HAW
HAMBURG

BACHELORTHESIS

Eneida Koltraka

ESG investing: Developing a recommender system for portfolio construction

FAKULTÄT TECHNIK UND INFORMATIK
Department Informatik

Faculty of Computer Science and Engineering
Department Computer Science

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN HAMBURG
Hamburg University of Applied Sciences

Eneida Koltraka

ESG investing: Developing a recommender system for portfolio construction

Bachelor thesis submitted as part of the bachelor examination
in the course of studies *Bachelor of Science Business Information Systems*
at the Department of Computer Science
of the Faculty of Computer Science and Engineering

of the University of Applied Sciences Hamburg

Supervising examiner: Prof. Dr. Thomas Clemen
Second examiner: Prof. Dr. Marina Tropmann-Frick

Submitted on: 08.03.2022

Eneida Koltraka

Title of Thesis

ESG investing: Developing a recommender system for portfolio construction

Keywords

ESG investing, Recommender system, Modern Portfolio Theory, Sustainable advisor

Abstract

Increasing concerns around sustainability issues reveal a growing market demand for ESG investment portfolios, currently being met by a number of digital advisor platforms. Due to the competition situation between these ventures, specifics of their ESG investing strategies cannot be made available to the public. In this thesis, a knowledge-based recommender system is conceptualized and developed in the form of a web application, to offer an alternative to recommender systems' literature and existing advisory platforms, that emphasizes sustainability. The recommender system selects assets based on user preferences and uses Markowitz' Modern Portfolio Theory to construct a personalized portfolio with focus on its ESG profile. The project is build using solely open-source software and publicly available ESG data and does not restrict the choice of an alternative asset universe. Considering limitations that apply to the construction of the knowledge base, the developed ESG recommender system produces qualitative recommendations and has a high usability.

Eneida Koltraka

Thema der Arbeit

ESG investing: Developing a recommender system for portfolio construction

Stichworte

ESG investing, Recommender system, Modern Portfolio Theory, Sustainable advisor

Kurzzusammenfassung

Die zunehmende Aufmerksamkeit für Nachhaltigkeitsthemen zeigt eine wachsende Nachfrage nach ESG-Investmentportfolios, die derzeit von einer Reihe digitaler Advisor-Plattformen gedeckt wird. Aufgrund der Wettbewerbssituation zwischen diesen Unternehmen können die Einzelheiten ihrer ESG-Anlagestrategien der Öffentlichkeit nicht zugänglich gemacht werden. In dieser Arbeit wird ein wissensbasiertes Empfehlungssystem in Form einer Webanwendung konzipiert und entwickelt, um eine Alternative zu der Literatur über Empfehlungssysteme und bestehenden Beratungsplattformen zu bieten, die den Schwerpunkt auf Nachhaltigkeit legt. Das Empfehlungssystem wählt Vermögenswerte auf der Grundlage von Benutzerpräferenzen aus und verwendet die moderne Portfoliotheorie von Markowitz, um ein personalisiertes Portfolio mit Schwerpunkt auf dem ESG-Profil zusammenzustellen. Das Projekt nutzt ausschließlich Open-Source-Software und öffentlich verfügbare ESG-Daten und schränkt die Auswahl eines alternativen Anlageuniversums nicht ein. Trotz der Einschränkungen, die für den Aufbau der Wissensbasis gelten, liefert das entwickelte ESG-Empfehlungssystem qualitative Empfehlungen und weist eine hohe Benutzerfreundlichkeit auf.

Table of Contents

List of Figures

List of Tables

Listings

List of Abbreviations

API	Application Programming Interface
CAPM	Capital Asset Pricing Model
CSS	Cascading Style Sheets
CSV	Comma-separated Values
DB	Database
ESG	Environmental, Social, Governance
ETF	Exchange-Traded Fund
ETL	Extraction, Transformation, Loading
GMVP	Global Minimum Variance Portfolio
GUI	Graphical User Interface
HTTP	Hypertext Transfer Protocol
IDE	Integrated Development Environment

MPT	Modern Portfolio Theory
MVC	Model-View-Controller
MVT	Model-View-Template
OECD	Organisation for Economic Co-operation and Development
QUS	Quality User Stories
REST	Representational State Transfer
RS	Recommender System
SRI	Socially Responsible Investing
SQL	Structured Query Language
U. S.	United States
UNGC	United Nations Global Compact
URL	Uniform Resource Locator
US	User Story
USD	United States Dollar

1 Introduction

1.1 Motivation

Sustainability, much like climate change, gender and social inequality issues, and several hot subject matters in today's society, is surrounded by a fuzziness that often acts as a barrier towards understanding and engaging with the topic itself. While on a high level the matter draws increased attention daily, it is challenging to break the topic down, to offer feasible solutions on an operational level (Schanzenbach & Sitkoff, 2020).

Dealing with a multifaceted topic like sustainability requires a general refinement of the area of study, to enable efficient problem-solving. Therefore, this thesis focuses on the incorporation of sustainability in the financial domain, particularly in financial advisory products. These advisors aim to offer a customized customer experience, which can be assisted by one widely researched field of Computer Science, that of recommender systems (Gigli, et al., 2017).

Recommender systems (RS) have been successfully used in literature to offer personalized financial products and recommend investment portfolios (Zibriczky, 2016). Still, research shows a gap in including the sustainability perspective in the recommendations. Though platforms that consider the investors' concern on environmental (E), social (S) and governance (G) performance of the assets exist in the market, there are currently no standard frameworks that predefine how these concerns should be mirrored in the recommended products, and given these ventures' profit-driven nature, transparency and credibility issues may arise. Moreover, the usage of these platforms is generally not free of cost, contributing to the accessibility challenge.

As a student of Business and Computer Science, with a deep personal interest in sustainability and environmental issues, I take advantage of the set of skills acquired from the curriculum of the Business and Computer Science programme to design and develop a recommender system for personalized investment portfolio recommendations with emphasis on sustainability. This recommender system is build using solely open-source software and publicly available data and should serve as a foundation for a transparent, highly adjustable, and free of charge recommendation tool for sustainable investment portfolios.

1.2 Goal setting

Technically speaking, the main goal of this thesis is to close the literature gap by developing a software application based on the design of a recommender system for investment portfolio construction, given specific user requirements and the assets' ESG performance.

Furthermore, this thesis aims to provide a step-by-step guide, from choosing the right recommender system for the goal, to developing the application and designing a friendly graphic user interface to showcase results. Simultaneously, it aims to offer a practical solution to the transparency, credibility, and accessibility issues of existing platforms by using best-practice strategies to incorporate ESG in the investment strategy, by providing an explanation of the process of recommendation and by using only available open-source software and data to develop.

Lastly, it seeks to inspire further research in the topics of "Sustainability and Computer Science" and proposes extensive features and functionalities for further development of the application.

1.3 Structure

This work is structured in six chapters, each serving a specific purpose in achieving the aforementioned goals.

Chapter one introduces the motivation of the author and states the objectives of the work.

Chapter two provides context for ESG investing and performs a literature and market search for existing technologies and ESG portfolio integration strategies. Respectively section 2.1 gives a definition of the term and highlights the increasing importance of the topic in the last years. Section 2.2 explores the state-of-the-art digital investment platforms and their approaches to ESG investing. Furthermore, it sheds light on the background process of ESG rating conducted by rating agencies and chooses one provider as source for the ESG data. Section 2.3 reviews strategies that incorporate ESG in different investment portfolios in literature and finally decides on one strategy to adapt in the recommender system workflow. Section 2.4 gives a summary of chapter two.

