

Titel

mit einem eventuell
ganz langen Untertitel

EXPOSÉ ZUR ABSCHLUSSARBEIT

von

Chrsitain Alexander Poplawski

Erster Prüfer: Prof. Dipl. Des. Christian Noss
Technische Hochschule Köln

Zweiter Prüfer: Dipl. Des. Liane Kirschner
Railslove GmbH

Köln, im Mai 2017

1 Thema der Abschlussarbeit

Thema der Abschlussarbeit soll die Umsetzung und Gestaltung einer Web-App auf Basis von bereits im Praxisprojekt erarbeiteten Konzepten sein. Die Web-App soll vorrangig Studenten dabei unterstützen, in ihren Gestaltungen von Artefakten, die im Hochschulalltag benötigt werden, eine Grundqualität zu sichern. Ziel der Abschlussarbeit soll es sein, ein funktionierendes und qualitativ hochwertiges Produkt zu erstellen. Dabei stellen sowohl der Weg vom Konzept und Proof of Concept zum fertigen Produkt als Ablauf, als auch die Umsetzung in den verschiedenen Bereichen die Themengebiete dar, die in der Abschlussarbeit behandelt werden sollen.

Im Praxisprojekt wurde versucht, in verschiedenen Bereichen der visuellen Gestaltung Regeln und Richtlinien zu definieren, die später von einem Computerprogramm bewertet werden können. Dieser Versuch war in verschiedenen Bereichen unterschiedlich erfolgreich. Um die Konzeptarbeit in der Abschlussarbeit möglichst gering zu halten, soll sich in dieser hauptsächlich auf die bereits gut abgedeckten Bereiche konzentriert werden. Eine Auflistung dieser Bereiche findet sich in Unterabschnitt 1.1: "Genereller Aufbau der Anwendung". Weiterhin wurde im Bereich Typografie ein Proof of Concept erstellt, auf dem aufgebaut werden kann.

Im Folgenden sollen die sich für die Abschlussarbeit ergebenden Themengebiete kurz erläutert werden.

1.1 Genereller Aufbau der Anwendung

Im Rahmen des Praxisprojektes wurden bereits grundlegende Konzepte zu Inhalten der Anwendung in verschiedenen Bereichen erstellt. Da hier nicht in allen Bereichen zufriedenstellende Inhalte gefunden werden konnten, sollen diese als erster Schritt der Abschlussarbeit auf ihre Umsetzbarkeit überprüft werden. Weiterhin muss festgestellt werden, welche Gebiete gegebenenfalls noch weiter vertieft werden müssen.

Die wichtigsten Beispiele für noch nicht hinreichend definierte Bereiche stellen hier der Einstiegspunkt und der Endpunkt der Anwendung dar. Viele der Kernbereiche der Anwendung greifen auf die verschiedenen Limitationen zurück, die

mit dem Gestaltungsziel des Nutzers (Plattform, Medium etc.) einher gehen. Diese müssen im ersten Schritt vom Nutzer definiert werden. Hier soll definiert werden, welche Informationen benötigt werden und wie der Nutzer diese der Anwendung mitteilen soll. Auch muss geklärt werden, in welcher Form die Anwendung Informationen an den Nutzer weiter gibt. So werden dem Nutzer bereits während der Nutzung Informationen durch seine Interaktion mit der Anwendung mitgeteilt, jedoch besteht bisher kein Konzept, nach dem diese dem Nutzer am Ende der Anwendung noch einmal gebündelt zur weiterverwendung angeboten werden.

In einer ersten funktionierenden Version der Anwendung sollten die folgenden Bereiche enthalten sein:

- Einstieg
- Typographie
- Layout & Grid
- Farben
- Outcomes

Je nach Zeitlichem Verlauf können diese noch um weitere Themengebiete erweitert werden.

1.2 Technologien und Architektur

Auch die Architektur der Anwendung und die verwendete Technologie [Technologie hier das richtige Wort?] soll diskutiert werden.

Die Anwendung soll mit Hilfe des Framework React.js umgesetzt werden. React.js ist ein Javascript-Framework, das 2013 von Facebook entwickelt wurde. Gackheimer (2015) beschreibt React wie folgt:

[...] React is a way to describe the user interface of an application and a mechanism to change that over time as data changes. React is made with declarative components that describe an interface. React

uses no observable data binding when building an application. React is also easy to manipulate, because you can take the components you create and combine them to make custom components that work as you expect every time because it can scale.

