EazyMenu

DIPLOMARBEIT

verfasst im Rahmen der

Reife- und Diplomprüfung

an der

Höheren Abteilung für Informatik

Eingereicht von:
Benjamin Besic
Bozidar Spasenovic
David Ignjatovic

Betreuer:
Prof. DDI Clemens EISSERER

Projektpartner:
Oberösterreichische Versicherung AG, Linz

Leonding, April 2022

Ich erkläre an Eides statt, dass ich die vorliegende Diplomarbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt bzw. die wörtlich oder sinngemäß entnommenen Stellen als solche kenntlich gemacht habe.

Die Arbeit wurde bisher in gleicher oder ähnlicher Weise keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Die vorliegende Diplomarbeit ist mit dem elektronisch übermittelten Textdokument identisch.

Leonding, April 2022 Benjamin Besic & Bozidar Spasenovic & David Ignjatovic

Zur Verbesserung der Lesbarkeit wurde in diesem Dokument auf eine geschlechtsneutrale Ausdrucksweise verzichtet. Alle verwendeten Formulierungen richten sich jedoch an alle Geschlechter.

Abstract

Brief summary of our amazing work. In English. This is the only time we have to include a picture within the text. The picture should somehow represent your thesis. This is untypical for scientific work but required by the powers that are. Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Zusammenfassung

Zusammenfassung unserer genialen Arbeit. Auf Deutsch. Das ist das einzige Mal, dass eine Grafik in den Textfluss eingebunden wird. Die gewählte Grafik soll irgendwie eure Arbeit repräsentieren. Das ist ungewöhnlich für eine wissenschaftliche Arbeit aber eine Anforderung der Obrigkeit. *Bitte auf keinen Fall mit der Zusammenfassung verwechseln, die den Abschluss der Arbeit bildet!* Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Inhaltsverzeichnis

1 Einleitung

1.1 Ausgangsituation

Die Oberösterreichische Versicherung ist der Marktführer im Versicherungsbereich in Oberösterreich und beschäftigt in ihrer Zentrale in Linz über 500 Mitarbeiter. Das Unternehmen besitzt eine Kantine, wo täglich 3 Hauptspeisen serviert werden. Dazu noch eine Vorspeise und eine Nachspeise.

1.2 Ist-Zustand

Die derzeitige Bestellmöglichkeit funktioniert über eine simple Datenbankanwendung mit IBM Notes. Dieses Programm ist auf jedem Rechner installiert und jeder Mitarbeiter ist mit seinen Daten bereits eingeloggt. Auf einem simplen Interface kann man zwischen den heutigen und zukünftigen Mahlzeiten wählen und die Uhrzeit, wann man diese konsumiert. Nach erfolgreicher Bestellung hat man eine kleine Übersicht über die vergangenen Bestellungen.

1.3 Problemstellung

Das derzeitige Programm ist sehr veraltet und nicht besonders benutzerfreundlich. Außerdem läuft das Programm lokal auf jedem Rechner und eine mobile Bestellung ist ausgeschlossen. Außerdem ist zu erwähnen ist, dass der Benutzer nur eine beschränkte Möglichkeit hat sein Bestellverhalten zu visualisieren bzw. zu analysieren.

1.4 Aufgabenstellung

Die Aufgabenstellung der Firma ist eine Webanwendung zu entwickeln, die den Vorgänger ablöst und ein modernes und benutzerfreundliches Interface hat. Zudem soll eine Bestellung über das Smartphone möglich sein. Zusätzlich soll ein Empfehlungssytem entwickelt werden, dass einem ermöglicht Mahlzeiten zu bestellen, die auf einen

abgestimmt sind. Außerdem soll man eine Einsicht über seine Bestellhistorie haben mit diversen Statistikelementen.

1.5 Ziel(e)

- Erleichterung des Bestellprozesses für den Benutzer
- Flexible Bestellungsmöglichkeiten
- Der Benutzer hat mehr Verständnis über sein Bestellverhalten

1.5.1 Zielgruppe

Mitarbeiter der OÖ Versicherung AG

2 Planung

Der Anfang der Planung war die Besprechung des alten Programms und was an dem nicht passt bzw. verbessert gehört.

2.1 Use Cases

2.1.1 Kantinenarbeiter

- Neue Menüs anlegen
  - Ein Kantinenmitarbeiter kann für jeden Tag neue Menüs mit drei Hauptspeisen und deren Kategorien, einer Vorspeise und einer Nachspeise anlegen.
- Vorhandene Menüs editieren
  - Die Bezeichnungen der bereits erstellten Menüs sollen verändert werden können.
- Übersicht der täglichen Bestellungen
  - Die Kantinenmitarbeiter sollen eine Übersicht, der an einem bestimmten Tag bestellten Menüs haben. Diese inkludiert die zusammengefasste Bestellanzahl der verschiedenen Menüs und eine Liste von allen Bestellungen.
- Bestellungsübersicht drucken
  - Die Übersicht wie vorher beschrieben soll zu einem PDF-Objekt konvertiert werden und dementsprechend ausgedruckt werden können.

2.1.2 Mitarbeiter

- Menüs bestellen
  - Ein Mitarbeiter hat eine Auswahl aller Menüs und kann für jeden Tag eine der drei Hauptspeisen auswählen. Nach der Auswahl kann er die Essenszeit auswählen, die Anzahl und nötige Kommentare hinzufügen.
- Menüs für andere Mitarbeiter bestellen
  - Ein Mitarbeiter kann den obrigen Bestellvorgang für einen anderen Mitarbeiter ausführen.
- Übersicht aller Bestellungen

  - Als Mitarbeiter soll man alle seine vergangenen Bestellungen und deren Informationen in einer Übersicht einsehen können. Diese Übersicht kann filtriert werden.
- Bestellungen stornieren
  - In der oben genannten Übersicht soll man die Möglichkeit haben eine Bestellung auszuwählen und zu stornieren, wenn dies möglich ist.
- Bestellstatistiken einsehen
  - Ein Mitarbeiter soll Diagramme zur Verfügung haben, wo er sein Bestellverhalten einsehen kann.

2.2 Datenmodell-Diagramme

Der nächste Arbeitsschritt war die Entwicklung eines Datenmodells, welches die Basis der Programmlogik sein soll. Dieses wurde mittels ERD-Diagramm erstellt.