Chapter three gives an overview of recommender systems and their main advantages and limitations, with the purpose of choosing the right recommender system for portfolio construction. Section 3.1 draws a picture of recommender systems and classifies them according to their filtering methods. Section 3.2 reviews the main applications of RS in finance and particularly in investment portfolio construction. In addition, it further explores the literature gap in using RS for ESG conform portfolio construction. Section 3.3 introduces the conceptual design of the RS for ESG investment portfolio recommendation. Section 3.4 gives a summary of chapter three.

Chapter four describes the practical implementation of the recommender system. Section 4.1 performs a requirements analysis following agile software development frameworks. Section 4.2 presents the system design and architectural decisions. Section 4.3 explains the implementation techniques in more detail and emphasizes essential parts of the code. Section 4.4 introduces the workflow. Section 4.5 discusses some of the quality assurance methods used in the development. Section 4.6 gives a summary of chapter four.

Chapter five concludes the research and the development by evaluating the quality of the recommendation in section 5.1 and the utility of the digital advisor in section 5.2. Section 5.3 gives a summary of chapter five.

Chapter six reiterates the key findings of the research in section 6.1 and explores the limitations and further research possibilities in section 6.2.

2 ESG investing

2.1 Overview

2.1.1 Definition

Whilst a strict definition of the term “ESG investing” is generally lacking, it is roughly described as “an umbrella term that encompasses any investment strategy that emphasizes a firm’s governance structure or the environmental or social impacts of the firm’s products or practices” (Schanzenbach & Sitkoff, 2020, p. 42). Accordingly, ESG stands for environmental (E), social (S) and governance (G). In this work, terms “ESG investing” and “socially responsible investing” (SRI) are used interchangeably.

2.1.2 Social origins

ESG investing originates in the socially responsible investing movement of the late 1970s, early 1980s, as a part of a divestment campaign aimed to cut financial ties of US companies involved with South Africa’s apartheid regime. NGO’s and universities were pressing to leave corporates engaged in controversial business practices powerless, by choosing to sell their stocks, bonds and investment funds (Schanzenbach & Sitkoff, 2020; Hauck, et al., 1983; Bosson, 2016; Eccles & Stroehle, 2018) . At this point in time, ESG investing was motivated by moral and ethical issues, rather than financial profit (Schanzenbach & Sitkoff, 2020). The concept ‘ESG’ was then introduced much later, in 2004, by the United Nations Global Compact (UNGC) Report on “Who Cares Wins: Connecting Financial Markets to a Changing World” and the 2005 “Freshfields Report” (Freshfields Bruckhaus Deringer, 2005) of the UN Environmental Program’s Finance Initiative (UNEP-FI), quickly becoming a key concept in the increasing interest of the capital market in sustainable finance (Eccles & Stroehle, 2018).

2.1.3 Relevance

As reported for the Reuters by Kerber and Jessop (2021), the booming interest in ESG investing reached a high in 2021, as extreme weather episodes became more frequent and social justice issues became more evident, in the light of events such as the Black Lives Matters protest. Moreover, the COVID-19 pandemic has raised awareness of the risks the financial system faces from social crises and existential threats like infectious diseases and anthropogenic climate change, leading to a grown momentum for ESG related financial decisions (Adams & Abhayawansa, 2021).

According to data from Refinitv Lipper, a record $649 billion was invested in 2021 in ESG-focused funds worldwide, accounting for 10% of the worldwide fund assets (Kerber & Jessop, 2021). A 2021 survey from Statista (Norrestad, 2022) shows that 82% of professional investors globally, 8% more than the previous year, are planning to increase their allocation of socially responsible investments in 2022.

The boost to ESG investing is also awarded to the millennial (born 1981-1996) investor and the generational wealth-transfer from baby-boomer parents (Ruggie & Middleton, 2019). A recent survey from the Morgan Stanely Institute for Sustainable Investing (2021) conveys that millennial investor interest in sustainable investing rose to an all-time high of 99% in 2021, especially after the pandemic. Osborne (2017) highlights how the affinity to technology in the younger generations is contributing to reshaping the economy, by demanding for more personalized experiences and products. This translates to using the technology to access information that helps choose investment strategies aligned with their values and beliefs.

A review of more than 1000 research papers exploring the linkage between ESG and financial performance since year 2015, conducted by Whelan et. al (2021) revealed that in portfolio construction, ESG strategies seem to (1) generate market rate or excess returns compared to conventional investment strategies, especially for long-term investors, and (2) provide downside protection during economic or social crisis. Also, there was a notably low amount of evidence showing a negative correlation between ESG and financial performance (Whelan, et al., 2021).

2.2 State-of-the-art ESG investing technology

2.2.1 Existing Platforms

The gap in the investment landscape for value based and ESG investing has already been noticed by several digital investment advisor platforms which provide options for investors who want to align their finances with their personal values, all while maintaining the ability to generate profit from their investment portfolios. The increase in demand from millennial, Gen X and Baby Boomer investors, have enabled automated advisors to gain a significant share in the SRI marketplace. Such platforms include Betterment, Wealthfront, Motif, Earthfolio, Wealthsimple and many more (Salampasis, 2017).

To build portfolios with a good ESG profile, many so-called robo-advisors or automated advisor platforms, select exchange traded funds (ETFs) that specialize in ESG investments and optimize portfolios using those ETFs. ETFs are investment securities that track a particular index, sector, commodity, or other assets and can be traded like regular stocks. Advantages of buying ETFs include low expense ratios and fewer broker commissions than stocks (Chen, 2022). Moreover, they are considered one of the easiest ways to build a diversified portfolio because they are composed of multiple underlying assets (O'Connell & Curry, 2021).

Each of these platforms follows a different strategy for ESG portfolio construction (Hayes, 2021). A short, illustrative comparison of four platforms and their ESG strategies based on Investopedia article (Hayes, 2021) and on the respective websites of Betterment (Betterment, n/d), Wealthfront (Wealthfront, n/d), Wealthsimple (Wealthsimple, n/d) and Earthfolio (Earthfolio, n/d) follows in Table 1. Betterment and Wealthfront rank the highest in several articles evaluating robo-advisors (Gravier, 2022; Tepper & Schmidt, 2022; Sette, 2022), while Wealthsimple and Earthfolio are chosen for their individual approach to ESG investing. The digital platforms are compared on attributes such as area of availability, minimum opening investment value, ESG investing strategy and the ESG ETFs they include in their portfolios. A list of the ESG ETFs can be found in Appendix G.1.

Though most research comes from the United States (U. S.) market, there is a market for personalized portfolio recommenders and for sustainable investing in Europe, too. Forum

Nachhaltige Geldanlagen (FNG) (english: Forum on Sustainable Investments) notices an increasing volume on German private investors sustainable investments. Same research shows that private investment in sustainable funds in Germany more than doubled from 2019 in 2020, reaching €39.8 billion, scoring a growth of 117%. This trend is followed by other German speaking countries such as Austria with 78% growth rate and Switzerland with 72% (FNG, 2021). Another study from Au et. al (2021) offers empirical insight on the characteristics of the German private investor in terms of probability of using a sustainable automated advisor. The survey indicates the existence of a demand for sustainable advisory tools, ranking higher among young and male investors (Au et. al, 2021). Such platforms currently exist in Germany and include Growney (Growney, n/d) and Oskar (Oskar, n/d) (Grzanna, 2020), but research from the Deutsche Bank explains why so few Germans invest in ETFs, and thus use robo-advisors less, by highlighting the differences between U.S. and German retirements systems and traditional use of financial advisory services (Kaya & von Martens, 2020).

