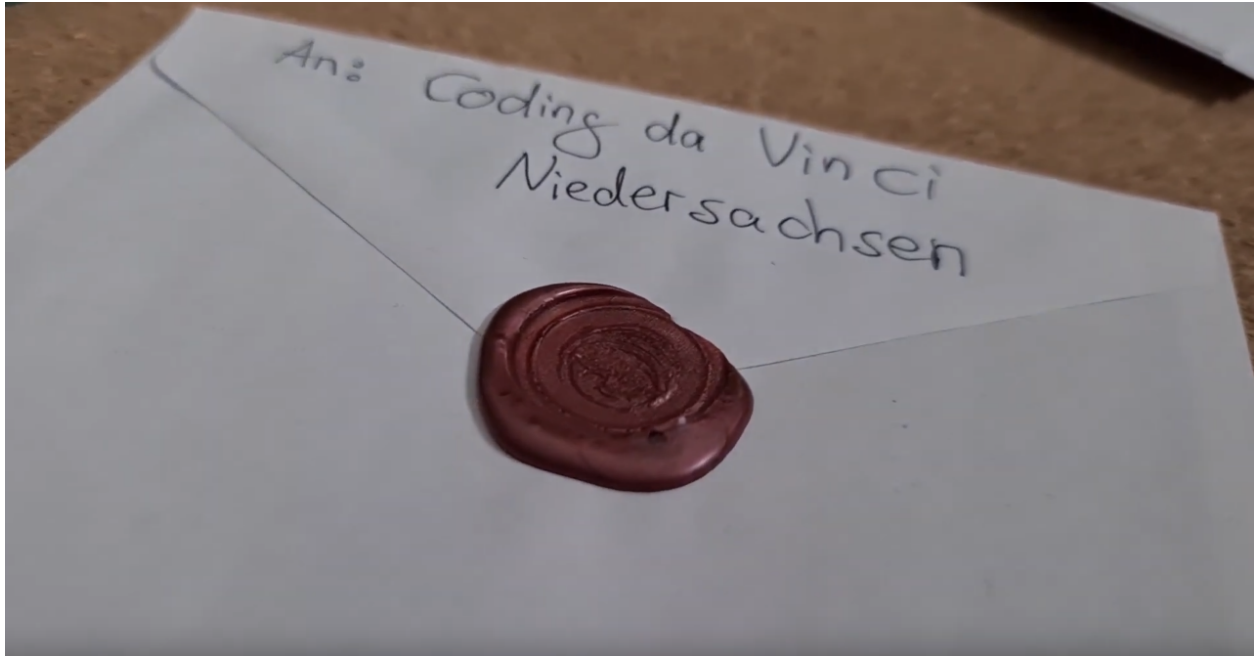


FabSeal

Abschlussbericht



Joana Bergsiek | Leonard Geier | Lisa Ihde
Tobias Markus | Dominik Meier | Paul Methfessel

September 2021

Coding Da Vinci Stipendium Niedersachsen

Inhaltsverzeichnis

[Inhaltsverzeichnis](#)

[Kurzzusammenfassung](#)

[Neue Datensätze](#)

[Siegelsammlung Paul Arnold Grun](#)

[Gemmen Abdrücke](#)

[Siegelmarken von Gymnasien](#)

[Pennies aus dem Vereinigten Königreich](#)

[Münzen aus China](#)

[Technische Umsetzung](#)

[FabSeal Browse](#)

[Vorstellung](#)

[Technische Umsetzung](#)

[Mögliche Erweiterungen](#)

[FabSeal Create](#)

[Vorstellung](#)

[Technische Umsetzung](#)

[Mögliche Erweiterungen](#)

[FabSeal Remix](#)

[Vorstellung](#)

[Technische Umsetzung](#)

[Mögliche Erweiterungen](#)

[Ausblick](#)

