|  |
| --- |
| Mittelstufenprojekt 2019 |
| RezeptFinder |
| Rezeptideen für Jedermann |
|  |
|  |
| Christoph Luber  Dominik Schulze  Felix Stein  Jendrik Dathe  Konstantin Schraubek  Niels Riehtmüller |

Inhalt

[1 Projektbeschreibung 2](#_Toc8206362)

[1.1 Projektumfeld 2](#_Toc8206363)

[1.2 Produktidee 2](#_Toc8206364)

[2 Projektvorbereitung 2](#_Toc8206365)

[2.1 Ist-Analyse 2](#_Toc8206366)

[2.2 Soll-Konzept 3](#_Toc8206367)

[2.3 Auswahl der Werkzeuge 4](#_Toc8206368)

[3 Projektdurchführung 5](#_Toc8206369)

[4 Unternehmerischer Anteil 6](#_Toc8206370)

[4.1 Marktforschung 6](#_Toc8206371)

[5 Projektabschluss 6](#_Toc8206372)

[6 Anhang 6](#_Toc8206373)

# 1 Projektbeschreibung

Der RezeptFinder ist ein Mittelstufenprojekt im Rahmen unserer Ausbildung. Unsere Idee ist es eine Webapplikation zu erschaffen, mit der man durch intuitive Bedienung schnell und ohne Probleme sein gewünschtes Rezept, basierend auf der Eingabe von gewünschten Vorgaben wie zum Beispiel der Kalorienanzahl, finden kann. Für die Realisierung des Projekts haben wir XX Stunden Zeit.

## 1.1 Projektumfeld

Die Firma „Firmenname“ wurde mit dem Eingang des Projektantrags gegründet. Im Rahmen des Projektes wird eine Softwarelösung in Form einer Webanwendung geschaffen, die dazu dient, Rezepte basierend auf den Vorlieben der Nutzer zu finden. Die sechs Gründer Felix Stein, Konstantin Schraubek, Jendrik Dathe, Dominik Schulze, Christoph Luber und Niels Riethmüller erachten den Markt als ausbaufähig und überaus lukrativ. Im Team wurde die Idee entwickelt und diese in eine agile Softwarelösung umgesetzt.

## 1.2 Produktidee

Die Idee ist eine umfangreiche Webanwendung für das Suchen nach Rezepten für koch freudige, aber auch für Personen, welche aus dem Kochalltag ausbrechen wollen und nach neuen koch alternativen suchen.

Viele Menschen haben das Problem, dass sie in ihrem Rezept Repertoire sehr eingeschränkt sind und wenige Ideen für neues haben. Zudem stecken viele Menschen ihrer kulturellen Küche fest und schauen nur selten über den exemplarischen Tellerrand. Mit unserem Produkt kann man ganz einfach wenige Zutaten suchen und daraus komplexe und leckere Gerichte vorgeschlagen bekommen. Natürlich wird auch auf die Essgewohnheiten der Nutzer eingegangen, man kann unter Anderem Zutaten komplett ausschließen oder auch nach bestimmten Diäten filtern. Um auch Personen, welche nicht unbedingt die Zeit zum Kochen finden, da diese einen stressigen und ereignisreichen Alltag haben, dennoch zum Kochen zu motivieren, existiert ebenso die Möglichkeit nach benötigter Zeit für ein Rezept filtern zu können. So soll es Möglich sein auch unter Zeitdruck leckere und aufregende Rezepte nachzukochen und dem inneren Chefkoch freien Lauf zu lassen. Um die etwas jüngere Demografie anzusprechen, haben wir uns für eine Webapplikation entschieden, da diese jederzeit auch von einem Mobilgerät aus aufgerufen werden kann. Heut zutage wissen viele junge Menschen nicht einmal was ein Buch ist oder es ist viel zu umständlich sich ein solches zu besorgen. Rezeptbücher sind einfach nicht mehr modern. Umso einfacher ist es kurz seinen Webbrowser zu öffnen und nach dem Rezeptfinder zu suchen, um dort je nach Heißhunger sein Rezept zu finden. Dies spart Zeit und vor allem Raum, welcher nicht mehr für unzählige Kochbücher bereitgestellt werden muss.

