# Einleitung

In Deutschland liegt die Aktionärsquote bei ca. 6%. Dies ist im internationalen Vergleich z.B. zu den USA (25%) oder Großbritannien (23%) deutlich geringer (<https://de.wikipedia.org/wiki/Aktion%C3%A4rsquote>). Die Gründe dafür sind vielseitig, zum einen gilt Deutschland als Risikoaverse Nation aber auch die stabil funktionierenden Pensionssysteme sorgten dafür, dass Aktien nicht so relevant für die Altersvorsorge sind wie beispielsweise in den USA. Durch die über die Jahre steigende Performance des Dax aber auch die Angst des Einbrechens der Rente überzeugt mittlerweile immer mehr Menschen in Aktien zu Investieren.

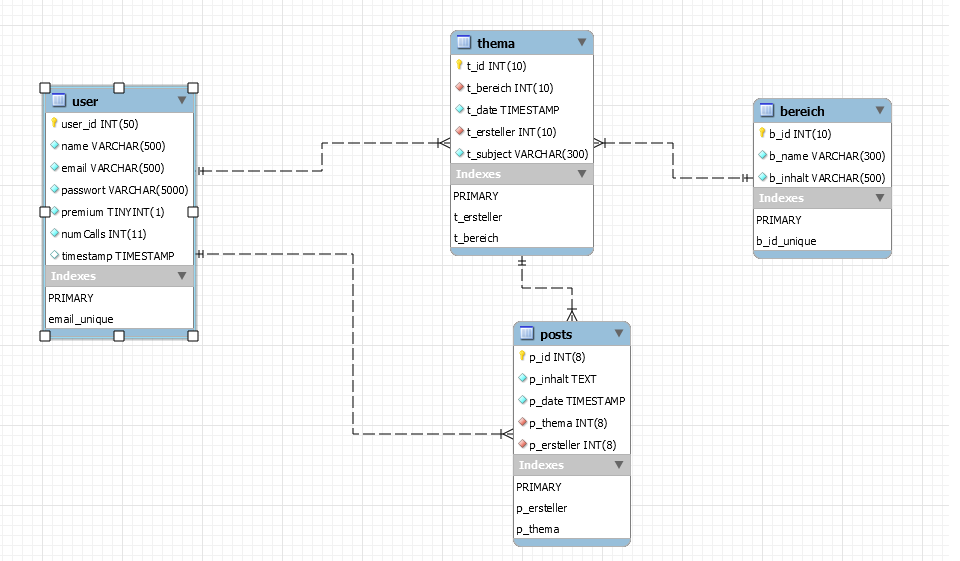
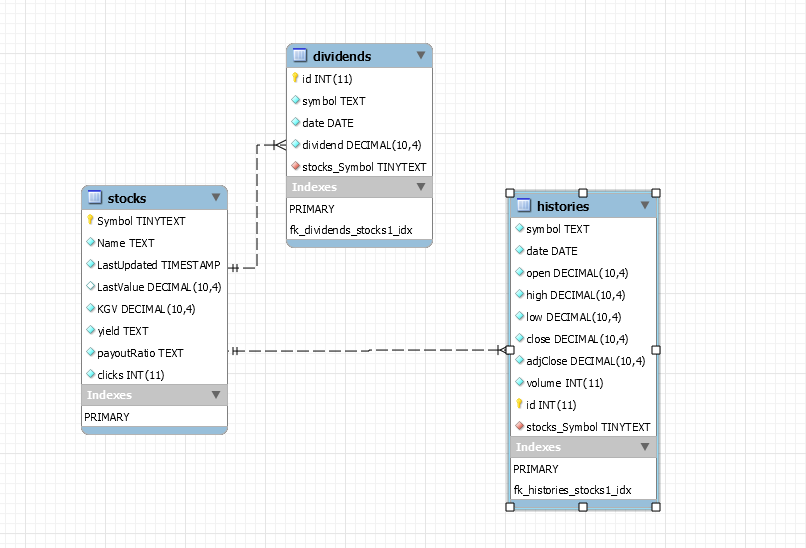
Mit dem Blick auf eine stabile Rente nutzen wir eine dividendenbasierte Strategie mit dem Ziel Aktien zu identifizieren, die besonders stabil und risikoavers ihre Dividenden auszahlen. Mit unserem Tool HYPE nutzen wir die wichtigsten Kennzahlen und bilden diese besonders einsteigerfreundlich mittels „Ampelsystem“ ab. Dabei geben wir noch kleine Hinweise zu den verschiedenen Kennzahlen, sowie die Entscheidungsgrößen, warum die Ampel nun grün, gelb oder rot zeigt. Ziel ist es so mittels risikoaverser Investitionen basierend auf leicht verständlichen Kennzahlen die „Börsenscheu“ in Deutschland zu senken.