Table 1 Digital advisor platforms and their ESG approach. Source: Own representation

Platform	***Betterment***	***Wealth-front***	***Wealthsimple***	***Earthfolio***
Area of availability	U. S.	U.S.	Canada and U. K.	U. S.
Minimum opening investment	$10	$500	$0	$25.000
ESG investing strategy	Seeks SRI funds for US large-cap stocks and emerging market stocks only	Allows users to restrict which companies they do not want to invest in	Optimize ETFs based on ESG criteria. They exclude certain companies from stock ETFs and seek municipal local issued bonds and short-term corporate bonds that have good ESG performance	Only offers SRI investment options. Mutual funds best-in-class
ESG ETFs	ESGU, SUSA, ESGE	ESGU, ESGD, ESGE etc.	ESGV, VSGX, SUSB, SPTL, LTPZ, GLDM	BAFWX, CGAFX etc.

2.2.2 Measuring ESG performance

Measuring and evaluating ESG performance of an asset or a company can be challenging and highly contextual. Even the ESG ratings services and ESG-themed mutual funds often disagree because the subjectivity of the topic makes it difficult to move from an abstract point of view to an operational one (Schanzenbach & Sitkoff, 2020). Having experienced a rising importance due to the development of the SRI market, ESG rating agencies have developed their own research methodologies.

Frameworks

The comparative analysis conducted by Escrig-Olmedo et. al (2019) assesses how most representative ESG information provider agencies in the financial market have integrated sustainability aspects in their evaluations, considering the European Commission Sustainable Finance Action Plan, The Sustainable Development Goals (SDGs), Paris 2016 Agreement on Climate Change and the increasing demand for monitoring corporate sustainability performance. The analysis includes well-known rating agencies such as: Refinitiv, ECP, FTSE Russell ESG Ratings, MSCI ESG Research, ISS-oekom, RobecoSAM, Sustainalytics and Vigeo EIRIS. Among the ESG rating agencies, only Refinitv and MSCI ESG Research consider the sustainability dimension of Financial Economic, but compared to MSCI, Refinitv does not evaluate all dimensions in a balanced way, giving each dimension a specific weight. Additionally, MSCI ESG Research Rating Framework incorporates the intergenerational perspective, evaluating how companies manage risks connected to future generation needs (Escrig-Olmedo et. al, 2019).

Methodologies

ESG rating agencies and data providers use different techniques to collect the ESG information needed for rating. Some use surveys, analyses of sustainability reports, interviews with company personnel and stakeholders (trade unions, NGOs etc.) and, increasingly, natural language processing (NLP) and artificial intelligence (AI) for web-scraping of unstructured data. Data is then fed into different metrics, qualitative and quantitative dimensions, frameworks, and conventions. However, a full understanding of the different contributions of these methodologies to the ratings is generally missing, due to the low transparency about the providers' used methods (Eccles & Stroehle, 2018).

While most of the abovementioned methods are considered to be part of the traditional ratings, another set of methods is used by alternative ESG rating agencies. Hughes, Urban and Wojcik (2021) compare traditional and alternative AI-based ESG ratings using respectively MSCI ESG ratings and ratings from Truvalue Labs (TVL). TVL alternative methodology uses complex algorithms to mine more than 10,000 English-language sources daily, such as news, trade blogs, journals, and NGO reports, which results in over 1 million datapoints a month, on over 10,000 companies. The research concludes that alternative ratings' reliance solely on external sources of big data available online and assigning base weights entirely on the volume of noise picked up by the algorithms in key issues, opens way to a more democratic evaluation based on public sentiment rather than on self-published corporate disclosure documents. However, risks such as fake news skewing results or analyst subjectivity in the engineering of algorithms, need to be considered (Hughes, et al., 2021).

ESG Database

The ESG Database chosen for this project is the MSCI ESG Fund Rating. Although there exist several other ESG data providers, the objective of this work is to integrate ESG scores in the portfolio construction using publicly available software and not to assess the quality of the ESG dataset, so a popular dataset is rather preferred.

MSCI is a pioneer in the construction of the ESG Scores that analyse performance over a long historical period and its ESG database has been widely adopted by many asset managers (Bruder et. al, 2019). Moreover, this database has served research purposes in several published papers such as in Giese et. al (2019), Schmidt (2020), Antoncic et. al (2020), Madison & Schiehll (2021) and many more.

MSCI ESG Ratings uses a rules-based methodology to measure a company's resilience to long-term, industry material environmental, social and governance risks. Leveraging Artificial Intelligence (AI), machine learning and natural language processing augmented with team of analysts, they research and rate companies on a 'AAA' to 'CCC' scale according to their exposure to industry-material ESG risks and their ability to manage those risks relative to peers

(MSCI ESG Research, 2020, p. 3). Their source of information is mostly public data such as government regulatory and NGO datasets, together with company disclosure documents and media sources. The rule-based methodology, which is a machine learning and data mining technique designed to find regularities in data and express them in the form of IF-THEN rule (Fürnkranz, 2013), addresses industry specific key issues and evaluates them, keeping track of real-time controversies and events. Combining scores from key ESG issues, an overall rating is published after being reviewed by industry and market checks, as well as a formal committee (MSCI ESG Research, 2020). The scale consists of seven levels starting from ‘AAA’, ‘AA’ representing leaders in the industry for managing ESG risks and opportunities; ‘A’, ‘BBB’, ‘B’ representing an average record of managing risks compared to industry peers; ‘B’, ‘CCC’ representing laggard in the industry because of high exposure and inability to manage ESG risk.

Explaining ESG Scores

The scoring methodology of MSCI ESG Ratings is explained from Bruder et. al (2019). On a top-down view, there are the three pillars: environmental (E), social (S) and governance (G). Each pillar receives a score from 0 to 10 and explores a range of different themes. For example, the environmental pillar considers themes such as climate change, natural capital use, pollution and waste produced, and environmental opportunities to transit to a greener business activity. The social pillar considers themes of human capital management, product liability, stakeholder interest and social opportunities for employees. Lastly, governance pillar examines corporate governance and behaviour. The global ESG score is cumulated as a weighted average of the three pillars. For different industries, the weight of each pillar is subject to change, e.g., the automobile sector gives more weight to the environmental pillar, the banking sector gives more weight to the governance pillar, and the pharmaceutical sector emphasizes the social pillar. Governance pillar carries a considerable weight across all companies (MSCI ESG Research, 2020). An illustration of the pillars and key issues can be seen in Table 2. Data is examined across 35 ESG key issues, which are weighted according to impact and time horizon of the risk or opportunity in the specific issue.