# 2 Projektvorbereitung

## 2.1 Ist-Analyse

Betrachtet man den Markt wird schnell klar, dass es schon viele ähnliche Produkte gibt. Unser Ziel ist es jedoch das Produkt einfacher und intuitiver, gerade in der Bedienung, zu gestalten. Auffällig ist das viele Konkurrenzprodukte sehr überladen wirken. So soll mit der leichten Bedienbarkeit und dem minimalistischen Design eine größere Zielgruppe angesprochen werden. Außerdem soll eine Registrierung nicht gezwungen sein, der Nutzer hat alle notwendigen Funktionen um nach Rezepten suchen und genauer filtern zu können. Keineswegs soll der Nutzer in seiner Rezept Suche dadurch eingeschränkt werden, dass er nicht bei uns registriert ist. Nur spezielle Funktionen, wie ein interner Verlauf und das Hinzufügen von Favoriten ist den registrierten Benutzern vorbehalten.

## 2.2 Soll-Konzept

In diesem Abschnitt werden der Funktionsumfang sowie die Anforderungen an die zu entwickelnde Software detailliert beschrieben.   
Die Anwendung soll über eine Landingpage verfügen, die der Nutzer als Erstes sieht.

Diese soll in zwei Hauptteile geteilt sein, oben soll eine Navigationsleiste und das Firmenlogo sein, unten eine Suchleiste.

Die Navigationsleiste besteht aus Home, Login/SignUp und Contact. Mit dem Unterpunkt Home gelangt man immer wieder zurück auf die Landingpage. Hinter dem Login/SignUp Button verbirgt sich die Anmeldung und Registrierung. Hier soll man sich mit einer E-Mail-Adresse, einem Nutzernamen und einem Passwort registrieren können, dies soll aber nicht notwendig für den Großteil der Benutzung der Seite sein. Wenn man schon einen Account besitzt, soll man sich auf der Login-Seite anmelden können. Es soll nur wenige exklusive Features für registrierte, beziehungsweise angemeldete Benutzer geben. Zu diesen Features soll ein Verlauf der angeschauten Rezepte zählen sowie eine Funktion, welche es möglich machen soll, Rezepte zu favorisiere, um diese schnell wieder aufrufen zu können. Angemeldete Benutzer sollen zudem die Möglichkeit haben ihre E-Mail-Adresse und ihr Passwort jederzeit ändern zu können. Der Unterpunkt Contact soll zum Impressum und den Kontaktdaten führen.

Mit der Suchleiste soll man die Möglichkeit haben nach verschiedensten Rezepten suchen zu können. Anschließend sollen die Suchergebnisse auf einer separaten Seite angezeigt werden. Auf dieser separaten Liste soll es die Möglichkeit geben mit weiteren Filtern die Ergebnisse weiter einzuschränken. Dies soll über eine Filtermaske ermöglicht werden, in welcher man die Option haben soll, verschiedene Filter auf die gefundenen Rezepte anzuwenden. Zu diesen Filtern soll unter anderem die Möglichkeit nach verschiedenen Diäten suchen zu können bestehen. Weitere Alternativen für das Filtern von Diäten sollen zum Beispiel „Low-Carb“ und „Low-Fat“ sein, aber auch andere Diät Optionen sollen vorhanden sein. Zudem soll es die Möglichkeit geben nach der benötigten Zeit für ein Rezept filtern zu können. Des Weiteren sollen Zutaten ausgeschlossen werden können und es soll die Möglichkeit geben einen Kalorienbereich festzulegen. Die Filter sollen beliebig miteinander kombinierbar sein.

Die Suchergebnisse sollen in Kartenform angezeigt werden. In diesen Karten soll der Name des Rezeptes, ein Bild des Gerichts, eine kurze Beschreibung der Zutaten und der Zeitaufwand des zu kochenden Rezepts angezeigt werden. Es soll möglich sein mit diesen Karten zu interagieren um somit auf eine externe Seite zu gelangen, welche genaue Angaben zum Rezept beinhalten soll. Auf diesen externen Seiten sollen im Idealfall ein großes Bild, alle Zutaten, alle Inhaltsstoffe, sowie dessen prozentualer Anteil am Tagesbedarf und die Diät Typen des Rezepts angezeigt werden. Am wichtigsten ist es jedoch, dass man durch das interagieren auf die genaue Anleitung des Rezepts verwiesen wird.

