

Duale Hochschule Baden-Württemberg Mannheim

Portfolioprüfung Fallstudie

Mission Smile – We create happiness

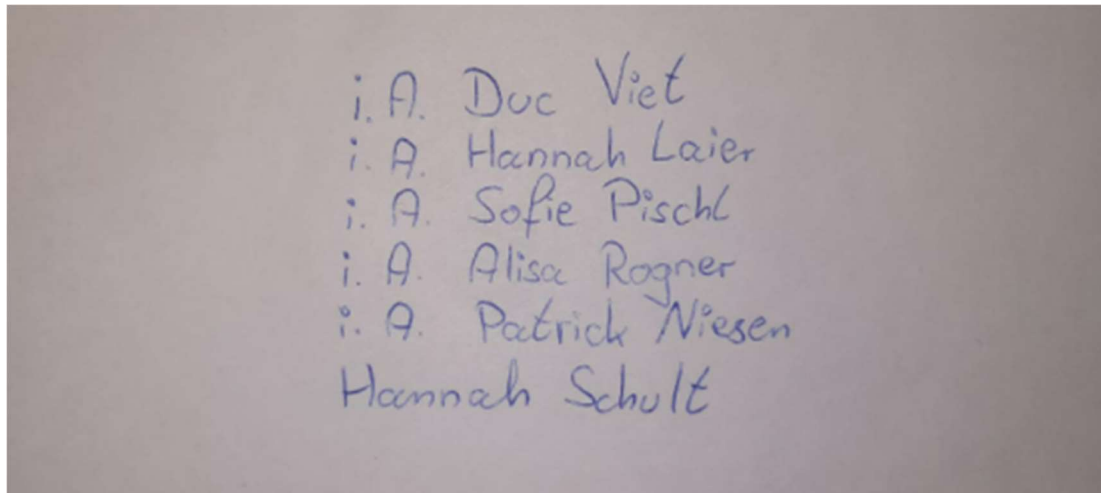
Studiengang Wirtschaftsinformatik

Data Science

Verfasser(in):	Duc Viet (9588548), Hannah Laier (1010595), Sofie Pischl (3943911), Alisa Rogner (7894464), Patrick Niesen (1137132), Hannah Schult (5373022)
Kurs:	WWI-20-DSB
Vorlesung:	Fallstudie
Dozent:	Benjamin Jung
Bearbeitungszeitraum:	16.11.2021 – 28.02.2022

Ehrenwörtliche Erklärung

Wir versichern hiermit, dass wir die vorliegende Arbeit mit dem Titel "Mission Smile – We create happiness" selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt haben.



Inhaltsverzeichnis

Inhaltsverzeichnis

Ehrenwörtliche Erklärung	ii
Inhaltsverzeichnis.....	iii
Abbildungsverzeichnis	iv
Glossar	v
1. Vision und Projekt.....	1
1.1. Vision.....	1
1.2. Projektplanung.....	2
1.3. Design.....	2
2. Projektvorgehen	5
2.1. Risiko-Analyse und Vorgehensweise im Krisenfall	5
2.2. Ausgearbeitetes SLA.....	5
3. Wirtschaftliche Analyse	8
3.1. Kosten/Nutzen Analyse.....	8
3.2. SWOT-Analyse	9
3.3. Auswirkungen auf Marketing/Branding.....	10
3.4. Einfluss auf die Supply Chain und Bestandsanalyse.....	11
3.5. Zielgruppenanalyse	12
4. Systementwurf	13
4.1. Ereignis-Reaktionsmodell	13
4.2. Aktivitätsdiagramm.....	13
4.3. Entscheidungstabelle.....	14
4.4. Beschreibung der verwendeten Algorithmen.....	14
4.5. Use-Case Schablone	15
5. Feedback und Zwischenstände	17
5.1. Zwischenstand Termin 07.01.2022.....	17
5.2. Zwischenstand Termin 13.01.2022.....	18
6. Applikation	20
6.1. Grundkonzepte.....	20
6.2. Technische Umsetzung.....	24
6.3. Probleme und deren Lösungen	25
7. Kritische Reflektion	28
8. Fazit.....	30

Abbildungsverzeichnis

Abbildungen

Abbildung 1 Design Mobile Version	3
Abbildung 2 Design Desktop Version	3
Abbildung 3 Neues Design Desktop Version	3
Abbildung 4 Logo Mission Smile.....	4
Abbildung 5 Beispiel Social Media Post.....	10
Abbildung 6 Ereignis-Reaktionsmodell.....	13
Abbildung 7 Aktivitätsdiagramm	13
Abbildung 8 1. Feedback Scales	17
Abbildung 9 1. Feedback Open.....	18
Abbildung 10 Design 2.0.....	18
Abbildung 11 Trello Board.....	19
Abbildung 12 Architektur	22

Glossar

Cold-calling: Eine Form der Kaltakquise, bei welcher bisher unbekannte Kunden angerufen werden, um Ihnen ein Produkt zu verkaufen

Bild-tag: Alternative Bildbeschreibung für blinde Menschen

Feature Extraction: Extraktion der primären Merkmale für die Weiterverarbeitung mit Algorithmen

Neuronales Netz: Ein Machine-Learning Algorithmus, bei dem künstliche Neuronen verwendet werden

Firebase: Eine Entwicklungs-Plattform, die verschiedene Funktionen und Werkzeuge für Entwickler anbietet

Web-Crawler: Softwareprogramme, welches das Internet durchsucht

React: JavaScript Softwarebibliothek, die ein Grundgerüst für die Ausgabe von User-Interface Komponenten zur Verfügung stellt

Next.js: Open-Source-Entwicklungsframework, das React-basierte Webanwendungsfunktionen wie serverseitiges Rendern und Generieren statischer Websites ermöglicht. Next js basiert auf React.

Deployment: Bereitstellung von Software zur produktiven Anwendung

Continuous Integration, Continuous Deployment (CI/CD): Ein Konzept, bei dem Integrations-, Tests- und Bereitstellungsphasen der Anwendung automatisiert werden.

Technology-Stack: Eine Liste aller Technologiedienste, die zur Entwicklung der Anwendung benötigt werden.

Rehydration: Ein Prozess in Next js, bei dem dynamische Inhalte einer Webseite heruntergeladen werden

Content Delivery Network (CDN): Ein Netz geografisch verteilter Server, welcher die Bereitstellung von Webinhalten beschleunigt

Application Programming Interface(API): Eine Programmierschnittstelle, die zur Anbindung an ein Softwaresystem verwendet wird.

Frontend: Teil der Anwendung, die der Benutzer sehen kann.

Backend: Das technische System hinter der Anwendung.

Framework: Framework stellt ein Programmiergerüst zur Verfügung, welches die Struktur einer Anwendung vorgibt und das Programmieren erleichtert.

Microservice: Das ist ein architekturbezogener und organisatorischer Ansatz in der Softwareentwicklung. Die Software besteht aus unabhängigen Services, die über APIs kommunizieren.

React MUI: Design-Bibliothek, die die Erstellung von Designkomponenten in React vereinfacht.

Heroku: Eine Plattform-as-a-Service, die das Hosten von Anwendungen anbietet

Search Engine Optimization: Maßnahmen, die die Sichtbarkeit einer Webseite im Internet verbessern.

1. Vision und Projekt

1.1. Vision

Laut der Peakon Studie aus dem Jahr 2014 geht fast jeder vierte Deutsche (23%) täglich unmotiviert ins Büro. Das ist nicht nur schlecht für Arbeitnehmer, sondern schadet auch Arbeitgebern, denn wenig motivierte Mitarbeiter kündigen eher und haben durchschnittlich bis zu 75% mehr Krankheitstage. Dies kann ein Unternehmen mit 10.000 Mitarbeitern jährlich über 48 Millionen Euro kosten¹.

Neben diesen Zahlen ist auch zu beachten, dass zufriedene Mitarbeiter glücklicher sind und eine bessere Arbeitsatmosphäre erzeugen, was dann wieder zu einer besseren Stimmung im Unternehmen führt. Deshalb ist es äußerst relevant dafür zu sorgen, dass die Mitarbeiter sich im Arbeitsalltag wohl fühlen und zufrieden sind.

Mission Smile hat sich zur Aufgabe gemacht, diese Mitarbeiterzufriedenheit zu messen und im Folgenden zu steigern. Dazu wird eine Webanwendung entwickelt, die mit Hilfe einer Emotionserkennung die Stimmung der Mitarbeiter analysiert und auswertet.

