Final Document: StockCompete

Aaron Nolan - 18423054

Ben Strickland - 18704489

1. Organisation / Motivation / Executive Summary

1.1 Executive Summary

1.1.1 Company Description Summary

We are StockCompete, a business originating in Dublin, Ireland that specialises in stock trading, specifically in the EU market. Our goal is to make stock trading as easy and accessible as possible to traders of all skill levels.

We provide an alternative to contemporary stock trading platforms by providing simplified trading information for non-professional traders to easily learn the trading process, and a gamified UI including a leaderboard and points system to encourage frequent participation from users, which will in turn teach them to become better at trading. Our initial customer segment will be non-professional traders based within the EU market. From our conducted research, it is clear that there is strong demand within the 18-24 age range specifically, however this service will still be available to all over the age of 18. Over time we hope to expand this service to work in all currencies and all markets outside the EU.

The main direct and indirect competitors we will compete with are Robinhood and eToro. In both cases we have competitive advantages, such as using a pricing model that isn't banned in certain countries, offering low commission fees to the user with no hidden fees attached and using gamification to make trading an enjoyable experience.

1.1.2 The Problem

Beginning to trade can be a daunting task for those with little experience in the area. Being overwhelmed with in-depth analysis graphs, technical terminology, the initial cost of starting and the time that needs to be spent on it are the immediate reasons people don't begin trading.

1.1.3 Analysis of the Problem

After investigating the problem further, we researched various methods of mitigating or resolving these problems to make trading more approachable and doable for non-professional traders. This consisted of searching through previously conducted research online for learning methodologies and areas where costs can be lowered compared to different competitors.

1.1.4 Our Solution

From the research we conducted, we identified gamification as an engaging method of learning for users. This can be seen in stock trading and other areas alike, for example Robinhood, Duolingo and Khan Academy. Based on our survey, it is a feature of interest to our market and has shown great potential based on its use within other fields. Applying this methodology alongside enticing features such as low commission fees, simplified trading information and the low minimum account balances required to buy stocks, we aim to resolve the recurring issues of cost, time and complexity.

1.2 Management Report

1.2.1 Organisational Structure

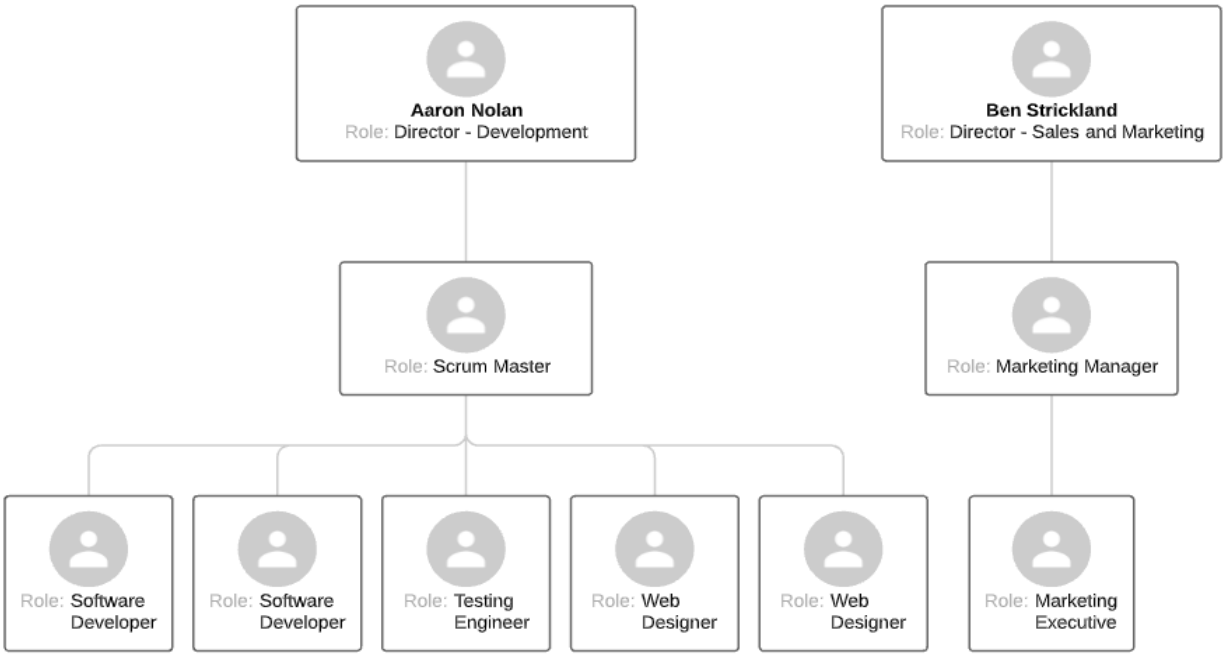


Fig.1 - Organisational Chart

Outlined above is the organisational chart of our company for the first year. As our revenue grows over time and more profits can be reinvested into the business, we hope to be able to grow the number of staff employed in these roles to further expand the application's development. The planned team expansion is detailed in Section 2.5.3. Each of the roles outlined in the chart play a vital role to the success of the company. We have outlined the roles and their responsibilities below.

Directors / Product Owners

Main responsibilities: Manages the funding and structure of the business, and prioritises the product backlog and deliverables for staff. The Director for Development will oversee the development team and ensure that the progress of the application is completed in a timely and satisfactory manner. The Director for Marketing will oversee the marketing team and ensure that advertising completed by the marketing team is correctly geared towards the application's target market, and that the application is publicised ahead of its release and frequently afterwards.

Scrum Master

Main responsibilities: Organising the development team by facilitating daily standup meetings, organising retrospectives at the end of each iteration and eliminating external development blockers. We aim to have two scrum masters overseeing the development and marketing teams respectively to ensure staff working on all areas of the application are catered to.

Software Developer

Main responsibilities: Programming the functionality of the application. Continuous development will have to be performed to ensure any potential bugs are quickly fixed and new features can be integrated.

Testing Engineer

Main responsibilities: Creating tests for functions written by software developers and reporting back when tests fail. This is necessary to prevent or mitigate any bugs of the application from being released and potentially encountered by users.

Web Designers

Main responsibilities: Designing the look and UX/UI of the application. The application will be designed with non-professional traders in mind, so web designers must develop a simplified and easy to understand UI that caters to this market.

Marketing Manager

Main responsibilities: Creating advertising campaigns publicising the application and communicating with third-party clients. The clients in this case would be representatives of the companies providing the APIs used by the application, or potential business partners looking to facilitate the advertising of the application, such as social media representatives.

Marketing Executive

Main responsibilities: Creating advertising campaigns for the application and managing social media. This involves working alongside the Marketing Manager to design advertising that appeals to the application's target market and communicating to existing users and potential customers via the business's social media accounts.

1.2.2 Team CVs

Ben's CV can be found in Figure 19.
Aaron's CV can be found in Figure 20.

1.2.3 Logs

We started our log recording on the week of 14/11/21. For the majority of the year we were filling in a word document with everything done each week, but due to the content being difficult to read we transferred it to an Excel spreadsheet. Both the document and Excel sheets for each of us are linked below:

Aaron's Log

Document	https://docs.google.com/document/d/1LY9Q1Kq2P_KDfH-4tFQiO2SY8iH9pM4vHPO9mAmTwEc/edit?usp=sharing
Excel	https://docs.google.com/spreadsheets/d/1vpmRer3UWkJA2QB4NbHCFiqlcplflCG-pyp47btU4Xk/edit?usp=sharing

Ben's Log

Document	https://docs.google.com/document/d/1SRfBoJi-Ez-C60psPZwG6gkdw2RLTLj5fGSWhYxcFn8/edit
Excel	https://docs.google.com/spreadsheets/d/1sufFYamB1utgYfnJRoTiCo9GNAjwNoVZwXO1cLKe5s0/edit#gid=0

1.2.4 Time description

November: Both team members created and submitted a second draft of the Ethics Document, and held a meeting with the project supervisor. Aaron created a survey.

December: Both team members sent the survey to participants, held a meeting with the project supervisor and created the VPC, BMC, and Financial Plan for the mid-term document. Ben created the Market Analysis section. Aaron created the Competitive Analysis section.

January: Both team members created the Functional Requirements section for the mid-term document, revised and submitted the mid-term document, held meetings with the project supervisor and created the BMC for the final document. Ben created the General Description section for the mid-term document, the Business Case Description, Market Analysis and Value Proposition sections for the final document, and set up the user model, sitemap, User Registration page and Node frontend for the prototype. Aaron created the Software Architecture and ERD sections for the mid-term document and the Executive Summary and Pricing Model sections for the final document, created a UI mockup, set up the Django REST framework, models for transactions and stock balance for the prototype, and configured the Index and Stock pages.

February: Ben started the User Transactions page, configured the User Dashboard and User Details pages, and added validation to the User Registration page for the prototype. Aaron completed the Stock page, created the Leaderboard page, added Twitter Sentiment Analysis and unit tests to the prototype, and added new fee types to the Transactions table.

March: Ben added validation and design to the User Details page, completed the User Dashboard page, completed the Management Report section, expanded the research used for the VPC, updated the BMC and Industry Trends sections for the final document, and completed the About and Welcome pages and fixed the formatting of the User Details and User Transactions pages for the prototype. Aaron created the Software Architecture, Use Cases, Financial Plan and ERD sections and updated the Pricing Models, Key Partners and Competitor Analysis sections for the final document, and updated styling for the prototype.

April: Both team members fixed the formatting of the final document and conducted a meeting with the project supervisor. Ben completed the Performance Requirements, Technical Challenges and Interface Rationale sections of the document. Aaron completed the styling of the About page, fixed test cases, ordered transactions in the User Transactions page by date, added a bank icon to the navbar, added a side nav, a CSV download and added access restrictions to pages by users not signed in.

2. Business Case

2.1 Description

StockCompete is a stock trading web application that provides its users with a range of features to allow them to trade in the EU Market. These features are detailed below.

Low commission fees (no hidden fees): A commission fee is a form of generated revenue used by brokerages, which consists of a service charge paid to the broker for "*handling sales of securities for a client*" [1]. We will apply a commission fee to sell transactions, consisting of 0.1% of the transaction's value, or a flat fee of €2.00 if the percentage of the value would otherwise be lower. By charging these low rates, we will be able to avoid using methods for generating revenue such as Payment for Order Flow, which has been banned in several countries and is planned to be outlawed in the EU entirely [2]. This will be discussed further in Section 2.3.6.

Simplified Trading Information: Trading platforms for professionals or more experienced traders have intricate graphs and technical terminology, such as "*candlestick charts"* [3] with trend lines for high, low and average, which are intimidating to non-professional traders. Our service will provide easy-to-read stock data by showing linear graphs, tables of values and general stock information. The table will include all the values, e.g. high, low and close, of the current day for all listed stocks**.** All of these components will make it easy for non-professional traders to digest the information and help them make better decisions on their trades.

Gamified UI: A user interface "*is the series of screens, pages, and visual elements—like buttons and icons—that enable a person to interact with a product or service*" [4]. The user interface implemented on our website will incorporate gamification elements - this involves *"the application of game-design elements and game principles in non-game contexts*" [5]. Having investigated various types of gamification and the benefits of each, we determined the use of a leaderboard and points system is best suited to our users. Each user will have a score assigned to them - upon sell trades, some points will be calculated based on the profit made by the user, and added to their total score. A leaderboard will be available displaying the ranking of users based on their total score, and each month, the top twenty users will receive various prizes, such as a premium account for the next month or an additional credit balance that they can use to continue trading.

User Dashboard: A user dashboard, as it is displayed on webpages, is "*a user interface that gives a current summary, usually in graphic, easy-to-read form, of key information relating to progress and performance*" [6]. On competing stock trading platforms, this is included to provide key information to the user about their account. In the case of our website, this will include a list of stocks currently owned by the user, including a graph displaying the total purchase value of each stock, and a table displaying the user's current stock balance.

2.2 Value Analysis

2.2.1 Who are the users?

The users of the application are anyone with internet access, an interest in learning how the stock market works and funds to invest in stock trading - specifically non-professionals that lack experience in the market.

2.2.2 Who are the customers?

Our customer base will be located in the EU, due to the market gap that has been identified there from leading industry competitors such as Robinhood not operating within the area.
Both of these sections are further discussed in Section 2.3.3.

2.2.3 How does it help a user/organisation to achieve some goal?

The application is designed to help non-professional traders become more familiar with stock trading and the concepts necessary to understand how to trade successfully. It does this by providing simplified trading information about individual stocks, such as a description of the company and its current closing price, a leaderboard system incentivising users to trade more frequently by offering prizes every month to the top twenty most successful traders, and a user dashboard detailing all information related to the user's account by providing real-time data visualisation on their current portfolio, including the quantity of each stock owned, the total value of each, the current cash balance held and the total amount of money invested.

Additional information is provided to the user that is designed to help them make an informed decision on potential purchases. On the homepage of the application, a list of articles detailing news about the stock market is displayed, and a separate list displaying articles relating to a specific company is included on the company's stock page, along with a feed with the company's most recent tweets. Sentiment analysis is applied to this feed to determine whether the sentiment expressed in the tweets is positive or negative, and a percentage is returned alongside the feed - the higher the percentage, the more positive the sentiment. Future expansion of the business may involve incorporating features that appeal to more experienced traders, such as future prediction of stock prices via machine learning.

2.3 Market Analysis

Market analysis refers to "*a quantitative and qualitative assessment of a market. It looks into the size of the market both in volume and in value, the various customer segments and buying patterns, the competition, and the economic environment in terms of barriers to entry and regulation*" [7]. This involves researching the number of potential customers and the demographic they belong to, the frequency of their spending, and competing businesses. To start creating our market analysis we first need to determine the target market.

2.3.1 Target Market

A target market is "*a group of people with some shared characteristics that a company has identified as potential customers for its products*" [8]. These characteristics could include age range, income, or level of experience.

We determined that our target market will be non-professional traders within the EU Market. To define our target market we first needed to survey many people to get an understanding of people's interests and issues regarding stock trading. As will be detailed in Section 2.3.4, from our results we created a Value Proposition Canvas. Based on the results of surveying our proposed target market, we outlined what their issues were both for those who have traded before and those who haven't. For those surveyed with previous trading experience, there were a lot of common issues faced when entering the trading world, e.g. cost, difficulty interpreting data and a lack of knowledge [Figure 21]. Additionally, when asked if they enjoyed using apps that incorporated gamified elements, such as Duolingo and Khan Academy, the majority of respondents (80%) responded Yes [Figure 26]. From our research, we concluded that our proposed target market is available.

2.3.2 Industry Trends

Following on from identifying our target market, we then looked into current trends to see if this industry is still growing. From our initial research, we found multiple articles detailing the increase in interest in stock trading during the pandemic, particularly within the 18-24 age range. For instance, in the UK, one survey found that "*16 per cent of Britons aged between 18 and 24 began investing for the first time, compared with 10 per cent across all age groups*" [9]. Additionally, similar trends have been observed overseas, with major online brokers in the US seeing "*new accounts grow as much as 170% in the first quarter*" [10] of 2020. Despite the large interest within this target market, it has been noticed that the younger demographics have been cautious about trading large amounts - for example, "*Analysis by UK investment platform FreeTrade found that investors under the age of 25 traded smaller amounts — as a consequence of having less money to invest — but also traded less frequently than older age groups*" [9]. Although this can be seen as a negative trend, we have proposed a solution, which is detailed in Section 2.3.4, to use gamification.
Conversely, older generations have seen an interest in investing more consistently over time, and are more likely to invest larger amounts of money - Fidelity Investments reports that "*almost 40% of 401(k) investors aged 60 to 69 hold about 67% or more of their portfolios in stocks*" [11]. This can be directly contrasted with the investment habits of younger traders - according to Personal Capital, "*younger investors in their 20s park a higher percentage of their assets in cash (28.4%) than any other age group except retirees in their 80s (29.4%) and 90s (31%)*" [12].

2.3.3 User Segments

As was detailed in Section 2.2.1, the main user segment for our application is non-professional traders within the EU market. This is because, from our research, we determined that there is a sizable number of people within this criteria who have recently been introduced to trading. Additionally, multiple competing companies such as Robinhood exclusively cater to the US, making it easier to gain market share in the EU.

2.3.4 Value Proposition Canvas (VPC)

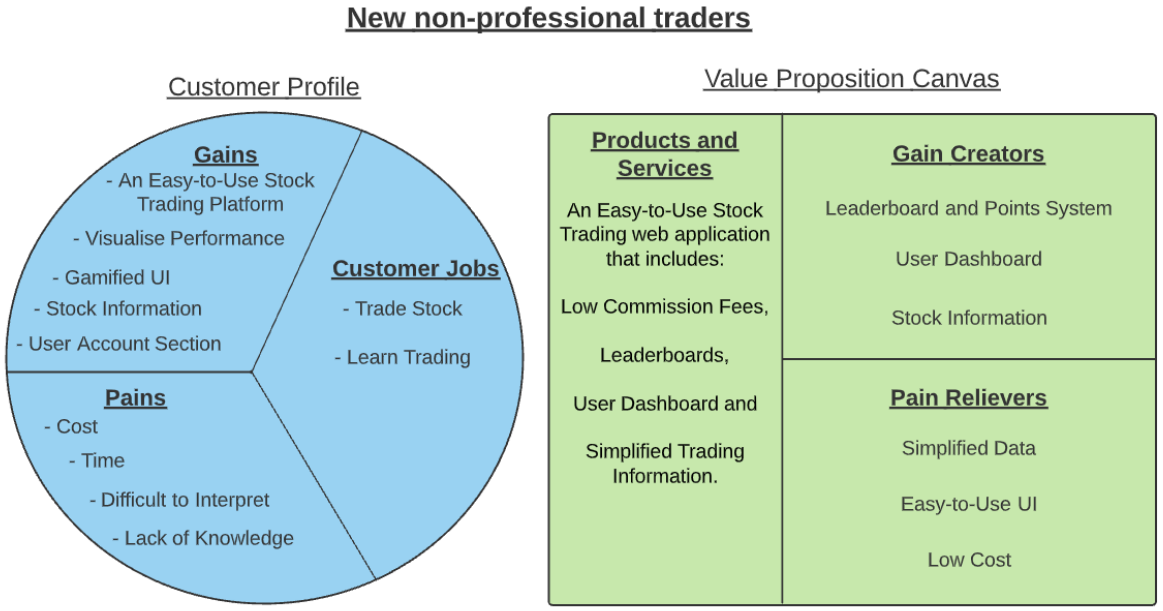


Fig. 2 - Value Proposition Diagram
Ref: Nolan, A. N., & Strickland, B. S. (2022, January). Final Year Project: Mid-Term Delivery.