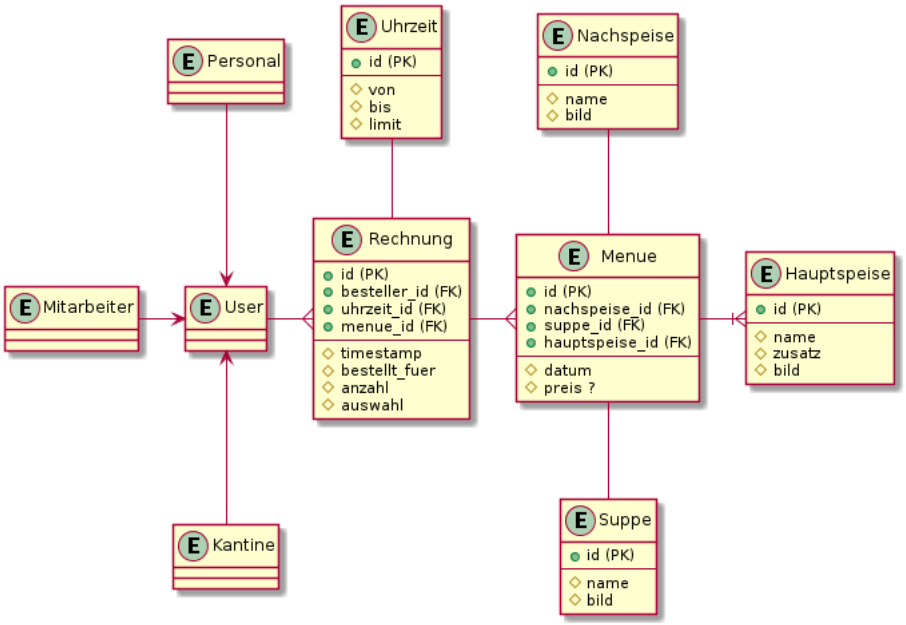

Abbildung 1: Erste Version des Datenmodells

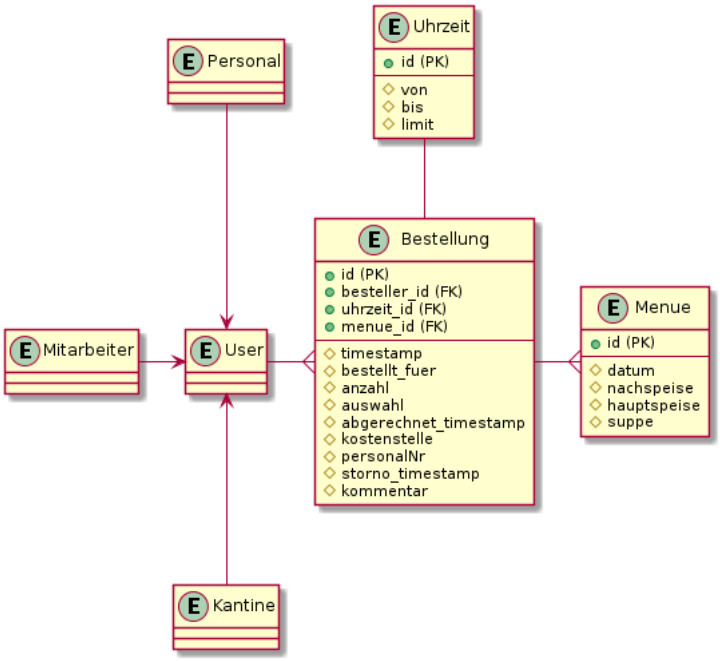

Abbildung 2: Finale Version des Datenmodells

2.3 UI entwickeln

Nachdem das Datenmodell feststand wurden UI-Prototypen entwickelt, die das Aussehen der Vue-App darstellen sollen.

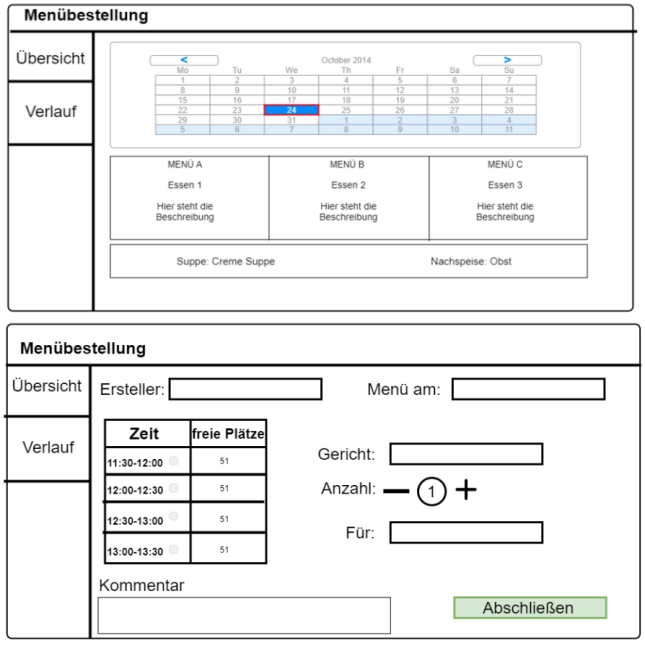

Abbildung 3: UI-Prototypen für den Bestellvorgang

Abbildung 4: UI-Prototypen für den den Login und die Übersicht

2.4 Technologien

Beim Entwickeln wurden folgende Technologien verwendet:

- docker 3.1
- Vue.js 2.6.14
- quarkus 2.5.0.Final
- Jetpack Compose 1.0.1
- Keycloak 14.0.0
- Java OpenJDK-11
- Java EE 8
- JBoss Wildfly 7.3.4.GA

3 State of Art

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula. Citing [1] properly.

Was ist eine Globally Unique Identifier (GUID)? Eine GUID kollidiert nicht gerne.

Kabellose Technologien sind in abgelegenen Gebieten wichtig [2].

4 Verwendete Technologien

4.1 IntelliJ IDEA

IntelliJ IDEA ist eine der führenden Entwicklungsumgebungen für die Programmiersprache Java. Sie wurde vom Unternehmen Jetbrains im Jahre 2000 entwickelt.
Außerdem bietet sie ebenfalls Entwicklungsmöglichkeiten für Kotlin, Groovy, Scala und auch Android. Sie ist immer auf dem neusten Entwicklungsstand, wird laufend mit Updates versorgt und unterstützt die derzeit gängigen Programmiertools wie Docker, Kubernetes, Maven, Datenbank-Tools, Git, Jakarta EE und viele weitere. Es gibt eine kostenpflichtige Ultimate Version und eine Community Version, die kostenfrei zur Verfügung gestellt wird.
IntelliJ zeichnet auch die Anzahl an Erweiterungen mittels Plugins aus. Die Umgebung besitzt auch eine sehr intuitive Intelligenz, die es dem Entwickler sehr einfach macht damit zu programmieren. [3]
Wir haben uns dafür entschieden, da wir damit viel Erfahrung hatten und die oben genannten Punkte unterstützten unsere Entscheidung enorm.

4.2 Android Studio

4.3 Git

Git ist ein Versionskontrollsystem (oft abgekürzt durch VCS) für Entwickler. Es ist ein Open-Source System, das im Jahre 2005 von Linus Torvald entwickelt wurde. Laut einer Stack Overflow-Umfrage von Entwicklern nutzen über 87 % der Entwickler Git. [4]
Zu aller erst muss man den Begriff Versionskontrolle erklären, um Git zu verstehen.

4.3.1 Versionskontrolle

Diese dient dazu, um den originalen Quellcode effizient mit mehreren Personen editieren bzw. entwickeln zu können.
Die Entwickler arbeiten mit Verzweigungen und Zusammenführungen. Jeder Entwickler kann Änderungen sicher durchführen, ohne seine Kollegen dabei zu behindern. Diese Änderungen können dann, sobald sie funktionsfähig sind, wieder in den Hauptquellcode eingebunden werden. Alle Änderungen sind zurückzuverfolgen und bei Bedarf kann man sie dann wieder zurücksetzen. [4]

4.3.2 Git Funktionsweise

Jeder Entwickler hat seine eigene Version des Projekts (Working Directory), die er frei bearbeiten kann. Diese bekommt man durch einen Klon des Projekts (Clone).
Diese Änderungen kann man aufteilen und in Paketen bereitstellen, nach dem man diese durch Commits trennt. Einen Commit kann man benennen.
Diese Commits kann man dann online veröffentlichen durch einen Push. Ein Push ist nur möglich, wenn man die aktuellste Version des Projekts auf seinen Rechner gezogen hat (Pull). Einem Push kann man einen bestimmten Zweig (Branch) zuordnen. Diese

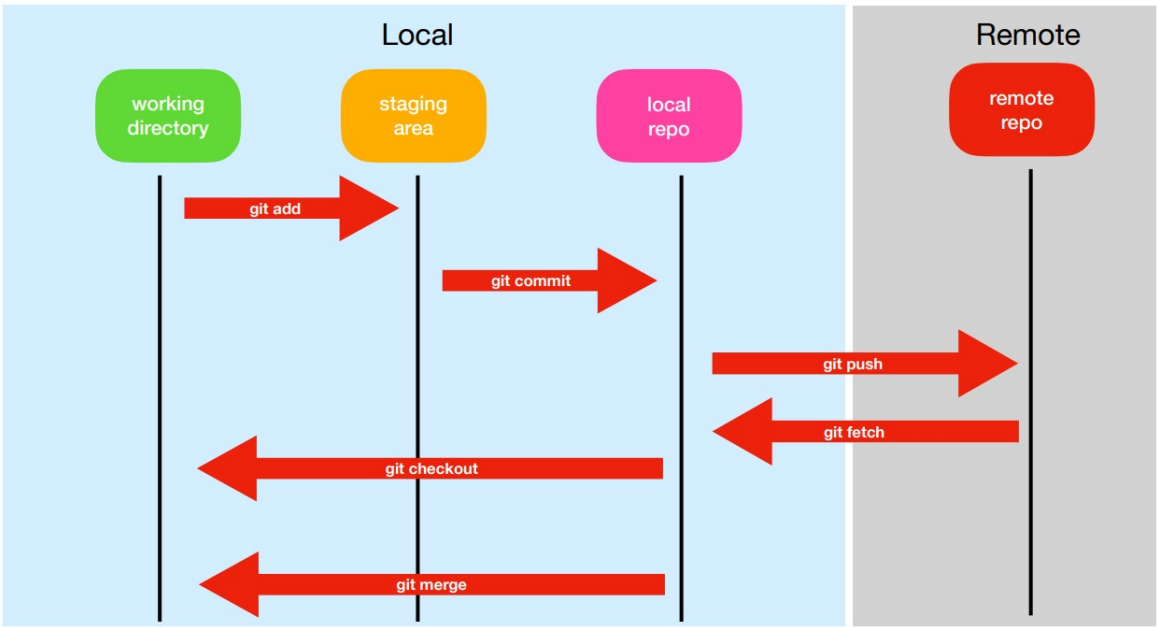

Abbildung 5: Darstellung des Git Workflows

Zweige dienen dazu, um das Projekt in verschiedene unabhängige Teile zu trennen, falls