Diese Emotionserkennung funktioniert auf Basis von Bildern, die von den Mitarbeitern des Unternehmens selbst zur Verfügung gestellt und an die Anwendung übergeben werden. Die Ergebnisse der Emotionsanalyse werden mithilfe eines Stimmungsbarometers visualisiert. Außerdem werden den Mitarbeitern bei negativen Emotionen sofort Empfehlungen gegeben, wie die aktuelle Stimmung verbessert werden kann. Zusätzlich wird die durchschnittliche Stimmung im Unternehmen oder in einzelnen Bereichen erfasst und anonymisiert an Vorgesetzte übergeben, sodass diese Maßnahmen ergreifen kann, um die Atmosphäre in den Abteilungen und damit im ganzen Unternehmen zu verbessern.

¹ <https://www.businessinsider.de/wirtschaft/studie-eine-eigenschaft-der-generation-z-ist-bei-arbeitgebern-besonders-gefuehrtet-2019-6/#:~:text=Peakon%20untersucht%20seit%202014%20weltweit%20die%20Motivation%20von,Zielen%20und%20Mission%20ihres%20Unternehmens%20inspiriert%20zu%20sein>

1.2. Projektplanung

Um das Ziel der Verbesserung der Mitarbeiterzufriedenheit zu verwirklichen, wurde das Projekt in verschiedene Aspekte unterteilt.

Diese waren der Designprozess, die wirtschaftliche Analyse, die Entwicklung der Software, sowie die Erstellung von Feedback und die Präsentation von Zwischenständen. Während die Designfindung und die wirtschaftliche Analyse vor allem zu Beginn des Projektes eine zentrale Rolle gespielt haben, waren die Software und das Feedback Themen, die über die gesamte Projektlaufzeit bearbeitet wurden. Um diese Aufgaben effizient zu bearbeiten, wurden die verschiedenen Themenbereiche im Team aufgeteilt und separat bearbeitet. In regelmäßigen Abständen haben sich dann alle Teammitglieder getroffen, um über Fortschritte, sowie Herausforderungen zu besprechen.

1.3. Design

Zunächst wurde das grundsätzliche Design gestaltet, das später als Orientierung für die visuelle Ausgestaltung dienen soll. Während der Programmierung der Anwendung werden jedoch gegebenenfalls einzelne Komponenten angepasst, um die Erstellung der Anwendung zu vereinfachen. Das konzeptionelle Design wird im Folgenden gezeigt.

Vor Login in die App soll eine Startseite gezeigt werden, die sowohl unser Logo als auch das Login Formular enthält. Nach erfolgreichem Einloggen in die App, soll der Nutzer zur Startseite gelangen, auf der die Stimmungsbarometer, sowie ein Button zum Foto hochladen ist. Daraufhin wird nach hochladen des Bildes die Stimmung erkannt, und ein passender Vorschlag zur Verbesserung generiert.

Insgesamt existieren zwei verschiedene Designvorschläge, da unsere Seite sowohl auf einem mobilen Gerät als auch auf einem Computer angezeigt werden soll.

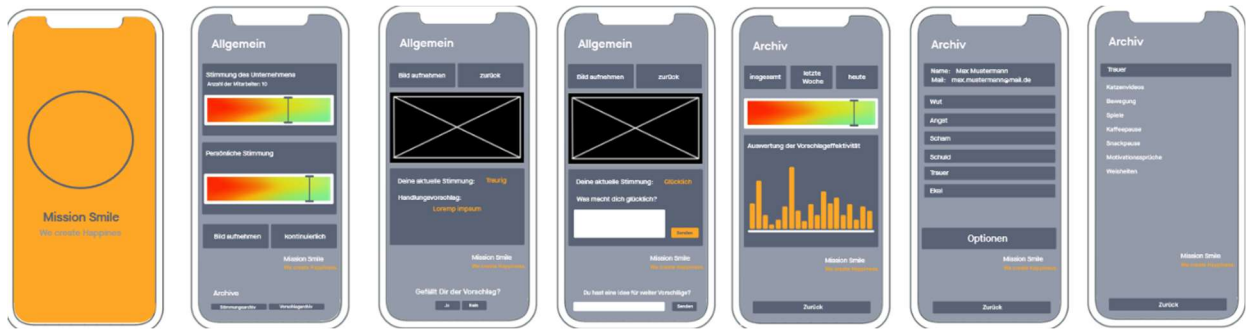


Abbildung 1 Design Mobile Version

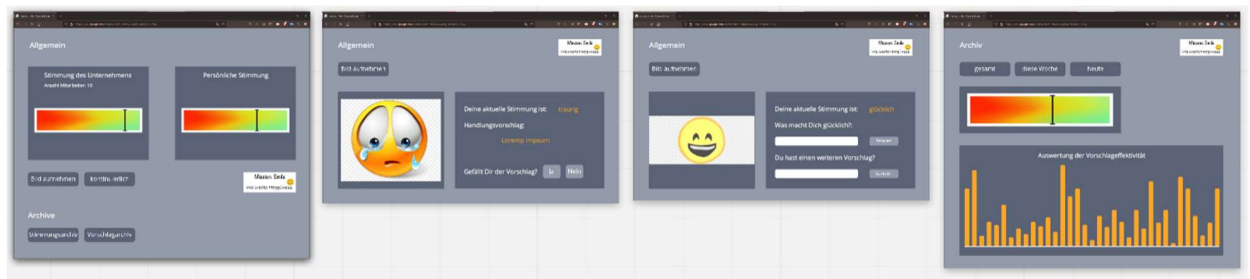


Abbildung 2 Design Desktop Version

Neue Version des Designs

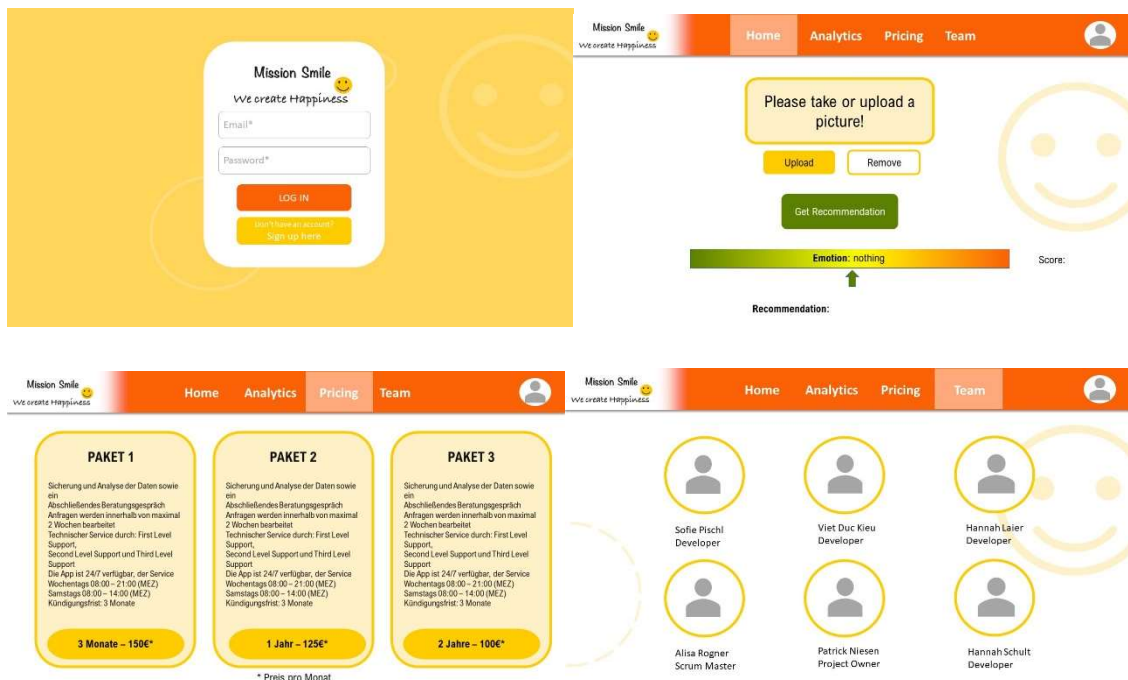


Abbildung 3 Neues Design Desktop Version

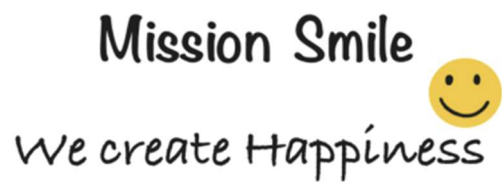


Abbildung 4 Logo Mission Smile

Zusätzlich zu den verschiedenen Designs wurde ein Logo entworfen, das als Erkennungsmerkmal unserer Firma dienen soll. Dieses besteht wie in Abbildung 4 zu sehen aus unserem Motto, sowie unserem Namen und dem Logo - einem lächelnden Emoji, der für die gute Laune steht, die wir verbreiten wollen.