Die Webseite stellen wir als Freemium Modell bereit. Bei kostenloser Nutzung sind zwar alle derzeitigen Funktionalitäten gegeben, jedoch ist es nur möglich maximal 5 Aktien am Tag zu betrachten. Bei einem Upgrade auf die Premium Version auf Abobasis können beliebig viele Aktien betrachtet werden und es steht ein persönlicher Support zur Verfügung. (WAS sind wirklichen Unterschiede ergänzen)

QUELLEN ergänzen

# Aufbau der Datenbank

Für unsere Webseite nutzen wir sieben Tabellen, wovon vier für Usermanagement und das Forum genutzt werden. Die restlichen drei Tabellen werden für das Speichern der aktienbezogenen Daten benötigt. Aus diesem Grund sind diese Tabellen nicht direkt mit dem *user* verknüpft. Zur übersichtlichen Visualisierung aller Beziehungen, nutzen wir ein ERR Modell (erstellt mit SQL Workbench). Siehe dazu Abbildung 1 und 2.

(abb .1 sjshdskjds)

# Funktionalitäten

Im Folgenden gehen wir auf die wichtigsten Funktionen unserer Webseite und deren Implementierung genauer ein. Technisch uninteressante Funktionalitäten, wie beispielsweise die Registrierung werden verkürzt abgehandelt. Zur Übersichtlichkeit stellen wir genutzte Tabellen aus der Datenbank im Fließtext kursiv dar (z.B. *user*).

Die Zuständigkeiten der Funktionen nennen wir immer am Ende einer Funktionalität, möchten aber an dieser Stelle hervorheben, dass viel gemeinsam (über github) programmiert wurde und sich viele Funktionen nicht klar trennen lassen.

## Accountmanagement:

Das Accountmanagement umfasst das Registrieren eines neuen Nutzers, sowie die Möglichkeit zur Anmeldung bestehender Nutzer. Außerdem können angemeldete Nutzer ihre Zugangsdaten ändern und die Premiumfunktion (ab-)bestellen. Des Weitern sind die wichtigen Seiten wie die Analysen und das Forum vom Zugriff durch nicht angemeldete Nutzer geschützt.

Mit Hilfe von html Formularen werden mittels POST Daten an Datenbank übermittelt und bei Gültigkeit in *user* eingetragen, bzw. bei der Anmeldung mit *user* abgeglichen. Der Schutz vor unbefugten Zugriff auf oben genannte Seiten ist mittels Sessions und ggf. einer Weiterleitung umgesetzt.

Hauptzuständiger: dlkjgask

## Analyseseite

Innerhalb der Analyseseite sind mehrere Funktionalitäten zusammengefasst, die wir im folgenden separat betrachten, um die Übersichtlichkeit zu wahren.

### Suchleiste

Die Suchleiste bildet den Ausgangspunkt der Aktienanalyse. In der Suchleiste kann nach dem Namen einer Aktie gesucht werden, wobei per Autovervollständigung passende vollständige Namen angezeigt werden. Die Autovervollständigung funktioniert derzeit für alle Aktien des Dax und noch einige bekannte Unternehmen weltweit (z.B. Apple). Es kann jedoch für jede beliebige Aktie die Analyse gestartet werden, wenn das Symbol (nach Yahoo Finance Standard) eingetragen wird.

Zur technischen Umsetzung haben wir das Framework JQueryUI genutzt, welches bereits eine „autococmplete“-Funktion mitliefert. Diese nutzt unser dictionary (stocks.js) in der wir label (Name der Aktie), sowie values (die Symbole) definieren.

Alternativ zur Suchleiste, kann man auch auf die Top-5 gesuchten Aktien zugreifen. Auf eine genauere Erläuterung verzichten wir an dieser Stelle, da eine einfache SQL-Abfrage an *stocks* gesendet wird, welche prüft, welche Aktie die meisten Aufrufe generiert hat.