Customer Profile

As part of our analysis of our target market, we created a customer profile detailing their characteristics. These were gathered via a survey we conducted that targeted non-professional traders. The Customer Jobs listed above refer to what the customer is attempting to do when using an application similar to ours - in this case, they would be attempting to trade stock and learn how to trade successfully.

From our research, we determined that they often encountered issues (or pains) when attempting to perform these jobs, which are listed in the Pains section as cost, time, difficulty in interpreting data and a lack of knowledge. These were determined via the aforementioned survey - all Pains listed were answers to the questions “If [you have traded stock before], were there any difficulties you faced?” and “If [you haven’t traded stock before], was there any reason(s) why?”. These can be seen in Figures 21 and 22 respectively. Industry reports also returned these Pains as primary issues - a questionnaire aimed at non-professional traders was conducted as part of a research document chronicling their experiences.

From this, it was revealed that significant time and cost were expended by interviewees to become more proficient at trading, with one reporting that he had “*spent several thousand pounds on purchasing training CDs and charting software and attending courses*” to learn charting skills, and “*At the time of interview he had spent all his spare time for nine months working on it and intended going full-time after another four months*” [13].

We also determined a list of features that the user would expect a stock trading application to have, which are listed in the Gains section as an easy-to-use trading platform, including a method to visualise their performance, a gamified user interface, simplified stock information and a user account section. The Gains can be seen in Figure 25.

Value Proposition Canvas

Having completed the Customer Profile, we created a Value Proposition Canvas. First, we identified features that could be implemented to alleviate the issues customers frequently encountered, which are listed in the Pain Relievers section as simplified trading data, an easy-to-use user interface, and a low cost applied to the customer's transactions.

We also identified a list of features that could be implemented to benefit customers when attempting to carry out these jobs, which are listed in the Gain Creators section as a user dashboard that displays account information and simplified stock information. Also listed is a leaderboard and points system, which aims to offer a method for visualising performance and a gamified user interface, which were listed in the Gains section of the Customer Profile.

The leaderboard system's implementation is done to provide a unique form of gamification within the stock market, which has proven to be a successful feature - Forbes reports that gamification in fintech has become "*closer to a $200 billion global industry with almost three billion consumers across all ages and demographics*" [14]. Using industry competitor Robinhood as an example, which has employed gamification as one of its major selling points from its inception, its user base increased exponentially over the first five years of its operation, growing from half a million users in 2014 to ten million users in 2019 [15]. Additionally, the use of gamification across other business models has seen great success - according to Fortune Business Insights, the global market size for industries employing gamification "*stood at USD 6.33 billion in 2019 and is projected to reach USD 37.00 billion by 2027*", with numerous companies across industries such as retail and education adopting the concept [16].

From these, we then identified a product that could fulfil the criteria outlined in the other sections - an easy-to-use stock trading web application that includes low commission fees, a leaderboard and points system, a user dashboard and simplified trading information.

2.3.5 Business Model Canvas (BMC)

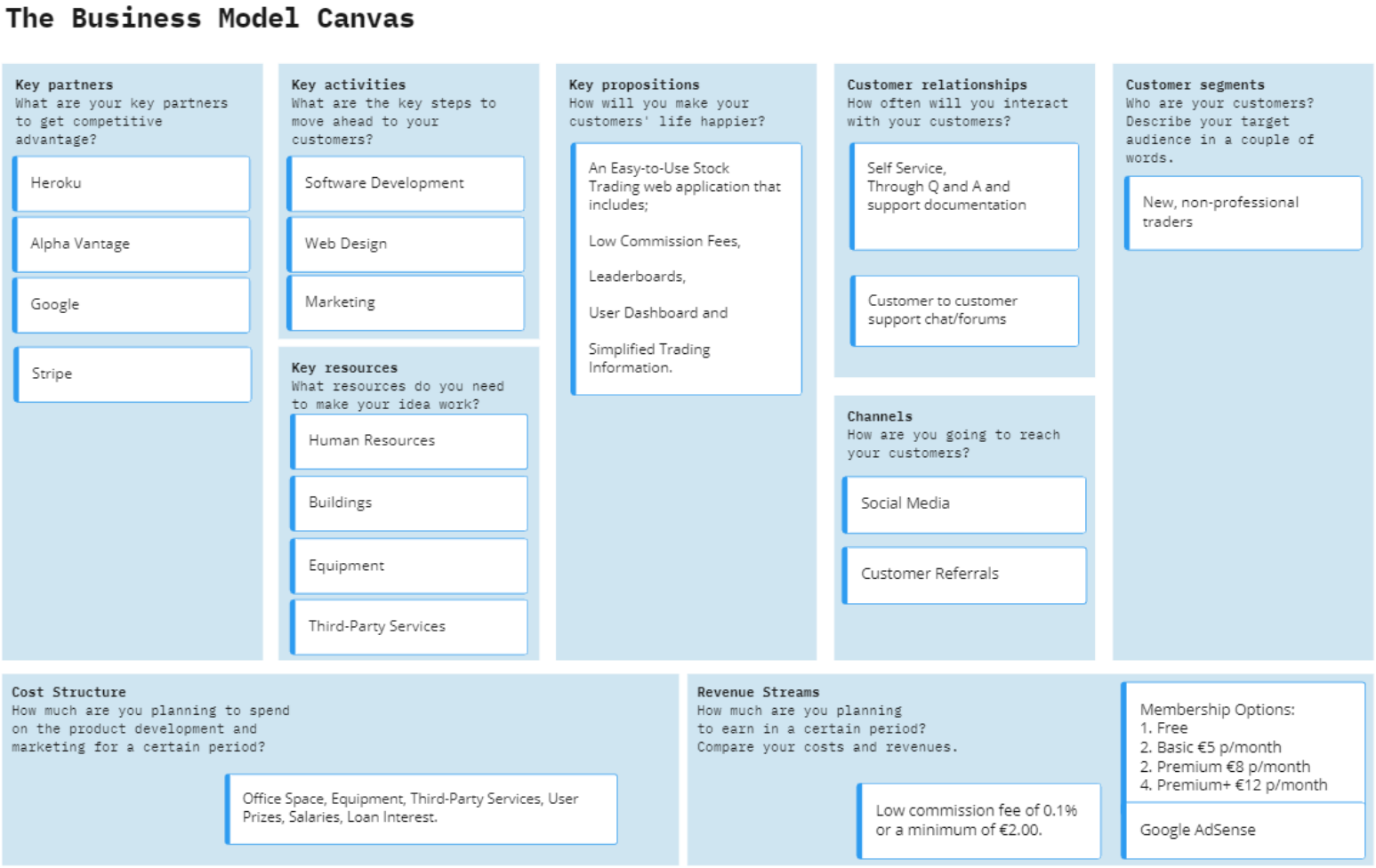


Fig. 3 - Business Model Canvas

The Business Model Canvas above outlines the main areas we researched and investigated in preparation for this business idea. These areas consist of Customer Segments, Key Propositions, Customer Relationships, Channels, Revenue Streams, Key Partners, Key Activities, Key Resources and Cost Structure. These sections cover all aspects of the business when planning and with the exception of Customer Segments and Key Propositions, which have been identified and detailed in Section 2.2 and Section 2.3.4, each is detailed below.

Customer Relationships

Customer relationships are how a business establishes a connection with its customers. It is used to get, keep and grow your customer base. For our business, we defined some effective methods of customer relationships. These methods are self-service through Q & A sections and customer-to-customer support. Based on understanding the trends of customer support where, *"80% of community programs report an increase in asking and answering behaviours, which are critical to capturing implicit knowledge and making it transparent and accessible"* [17], it is clear that these methods are in line with current methods and trends.

Channels

Channels are the medium through which we reach our customers and find new ones. Our main channel will be through social media and customer referrals. These channels allow us to reach our customer base in a cheap but effective way. Social Media is a cheap, easily accessible and effective way of advertising our service as well as keeping our customers up to date with current events in the company. Customer referrals are a way of having reviews and recommendations from our customers to entice new customers to join. Based on research, "*recommendations and reviews are trusted more than conventional advertisements*". The "*customers that have been referred by a friend, family member, or colleague have a lifetime value that is, on average, 25% higher than that of other customers*" [18] which is a much higher value to us in the long term.

Revenue Streams

Identifying revenue streams is a key aspect of any business as they are how a company generates an income. We've identified the main revenue streams for our business idea as a low commission fee, which will be 0.1% of the value of a sell transaction, or a set cost of €2.00 if the value would otherwise be less, a tiered monthly membership system which allows users to pay a monthly subscription fee to gain access to extra features such as a set number of commission-free trades per month, and a partnership with Google AdSense, which pays an amount to the company based on the number of views and clicks their ads get from our users. This section will be further elaborated on in Section 2.3.6.

Key Partners

Our business will not be able to create all the services it needs to run from the start. To solve this, business partners are made who will provide our business with some essential services we need to run. The business partners we've identified are Heroku, AlphaVantage, Google and Stripe. Heroku will provide us with a database to store our data and a server to make our web app available to the public. AlphaVantage is a third-party API that will provide us with stock time-series data and real-time trading data to populate stock prices and graphs on our website. Google will provide us access to their Google AdSense, which will allow us to display ads on our website. This will generate a small additional amount of revenue every month. Stripe will provide us with an API for processing user payments, which will provide additional security for user credit card information.

Key Activities

Several tasks will be necessary to carry out for the application to be completed and to acquire a user base. According to our research, these will consist of Software Development, which will involve backend development (configuring the content of the server, such as the database and framework) and frontend development (creating the visual representation of the application that users will interact with). Also required will be Web Design, which will involve creating the visual design of the application while taking into consideration what the user may "*want, need, and feel*" [19]. Additionally, Marketing is a key activity for the business, as advertising will be vital for attracting customers to the application.

Key Resources

For the application to be developed and for the business to continually operate, there are several resources required. In this case, these resources consist of Human Resources, which will include software developers to program the application, web designers to create the visual layout of the application, and marketers to design advertisements promoting the application. Additionally required will be Buildings for employees to work in, Equipment such as computer hardware and software will be required for employees to work on, and Third-Party Services required for the development of the application, such as Alpha Vantage's API, Heroku's cloud application services and Google's AdSense program.

Cost Structure

There are areas of expenditure that the business will be required to pay to develop and continue to provide the application. This will be divided between Office Space, which is required for employees to work in, Equipment, which employees will require to complete work, Third-Party Services, which incur a cost for the use of their services, User Prizes, which will be given to users at the top of the application's leaderboard every month, Salaries, which will be paid to employees, and Loan Interest, which will be paid to investors who provided capital for the business. This section will be further elaborated on in Section 2.5.

2.3.6 Pricing Models

To generate revenue from the application and ensure that the business remains a profitable venture, there are two pricing model implemented that outlines how the business will generate revenue, Commission Fees and Account Memberships.

Commission Fees

"Commission Fee means a fee the Company receives as an agent and/or broker for a customer at the rate determined by the Company or the relevant laws which shall include taxes, fees, duties and any other expenses which may be incurred" [20].

Our commission fee is a charge that will be applied on sell transactions, consisting of 0.1% of the transaction's value or a fee of €2 if the value would otherwise be less than that. For example, if a Sell transaction was made at a total of €1000, the commission would equal €10.00, which is greater than the low rate of €2.00 and therefore the user will be charged €10.00. This will also be subject to any additional fees that may be charged in creating their market order.

Commission Fee Rates

Minimum Fee	**General Fee**
€2.00	0.1%

Competitors such as eToro, Degiro and plus500 offer low to no commission fees, however they charge hidden fees such as overnight, weekend and account fees instead or charge based on bid spreads. Other competitors offer no commission fees by using an income stream called payment for order flow (PFOF), a legally questionable revenue stream that is banned in some EU countries [21]. These are further detailed in Section 2.3.7.

Account Memberships

Alternatively, users can avail of the tiered account membership options, where a monthly fee is paid to access premium features such as the removal of ads and the aforementioned commission fees. There will be three options offered for membership:

Account Membership Rates

Basic (€5)	**Premium (€8)**	**Premium+ (€12)**
2 Commission-Free Trades	3 Commission-Free Trades	5 Commission-Free Trades
Removal of Ads	Removal of Ads	Removal of Ads
-	Google News	Google News
-	-	Twitter Sentiment Analysis

With further development, we plan to offer additional features as part of this subscription plan. Pricing models based on account membership types can be seen on competitor platforms such as Robinhood [22], eToro [23] and Revolut [24]. As these competitors have been around for several years it can be assumed that using this kind of pricing model is effective in the market.

Payment for Order Flow (PFOF)

"*Payment for order flow is the compensation and benefit a brokerage firm receives for directing orders to different parties for trade execution. The brokerage firm receives a small payment, usually fractions of a penny per share, as compensation for directing the order to a particular market maker.*" [21].

This pricing model allows companies to make a profit off of the customers' trade and not have to charge the customer commission fees and is currently employed by Robinhood, Fidelity and E-Trade. There are some legal issues with using this process however, as the EU has currently banned it in several countries such as the United Kingdom, Canada and Australia [2] and is continuing to investigate this process [25]. This investigation may potentially see PFOF being completely banned in all EU markets.

2.3.7 Competitor Analysis

Competitor Analysis involves analysing existing businesses serving the current market and comparing them to our business, by identifying the key differences and selling points of their services to ours. We have analysed over ten companies [Figures 23 and 24] and some of the top competitors our service would be up against are Plus500, eToro and Revolut. These three platforms all possess characteristics similar to what we plan on implementing into our application, and have different methods of generating revenue.

plus500

Plus500's platform allows users to trade in global stock exchanges, including those based in the US and EU Market. It generates revenue through its low commission fees, spreads and hidden fees. They also users to complete stock trades and Contract for Differences (CFD) trades and their main target market is the EU. In comparison to our platform, although

plus500 allows users to trade CFDs and also US stocks, they do not implement a gamified UI. Their low commission fees for trading in Ireland, for example, are at a rate of 0.075% or a minimum of €5.25, in comparison to a rate of 0.1% or a minimum of €2.

Interactive Brokers

Interactive Brokers are an international trading platform for professional traders. They carry out trades on almost all global stock exchanges and cater to traders transacting in the 10s-100s of millions. It offers trades on stocks and CFDs which allows users to trade based on their preferences and knowledge, however its advanced, in-depth analytical tools would be too overwhelming for a non-professional trader. It doesn't offer a gamified platform and an easy-to-read interface.

eToro

eToro is a stock trading platform that operates globally (except in the US). They offer no commission fee to their users and generate revenue through several ‘hidden fees’, which are not highlighted to the user and are applied for a variety of reasons, such as user inactivity and withdrawal of funds. eToro has not implemented any form of gamification and operates as a standard platform that trades CFDs only, which are charged by spread fees applied to transactions instead of commission fees. For this reason, they will not provide competition for the type of trading we are implementing.

Competitive Advantage

Our competitive advantage over contemporary trading platforms are the use of a gamified UI, presentation of easy-to-read trading information to users, specialisation in the EU Stock Market, and emphasis on catering to non-professional traders.

Indirect Competitor: Robinhood

Robinhood is a gamified trading platform based in the United States. Its marketing primarily consists of advertising campaigns promoting its website and mobile application through social media, and it generates revenue through the use of the ‘Payment for Order Flow’ (PFOF) process, which allows them to offer no commission fees to its customers. We have deemed them an indirect competitor as they aren’t based in our market and also may be completely banned from the EU market depending on the ongoing investigations of PFOF.

2.4 Research

Our primary research consisted of a survey collecting anonymised data. It was targeted toward people with little or no knowledge or experience with stock trading and aimed to gather more information on our target market. Examples of questions used within the survey can be found in Figures 21, 22, 25 and 26. The answers gathered were used to further detail our Value Proposition Canvas, and understand the features necessary for implementation.
Our secondary research consisted of several industry reports containing information on industry competitors' performance via statistics, such as their annual revenue and user base. Industry trends were also researched to understand the current interest in the market if there were any gaps (such as in a particular location or demographic) that we could potentially cater to, and what features previously implemented by competitors had seen success.

2.5 Financial Projection and Requirements

INVESTMENT INCOME	Year 1	Year 2	Year 3	TOTALS
Enterprise Ireland	€230,000	€0	€0	€230,000
BOI Loan	€60,000	€0	€0	€60,000
TOTALS	€290,000	€0	€0	€290,000
OPERATING REVENUE	Year 1	Year 2	Year 3	TOTALS
Commission Fees	€434,160	€2,819,832	€7,187,400	€10,441,392
Subscriptions	€165,825	€1,077,019	€2,745,188	€3,988,032
Ads	€32,232	€82,116	€210,432	€324,780
TOTALS	€632,217	€3,978,968	€10,143,020	€14,754,204
MONTHLY EXPENSES	Year 1	Year 2	Year 3	TOTALS
Product Owner	€70,000	€360,000	€1,200,000	€1,630,000
Development Team				
Scrum Master	€53,200	€106,400	€106,400	€266,000
Software Developer	€123,500	€247,000	€370,500	€741,000
Test Engineer	€35,200	€70,400	€140,800	€246,400
Web Designer	€57,000	€82,500	€114,000	€253,500
Marketing Team				
Marketing Manager	€38,732	€43,035	€86,070	€167,837
Marketing Executive	€25,650	€85,500	€171,000	€282,150
Budget	€63,224	€1,943,846	€2,400,000	€4,407,070
Equipment				
Laptop (Development)	€8,840	€5,525	€14,365	€28,729
Laptop (Marketing)	€1,550	€1,550	€4,650	€7,750
Monitor	€7,180	€5,026	€6,462	€18,667
Keyboard and Mouse	€250	€175	€950	€1,375
Heroku (Domain Hosting)	€6,000	€6,000	€6,000	€18,000
Rent	€36,000	€108,000	€108,000	€252,000
Misc	€6,000	€6,000	€6,000	€18,000
User Prizes	€1,200	€2,400	€2,400	€6,000
Loan Interest	€41,380	€20,690	€0	€62,070
BURN RATE	€574,905	€3,094,047	€4,737,596	€8,406,548
OVERVIEW	Year 1	Year 2	Year 3	TOTALS
CUMMULATIVE EXPENSES	€574,905	€3,094,047	€4,737,596	€8,406,548
CUMMULATIVE INCOME	€922,217	€4,326,279	€11,375,252	€16,623,748
STATUS	€347,312	€1,232,233	€6,637,656	€8,217,200

Fig. 4 - Three Year Financial Projection

This figure displays the summarised financial projection for the first three years. These can be seen further broken down in Figures 27, 28 and 29.