2. Projektvorgehen

Als Arbeitsweise wurde die Scrum-Methode gewählt. Daily Meetings finden dabei alle zwei bis drei Tage statt, wobei ein Sprint zwei Wochen dauert. Die Aufgaben für das Backlog werden kontinuierlich erstellt und es gibt einen Sprintwechsel aus Review, Retrospektive und Planung. Die Meilensteine bilden die Zwischenstände für die einzelnen Treffen. Das Development Team bilden Viet Duc, Hannah S., Hannah L. und Sofie. Scrum Master ist Alisa und Product Owner ist Patrick. Für die Kommunikation verwenden wir Discord, als Übersicht für die Aufgaben wird Trello genutzt und für das Feedback Mentimeter. Die Abstimmung mit dem Kunden findet in den dafür vorgesehenen Vorlesungen statt.

2.1. Risiko-Analyse und Vorgehensweise im Krisenfall

Im Krisenfall wird ein außerplanmäßiges Meeting zur Problemlösung einberufen. Konkret könnte es Probleme bei den verteilten Aufgaben geben. In dem Fall ist frühe Kommunikation wichtig und das Erfragen von Hilfe. Sollte eine Aufgabe in vorgesehener Zeit nicht zu lösen sein, wird zur Prävention ein Zeitpuffer eingebaut und bei Eintritt wird der Zeitplan angepasst. Vorab wird realistisch geschätzt wie viel Zeit pro Aufgabe benötigt wird.

2.2. Ausgearbeitetes SLA

Das Service Level Agreement wird dem Kunden vor Erbringung der Leistung übergeben.

1. Zweck

Klare Aufteilung der Aufgaben beider Seiten, um Spannungen zu vermindern und zu vermeiden.

2. Vertragspartner

Vertragspartner sind die Mission Smile AG und Kunde X.

3. Leistungsbeschreibung

Wir decken folgende Leistungen ab: technischen Service und Betrieb durch First Level Support, Second Level Support und Third Level Support, Behebung von Systemfehlern, Bearbeitung von Anwendungssystemausfällen, Kundenberatung und Kundenbetreuung, Wartung, Sicherung der Daten, Analyse der Daten, Abschließendes Beratungsgespräch und jährliche Besprechungen.

4. Verantwortung des Dienstleisters

Unter die Verantwortung des Dienstleisters fällt folgendes: technischen Service und Betrieb durch First Level Support, Second Level Support und Third Level Support, Behebung von Systemfehlern, Bearbeitung von Anwendungssystemausfällen, Tickets innerhalb von fünf Stunden – eine Woche – zwei Wochen lösen je nach Dringlichkeit, Anfragen innerhalb von zwei Wochen lösen und Vereinbarungen einhalten.

5. Verantwortung des Kunden

Folgendes erwarten wir von unseren Kunden: Bereitstellung eines Ansprechpartners, rechtzeitige Kostendeckung, Einhaltung von Vereinbarungen und Motivation der Mitarbeiter zur Nutzung der App.

6. Verfügbarkeit des Service

Wochentags sind wir von 08:00 – 21:00 (MEZ) erreichbar und samstags von 08:00 – 14:00 (MEZ). Ausnahmen bei Updates, Stromausfall, Netzwerkausfall. Die App und alles damit direkt Verbundene ist 24 Stunden am Tag, sieben Tage die Woche verfügbar.

7. Service – Level – Kennzahlen

Kundenzufriedenheit bis zu einmal im Quartal (via Fragebögen), die pro Tag gemachten Bilder und die Mitarbeiteranzahl, welche die App verwenden.

8. Monitoring und Reporting

Es werden die Kennzahlen und Abweichung von vereinbarten Standards überprüft.

9. Eskalationsmanagement

Ansprechpartner auf beiden Seiten bekannt.

10. Preisgestaltung

Die Fixkosten sind im Vertrag festgelegt, zu den variablen Kosten gehören auch die Extraleistungen.

11. Rechtsfolgen bei Nichteinhaltung

Folgen sind fristlose Kündigungen, eine Strafzahlung oder Rückerstattung.

12. Vertragslaufzeit

Die typische Vertragslaufzeit geht beispielsweise vom 01.01.2022 – 01.01.2023. Es kann bis zu einem Monat vor Vertragsanfang zurückgetreten werden und die Kündigungsfrist liegt bei 3 Monaten, ansonst verlängert sich der Vertrag automatisch um ein weiteres Jahr. Verfügbar sind auch Abos mit kürzeren Laufzeiten. Auf Anfrage ist ein Testabonnement in Länge von drei Monaten verfügbar.

3. Wirtschaftliche Analyse

In Bezug auf die Wirtschaftliche Analyse wurden bereits zu Beginn des Projektes die Bereiche Kosten/Nutzen Analyse, Swot Analyse, eine Untersuchung der Auswirkungen auf Marketing/Branding und des Einflusses auf die Supply Chain und den Bestand sowie eine Zielgruppenanalyse durchgeführt.

3.1. Kosten/Nutzen Analyse

Primär fallen die Kosten zum Erwerb der Lizenz für die Kosten der App an. Diese sind vor allem abhängig von der geplanten Nutzungsdauer des Endkunden, die im Service Level Agreement bereits erläutert wurde.

Die Kosten, die zusätzlich für unser Produkt anfallen, sind sowohl für uns als Implementierungsteam als auch für die Endnutzer hauptsächlich Arbeitszeit.

Beim Endnutzer fällt dabei die Zeit an, die für die Nutzung unserer App anfällt. Dies beläuft sich allerdings auf höchstens 5-10 Minuten pro Tag. Auf Seiten des Implementierungsteams, also auf Seiten von Mission Smile, fällt wiederum die Zeit an, die zur ersten Implementierung unseres Produktes benötigt wird. Diese wird auf einige Wochen geschätzt. Zusätzlich fallen für Mission Smile Kosten für das Hosting der Webseite und gegebenenfalls in Anspruchnahme weiterer Dienste an.

Des Weiteren entstehen Kosten für die Umsetzungskontrolle und Fehlerreduktion, um unsere App stetig weiter zu verbessern als auch für Individuelle Anpassungen an den Kunden, da auch diese Veränderungen vor allem Arbeitszeit kosten.

Zuletzt fällt sowohl für den Kunden als auch für das Team Arbeitszeit an, die für Feedbackgespräche zu unserem Produkt genutzt werden, aus denen dann mögliche Verbesserungen abgeleitet werden, um unsere App stetig zu verbessern.

Der Nutzen, den unser Produkt generiert, liegt vor allem im Bereich einer Verbesserung der Motivation der Mitarbeiter unseres Kunden. Das führt zu steigender Leistungsbereitschaft und auch zu einer Verbesserung der Arbeitsabläufe im Unternehmen da bekannt ist, dass zufriedene Mitarbeiter bessere Arbeit leisten und somit mehr Gewinn für das Unternehmen generieren können. Zudem fördert das Produkt die Kommunikation zwischen Mitarbeitern und

Managern. Bei konstant schlechter Stimmung im Unternehmen kann so eine Lösung für die möglichen Problemstellen gefunden werden, da die Mitarbeiter so aufgrund der anonymen Bewertung keine negativen Konsequenzen erwarten müssen.

Zudem kann durch diese Gespräche auch die Beschwerdequote im Unternehmen verringert werden, da, falls ein Problem bekannt wird, aufgrund der kontinuierlichen Umfragen schneller entgegengewirkt werden kann.

Zusätzlich kann sich durch die Nutzung unserer Dienstleistung auch das Unternehmensimage verbessern. Glückliche Mitarbeiter würden ihren Arbeitsplatz eher weiterempfehlen, sodass sich zukünftig mehr Personen dort bewerben. Außerdem kann der Umsatz gesteigert werden, sodass Kunden das Unternehmen besser wahrnehmen.

3.2. SWOT-Analyse

Betrachtet man die Stärken (Strengths) der Anwendung, ist in erster Linie die Mitarbeiterzufriedenheit und die daraus resultierende Produktivitäts- sowie Gewinnsteigerung zu betrachten. Außerdem sinkt die Fluktuationsrate, da sich Mitarbeiter im Unternehmen wohl fühlen. Das führt auch zu einer steigenden Leistungsbereitschaft, sowie einer Verbesserung der Arbeitsabläufe. Zusätzlich wird durch die Anwendung die Kommunikation zwischen Mitarbeitern und Management gefördert ².

Die Chancen (Opportunities) der Anwendung sind, dass sie durch das Feedback der Nutzer ständig weiterentwickelt werden kann. Das könnte geschehen, indem das Feedback der Nutzer umgesetzt wird.

Durch die vermehrte Kommunikation zwischen Management und Mitarbeitern kann im Folgenden die Beschwerdequote verringert werden. Dadurch kann sich dann auch das gesamte Unternehmensimage verbessern.