## 2.3 Auswahl der Werkzeuge

Das Produkt wird nicht über herkömmliche Wege wie CDs oder den Download angeboten, sondern als Webanwendung. Dies hat viele Vorteile. Man hat von überall, vorausgesetzt es besteht eine Verbindung zum Internet, Zugriff auf den RezeptFinder. Außerdem wird unser Produkt dadurch plattformunabhängig und kann auch von Smartphones und Tablets unterwegs aufgerufen werden. Daher benutzen wir für den Webentwicklungsbereich gängige Werkzeuge.

Aufgebaut ist unsere Software auf dem verbreiteten Webframework Angular. Angular bietet leicht die Möglichkeit komplexe Single-Page-Applikationen basierend auf JavaScript zu erstellen. Bei einer Single-Page-Applikation wird der Nutzer auf einer Seite gehalten, diese ändern aber je nach Aktion der Nutzer ihr Aussehen dynamisch. Dieses Verhalten ähnelt stark dem Verhalten nachdem man bei Google eine Suchanfrage gestellt hat. Ein weiteres gutes Beispiel ist zudem Googles E-Mail-Dienst Gmail. Öffnet man Gmail im Webbrowser, so wird die Anwendung geladen und einige Daten werden zwischengespeichert, darunter die Informationen zu einigen E-Mails. Deaktiviert ihr nun die Internetverbindung, könnt ihr trotzdem alle Mails öffnen und euch sogar die Nachrichten anzeigen lassen. Lediglich einige Bilder können nicht richtig geladen werden. Neben Gmail sind auch Google Maps und Twitter als Single-Page-Application aufgebaut.

Neben Angular wird für die Positionierung der Webelemente weiterhin auf HTML vertraut. TypeScript ist ein weiterer wichtiger Teil unserer Entwicklung, TypesScript ist eine auf JavaScript basierende Scriptsprache, allerdings im Gegensatz zu JavaScript typisiert, was einige Sachen deutlich erleichtert und viel mehr Übersicht in die Programmabläufe bringt. Weil die Webbrowser TypeScript nicht verstehen wird bei der Erstellung der Webanwendung der TypeScript Anteil in, für die Browser verständliches, JavaScript übersetzt. Dieses ist für Menschen aber dann sehr schwer zu verstehen.

Auf Angular-basierende Seiten haben noch einen gewaltigen Vorteil mehr, der Server welche die Webapplikation zur Verfügung stellt schickt alle erforderlichen Daten an den Client. Dieser baut die Seite dann komplett selbst auf und ist vom Server “getrennt”. Das reduziert die Serverlast, gerade bei viel besuchten Seiten (siehe Beispiel Twitter) erheblich. Der Browser interpretiert dann die Benutzereingaben und passt die Seite selbstständig an ohne erneut auf den Server zuzugreifen.

Die Arbeit mit den Daten erfolgt mithilfe von Viewmodels, das heißt, dass jede Instanz in der Datenbank über eine Repräsentation in Form einer Klasse in TypeScript verfügt. Bei uns trifft das nur auf die registrierten Benutzer der Seite und deren favorisierten Rezepte zu.  
Um die favorisierten Rezepte der Nutzer und die Daten der Nutzer selbst dauerhaft zu speichern wird eine relationale Datenbank verwendet. Wir haben uns für SQLite wegen der sehr leichten Handhabung und Größe der Datenbank entschieden. Um die Datenbank der Angular Seite zur Verfügung zu stellen haben wir uns entschieden ein eigenes Backend zu schreiben. Dieses läuft über NodeJS und wurde mit JavaScript entwickelt. Es ist für die Handhabung der Datenbank zuständig und verbindet diese mit dem Frontend und kümmert sich um die Bereitstellung der Datenbankdaten. Die Verknüpfung zwischen Front- und Backend wurde dabei großteilig mit Socket.io umgesetzt. Diese Bibliothek ermöglicht es im Prinzip Funktionen innerhalb des Backends zu erstellen welche nur auf einen vorher definierten Emit reagieren. Dies hilft Übersicht im Backend zu behalten und im Frontend die Backend Aufrufe so einfach wie Möglich zu gestalten.