Table 2 MSCI ESG Rating Framework. Source: Adapted from MSCI ESG Research (2020)

3 Pillars	***10 Themes***	***35 ESG Key Issues***	
Environment	Climate Change	Carbon Emissions Product Carbon footprint	Financing Environmental Impact Climate Change Vulnerability
	Natural Capital	Water Stress Biodiversity & Land Use	Raw Material Sourcing
	Pollution & Waste	Toxic Emissions & Waste Packaging Material & Waste	Electronic Waste
	Environmental Opportunities	Opportunities in Clean Tech, in Green Building	Opportunities in Renewable Energy
Social	Human Capital	Labor Management Health & Safety	Human Capital Development Supply Chain Labor Standards
	Product Liability	Product Safety & Quality Chemical Safety Financial Product Safety	Privacy & Data Security Responsible Investment Health & Demographic Risk
	Stakeholder Opposition	Controversial Sources	Community Relations
	Social Opportunities	Access to Communications Access to Finance	Access to Health Care Opportunities in Nutrition & Health
Governance	Corporate Governance	Ownership & Control Board	Pay Accounting
	Corporate Behavior	Business Ethics	Tax transparency

2.3 Incorporating ESG criteria in portfolio construction

Various ways of incorporating ESG in portfolio construction have been mentioned in literature. Referring to Esty and Cort (2020), some of the general trends surrounding sustainable investing strategies are negative exclusion or divesting, green alpha and smart beta strategies, and impact investing. In detail, negative exclusion or divesting happens when investors refuse to invest in companies that offer harmful goods or services, for example tobacco and alcohol industries, while alpha and beta are used as key metrics to measure the performance of an asset or a portfolio of assets. Green alpha strategies focus on maximizing return on good ranking ESG assets, while smart beta strategies focus on benefiting from the low risk that characterizes these investments. Lastly, impact investing seeks to find investments with the highest benefit for the society, despite the financial outcome (Esty & Cort, 2020).

2.3.1 ESG in Fixed income portfolios

Klein (2015) discusses three common approaches to incorporating ESG in fixed income portfolios. One approach, also mentioned above, is excluding companies with critical ESG incidents or sectors such as weapons and alcohol. Second is a best-in-class approach, meaning the sector is not completely excluded but it is avoided investing in poor ESG performers in the sector. Third is a moderated best-in-class approach, implying investment in corporate bonds with poor ESG ratings will only be allowed if the ESG risks are well known, the credit spread is compensating for the risk, and there is confidence in future improvement. The author leans toward the third approach, highlighting the importance of flexibility in portfolio construction and the impact of the challenge presented to the issuer to address ESG related issues and improve their performance.

2.3.2 ESG in Equity portfolios

Comparatively, Bender et. al (2018) examine ESG integration in equity portfolios. For passively indexed, core equity portfolios (Equity ETFs) they derive three use cases: Equity Core Beta (Screened & Cap Weighted), Equity Core Beta (Optimized) and Smart Beta Equity Core. In the first case, the asset universe is screened based on specific ESG metrics; in the second case, an optimization framework is applied to build the portfolio and in the last case, the investor targets a specific driver of return, while incorporating ESG.

Bender, Sun and Wang (2017) further investigate ways to construct portfolios that blend ESG metrics into the Smart Beta or factor investing strategies. The first approach ESG screens the asset universe according to the investor’s comfort level and then applies traditional portfolio optimization methods to build the final portfolio. Here the ESG decision is separated from the factor portfolio construction. Another framework, suitable for investors that consider ESG an important factor of their portfolio, uses an unscreened universe and treats ESG as a standalone additional smart beta factor. An alternative approach is using an unscreened universe and considering ESG as a subcomponent of quality, since the existence of a positive association between ESG and quality has been argued in literature (Antoncic, et al., 2020).

The three strategies were compared in empirical historical simulations that covered a time horizon of approximately 8 years and ESG ratings from two of the aforementioned ESG rating agencies: Sustainalytics and MSCI. The study concludes that the first approach, exclusion, or screening, has the best performance among approaches since it puts the least constraints to the optimization problem. However, it might not always guarantee a positive ESG exposure, because even after the exclusion of the worst ESG issuers, the optimizer might still choose bad performers, to maximize other smart beta factors. Integrating ESG in the optimization problem, as standalone or a subcomponent of a factor, sharpens the problem and secures positive ESG exposure. In particular, the incorporation as a standalone factor, guarantees a stronger positive ESG exposure. But, if the ESG metrics do not have additional alpha (return), it tends to reduce absolute risk-adjusted return (Bender, Sun & Wang, 2017).

2.3.3 ESG in the recommended portfolio

Among all approaches, one of the most popular strategies among investors to incorporate ESG in a portfolio, is the excluding, or screening strategy (de Franco, 2020). This strategy is also adapted in the portfolio construction process. It uses the MSCI ESG Fund Rating system, which maps the ESG Quality score of funds to ‘AAA’ to ‘CCC’ letter categories (see section 2.2.2). The screening will be applied using the same methodology as in Bender, Sun and Wang (2017, p. 94). For the investor with low level of ESG concern, the lowest ESG category is removed, the assets ranking ‘CCC’. For investors with a moderate level of ESG concern, ‘CCC’ and ‘B’ are excluded from the universe. For investors with a high concern of ESG performance, assets ranking ‘CCC’, ‘B’, ‘BB’ and ‘BBB’ are excluded, leaving only ‘A’, ‘AA’ and ‘AAA’ categories, or the best ESG performers. Furthermore, a traditional portfolio optimization method is employed.

2.4 Summary

ESG investing, representing the range of investing strategies with emphasis on sustainability, has gained momentum especially in the last few years. This increasing relevance is attributed, among others, to recent events, generational wealth transfer and new insights gained on the relation between sustainability and financial performance of investment portfolios.

The market for sustainable digital advisors is also growing, with several existing platforms broadening their offers to include ESG investing strategies in their financial services. This market is larger in the U.S. than in Europe, but research shows that European and particularly German-speaking countries are catching up on the trend.

With the absence of strict definitions and standard frameworks, the process of rating the ESG performance of an asset remains debatable. Independent rating agencies use different methodologies to perform their ratings, making the choice of one ESG data provider relatively difficult. For this work, data is extracted from MSCI ESG Ratings, a popular provider choice in industry and academia.

Among researched ESG incorporation strategies in the construction of investment portfolios, the excluding or screening strategy stands out as a popular and efficient solution for different types of investment portfolios, therefore it is also adapted in this work.

3 Portfolio recommendation: A practical approach

3.1 Theoretical Background

This chapter seeks to give an overview of the state-of-the art research in recommender systems (RS) and their application in finance, with respect to the specific domain of portfolio construction. Then a knowledge-based recommender system for ESG portfolio construction is designed.

3.1.1 Recommender systems (RS)

Modern technologies and the ever-growing amount of data has made it challenging for users to choose between the vast number of products and services offered online. It is substantial for companies like Amazon or Netflix to help users come to a decision regarding their wide catalogue of products, so to satisfy customer needs.

Some of these service providers might seek to improve user experience with their application, by suggesting products to undecided users. To recommend content or other type of data to their users and to support the decision-making process, many use recommender systems. These systems are highly applicable to various industries and have been a subject to large body of research over the years (Sharma et. al, 2022).

Examples of recommender system applications can be found in movies (Netflix), e-commerce (Amazon), songs (Spotify), videos (YouTube), social networks (Facebook, Instagram) and many more.