Die Schwächen (Weaknesses) sind jedoch, dass die Mitarbeiter die Anwendung freiwillig nutzen müssen. Bei geringer Bereitschaft der Mitarbeiter, die Anwendung

² <https://idw-online.de/de/news703745>

zu nutzen, entsteht eine Herausforderung. Außerdem könnten Emotionen vorgetäuscht werden, sodass die Anwendung nicht mehr ihren Zweck erfüllen kann. Die einzige Gefahr (Thread) wäre, dass sich die Mitarbeiter weigern, die Anwendung zu benutzen.

3.3. Auswirkungen auf Marketing/Branding

Da Mission Smile ein Start-Up ist, gibt es bislang noch keine Marketing- oder Branding Strategie. Dadurch besteht die Möglichkeit das Branding von Grund auf neu zu gestalten. Die Marketing- und Branding Strategie des Produkts entspricht der des Unternehmens, da Mission Smile bislang nur die beschriebene Anwendung vertreibt.

Da die Software neuartig ist, es bislang nur wenig Konkurrenz gibt und eine Marktlücke gedeckt wird, wird eine Hochpreisstrategie genutzt. Bei langfristiger Nutzung der Software werden dem Kunden Rabatte angeboten, um mehr Umsätze zu generieren und langfristig zu planen. Der Preis ist abhängig von der Unternehmensgröße der Kunden.

Der Vertrieb der Anwendung findet über unsere Webseite im Internet statt und zusätzlich werden Kunden über Messen und Cold Calling akquiriert. Beworben wird mit Werbeanzeigen auf Seiten oder in Foren für Führungskräfte, sowie mit Werbung über LinkedIn.



Abbildung 5 Beispiel Social Media Post

Bei dem Produkt legen wir hohen Wert auf Nutzerfreundlichkeit. Um eine stärkere Marketingwirkung zu erzielen werden zunächst keine Produktvariationen- oder Differenzierungen angeboten, wobei die Anwendung immer individuell auf den Kunden angepasst wird.

Als primäre Marketingstrategien wird sowohl SEO (Search Engine Optimization) Marketing, als auch E-Mail-Marketing genutzt.

Das SEO-Marketing soll dabei dazu dienen alle potenziellen Kunden auf das Unternehmen aufmerksam zu machen. Dazu werden qualitativ hochwertige Inhalte genutzt und der Anwendungsfall des Produkts wird genau beschrieben. Das Design der Webseite ist ansprechend und es werden interne Verknüpfungen implementiert. Außerdem werden Bild-Tags und Videobeschreibungen genutzt, um die Seite auch für Menschen mit Behinderungen zugänglich zu machen.

Mit Hilfe des E-Mail-Marketings sollen Kunden langfristig an unser Unternehmen gebunden werden.

Hinsichtlich des Branding wird ein Markenprodukt eingesetzt, um dem Kunden Qualität zu vermitteln. Dadurch soll auch der Kaufprozess für den Kunden erleichtert werden. Ansonsten werden grundsätzlich freundliche, helle Farben sowie der lächelnde Smiley als primäres Symbol verwendet.

3.4. Einfluss auf die Supply Chain und Bestandsanalyse

Da es sich um eine Dienstleistung und kein greifbares Produkt handelt, sehen wir keine Auswirkungen auf die Supply Chain sowie keine beträchtlichen Auswirkungen auf den Bestand. Die einzige Komponente, die womöglich durch die App beeinflusst wird, sind die Mitarbeiter unseres Kunden, da diese, wenn sie motivierter sind auch bessere Ergebnisse erzielen können und somit zu einer Verbesserung der Supply Chain beitragen können.

3.5. Zielgruppenanalyse

Die Zielgruppe des Produkts werden ausschließlich Geschäftskunden sein. Insbesondere liegt der Fokus auf Führungskräften, da das Produkt für Unternehmen spezialisiert ist und in privatem Bereich keine Anwendung findet. Bei den Berufsfeldern gibt es jedoch keine Einschränkung. Die Unternehmensgröße ist für das Produkt irrelevant, sie hat lediglich Einfluss auf den Preis. Kaufmuster von Unternehmen hinsichtlich ihrer Größe, ihres Standorts und ihrer Unternehmensgröße können zum aktuellen Zeitpunkt der Produktentwicklung nicht festgelegt werden, da noch keine Produkte verkauft wurden. Allerdings gibt es produktseitig keine Einschränkungen bei einer der Kriterien. Die Zielgruppe sollte ernstes Interesse an der Mitarbeiterzufriedenheit haben und diese verbessern und analysieren wollen. Zudem sollten die Werte des Mitarbeiters als wichtigstes Kapital geteilt werden.

Das Kaufverhalten ist abhängig vom Kaufmotiv, dem Beeinflusser, dem Verkaufsort, der Kauffrequenz, dem Wochentag und der Uhrzeit des Kaufes. Das Kaufmotiv sind Motivationen wie die Verbesserung und Feststellung der Mitarbeiterzufriedenheit sowie die Entdeckung, welche Zufriedenheitsverbesserungsmöglichkeiten am besten helfen. Die Kaufentscheidung der Zielgruppe wird beeinflusst durch das Vermögen des Unternehmens und durch die verschiedenen Abteilungsleiter, die ihre Mitarbeiter zufriedener machen wollen. Der Verkaufsort ist hauptsächlich online und es gibt verschiedene Pakete, die gebucht werden können. Die Pakete haben unterschiedlich lange Laufzeiten und eine Testlaufzeit; die Preise sind dem angepasst und es kommen die individuellen Installationskosten dazu. Die Kauffrequenz ist abhängig von dem gewählten Paket und der Zufriedenheit, ob das Paket erneut gebucht wird. Da das Produkt noch nicht verkauft wird, kann noch kein bevorzugter Verkaufstag oder Uhrzeit ermittelt werden.

Der Preis kann je nach Größe des Unternehmens angepasst werden und setzt sich zusammen aus den Kosten für das Produkt und den Installationskosten vor Ort mit, unter anderem Anbringung von Kameras.

4. Systementwurf

4.1. Ereignis-Reaktionsmodell

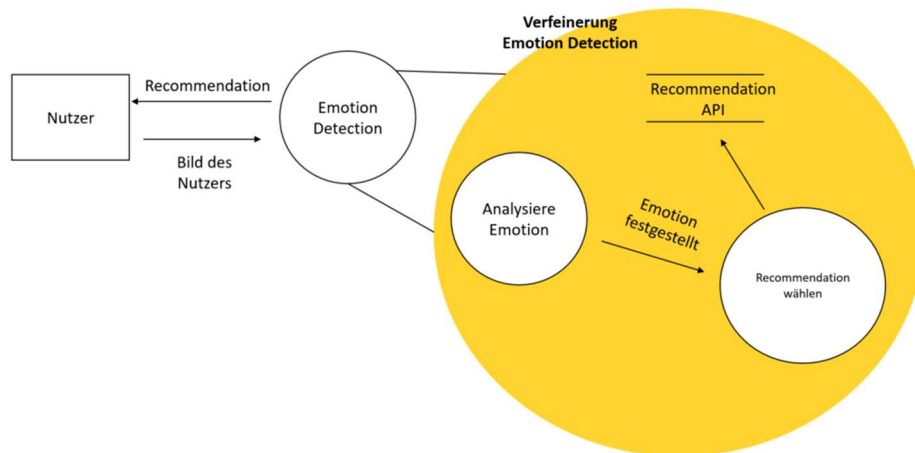


Abbildung 6 Ereignis-Reaktionsmodell

4.2. Aktivitätsdiagramm

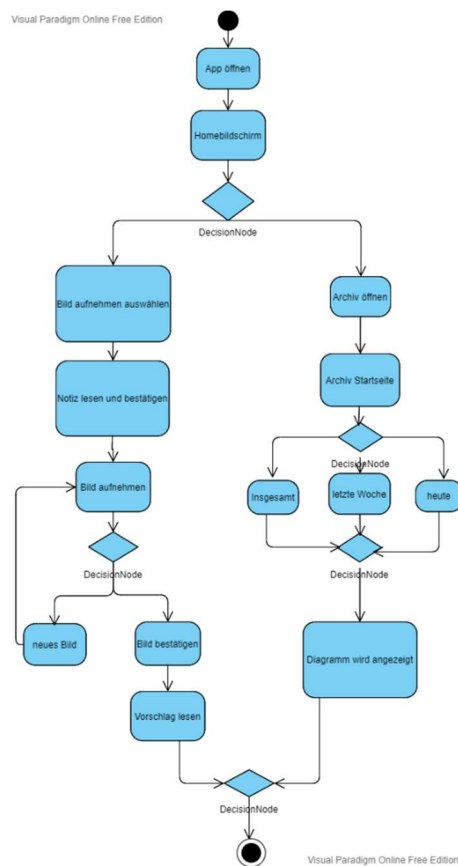


Abbildung 7 Aktivitätsdiagramm ³

³ Startseite = Analytic Seite, Zeitraum nicht änderbar → Ausblick

4.3. Entscheidungstabelle

Bedingung	Aktion	Prozente für Stimmungsbarometer
Emotion: happy	Recommendation aus Liste "happy"	100 %
Emotion: sad	Recommendation aus Liste "sad"	0 %
Emotion: angry	Recommendation aus Liste "angry"	25 %
Emotion: neutral	Recommendation aus Liste "neutral"	50 %
Emotion: fear	Recommendation aus Liste "fear"	0 %
Emotion: surprise	Recommendation aus Liste "surprise"	75 %
Emotion: disgust	Recommendation aus Liste "disgust"	25 %