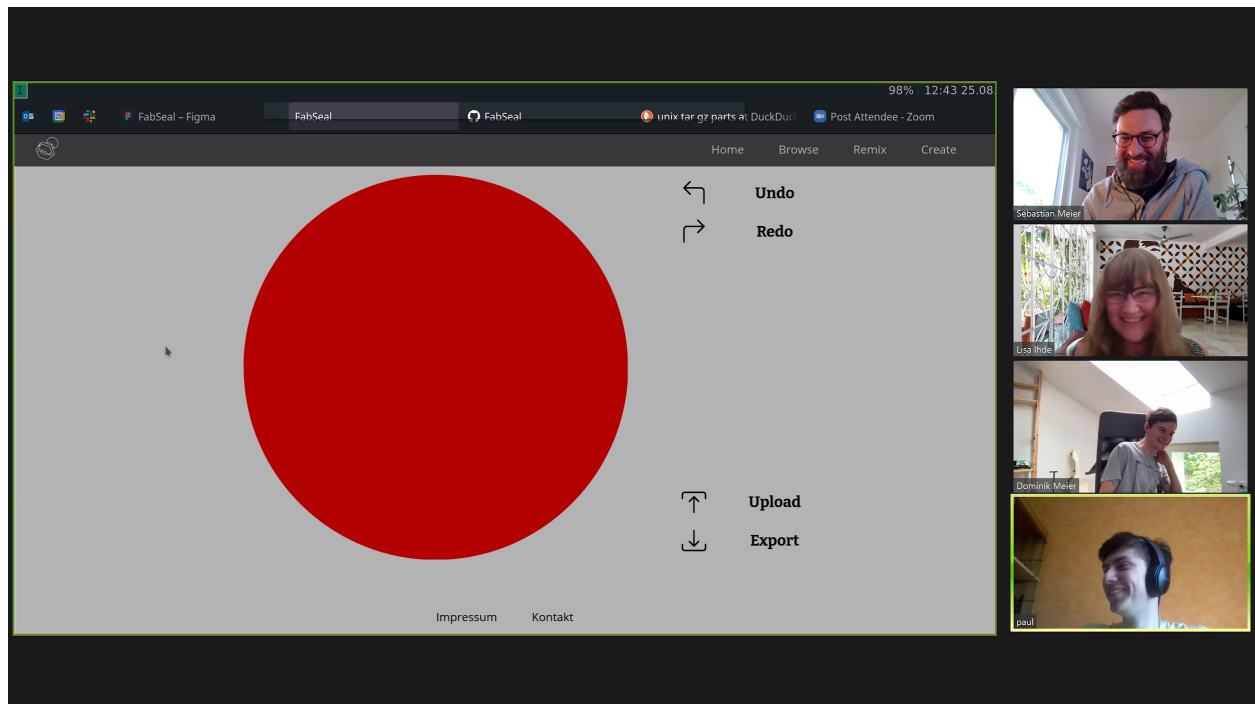
[User Testing](#)

[Fortführung / Hosting des Projekts](#)

[Linksammlung](#)

Kurzzusammenfassung

FabSeal ist 2020 im Rahmen von Coding da Vinci Niedersachsen mit der Absicht entstanden, die Siegelbilder der Paul Arnold Grun Sammlung zu Analysieren und 3D-Modelle aus ihnen zu generieren. Über die letzten Monate haben wir dieses Projekt weiterentwickelt, um die bestehende Funktionalität zu verbessern und neue zur Verfügung zu stellen. In diesem Bericht erläutern wir, wie wir das Projekt um neue Datensätze erweitert haben, wie wir diese Daten für den Nutzer erkundbar machen (FabSeal Browse) und auf welche Weise sogar eigene Motive verwendet werden können, um ganz individuelle Siegel zu erschaffen (FabSeal Create). Zudem können vorhandene Elemente zu neuen Siegel kombiniert werden (FabSeal Remix) . Wir hoffen, auf auf diese Art und Weise die faszinierende Welt der Siegel greifbarer zu machen.



Neue Datensätze

Als ein Ziel des Stipendiums haben wir angegeben, FabSeal um weitere Datensätze zu erweitern. Dabei haben wir uns nicht nur auf ehemalige Coding-Da-Vinci-Datensätze beschränkt, sondern auch versucht, Datensätze von anderen Websites aufzubereiten, falls diese in einer kompatiblen Lizenz vorliegen. Insgesamt bieten wir nun fünf Datensätze aus den vier verschiedenen Kategorien Siegel, Gemmen, Wappen und Münzen zum Betrachten und Fabrizieren an.

Im Folgenden werden wir alle Datensätze einmal kurz vorstellen, und auf Besonderheiten bei der Verarbeitung eingehen.

<p style="text-align: center;">Siegelsammlung Paul Arnold Grun¹</p> 	<p>Die Siegel aus der Siegelsammlung von Paul Arnold Grun waren der erste Datensatz den wir verarbeitet haben. Herausforderungen waren hier vor allem die Extraktion der Siegelfläche, da die Siegel wie links dargestellt auf Papier aufgeklebt wurden. Mit verschiedenen Heuristiken waren wir jedoch in der Lage, für die meisten Siegel ein sehr gutes Ergebnis zu erzielen. Der Datensatz ist als CC BY-SA 4.0 Niedersächsische Staats- und Universitätsbibliothek Göttingen freigegeben.</p>
<p style="text-align: center;">Gemmen Abdrücke²</p>	<p>Gemmen sind Schmucksteine, die im Fall dieser Sammlung antike Szenen zeigen. Bei diesem Datenset handelt es sich jedoch nicht um die originale Schmuckstücke, sondern Gipsabdrücke davon. Für unseren Shape-From-Shading-Algorithmus ist das von Vorteil, da dadurch das Material immer gleich beschaffen ist. Zudem erzeugt die matte Oberfläche gute Konturen. Der Datensatz ist als CC BY-SA 4.0</p>