2.5.1 Investment Income

Using Bank of Ireland's Loan Application, we will withdraw a €60,000 loan in April of the first year to cover excess costs. We estimate that this will cost us a monthly repayment of €5,172.54, which totals €62,070.47 [26]. Enterprise Ireland will also provide a loan of €230,000 via their High Potential Start-Up (HPSU) Funding, which supports business start-ups that plan to create ten jobs and €1,000,000 in revenue within the first three years of operation [27].

2.5.2 Operating Revenue

Estimating User Growth

Number of Users (Competitors)

Based on the information gathered from Statista, we created a table [Figure 30] outlining the number of users after each of the first three years for eToro and Robinhood. We chose these competitors as they had the most reliable data to use for our estimation. From this, we decided to take the lower values for estimating our user growth in the first three years and then further deducted 30% of the total user count as we cannot guarantee we will have the same marketing success. This would mean our estimated user growth rate would be:

Number of Users (StockCompete)

Company	**Year 1**	**Year 2**	**Year 3**
StockCompete	119,000	301,000	770,000

This is further broken down in Figure 31.

Paid Account Memberships

As data is not open to the public about the quantity of users and their type in industry competitors, we have no reliable method of accurately estimating the percentage of users that will purchase an account membership. For this reason, we will assume that the low amount of only 5% of users will purchase a paid account membership and we will divide this evenly between the three account memberships each month [Figure 32].

Google Ads Memberships

Using Google Adsense's revenue predictor [30], we calculated the revenue for each year.

	Year 1	**Year 2**	**Year 3**
Google Adsense	€32,230	€82,118	€210,430

Commission Fees

As outlined in Section 2.3.6, we will charge a minimum €2 fee per sell transaction or 0.1% of the total in commission fees. There was no data we could find identifying the quantity of users with paid memberships in competing platforms. For estimation purposes, we will take a very small estimate of 5% of users that will purchase a membership each month.

From our total set of users; 5% will purchase monthly memberships (as outlined above) and 35% may be inactive or are holding their trades for longer periods of time (this is guessed for worst case scenario). For these reasons, we have deducted them from the total user population and calculated the total commission fees of the remaining 60% of users if they were to make only one trade per month [Figure 33].

Research

We also investigated the generated revenue of competitors and found that; As of 2021, Robinhood had 22.5 million active users, and as of 2020, generated $959 million (~€864

million) in revenue [15] and as of June 2021, eToro had 20 million active users, and as of 2020, generated $605 million (~€545 million) in revenue [31].

2.5.3 Monthly Expenses

Development Team

Based on our research, our agile team will need between 3-9 people [32]. Having investigated the necessary parts of the team we determined the number of people needed each year. In the first year, we will start with one full development team, in year two we will expand to two smaller sized development teams and by year three we estimate that we will have two full development teams. These roles are detailed on a year by year basis below:

Year 1

No.	Employee	Quantity	Salary	Shares Deducted *	Total
1	Scrum Master	1	€56,000 [33]	€2,800	€53,200
2	Software Developer	2	€65,000 [34]	€3,250	€123,500
3	Testing Engineer	1	€36,000 [35]	€1,800	€35,200
4	Web Designer	2	€30,000 [36]	€1,500	€57,000

Year 2

No.	Employee	Quantity	Salary	Shares Deducted *	Total
1	Scrum Master	2 (+1)	€56,000 [33]	€2,800	€106,400
2	Software Developer	4 (+2)	€65,000 [34]	€3,250	€247,000
3	Testing Engineer	2 (+1)	€36,000 [35]	€1,800	€70,400
4	Web Designer	3 (+1)	€30,000 [36]	€1,500	€82,500

Year 3

No.	Employee	Quantity	Salary	Shares Deducted *	Total
1	Scrum Master	2	€56,000 [33]	€2,800	€106,400
2	Software Developer	6 (+2)	€65,000 [34]	€3,250	€370,500
3	Testing Engineer	4 (+2)	€36,000 [35]	€1,800	€140,800
4	Web Designer	4 (+1)	€30,000 [36]	€1,500	€114,000

* As we will be offering share options, the salaries outlined above will be decreased by 5%.

Marketing Team

For our service to reach the public we need to be marketing our product and have people to deal with any potential partners and customers. We also need to allocate a marketing budget to the team to help them promote the application. For this, we have decided to create an initial team of two marketers and allocate a budget based on projected revenue. As there will be no product available until six months after starting, we estimate that the marketing team will only need to start in month four (April). This team will then expand over the three years from a team of two to four and then to eight.

To calculate the marketing team budget for advertising necessary to make our service known, we will take 10% of the yearly estimated revenue in the first year, just lower than 50% in the second year and roughly 23% in the third year [37].

Year 1

No.	Employee	Quantity	Salary	Shares Deducted*	Total (9 Months)
1	Marketing Manager	1	€45,300 [38]	€2,265	€38,732
2	Marketing Executive	1	€30,000 [39]	€1,500	€25,650

Marketing Team Budget	€63,224

Year 2

No.	Employee	Quantity	Salary	Shares Deducted *	Total
1	Marketing Manager	1	€45,300 [38]	€2,265	€43,035
2	Marketing Executive	3 (+2)	€30,000 [39]	€1,500	€85,500

Marketing Team Budget	€1,943,846

Year 3

No.	Employee	Quantity	Salary	Shares Deducted *	Total
1	Marketing Manager	2 (+1)	€45,300 [38]	€2,265	€86,070
2	Marketing Executive	6 (+3)	€30,000 [39]	€1,500	€171,000

Marketing Team Budget	€2,400,000

Equipment

We have outlined all necessary equipment needed to help our employees complete their jobs. For each piece of equipment, we compared specs, pricing and reliability to ensure we choose the best equipment for our business.

Laptops, Keyboards, Mice, Mousepads and Headsets

For the **development team**, each employee will need a powerful laptop to be able to run the server locally and perform their daily tasks. We estimate that laptops with 16GB RAM and a newer intel processor are enough to carry out daily tasks. For these reasons, we've opted to use Lenovo ThinkBook 15 G2 [40] for €1104.97 per laptop. For the **marketing team**, we found a laptop with 8GB RAM and i5 processor that we believe to be sufficient, Lenovo ThinkBook 15 Gen 2 [41] at €774.97 per laptop. We will need to supply our employees with Keyboards, Mice, Mousepads and Headsets for them to carry out their daily tasks. This will cost €24.90 per group set [42].

We expect to reinvest in these initial teams after two years to ensure these critical pieces of equipment are always working and able to run the tools needed.

Monitors

We have chosen to give widescreen monitors to both development and marketing teams as this will make it easier for them to have multiple windows open and make their work easier. These monitors are Samsung S65UA 34" Ultra WQHD HDR Curved Monitors and are priced at €717.97 each [43].

Heroku

For the server to support a high number of concurrent users, and return stock information at low latency, a high-performance server will be required. For this, we have chosen Heroku, a third-party cloud platform that offers this service. Their 'Performance L' server option comes with options for automatically scaling performance based on usage and is built with 14GB of RAM. It is priced at €500 a month [44].

Rent

An office space will be required for employees to work in, which must include necessary furniture and internet access. For the first year of operation, we plan the employment of ten staff, and based on existing office spaces in Dublin that would accommodate this, we estimate a rental cost of €3,000 a month [45]. In subsequent years, a larger office space will be required to accommodate more staff - fifteen staff in the second year and twenty staff in the third year respectively. For this, we estimate a rental cost of €9,000 a month [46].

User Prizes

Prizes will be given to the top-performing users each month. For first place, we plan to give €300, for second place €200 and third place €100. For fourth, to tenth place, we plan to give €50, and for eleventh, to twentieth place, a premium subscription lasting one month will be granted. This totals €950 for twenty users. In subsequent years, we would look to expand the number of users eligible for prizes to accommodate the larger user base.

Miscellaneous

Funds will need to be allocated for miscellaneous expenses that may occur unforeseen, such as emergency repairs, or for events that may not be planned over a long time, such as staff social events. We plan to set €500 a month aside for this purpose.

Loan Interest

A loan of €60,000 will result in twelve monthly repayments of €5,172.54.

2.6 BPMN (Business Process Model and Notation)

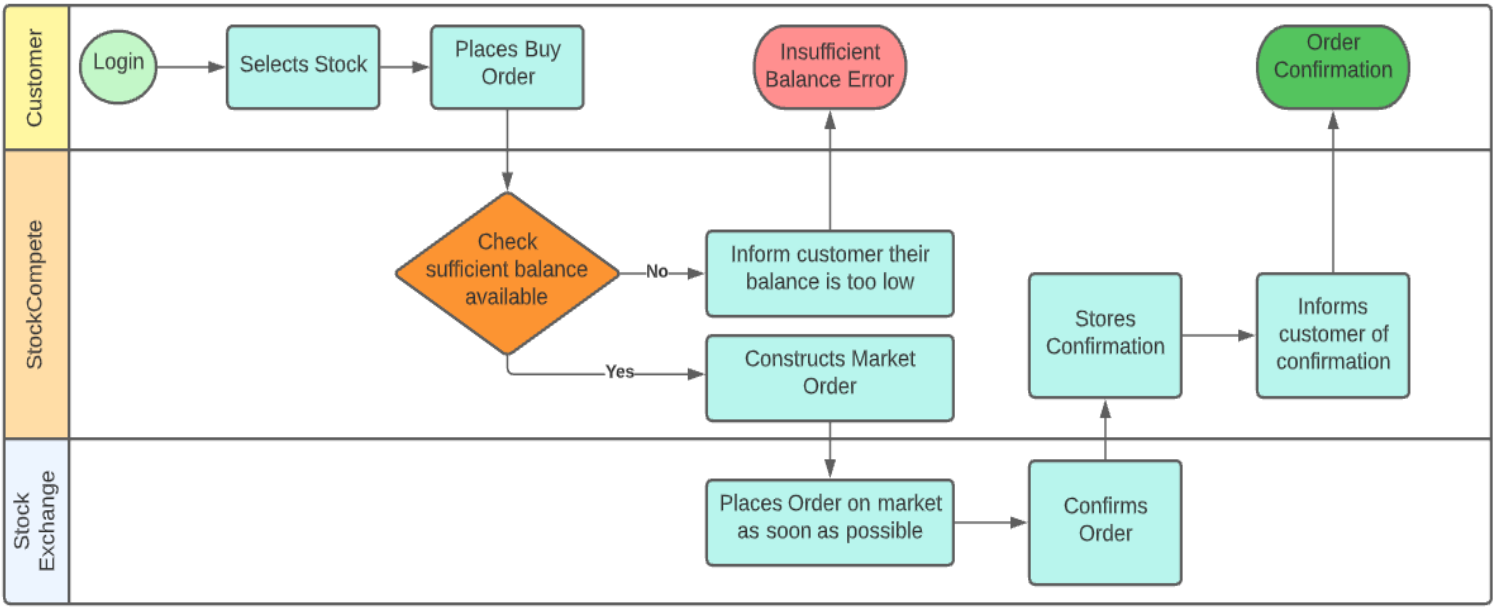


Fig. 5 - Market Order Business Process

3. Technical Delivery

3.1 Software Architecture

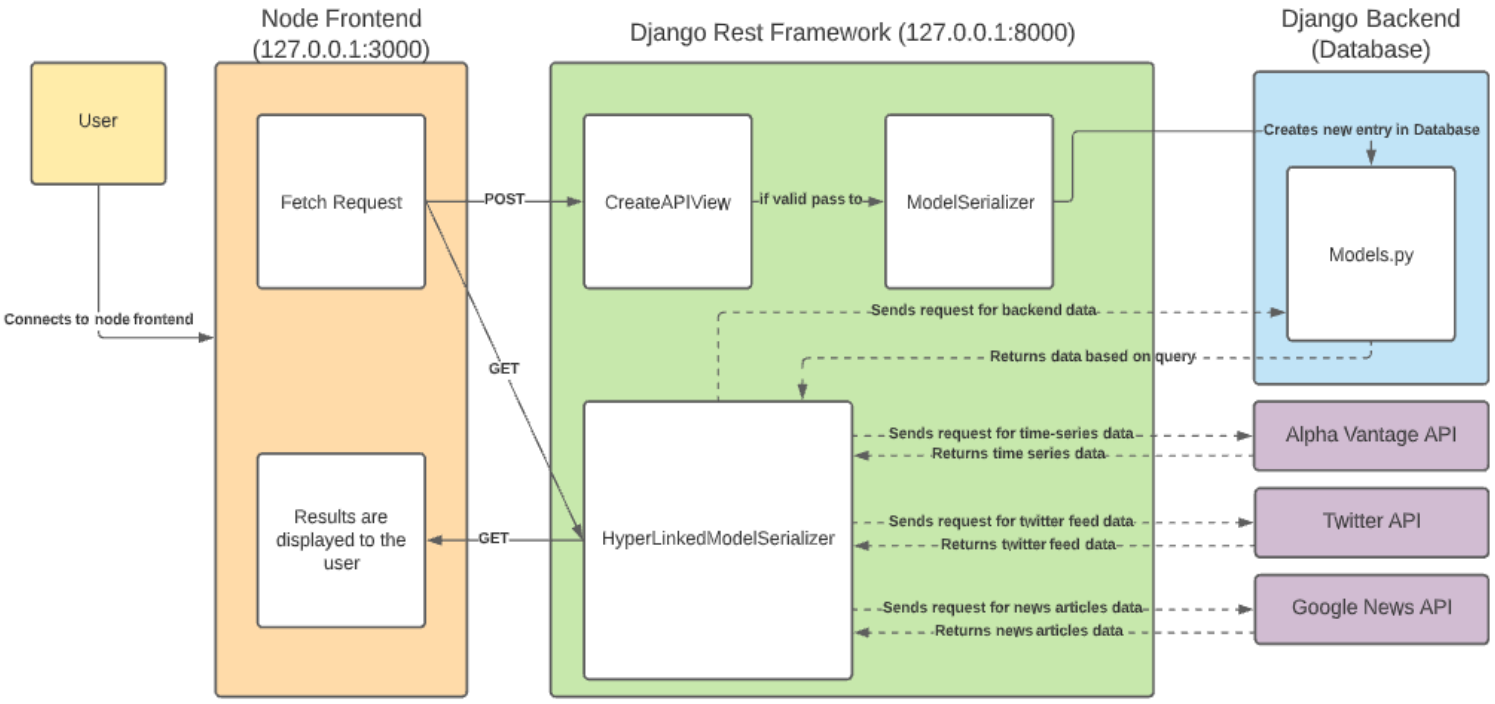


Fig. 6 - Software Architecture Diagram

The system software architecture visually displays high-level components of a system, the structure they are in and how they will interact and communicate with each other. The architecture of our project consists of NodeJS as our frontend framework, Django REST Framework as our REST API and Django as our backend framework. It also includes a range of third-party APIs such as Twitter and Alpha Vantage to help us gain access to data needed to provide our service. With reference to the diagram above, the components will be detailed below:

3.1.1 User

The user is the person who will be interacting with our system, primarily our customers. From their browser, they will access the service by directing it to the frontend URL (127.0.0.1:3000). This URL on port 3000 displays the NodeJS Frontend.

3.1.2 Node Frontend

The Node Frontend is built using Express.js. Express.js is a lightweight, fast and minimalist web framework for NodeJS. We chose this framework because of the tight time frame we have to complete development and also so we can have a frontend framework. In a more professional setting, this work would be translated to React for scalability and maintainability.

Express is responsible for displaying content to the user and allowing them to interact with the system via 127.0.0.1:3000. When the user first loads the web page, several GET requests are sent to the REST API for different data, e.g. stock prices, news articles and Twitter feeds. The REST API then returns a JSON object, which is deconstructed and displayed on the page. It then allows the user to input details, which are then sent to the REST API via a POST request, which is then stored on the database via the models.

3.1.3 Django REST Framework (DRF)

The centre part of this architecture is the Django REST Framework. A REST API is an "*architectural style for building web services that interact via an HTTP protocol*" [47]. Django REST Framework is a Django extension that "*provides functionality to get started creating APIs faster*" [48]. In this architecture, DRF allows the frontend to interact with the database while also ensuring its security by restricting access to only port 3000 (NodeJS frontend) and including API token checks to restrict access to sensitive data such as user or account information. The Node frontend can interact with the DRF via fetch requests, which are received by serializers and views to fetch data. If the fetch was a GET request, these views would then retrieve data from the relevant model and, if specified, filter them by given parameters. The results are then returned through the serializers, converted to JSON and returned to the Node frontend. If the fetch was a POST request, these views would then verify the access token, process the data parameters given and then add them to the relevant model. An HTTP Response code is then returned to the frontend so it can be confirmed if the POST request succeeded.

3.1.4 Django Backend

Django is a web framework with an MVT Architecture. This means that models are used to store data for the system, Views are used to manipulate the data and carry out the system's logic and Templates are used to render the Django interface. For this system, however, the architecture is a bit different as the DRF extension removes the need for the template, as data is passed to an accessible REST API, as mentioned above.

These models are used to store almost all data used in the system, such as users, transactions and leaderboards. When POST requests are made, the Django models are given new sets of data to store and when GET requests are made, data is taken from these models and sent back through the REST API. The database structures are described in further detail in Section 3.2.

3.1.5 Third-Party APIs

Third-Party APIs are used to provide our service. They give us access to data needed to provide real-time data and information related to stocks. These APIs consist of:

AlphaVantage

AlphaVantage is an API that provides time-series data among a range of other factors on a large variety of stocks. This API gives access to historical time-series data for up to twenty years. This is then formatted to be displayed in a linear graph of different time intervals.
Due to late changes in services provided by this API, we could not retrieve the current day's stock values. We then used FinnHub to gain access to real-time stock values and also had these values appended to the end of time-series data.

Twitter

Twitter is a social media platform used by the public and enterprises alike. Companies use this platform to inform potential customers of upcoming events, services provided and any company updates. We used their API to gain access to Twitter feeds of different company accounts. Sentiment analysis was then performed on these tweets and the results were also passed back to NodeJS.

Google News

Google News is a service provided by Google. It allows users to access news articles by scalping numerous news and articles websites for specific strings. For our service, we will use this API to access articles related to the Stock Market and also for individual stocks to provide our users with a better insight into the market in its current state and help them make better-informed decisions when trading.