4.4. Beschreibung der verwendeten Algorithmen

Der Kern der Anwendung ist der FER (Facial Expression Recognition) Algorithmus. Er analysiert, welche Emotionen der Mensch in dem Bild zeigt. Grundlegend ist hier die Verwendung von Feature Extraction. Die eingegebenen Bilder müssen zunächst in ihrer Dimensionalität reduziert werden, um später effizient von dem verwendeten Machine Learning Algorithmus analysiert werden zu können. Dabei ist es für den

Algorithmus besonders wichtig Nase, Mund, Augen und Augenbrauen, sowie den Abstand dazwischen zu erkennen. Deshalb können die Bilder auf den Bereich von Kinn bis Stirn zugeschnitten werden und in ihrer Farbigkeit reduziert werden. Der Bereich über der Stirn und unter dem Kinn, sowie die Farbigkeit sind nicht relevant für den Algorithmus und damit redundant. Die vorverarbeiteten Bilder können dann mit Hilfe von Deep Learning verarbeitet werden. Hier wird ein neuronales Netz verwendet, um die Bilder zu analysieren und damit die Emotionen zu erkennen. Über eine Schnittstelle werden die erkannten Emotionen dann wieder zurück an die Anwendung gegeben. Bei der Schnittstelle handelt es sich um eine REST API, die mithilfe eines POST Requests die Bilder an die Anwendung sendet.

4.5. Use-Case Schablone

Um die Funktionsweise unserer Anwendung nochmals zu verdeutlichen, wurden zwei Use Cases erstellt.

Der erste der beiden Use Cases, ist Use Case #001 Foto aufnehmen, dessen Ziel es ist die Emotion des Kunden herauszufinden und einen personalisierten Vorschlag zu erstellen. Dafür muss die Anwendung eine Berechtigung haben auf die Kamera, bzw. auf das Foto zuzugreifen, sowie der Kunde ein angemeldetes Konto besitzen muss. Bei erfolgreicher Emotionen Erkennung erhält der Mitarbeiter einen Vorschlag zur Verbesserung seiner Stimmung. Wenn die Prüfung, ob der Kunde registriert ist, gescheitert ist, wird der Kunde gebeten sich zu registrieren, sowie Kamera - Berechtigungen an die App zu vergeben.

1. **Use Case:** Foto aufnehmen #001
2. **Ziel:** Emotion des Kunden wird analysiert und ein personalisierter Vorschlag generiert
3. **Vorbedingung:** App Berechtigung auf Kamera, Kunde muss Konto haben + angemeldet sein
4. **Nachbedingung Erfolg:** Kunde erhält personalisierten Vorschlag
5. **Nachbedingung Fehlschlag:** Fehlermeldung + Kunde muss noch ein Foto aufnehmen
6. **Akteure:** Kunde
7. **Auslösendes Ereignis:** Kunde drückt Button „Foto aufnehmen“
8. **Beschreibung:**

- 8.1. Prüfung, ob Kunde bereits registriert
- 8.2. Prüfung Berechtigung Kamera
- 8.3. Foto aufnehmen
- 8.4. Emotionsanalyse
- 8.5. Vorschlag Generierung

Erweiterungen:

1a Wenn nicht registriert → Neuregistrierung

1b Wenn keine Kamera Berechtigung → Berechtigung erteilen

Der zweite Use Case, der generiert wurde, ist Use Case #002 Analyse öffnen, der dazu da ist, um seine bisherigen Emotionsanalysen auf einen Blick zu sehen, indem er auf die Analyse-Fläche drückt. Dies funktioniert wie folgt: Der beim Login Prozess erstellte Token wird von Firebase verwendet, um die Nutzerdaten abzufragen. Vorbedingung hierfür ist allerdings, dass der Kunde bereits Fotos aufgenommen hat.

- 1. **Use Case:** Analyse öffnen #002
- 2. **Ziel:** Kunde kann Stimmungsbarometer der letzten Zeit anschauen
- 3. **Vorbedingung:** Kunde muss schon Bilder gemacht haben
- 4. **Nachbedingung Erfolg:** Kunde sieht Stimmungsbarometer
- 5. **Nachbedingung Fehlschlag:** Kunde sieht keine Stimmungsbarometer
- 6. **Akteure:** Kunde
- 7. **Auslösendes Ereignis:** Kunde drückt auf die Analyse-Fläche
- 8. **Beschreibung:**
 - 8.1 Prüfung, ob Kunde bereits registriert
 - 8.2 Analyse-Fläche drücken
 - 8.3 Zeitraum auswählen
 - 8.4 Stimmungsbarometer erscheint

Erweiterungen:

1a Wenn nicht registriert → Neuregistrierung

3a Wenn in dem Zeitraum keine Bilder vorhanden → Aufforderung zum regelmäßigen Fotografieren

5. Feedback und Zwischenstände

Während der regelmäßigen Sprints wurden folgende Zwischenstände festgehalten. Diese wurden meist unterteilt in die Bereiche Design, Aufgaben und Feedback.

5.1. Zwischenstand Termin 07.01.2022

Zwischenstand der Aufgaben:

Für die Verteilung der Aufgaben wurde Trello verwendet, um einen Product Backlog zu erstellen. Dieser ist in die Kategorien Commitment, Work in Progress, Under Review und Done aufgeteilt. Wie in Abbildung 7 zu sehen ist, stehen wir zum Zeitpunkt dieses Zwischenstandes noch am Anfang unseres Projektes, weshalb hier viele Aufgaben noch in den Bereichen Commitment und Work in Progress zu finden sind.

Feedback:

Am Ende des ersten Sprints wurde zudem ein erstes internes Feedback erstellt, bei dem alle Teammitglieder ihre Stimmung in Bezug auf das Projekt und Verbesserungsvorschläge mitteilen konnten. Hier ist deutlich zu sehen, dass das Team sehr zufrieden mit der Stimmung im Team und der bisherigen Arbeit ist. Es gab noch keine Verbesserungsvorschläge.



Abbildung 8 1. Feedback Scales

Was lief besonders gut?/Was hätte besser laufen können?

Mentimeter

+ Jeder hat sich an die Fristen gehalten	Aufgabenteilung	Rückmeldungen
Das Aufteilen der Aufgaben lief ziemlich gut :)	positiv :)	Arbeitsverteilung über Trello lief besonders gut Bearbeitung der einzelnen Themen lief sehr gut
Alles gut	Stimmung im Team ist gut; alle sehr motiviert	

Abbildung 9 1. Feedback Open

5.2. Zwischenstand Termin 13.01.2022

Zwischenstand Design:

Zum Zeitpunkt dieses Zwischenstandes ist deutlich zu erkennen, dass sich beim Design viel verändert hat. Zur Erstellung dieser Designvorlage für Mobile Endnutzer wurde das Programm Invisio genutzt.

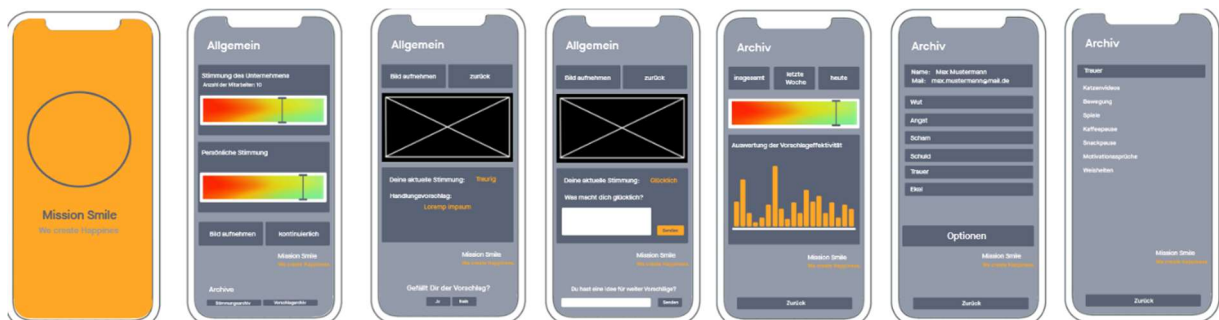


Abbildung 10 Design 2.0

Zwischenstand der Aufgaben:

Im Vergleich zu letztem Mal hat sich in den Bereichen “Under review” und “Done” nicht viel verändert, allerdings kann man deutlich sehen, dass einige Karten in den Bereich “Work in progress” aus dem Bereich “Commitment” verschoben wurden.