Der Fokus des Frameworks liegt also auf interaktiven User Interfaces, in denen sich die enthaltenen Daten häufig ändern. Da die Anwendung dem Nutzer interaktives Feedback zu seinen Eingaben geben soll, eignet sich React gut für diese Zwecke.

Der Proof of Concept im Praxisprojekt wurde bereits erfolgreich mit Hilfe von React.js umgesetzt und unterstützt die Entscheidung zur Wahl des Frameworks weiter.

Trotzdem sollen in der Abschlussarbeit mögliche andere Frameworks zur Umsetzung diskutiert und verglichen werden. Hier bieten sich vor allem andere Javascript Frameworks wie zum Beispiel Ember.js, Angular.js, Vue.js oder Backbone.js an.

Da für die Anwendung keine persistente Datenhaltung benötigt wird, kann auf eine klassische Client-Server-Architektur verzichtet werden und die komplette Logik kann auf dem Client ausgeführt werden.

Trotzdem sollte auch im Client eine Architektur entworfen werden. Wie oben bereits erwähnt hängen Teile der Anwendung voneinander ab und frühere Schritte beeinflussen spätere. Deshalb soll zu Beginn ein Modell der Anwendung und der Datenhaltung angefertigt werden. Für die Datenhaltung soll auf Redux zurückgegriffen werden.

Redux verwaltet den Zustand (oder *state*) einer Anwendung und sorgt dafür, dass dieser Konsistent bleibt. Diese Konsistenz wird durch drei Grundprinzipien von Redux erreicht:

1. Der Zustand wird nur an einer zentralen Stelle verwaltet.
2. Der Zustand kann nur gelesen und nicht direkt manipuliert werden.
3. Der Zustand kann nur durch *pure functions*¹ verändert werden.

Redux wurde bereits beim Proof of Concept für den Bereich Typographie im Praxisprojekt verwendet. Hier konnte der Nutzer verschiedene Werte ändern, die

sich auf verschiedene DOM-Elemente auswirkten. Redux kontne dafür soegen, dass sich diese Änderungen nicht gegenseitig überschreiben und der Zustand der Anwendung konsistent bleibt.

[Vielleicht diagramm, wie das im PP gemacht wurde?]

1.3 Gestaltung der Anwendung

Auch die Gestaltung der Anwendung soll Teil der Bachelorarbeit sein. Hierbei soll darauf geachtet werden, dass die Gestaltung der Anwendung möglichst in den Hintergrund tritt, um den Nutzer nicht von der Gestaltung abzulenken, die er mittels der Anwendung erstellt. Der Designprozess soll hierbei auf das komponentenbasierte Verhalten von React.js angepasst sein. Ganze Seiten sollen nur in Form von Wireframes erstellt werden, die einzelnen Komponenten auf den Seiten dann im Detail gestaltet werden. Am Ende des Designprozesses soll Verzeichnis von Komponenten entstehen, die in der Anwendung verwendet werden können.

Um dieses Verzeichnis auch in die Anwendung übernehmen zu können, soll auf React-Storybook zurück gegriffen werden. React Storybooks ermöglicht es, einen *living styleguide* für die Verwendung in der Anwendung zu erstellen. Der Code für die verschiedenen Komponenten im Storybook ist dem, der in der Anwendung verwendet wird dabei sehr ähnlich. Im Storybook können außerdem verschiedene Stati und Funktionen der Komponenten abgebildet werden. Abb. X zeigt dies am Beispiel von XYZ.

[Abbildung mit Beispiel von Storybooks]

Durch die komponentenbasierte Form von React bietet es sich an, CSS auch mit in diesen Komponenten zu verarbeiten, um sowohl die funktion als auch das Aussehen in einer in sich geschlossenen Komponente zu speichern. [Satz ist große kaka, muss noch besser werden.]

