

TBD

Bachelorarbeit II
zur Erlangung des akademischen Grades

Bachelor of Science in Engineering (BSc)

Fachhochschule Vorarlberg
Informatik – Software and Information Engineering

Betreut von
Prof. (FH) Dipl. Inform. Thomas Feilhauer

Vorgelegt von
Dominic Luidold
Dornbirn, 20. Mai 2021

Widmung

TODO

„*TODO*“
TODO

Kurzreferat

TODO

TODO

Abstract

TODO

TODO

Geschlechtergerechte Sprache

Der Verfasser der vorliegenden Arbeit bekennt sich zu einer geschlechtergerechten Sprachverwendung.

Um die Lesbarkeit zu gewährleisten und zugunsten der Textökonomie werden die verwendeten Personen beziehungsweise Personengruppen fix männlich oder weiblich zugeordnet. Zum Beispiel wird immer „die Entwicklerin“ und „der Benutzer“ verwendet. Es wurde besonders darauf geachtet, stereotype Rollenbeschreibungen zu vermeiden. Die insgesamt eventuell dadurch hervorgerufene Irritation bei den Lesenden ist gewünscht und soll dazu beitragen, eine Bewusstheit für die bestehende, Frauen diskriminierende Sprachgewohnheit (generelle Verwendung der männlichen Begriffe für beide Geschlechter) zu wecken beziehungsweise zu stärken.

Inhaltsverzeichnis

Abbildungsverzeichnis	8
Abkürzungsverzeichnis	9
1 Einleitung	10
1.1 Motivation	10
1.2 Problemstellung	11
Literaturverzeichnis	12
Eidesstattliche Erklärung	13

Abbildungsverzeichnis

1.1	Liste möglicher JavaScript-Frameworks zur Umsetzung von Single-page Applications	11
-----	--	----

Abkürzungsverzeichnis

MPA Multi-page Application

SPA Single-page Application

1 Einleitung

Diese Bachelorarbeit verfolgt das Ziel, einen Einblick in die Single-page Application (SPA) Frameworks *Angular*¹ und *Vaadin*² zu geben und deren Gemeinsamkeiten, Unterschiede sowie Vor- und Nachteile zu beleuchten.

Um ein grundlegendes Verständnis über die Thematik von Single-page Applications zu erlangen, wird zu Beginn der Arbeit auf das Konzept einer SPA eingegangen und die zugrundeliegende Herangehensweise mit der einer klassischen Multi-page Application (MPA) beziehungsweise eines Rich Clients verglichen. Im weiteren Verlauf werden die unterschiedlichen Ansätze von Angular und Vaadin genauer betrachtet und eine tatsächliche Umsetzung der zuvor erläuterten Technologien mittels zweier Demo-Applikationen getestet. Am Ende dieser Arbeit wird darauf eingegangen, ob sich - anhand unterschiedlicher Kriterien und Anwendungsfälle - eine Empfehlung für eines der beiden SPA Frameworks aussprechen lässt.

1.1 Motivation

In den letzten Jahren lässt sich beobachten, dass Webapplikationen, Apps und Anwendungen allgemein verstärkt mittels des SPA-Ansatzes umgesetzt werden und somit auf einen Thin Client - im Gegensatz zu klassischeren Rich Clients und Multi-page Applications - setzen. Für die Umsetzung einer solchen Applikation stehen eine Vielzahl von Frameworks zur Verfügung, die darüber hinaus weitere Features bieten und Entwickler:innen bei der Umsetzung unterstützen.

¹Angular (<https://angular.io>)

²Vaadin (<https://vaadin.com>)

Die richtige Wahl des Frameworks, der jeweiligen Technologien und der im Hintergrund agierenden Strukturen spielen eine wesentliche Rolle bei der Planung und Umsetzung eines neuen Projektes. Welches Framework sich besser eignet, lässt sich oftmals nicht auf den ersten (oder sogar zweiten) Blick feststellen. Diese Arbeit befasst sich daher genauer mit dem Konzept von Single-page Applications und vergleicht zwei darauf aufbauende Frameworks, die mit deutlich unterschiedlichen Technologie-Stacks arbeiten und zu vergleichbaren Lösungen führen.

1.2 Problemstellung

Die in Abschnitt 1.1 auf Seite 10 angesprochene Vielzahl an SPA-Frameworks bietet grundlegend den Vorteil, dass eine große Auswahlmöglichkeit und eine gewisse Konkurrenz untereinander zu einem hohen Qualitätsstandard führt. Zudem wird dadurch sichergestellt, dass es für jedes Projekt - unabhängig von den jeweiligen Anforderungen und etwaigen Eigenheiten - eine Möglichkeit gibt, dieses mit einem der verfügbaren Frameworks umzusetzen. Auf der anderen Seite führt die stetig wachsende Anzahl an Möglichkeiten jedoch dazu, dass sich meist nur schwer beurteilen lässt, welches Framework sich für die Umsetzung einer Applikation bestmöglich eignet.



Abbildung 1.1: Liste möglicher JavaScript-Frameworks zur Umsetzung von Single-page Applications. Quelle: A. 2020

Neben

Literatur

A., Sviatoslav (Jan. 2020). *The Best JS Frameworks for Front End*. URL: <https://rubygarage.org/blog/best-javascript-frameworks-for-front-end> (besucht am 08.02.2021).

Eidesstattliche Erklärung

Ich erkläre hiermit an Eides statt, dass ich die vorliegende Bachelorarbeit I selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus fremden Quellen direkt oder indirekt übernommenen Stellen sind als solche kenntlich gemacht. Die Arbeit wurde bisher weder in gleicher noch in ähnlicher Form einer anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Dornbirn, am 20. Mai 2021

Dominic Luidold