¹ <https://codingdavinci.de/de/daten/siegelsammlung-paul-arnold-grun>

² <https://codingdavinci.de/daten/goetter-und-helden-im-miniaturformat>

	<p>Antikensammlung der Friedrich-Alexander-Universität Erlangen-Nürnberg freigegeben.</p>
<p>Siegelmarken von Gymnasien³</p> 	<p>Die Siegelmarken haben wir ausgesucht, weil und die verschiedenen Farben angesprochen haben, da viele unserer Datensätze bis jetzt monochromatisch waren. Das besondere an diesem Datensatz ist, dass die Siegel an sich zwar keine Tiefe besitzen, der Shape-From-Shading-Algorithmus jedoch trotzdem ein Höhenbild erzeugen kann, welches dann ähnlich wie ein Relief den verschiedenen Farben verschiedene Höhen zuordnet. Da der Datensatz mit insgesamt circa 43.000 Siegel zu groß für eine Einbettung wäre, haben wir uns für eine Teilmenge entschieden, nämlich Wappen von Gymnasien. Für die Höhenbild-Erzeugung wäre wahrscheinlich eine simple Codierung von verschiedenen Farben zu verschiedenen Höhen eine einfachere und effizientere Lösung. Problematisch war auch die schlechte Qualität der Siegel. Der Datensatz ist public domain des veikkosarchiv.</p>
<p>Pennies aus dem Vereinigten Königreich</p>	<p>Die beiden letzten Datensätze sind aus Münzen, und wurden von der Website des Kenom-Projektes heruntergeladen. Durch die Schlagworte "Großbritannien" und "Penny" haben wir hier eine Filterung vorgenommen.</p>

³ <https://codingdavinci.de/daten/siegelmarken-zwischen-1850-1945> (Auswahl)

 <p>4</p>	<p>Der Datensatz besteht aus Münzen, welche als CC BY-NC 4.0 Niedersächsisches Landesmuseum freigegeben sind.</p>
<p>Münzen aus China⁵</p> 	<p>Auch dieser Datensatz aus chinesischen Münzen wurde durch eine Filterung des Kenom-Dataset generiert, und zeigt verschiedene Arten von Münzen. Für eine korrekte Erzeugung eines 3D-Modells, welches das Loch in der Mitte der Münze mit “ausstanzt”, hätten wir einen weiteren Schritt in der Nachbearbeitung hinzufügen müssen. Auch mit dieser Version des Shape-From-Shading-Algorithmus wird jedoch einfach eine glatte Fläche in der Mitte erzielt. Der Datensatz ist als CC BY-NC-SA 3.0 DE Kulturstiftung Sachsen-Anhalt, Kunstmuseum Moritzburg Halle freigegeben.</p>