man zum Beispiel an einer Demo Version weiterschreiben möchte.
Nach einem erfolgreichen Push sind die Änderungen online auf dem Repository zu finden. Ein Repository ist der "Ordner", wo alle Dateien online gespeichert zu finden sind.
Ein Vorteil, den Git ebenfalls bietet ist das jeder Commit eine Version des Projekts ist, die man bis zum Zeitpunkt vom Commit herunterladen oder klonen kann. Jedes Projekt kann privat oder öffentlich gemacht werden, sodass auch Personen, die keine Entwickler sind, darauf zugreifen können. [5]

4.3.3 Git Befehle

Clone

Es initialisiert ein Git Repository auf dem Rechner und ladet die zugehörigen Dateien runter. Wenn man es nicht spezifisch angibt, klont es den Master Branch. Der Master Branch ist der Hauptzweig, eines jeden Git Projekts. Innerhalb des erstellten Ordners können alle weiteren Git Befehle ausgeführt werden. [6]

Commit

Ein Commit beschreibt Änderungen, die man im Projekt gemacht hat. Jeder Commit hat eine Bezeichnung, mit der der Entwickler die Änderungen, die er gemacht hat beschreiben kann. Zu jedem Commit gehören auch die Dateien die dabei geändert bzw. hinzugefügt wurden.
Es speichert den Zustand des gesamten Projekts bis zu dem Zeitpunkt und kann danach jederzeit abgerufen oder rückgängig gemacht werden. Diese Änderungen bleiben aber zunächst nur lokal auf dem Rechner. [6]

Push

Ein Push dient dazu, um die lokalen Änderungen (Commits) zu veröffentlichen. Es kopiert den aktuellen, lokalen Stand und speichert diesen auf das vom Internet erreichbare Repository.
Einem Push kann ein Zweig (Branch) zugeordnet werden um die Änderungen zuzuordnen. Ansonsten wird der Master Branch genommen. [6]

Pull

Der Pull Befehl kopiert die Inhalte vom öffentlichen Repository und fasst diese mit den lokalen Zustand auf dem rechner zusammen (merge). Es dient dazu die aktuelle Version des Projekts auf den Rechner herunterzuladen.
Falls es Konflikte zwischen der derzeitigen und neusten Version gibt, werden die Änderungen zusammengeführt. [6]

Branch

Die Branch Befehle dienen dazu um eine neue Abzweigung des Projekts (Branch) zu erstellen.
Branches erstellt man, wenn man an einer neuen Version des Projekts arbeitet und diese vom Hauptteil trennen will. Meistens werden dadurch neue Funktionen programmiert, die später wieder in den Master Branch eingebunden werden.
Der Vorteil daran ist, dass man an neuen Funktionalitäten experimentieren kann, ohne den Hauptentwicklungsstand zu beeinflussen. Branches können jederzeit gelöscht oder wieder ins Hauptprojekt integriert werden. Es kann simultan am Master Branch und an Nebenzweigen gearbeitet werden durch trennen mit dem Push Befehl. [5]

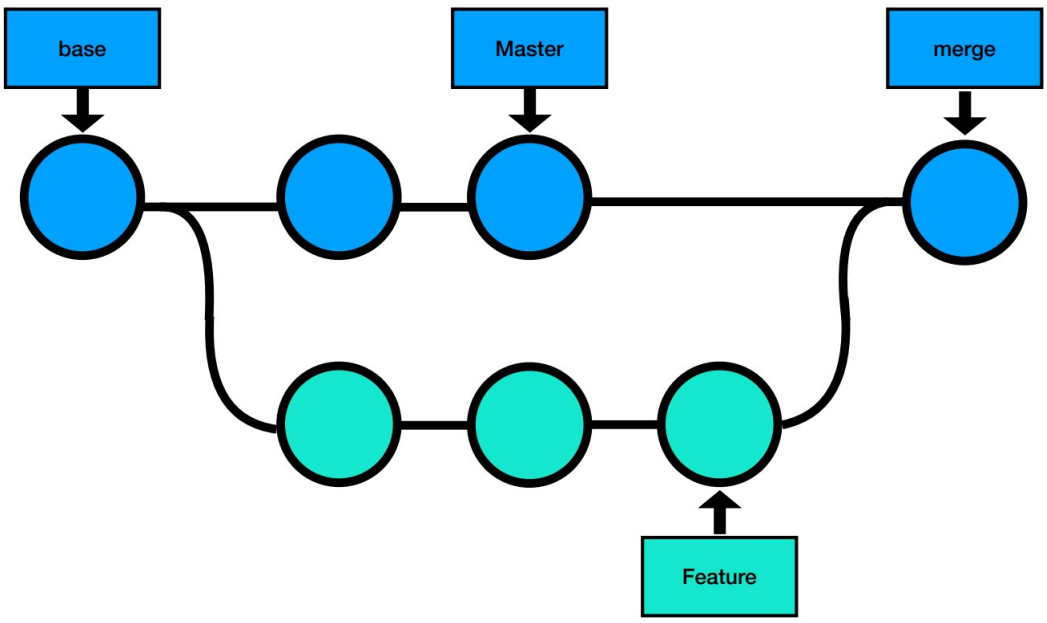

Abbildung 6: Darstellung von Branches in einem Repository

4.3.4 GitHub

GitHub ist ein gewinnorientiertes Unternehmen, dass einen auf Cloud basierten Git Repository Hosting-Service anbietet. Es wurde im Februar 2008 gestartet und von Chris Wanstrath, PJ Hyett, Scott Chacon und Tom Preston-Werner entwickelt. Es ist das beliebteste Tool um Softwareprojekte zu verwalten und wird von über 73 Millionen Entwicklern und über 4 Millionen Organisationen benutzt. Dazu ist es das größte und am meisten fortgeschrittene Entwicklungssystem, das es gibt. [7]
Es vereinfacht die Nutzung von Git für Teams und auch Einzelpersonen. Jeder kann sich einen GitHub Account erstellen und direkt loslegen und seine Arbeiten auf Repositories veröffentlichen. Es ist nicht nur zwingend für Code-basierte Projekte verwendbar sondern auch Websiten erstellen und das Schreiben von Büchern ist möglich.
Was GitHub ausmacht ist die Benutzerfreundlichkeit und die Integration von Git. Außerdem bietet Github viele andere Funktionen wie zum Beispiel ein Projekt Board an, was es erleichtert innerhalb eines Teams, Probleme besser lösen zu können.
Es gibt ebenfalls bezahlte Pläne, die es vor allem Organisationen und Unternehmen leichter macht Unternehmensprojekte zu verwalten durch zusätzliche Funktionen. [4]

4.4 Java

4.5 Java EE

4.5.1 Java EE vs. Quarkus

4.5.2 JPA

4.6 Quarkus

4.6.1 Hibernate

4.6.2 Panache

4.7 Maven

4.8 JBoss

4.9 Cypress

4.10 Keycloak

Keycloak ist eine Open-Source Software, die Single-Sign On mit Identity und Access Management für moderne Applikationen bereitstellt. Der erste Release war 2014 als Wildfly Community Projekt, seit 2018 aber steht es unter der Verwaltung von RedHat. [8]
Es hat mehrere Distributionen und ist mit einer Vielzahl von Frameworks und Tools kompatibel bzw. integriert wie z.B.: Quarkus, Angular, Vue.js, Spring, usw. [9]

4.10.1 Distributionen

Server

Die Standalone Applikation ist downloadbar als .zip auf der Keycloak Seite. Es gibt zwei Versionen vom Server, zu einem die Wildy Application Server Version und auch eine Version, die über Quarkus läuft.

Docker Image

Genau so wie beim Server gibt es auch hier zwei verschiedene, offizielle Docker-Images, eins, dass auf dem Wildfly Server basiert und eins, dass auf Quarkus basiert.

