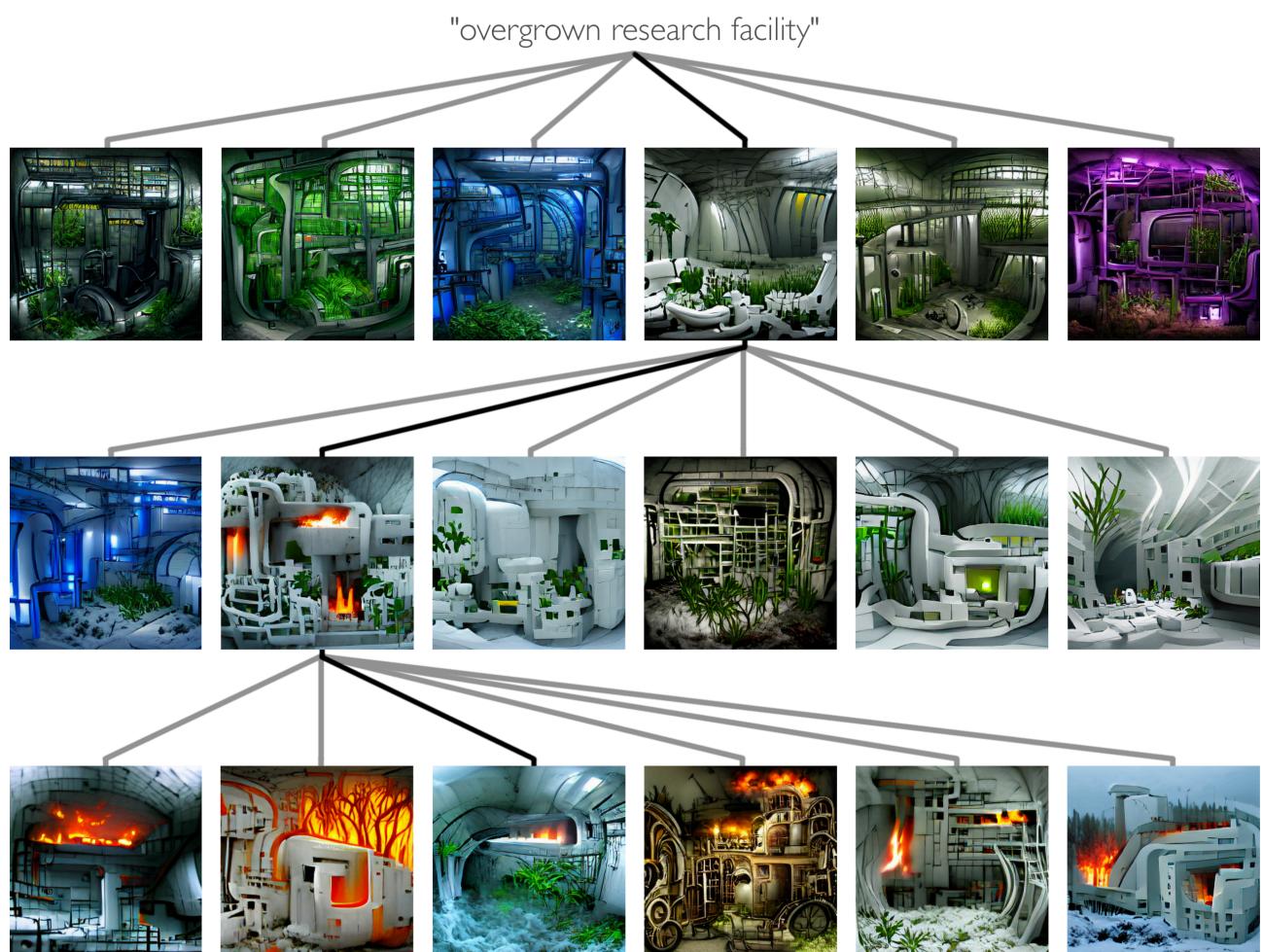


Bild 5: Der Generationsprozess kann mit einer Baumstruktur visualisiert werden. Bei jeder Generierung werden der Suchanfrage neue Adjektive hinzugefügt, und das vom Nutzer bevorzugte wird beibehalten. Auf diese Weise wird das Bild immer detailreicher.