Für die sichere Aufbewahrung der Nutzerpasswörter haben wir die 3rd Party JavaScript Klasse “Crypto-JS” genutzt. Diese biete viele verschiedene Verschlüsslungs-Algorithmen, von AES bis SHA.

Die Rezeptdaten bekommen wir mit Hilfe der Edamam API. Unter einer API versteht man eine Programmierschnittstelle, im Falle von Edamam ist es eine WebAPI. Diese bietet die Möglichkeit mit vordefinierten Anfragen Daten gezielt abzurufen. Will ein Besucher unserer Seite zum Beispiel Hänchen Rezepte angezeigt bekommen, so wird an die Edamam API ein Post geschickt welcher im Prinzip aussagt, dass alle Rezepte, welche Hähnchen beinhalten aus der Edamam Datenbank herausgefiltert werden sollen. Die API schickt dann eine formatierte Antwort, im Fall der Edamam API im JSON Format, über alle Rezepte welchen Hähnchen beinhalten. Diese Response können wir dann auslesen und auf unserer Seite visualisieren. Die Datensammlung von Edamam wird ständig aktualisiert und bekommt immer mehr Rezepte. Die Daten von Edamam sind sehr detailliert, jedes Rezept ist genau beschrieben, von Nähr- und Inhaltsstoffen, über Kolorieren, bis hin zu verschiedensten Diäten.

Um mit einem Team von 5 Entwicklern gleichzeitig an dem Projekt arbeiten zu können, haben wir das Projekt intensiv geplant. Mit dem Projektplanungskonzept “Scrum” (siehe Abschnitt 3.1) kann agiles Arbeiten, welches in der Softwareentwicklung schon lange Anwendung findet, gut umgesetzt werden. Außerdem haben wir die Versionsverwaltungs-Software Git benutzt. Mit Hilfe von Git werden die Quell- und Projektdateien auf einem Git-Server geladen und somit nicht mehr Lokal gehalten. Dadurch können auch mehrere Entwickler an derselben Datei arbeiten. Sollte es zu dem Fall kommen, dass mehrere Entwickler an der gleichen Datei arbeiten und es somit zu unterschiedlichen Änderungen an der gleichen Datei kommt, bietet Git auch dafür eine Lösung. Git führt die Änderungen beider Dateiversionen intelligent, mithilfe einiger Angaben des Nutzers, zusammen. Ein Versionierungs-System dient außerdem dazu Änderungen an der Software verfolgen zu können.

# 3 Projektdurchführung

3.1 Organisation  
Das Projekt wurde nach dem Vorgehensmodell SCRUM erarbeitet. Unserer Scrum Management wurde mit der Webanwendung FlyingDonut umgesetzt.  
Bei Scrum wird die ganze Aufgabe in kleine, detaillierte Aufgaben zerlegt. So wird aus “Datenbank machen”, “Datenbank planen, Datenbank auswählen, Datenbank aufsetzen etc.). Weil die Aufgabe in viele kleine Zerlegt wird können mehrere Entwickler leicht daran arbeiten, da jede Aufgabe für sich Unabhängig ist. Außerdem wird bei Scrum die Arbeit in sogenannten “Sprints” geplant. Ein Sprint ist üblicherweise zwei Wochen lang. Am Anfang jedes Sprints steht die Sprintplanung, in der wird bestimmt welche der kleineren Aufgabenpakete in aktuellen Sprint bearbeitet werden sollen und während des aktuellen Sprints werden auch nur diese bearbeitet. Jedoch kann es dazu kommen, dass neue Probleme und Baustellen entstehen. In diesem Fall besteht auch die Möglichkeit neue Aufgaben in den Sprint einzuplanen. Wer mit einer Aufgabe fertig ist, sucht sich eine neue Aufgabe aus dem Backlog (Sammlung aller kleineren Aufgaben).  
Nach jedem Sprint findet ein Sprint-Review statt, in diesem wird über die Ergebnisse des Sprints gesprochen und überprüft ob die Ziele erreicht wurden.