2 Relevanz des Themas

Die Relevanz der Anwendung an sich wurde bereits im Praxisprojekt erläutert, daher soll hier nur eine Kurzfassung der Beweisführung folgen: [Lindgaard et al] haben gezeigt, dass Menschen sich in nur 50ms ein Urteil über die Gestaltung einer Website bilden. Der erste Eindruck spielt bei Menschen auch bei der späteren Bewertung einer Sache noch eine wichtige Rolle. Von diesem ersten Eindruck lassen sich Menschen nicht ohne weiteres wieder abbringen. Daher ist es wichtig, dass auch Artefakte in Modulen, in denen die Gestaltung nicht bewertet wird, ein solides Design aufweise, um den ersten Eindruck so positiv wie möglich zu gestalten.

Hier seien aber noch zwei weitere Quellen aufgeführt, die die Korrektheit dieser These [ist das eine These?] weiter stützen. [Hier noch die zwei Quellen vorstellen]

Neben der Relevanz des Produktes, das in Rahmen der Arbeit entstehen soll ist aber auch das Meta-Thema der Arbeit zu rechtfertigen: Die Entwicklung von einem Konzept oder PoC zu einem fertigen Produkt. Dieses Thema ist meiner Meinung nach für einen Studenten der Medieninformatik sehr passend, da in diesem Thema viele Aspekte des gesamten Studiums vereint werden. In den weiter oben genannten Bereichen kommen Inhalte aus den Modulen AP 1&2, GdVK, WBA 1, ST, PM und EIS vor. Daher bietet das Thema eine gute Verbindung zwischen den verschiedenen Disziplinen innerhalb des Studiums, verbunden mit einer wissenschaftlichen Diskussion verschiedener Vorgehensweisen und Abläufe.

3 Durchführung der Arbeit

3.1 Zielsetzung

Ziel der Arbeit soll es sein, ein funktionierendes Produkt zu entwickeln, das von Nutzern gewinnbringend verwendet werden kann. Weiterhin soll die Arbeit Abläufe und Entscheidungen für die Umsetzung eines Projektes, wie sie auch in der Industrie Anwendung finden, im kleinen Rahmen widerspiegeln.

3.2 Erste Gliederung

Hier soll auch eine erste Gliederung der Arbeit erfolgen. Diese Gliederung besitzt keinen Anspruch auf Vollständigkeit und kann während der Durchführung der Arbeit noch verändert werden. Gerade für den zweiten Teil gestaltet es sich schwierig, konkretere Voraussagen über den Inhalt zu machen, da viele der Artefakte, auch den sich dieser bezieht, noch nicht existieren.

- Einleitung
- Relevanz
- Zielsetzung
- 1.Teil: Theoretische Grundlagen
 - Struktur der Anwendung
 - * Stand Praxisprojekt
 - * Einstieg
 - * Outcome
 - Verwendete Technologie
 - * Vue.js
 - * Angular.js
 - * React.js
 - Architektur der Anwendung
 - * redux
 - Vorgehen bei der Gestaltung
- 2.Teil: Umsetzung
 - Gestaltung
 - React Storybooks
 - Redux Store
 - Algorithm
- Fazit
- Ausblick

3.3 Zeitplan und Organisation

Die Arbeit soll so bald wie möglich, spätestens aber am 31.5.2017 angemeldet werden. Da kein erhöhter empirischer Anteil besteht, ist die reguläre Dauer von 9 Wochen zur Beendigung der Arbeit vorgesehen. Die Arbeit soll also (je nach Anmeldung) spätestens in der ersten Augustwoche 2017 abgegeben werden.

Als zweitprüferin ist Dipl. Des. Liane Kirschner vorgesehen, die aktuell bei der Railslove GmbH in Köln tätig ist.

¹*pure functions* sind funktionen, die keine Nebeneffekte aufweisen, also beim Aufruf mit den gleichen Argumenten zwangsweise die gleichen Ergebnisse liefern Haverbeke (2014). So kann eine Funktion, deren Ergebnis beispielsweise vom aktuellen Datum abhängt, niemals *pure* sein.

Abbildungsverzeichnis

Tabellenverzeichnis

Literatur

thr

Three Principles. <http://redux.js.org/docs/introduction/ThreePrinciples.html>

Gackenheimer 2015

GACKENHEIMER, Cory: What Is React? In: *Introduction to React*. Springer, 2015, S. 1–20

Haverbeke 2014

HAVERBEKE, Marijn: *Eloquent JavaScript: A Modern Introduction to Programming*. 2. No Starch Press, 2014 <http://amazon.com/o/ASIN/1593275846/>. – ISBN 9781593275846