## Einordnung nach Margaret Boden

Die Kreativität in den beiden genannten Bereichen lässt sich nach Margaret Boden wie folgt klassifizieren:

Einerseits die Kreativität des Programms selbst, "Combinational Creativity". Ob man neuronalen Netzwerken Kreativität zuschreiben kann ist umstritten. Jedoch kann man sagen, dass ein neuronales Netz nachdem es trainiert wurde eine "eigene" Interpretation der Daten hat. Die daraus folgenden neu generierten Bilder sind sogesehen eine Kombination der erlernten Merkmale und Verbindungen.

Das gesamte Ziel der Anwendung war die Unterstützung im kreativen Prozess des Nutzers. Diese Kreativität ließe sich nach Margaret Boden als "Explorative Creativity" bezeichnen. Die Grundidee dabei ist das Entwickeln neuer Konzepte innerhalb gegebener Rahmenbedingungen. Diese Rahmenbedingungen setzt der Nutzer selbst durch Input-Variablen im Suchfeld des Produkts und beeinflusst sie später durch Likes um den entstehenden Output weiter in Richtung seiner Vorstellungen zu ändern. Damit regt das Tool die explorative Kreativität des Nutzers visuell an indem es ihn motiviert anhand der entstehenden Concept Art neue Spielwelten zu durchdenken.

### 3. Arbeitsteilung im Team

Um WoDone erfolgreich umsetzen zu können, teilten wir uns im Team zunächst in zwei Gruppen auf: Einer Gruppe “Frontend”, die primär für die Benutzeroberfläche zuständig war, und einer Gruppe “Backend”, die primär für die Bildgenerierung selbst zuständig war. In der Endphase des Projekts wurde diese Gruppenstruktur dann schrittweise aufgelöst und alle arbeiteten gemeinsam an den verbliebenen Aufgaben.

Eine besondere Herausforderung bei der Arbeitsaufteilung im Team war, dass Herr Gollreiter und Herr Wittgenstein als Studenten der Ingenieurwissenschaften deutlich weniger Programmiererfahrung hatten als Herr Höpp und Herr Stöffler. Aufgrund des sehr engen, gegebenen Zeitraums, konzentrierte sich die Arbeit der beiden trotz sehr schneller Lernfortschritte eher auf Research, Dokumentation, Konzept und Design. Dennoch programmierten Herr Gollreiter und Herr Wittgenstein immer wieder erfolgreich kleinere Funktionen und Anpassungen. Da diese oft von Herrn Höpp oder Herrn Stöffler noch optimiert wurden, wurden die entsprechenden Commits leider häufig nicht direkt übernommen und gemerget, sondern der Code aus Zeitgründen oft direkt in andere Branches kopiert. Einige der ursprünglichen Branches von Herrn Gollreiter und Herrn Wittgenstein befinden sich noch in Gitlab.

#### Frontend

Der Gruppe “Frontend” gehörten Herr Höpp und Herr Wittgenstein an.

Zu den ersten Aufgaben der Gruppe zählte, das Design und User-Experience der Anwendung festzulegen, also zu bestimmen, auf welche Art und Weise der Nutzer mit WoDone interagieren kann. Anschließend war die Gruppe damit befasst, diese Entscheidungen in Code umzusetzen und das Frontend zu entwickeln. Hinzu kam die Definition der WoDone-API und die Entwicklung des Flask-Backends, nachdem festgelegt worden war, dass die Bildgenerierung in einen eigenständigen und unabhängigen Dienst ausgelagert wird, welcher ausschließlich über die Datenbank mit dem Frontend in Verbindung steht.