3.2 Database

3.2.1 Logical Database Design

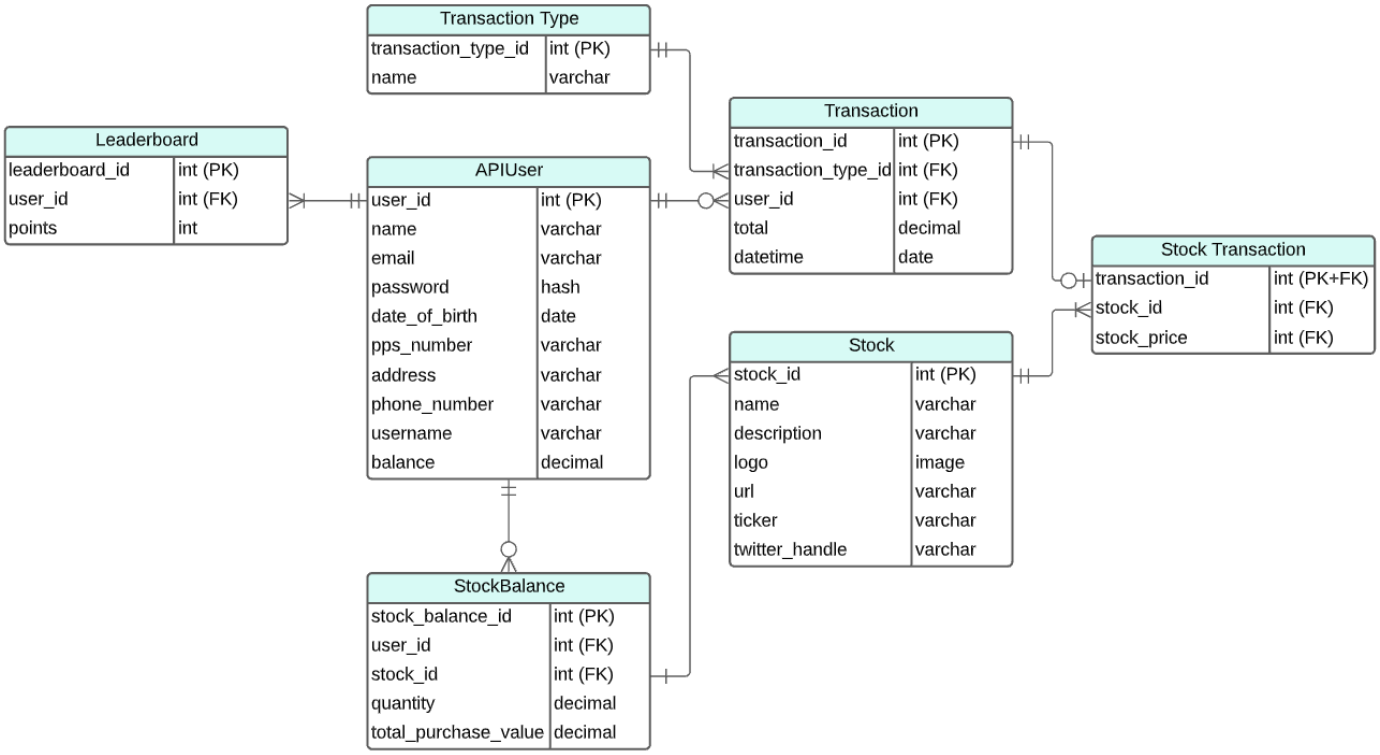


Fig. 7 - Entity Relationship Diagram

The database, as outlined in Section 3.1.4, is an SQLite database that is configured through the inbuilt templates of Django. These templates are called models. As shown in the diagram above, the tables labelled outline the main data used to complete our service.

The APIUser has an entry for every user registered. Each entry contains the user's name, email address, password, date of birth, PPSN, address, phone number, username and balance. The password will be read in as a string, however due to Django's inbuilt User model, the password is then encrypted as a hash variable. This means it can't be decrypted and the only way to determine if the password is correct when they sign in is to convert their input to a hash and compare the two hash values. The other tables store data related to stocks, leaderboards and different types of transactions.

3.2.2 Data Storage Requirements

"SQLite provides an excellent development alternative for applications that are predominantly read-only or require a smaller installation footprint." [49]. Due to this, we plan to transfer to a PostgreSQL database, which is quite easy to do due to Django's abstraction layer. This will provide better 'security', 'extendability' and 'reliability' ("ACID compliant") [50].

3.3 Functional Layout

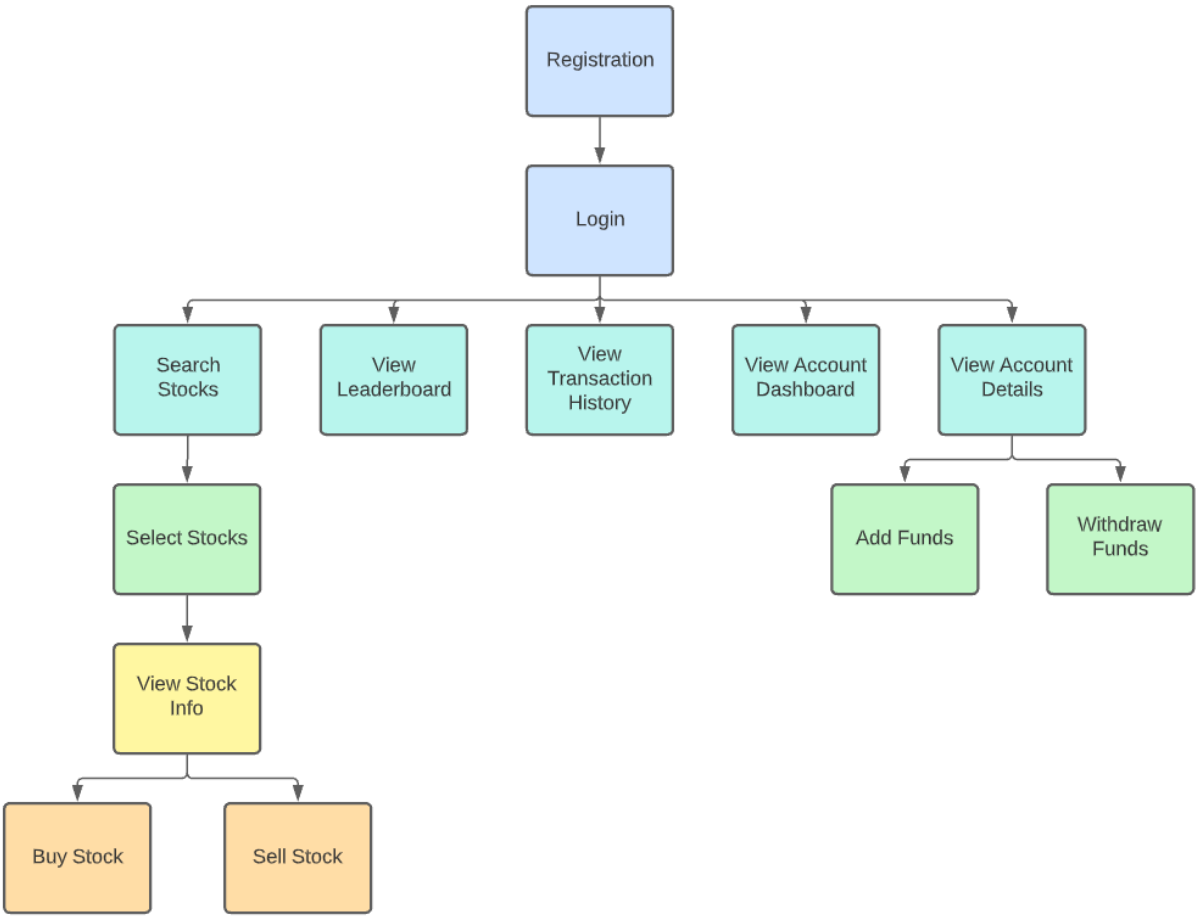


Fig. 8 - Functional Layout Diagram

The diagram above outlines the main functionalities of the system. The user must first register and then sign in. From here they can search stocks, view the monthly leaderboard, transaction history, account dashboard or account details.

After searching for a stock they can then select one and view a further breakdown of information related to it. This includes a company mission statement, current stock value, time-series graphs, news articles and Twitter feeds with sentiment analysis. They will then have the option to buy or sell stocks depending on their stock or account balances.

If they view the account dashboard they can then fill in payment details and either add or withdraw funds from their account. If they withdraw funds it is the user's responsibility to report any taxes due.

These functions are further explained in the use cases identified in Section 3.4 below.

3.4 Use Cases

3.4.1 Data Flow Diagrams

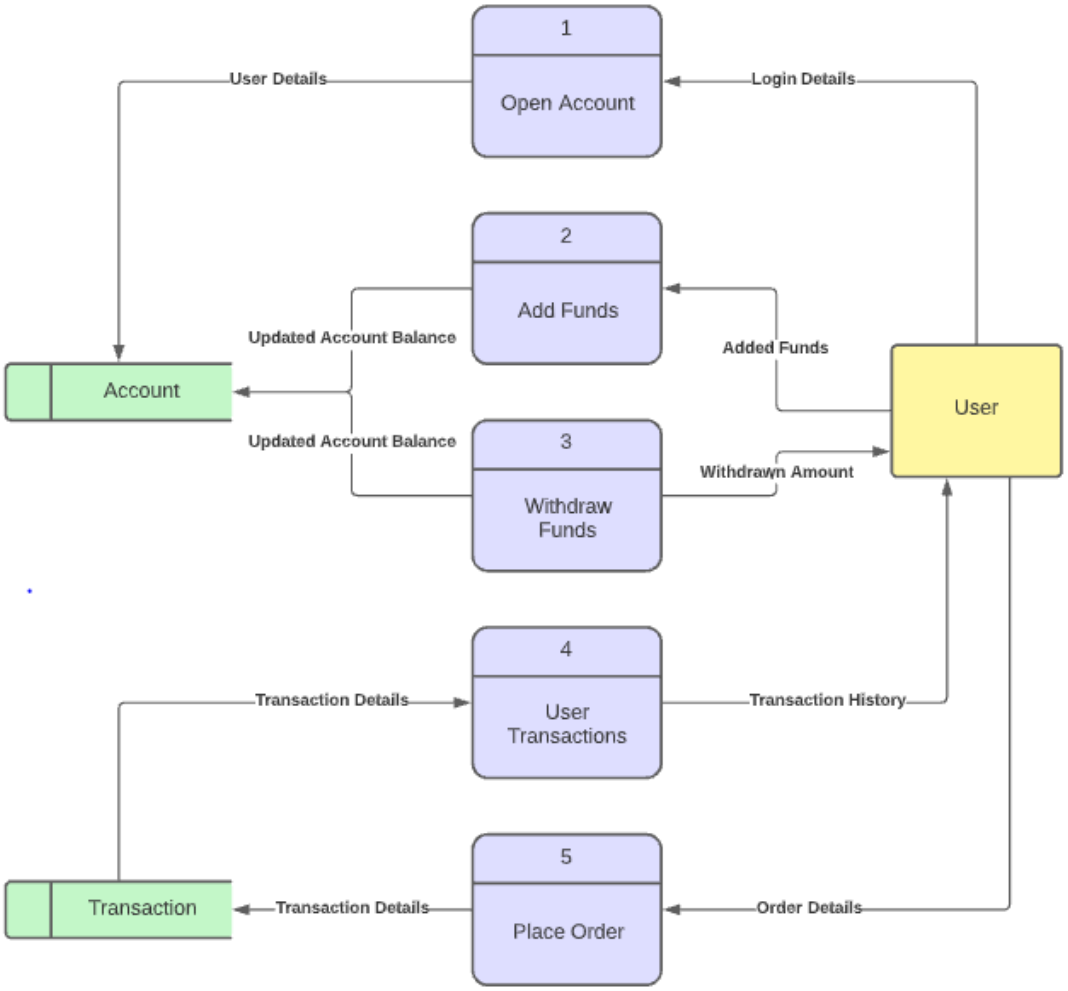


Fig. 9 - Data Flow Diagram

The external entity - in this case, the User - interacts with the:

- **Open Account** process by entering their login details,
- **User Transactions** process by receiving their transaction history,
- **Add Funds** process by adding an amount of their choosing,
- **Withdraw Funds** process by withdrawing an amount of their choosing and
- **Place Order** process by buying or selling stock.

There are two data stores for Accounts and Transactions respectively.

- The Open Account, Add Funds and Withdraw Funds processes interact with the Account data store by sending updated account balances and user details to it.
- The Transaction data store interacts with the User Transactions process by sending the transaction details from a user's order to it.
- The Place Order process interacts with the Transaction data store by sending information from a user's buy or sells transaction to it.

3.4.2 Use Case Diagrams

Sign Up

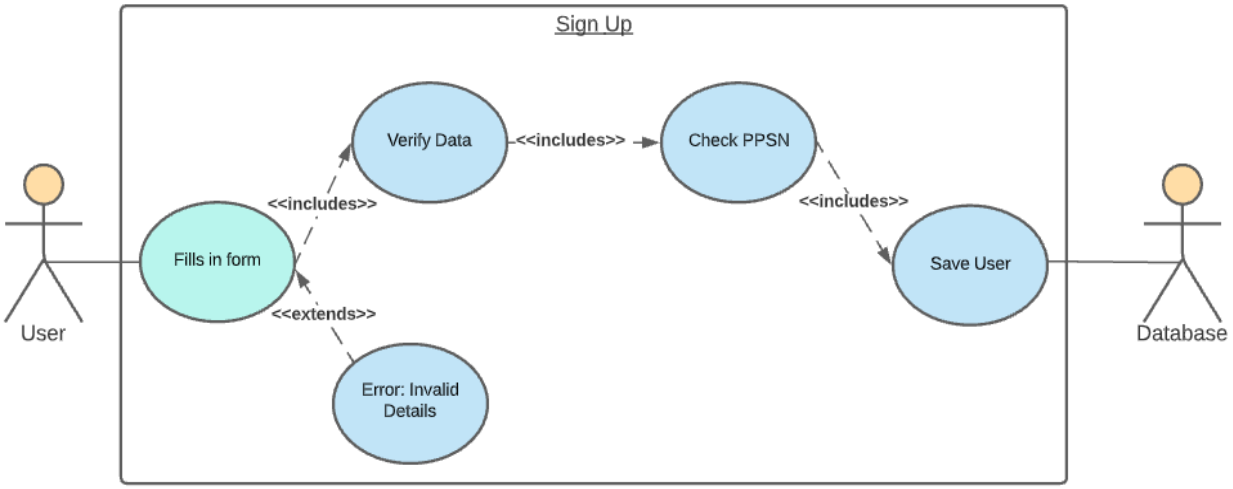


Fig. 10 - Use Case Diagram: Sign Up

The user must sign up to use the platform. This is done by entering their details into a form on the registration page, which includes their desired username and password, full name, date of birth, home address, email address, phone number and PPS number.

Once they fill in the form, the system will check that the information entered is valid by checking the syntax (e.g. the type and format) and validating the date of birth (if the user is under 18, they will be prevented from registering). The PPS number is checked separately by verifying the syntax - it must consist of seven numbers and a letter at the end. If all checks pass, the user's account is saved to the system's database, and they are redirected to the login page. If the information entered is incorrect, the user is presented with an error message notifying them of this, and prompting them to re-enter their details.

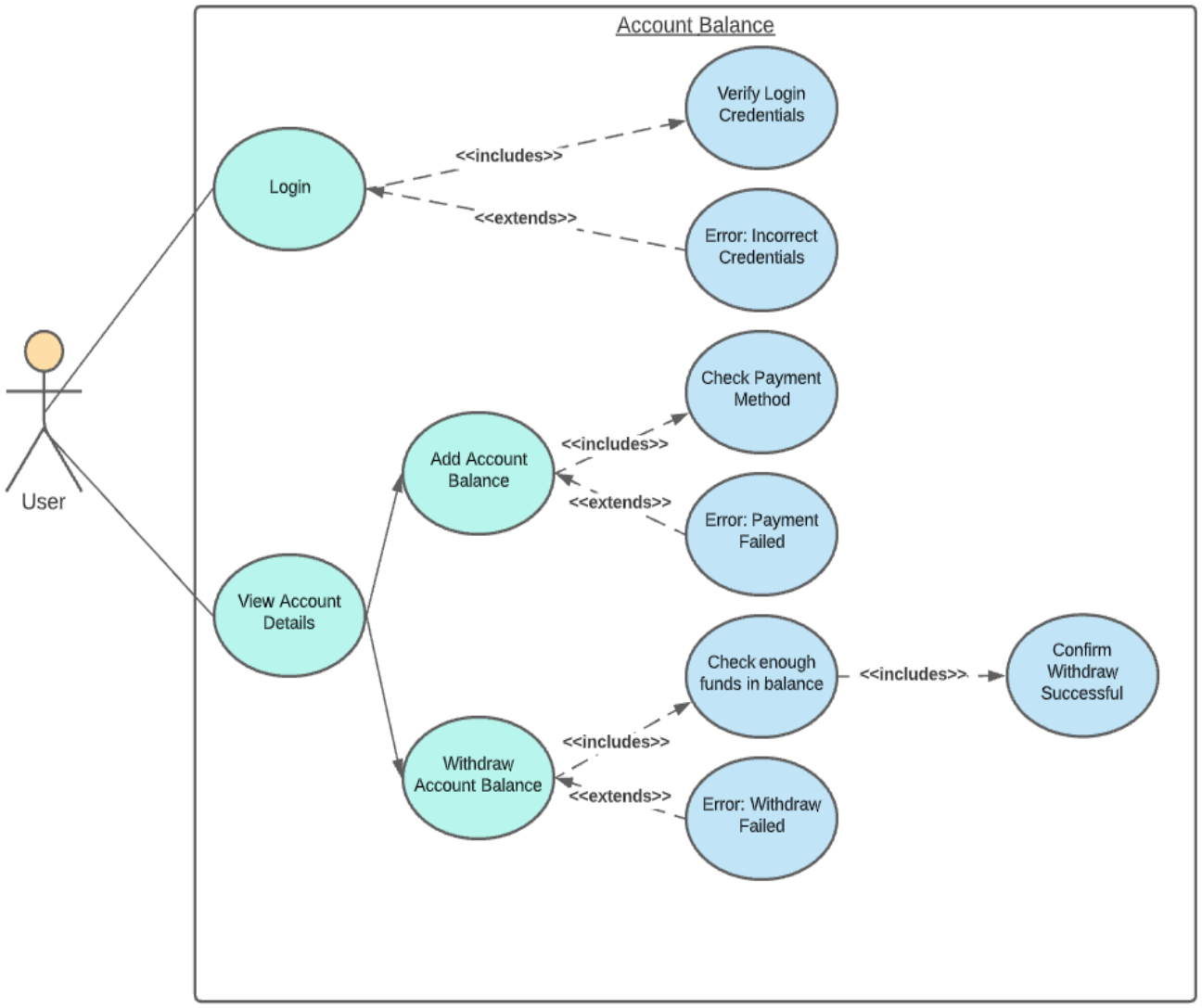


Fig. 11 - Use Case Diagram: Account Balance

The user needs an account balance to trade. As outlined above the user will first log in using their username and password. Once verified and signed in, they must then select account details from the navigation menu. The user can then view their details entered at registration and their current account balance, and add or withdraw funds from their account balance. The user must enter their credit/debit card information to add funds. The system will then verify that this information is correct and complete the transaction. If the information is incorrect, or the user does not have enough funds in their bank account, an error message will be displayed notifying the user of this. To withdraw funds, the user must have a sufficient amount in their account balance. The system will then verify that they do and complete the transaction. If they do not, an error message will be displayed notifying the user of this.