Operator

Dies ist eine Distribution für Kubernetes und OpenShift, basierend auf der Operator SDK. [9]

4.10.2 IAM (Identity Access Management)

Eine der Hauptfunktionen von Keycloak ist das IAM.
IAM oder auch IdM (Identity Management) ist ein Framework, das zum Authentifizieren von Benutzern und deren Rechten genutzt wird.
Es prüft ob der Benutzer Zugang zu bestimmten Bereichen, Dateien und anderen Ressourcen hat, auf die er zugreifen will. Es prüft auch wer diese Rechte verändern darf. IAM-Systeme stellen auch nützliche Admin-Tools zur Verfügung, um z.B. Nutzerrechte zu ändern oder die Aktivität von Benutzern zu verfolgen. [10]
IAM hat vier Hauptfunktionen:

- **Die Identitäts-Funktion:** Es umfasst die Erstellung von Identitäten (Benutzern), das Managen und Löschen von diesen.
- **Die User Zugriff-Funktion (log-on):** Dies umfasst ein Interface, indem der Benutzer seine Zugriffsdaten eingeben (übermitteln) kann.
- **Die Service-Funktion:** Für Benutzer und deren Geräte stellt das System personalisierte, rollenabhängige, multimedia, online und on-demand Services zur Verfügung. [10]

4.10.3 Single Sign-On (SSO)

SSO ist eine Variante der Zugangskontrolle, deren sich Keycloak bedient.
Es ist für mehrere und zusammenhängende Softwaresysteme gedacht, wo der Benutzer nur einmal sein Passwort und Benutzernamen eingeben muss, um auf alle Systeme zugreifen zu können, ohne sich dazwischen neu identifizieren zu müssen.
SSO ist typischerweise zu ermöglichen mit LDAP (Lightweight Directory Access Protocol). Bei IP-Netzwerken funktioniert das Ganze über Cookies, die gespeichert bleiben

und die Seiten müssen eine gleiche, übergeordnete DNS Domain haben.
Authentifizierungsschemas, die dies unterstützen sind: OAuth, OpenID, OpenID Connect und Facebook Connect. Diese ermöglichen das Einloggen über mehrere verschiedene Websites mit den selben Anmeldedaten.[10]

Vorteile von SSO

- Reduziert das Risiko beim Verwenden von Drittanbieter-Websites
- Reduziert die Passwortschwäche von den verschiedenen User und Passwort Kombinationen.
- Reduziert die Zeit, die man für das Eingeben der verschiedenen Identitäten braucht
- Reduziert IT-Helpdesk Anrufe wegen Passwörtern, daraus werden auch die IT-Kosten reduziert [10]

4.10.4 Features

Multiple Protocols Support

Gerade unterstützt Keycloak drei verschiedene Authentifizierungsprotokolle: OpenID Connect, OAuth 2.0 und SAML 2.0 [9]

SSO (Single Sign-On)

Siehe hier

Admin Konsole

Keycloak stellt eine web-basiertes Interface zur Verfügung, wo der Admin seine Konfigurationen intuitiv vornehmen kann. [9]

User Identity und Accesses

Siehe hier

External Identity Source Sync

Wenn man ein eigene User-Datenbank hat, kann man diese mit Keycloak verbinden und synchronisieren. Standardmäßig unterstützt es LDAP und Active Directory, aber

diese kann man durch selbstgeschriebene Extensions erweitern. Diese kann man mit der Keycloak User-Storage API machen. Es garantiert aber nicht, dass Keycloak alle Funktionen aufweist, die auch die Datenbank hat. [9]

Identity Brokering

Keycloak kann auch als Proxy zwischen den Usern und einem oder mehreren externen Identity Provider fungieren. Diese können im Admin Panel editiert werden. [9]

Social Identity Providers

Zusätzlich erlaubt Keycloak die Benutzung von Social Identity Providern. Es hat eine eingebaute Unterstützung für Google, Twitter, Facebook, Stack Overflow, diese müssen aber im Admin Panel manuell konfiguriert werden. Die volle Liste der kompatiblen Social Identity Provider kann in der Keycloak Dokumentation gefunden werden. [9]

Anpassung von Seiten

Keycloak erlaubt eine Anpassung aller Seiten, die dem User angezeigt werden (z.B. Login-Page). Diese sind im .ftl Format, somit kann man klassisches HTML und CSS zum editieren verwenden. Auch Javascript steht zur Verfügung, damit ist der Anpassungsspielraum unendlich. [9]

4.10.5 Funktionsweise

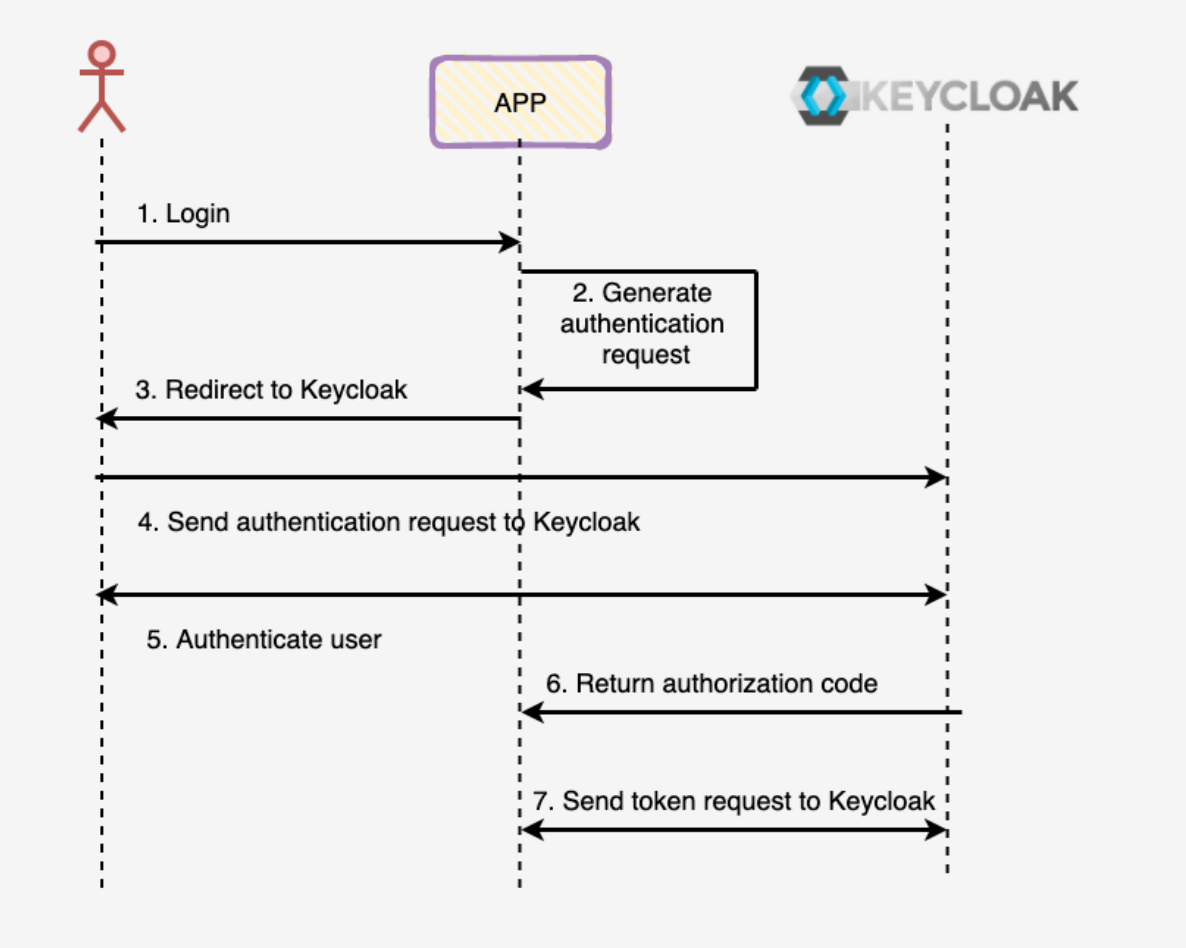

Abbildung 7: Keycloak Funktionsweise
https://miro.medium.com/max/1400/1*1McvnvrW6wh37ECYpmTSxw.png