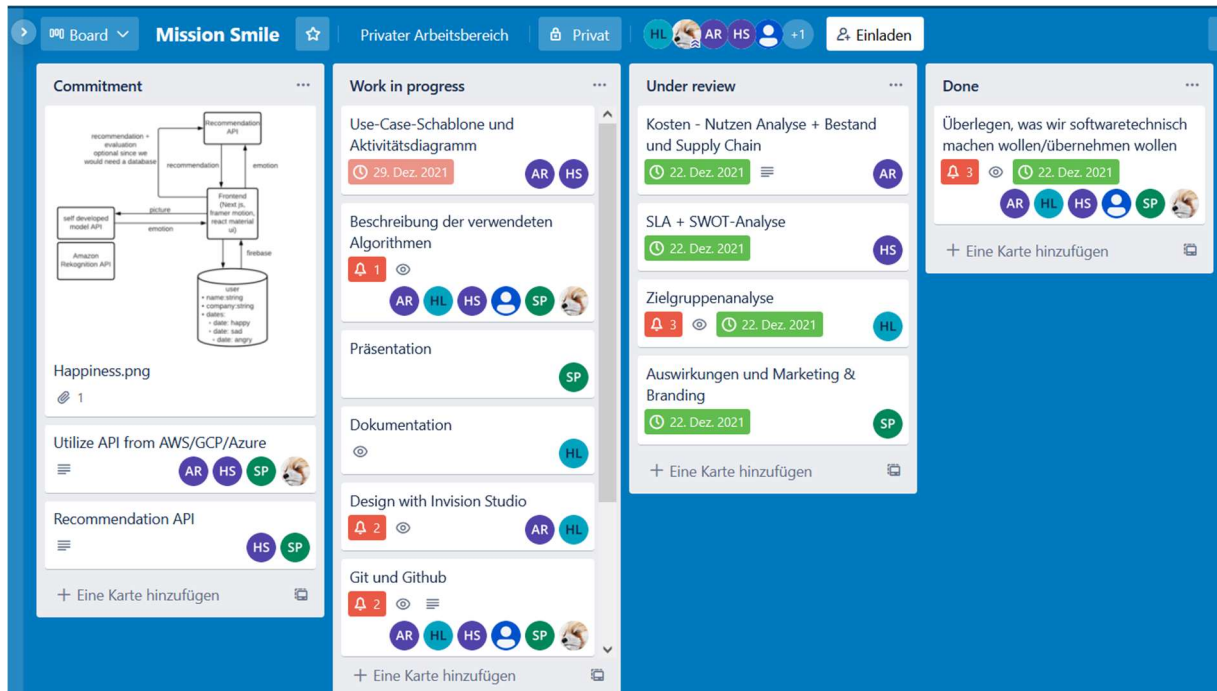


Abbildung 11 Trello Board

Feedback:

Beim nächsten Feedback ist auffällig, dass sich die Stimmung im Team als auch die Zufriedenheit mit der gelieferten Arbeit nicht verändert hat. Die Auslastung im Team wurde als Kategorie hinzugefügt. 1 heißt hierbei, dass das Teammitglied unterfordert ist, 3 bedeutet, dass die Auslastung stimmt, und 5 bedeutet, dass ein Teammitglied überfordert ist. Bei einem Wert von 3,2 ist davon auszugehen, dass die Auslastung im Team gut ist. Auch diesmal gibt es keine weiteren Verbesserungsvorschläge.

6. Applikation

6.1. Grundkonzepte

Grundsätzlich wurde bei der Entwicklung mit Microservices anstatt von einer monolithischen Anwendung gearbeitet. Hier besteht die Anwendung aus unabhängigen Prozessen, die miteinander über eine REST API und über Aufrufe über das Netzwerk kommunizieren. Die Verwendung von mehreren kleineren, unabhängigen Prozessen bietet vor allem für agile Projekte einige Vorteile.

Ein wichtiger Punkt ist, dass die Zusammenarbeit im Team besser funktioniert, da leichter einzelne, kleinere Aufgaben abgeleitet werden können. Dadurch kann die Anwendung von mehreren Entwicklern zeitgleich unabhängig bearbeitet werden, es wird keine ständige Abstimmung benötigt.

Auch das Bereitstellen der Anwendung ist dadurch einfacher, da die einzelnen Teile aufgrund ihrer geringeren Größe und da sie deshalb weniger Tests benötigen einfacher zu testen sind. Dadurch ist auch der Deployment-Prozess kürzer, sodass eine CI/CD Pipeline implementiert werden kann. Außerdem kann bei Fehlern leichter auf ältere Versionen zurückgegriffen werden und es können schneller Änderungen vorgenommen werden.

Zudem bieten Microservices den Vorteil, dass die Programmiersprache frei wählbar ist. So binden wir uns nicht langfristig an einen Technology-Stack und können bei Bedarf schnell zu einem besseren, schnelleren Technologie-Stack wechseln. Dadurch ist es möglich, die Emotionserkennung und die dann ausgegebene Empfehlung in Python zu implementieren und das Frontend mit JavaScript umzusetzen. Python bietet hier im Vergleich zu R den Vorteil, dass viele Pakete im Bereich Machine Learning in Python entwickelt wurden, auf die zurückgegriffen werden kann.

Zudem ist Anwendung durch ihre Unabhängigkeit insgesamt robuster, was Systemausfällen entgegenwirkt. Falls ein Service ausfällt, können so andere Services weiterlaufen, sodass nicht die gesamte Anwendung ausfällt. Der einzelne Service kann dann auch leichter wieder gestartet werden, als wenn das gesamte System neu gestartet werden müsste. Auch die Wartung ist einfacher, da einzelne Komponenten separat betrachtet werden können.

Im späteren Verlauf bietet sich durch die Verwendung der Microservices auch ein Vorteil beim Updaten der Anwendung, da einzelne Teile mit wenig Aufwand ersetzt oder verändert werden können. Es muss nicht die ganze Anwendung angepasst werden.

Herausforderungen bei der Verwendung von Microservices sind grundsätzlich, dass sie komplexer sind. Im beschriebenen Fall überwiegen die Vorteile der Microservices jedoch deutlich. Vor allem die Skalierbarkeit von Microservices bietet einen wichtigen Vorteil. Es kann je nach Bedarf angepasst werden, welche Microservices in welcher Menge genutzt werden, sodass keine übermäßigen Kosten entstehen.

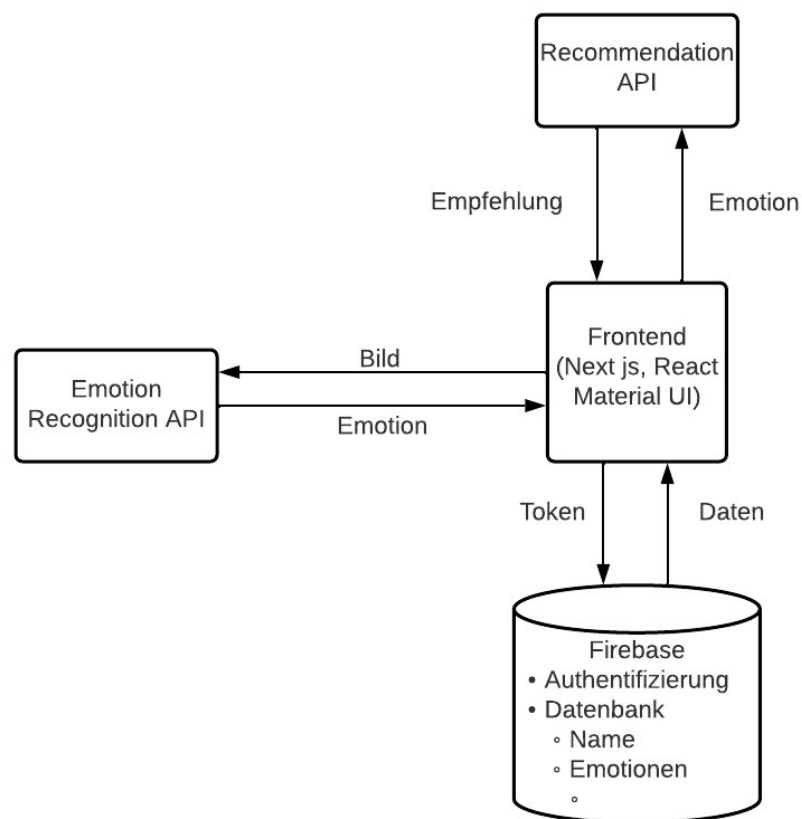


Abbildung 12 Architektur

Unsere Architektur lässt sich in vier Komponenten aufteilen:

- **Frontend:** Oberfläche der Anwendung, mit der der Nutzer agiert
- **Datenbank:** Daten über die Emotionen des Nutzers werden bei Firebase abgespeichert

- **Recommendation-API:** Erhält eine Motion und gibt eine entsprechende Handlungsempfehlung zurück
- **Emotion-API:** Bekommt das Bild des Nutzers und gibt die erkannte Emotion zurück

Frontend

Das Frontend wurde mit Next.js entwickelt, da die Anwendung durch server-side-rendering für Suchmaschinen optimiert werden kann. Statische Dateien wie HTML können auf verteilten CDN-Servern abgespeichert werden, sodass der Webcrawler die Struktur und den Inhalt der Webseite aufrufen kann. Wenn der Nutzer die Webseite öffnet, werden dynamische Inhalte zusätzlich noch heruntergeladen, um die Webseite interaktiv zu gestalten. Dieser Prozess wird auch Rehydration genannt.

Bei einer normalen React-Webseite dagegen gibt es keine klare Trennung zwischen dynamischen und statischen Inhalten. Der Crawler kann somit nicht unterscheiden, welche Inhalte wichtig sind.

Auch die Performance und Skalierbarkeit kann verbessert werden. Des Weiteren basiert Next.js auf React, was in unserem Fall praktisch ist, da alle Teammitglieder React bereits kennen und sich nicht mit einem neuen Konzept auseinandersetzen müssen.

Außerdem wurde für das Frontend die Frontend-UI-Library React Material (MUI) verwendet. Dadurch konnte die Entwicklung verkürzt werden, da auf Elemente aus der Library zurückgegriffen werden konnte. Außerdem wurde dadurch die Nutzerfreundlichkeit erhöht, da die Verwendung von MUI gebräuchlich ist und deshalb bereits bekannt ist. Das führt dazu, dass sich Interessenten schneller zurechtfinden können und ein besseres Nutzererlebnis haben.

Datenbank

Für die Authentifizierung und Datenbank wurde Firebase genutzt. Firebase wird als Service angeboten und bietet dadurch sowohl eine gute als auch kostengünstige Skalierbarkeit. Außerdem ist hier die Entwicklungszeit besonders kurz und die Performance und Geschwindigkeit hoch.

Ein weiterer Vorteil von Firebase ist die Verwendung von nicht-relationalen Datenbanken. Nicht-relationale Datenbanken haben den Vorteil, dass keine Daten-Schemata festgelegt werden müssen. Dadurch vereinfacht die Datenbank das Aufnehmen von neuen Datensätzen. Aus diesem Grund eignet sie sich für unser Projekt, da die Datenbank flexibel an unseren dynamischen Anforderungen anpassen kann.

Die genutzte Version von Firebase ist die Version neun. Hier stellt sich die Herausforderung, dass diese Version noch neu ist, da sie vor einem halben Jahr veröffentlicht wurde, sodass es bislang nur wenige Anleitungen und Kurse dazu gibt. Trotzdem wird in der Anwendung Version neun genutzt, da in dieser Version der Ansatz der funktionalen Programmierung verfolgt wird. Dadurch gibt es kleinere Packages, da immer nur die Funktionen importiert werden müssen, die benötigt werden. Bei anderen Versionen müsste stattdessen alles importiert werden und es könnte erst später ausgewählt werden, welche Funktionen wirklich benötigt werden. Werden nur die benötigten Funktionen importiert, ist die Anwendung schneller, da wenige Daten heruntergeladen werden müssen.

Recommendation-API

Für die Empfehlungsanwendung wurde Flask als Framework verwendet, da dieses eine ausführliche Dokumentation anbietet. Darüber hinaus ist das Framework sehr schlank und vereinfacht somit den Deployment-Prozess. Ein Nachteil von Flask ist jedoch die geringe Anzahl von Funktionalitäten. Da unsere Anforderungen für den Anwendungszweck jedoch sehr gering sind, ist der Nachteil kaum zu berücksichtigen.

Emotion-API

Für die Emotionserkennung wurde ebenfalls das Framework Flask verwendet, da dieses in Python entwickelt wurde. Dadurch basiert auch das verwendete ML-Modell auf Python und lässt sich deshalb gut in die Anwendung einbinden.

Deployment

Das Frontend wurde auf Vercel deployt, da Vercel CDN-Server anbietet. Durch diese wird der Ladeprozess beschleunigt und somit wird die Nutzererfahrung verbessert. Für die anderen beiden API wurde Heroku verwendet, da diese ein kostenloses Kontingenz für das Hosten von Anwendungen anbietet. Der Nachteil ist

jedoch, dass die Anwendungen in den USA gehostet werden. Die Reaktionszeit der Anwendung ist deshalb langsamer. Des Weiteren werden die Anwendungen nach 30 Minuten inaktiv, wenn diese nicht genutzt werden. Bei Reaktivierung dauert es dann etwa eine Minute bis die Anwendung wieder aktiv ist.

6.2. Technische Umsetzung

Frontend

Unser Code setzt sich aus verschiedenen Teilen zusammen, die mit Hilfe von Ordnern strukturiert wurden. Dadurch bleibt das Projekt übersichtlich und es kann sich schnell darin zurechtgefunden werden. In dem Ordner "src" liegt der komplette Code der Anwendung.

Die Komponenten appBar, authRoute, emotionmeter, layout und pricingCard liegen in dem Ordner "components". Die verwendeten Emotionen werden im Ordner „constants“ global definiert. Im Ordner „contexts“ befinden sich Dateien, in denen Funktionen und Variablen global deklariert werden. Diese Funktionen und Variablen können anschließend an andere Seiten und Komponenten übergeben werden. In der AuthContext Datei befinden sich Funktionen, die für die Authentifizierung des Nutzers über Firebase benötigt werden. Beispielsweise sind Funktionen wie signup(), login() und signout() dort definiert.

Der darin liegende Ordner „Pages“ beinhaltet den Code für die einzelnen Seiten unserer Anwendung. Hier ist vor allem die Analytics.js- und die Index.js-Datei relevant. In der Analytics Datei werden alle erfassten Emotionen erfasst und statistisch ausgewertet. In der Index Datei ist die Startseite definiert. Hier können Bilder hochgeladen werden und die dazu passende Emotion und Empfehlung wird ausgegeben. Alle anderen Dateien haben einen semantisch passenden Namen. Beispielsweise enthält die Team.js-Datei die Team-Seite und die Pricing.js-Datei die Preis-Seite.

Die verwendeten Bilder werden im Ordern "public" abgespeichert. Hier sind sowohl die Bilder der Teammitglieder als auch Bilder, die im Hintergrund der App verwendet werden, abgespeichert.

Die CSS-Dateien der Anwendung befinden sich in dem Ordner "styles", jedoch wird das Layout häufig bereits in den jeweiligen js-Dateien der Seite definiert, da mit MUI-

Komponenten gearbeitet wird. Das führt dazu, dass keine separaten CSS-Dateien verwendet werden müssen. So bleibt der Code der Anwendung übersichtlicher und die Implementierung ist einfacher. Die benötigten Komponenten werden jeweils aus der Bibliothek importiert und können bei Bedarf angepasst werden. Zudem liegt hier auch die "mytheme.js" Datei, die in der App.js Datei aufgerufen wird und damit die verwendeten Farben an alle Seiten übergibt.

Recommendation-API

Die App.py erhält den ganzen Quellcode. Alle anderen Dateien wie bspw. Procfile sind lediglich für das Deployment wichtig. In App.py wird eine Schnittstelle definiert, die über eine GET-Anfrage erreicht werden kann. Der Nutzer schickt seine Emotion mit einer GET-Anfrage an die Anwendung und bekommt anschließend eine Handlungsempfehlung zurück.

Emotion-API

Bei der Anwendung ist ebenfalls lediglich die App.py wichtig. Die Anwendung bietet ebenfalls eine Schnittstelle, die durch eine GET-Anfrage erreicht werden kann. Zunächst wird das Nutzer-Bild mit Base64 kodiert und anschließend an die Anwendung geschickt. Die Emotion-API setzt das kodierte Bild mit Base64 wieder zusammen und gibt dieses anschließend an das FER-Modell weiter. Das Modell evaluiert das Bild und gibt anschließend eine Emotion mit einer Genauigkeit-Wahrscheinlichkeit zurück. Zum Schluss werden die Emotion sowie die Wahrscheinlichkeit als Antwort der GET-Anfrage zurückgegeben.