3.1.2 Classification

According to the extended review of RS published from Sharma et. al (2022), RS can be personalized and impersonalized. A personalized RS maintains a user model to generate a recommendation, whereas an impersonalized RS generates recommendations based on the history of user interactions with one product e.g., most viewed or most purchased products. Based on what approach they use to generate a recommendation, recommender systems can be classified into Collaborative filtering, Content-based, Knowledge-based and Hybrid RS (Sharma et. al 2022). Fig. 1 demonstrates the major techniques used.

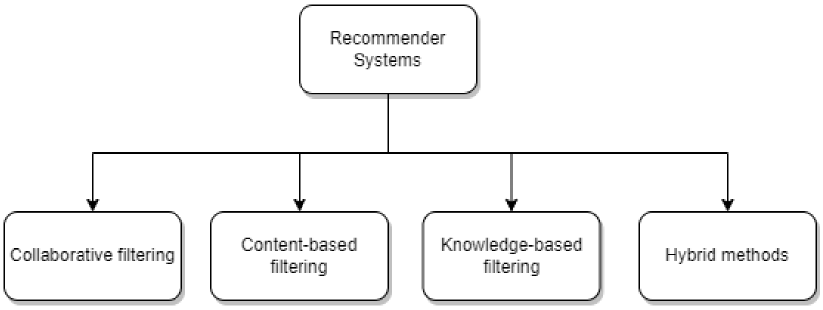


Figure 1 Recommender systems classification. Source: Adapted from Sharma, et al.(2022)

Collaborative filtering

These methods filter based on similarities between different users. They recommend content based on *user* preference for an *item* (Fig. 2), which is expressed in form of a *rating* and is frequently represented as a (User, Item, Rating) triple. Ratings can be unary (has purchased), binary (likes/dislikes), ternary (likes/dislikes/seen but not expressed preference) or based on a scale (i.e., 0-5 stars). The sets of rating triples form a sparse matrix called *ratings matrix*, where for (User, Item) pairs with no ratings available, the value remains unknown. The recommendation is a two-step problem (Ekstrand, Riedl & Konstan, 2010):

1- Prediction task: Given a user and an item, predict what is the preference of the user for the item. This is equivalent to finding the missing values in a matrix.
2- Recommendation task: Given a user, produce the best ranked list of n items for a user's need.

An example of a ratings matrix can be seen in the figure (Fig. 2) below:

	Item A	Item B	Item C
User 1	3	4	2
User 2	?	5	4
User 3	5	3	?

Figure 2 A sample ratings matrix. Source: Own representation

Some of the advantages of collaborative filtering methods (Fig. 3a) are easy implementation, improved prediction and that there is no need for additional users' information. But these methods cannot function if there are no ratings available, so the quality of recommendations suffers in the beginning (cold start problem). Also, these techniques generate a huge number of recommendations over billions of users and products, which ask for a significant computational power (Shah et. al, 2017).

Content based filtering

These filtering methods recommend content based on the correlation between the items and user's preferences (Fig. 3b). It can also be seen as a classification task, of separating the items into a positive class 'c' (relevant to user) and negative class '- c' (irrelevant to user). An item is expressed as a vector X = (x_1, x_2,…, x_n) of n components (qualities of the item). Once a user has expressed interest on an item, a class is attributed to this item. The recommendation task is to select a function h(x) that finds similarities between user's preferences and new, unseen items and classifies the new items as relevant or irrelevant to the user (van Meteren & van Someren, 2000).

Advantages of these methods include user independence after the user has built a profile with their ratings, transparency to the active user (e.g., because you liked similar items) and recommendation of items that have not yet been rated by any user. On the other hand, the filter can suffer from over-specialization since it always recommends the same type of items (Shah et. al, 2017).

Knowledge-based RSs

Knowledge-based RSs (Fig. 3c) base their recommendation on a domain knowledge. Users get recommendations based only on their particular set of requirements and the behaviour of other users is not considered or at least, is not of high importance for the recommendation (Bouraga et. al, 2014). This can be very useful in scenarios like purchasing real estate, cars, financial services and other luxury goods. In these scenarios, collaborative and content-based filtering will not perform well, because people do not buy these items frequently, therefore they are not a lot of available user ratings related to them (Sharma et. al, 2022). Another concern that comes from the financial domain is the privacy issue. Users are not keen on releasing their personal or financial data, thus other filtering methods will not be able to get the data they need (Zibriczky, 2016).

Knowledge based RS do not need a large dataset to function, so they address the cold-start problem. Also, the domain knowledge is noise-free, so the recommendations are more reliable. Limitations include constructing the domain knowledge and maintaining it, together with the expertise needed for knowledge representation (Bouraga et. al, 2014).

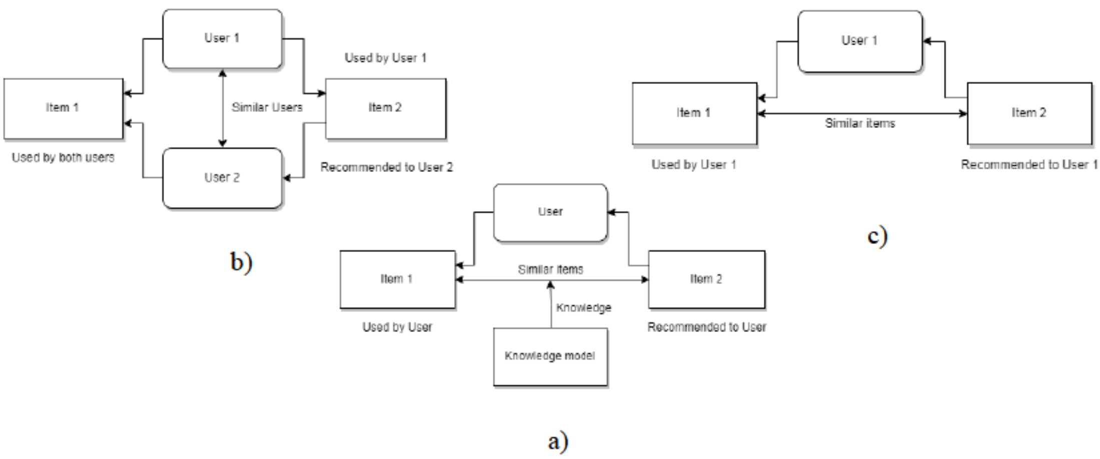


Figure 3 Recommender systems' filtering methods: a) Collaborative filtering; b) Content-based filtering; c) Knowledge-based filtering. Source: Redrawn from Sharma, et al. (2022)

Hybrid methods

Hybrid methods successfully capture the benefits of the aforementioned approaches and try to address the limitations these methods face. They use user's past information to generate recommendations. The only limitation they face is the relevance of the recommendation if the user has stopped using the application for a period of time (Sharma et. al, 2022).

3.2 Recommender systems in finance

3.2.1 Applications of RS in financial services

Compared to other applications of RS, in financial products, the success of the recommendation cannot be determined solely on user feedback. The utility of financial products can only be measured long-term, since it depends on external factors such as market returns, government regulations, fluctuations of currency, etc. In addition, expert knowledge is necessary to judge on the quality of a recommendation. This demands for stricter and more explicit expression of user's wishes to begin with, relative to conventional e-commerce item recommendation (Zibriczky, 2016).