Keycloak speichert sich einen Public Link der Applikation. Wenn dieser Link von einem User geöffnet wird, leitet Keycloak diesen zu einer Keycloak Authentication Page weiter. Nachdem erfolgreichen Einloggen leitet Keycloak den User dann zu dem eigentlich gewünschtem Link weiter und gibt einen Token mit Zeitstempel mit. Die Applikation kann dann den Token verwenden um den User und seine Rechte zu identifizieren. [10]
Bei der Funktionsweise sind wichtige Begriffe von Nöten:

Realm

Ein Realm kann man sich als Mieter vorstellen, der einen Bereich zur Verfügung stellt. Dieser sogenannte Realm ist vollkommen isoliert (im Sinne von Usern, Konfigurationen, Rollen, etc.) von anderen Realms.
Aus diesem Grund kann man einen internen Realm für z.B. Mitarbeiter machen und einen externen für z.B. Kunden. Somit sind die beiden von einander getrennt. [11]

Client

Clients sind die Anwendungen, die eine Authentifizierung von Keycloak fordern, also z.B. eine Webapp. Diese können aber auch mobile oder native Applikationen sein. Sie können jeder Servicetyp sein wie REST API's, gRPC oder WebSockets, die nur eine simple Authentifizierung und Rollenvergabe mittels Access Tokens benötigen. [11]

Rollen

Eine Rolle repräsentiert eine Rolle in der Organisation bzw. in der Applikation. Ein User kann z.B. eine Admin-Rolle bekommen und somit alles an der Applikation konfigurieren. Wenn man dies nicht will kann man Rollen und deren Spielraum eingrenzen, sodass die Applikation nicht willkürlich verändert wird.
Keycloak unterstützt ebenfalls zusammengesetzte Rollen. Diese Funktion sollte man aber vorsichtig behandeln, da diese die Komplexität der Applikation erhöht und diese somit schwerer zu pflegen ist. [11]

User

Der User ist der eigentliche Benutzer der Applikation. Dieser kann dementsprechend zu einem Realm gehören und seine eigenen Rechte haben. Dieser identifiziert sich durch den Client und seine Rollen im Realm werden der Applikation mitgegeben.

4.11 Oracle Datenbank

4.12 Vue.js

Vue.js ist ein JavaScript-Webframework, das zum Erstellen von Single-Page-Webanwendungen dient. Es wurde von einem kleinen Team im Jahre 2014 entwickelt mit dem ursprünglichem Autor Evan You.
Vue ist relativ neu und die große Stärke von Vue ist die einfache Lernkurve, die Vielseitigkeit und die Leichtgewichtigkeit. Man benötigt Kenntnisse in JavaScript, HTML, CSS und schon kann man loslegen mit deren ausführlich dokumentierten Guide[12]. [13] [14]

4.12.1 Vue Funktionsweise

Model-View-Viewmodel (MVVM) Pattern

Vue.js benutzt das MVVM Pattern. Das Pattern trennt die Darstellung von der Logik der Benutzer-UI's. Dazu ist ein Datenbindungsmechanismus vorausgesetzt. Dadurch können sich Entwickler und Interfacedesigner trennen und ihre Aufgaben im Projekt aufteilen.
Dieses Pattern wurde 2005 von John Gossman veröffentlicht. [15]

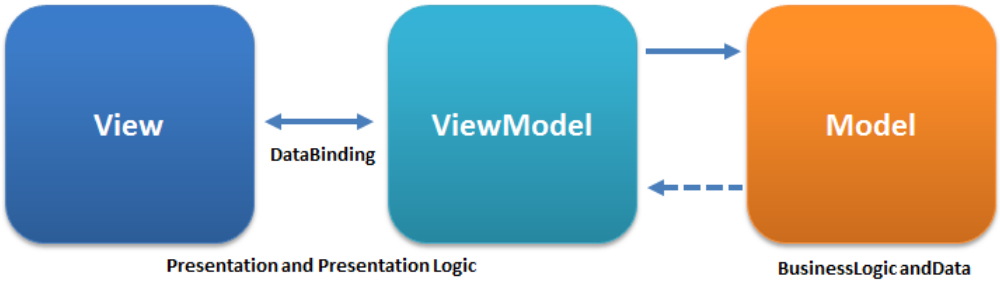

Abbildung 8: Darstellung von Branches in einem Repository
https://upload.wikimedia.org/wikipedia/commons/8/87/MVVMPattern.png

- **View:** Enthält alle Elemente die durch die Benutzeroberfläche angezeigt werden. Es bindet sich an das ViewModel, welches die Eigenschaften der View bestimmt.
- **ViewModel:** Es enthält die Logik des UI's. Es tauscht sich mit dem Model aus und benützt seine Methoden und Dienste. Gleichzeitig gibt es der View Eigenschaften, die dem Model entsprechen. Es bindet Daten mit der View und sich selbst (DataBinding).
- **Model:** Diese Schicht enthält alle Daten die der Benutzer manipuliert oder aufruft. Es enthält die gesamte Geschäftslogik.[15]

Vue Instance

Jede Vue Applikation beginnt mit der Erstellung einer Vue Instanz.

Listing 1: Vue Instanz

```
1       var vm = new Vue({
2           // options
3       })
```

Die Variable vm steht für ViewModel, was unsere Vue Instanz darstellt. Man kann jeder Instanz Optionen zuweisen, um sie zu konfigurieren. Diese Instanz wird auch als Root Instanz bezeichnet und bildet den Stamm eines Baumes mit Komponenten.

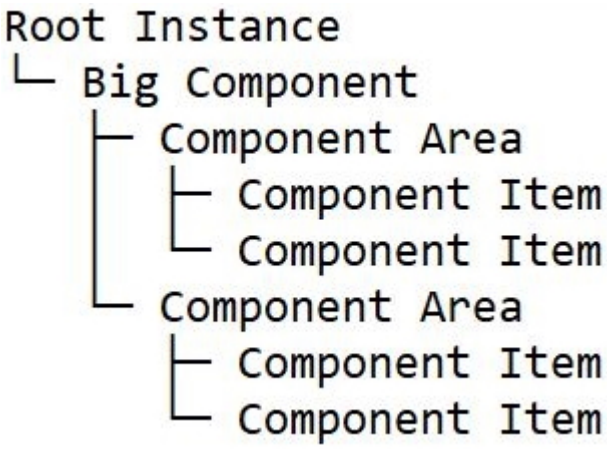

Abbildung 9: Der Stammbaum einer Root Instanz

Zu einer Instanz gehört auch der data-Bereich. Dieser beherbergt alle Properties einer Instanz und diese Properties reagieren auf Veränderungen im Code. Noch dazu kann jede Instanz Methoden haben.
Jede Instanz hat auch seine Lifecycle Hooks, dies sind Methoden, die zu bestimmten Zeitpunkten einer Instanz ausgeführt werden. [16] Diese sind:

- **created**
- **mounted**
- **updated**
- **destroyed**

Lifecycle Diagram

Das Diagramm hier stellt den Ablauf einer Erstellung einer neuen Vue Instanz dar.