Fig. 12 - Use Case Diagram: Trade Stock

The above diagram outlines the process of a user buying or selling stock. The user must first log in using their username and password. Once verified and signed in, the user can then search through the list of stocks that are available on the platform. When they have found the desired stock, they can then select it to be brought to an individual stock's page. On this page, the user is given an array of information to help them make well-informed decisions.

This information consists of real-time time-series data, company information, news articles relating to that stock, sentiment analysis performed on the company's recent Twitter feed, the user's account balance and their stock balance.

The user can then buy stock, upon which the system will check that they have sufficient funds in their account balance. If they do, the trade is completed and leaderboard points are added to their account. The user can also sell a stock, upon which the system will check that they have sufficient stocks in their stock balance and, with successful sell transactions, leaderboard points will be added to their account. If their balance is too low when making these trades, the system will return an error message notifying the user their trade cannot be processed.

3.5 External software interfaces

External software interfaces refer to the software provided by third parties that are required by the application for specific purposes.

3.5.1 Euronext - Stock Exchanges

Euronext is a stock exchange based in Europe that caters to multiple European countries and over 1,900 companies by providing multiple types of trading assets such as equities, bonds and foreign currencies. It also offers a variety of order flows and "*A single entry point to multiple listing markets*" [51] for access to stock trading companies. Although this isn't currently implemented as part of the prototype, we plan to partner with them to execute a variety of orders, including market orders, limit orders and stop-loss orders.

3.5.2 Stripe - Payment System

Stripe is a financial services company that provides a payment processing platform designed to integrate with e-commerce websites. It is available in 35 countries, supports over 135 currencies and payment methods and caters to over 100,000 businesses [52]. Although this isn't currently implemented as part of the prototype, we plan to partner with them to process our customer's payments. Stripe charges a fee for each transaction processed by them, which will be charged to the user. Both Stripe and StockCompete will not be responsible for handling tax on the user's behalf, and users will be required to report any tax due for profit made on their trades.

3.6 Performance Requirements

These are a set of criteria that define how the application and the technology powering it must function. We have split the list of requirements into two categories: Speed and Storage.

3.6.1 Speed

Speed refers to how quickly data can be sent from the application's server to the user, and how quickly data and requests can be sent and processed from the user to the application. The speed requirements we've defined are:

- Third-Party API calls will return information at a high speed and without errors.
- Pages will have a load time of < 3 seconds.
- User transactions will be processed at a high speed (< 5 seconds).
- Graphs will be rendered at a high speed (<5 seconds).

3.6.2 Storage

Storage refers to the amount of data that can be stored and accessed from within the application's database. The storage requirements we've defined are:

- The server will have enough storage for up to a million registered accounts for the first three years of operation.
- The application must be constantly available and accessible by up to a million people at any given time for the first three years of operation.
- Data must be stored in a backup to prevent irretrievable data loss.

3.6.3 Security

Security refers to the measures taken to protect data used or stored by the application.
User information entered on signup will be saved in a Django model. Django provides encryption tools via a password storing system that incorporates a hashing algorithm, a set number of algorithm iterations, a random seed and a resulting hash to securely encrypt a string. We plan to implement this system to encrypt all user passwords provided to prevent anyone from viewing them, even in the event of a security breach [53]. Additionally, information entered on signup will be validated via several syntax checks to ensure no incorrect information is stored.

User payment information will be processed by Stripe, a third-party payment processing platform so that payment information does not have to be stored within the application's database. This will keep user details safe in the event of a security breach and prevent potential fraud if the payment details entered were incorrect or were connected to a bank account with insufficient funds for the transaction.

3.7 Technical Challenges

Technical Challenges refer to the difficulties that could be encountered when attempting to bring the application to market. Our primary concern for the future of the project is scalability. The technologies used to run the application must be capable of supporting an increasingly large number of concurrent users over time. For this, we would consider replacing Express.js as the frontend framework - as it is a lightweight framework not suited for large-scale applications - with React, a more feature-rich framework designed with scalability in mind.

More consideration will need to be given to the third-party services used as well. As concurrent usage of the application grows, we will need to look at upgrading the subscription level provided by Heroku to gain access to more powerful servers that could support this usage and ensure the application is constantly available and performs at a high speed. Additionally, we would need to look at upgrading our subscription with AlphaVantage to allow for a greater number of API calls.

Further optimization of the application may need to be carried out over its lifecycle to reduce the loading time of components, such as the graphs displayed on the Stock and User Dashboard pages and the news articles displayed on the homepage and Stock page.

The system used for rewarding points to users will need to be reevaluated to ensure it cannot be exploited and that all users are on an equal playing field regardless of the amount of money invested. To do this, we plan to base the number of points awarded for transactions based on the percentage increase of the stock's value when the user sells it compared to the value they bought it at, rather than the increase in euro.

To be made publicly available to customers, the application must adhere to GDPR principles, meaning that user's data can be deleted upon their request at any time, appropriate security measures are taken to protect data, and that personal data is collected for explicit and legitimate purposes [54].

3.8 Source Code

For our source code we used a range of techniques that were covered in Programming Fundamentals Ⅰ-Ⅲ, Developing Internet Applications, App Development and Fundamentals of Software Testing.

3.8.1 REST API Endpoint

```
class TransactionViewSet(viewsets.ModelViewSet):
    '''
    API Endpoint: Lists All Transactions
    '''
    queryset = Transaction.objects.all()
    serializer_class = TransactionSerializer
    permission_classes = [IsAuthenticated]

    def get_queryset(self):
        user = self.request.user
        if user.is_superuser:
            return Transaction.objects.all()
        else:
            return Transaction.objects.filter(user_id=user.id)
```

Fig. 13 - View to access Transactions

As taught to us in App Development, REST (Representational State Transfer) is an "API that conforms to the constraints of REST architectural style and allows for interaction with RESTful web services" [55]. This allows the transfer of data through HTTP requests and in this case using JSON formatted data.

This endpoint outlined above shows the extra layer of security added to the system when the frontend (NodeJS) must communicate to the backend (Django). As coded above, the request must come from an authenticated user to retrieve data from the model. If the user is not authenticated the REST API will return an HTTP 401 Unauthorised to the frontend. If the user is authenticated we then determine if this is a general user or an admin. If it is a general user we want to only retrieve their relevant information. If it is an admin user we want to return all users' data.

3.8.2 Backend Sell Transaction

```
# Sell Stock
if type.name == "Sell":
    try:
        # Check Stock Balance Exists
        stock_bal = StockBalance.objects.get(user_id=user,stock_id=stock_id.id)
    except:
        # User has no stock balance
        raise serializers.ValidationError("User doesn't own selected stock")

    if stock_bal.quantity >= Decimal(total/stock_price):
        # Create a new transaction object
        transaction = Transaction.objects.create(
            user_id=user,
            type=type,
            total=total
        )

        # Create a stock transaction
        stock_transaction = StockTransaction.objects.create(
            transaction_id=transaction,
            stock_id = stock_id,
            stock_price=stock_price,
        )

        # Calculate Commission fee to be applied
        commission = Decimal(total) * Decimal(.01)
        if commission < 0.75:
            commission = 0.75

        # Create a commission fee transaction
        Transaction.objects.create(
            user_id=user,
            type=TransactionType.objects.get(name="Commission"),
            total=commission
        )

        # Update Stock Balance
        stock_bal.total_purchase_value -= Decimal(stock_bal.average_value() * Decimal(stock_transaction.quantity()))
        stock_bal.quantity -= Decimal(stock_transaction.quantity())
        stock_bal.save()

        # Update Account Balance
        user.balance += Decimal(total) - Decimal(commission)
        user.save()

        # Update leaderboard score
        leaderboard = Leaderboard.objects.get(user_id=user)
        points = Point.objects.get(trans_type=type)
        leaderboard.points += points.points
        leaderboard.save()

        return transaction
    else:
        raise serializers.ValidationError("Stock Balance is too low")
```

Fig. 14 - Sell Stock Transaction

For our trading platform, users are able to complete an array of transactions. One of these transactions consists of selling a stock. Having learned various programming techniques from Programming Fundamentals Ⅰ-Ⅲ, Developing Internet Applications and App Development, we carried out the following tasks to complete a stock transaction:

- Checking Sufficient Stock Balance
- Creating a Sell Transaction Object
- Creating a Stock Transaction Object
- Calculate Commission
- Creating a Commission Transaction Object
- Adjusting Stock Balance
- Adjusting Account Balance
- Adding Leaderboard Points

This covered topics such as data types, python conditions and object instances.

3.8.3 Unit Testing

```
class GetCurrentUser(APITestCase):

    def setUp(self):
        self.user = APIUser.objects.create_user(
            username="TestCase",
            password="test",
            dob="2000-05-07",
            address="123 lane rd",
            email="test@mail.com")

    def authenticate(self):
        response = self.client.post(reverse('token_obtain_pair'), {'username': 'TestCase', 'password': 'test'})
        self.client.credentials(HTTP_AUTHORIZATION="Bearer " + response.data['access'])

    def test_get_current_user_authenticated(self):
        '''
        TEST: Successful Authorized to access current user info
        '''
        self.authenticate()
        response = self.client.get(reverse("apiuser-detail", kwargs={'pk': 1}))
        self.assertEqual(response.status_code, status.HTTP_200_OK)
        self.assertEqual(response.data['email'], "test@mail.com")

    def test_get_current_user_unauthenticated(self):
        '''
        TEST: Fail Unauthorized to access current user info
        '''
        self.client.force_authenticate(user=None)
        response = self.client.get(reverse("apiuser-detail", kwargs={'pk': 1}))

        self.assertEqual(response.status_code, status.HTTP_401_UNAUTHORIZED)
```

Fig. 15 - Unit Tests

As was taught to us in the Fundamentals of Software Testing module and also from experience in our internships, the main process employed by Agile Software Development teams is Test Driven Development. "*Test-Driven Development (TDD) is a software development approach in which test cases are developed to specify and validate what the code will do.*" [56].

The code highlighted above is a suite of API unit tests designed for the getCurrentUser API Endpoint. The test initialises a user object and then creates an authentication token for accessing this user object. We then test that we can successfully access this user object using this token and then run another test to verify that a false authentication wouldn't work.

3.9 Interface Rationale

The user interface incorporated by the application is designed to appeal to non-professional traders. We ensured the design was appropriate by establishing a list of criteria from feedback gained by our conducted survey and converting the criteria into a list of actionable user stories containing the application's requirements.

The criteria were:

- Ensure that the data presented was easy to interpret
- Ensure that gamification is incorporated into the UI in a unique method
- Ensure that the process of stock trading is easy to learn from the information provided through the UI

3.9.1 Personae

The term 'personae' refers to the target demographic we aim to cater to with our application. In our case, this would be non-professional traders of all ages.

3.9.2 Colours

We emphasised the use of bright colours in the design of pages to have a greater eye-catching effect on specific elements of the page, and thus draw the user's attention to them. We relied on the use of the colour wheel to pick colours for elements that complemented each other. For example, the navigation bar at the top of each page uses a blue-purple gradient, and shades of yellow were applied to many headings of content across the pages to match it.

3.9.3 Stories

A user story is "*an informal, general explanation of a software feature written from the perspective of the end user. Its purpose is to articulate how a software feature will provide value to the customer*" [57]. We identified three stories to develop our UIs as outlined below:

Story 1: *'As a non-professional trader, I want easy to understand information about my account so that I can learn how to trade.'*

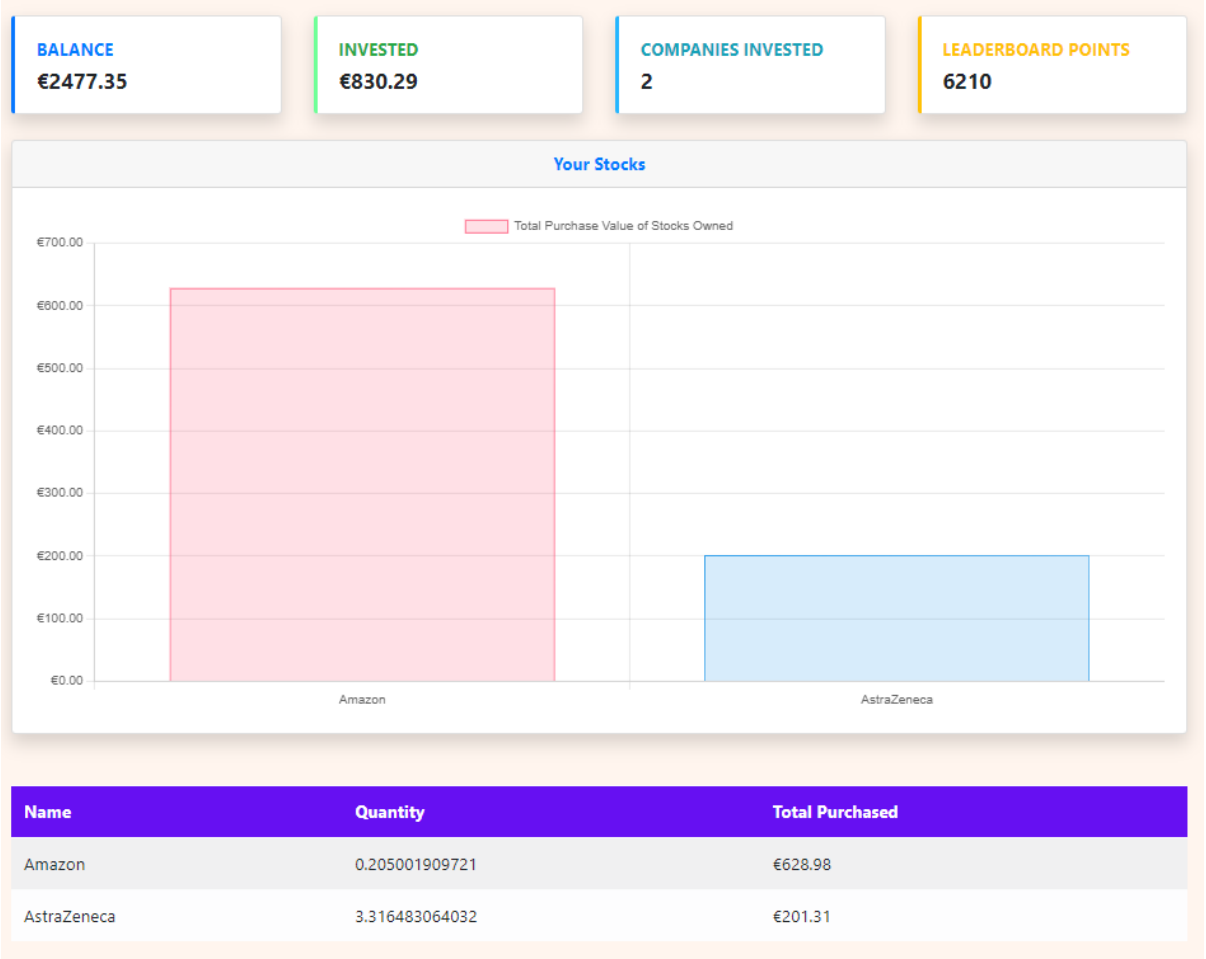


Fig. 16 - User Dashboard

For the User Dashboard, we aimed to provide all information relating to the user's current balances and investments in an easy to read and appealing format.

Figures relating to the user's current balance, money invested, companies invested in and leaderboard points are displayed at the top of the page, with bright colored headings and large text paired to each of them.

A graph displaying the stocks currently owned by the user, along with their total purchase value is displayed underneath, which aims to make the user's current holdings easier to interpret.

A table is displayed below the graph which also shows this information in a different format.

Story 2: *'As a non-professional trader, I want extensive stock/company information so that I can make more informed decisions for trading.'*

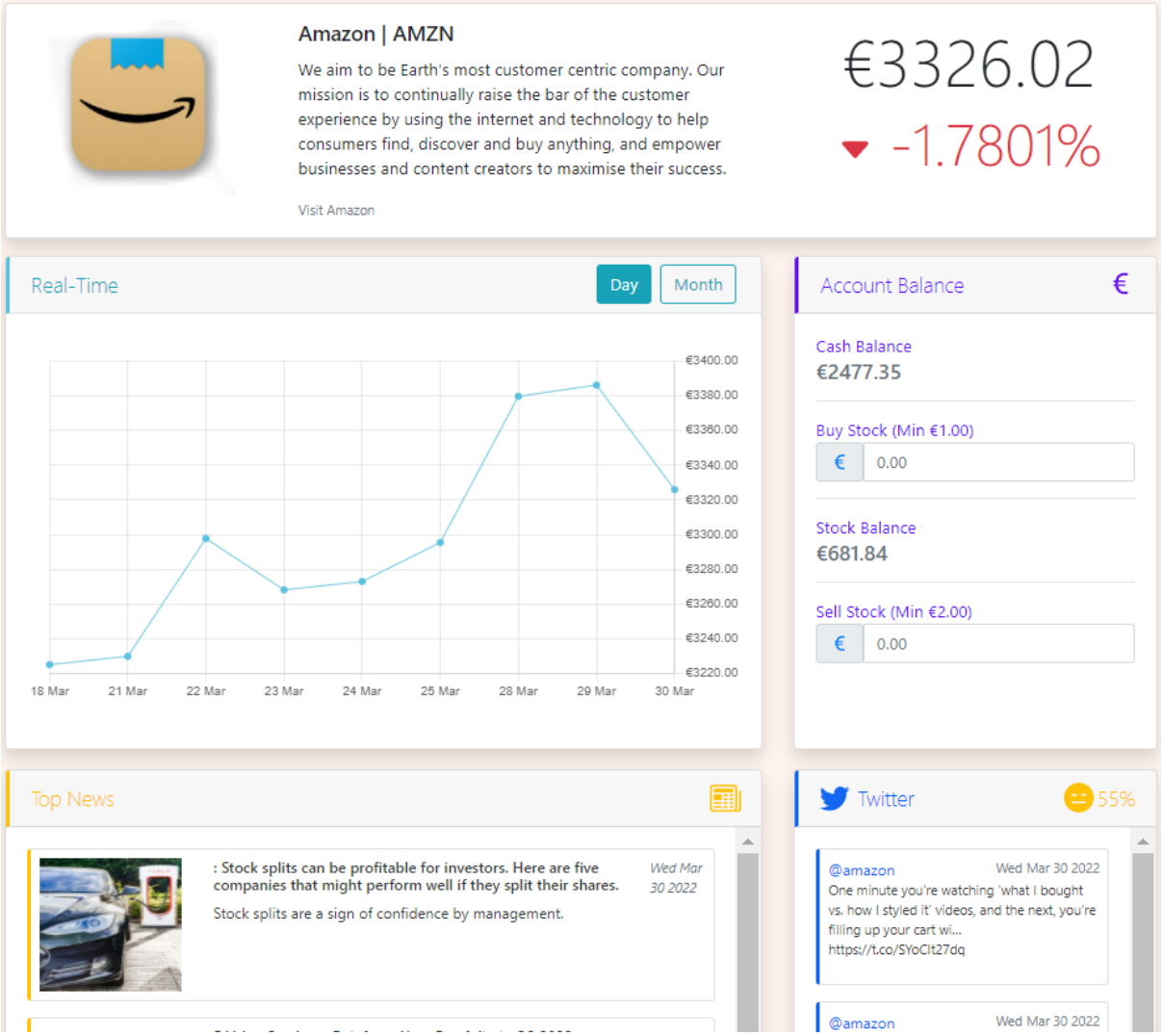


Figure 17 - Stock Page

For the individual stock page, we aimed to minimise the amount of complicated terminology used concerning the stock while still providing a useful amount of information to the user. The stock's name, current price and 24-hour percentage difference is shown at the top of the page in large writing to catch the user's attention, as this would be the most important information.