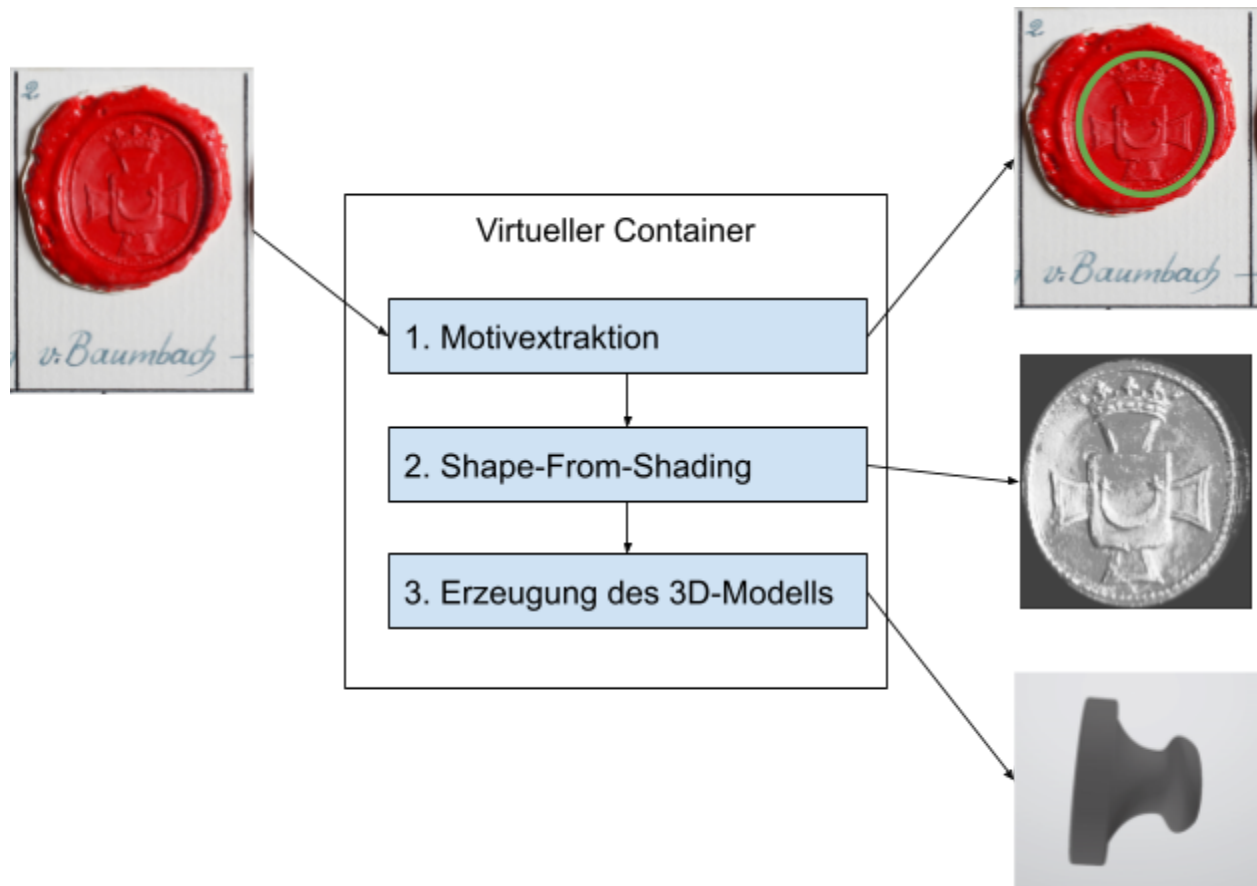
Technische Umsetzung

Insgesamt sind unsere vollen Datensätze am Ende mit einer Größe von circa 45 GB immens groß, was auch zu Schwierigkeiten in der Verarbeitung geführt hat. Um die langen Laufzeiten der Pipeline zu ermöglichen, haben wir uns einen gehosteten Server gemietet, auf dem wir diese dann auch über Nacht laufen lassen konnten.

Zudem haben wir verschiedene Möglichkeiten eingebaut, die Pipeline an den Datensatz anzupassen. Die Segmentierung des Waxes war zum Beispiel bei den Münzdatensätzen nicht notwendig, und konnte somit bei der Verarbeitung übergangen werden.

⁴ <https://www.kenom.de/> (eigene Auswahl, Stichworte “Großbritannien”, “Penny”)

⁵ <https://www.kenom.de/> (eigene Auswahl, Stichwort “China”)



Die Verarbeitung der Datensätze haben wir in einen virtuellen Docker Container ausgelagert, da es so einfacher war die Abhängigkeiten der verschiedenen Module bereitzustellen.

Wir ein Bild verarbeitet, durchläuft es zuerst den Schritt Motivextraktion. Hierbei wird versucht die Motivfläche des Siegels zu erkennen. Das Motiv wird dann ausgegeben, um es zum Beispiel auf der Webseite anzeigen zu können.

Zudem wird es in den Shape-From-Shading-Algorithmus weitergegeben, welcher das Höhenbild erzeugt. Hier ist in Graustufen kodiert, wie weit nach oben das Motiv am gegebene Pixel "herausgedrückt" ist. Dieses Bild nutzen wir zum Beispiel für die Generierung der interaktiven Ansichten auf der Webseite.

Zudem wird es zur Erzeugung des 3D-Modells benötigt. Dies geschieht mit Hilfe von Blender⁶, einer 3D-Modellierungssoftware, die wir per Skript automatisiert bedienen.

Zusätzlich zu der Verarbeitung der Bilder sind noch einige Arbeitsschritte notwendig,

⁶ <https://www.blender.org/>

wie die Extraktion von Metadaten der Bilder, und die Formatierung in das vom Server erwartete Format. Hierzu haben wir größtenteils kleine Python-Skripte benutzt, welche sich in unserem Pipeline-Repository befinden.

FabSeal Browse

FabSeal Browse ist der Name für das Explorieren der bereitgestellten Datensätze. Es ermöglicht Nutzer sich die einzelnen Elemente der Datensätze anzuschauen, und die generierten 3D-Modelle anzuschauen und herunterzuladen.

Vorstellung

In der Übersicht von FabSeal Browse werden dem Nutzer sämtliche Datensätze auf Cards angezeigt, die auf der Webseite verfügbar sind. Wählt der Nutzer einen aus, werden ihm alle dazugehörigen Siegel (oder Münzen, Gemmen oder Marken), ebenfalls auf Cards, gezeigt. Bei größeren Datensätzen werden die Inhalte auf mehrere Seiten verteilt. Jede Card enthält den Namen, die Tags (etwa zu sehende Motive) und ein Bild von dem dazugehörigen Siegel.