Hauptzuständiger: dlkjgask

### Generieren der Finanzdaten

Um die Analyseseite für ein übergebenes Symbol zu laden, wird zunächst überprüft, ob die Aktie bereits vor weniger als 24h in *stocks* geladen wurde, da für unseren Zweck eine Aktualität von 1-2 Tage vollkommen ausreicht. Ist die Aktie bereits vorhanden, kann direkt das Frontend geladen werden, wenn nicht müssen die Daten erst von Yahoo Finance abgerufen werden.

Auf technischer Ebene ist diese Funktion in zwei Hauptfunktionen zu unterteilen, den Scraper und die cURL-Abfrage. Zunächst werden im Scraper mittels simple\_html\_dom-Framework 2 verschiedene Yahoo Finance Seiten gescraped. Dabei ist Yahoo jedoch sehr dynamisch. Einerseits kann man gewissen Kennzahlen mittels html-Attribut finden und auslesen. Andererseits muss per Schleifendurchlauf nach bestimmten Strings gesucht werden, denen dann die für uns relevante Kennzahl folgt. Zusätzlich wird eine cURL-Abfrage benötigt, welche per POST-Request die csv-Dateien des Aktien- und Dividendenverlaufs laden. An dieser Stelle ist eine cURL-Abfrage nötig, da Yahoo mit dem Einstellen ihrer API einen crumb (Cookie) setzt, welcher vorhanden sein muss, um csv-Dateien herunter zu laden. Den Crumb kann man jedoch mit Hilfe von cURL umgehen.

Hauptzuständiger: dlkjgask

### Anzeigen von Graphen und bewertenden Ampeln

Auf unserer Seite Analysen können alle Aktien, die bei Yahoo Finance gelistet sind, angezeigt werden. Dabei stellen wir den Aktienkursverlauf und die Dividendenauszahlungshöhe und -rate als Graph dar. Unterhalb dieser zwei Graphen, sind diverse Kennzahlen in einer Tabelle aufgelistet, welche das Interpretieren der Aktie übernehmen. Dabei besteht eine Reihe immer aus dem Namen der Kennzahl mit einem Infobutton für weitere Informationen, den kritischen Werten, wann die Ampel grün, gelb oder rot ist, dem aktuellen Wert für die Aktie und der Ampel selbst.

Um die Daten für eine Aktie auf der Webseite anzuzeigen, nutzen wir neben html CanvasJS (für die Graphen) und JavaScript / JQuery (für allgemeines). Um die Graphen mit CanvasJS anzuzeigen, laden wir die gespeicherten Daten als Array aus der Datenbank, passen noch den Timestamp an und hinterlegen diese x-Werte (Timestamp) und y-Werte (Aktienkurs bzw. Dividendenhöhe) als Datengrundlage. Mittels CanvasJS können diese Daten dann als Line-Chart bzw. als Balkendiagramm dargestellt werden. Für die Tabelle der Kennzahlen überprüfen wir zuerst, ob die Kennzahl der Aktie in einem bestimmten Wertebereich liegt (Hardgecoded für jede Kennzahl). Je nach dem in wo die Kennzahl hineinfällt, setzten wir die Bildquelle unterschiedlich, um eine grüne, gelbe oder rote Ampel anzeigen zu können.

Hauptzuständiger: dlkjgask

## Forum

Das Forum ist einfach und intuitiv gehalten. Man kann Kategorien erstellen, in diesen Themen posten und zu verschiedenen Themen Beiträge verfassen. Dies ermöglicht einen einfachen Wissensaustausch der Community auf einer Plattform.

Auf die technische Umsetzung muss nicht weiter eingegangen werden, da alles über html-Formulare, sowie SQL-Statements umgesetzt ist. Besonderheiten gibt es hier keine.

## Chatbot

Der Chatbot unserer Seite soll die Navigation vereinfachen und die generelle Userexperience verbessern. Bisher hat dieser einige grundlegende Funktionen, wie das anzeigen der meistgesuchten Aktien oder das Navigieren zu den Analysen. Hier ist jedoch herauszustellen, das dies nur eine initiale Beladung von Daten darstellt und in der Zukunft noch ausgebaut werden kann. Für eingeloggte und nicht-eingeloggte Nutzer unterscheidet sich der Chatbot.