Zatevakhina, Dedyukhina and Klioutchnikov (2019) provide with an overview of the main filtering methods of RS in the financial environment. These methods include:

- Content-based filtering
- Collaborative filtering
- Hybrid methods
- Deep Learn & Enhanced Learn filtering (offer wide coverage of content, even when user interaction is low)
- Knowledge-based filtering
- Utility-based filtering (use of artificial intelligence to personalize recommendations)

3.2.2 Applications of RS in portfolio construction

Portfolio construction techniques, otherwise known as asset allocation techniques, have often been subject to the application of recommender systems. This task requires a solution tailored to users' specific needs, which usually differ from one to the other, meaning content or collaborative filtering would face many limitations. The recommender system must overcome the cold start problem.

As Zibriczky (2016) reviews in his work, non-personalized and personalized recommendation models have emerged, using different filtering methods to propose solutions to the asset allocation and portfolio management problem. Among others, following existing applications of RSs are cited in the author's review: automated recommendations to prevent investors' irrationality in portfolio selection (Elton & Gruber, 2000); application of intelligent agents in portfolio management (Sycara et. al, 1998); extension of Modern Portfolio Theory by fuzzy techniques to model risk-aversion (Zhang Z. & Zhang Ch., 2006); risk estimation of portfolios (Bermudez et. al, 2007); composing optimal portfolios (Fasanghari & Ali Montazer, 2010; Felfernig et. al, 2007); generating efficient portfolios using stock-clustering methods (Nanda, Mahanty & Tiwari, 2010); case-based reasoning with user metadata (Musto et. al, 2014); fuzzy model to transform user ontology and portfolio ontology to a bi-dimensional matrix and recommend based on distance of the models (Garcia-Crespo et. al, 2012; Gonzalez Carrasco et. al, 2012); decision support system for assisting strategic asset allocation (Beraldi, Violi & De Simone, 2011). These works have been independently reviewed to define their approach to ESG aspects of the portfolios. Even though some of the models are extensible, there is no evidence of a practical solution that covers the ESG perspective of the portfolios.

Given the formulation of the problem and after having weighed the advantages and disadvantages of the different filtering methods, the use of a knowledge-based filtering method proves most suitable. Among the financial aspects of the portfolio, this RS should consider the user's level of concern around ESG performance of the assets.

The traditional recommendation of a portfolio can be seen as a two-step problem: 1) select a well-diversified portfolio for the best risk-return trade-off, 2) perform the asset allocation which best fits the investor's profile (Zibriczky, 2016). For step 1) of the portfolio recommendation task, the well-known Modern Portfolio Theory (MPT), published by Nobelist Henry Markowitz (1952), will be employed. For step 2) of the recommendation task, a knowledge-based filtering method will be incorporated, as it is described in the work of Felfernig et. al (2006; 2015). Both these methods will be further explained when necessary. In the ESG RS, step two is performed twice, once before step one, to screen the assets based on investor's preferences about ESG issues and to determine which portfolio optimization method to use, and once after step one, to determine assets and quantities to be included in the recommended end-portfolio.

3.3 A knowledge-based RS for ESG portfolio construction

To be able to test the quality of the recommender system, a universe of 20 example ETFs is chosen. The ETFs are selected using ETF Database, which is a well-known data source for developments on ETFs. The used criteria for the selection are: 10 US-based equity (stock) and 10 fixed income (bond) ETFs with the largest amount of assets under management and expense ratios less than 1%, representatives of different ESG performance levels based on the ETF Database ESG ranking. A list of these ETFs is included in Appendix G.2. The purpose of this selection is to feed data to the RS, in order to fully test the operation of the RS with focus on the ESG profile of the assets. For other purposes, such as maximizing return in the real-world financial environment, another set of selection criteria should be suggested. The modularity of the system should make it possible for this data to be replaceable, to achieve all sorts of purposes, while maintaining the main functionalities intact.

Based on the example of the financial advisor introduced by Felfernig et. al (2006; 2015), the recommender knowledge base for the ESG recommender system is defined as follows. Such knowledge base generally consists of two variable sets: Costumer Properties and Product Properties and three sets of constraints: Compatibility Constraints, Filter Constraints and Product Constraints.

Customer Properties represent possible customer requirements such as:

- risk tolerance level (can be: 'low-moderate', 'moderate - high'),
- budget (in USD),

- time horizon of investment (in years),
- ESG concern level (can be: ‘low’, ‘moderate’, ‘high’).

Product Properties represent possible product instances such as:

- product type (‘equity ETF’, ‘financial income ETF’),
- ESG rating (‘AAA’, ‘AA’, ‘A’, ‘BBB’, ‘BB’, ‘B’, ‘CCC’)

Compatibility Constraints restrict possible customer requirements.

- Budget must be greater or equal than $500 and less or equal than $500,000, for the system to recommend meaningful portfolios.
- Investment time horizon must be greater or equal than one year and less or equal than 50 years, for the system to recommend meaningful portfolios.
- These constraints can be combined willingly.

Filter Constraints establish the relationship between customer properties and product properties:

- ESG concern level is ‘low’ means ESG rating cannot be of type ‘CCC’,
- ESG concern level is ‘moderate’ means ESG rating cannot be of types ‘CCC’, ‘B’,
- ESG concern level is ‘high’ means ESG rating cannot be of types ‘CCC’, ‘B’, ‘BB’, ‘BBB’.

Product Constraints represent constraints of allowed instantiations of product properties (offered set of products):

- There are no constraints in the combination of product properties. Products can be of equity or fixed income types and have different ESG ratings.

The diagram in Fig. 4 gives an approximate idea of how the system should work. After the user has expressed the requirements, the ETFs are screened based on the rules set from the knowledge base. Then, using portfolio construction methods, a portfolio is created with selected ETFs. Lastly the system optimizes the weights for each asset and based on the user budget, performs an asset allocation. These results are represented through a user interface.

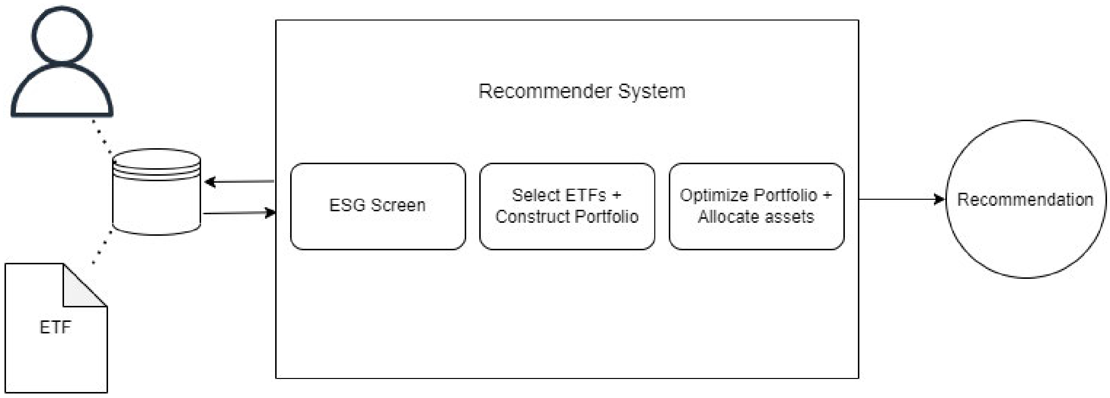


Figure 4 Knowledge-based ESG recommender system draft. Source: Own representation

3.4 Summary

Recommender systems experience a high applicability in different domains, making them a common research area in Computer Science. Literature classifies RSs based on the filtering methods they use in four categories such as collaborative filtering RSs, content-based filtering RSs, knowledge-based RSs, and hybrid RSs. Each of these RSs possesses its own advantages and disadvantages and is best used in specific use case scenarios.