Wählt der Nutzer nun eines der Siegel aus, wird er zu dessen Detailansicht weitergeleitet. Dieses enthält zusätzlich zu den Informationen auf der Card auch ein bewegbares 3D-Modell des Siegels, einen Link, über den man dieses samt Stempelgriff in einem ZIP-Archiv herunterladen kann, und Lizenzinformationen.

Technische Umsetzung

Die Cards der Übersichts- und Datensatzansichten werden dynamisch mit Vue generiert. Das 3D-Modell auf der Detailseite wird hingegen mit Three.js in Echtzeit gerendert. Beinahe alle Daten werden statisch vom Backend bereitgestellt — eine Ausnahme stellt dabei das ZIP-Archiv dar, das der Nutzer herunterladen kann. Da der Stempelgriff und die dazugehörigen Lizenzinformationen bei jedem Modell gleich sind, sparen wir Speicherplatz, indem wir die Archive nicht im Vornherein, sondern erst wenn sie angefordert werden, generieren.

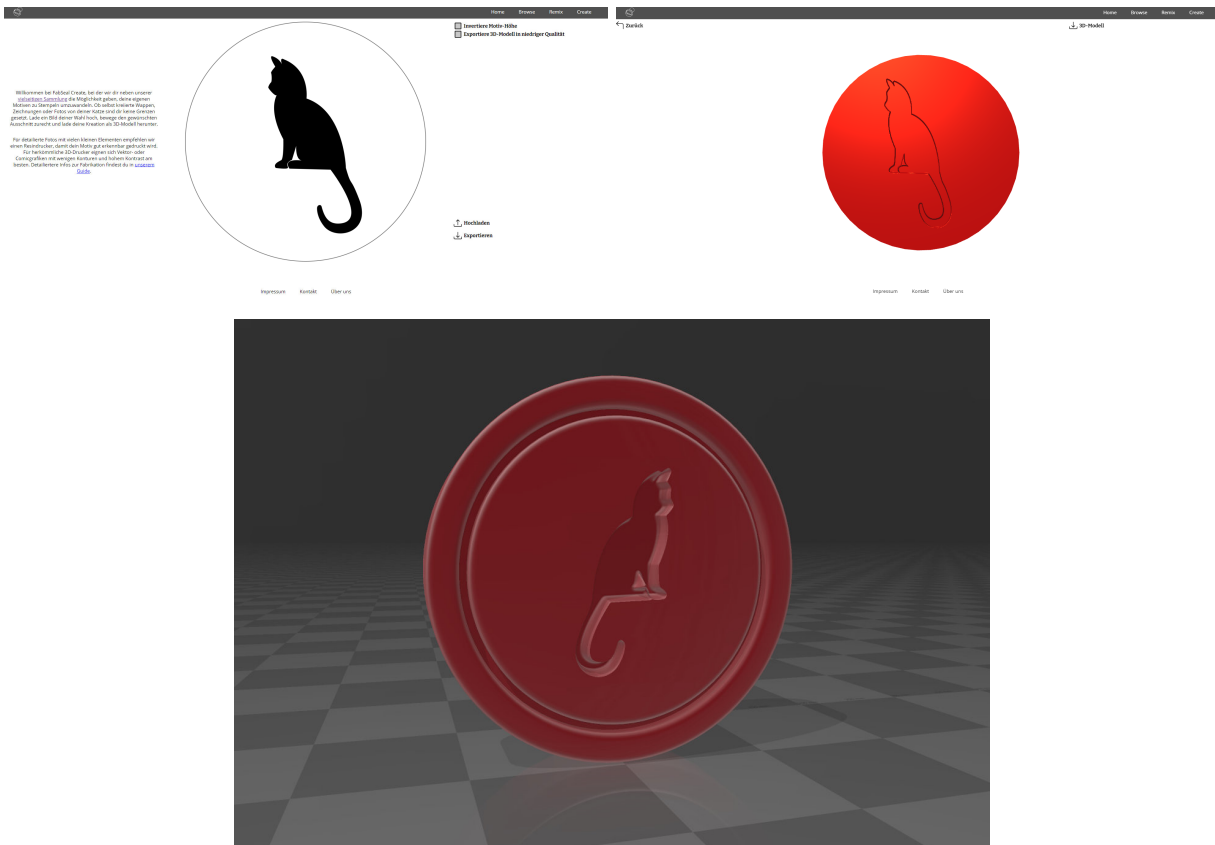
Mögliche Erweiterungen

Eine Filterfunktion wäre insbesondere hilfreich für Expertennutzer, die wissen, nach welchen Siegeln und Motiven sie suchen. Für Nutzer, die sich mit den Datensätzen weniger gut auskennen, wäre es hingegen sinnvoll, andere Darstellungsweisen als die gegenwärtige tabellarische anzubieten. FabSeal hatte beispielsweise ursprünglich eine Cluster-Ansicht, in der Siegel, die einander ähnlich sind, nah beieinander platziert sind. Das Problem hierbei ist jedoch, eine geeignete Metrik zu finden, mit der man die Distanzen zwischen Siegeln bestimmen kann.