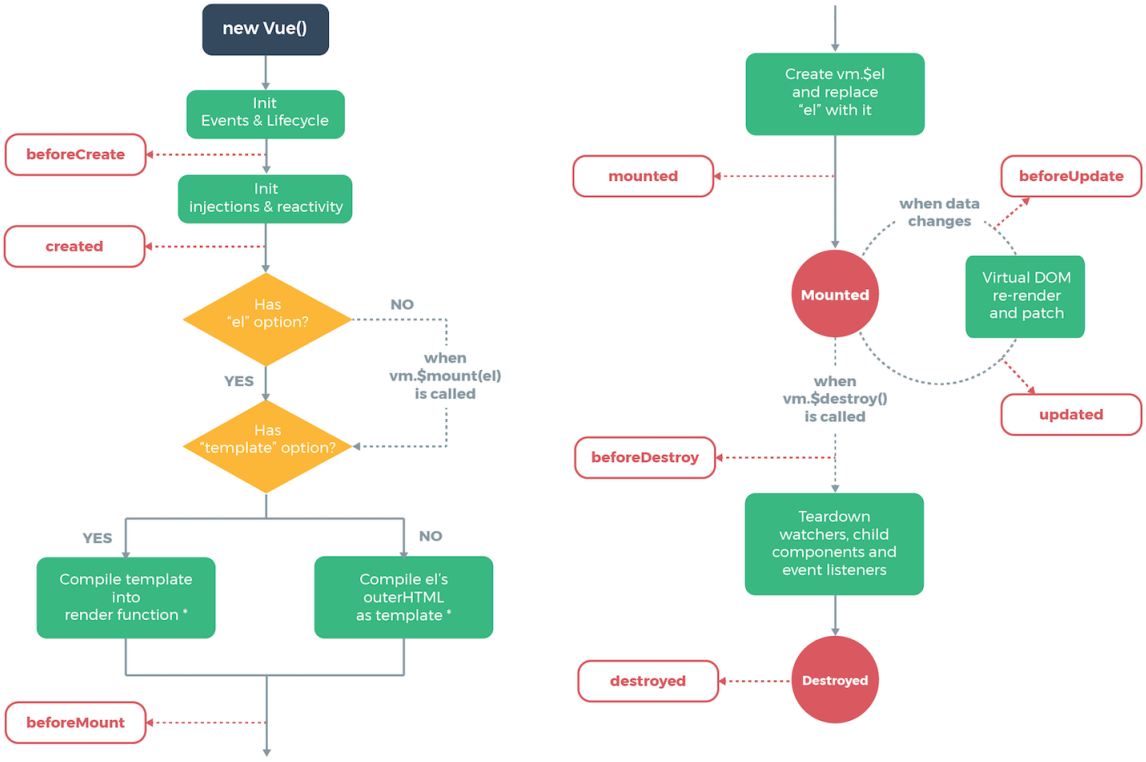

Abbildung 10: Diagramm des Ablaufs einer Vue Instanz
https://www.oreilly.com/library/view/full-stack-vuejs-2/9781788299589/assets/9f308e86-bbbe-489c-9f93-06abe2675081.png

Vue Components

Komponenten sind wiederverwendbare Vue Instanzen mit einem eigenen Namen. Diese Komponenten können mittels HTML-Tags in anderen Komponenten verwendet werden. Eine Vue.js Seite ist meistens in mehrere Komponenten aufgeteilt, um größere Bereiche auf der Seite zu trennen und übersichtlicher zu gestalten. Diese können miteinander kommunizieren und Daten austauschen, um z.B. Daten, die ständig verändert werden ordentlich darzustellen.[17]

Die folgende Grafik veranschaulicht den Component Tree, der zeigt wie die Single-Page Anwendung umgesetzt ist. Die grünen Boxen zeigen die miteinander vernetzten Komponenten, jeder Komponent kann mehrere Unterkomponenten haben, je nach Einteilung der Webseite.

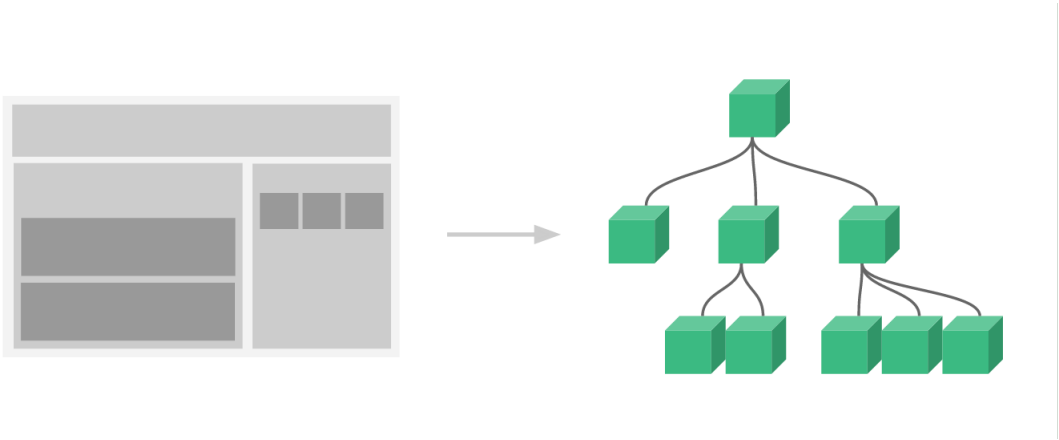

Abbildung 11: Darstellung von vernetzten Komponenten innerhalb einer Webpage
https://vuejs.org/images/components.png?_sw-precache=b5c08269dfc26ae6d7db3801e9efd296

4.12.2 Angular vs. Vue

Marktstatistik

Angular:

- Angular wird eher für Seiten benutzt mit hohen Aufrufzahlen
- Nachdem was bekannt ist benutzen unter 0.4% aller Webseiten Angular
- Angular wird von 16.1% Entwicklern weltweit verwendet [18]

Vue:

- Es gibt mehr als 1.523.449 Millionen Webseiten, die Vue benutzen
- Der Marktanteil von Vue beträgt nicht mehr als 0.5% [18]

Vor- und Nachteile

Von der Lernkurve her ist Vue deutlich im Vorteil, da es einfacher zu verstehen ist als Angular. Angular benötigt viel Einarbeitungszeit, bis man die Funktionsweise verstanden hat. Angular ist eher für umfangreiche Projekte gedacht, währenddessen Vue auf geringe Größe und hohe Performance abzielt.
In Angular sind die Logik und das Aussehen strikt getrennt, währenddessen in Vue alles in einem File zu finden ist und mehr an HTML erinnert durch die Scripts, die man schreibt. Ein Vorteil von Angular ist die Implementierung von Typescript, was eine Weiterentwicklung von JavaScript ist, die versucht die Macken von JavaScript zu verbessern.
Was noch zu erwägen ist ist, dass Angular weiter verbreitet ist als Vue und es oft

Features oder Plugins gibt, die in Angular selbstverständlich sind, aber in Vue nicht aufzufinden sind. Dies sollte sich aber im Laufe der Zeit verbessern.
Generell kann man sagen, dass das Programmieren mit Angular eher and die Programmierung mit Java erinnert mit den Objekten, Abhängigkeiten, Konstruktoren, usw. Während Vue an das Programmieren von Websites mittels HTML und JavaScript erinnert. [19]

Warum Vue?

Unsere Entscheidung Vue zu nehmen ist einerseits von der Firma beeinflusst worden, da sie es vorgeschlagen haben und es bei ihnen das gängige Framework ist.
Andere Faktoren waren noch, dass wir Angular in der Schule oft verwendet haben und wollten durch Vue eine andere Methode ausprobieren, um Webanwendungen zu erstellen. Vue ist die bessere Wahl für den Umfang unserer Webanwendung gewesen und die bessere Performance in Vue ist wichtig, damit die Bestellungen reibungslos ablaufen können im Echtbetrieb.

4.13 HTML

HTML genannt HyperText Markup Language, ist eine einheitliche, textbasierte, Auszeichnungssprache für Webdokumente. HTML definiert ganz allgemein gesehen die Struktur eines Dokuments. Am 13.März 1989 wurde am CERN in Genf das Konzept HTML von Tim Berners-Lee vorgeschlagen, um eine einheitliche Methode zu finden Dokumente öffentlich zu übermitteln. Seitdem ist HTML einer Grundbausteine des World Wide Webs geworden.
HTML basiert auf sogenannten Tags, die verschiedene Inhalte auf der Website definieren sollten. Diese basieren auf einer bestimmten Syntax, um das Schreiben zu vereinheitlichen.
Es ist im Grunde eigentlich keine Programmiersprache, sondern eine statische Sprache, die zur Definierung benutzt wird. Das HTML Gerüst wird von einem Browser eingelesen und dieser generiert dann auf der Basis des HTML Files eine Website. [20] [21]

Listing 2: HTML File Grundgerüst

```
1 <!DOCTYPE html>
2 <html>
3     <head>
4         <title> Titel der Webseite </title>
5     </head>
6     <body>
```

```
7           <h1> Ueberschrift </h1>
8       </body>
9   </html>
```

Wie man im Beispiel sieht gibt bei Tags ein Beginn und ein Ende und der Inhalt dazwischen wird dargestellt. Es gibt auch Tags ohne Beginn und Ende, sogenannte inhaltslose Tags. Doch HTML wird meist nie allein verwendet, erst in der Kombination mit CSS, Bootstrap und JavaScript kann man eine gute Website erstellen.