Während Herr Wittgenstein sich intensiv mit dem Design und der Optimierung der User-Experience auseinandersetzte, lag der Schwerpunkt von Herrn Höpp auf der Umsetzung des Frontends in Programmcode.

#### Backend

Der Gruppe “Backend” gehörten Herr Stöffler und Herr Gollreiter an.

Die erste Aufgabe der Gruppe war es, ein passendes text-to-image Netz zu finden und ein Verständnis dafür zu entwickeln, wie es funktioniert und wie man es beeinflussen kann. Dabei war es besonders wichtig aus der Vielzahl an Kandidaten ein neuronales Netz zu finden, welches besonders gut mit Texteingaben umgehen kann, die Landschaften und bestimmte Szenarien beschreiben. Ebenso wichtig war es, herauszubekommen, welche Texteingaben gut in diesem

Netz funktionieren und wie sich bestimmte Parameter (z.B. die Bilddimensionen) auf die Generierung auswirken. Für diese Aufgabe war Herr Gollreiter verantwortlich.

Nach der Entscheidung für das Netz war die nächste Aufgabe das gefundene Netzwerk in unser Programm einzubetten. Wie oben schon beschrieben ist ein Hauptbestandteil unseres Programms der “Like-Algorithmus”. Dieser bietet dem Nutzer interaktiv die Möglichkeit dem Netzwerk Feedback über das generierte Bild zu geben. Die Herausforderung hier war es einen Algorithmus zu entwickeln, der nur durch einen “like” ein immer präziseres Bild liefert. Für diesen Teil des Backend-Development war Herr Stöffler verantwortlich.

### **Endphase des Projekts**

Nachdem die beiden Gruppen “Frontend” und “Backend” ihre primären Aufgaben erfüllt hatten, es also einen fertigen Prototypen des Frontends sowie eine funktionierende Bildgenerierung gab, arbeiteten alle zusammen an der Zusammenführung von Frontend und Bildgenerierung.

Herr Höpp und Herr Stöffler befassten sich in diesem Zuge intensiv dem Deployment der Software, also mit Docker bzw. Docker-Compose, um zunächst alle Dienste zuverlässig gemeinsam starten zu können. Anschließend implementierten sie die noch fehlenden Softwarebausteine, damit nach Anfrage durch das Frontend auch tatsächlich Bilder generiert und angezeigt werden können.

Herr Gollreiter und Herr Wittgenstein bereiteten in der Endphase die Präsentation vor und dokumentierten das Projekt.

## 4. Technische Details

Eine besondere Herausforderung bei der Umsetzung von WoDone war die hohe Rechenleistung, welche das neuronale Netz benötigt. Ohne Grafikbeschleunigung kann Generierung eines einzelnen Bildes länger als eine Stunde dauern, weswegen wir eine Entwicklungsumgebung mit Grafikbeschleunigung benötigten. Diese wurde uns dankenswerterweise vom Lehrstuhl zur Verfügung gestellt. Indem wir die Auflösung der generierten Bilder auf 300x300 Pixel beschränkten, konnten wir so akzeptable Renderzeiten von circa einer Minute erreichen. Aufgrund des enormen Ressourcenbedarfs auch an Grafikspeicher mussten wir zudem die Anzahl der gleichzeitig generierten Bilder auf eins beschränken. Dies erforderte zusätzlichen Programmieraufwand, da eine Warteschleife implementiert werden musste. Für ein wirklich flüssiges Nutzungserlebnis insbesondere bei mehreren Nutzern wäre eine deutlich höhere Rechenleistung (meherere High-End-GPUs) notwendig.

Eine detaillierte Erläuterung der technischen Umsetzung und der Funktionsweise von WoDone sowie Quellenhinweise für Code Dritter können der Readme.md-Datei und den ausführlichen Kommentaren in allen relevanten Dateien des Quelltextes entnommen werden.