FabSeal Create

Bei unseren Designtreffen mit Sebastian kristallisierte sich schnell der Wunsch heraus, Nutzer*innen das Kreieren eigener Stempelmotive zu ermöglichen. Neben FabSeal Browse, welches Daten- und Kulturinteressierte anspricht, ist FabSeal Create an Maker*innen und 3D-Druck-Interessenten*innen gerichtet. Durch die simple Benutzerschnittstelle eines Foto Uploads sollen dabei Anfänger*innen als auch Expert*innen in dem Gebiet für neue Projektideen inspiriert werden. Über diesem Wege generieren wir ebenfalls neue Augen für die vorhandenen Datensammlungen. Besonders in Anbetracht Schüler*innen mit dem Projekt vertraut zu machen, sehen wir in FabSeal Create ein großes Begeisterungspotenzial für den 3D-Druck.

Eine Veranschaulichung von FabSeal Creates Workflow anhand einer Katzengrafik.



Vorstellung

FabSeal Create priorisiert in seinem Design Simplizität und schnelles Ausprobieren, um den Anforderungen der genannten Nutzergruppe gerecht zu werden. Dementsprechend steht die Eingabe für den Fotoupload im Fokus - Einstellungen gibt es über Checkboxes nur zwei: Ob die Höhe des Motiv auf dem Stempel invertiert werden soll (hohe Stellen werden zu niedrigen und vice versa) und ob das finale 3D Modell eine geringere Qualität haben soll. Die kleinere Vertices-Anzahl verbessert die Ergebnisse von Fotografien mit hausüblichen 3D-Druckern auf Kosten des Detailgrads.

Nachdem ein Foto hochgeladen wurde, können Nutzer*innen in einer Detailansicht ihr Ergebnis betrachten. Diese Detailansicht ist identisch zu der von FabSeal Remix. Dort kann der selbst kreierte Stempel dann heruntergeladen werden. Wie in der technischen Umsetzung erwähnt, ist FabSeal Create für Comiczeichnungen, als auch Vektorgrafiken, mit hohen Kontrasten am besten geeignet.

Technische Umsetzung

FabSeal Create erfordert den Großteil unserer Pipeline-Schritte und war eine der größten technischen Herausforderungen der letzten Wochen. Während eine nacheinanderfolgende Ausführung der einzelnen Schritte offline auf der eigenen Maschine schnell getan ist, standen wir hier vor der Herausforderung, wie wir durch Webanfragen Python oder Blender Skripte starten.

Wir entwickelten [fabseal-micro](#), ein Backend, welches mithilfe von Script Embedding Webanfragen zu den jeweiligen Skripten übersetzt und die separaten Ergebnisse jeweils abrufbar zur Verfügung stellt. Es kam ebenfalls ein [Blender-Preprocessing](#) hinzu. Da Shape From Shading komplizierte Parameter für ein optimales Ergebnis benötigt, welches FabSeal Create weniger einladend gestaltet hätten, wird das hochgeladene Bild mit dem neuen Preprocessing-Algorithmus zu einem Höhenbild umgewandelt. Dieser wandelt das Bild lediglich in Graustufen um und erstellt einen kreisförmigen Ausschnitt des Motivs mit weißem Rand, welcher für unser [Blender-Skript](#) optimale Ergebnisse liefert. Beispielbilder, weswegen dieser Schritt so wichtig sind, lässt sich an [dessen readme](#) betrachten.

Kein Shape From Shading zu benutzen bringt natürlich schlechtere Ergebnisse für Fotografien mit sich, da beispielsweise der Lichtwinkel bei der Umwandlung nicht berücksichtigt wird. In unseren Datensätzen sind die Lichtwinkel pro Datensatz gleich, weswegen Shape From Shading, einmal konfiguriert, innerhalb eines Datensatzes universell einsetzbar war. Falsch eingegebene Lichtwinkel führen im Allgemeinen zu

schlechteren Ergebnissen als eine simple Umwandlung, für die wir uns mit Blick auf die limitierte Zeit letztendlich entschieden haben. Allerdings sind Lichtwinkel in Comiczeichnungen und Vektorgrafiken nicht vorhanden, weswegen sie mit dem derzeit naiven Preprocessing die besten Ergebnisse erzielen. Eine mögliche Erweiterung von FabSeal Create könnte Shape From Shading mit guten Standardwerten einsetzen, oder dessen Einstellungen für Expert*innen anbieten.