4.14 CSS

CSS (Cascading Style Sheets) ist die Sprache, die benutzt wird eine HTML Seite visuell zu gestalten. CSS ist wie HTML keine Programmiersprache, sie wurde dafür entwickelt um das Aussehen von HTML Seiten einheitlich zu verändern.
Eine CSS-Datei wird durch einen Tag mit dem zugehörigen HTML Dokument verbunden und dadurch werden die Änderungen angewendet. [22]

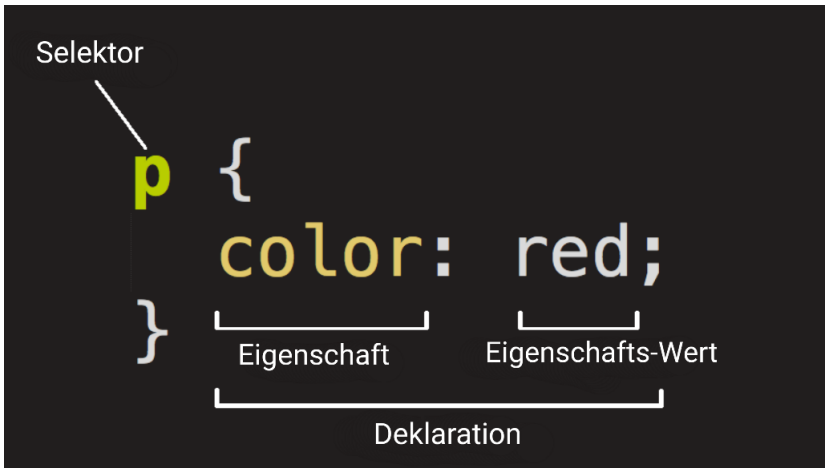

Abbildung 12: Aufbau einer CSS-Regel
https://media.prod.mdn.mozit.cloud/attachments/2017/09/27/15467/3889d04d90c10b27e863c6850d588c43/css-example.png

Die Struktur von CSS-Dateien wird durch Regeln beschrieben. Jede Regel hat einen Selektor, der auf ein zugehöriges HTML Tag zugreift. Innerhalb der Deklaration wird dann der Wert einer bestimmten Eigenschaft gesetzt (Mehrere Eigenschaften sind möglich). Es gibt eine große Auswahl von Eigenschaften wie z.B.: Schriftfarbe, Textart, Positionierung, etc. [22]

4.15 JavaScript

JavaScript ist eine leichtgewichtige Skriptsprache, die 1995 vom Softwareunternehmen Netscape entwickelt geworden ist, um die Möglichkeiten von HTML und CSS zu erweitern. Bekannt ist sie hauptsächlich als Sprache für Webseiten geworden, jedoch wird sie auch in vielen Umgebungen außerhalb des Browsers oft benutzt, wie z.B. Servern.
Sie unterstützt objektorientierte, imperative als auch deklarative Programmierung. JavaScript folgt dem Standard ECMAScript, welche alle Browser unterstützen. JavaScript dient dazu auf Webseiten Benutzerinteraktionen auszuwerten, Inhalte zu verändern, nachzuladen oder zu generieren. Sie bietet normalen HTML Webseiten eine Großzahl an Verbesserung, durch Hinzufügen von Programmierelementen.
Jedoch sollte man JavaScript nicht mit der Programmiersprache Java verwechseln. Beide sind verschiedene Handelsmarken der Firma Oracle und ähneln einander höchstens mit der Syntax. [23] [24]

4.16 JSON Web Token (JWT)

JSON Web Token ist ein nach RFC 7519 genormter Standard, um Daten sicher zwischen zwei Parteien auszutauschen. Es wird in der Form eines JSON-Objektes übertragen. Die Information, die der Token enthält kann verifiziert werden, weil es eine digitale Unterschrift enthält. Der Token selbst kann entweder mit einem geheimen Schlüssel (HMAC-Algorithmus) oder einem private/public Schlüsselpaar (RSA oder ECDSA) verschlüsselt werden.
Beliebte Anwendungsfälle des Tokens sind Authentifizierung und Datenaustausch. Der Token selbst enthält Informationen über den Absender und ob er die nötigen Zugriffsrechte hat. [25] [26]

4.16.1 Aufbau eines JWT

Ein signierter JWT besteht aus 3 Teilen, getrennt durch einen Punkt. Jeder dieser Teile wird mit Base64 kodiert.
Jeder JWT hat auch eine Gültigkeitsdauer, wenn diese abgelaufen ist, gilt ein Token als ungültig.

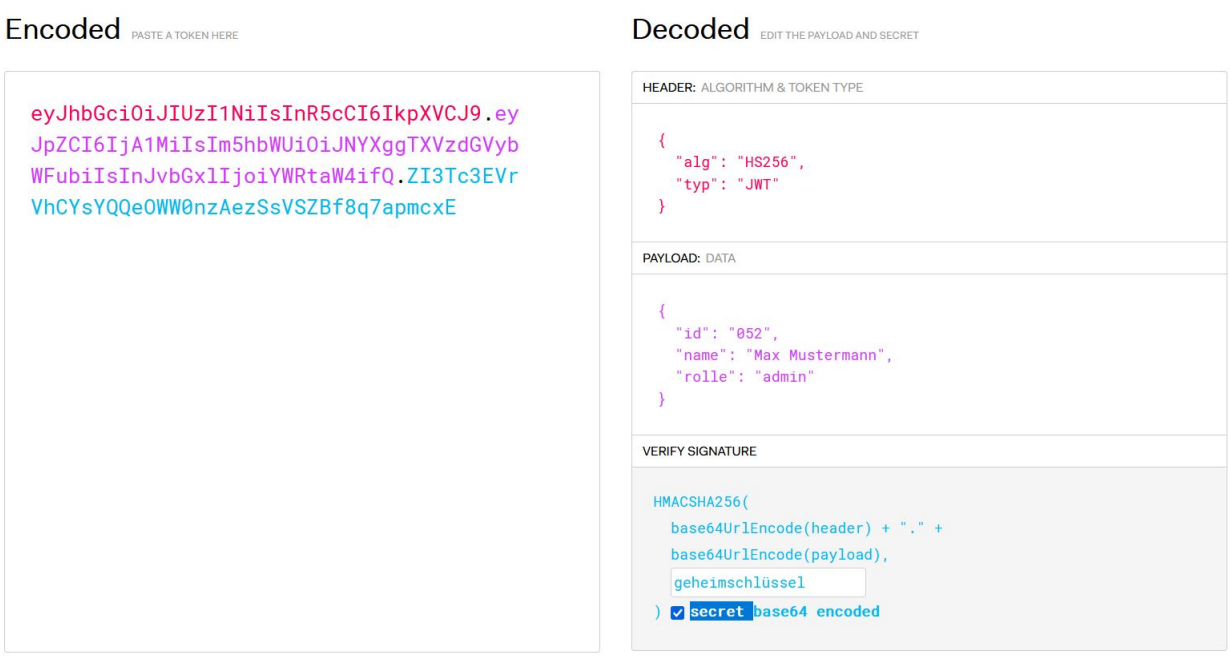

Abbildung 13: Aufbau eines JSON Web Tokens (Ausschnitt von jwt.io)
https://jwt.io/

Header

Der Header besteht meist aus zwei Teilen und liefert Informationen über den Typ des Tokens und den verwendeten Signatur- bzw. Verschlüsselungsalgorithmus.

- Der "typ"-Wert beschreibt den IANA Medientypen des Tokens, oft wird "application/jwt" verwendet.
- Der "alg"-Wert gibt an welcher Algorithmus zum Signieren des Tokens verwendet wurde. Es ist möglich auch keine Verschlüsselung anzugeben, was jedoch nicht zu empfehlen ist. [26]

Payload

Der Payload ist der Teil des Tokens, der die tatsächlichen Daten bzw. Informationen enthält, die übermittelt werden sollen. Sie werden als Key-/Value-Paare bereitgestellt, diese Werte werden bei JWT als Claims bezeichnet. [26] Davon gibt es drei verschiedene Arten:

- **Registrierte Claims** sind standardisiert und im JWT Claim Register festgelegt. Es wird empfohlen diese zu verwenden.
- **Öffentliche Claims** sind nach belieben definierbar.