## 3.2 Implementierungsdetails

dadad

## 3.3 Einschränkungen

Im Verlauf des Projekts kam es zu unvorhersehbaren Problemen. Die das planmäßige fertigstellen beeinträchtigt haben. Einige Verzögerungen kamen durch Krankheit im Team zustande. Anhand der mangelnden Erfahrung im Scrum Ablauf kam es bei schätzen der Zeiten zu erheblichen Fehleinschätzungen, teilweise haben wir für kurze Aufgaben deutlich zu viel Zeit geplant, wiederrum bei langen Aufgaben deutlich zu wenig. Das hat die Projektplanung maßgeblich beeinflusst.   
Bei der Umsetzung der Datenbank hatten wir schwerwiegende Probleme, sodass wir im Zuge des Projektes das Datenbanksystem zweimal komplett neu gemacht haben. Zudem ist mehr Zeit als nötig in die Planung der Aufgaben geflossen.

# 4 Unternehmerischer Anteil

4.1 Marktforschung  
Die Markforschung ist für den Lifecycle eines Produktes sehr wichtig. Sie dient dazu Informationen über die Zielgruppe und die Konkurrenz in Erfahrung bringen. Außerdem werden wichtige Hinweise über Zufriedenheit und Kaufverhalten, sowie neuste Trends und den Zusammenhang dieser, mit der Auswirkung auf die potentiellen Benutzer gesammelt. Es gilt dann all diese Informationen und Hinweise auszuwerten und zu Verstehen. *Aus den gewonnenen Erkenntnissen werden wir in diesem Abschnitt, aus den Ergebnisse, eine Markt- und Konkurrenzanalyse, sowie eine Wirtschaftlichkeitsbetrachtung darstellen.*

## 4.1.1 Marktanalyse

## 4.1.2 Konkurrenzanalyse

## 4.2 Marketingkonzept

## 4.2.1 Kommunikationspolitik

Eines der größten Probleme für neue Produkte ist es von der Masse wahrgenommen zu werden. Um dieses Problem anzugehen, haben wir uns ein Werbekonzept ausgedacht. Dieses umfasst die Schaltung von Online Werbung über Google AdSense, hierfür werden wir verschiedene Werbebanner erarbeiten.

Über Google AdSense ist es kinderleicht Zielgruppen gerechte Werbung zu schalten. Google analysiert das Nutzerverhalten und sucht Anhand dieser Kriterien passende Werbung, die dann wiederrum über AdSense dynamisch geschaltet wird. Für unseren RezeptFinder ist die anvisierte Zielgruppe recht groß. Jeder Mensch hat Berührungspunkte mit dem zubereiten von Speisen. Essen ist ein Grundbedürfnis, wir wollen den Menschen eine Möglichkeit geben mit wenigen Klicks und Eingaben ein Gericht zu finden.

# 5 Projektabschluss

## 5.1 Qualitätssicherung

Qualitätssicherung ist der Grundstein bei jedem guten Produkt. Um in unserem Produkt die Qualität zu gewehrleisten haben wir Tests für den Programmcode und die Integrität der einzelnen Funktionen erstellt.

Um die Qualität weiterhin zu verbessern sind in Zukunft mehrere Projekte geplant. Unteranderem sollen Nutzer zufällig ausgewählt werden um an einer Zufriedenheitsumfrage teilzunehmen. Hier können sie gegeben falls auch Verbesserungsvorschläge und Wünsche äußern. *Weiterhin ist eine enge Zusammenarbeit mit von der Universität in Zürich geplant.*

## 5.2 Projektziel/Fazit

Wir betrachten unser Projekt als Erfolg und dahingehend das Ziel als erreicht. Viele Funktionen die im Soll-Konzept aufgezeigt wurden sind umgesetzt.

## 5.3 Ausblick

Für die Zukunft haben wir geplant unser Produkt noch um eine mobile App zu erweitern. Eine App ist für viele Benutzer ansprechender, da heute viel mehr Menschen ihr Smartphone als „Computer“ benutzen. Eine App sehen die meisten Menschen als bequemer, da man diese sofort öffnen kann und nicht erst im Browser nach der richtigen Webseite suchen muss. Eine App reduziert außerdem das verbrauchte Datenvolumen.

Weiterhin ist eine Einbindung von verfügbaren Lieferdiensten in der Gegend geplant. Wenn der Nutzer sich also für ein Gericht entschieden hat, wird ihm die Möglichkeit geboten dieses, wenn verfügbar, über lokale Lieferdienste zu ordern. Dieses Funktion bietet uns die Option eine Provision von den Lieferdiensten zu beziehen.

# 6 Anhang