



[Thema der Bachelorarbeit]

[Vor- und Zuname des/der Bachelorkandidaten/in]

Konstanz, [Datum]

BACHELORARBEIT

BACHELORARBEIT

zur Erlangung des akademischen Grades

Bachelor of Science (B. Sc.)

an der

Hochschule Konstanz

Technik, Wirtschaft und Gestaltung

Fakultät Informatik

Studiengang [Software-Engineering/Technische
Informatik/Wirtschaftsinformatik]

Thema: **[Thema der Bachelorarbeit]**

Bachelorkandidat: [Vor- und Zuname des/der Bachelorkandidaten/in], [Straße],
[PLZ][Ort]

1. Prüfer: [Titel, Vor- und Zuname des 1. Prüfers]

2. Prüfer: [Titel, Vor- und Zuname des 2. Prüfers]

Ausgabedatum: [Datum]

Abgabedatum: [Datum]

Zusammenfassung (Abstract)

Thema: [Thema der Bachelorarbeit]

Bachelorkandidat: [Vor- und Zuname des/der Bachelorkandidaten/in]

Firma: [HTWG oder Firmenname]

Betreuer: [Titel, Vor- und Zuname des 1. Prüfers]

[Titel, Vor- und Zuname des 2. Prüfers]

Abgabedatum: [Datum]

Schlagworte: [Platz, für, spezifische, Schlagworte, zur, Ausarbeitung]

[Text der Zusammenfassung etwa 150 Worte. Es soll der Lösungsweg beschrieben sein.]

Ehrenwörtliche Erklärung

Hiermit erkläre ich [*Vor- und Zuname des/der Bachelorkandidaten/in*], geboren am [*Datum*] in [*Geburtsort*], dass ich

- (1) meine Bachelorarbeit mit dem Titel

[Thema der Bachelorarbeit]

bei der [HTWG oder Firmennamen] unter Anleitung von [Titel, Vor- und Zuname des 1. Prüfers] selbständig und ohne fremde Hilfe angefertigt und keine anderen als die angeführten Hilfen benutzt habe;

- (2) die Übernahme wörtlicher Zitate, von Tabellen, Zeichnungen, Bildern und Programmen aus der Literatur oder anderen Quellen (Internet) sowie die Verwendung der Gedanken anderer Autoren an den entsprechenden Stellen innerhalb der Arbeit gekennzeichnet habe.

Ich bin mir bewusst, dass eine falsche Erklärung rechtliche Folgen haben wird.

Konstanz, [Datum]

(Unterschrift)

Inhaltsverzeichnis

Ehrenwörtliche Erklärung	3
Abkürzungsverzeichnis	5
1 Einleitung	6
1.1 Quellen	7

Abkürzungsverzeichnis

HTML Hypertext Markup Language

W3C World Wide Web Consortium

Kapitel 1

Einleitung

Der Begriff “Web Components” ist ein Dachbegriff für mehrere entstehende Standards [Blog, 2014], die es für Webentwickler ermöglichen sollen, komplexe Anwendungsentwicklungen mit einer neuen Sammlung an Werkzeugen zu vereinfachen. Diese sollen die Wartbarkeit, Interoperabilität und Kapselung verbessern und somit ein Plugin-System für das Web schaffen. Durch die neuen Standards soll das Web zu einer Plattform werden, die es ermöglicht die Web-Sprache selbst, Hypertext Markup Language (HTML), zu erweitern. Dies ist bisher nicht möglich, da die HTML-Technologie, und somit die Möglichkeiten HTML-Tags zu benutzen, vom World Wide Web Consortium (W3C) definiert und standardisiert wird. Unter den wichtigsten der neuen Standard sind die folgenden vier Technologien aufzuführen: Custom Elements, Shadow DOM, HTML Templates und HTML Imports. Custom Elements ermöglichen es einem Webentwickler eigene HTML-Tags und deren Verhalten zu definieren, oder bereits vorhandene oder native HTML-Tags zu erweitern. Das Shadow DOM (Document Object Model) stellt einen Sub-DOM in einem HTML Element bereit, welcher dem Element zugehöriges Markup, CSS und JavaScript kapselt. HTML Templates stellen, wie der Name impliziert, einen Template-Mechanismus für HTML bereit [Peter Kröner 2014]. HTML Imports erlauben das Laden von HTML-Dokumenten in andere HTML-Dokumente [Peter Kröner 2 2014].

Diese neuen Technologien werden allerdings noch nicht von den populärsten Browsern, zu welchen Google Chrome, Mozilla Firefox, Opera und der Internet Explorer beziehungsweise Edge gehören, unterstützt. Des Weiteren ist das Implementieren einer Applikation, welche diese Technologien nativ benutzt, bisher sehr komplex und schwierig zu organisieren. Im Zuge dessen, entwickelt Google aktiv an einer Bibliothek namens Polymer, welche sich diesen Problemen annimmt.

Polymer stellt eine Sammlung von Polyfills (webcomponents.js) bereit, die älteren Browsern die nötigen Features für den Einsatz von Web Components beibringen. Ebenso soll das Erstellen von eigenen HTML-Elementen mit der Polymer-Bibliothek, und der damit bereitgestellten API, für Entwickler komfortabel gemacht werden. Um nun bereits entwickelte Web Components einfach wiederverwenden zu können, stellt Polymer eine Sammlung von fertigen Polymer-Elementen bereit.

Im Rahmen dieser Bachelorarbeit sollen die verschiedenen Technologien unter dem Dachbegriff Web Components, sowie deren Funktionsweise ohne, als auch mit der Polymer Bibliothek, untersucht werden. Zur Veranschaulichung soll eine Web Komponente mit Hilfe von Polymer implementiert werden und mit einer ähnlichen Implementierung mit AngularJS verglichen werden. Am Beispiel einer Web Komponente in Form einer Multi-Navigations-Applikation sollen die Vor- und Nachteile des Einsatzes von Polymer in Hinblick auf Implementierung und Performanz dargestellt werden.

1.1 Quellen

- [Peter Kröner 2014] Peter Kröner, Web Components erklärt - Teil 1: Was sind Web Components?, 2014, <http://www.peterkroener.de/web-components-erklart-teil-1-was-sind-web-components/>
- [Peter Kröner 2 2014] Peter Kröner, Das Web der Zukunft, 2014, <http://webkrauts.de/artikel/2014/web-der-zukunft>
- [SelfHTML 2014] SelfHTML, Web Components - eine Einführung, 2014, <https://blog.selfhtml.org/2014/12/09/web-components-eine-einfuehrung/>

Literaturverzeichnis

[Blog, 2014] Blog, S. (2014). Web Components – eine Einführung.
url<https://blog.selfhtml.org/2014/12/09/web-components-eine-einfuehrung/>. (Visited on 20.11.2015).

[Kröner, 2014a] Kröner, P. (2014a). Das Web der Zukunft.
url<http://webkrauts.de/artikel/2014/das-web-der-zukunft>. (Visited on 20.11.2015).

[Kröner, 2014b] Kröner, P. (2014b). Web Components erklärt, Teil 1: Was sind Web Components?
url<http://www.peterkroener.de/web-components-erklaert-teil-1-was-sind-web-components>. (Visited on 20.11.2015).