A graph showing the stock's price over time is displayed underneath, along with large, bright-coloured buttons highlighting the options for changing the graph's time range, and options for buying and selling the stock are displayed beside it.

These options also highlight the user's current cash and stock balances, so the user knows how much they can afford to buy or sell and thus make a more informed decision.

A Twitter feed for the stock and news articles relating to it are displayed underneath the graph, containing eye-catching headings paired with neatly formatted cards displaying the tweets and article summaries respectively, which make the contents easier to read.

Story 3: *'As a non-professional-trader, I want gamification elements so that the trading experience is more engaging.'*

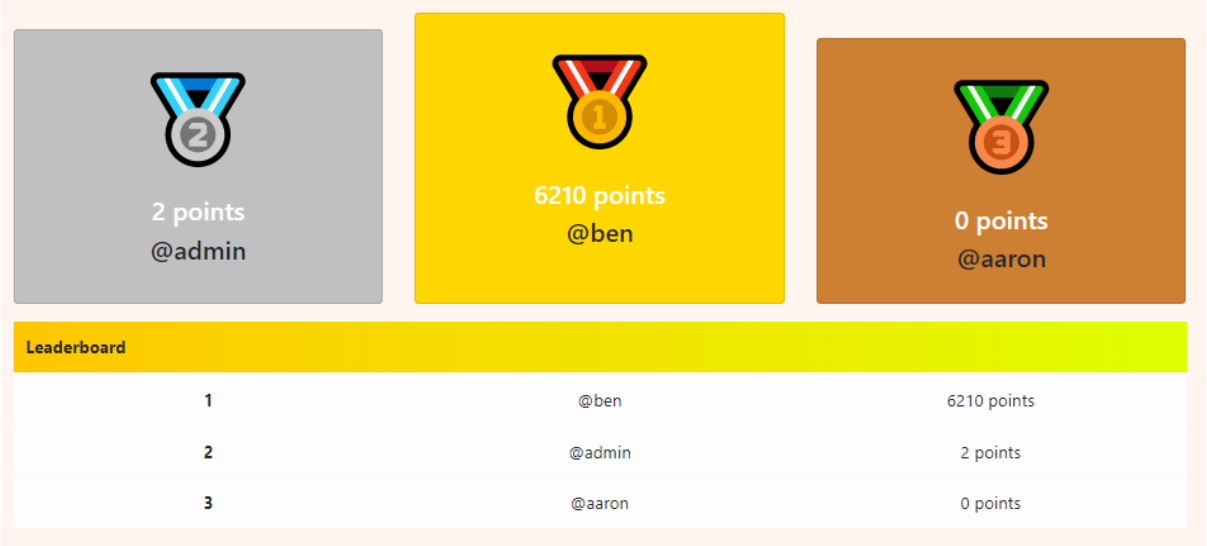


Figure 18 - Leaderboard page

For the Leaderboard page, we aimed to incorporate gamification in a unique method that would encourage users to be more proactive and competitive with their trading. For this, we designed a list displaying the ranking of each user based on their points, with the top three users highlighted in large, eye-catching icons at the top of the page.

References

1. Frankenfield, J. F. (2021, April 27). What Is a Commission in Financial Services? Investopedia. https://www.investopedia.com/terms/c/commission.asp
2. Kolakowski, M. K. (2021, October 22). SEC Considers Banning Payment for Order Flow. https://www.investopedia.com/sec-considers-banning-payment-for-order-flow-5199447
3. Mitchell, C. M. (2021, March 25). Understanding a Candlestick Chart. Investopedia. https://www.investopedia.com/trading/candlestick-charting-what-is-it/
4. UI vs. UX: What's the Difference? | UserTesting Blog. (2018, October 16). UserTesting. https://www.usertesting.com/blog/ui-vs-ux
5. Fitz-Walter, Z. F. (n.d.). What is Gamification? Gamify. Retrieved January 13, 2022, from https://www.gamify.com/what-is-gamification
6. Definition of dashboard | Dictionary.com. (n.d.). Dictionary. Retrieved January 13, 2022, from https://www.dictionary.com/browse/dashboard
7. How to do a market analysis for a business plan. (n.d.). The Business Plan Shop. Retrieved January 16, 2022, from https://www.thebusinessplanshop.com/blog/en/entry/market_analysis_for_business_plan
8. James, M. (2021, September 10). Target Market. Investopedia. https://www.investopedia.com/terms/t/target-market.asp

9. Darbyshire, M. (2021, April 9). Gen Z embraces investing — with signs of caution. Financial Times. https://www.ft.com/content/2ca31de6-2a2a-4e8f-8e0c-f5c313ce8fe3
10. Fitzgerald, M. (2020, May 12). Young investors pile into stocks, seeing “generational-buying moment” instead of risk. CNBC. https://www.cnbc.com/2020/05/12/young-investors-pile-into-stocks-seeing-generational-buying-moment-instead-of-risk.html
11. Tergesen, A. (2022, January 26). Older Investors Have a Lot of Money in Stocks. How to Check if It’s Too Much. Wall Street Journal. https://www.wsj.com/articles/older-investors-have-a-lot-of-money-in-stocks-how-to-check-if-its-too-much-11643215304
12. Simon, B. (2021, July 6). Breaking Down the Average Portfolio Mix by Investor Age. Personal Capital. https://www.personalcapital.com/blog/investing-markets/average-portfolio-mix-by-investor-age/
13. Roscoe, Howorth, P. C. (2008, July 17). Identification through technical analysis: A study of charting and UK non-professional investors. ScienceDirect. https://www.sciencedirect.com/science/article/pii/S0361368208000354?casa_token=30Djxk0iMAAAAAAA:Sj4f616vDaScZEIKClSncVinZApPhCLrVvXU_tu4PcNxCBM6OenBLLsRqZE0qDzUs-KPkPw
14. Soppitt, B. (2021, January 8). The Gamification Of Investing Brings Opportunity – And Risks. Forbes. https://www.forbes.com/sites/forbesbusinesscouncil/2021/01/08/the-gamification-of-investing-brings-opportunity--and-risks/?sh=43ca75ecc08e
15. Curry, D. (2022, January 11). Robinhood Revenue and Usage Statistics (2022). Business of Apps. https://www.businessofapps.com/data/robinhood-statistics/
16. Gamification Market Size, Share & Industry Analysis 2020–2027. (2019). Fortune Business Insights. https://www.fortunebusinessinsights.com/industry-reports/gamification-market-100632
17. Schram, M. (2021, January 12). Transforming Your Customer Forum into a Customer Community. https://www.cmnty.com/blog/transform-customer-forum/
18. Miller, G. (n.d.). 14 Benefits of Referral Marketing to Inspire Your Strategy. https://www.annexcloud.com/blog/14-benefits-referral-marketing-inspire-strategy/
19. Lomas, A. (2021, September 13). A 15-Step Guide on How to Build a Web App. Net Solutions. https://www.netsolutions.com/insights/how-to-build-a-web-app/#designing-ux-ui-stage
20. Law Insider. 2022. Commission Fee Definition | Law Insider. https://www.lawinsider.com/dictionary/commission-fee
21. Carey, T. W. (2021, December 3). Payment for Order Flow (PFOF). Investopedia. https://www.investopedia.com/terms/p/paymentoforderflow.asp
22. Robinhood Brokerage Accounts. (n.d.). Robinhood. Retrieved March 14, 2022, from https://robinhood.com/us/en/support/articles/robinhood-accounts/
23. Client Categorization. (n.d.). eToro. Retrieved March 14, 2022, from https://www.etoro.com/trading/professional/
24. Revolut. (n.d.). Revolut – Compare plans. Retrieved March 14, 2022, from https://www.revolut.com/our-pricing-plans
25. The issue of payment for order flow. (2021, August 4). FESE. https://www.fese.eu/blog/the-issue-of-payment-for-order-flow/

26. Bank of Ireland - Digital Application Centre. (n.d.). BOI Digital Applications. https://digital.bankofireland.com/business-lending/small-business-loan#application/page/1
27. Overview - Enterprise Ireland. (n.d.). Enterprise Ireland. https://www.enterprise-ireland.com/en/funding-supports/Company/HPSU-Funding/
28. Norrestad, F. (2021, October). Number of users of Robinhood from 2014 to 3rd quarter 2021. Statista. https://www.statista.com/statistics/822176/number-of-users-robinhood/
29. Statista. (2022b, February 28). eToro's registered users 2008–2021. https://www.statista.com/statistics/1262787/etoro-registered-users/
30. Google AdSense - General Information. (n.d.). Google AdSense. https://www.google.com/adsense/start/
31. Curry, D. (2022, January 19). eToro Revenue and Usage Statistics (2022). Business of Apps. https://www.businessofapps.com/data/etoro-statistics/
32. Butkus, V. (2019, February 20). Too Big to Scale – Optimal Scrum Team Size Guide. Toptal Product Blog. https://www.toptal.com/product-managers/agile/scrum-team-size
33. Scrum Master Salary in Ireland. (2022, March 1). PayScale. https://www.payscale.com/research/IE/Job=ScrumMaster/Salary
34. Senior Software Engineer Salary in Ireland. (2022, February 28). PayScale. https://www.payscale.com/research/IE/Job=Senior_Software_Engineer/Salary
35. Test / Quality Assurance (QA) Engineer (Computer Software) Salary in Ireland. (2022, February 28). PayScale. https://www.payscale.com/research/IE/Job=Test_%2F_Quality_Assurance_(QA)_Engineer_(Computer_Software)/Salary
36. Web Designer Salary in Ireland. (2021, July 31). PayScale. https://www.payscale.com/research/IE/Job=Web_Designer/Salary
37. Pigford, J. (2021, April 14). How to create a marketing budget for your startup. Baremetrics. https://baremetrics.com/academy/how-to-create-a-marketing-budget-for-your-startup
38. Marketing Manager Salary in Ireland. (2022, February 23). PayScale. https://www.payscale.com/research/IE/Job=Marketing_Manager/Salary
39. Marketing Executive Salary in Ireland. (2022, February 28). PayScale. https://www.payscale.com/research/IE/Job=Marketing_Executive/Salary
40. Lenovo ThinkBook 15 G2 ITL Core i7-1165 16GB 512GB SSD 15.6 Inch Windows 11 Pro Laptop. (n.d.). LaptopsDirect. https://www.laptopsdirect.ie/lenovo-thinkbook-15-g2-itl-core-i7-1165-16gb-512gb-ssd-15.6-inch-windows-11-20ve00rquk/version.asp?refsource=ldieadwords&mkwid=sM1AZ3pHg_dc&pcrid=565157537193&product=20VE00RQUK&pgrid=135963038531&ptaid=pla-1467160493711&channel=googlesearch&gclid=Cj0KCQjwz7uRBhDRARIsAFqjulm1EkQRzrJfgQzYbM4poQa32EIRAvbviFGnA4kJSDVVTV0WYidf7nAaAo0cEALw_wcB
41. Lenovo ThinkBook 15 Gen 2 Core i5-1135 8GB 256GB SSD 15.6 Inch Windows 11 Pro Laptop. (n.d.). LaptopsDirect. https://www.laptopsdirect.ie/lenovo-thinkbook-15-gen-2-core-i5-1135-8gb-256gb-ssd-15.6-inch-windows-11-p-20ve00rnuk/version.asp
42. Trust Primo 4-in-1 Home Office Set. (n.d.). Harvey Norman. https://www.harveynorman.ie/computing/computer-accessories/trust-primo-4-in-1-ho

me-office-set.html?gclid=Cj0KCQjwz7uRBhDRARIsAFqjulmD2alRZrBs37u7q77fhfp7kBRh9xoIImj5OxBXX-q-a9ofsXfHJIkaAiW4EALw_wcB
43. Samsung S65UA 34" Ultra WQHD HDR Curved Monitor. (n.d.). LaptopsDirect. https://www.laptopsdirect.ie/samsung-s65ua-34-ultra-wqhd-hdr-curved-monitor-ls34a650uxuxxu/version.asp
44. Pricing. (n.d.). Heroku. Retrieved January 13, 2022, from https://www.heroku.com/pricing
45. Coworking desks for 10 - €299/desk /month - Dublin. (n.d.). Rubberdesk. https://rubberdesk.ie/coworking/for-10-in-dublin-dublin-just-299-per-month/3960db65a07c0
46. Private offices for 30 - €9,000/month - Dublin 8, Dublin. (n.d.). Rubberdesk. https://rubberdesk.ie/office-space/for-30-in-dublin-8-d08-just-9000-per-month/5458cf29f5e24
47. Kulkarni, A. (2021, November 17). API vs REST API Simplified: 6 Critical Differences. Hevo. https://hevodata.com/learn/api-vs-rest-api/
48. Bastian, S. (2022, January 6). Learning Django: REST Framework and MVT Architecture. Medium. https://medium.com/geekculture/learning-django-rest-framework-and-mvt-architecture-1ca173163f1c
49. Databases | Django documentation. (n.d.). Django. https://docs.djangoproject.com/en/4.0/ref/databases/#sqlite-notes
50. SQLite vs PostgreSQL - Which database to use and why? (2018, August 30). TablePlus. https://tableplus.com/blog/2018/08/sqlite-vs-postgresql-which-database-to-use-and-why.html
51. Our Business. (n.d.). Euronext. https://www.euronext.com/en/about/our-business
52. Online payment processing for internet businesses. (n.d.). Stripe. https://stripe.com/ie
53. Password management in Django | Django documentation. (n.d.). Django. https://docs.djangoproject.com/en/4.0/topics/auth/passwords/
54. Principles of Data Protection. (n.d.). Data Protection Commission. https://www.dataprotection.ie/en/individuals/data-protection-basics/principles-data-protection
55. What is a REST API? (2020, May 8). Red Hat. https://www.redhat.com/en/topics/api/what-is-a-rest-api
56. Hamilton, T. (2022, February 26). What is Test-Driven Development (TDD)? Tutorial with Example. Guru99. https://www.guru99.com/test-driven-development.html
57. Atlassian. (n.d.). User Stories | Examples and Template. https://www.atlassian.com/agile/project-management/user-stories

Appendix

Figure 19 - Ben's CV

Profile

I am a student at DCU in my final year of an Honours degree in Enterprise Computing. The modules I have completed so far have allowed me to gain skills in numerous programming languages, web design, and project management. I also use websites such as HackerRank and Udemy outside of college to further develop my knowledge in these areas, and have completed the Microsoft Azure Fundamentals course, which allowed me to gain proficiency with cloud computing concepts and tools.

Outside of work and college my pastimes consist of playing sports, going on nature walks and reading. In 2021, I completed an internship that allowed me to gain experience in a professional environment working with an Agile development team. I also previously worked weekends as a Customer Care Agent, which allowed me to learn skills in customer service and teamwork.

Education

Enterprise Computing - DCU (2018 - 2022): Currently averaging 1.1 grade.

Third Year Project: Real-Time DCU Car Park Web Application (Grade: 90%)

- Created a web application containing real-time information for DCU car parks.
  - Imported JSON objects into Django models to retrieve live information on each car park
  - Designed web pages containing supplementary images and a Google Maps embed for each car park's location
- Gained experience working with Django Web Framework, APIs and webpage design.

Leaving Cert Results (2018): English (H4), Spanish (H4), Business (H4), Economics (H5), Maths (H6), Irish (O3)

Work Experience

eShopWorld (January 2021 – September 2021)
Software Development Intern

- Worked in frontend development including bug fixing, code reviewing and feature development
- Engaged in frequent collaboration with team members during daily stand-up meetings and bi-weekly sprints
- Presented work and incorporated feedback from other developers and product managers

OCS One Complete Solution (January 2019 – October 2020)
Customer Care Agent

- Assisted passengers with disabilities through Dublin Airport using equipment relevant to their individual needs e.g., wheelchair, aisle chair, etc.
- Liaised and worked professionally with all airport and airline personnel
- Provided customer service and answered questions of passengers when asked/required

Skills

Languages: Python, HTML/CSS, MySQL, Bash, JavaScript, TypeScript, R
Tools: Microsoft Office/Excel/Azure, phpMyAdmin, WireShark, Git
Frameworks: Django, Angular

Above is Ben Strickland's CV.

Figure 20 - Aaron's CV

Profile

Final year Enterprise Computing student at DCU expecting to graduate with a 1.1 in May. To date, I have averaged a 1.1 in my degree, gained experience in an agile team working on front-end development. This also gained me experience in multiple programming languages and concepts such as JavaScript, Python and Object Orientated Programming.

I'm a determined and knowledgeable software developer who can think outside the box. A student who enjoys working with all types of personalities and seeking the chance to architect new software applications and continue to improve my knowledge and skills.

IT Skills

Alongside having knowledge and experience with Agile and code reviews these are my main technical skills:

- **Languages**: Python, JavaScript, JAVA, SQL, CSS and HTML.
- **Frameworks:** Django and NodeJS.
- **Tools:** Git, Git Bash, Linux OS, SAP JIRA and Gerrit.