- **Private Claims** sind für Informationen gedacht, die speziell auf unsere Anwendung angepasst sind wie z.B. "Benutzer-ID". [26]

Signature

Diese wird durch Base64-Kodierung des Headers, des Payloads und der angegebenen Signaturmethode erzeugt. Der Aufbau ist definiert nach dem RFC 7515 Standard, auch genannt JWS (JSON Web Signature). Damit die Signatur funktioniert, muss man einen geheimen Schlüssel verwenden, der nur dem Ursprung bekannt ist. [26]

4.16.2 Sicherungsverfahren

Keine Sicherung

Wenn die Daten keiner Verschlüsselung bedürfen, kann im Header "none" angegeben werden. In diesem Fall wird keine Signatur generiert, dadurch fällt auch der Signature-Teil weg.
Ohne Sicherung lässt sich die Nachricht nach einer Base64-Entschlüsselung klar und deutlich lesen. Der Absender oder ob die Nachricht im Laufe verändert worden ist, ist nicht mehr verifizierbar.[26]

Signatur (JWS)

Im Normalfall reicht es nur zu prüfen, ob die Daten vom richtigen Absender kommen und ob Veränderungen geschehen sind. Da kommt die JWS (JSON Web Signature) zum Einsatz, die genau die vorher genannten Sachen überprüft.
Bei diesem Verfahren lässt sich die Payload nach Base64-Entschlüsselung klar und deutlich lesen. [26]

Signatur (JWS) und Verschlüsselung (JWE)

Es ist möglich zusätzlich zum JWS noch eine JWE (JSON Web Encryption) zu benützen. JWE verschlüsselt den Inhalt des Payloads, diese werden danach mit JWS signiert. Um die Inhalte dann zu entschlüsseln wird noch ein Kennwort oder ein privater Schlüssel angegeben.
Damit ist der Absender verifiziert, die Nachricht authentisch und der Payload ist nicht lesbar nach einer Base64-Entschlüsselung. [26]

4.17 Progressive Web App(PWA)

4.18 Google Charts

4.19 Jetpack Compose

4.20 Docker

4.20.1 Docker Compose

5 Implementierung

Siehe tolle Daten in Tab. 1.

Siehe und staune in Abb. 14. Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

	Regular Customers	**Random Customers**
Age	20-40	>60
Education	university	high school

Tabelle 1: Ein paar tabellarische Daten

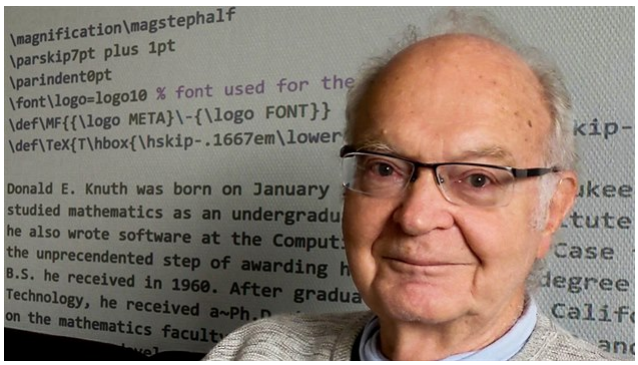

Abbildung 14: Don Knuth – CS Allfather

Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetuer a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetuer. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus. Dann betrachte den Code in Listing 3.

Listing 3: Some code

```
# Program to find the sum of all numbers stored in a list (the not-Pythonic-way)

# List of numbers
numbers = [6, 5, 3, 8, 4, 2, 5, 4, 11]

# variable to store the sum
sum = 0

# iterate over the list
for val in numbers:
    sum = sum+val

print("The sum is", sum)
```

6 Zusammenfassung

Aufzählungen:

- Itemize Level 1
  - Itemize Level 2
    - Itemize Level 3 (vermeiden)

1. Enumerate Level 1
   a. Enumerate Level 2
      i. Enumerate Level 3 (vermeiden)

Desc Level 1

Desc Level 2 (vermeiden)

Desc Level 3 (vermeiden)

Literaturverzeichnis

[1] P. Rechenberg, G. Pomberger *et al.*, *Informatik Handbuch*, 4. Aufl. München – Wien: Hanser Verlag, 2006.

[2] Association for Progressive Communications, „Wireless technology is irreplaceable for providing access in remote and scarcely populated regions," 2006, letzter Zugriff am 23.05.2021. Online verfügbar: http://www.apc.org/en/news/strategic/world/wireless-technology-irreplaceable-providing-access

[3] Jetbrains, „IntelliJ IDEA," letzter Zugriff am 5.11.2021. Online verfügbar: https://www.jetbrains.com/de-de/idea/

[4] Kinsta, „Github erklärt," letzter Zugriff am 5.11.2021. Online verfügbar: https://kinsta.com/de/wissensdatenbank/was-ist-github/

[5] J. Strumpflohner, „Git Explained," 4 2013, letzter Zugriff am 8.11.2021. Online verfügbar: https://juristr.com/blog/2013/04/git-explained/

[6] StefanT123, „Git Commands Explained," *dev.to*, 6 2020, letzter Zugriff am 8.11.2021. Online verfügbar: https://dev.to/stefant123/basic-git-commands-explained-1cjd

[7] „GitHub Seite," letzter Zugriff am 8.11.2021. Online verfügbar: https://github.com/

[8] W. Contributors, „Keycloak," letzter Zugriff am 26.11.2021. Online verfügbar: https://en.wikipedia.org/wiki/Keycloak

[9] B. Żyliński, „What Keycloak Is and What It Does?" *DZone*, 10 2021, letzter Zugriff am 26.11.2021. Online verfügbar: https://dzone.com/articles/what-is-keycloak-and-when-it-may-help-you

[10] CoMakeIT, „A Quick Guide to Using Keycloak for Identity and Access Management," 8 2018, letzter Zugriff am 26.11.2021. Online verfügbar: https://www.comakeit.com/blog/quick-guide-using-keycloak-identity-access-management/

[11] A. İLERİ, „Introduction to Keycloak," *Codex*, 8 2021, letzter Zugriff am 06.01.2022. Online verfügbar: https://medium.com/codex/introduction-to-keycloak-227c3902754a

[12] „Vue Guide," letzter Zugriff am 10.11.2021. Online verfügbar: https://vuejs.org/v2/guide/

[13] G. Roden, „Was man über Vue.js wissen sollte," *heise.de*, 11 2020, letzter Zugriff am 10.11.2021. Online verfügbar: https://www.heise.de/developer/artikel/Was-man-ueber-Vue-js-wissen-sollte-4969211.html

[14] „Vue.js," letzter Zugriff am 10.11.2021. Online verfügbar: https://de.wikipedia.org/wiki/Vue.js

[15] „Model-View-ViewModel," letzter Zugriff am 10.11.2021. Online verfügbar: https://de.wikipedia.org/wiki/Model_View_ViewModel

[16] „Vue Guide Instance," letzter Zugriff am 10.11.2021. Online verfügbar: https://vuejs.org/v2/guide/instance.html

[17] „Vue Guide Components," letzter Zugriff am 24.11.2021. Online verfügbar: https://vuejs.org/v2/guide/components.html

[18] H. Dhaduk, „Angular vs Vue: Which Framework to Choose in 2021?" 6 2021, letzter Zugriff am 24.11.2021. Online verfügbar: https://www.simform.com/blog/angular-vs-vue/

[19] HOSTTEST-Redaktion, „Angular vs. React vs. vue.js," *HOSTtest*, 4 2021, letzter Zugriff am 24.11.2021. Online verfügbar: https://www.hosttest.de/artikel/angular-vs-react-vs-vue-js-was-ist-das-beste-framework

[20] A-Coding-Project, „HTML: Tutorial, Tipps and Tricks," letzter Zugriff am 24.11.2021. Online verfügbar: https://developer.mozilla.org/de/docs/Web/HTML

[21] „HTML," letzter Zugriff am 24.11.2021. Online verfügbar: https://www.seo-kueche.de/lexikon/html/

[22] M. Contributors, „CSS-Grundlagen," letzter Zugriff am 24.11.2021. Online verfügbar: https://developer.mozilla.org/de/docs/Learn/Getting_started_with_the_web/CSS_basics

[23] „JavaScript," letzter Zugriff am 25.11.2021. Online verfügbar: https://de.wikipedia.org/wiki/JavaScript

[24] M. Contributors, „JavaScript," letzter Zugriff am 25.11.2021.

[25] „Introduction to JSON Web Tokens," letzter Zugriff am 25.11.2021. Online verfügbar: https://jwt.io/introduction

[26] I. Writers, „JSON Web Token (JWT) vorgestellt," *Ionos*, 7 2020, letzter Zugriff am 25.11.2021. Online verfügbar: https://www.ionos.at/digitalguide/websites/web-entwicklung/json-web-token-jwt-vorgestellt/

Abbildungsverzeichnis

Tabellenverzeichnis

Anhang