Zur Umsetzung nutzen wir das auf JavaScript basierende Framework BotUI. Eine genaue Erklärung des Frameworks und dessen Nutzung würde an dieser stelle zu weit gehen. Grundsätzlich agiert unser (deterministische) Chatbot auf Eingaben des Nutzers. So wird beispielsweise nach dem Namen des Nutzers per input-Feld gefragt und das Ergebnis im Folgenden Begrüßungssatz genutzt. Auf diese Art und Weise funktionieren auch die input Buttons, wobei bei diesen das Ergebnis durch Reihe von if-Abfragen läuft, um zu determinieren was passieren muss. Eine interessante Funktion versteckt sich noch hinter dem Weiterleiten. Dadurch, dass wir beim öffnen des Chatbots das parent Fenster als variable mitgeben, können wir z.B. wenn auf eine der top Aktien gedrückt wurde, diese im Hauptfenster (und nicht innerhalb des Chatbots) öffnen.

Hauptzuständiger: dlkjgask

## Responsive Design

Unsere Webseite ist durchweg responsive, dass bedeutet, dass sie auf Bildschirmen mit kleinerer Auflösung trotzdem noch gut angezeigt werden kann. Dies ist besonders im Zeitalter der hand-held-devices essenziell.

Auf technischer Seite geschieht die Umsetzung des responsive Designs mittels des weit verbreiteten Framework Bootstrap 4.0. Dieses Unterteilt das Layout einer Webseite in beliebig viele Reihen und 12 Spalten. Mit Hilfe von CSS-Klassenbezeichnungen greift man auf dieses Framework zu und bestimmt, ab welcher Bildschirmgröße eine Spalte in eine neue Reihe umgebrochen werden soll. Auch wenn es viele fertige Templates basierend auf Bootstrap gibt, haben wir uns für unser eigenes Layout entschieden.

Hauptzuständiger: dlkjgask

# Vorkenntnisse

Die Vorkenntnisse innerhalb unserer Gruppe sind relativ homogen, wenn auch gewisse Unterschiede bestehen. Yorrick hat mit seinem Bachelor der Informatik das meiste theoretische Wissen in der Gruppe. Zwar war PHP kein Bestandteil seiner Module, jedoch konnte er sich durch das Wissen anderer Programmiersprachen schnell einarbeiten. Dadurch war er innerhalb der Gruppe schnell für das Backend zuständig. Hagen hat als theoretischen Hintergrund das wirtschaftswissenschaftliche Studium, in dem er neben einführenden den einführenden Modulen „Einführung in die EDV“ und „E-Business“ hat Hagen auch bereits das Modul „Computational Intelligence“ und „Intelligent Information Systems“ gehört. Durch sein Wissen besonders durch bereits absolvierte Uni-Projekte, ist Hagen der Allrounder der Gruppe, der meist, wenn auch nicht ausschließlich die Schnittstelle zwischen Front- und Backend bildet. Das Frontend wiederum, das meist aus JavaScript, JQuery und weiterer Frameworks bestand, wurde hauptsächlich von Patrick übernommen. Neben den gleichen Modulen die auch Hagen in der Universität gehört hat, arbeitet er als Werkstudent im Bereich des Webdevelopments. Dadurch, dass er dort oft mit JQuery, Bootstrap und Co arbeitet, hat es sich angeboten, dass er dies auch innerhalb des Projekts übernimmt.

# Ausblick

Die Grundfunktionalitäten unserer Webseite sind implementiert und funktionieren. Darauf aufbauend kann man besonders für Premiumnutzer den Funktionsumfang erweitern. Besonders das direkte Vergleichen von Aktien auf einer Seite bietet sich hierbei an. Des Weiteren kann man ein Musterdepot implementieren, damit z.B. junge Nutzer mit Aktien arbeiten können, ohne echtes Geld einsetzen zu müssen. Außerdem kann man sich überlegen das Konzept des einfachen Darstellens der Dividendenstrategie auf andere Strategien zu übertragen. Diese sollten dann jedoch stark getrennt von der Dividenden Analyse stattfinden, um die Einfachheit und Übersichtlichkeit bei zu behalten.