Mögliche Erweiterungen

Eine Idee ist der “Maker-Modus”, welcher weitere Einstellungen für Kenner*innen des 3D-Drucks anbieten soll. Zwar ist unsere Schnittstelle simpel, aber Maker*innen fühlen sich von dieser Simplizität eventuell limitiert, wenn sie näher in die Materie der selbst fabrizierten Stempel eintauchen wollen. Ihre einzige Möglichkeit ist bislang die Post-Prozessierung in Blender. Daraus folgend könnten bereits im Export Einstellungen wie “Reliefhöhe”, “Modellgröße” oder “Vertice-Anzahl” bereitgestellt werden.

Weitergehend wäre eine Kooperation mit Shape From Shading vorstellbar. Momentan wird - wie in der technischen Umsetzung erwähnt - das hochgeladene Foto einfach in Graustufen umgewandelt, da die Parameter von Shape From Shading für Laien verwirrend sein könnten. Für den Anwendungsfall von Fotografien könnten passend eingestellte Parameter, beispielsweise der vorhandene Lichtwinkel, die Ergebnisse des Höhenbildes für Fotografien verbessern.

FabSeal Remix

Um die verschiedenen Motive und die Symbolik der Siegel nicht nur statisch durch Browse zu erfahren, haben wir FabSeal Remix ins Leben gerufen. Es erlaubt Nutzer*innen Elemente aus echten Siegeln zu verwenden, um ihr eigenes, persönliches Siegel zu kreieren. Dieses können sie dann, wie bei FabSeal Create, selbst über einen 3D-Drucker ausdrucken.

Vorstellung

Die Oberfläche von FabSeal Remix soll sowohl unkompliziert sein, als auch zum Experimentieren anregen. In der Mitte befindet sich das bearbeitbare Siegel, welches vor der Benutzung einen kurzen Hilfetext enthält. Auf der linken Seite ist eine Liste an handverlesenen Siegel-Motiven. Durch Selektion kann ein solches auf dem Siegel platziert werden. Motive können jederzeit bewegt, rotiert und skaliert werden. Auch ein Undo- und Redo-Button sind verfügbar, um Fehler schnell wieder rückgängig machen zu können. Weiterhin kann jederzeit eine Vorschau des 3D-Siegels angesehen werden. Sobald das Siegel fertig ist, kann es exportiert und mit einem 3D-Drucker physisch hergestellt werden.

Technische Umsetzung

Das Feature FabSeal Remix hat drei größere technische Herausforderungen:

1. Die Erstellung der Motive
2. Das interaktive Siegel
3. Der Export der nutzergenerierten Anordnung zu einem druckbaren Modell.

Das erste Problem haben wir bisher manuell gelöst, indem wir in Adobe Photoshop ein Motiv durch eine Maske auswählen. Durch diese Maske kann das Motiv sowohl aus dem Originalbild, als auch aus dem Höhenbild ausgeschnitten werden.

Für das interaktive Siegel kommt uns eine Bibliothek namens fabric.js zu gute. Sie erlaubt Nutzer*innen das freie Platzieren von grafischen Elementen auf einer Canvas. Sobald ein Motiv ausgewählt wurde, übergeben wir es an fabric.js. Nur die Undo- und Redo-Funktion mussten wir selbst implementieren.

Um das erstellte Siegel zu einem 3D-Modell zu konvertieren, brauchen wir zuerst das Höhenbild der Siegels. Dieses bauen wir, indem wir die platzierten Motive durch ihre passenden Höhenbilder ersetzen und das Siegel rendern. Das fertige Bild wird nun an

fabseal-Micro gegeben, was uns über das Blender-Script das 3D-Modell erzeugt.

Mögliche Erweiterungen

In unseren Brainstorming-Treffen hatten wir jede Menge weitere Ideen für Features, die wir in FabSeal Remix einbauen könnten. Es ist allerdings zu beachten, dass diese Nutzer*innen zwar neue Möglichkeiten und mehr Kontrolle über das Resultat geben, aber sich die Komplexität der Oberfläche erhöhen würde.

Zum einen könnten neue Werkzeuge mehr Bearbeitungsmöglichkeiten eröffnen: Beispielsweise ein Zeichenstift und Radierer, mit dem eigene Motive gemalt werden könnten. Für das gleiche Feature müssten Nutzer*innen bislang auf andere Mal-Tools wie Paint zurückgreifen und ihr Ergebnis mit Fabseal Create hochladen.

Um mehr Kontrolle über die Auswahl der Motive zu haben, kann die Liste eine Such- und Filterfunktion bekommen. Einstellungen für den Hintergrund könnten die Farbe, Form und Art des Siegels ändern (was allerdings zusätzliche technische Herausforderungen mit sich zieht).