Numerous RSs have also been used in the finance domain. Moreover, research is extended to the application of RSs in portfolio construction. For the task of portfolio construction with focus on incorporating ESG in the investment strategy, a knowledge-based RS is designed and used as a theoretical blueprint for the development phase.

4 Designing a web application to solve the recommendation task

4.1 Requirements Analysis

After the recommender system has been conceptualized, an application is needed to perform the task of solving the recommendation problem. Example applications are mobile apps or browser-based applications. Cerf (2016) compares the two in terms of portability, accessibility, and reliance on the Internet to function. According to the author, web applications offer the highest portability as they can be used on different platforms (mobile, tablet, laptop) and are the most accessible, because of their compatibility with different browsers, whereas mobile apps are built to run in specific operating systems (e.g., IOS or Android). However, he argues, while both rely on the Internet for the most part of their functions, mobile apps can store information locally, preserving a certain degree of functionality when offline.

Goal is to offer a minimum viable product of a financial advisory tool that is free of cost and accessible for every user, regardless of platform, browser, or operating system. The offline usage of the tool is not considered a priority in the scope of this work. Therefore, a web application meets the stated purpose best. The source code for the application is included in electronic format attached to this document.

The requirements formulated in the following sections represent a starting requirement set for development, as found relevant from the author of this thesis. The target user group includes any private investor with interest on sustainable digital financial advisors which based on the study from Au et. al (2021) is likely to have some prior investment experience and to have a high cost-awareness.

4.1.1 Non-functional requirements

Next, requirements for the web-based recommender system are defined. This application is developed using aspects of agile software development frameworks. Such aspects include starting with developing a small portion of software that is fully functional and adding more features with time, all while regularly testing to guarantee quality software. Agile development puts emphasis on working software rather than comprehensive documentation and prioritizes response to change over following a plan (Highsmith & Cockburn, 2001). A short development time is preferred, to offer a viable product with focus on functionality first, and on the design second.

One of the key requirements for the system is modularity. The system is separated in modular components that can be easily adjusted for other purposes, without needing to make changes to the whole system. Functionalities are encapsulated in modules, to ensure scalability in the future. The web application is supported by several browsers, due to the characteristics of used frameworks.

Another important aspect of the application is user experience. The design and development of the application is led by the intention of providing an application with intuitive usage and informative function. This is mirrored in the design of the user interface as well.

4.1.2 Functional requirements

For the system to be a robust digital financial advisor for ESG focused portfolios, it should mainly provide two types of data about the portfolio: general financial data and ESG related data. To formulate the functional requirements for this project, the Quality User Story (QUS) Framework, proposed by Lucassen et. al (2015) was applied. These guidelines intend to improve the quality of user stories. Below is a representation of the user stories for the ESG Recommender system (Table 3). The user stories adhere to the 14 quality criteria requirement engineers should conform to, according to the authors of QUS Framework. Such qualities include syntactic, semantic, and pragmatic qualities, therefore should be, among others: atomic, minimal, problem-oriented, complete, uniform and unique (Lucassen et. al, 2015).

The User Stories give a general idea of the required functionalities, but they do not provide details on the implementation. Some User Stories are linked together, as one of them represents a more granular approach to the other. Implementation details are included in the implementation section of this work and are respectively mapped to the User Story they belong to.

Table 3 User Stories. Source: Own representation

ID	***Description***
US1	As a User, I want to see a personalized recommendation for an all-ETF portfolio
US1a	As a User, I want to establish my own criteria for budget, time of investment, risk tolerance level and ESG concern level
US2	As a User, I want to see the capital allocation to the assets included in the portfolio
US2a	As a User, I want to see a visual representation of the allocated capital
US3	As a User, I want to see the annual volatility of the recommended portfolio
US4	As a User, I want to see the performance of the portfolio in terms of expected annual return and expected return for the investment time horizon
US5	As a User, I want to see the ESG overall performance of the portfolio, quantified in a certain scale
US6	As a User, I want to see relevant ESG data for each asset
US6-Note	Relevant data (according to MSCI): - Asset rating - Environmental: Green-based versus fuel-based revenue - Environmental: Carbon emissions intensity - Social: Holdings that have violated international guidelines - Social: Holdings that have been involved in controversial business activities - Governance: Independence of board of directors - Governance: Diversity, women on boards

4.2 System design

Using the results from the requirements analysis done in the previous section, a decision is made to build a RESTful web service, to interact with users and solve the recommendation task. Representational State Transfer (REST) is an architectural style that defines specific constraints such as the uniform interface and the communication protocol, typically Hypertext Transfer Protocol (HTTP), to assure a high performance of the application on the web (Oracle, n/d).

4.2.1 Script language

The backend for the web application is written in Python. Python is an object-oriented programming language with an expressive syntax, that is human-readable and easy to learn. It is open source, has high portability on many platforms, an interactive interpreter for real-time development and can interact with various other software. Most importantly, its large number of standard libraries and uncomplicated additional download of external libraries, together with the existence of Python bindings to standard GUI toolkits, makes it excel as a language for scientific computing with rapid development of a user interface (Oliphant, 2007). These qualities support the choice of Python as a programming language for our purpose of building a web application for portfolio recommendation. Python version used in this project is the stable release 3.8.

The frontend of the web application uses Bootstrap. Bootstrap is the most popular HTML, CSS, and JavaScript framework for responsive website development. It accelerates web development through a wide array of HTML and CSS based design templates, as well as JavaScript plug-ins for responsive elements. Bootstrap is open-source and supported by all popular browsers, well documented and easy to get started (Gaikwad & Adkar, 2019).

4.2.2 Framework

The web application was built using an efficient web framework for Python programming such as Django. Although Django is slightly surpassed by Flask as most used web framework by professional developers in the Stack Overflow Developer Survey (2021), it focuses, in comparison to Flask, on getting started right out of the box using ready-made modules (Liawatimena et. al, 2018). This way it shortens development time, which is an important criterion in developing this project. Moreover, Django is scalable, mature and can interact with several database management systems such as Oracle, MySQL, SQLite and PostgreSQL (Liawatimena et. al, 2018).

Django uses the Model-View-Template (MVT) pattern (Fig. 5), which consists of three components exhibited in the diagram below (Luo & Ahuja, n/d). This pattern is similar to the Model-View-Controller (MVC) pattern, but it has some significant differences. For example, the view in MVC decides *how* the data is represented, while in MVT, the view decides *which* data is represented. The view in MVT delegates the task of *how* to represent the data to the template. As for the controller role, it is played by the framework itself: it sends requests to the right view, according to the URL configuration (Django, n/d).