Education and Qualifications

Enterprise Computing - (2018 - Present)
Modules included: Web Design, Networks and Internet, Maths, Python Programming, Database Management, IT Architecture, Software Testing and App Development.

Third Year Project: A Real-Time DCU Car Park Web Application (Grade: 98%)

- Used Django Web Framework to develop a web application to help lecturers, students and other members of DCU for parking their cars. This Web Application allowed users to:
    - see available spaces in their selected car park and
    - get directions from their determined point of origin to a selected DCU car park.
- Gained web development skills and got experience working with Django, APIs and Google Cloud Platform.

Leaving Certificate - 2018
Maths H3, English H6, Irish O5, Physics H6, Biology H5, Geography H5, Music H

Work Experience

SAP - VT Star Program (Feb '21 – Sept '21) **Position:** Augmented Analytics BI Intern

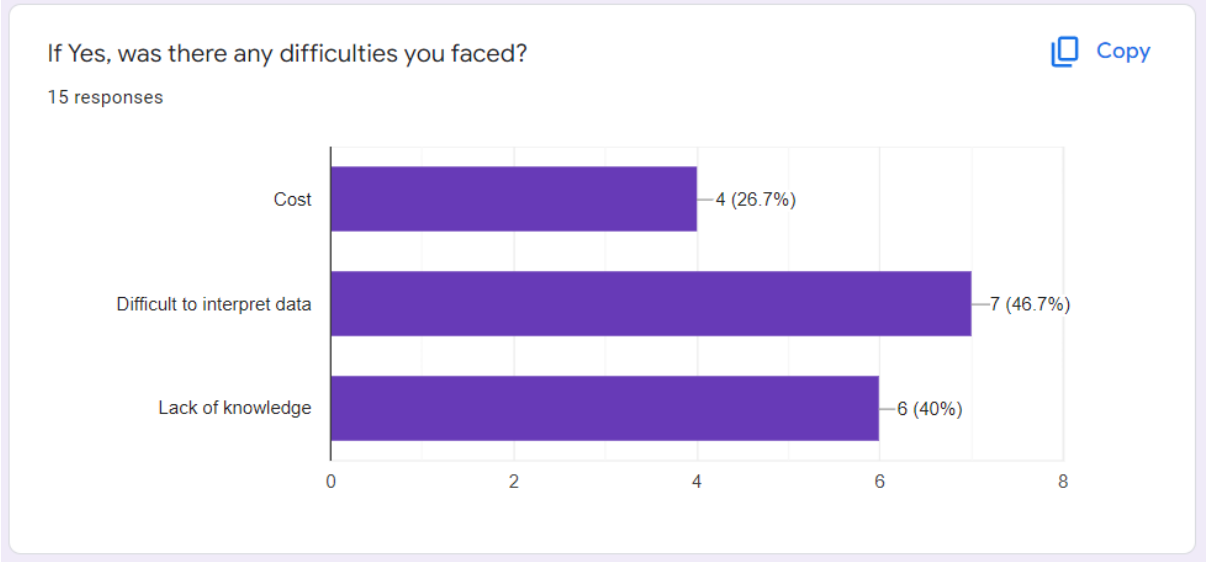
My role was based in SAP's Augmented Analytics BI Team located in Dublin, Ireland. As part of this 8-month placement, I maintained and enhanced SAP Analytics Cloud, in particular Smart Discovery.

Duties:

- Maintained and developed automation tests.
- Developed new, efficient and well-tested code for a variety of different feature fixes.
- Investigated, planned and developed new features based on user stories and included extensive accompanying unit tests.
- Worked on an agile team, attending daily stand-ups and working to two-week sprints.
- Carried out consistent code reviews on newly pushed code.

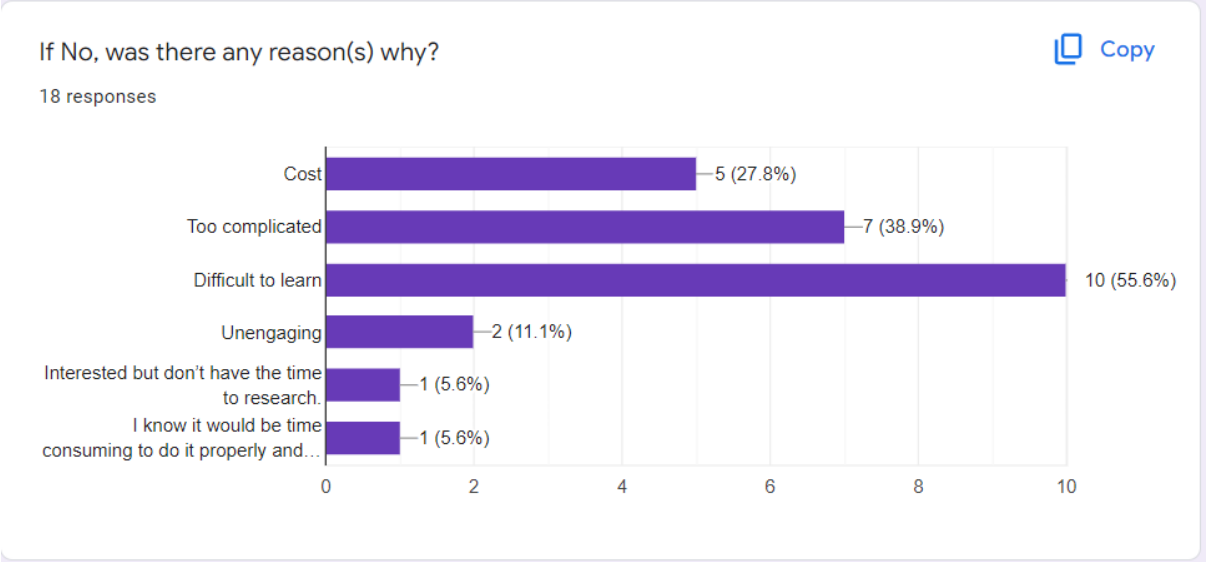
Above is Aaron Nolan's CV.

Figure 21 - Survey: If yes, difficulties faced



The survey responses to the question “If Yes, was there any difficulties you faced?”. The “Yes” in this case refers to if the user answered “Yes” to the question “Have you traded stock before?”.

Figure 22 - Survey: If no, reasons why



The survey responses to the question “If No, was there any reasons why?”. The “No” in this case refers to if the user answered “No” to the question “Have you traded stock before?”.

Figure 23 - Competitor Analysis (Part 1)

		Pricing Model						Features		
No.	Company Name	Hidden Fees	Commission Fees	Account Memberships	PFOF	Spread	Stop Loss	Gamification	Low Commission Fees (EU)	Simplified UI
1	Robinhood	-	-	✓	✓	-	-	✓	✓	✓
2	eToro	✓	-	✓	-	✓	-	-	✓	-
3	Degiro	✓	✓	-	-	-	-	-	✓	-
4	plus500	✓	✓	-	-	✓	✓	-	✓	-
5	naga	✓	✓	-	-	✓	-	-	✓	-
6	Interactive Brokers	-	✓	-	-	-	-	-	-	-
7	XTB	✓	✓	-	✓	-	-	-	-	-
8	AvaTrade	✓	-	-	-	✓	-	✓	✓	✓
9	Firstrade	-	-	-	✓	-	-	-	✓	-
10	Revolut	-	✓	✓	-	-	-	-	✓	✓

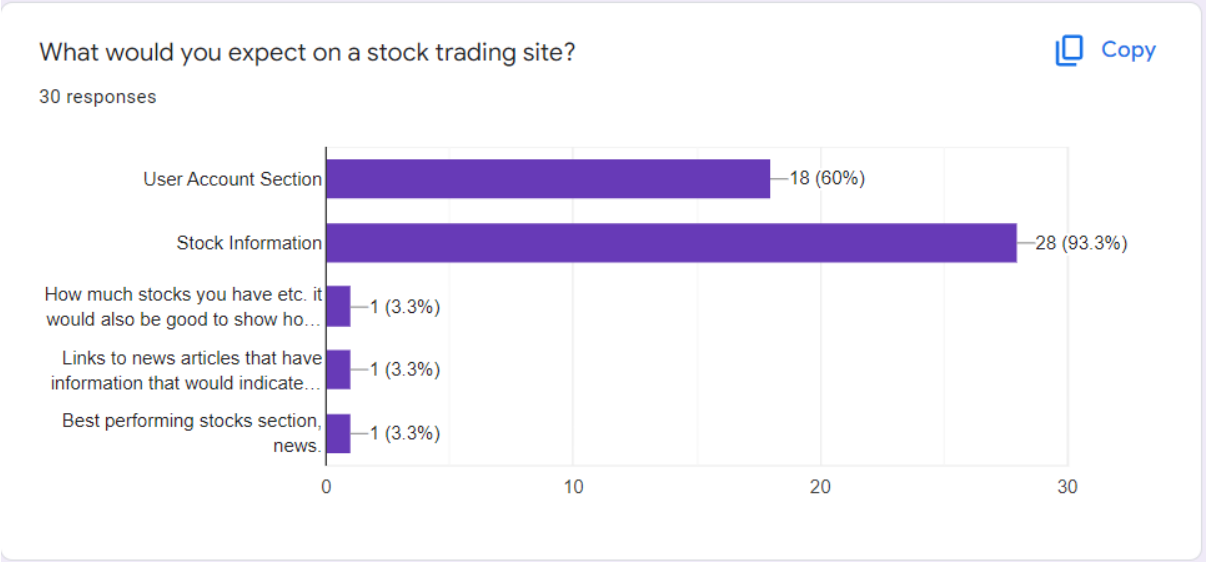
This spreadsheet details the pricing models and features used by competing stock brokers.

Figure 24 - Competitor Analysis (Part 2)

		Target Market		Stock Exchanges		Trading Type	
No.	Company Name	US	EU	US	EU	Stock	CFD
1	Robinhood	✓	-	✓	-	✓	-
2	eToro	-	✓	✓	✓	-	✓
3	Degiro	-	✓	✓	✓	✓	-
4	plus500	-	✓	✓	✓	✓	✓
5	naga	-	✓	-	✓	-	✓
6	Interactive Brokers (Advanced Traders)	✓	✓	-	✓	✓	✓
7	XTB	-	✓	✓	✓	-	✓
8	AvaTrade	-	✓	✓	✓	-	✓
9	Firstrade	✓	✓	✓	-	✓	✓
10	Revolut	-	✓	✓	-	✓	-

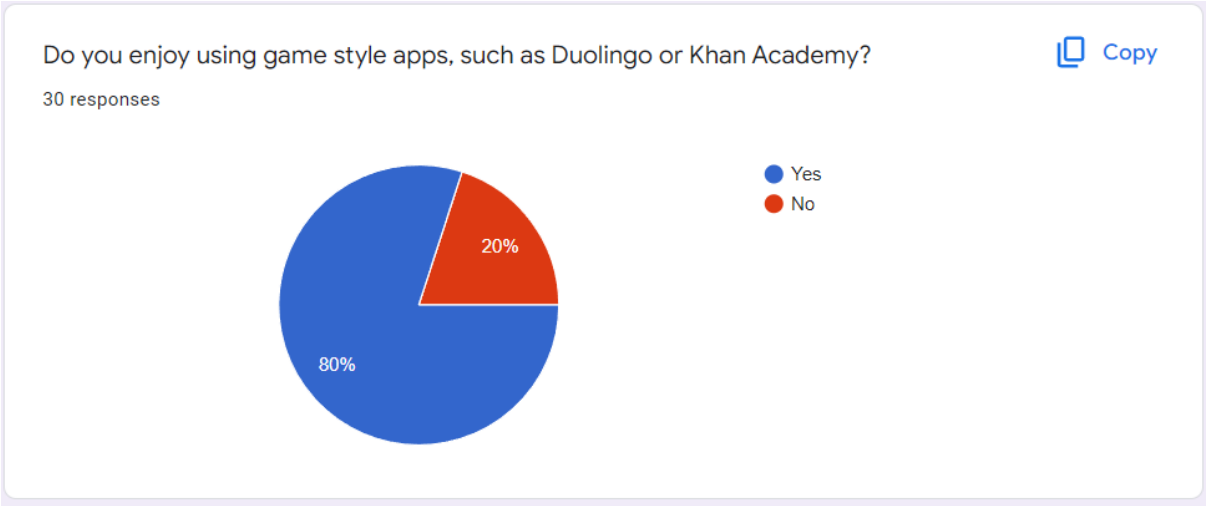
This spreadsheet details the target market, stock exchange regions and trading types used by competing stock brokers.

Figure 25 - Survey: What would you expect on a stock trading site?



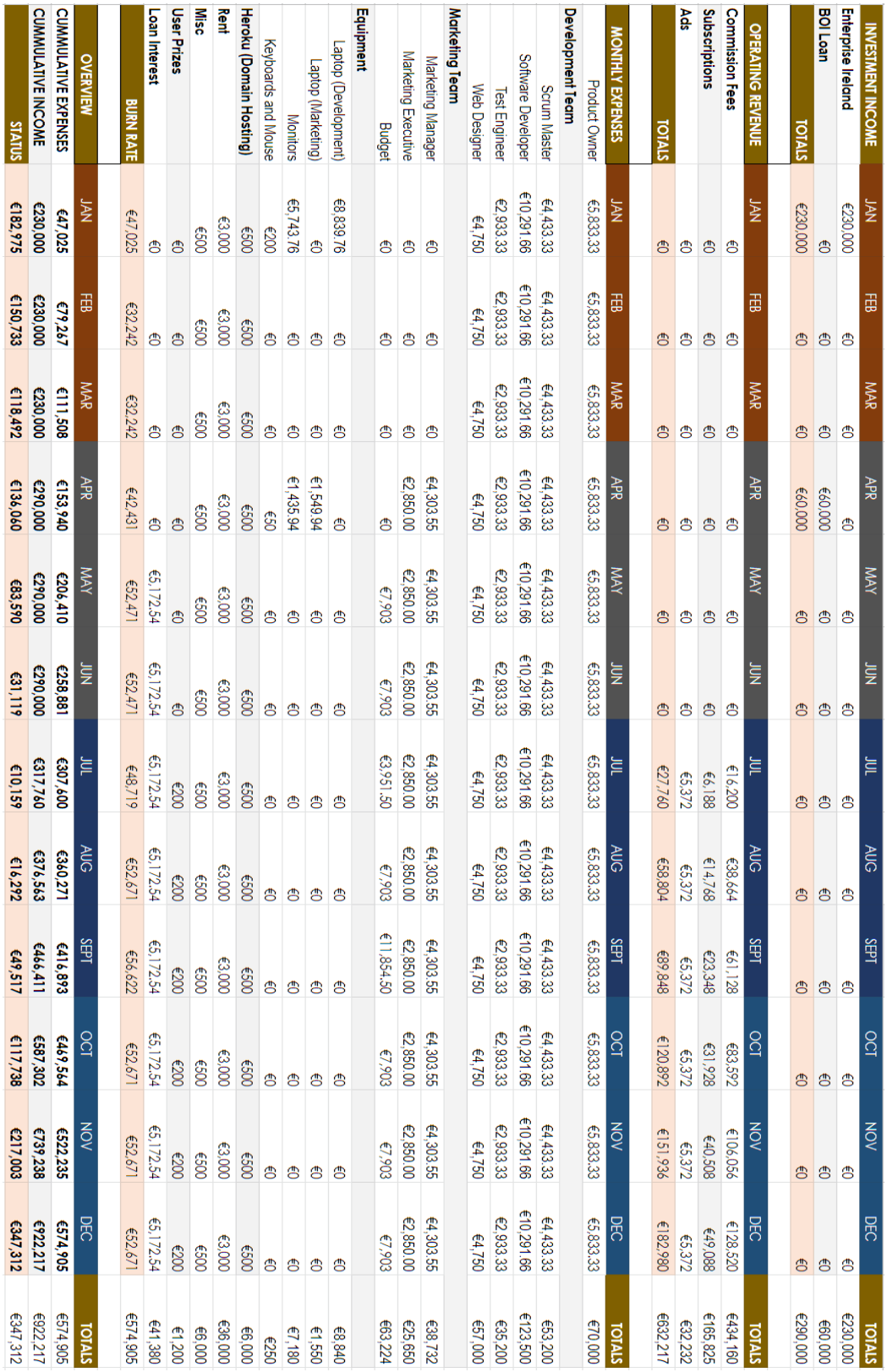
The survey responses to the question “What would you expect on a stock trading site?”.

Figure 26 - Survey: Interest in gamification



The survey responses to the question “Do you enjoy using game-style apps, such as Duolingo or Khan Academy?”.