Ausblick

User Testing

Wir haben mit vier Personen am 7. und 8. September 2021 unser neues Webdesign testen lassen. Die vier Personen haben verschiedene Personas von uns abgedeckt, sodass wir von Digital Natives hinzu technischen Anfänger*innen alles dabei hatten. Vor allem waren sie alle kulturbegeistert und wir hatten ein gutes Gefühl, dass unser Projekt begeistern wird.

Wir haben uns für das User Testing ein paar Aufgaben überlegt, die gelöst werden sollten. Nachfolgend sind die gestellten Aufgaben aufgelistet:

1. Finde heraus worum es geht.
2. Wofür steht "FabSeal" konkret?
3. Finde heraus, wie man uns kontaktiert.
4. Lade das Siegel eines Gymnasiums herunter.
5. Sieh dir das Original Gemmen Bild vom Skarabäus an.
6. Finde heraus, was du beim 3D-Druck beachten musst.
7. Lade ein 3D-Modell eines beliebigen Siegels aus der Sammlung von Grun herunter.
8. Erstelle mit Create ein eigenes Siegel und lade das 3D-Modell davon herunter.

Das Testen startete damit, dass wir die Nutzer*innen erstmal selbständig die Webseite entdecken ließen und sie sagen sollten, was sie denken. Danach haben wir die Aufgaben nacheinander gestellt.

Wir stellten fest, dass die Nutzerführung der Webseite durchweg noch nicht so ankommt, wie sie gedacht war, und wir somit an der Nutzerfreundlichkeit arbeiten müssen. Beispielsweise war einigen nicht klar, dass man auf unserer Startseite runterscrollen kann und sie haben daher nicht das FAQ auf Anhieb gefunden. Das FAQ scheint auch zu umfangreich, da meist nicht gelesen wurde, was dort steht. Unter einen der FAQ-Fragen würde man theoretisch zu unserer Anleitung für den 3D-Druck kommen. Dies wurde aber eher überlesen und sie fanden einen anderen Weg dort hin.

Generell kam die 3D-Vorschau und die Create-Funktionalität sehr gut an. Bei der 3D-Vorschau wurde aber bemängelt, dass es schwer sei, dort das Motiv zu erkennen und diese Vorschau verbessert werden sollte. Auch wurde das selbständige 3D-Drucken als eine Hürde wahrgenommen und es wurde sich gewünscht, dass man das Modell

irgendwo bestellen könnte, sodass der*die Nutzer*in es nicht selbst drucken müssten.

Eine Testperson hat vorgeschlagen, dass wir auch fertige, gedruckte Siegelstempel von anderen Besucher*innen auf der Seite zeigen sollten, damit man mehr versteht, was alles möglich ist. Auch sollten wir Hinweise ergänzen, was für Bilder sich am besten eignen bezüglich Hintergrund/Vordergrund wie freigestellten oder homogenen Hintergrund. Generell fielen bei einigen der Tester*innen erhöhte Ladezeiten für die 3D-Vorschau und das Herunterladen des Siegelstempelmodells auf.

Abschließend können wir sagen, dass das Projekt sehr gelobt wurde und auch das Design der Webseite. Wir denken, dass wir mit ein paar gezielten Änderungen die Probleme beseitigen können und würden dann die Testpersonen nochmal kontaktieren.

Fortführung / Hosting des Projekts

Die öffentliche Webseite <https://fabseal.de> wird aktuell auf einem Server des Anbieters Hetzner gehostet. Die monatlichen Kosten belaufen sich auf etwa 6 € (da auf dem Server noch andere Dienste bereitgestellt werden, ist der Anteil nicht direkt zu bestimmen). Theoretisch wäre ein Hosting auch auf einem separaten Server mit geringeren Kosten möglich.

Im Laufe des Projekts haben wir uns zu Fördermöglichkeiten für den langfristigen Betrieb von <https://fabseal.de> erkundigt, allerdings ist eine derartige Förderung nach heutigem Wissensstand nicht möglich. Da der Server unabhängig von FabSeal sowieso genutzt wird, ist zumindest in den nächsten Jahren die Bereitstellung von FabSeal gesichert.

Für die Speicherung der Datensätze in voller Auflösung werden etwa 50 GB benötigt, was sich in 2 € an monatlichen Zusatzkosten niederschlägt. Aktuell prüfen wir noch, ob es möglich ist die Datensätze effizienter abzuspeichern, um die langfristigen Kosten zu reduzieren.

Linksammlung

Website des Projekts: <https://fabseal.de/>

Code-Repository: <https://github.com/Siegler-von-Catan>

Coding-Da-Vinci Seite: <https://codingdavinci.de/projekte/fabseal>