6.3. Probleme und deren Lösungen

1. Seiten verstecken, die der Nutzer erst nach dem Einloggen öffnen kann

Dieses Problem war sehr zeitintensiv. Es wurden mehrere Ansätze verfolgt bis schließlich folgende Lösung umgesetzt wurde: Es wird eine Variable im UseContext benötigt. Diese Variable bekommt bei Login einen Wert. Bei jeder „versteckten Seite“ wird überprüft, ob der Variable ein Wert zugeteilt wurde.

2. Das Verbinden von den Backend-Microservices mit der React-Anwendung.

Da die Microservices über das Netzwerk kommunizieren, müssen die Schnittstellen sowie die Eingänge wie bspw. Ports im Netzwerk festgelegt werden. Dieser Prozess

dauert auf dem lokalen Rechner sehr lange. Des Weiteren können sich die Schnittstellen verändern, nachdem die Anwendungen auf Heroku deployt wurden. Alle diese Schritte müssen bei jeder Änderung berücksichtigt werden.

Darüber hinaus stellt die Kommunikation durch das Netzwerk weitere Herausforderungen dar. Ein gutes Beispiel dafür ist das Verschicken des Nutzerbildes an die Emotionserkennung-API. Das Bild muss mit Base64 zu einem String codiert werden, damit das Bild übertragen werden kann. Durch das Deployment mussten die Schnittstellen nochmal geändert werden, damit die Komponenten wissen, wie sie darauf zugreifen können (anstelle eines lokalen Links bekommen die Schnittstellen nun einen richtigen Link).

3. Erlernen von Technologien

Das war eins der größten und zeitaufwändigsten Herausforderungen. Es wurden viele Technologien verwendet, mit denen das Team teilweise keine oder nur wenig Erfahrung hatte. Deshalb musste viel Wissen angeeignet werden, bevor die praktische Umsetzung beginnen konnte.

Beispielsweise war das Deployment, besonders das Schreiben der Dockerfile-Datei und die gesamte Dauer neu zu erlernen. Auch der Umgang mit Firebase war für manche neu zu erlernen.

4. Firebase v9

Die Version gibt es erst seit einem halben Jahr und deshalb gibt es bisher nur wenige Tutorials darüber. Wir haben uns trotzdem für v9 entschieden, da v9 den Ansatz der funktionalen Programmierung verfolgt und dies den Vorteil von kleineren Packages und einer besseren Performance mit sich bringt. Näheres dazu im Kapitel 6.1. Grundkonzepte.

5. Richtige Emotion Detection

Die Wahl der richtigen Emotion Detection war ebenso eine Herausforderung. Es wurden verschiedene Modelle getestet, aber viele hatten entweder eine schlechte Güte oder konnten aufgrund von anderen Problemen nicht richtig laufen.

6. Zeit

Durch Probleme wie beispielsweise Punkt 2 und Punkt 3 ist auch die Zeit etwas knapp geworden, da die Entwicklungszeit auch auf die Prüfungsphase und am Ende auch auf die Praxisphase fiel und deshalb weniger Zeit für das Projekt blieb.

7. Kritische Reflektion

Im Folgenden wird das Projekt "Mission Smile" hinsichtlich Vorgehensweise, technischer Aspekte und Zusammenarbeit kritisch gewürdigt. Am Ende wird ein Ausblick in die Zukunft gegeben.

Hinsichtlich des Themas lässt sich sagen, dass wir ein vom Dozenten vorgeschlagenes Thema – die Emotionserkennung – aufgegriffen haben und dieses weiter ausgearbeitet haben. So ist die Anwendung nicht nur eine Emotionserkennung, sondern sie verarbeitet die gewonnenen Informationen auch weiter, um so die Mitarbeiterzufriedenheit in einem Unternehmen zu verbessern.

Die Vorgehensweise wurde im Abschnitt "2. Projekt-Vorgehen" bereits beschrieben. Insgesamt ist das Vorgehen gut verlaufen, schließlich konnten wir unsere Ziele alle erfüllen. Allerdings hätten wir früher mit der Umsetzungsphase beginnen können - also schon im ersten Sprint. Dann hätten wir einen größeren zeitlichen Puffer am Ende gehabt.

Zu den technischen Aspekten wurden die aufgetretenen Probleme im Abschnitt "6.2 Probleme und deren Lösungen" beschrieben. Zusammenfassend wurden alle Probleme gelöst.

Das Team ist mit der Funktionalität der Anwendung und ihrer technischen Umsetzung sehr zufrieden. Alle Punkte, welche am Anfang konzipiert wurden, konnten umgesetzt werden. So funktioniert das Login, die Emotion Detection, die Recommendation Rückgabe sowie das Stimmungsbarometer.

Innerhalb der Dokumentation wurden nur die Hauptfunktionalitäten erklärt, damit die Seitenanzahl-Vorgabe nicht überschritten wurde.

Kleinere Punkte konnten aus zeitlichen Gründen leider nicht perfekt umgesetzt werden. Beispielsweise sollten die Handlungsempfehlungen ursprünglich personalisierter sein: Der Nutzer sollte eigene Vorschläge einspeisen oder löschen können. Stattdessen wurden die personalisierten Vorschläge nun durch regelmäßige Fragebögen der Mission Smile AG umgesetzt.

Ein weiterer Punkt ist unser Design. Unser Design hat sich im Laufe der Entwicklung etwas von unserer ursprünglichen Vorstellung abgeändert.

Insgesamt haben wir durch das Projekt viele neue Technologien erlernen können. Wie auch schon bei den Problemen beschrieben, musste sich das Team in viele Themenbereiche neu einlesen und diese dann zum ersten Mal selbst programmieren (beispielsweise Firebase, Deployment, ...). Auch die Nutzung von GitHub war für manche Teammitglieder neu und somit haben sie viel für zukünftige Projekte mitnehmen können.

Bei der Zusammenarbeit gab es gegen Ende ein Problem. Es ist aufgefallen, dass ein Teammitglied seine Aufgaben nicht so erfüllt hat, wie bei den Meetings kommuniziert wurde. Das hat für Mehraufwand im letzten Sprint bei den anderen Teammitgliedern geführt. Dies wurde auch bei der Punktevergabe berücksichtigt.

Ansonsten lief die Zusammenarbeit im restlichen Team positiv ab. Jeder wusste, auch durch Trello, über die Aufgaben der anderen Teammitglieder Bescheid. Durch die regelmäßigen Meetings konnte auch immer Unterstützung bei Problemen durch andere Teammitglieder garantiert werden.

Im Folgendem wird ein kleiner Ausblick der Anwendung gegeben, wie es mit dem Projekt weiter gehen kann.

Die Anwendung kann in Zukunft weiterentwickelt werden. Beispielsweise kann die Anwendung auf einem anderen Server deployt werden, damit sie schneller reagiert. Auch an dem Stimmungsbarometer könnten weitere Funktionen zugefügt werden. Zum Beispiel könnte eine Funktion zugefügt werden, die filtert, welcher Zeitraum der Emotionen beachtet werden soll.

Bei den Recommendations könnte man Funktionen ergänzen, die es dem Nutzer erlauben direkt in der Anwendung Recommendations zu ergänzen oder aus der Datenbank zu löschen.

8. Fazit

Das Ergebnis des Projektes ist eine Anwendung, welche anhand der Emotion eines Mitarbeiters eine Handlungsempfehlung zurückgibt, die dem Nutzer hilft, seine Stimmung zu verbessern. Somit wurde eine vollfunktionstüchtige Anwendung entwickelt, welche die am Anfang konzipierten Anforderungen komplett erfüllt.

Das Team arbeitet, bis auf eine Ausnahme, harmonisch und konstruktiv zusammen. Aufgaben wurden gerecht verteilt und an das Vorwissen und an die Fähigkeiten der Teammitglieder angepasst. Durch die Scrum Vorgehensweise waren alle Teammitglieder ständig in Kontakt und Fortschritte sowie Probleme konnten regelmäßig ausgetauscht werden.

Auf technischer Ebene ist das Endprodukt gut ausgefeilt und alle Anforderungen konnten umgesetzt werden. Punkte, welche zu optimieren wären, wurden in der Dokumentation angesprochen, aber hätten den zeitlichen Rahmen des Projekts überschritten.

Zusammenfassend wurde innerhalb der Vorlesungsreihe "Fallstudie" eine funktionierende Anwendung erfolgreich entwickelt.