Figure 27 - First Year Plan



INVESTMENT INCOME	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEPT	OCT	NOV	DEC	TOTALS
Enterprise Ireland	€230,000	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€230,000
BOI Loan	€0	€0	€0	€60,000	€0	€0	€0	€0	€0	€0	€0	€0	€60,000
TOTALS	€230,000	€0	€0	€60,000	€0	€0	€0	€0	€0	€0	€0	€0	€290,000
OPERATING REVENUE	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEPT	OCT	NOV	DEC	TOTALS
Commission Fees	€0	€0	€0	€0	€0	€0	€16,200	€38,664	€61,128	€83,592	€106,056	€128,520	€434,160
Subscriptions	€0	€0	€0	€0	€0	€0	€6,188	€14,768	€23,348	€31,928	€40,508	€49,088	€165,825
Ads	€0	€0	€0	€0	€0	€0	€5,372	€5,372	€5,372	€5,372	€5,372	€5,372	€32,232
TOTALS	€0	€0	€0	€0	€0	€0	€27,760	€58,804	€89,848	€120,892	€151,936	€182,980	€632,217
MONTHLY EXPENSES	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEPT	OCT	NOV	DEC	TOTALS
Product Owner	€5,833.33	€5,833.33	€5,833.33	€5,833.33	€5,833.33	€5,833.33	€5,833.33	€5,833.33	€5,833.33	€5,833.33	€5,833.33	€5,833.33	€70,000
Development Team													
Scrum Master	€4,433.33	€4,433.33	€4,433.33	€4,433.33	€4,433.33	€4,433.33	€4,433.33	€4,433.33	€4,433.33	€4,433.33	€4,433.33	€4,433.33	€53,200
Software Developer	€10,291.66	€10,291.66	€10,291.66	€10,291.66	€10,291.66	€10,291.66	€10,291.66	€10,291.66	€10,291.66	€10,291.66	€10,291.66	€10,291.66	€123,500
Test Engineer	€2,933.33	€2,933.33	€2,933.33	€2,933.33	€2,933.33	€2,933.33	€2,933.33	€2,933.33	€2,933.33	€2,933.33	€2,933.33	€2,933.33	€35,200
Web Designer	€4,750	€4,750	€4,750	€4,750	€4,750	€4,750	€4,750	€4,750	€4,750	€4,750	€4,750	€4,750	€57,000
Marketing Team													
Marketing Manager	€0	€0	€0	€4,303.55	€4,303.55	€4,303.55	€4,303.55	€4,303.55	€4,303.55	€4,303.55	€4,303.55	€4,303.55	€38,732
Marketing Executive	€0	€0	€0	€2,850.00	€2,850.00	€2,850.00	€2,850.00	€2,850.00	€2,850.00	€2,850.00	€2,850.00	€2,850.00	€25,650
Budget	€0	€0	€0	€0	€7,903	€7,903	€3,951.50	€7,903	€11,854.50	€7,903	€7,903	€7,903	€63,224
Equipment													
Laptop (Development)	€8,839.76	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€8,840
Laptop (Marketing)	€0	€0	€0	€1,549.94	€0	€0	€0	€0	€0	€0	€0	€0	€1,550
Monitors	€5,743.76	€0	€0	€1,435.94	€0	€0	€0	€0	€0	€0	€0	€0	€7,180
Keyboards and Mouse	€200	€0	€0	€50	€0	€0	€0	€0	€0	€0	€0	€0	€250
Heroku (Domain Hosting)	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€6,000
Rent	€3,000	€3,000	€3,000	€3,000	€3,000	€3,000	€3,000	€3,000	€3,000	€3,000	€3,000	€3,000	€36,000
Misc	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€6,000
User Prizes	€0	€0	€0	€0	€0	€0	€200	€200	€200	€200	€200	€200	€1,200
Loan Interest	€0	€0	€0	€0	€5,172.54	€5,172.54	€5,172.54	€5,172.54	€5,172.54	€5,172.54	€5,172.54	€5,172.54	€41,380
BURN RATE	€47,025	€32,242	€32,242	€42,431	€52,471	€52,471	€48,719	€52,671	€56,622	€52,671	€52,671	€52,671	€574,905
OVERVIEW	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEPT	OCT	NOV	DEC	TOTALS
CUMMULATIVE EXPENSES	€47,025	€79,267	€111,508	€153,940	€206,410	€258,881	€307,600	€360,271	€416,893	€469,564	€522,235	€574,905	€574,905
CUMMULATIVE INCOME	€230,000	€230,000	€230,000	€290,000	€290,000	€290,000	€317,760	€376,563	€466,411	€587,302	€739,238	€922,217	€922,217
STATUS	€182,975	€150,733	€118,492	€136,060	€83,590	€31,119	€10,159	€16,292	€49,517	€117,738	€217,003	€347,312	€347,312

This Excel sheet contains the details of the first year financial plan.

Figure 28 - Second Year Plan

INVESTMENT INCOME	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEPT	OCT	NOV	DEC	TOTALS
Enterprise Ireland	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0
BOI Loan	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0
TOTALS	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0
OPERATING REVENUE	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEPT	OCT	NOV	DEC	TOTALS
Commission Fees	€144,899	€161,279	€177,658	€194,037	€210,416	€226,796	€243,175	€259,554	€275,934	€292,313	€308,692	€325,080	€2,819,832
Subscriptions	€55,343	€61,599	€67,855	€74,111	€80,367	€86,623	€92,879	€99,135	€105,391	€111,647	€117,903	€124,163	€1,077,019
Ads	€6,843	€6,843	€6,843	€6,843	€6,843	€6,843	€6,843	€6,843	€6,843	€6,843	€6,843	€6,843	€82,116
TOTALS	€207,086	€229,721	€252,356	€274,992	€297,627	€320,262	€342,897	€365,533	€388,168	€410,803	€433,438	€456,086	€3,978,968
MONTHLY EXPENSES	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEPT	OCT	NOV	DEC	TOTALS
Product Owner	€30,000	€30,000	€30,000	€30,000	€30,000	€30,000	€30,000	€30,000	€30,000	€30,000	€30,000	€30,000	€360,000
Development Team													
Scrum Master	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€106,400
Software Developer	€20,583.33	€20,583.33	€20,583.33	€20,583.33	€20,583.33	€20,583.33	€20,583.33	€20,583.33	€20,583.33	€20,583.33	€20,583.33	€20,583.33	€247,000
Test Engineer	€5,866.66	€5,866.66	€5,866.66	€5,866.66	€5,866.66	€5,866.66	€5,866.66	€5,866.66	€5,866.66	€5,866.66	€5,866.66	€5,866.66	€70,400
Web Designer	€6,875	€6,875	€6,875	€6,875	€6,875	€6,875	€6,875	€6,875	€6,875	€6,875	€6,875	€6,875	€82,500
Marketing Team													
Marketing Manager	€3,586.25	€3,586.25	€3,586.25	€3,586.25	€3,586.25	€3,586.25	€3,586.25	€3,586.25	€3,586.25	€3,586.25	€3,586.25	€3,586.25	€43,035
Marketing Executive	€7,125.00	€7,125.00	€7,125.00	€7,125.00	€7,125.00	€7,125.00	€7,125.00	€7,125.00	€7,125.00	€7,125.00	€7,125.00	€7,125.00	€85,500
Budget	€92,222	€351,625	€150,000	€150,000	€150,000	€150,000	€150,000	€150,000	€150,000	€150,000	€150,000	€150,000	€1,943,846
Equipment													
Laptop (Development)	€5,524.85	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€5,525
Laptop (Marketing)	€1,549.94	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€1,550
Monitors	€5,025.79	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€5,026
Keyboard and Mous	€175	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€175
Heroku (Domain Hosting)	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€6,000
Rent	€9,000	€9,000	€9,000	€9,000	€9,000	€9,000	€9,000	€9,000	€9,000	€9,000	€9,000	€9,000	€108,000
Misc	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€6,000
User Prizes	€200	€200	€200	€200	€200	€200	€200	€200	€200	€200	€200	€200	€2,400
Loan Interest	€5,172.54	€5,172.54	€5,172.54	€5,172.54	€0	€0	€0	€0	€0	€0	€0	€0	€20,690
BURN RATE	€202,773	€449,900	€248,275	€248,275	€243,103	€243,103	€243,103	€243,103	€243,103	€243,103	€243,103	€243,103	€3,094,047
OVERVIEW	**JAN**	**FEB**	**MAR**	**APR**	**MAY**	**JUN**	**JUL**	**AUG**	**SEPT**	**OCT**	**NOV**	**DEC**	**TOTALS**
CUMMULATIVE EXPENSES	**€202,773**	**€652,673**	**€900,948**	**€1,149,224**	**€1,392,327**	**€1,635,429**	**€1,878,532**	**€2,121,635**	**€2,364,738**	**€2,607,841**	**€2,850,944**	**€3,094,047**	€3,094,047
CUMMULATIVE INCOME	**€554,397**	**€784,118**	**€1,036,475**	**€1,311,466**	**€1,609,093**	**€1,929,355**	**€2,272,252**	**€2,637,785**	**€3,025,953**	**€3,436,756**	**€3,870,194**	**€4,326,279**	€4,326,279
STATUS	**€351,625**	**€131,446**	**€135,526**	**€162,242**	**€216,766**	**€293,925**	**€393,720**	**€516,150**	**€661,214**	**€828,915**	**€1,019,250**	**€1,232,233**	€1,232,233

This Excel sheet contains the details of the second year financial plan.

Figure 29 - Third Year Plan

INVESTMENT INCOME	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEPT	OCT	NOV	DEC	TOTALS
Enterprise Ireland	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0
BOI Loan	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0
TOTALS	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0
OPERATING REVENUE	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEPT	OCT	NOV	DEC	TOTALS
Commission Fees	€367,200	€409,320	€451,440	€493,560	€535,680	€577,800	€619,920	€662,040	€704,160	€746,280	€788,400	€831,600	€7,187,400
Subscriptions	€140,250	€156,338	€172,425	€188,513	€204,600	€220,688	€236,775	€252,863	€268,950	€285,038	€301,125	€317,625	€2,745,188
Ads	€17,536	€17,536	€17,536	€17,536	€17,536	€17,536	€17,536	€17,536	€17,536	€17,536	€17,536	€17,536	€210,432
TOTALS	€524,986	€583,194	€641,401	€699,609	€757,816	€816,024	€874,231	€932,439	€990,646	€1,048,854	€1,107,061	€1,166,761	€10,143,020
MONTHLY EXPENSES	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEPT	OCT	NOV	DEC	TOTALS
Product Owner	€100,000	€100,000	€100,000	€100,000	€100,000	€100,000	€100,000	€100,000	€100,000	€100,000	€100,000	€100,000	€1,200,000
Development Team													
Scrum Master	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€8,866.66	€106,400
Software Developer	€30,875.00	€30,875.00	€30,875.00	€30,875.00	€30,875.00	€30,875.00	€30,875.00	€30,875.00	€30,875.00	€30,875.00	€30,875.00	€30,875.00	€370,500
Test Engineer	€11,733.33	€11,733.33	€11,733.33	€11,733.33	€11,733.33	€11,733.33	€11,733.33	€11,733.33	€11,733.33	€11,733.33	€11,733.33	€11,733.33	€140,800
Web Designer	€9,500	€9,500	€9,500	€9,500	€9,500	€9,500	€9,500	€9,500	€9,500	€9,500	€9,500	€9,500	€114,000
Marketing Team													
Marketing Manager	€7,172.50	€7,172.50	€7,172.50	€7,172.50	€7,172.50	€7,172.50	€7,172.50	€7,172.50	€7,172.50	€7,172.50	€7,172.50	€7,172.50	€86,070
Marketing Executive	€14,250.00	€14,250.00	€14,250.00	€14,250.00	€14,250.00	€14,250.00	€14,250.00	€14,250.00	€14,250.00	€14,250.00	€14,250.00	€14,250.00	€171,000
Budget	€200,000	€200,000	€200,000	€200,000	€200,000	€200,000	€200,000	€200,000	€200,000	€200,000	€200,000	€200,000	€2,400,000
Equipment													
Laptop (Development)	€14,364.61	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€14,365
Laptop (Marketing)	€4,649.82	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€4,650
Monitor	€6,461.73	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€6,462
Keyboard and Mouse	€950.00	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€950
Heroku (Domain Hosting)	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€6,000
Rent	€9,000	€9,000	€9,000	€9,000	€9,000	€9,000	€9,000	€9,000	€9,000	€9,000	€9,000	€9,000	€108,000
Misc	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€500	€6,000
User Prizes	€200	€200	€200	€200	€200	€200	€200	€200	€200	€200	€200	€200	€2,400
Loan Interest	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0	€0
BURN RATE	€319,024	€292,597	€292,597	€292,597	€292,597	€292,597	€292,597	€292,597	€292,597	€292,597	€292,597	€292,597	€3,537,596
OVERVIEW	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEPT	OCT	NOV	DEC	TOTALS
CUMMULATIVE EXPENSES	€319,024	€611,621	€904,219	€1,196,816	€1,489,414	€1,782,011	€2,074,609	€2,367,206	€2,659,804	€2,952,401	€3,244,999	€3,537,596	€3,537,596
CUMMULATIVE INCOME	€1,757,219	€2,340,412	€2,981,813	€3,681,422	€4,439,238	€5,255,261	€6,129,492	€7,061,931	€8,052,577	€9,101,430	€10,208,491	€11,375,252	€11,375,252
STATUS	€1,438,195	€1,728,791	€2,077,594	€2,484,605	€2,949,824	€3,473,250	€4,054,883	€4,694,724	€5,392,773	€6,149,029	€6,963,493	€7,837,656	€7,837,656

This Excel sheet contains the details of the third year financial plan.

Figure 30 - Number of Users (Competitors)

Company	Year 1	Year 2	Year 3
Robinhood [28]	500,000	1,000,000	2,000,000
eToro [29]	170,000	430,000	1,100,000

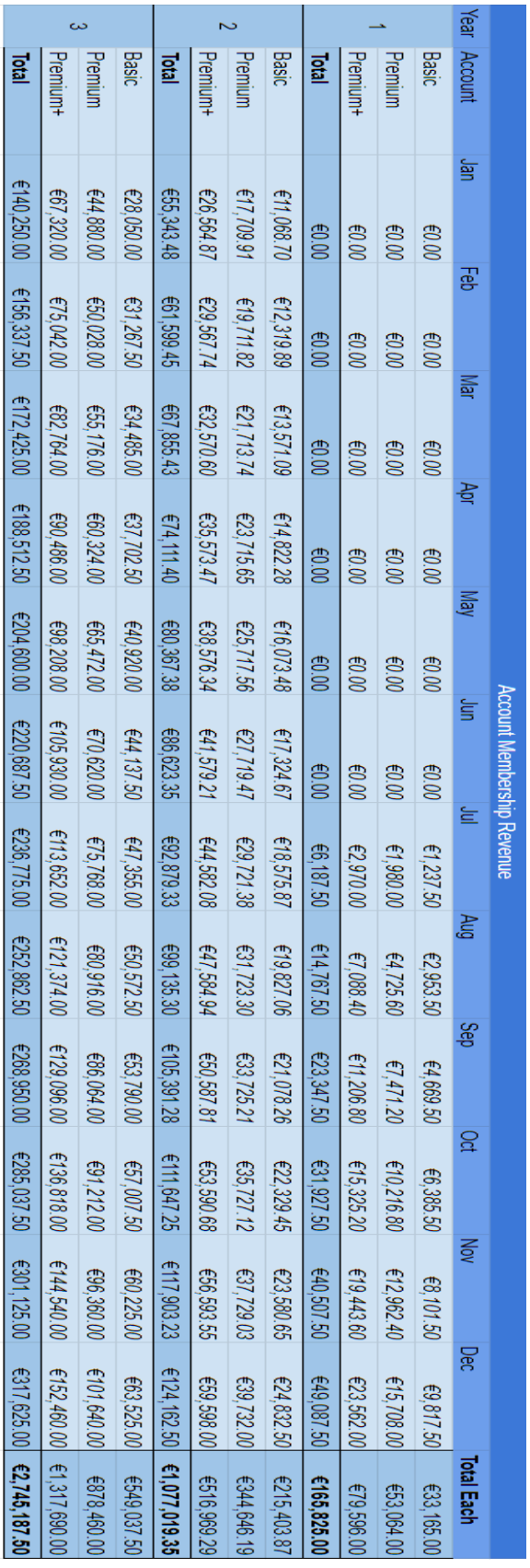
The table above contains the amount of users that Robinhood and eToro obtained within their respective first three years of operation.

Figure 31 - Number of Users (StockCompete)

Number of Users												
Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1	0	0	0	0	1000	8000	15000	35800	56600	77400	98200	119000
2	134166	149332	164498	179664	194830	209996	225162	240328	255494	270660	285826	301000
3	340000	379000	418000	457000	496000	535000	574000	613000	652000	691000	730000	770000

The table above shows the projected number of users for StockCompete within the first three years of operation.

Figure 32 - Paid Account Memberships



Account Membership Revenue													
Year	Account	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Total Each
1	Basic	€0.00	€0.00	€0.00	€0.00	€0.00	€0.00	€1,237.50	€2,953.50	€4,669.50	€6,385.50	€8,101.50	€9,817.50	€33,165.00
	Premium	€0.00	€0.00	€0.00	€0.00	€0.00	€0.00	€1,980.00	€4,725.60	€7,471.20	€10,216.80	€12,962.40	€15,708.00	€53,064.00
	Premium+	€0.00	€0.00	€0.00	€0.00	€0.00	€0.00	€2,970.00	€7,088.40	€11,206.80	€15,325.20	€19,443.60	€23,562.00	€79,596.00
	Total	€0.00	€0.00	€0.00	€0.00	€0.00	€0.00	€6,187.50	€14,767.50	€23,347.50	€31,927.50	€40,507.50	€49,087.50	**€165,825.00**
2	Basic	€11,068.70	€12,319.89	€13,571.09	€14,822.28	€16,073.48	€17,324.67	€18,575.87	€19,827.06	€21,078.26	€22,329.45	€23,580.65	€24,832.50	€215,403.87
	Premium	€17,709.91	€19,711.82	€21,713.74	€23,715.65	€25,717.56	€27,719.47	€29,721.38	€31,723.30	€33,725.21	€35,727.12	€37,729.03	€39,732.00	€344,646.19
	Premium+	€26,564.87	€29,567.74	€32,570.60	€35,573.47	€38,576.34	€41,579.21	€44,582.08	€47,584.94	€50,587.81	€53,590.68	€56,593.55	€59,598.00	€516,969.29
	Total	€55,343.48	€61,599.45	€67,855.43	€74,111.40	€80,367.38	€86,623.35	€92,879.33	€99,135.30	€105,391.28	€111,647.25	€117,903.23	€124,162.50	**€1,077,019.35**
3	Basic	€28,050.00	€31,267.50	€34,485.00	€37,702.50	€40,920.00	€44,137.50	€47,355.00	€50,572.50	€53,790.00	€57,007.50	€60,225.00	€63,525.00	€549,037.50
	Premium	€44,880.00	€50,028.00	€55,176.00	€60,324.00	€65,472.00	€70,620.00	€75,768.00	€80,916.00	€86,064.00	€91,212.00	€96,360.00	€101,640.00	€878,460.00
	Premium+	€67,320.00	€75,042.00	€82,764.00	€90,486.00	€98,208.00	€105,930.00	€113,652.00	€121,374.00	€129,096.00	€136,818.00	€144,540.00	€152,460.00	€1,317,690.00
	Total	€140,250.00	€156,337.50	€172,425.00	€188,512.50	€204,600.00	€220,687.50	€236,775.00	€252,862.50	€268,950.00	€285,037.50	€301,125.00	€317,625.00	**€2,745,187.50**

The table above shows the projected revenue generated by StockCompete by paid account memberships within the first three years of operation.

Figure 33 - Commission Fees

Commission Fee Revenue													
Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Total
1	€0.00	€0.00	€0.00	€0.00	€0.00	€0.00	€24,300.00	€57,996.00	€91,692.00	€125,388.00	€159,084.00	€192,780.00	€651,240.00
2	€217,348.92	€241,917.84	€266,486.76	€291,055.68	€315,624.60	€340,193.52	€364,762.44	€389,331.36	€413,900.28	€438,469.20	€463,038.12	€487,620.00	€4,229,748.72
3	€550,800.00	€613,980.00	€677,160.00	€740,340.00	€803,520.00	€866,700.00	€929,880.00	€993,060.00	€1,056,240.00	€1,119,420.00	€1,182,600.00	€1,247,400.00	€10,781,100.00

The table above shows the projected revenue from commission fees generated by StockCompete within the first three years of operation.