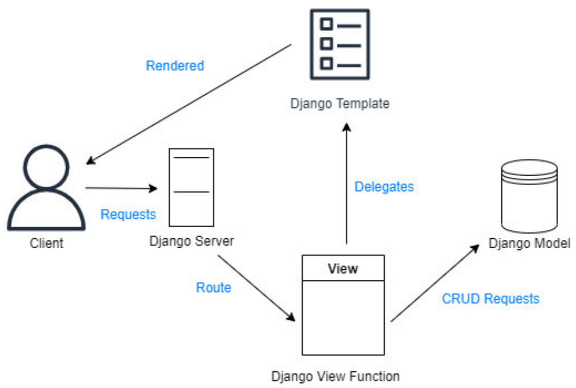


Figure 5 Django MVT pattern. Source: Redrawn from Luo & Ahuja (n/d)

4.2.3 Database

To save the user information acquired from the user interface and to store the ETF asset data, a database is required. Openko et. al (2019) review the problem of choosing a Database Management System (DBMS) in modern information systems. They conclude that the relational DBMSs are the best for storing most types of "simple" data and will remain popular in the upcoming years, despite the popularity rise of non-relational DBMSs. StackOverflow Developer Survey 2021 ranks MySQL as the most used DBMS among developers, with around half of the responders choosing it across all DBMSs. SQLite was also considered, as it is suitable for small projects and android based application systems (Bhosale et al., 2015). Because of its popularity and to secure scalability in the future, MySQL was chosen as the DBMS for this project. The connection with the Django framework was enabled by the Python interface to MySQL - mysqlclient.

4.2.4 Architecture

The application is built using standard Django app development process as shown in Fig. 6. This process starts with the initialization phase, where the project and the apps are created. It continues with the core development phase, where Django models and views are created, views are mapped to the URLs and a simple user interface is build. For each iteration, tests are performed to ensure that everything works properly. After the core development has finished, a third-party frontend is added to aid the design of the user interface. Lastly, the phase of deployment and monitoring consists of deploying the application and checking for performance (Luo & Ahuja, n/d).

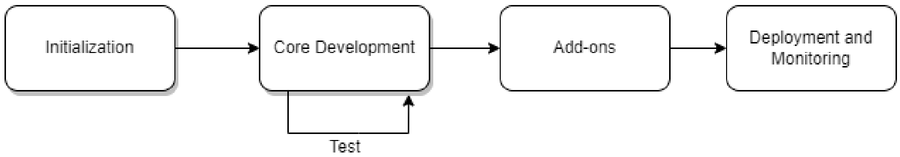


Figure 6 Django development process. Source: Adapted from Luo & Ahuja (n/d)

In the initialization phase, a Python project named EsgRecSys is created. The project can either be created using the command line or an integrated development environment (IDE). This project is created using the PyCharm IDE, which offers specific support for the Django framework and supports scientific packages (Jetbrains, n/d) needed to complete this project.

The Python project contains a Django project named *mysite*, that in turn contains a Django application named *esgrs*. The difference between a Django project and an app is that an app is a web application with a specific function, while a project is a collection of configuration and apps for a particular website (Django, n/d).

The project structure is snapshotted in Fig. 7 from the working IDE.

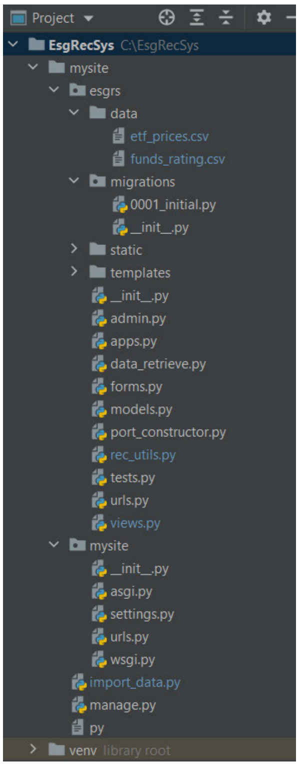


Figure 7 Project file structure. Source: Own representation

Important Django standard components in this project include:

In Django project *mysite*:

- *manage.py*: command line interface to create apps, admin users, to run server
- *settings.py*: configuration file, contains project configuration settings
- *urls.py*: contains URLs and manages routing

In Django app *esgrs*:

- *admin.py*: admin site to interact with the database from Django database API

- *models.py*: contains Django model classes
- *views.py*: contains views
- *urls.py*: includes URL declarations and routing for the views
- *apps.py*: application configuration
- *migrations*: model migration script folder
- *tests.py*: offers a testing environment

Apart from editing standard classes and directories, following additions are made to the project:

In Django project *mysite:*

- *import_data.py:* offers methods to delete ETF records from the database, to add new ones from data files, or to update existing records. These actions can also be taken using the Django admin interface. It offers a simplified approach for Create-Read-Update-Delete (CRUD) operations that bypass the use of views.

In Django app *esgrs*:

- *data*: directory containing data files
- static: directory containing CSS and image files
- *template*s: directory containing templates
- *data_retrieve.py:* downloads ETF price data and saves it in a file in data directory
- *forms.py*: creates a form using Django models in class *models*
- *port_constructor.py*: uses price data to create portfolios
- *rec_utils.py*: contains the business logic to offer personalized recommendations

4.3 Implementation

4.3.1 Project Setup

Once the Django project and application have been created, the Django app is included in the INSTALLED_APPS settings (List. 1).

```
settings.py
INSTALLED_APPS = [
    'esgrs.apps.EsgrsConfig',
    ...
]
```

Listing 1 Settings.py: Install application

Next, a MySQL DB instance has been created using the MySQL Workbench UI with following Structured Query Language (SQL) command (List. 2).

```
CREATE DATABASE db_esg
```

Listing 2 MySQL: Create database

In settings.py, the database is configured (List. 3).

```
settings.py
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'db_esg',
        ...
    }}
```

Listing 3 Settings.py: Configure database

After finishing the initialization phase, the core development starts.

4.3.2 Data extraction, transformation, and loading (ETL)

In order to store the information needed for this project, it is essential to communicate with the database. Django includes a default object-relational mapping layer (ORM) that enables the interaction with relational databases. It offers models, each representing a single database table with model attributes mapping to fields in the database, and an automatically generated database-access API to make queries (Django, n/d).

As derived from the requirements analysis, data on two entities need to be extracted to perform calculations and recommend a personalized portfolio: user profile data and asset data, the latter including ESG profile and ETF price data. Since price data is subject to daily change and covers a historical period of five years, it has been decided to avoid storing the data in a database table, to save the computational time needed to perform an ETL process each time a new set of assets need to be evaluated, upon request from user. Instead, price data is saved as a CSV file, updatable upon browser request when using the ESG RS tool. For User and ETF ESG profiles, two Django models are created.

User profile data

A user profile model is created with attributes such as budget, investment time horizon, risk tolerance level and ESG concern level (List. 4). All fields are required and risk and ESG levels are set to predefined choices. To control the users' input on budget and investment time, as defined in the knowledge-based recommender system constraints, Django build-in validators are used. Each time the user input violates these constraints, a validation error is raised, which causes the form data to be invalid and the system to redirect to an error page.

```
models.py
...
class UserProfile(models.Model):
       ...
       budget = models.PositiveIntegerField(verbose_name='Budget',
                default=500, validators=[MinValueValidator(500),
                MaxValueValidator(500000)])
       inv_time = models.PositiveIntegerField(
                 verbose_name='Investment time horizon', default=1,
                 validators=[MinValueValidator(1),MaxValueValidator(50)])
       risk_level = models.CharField(verbose_name='Risk tolerance level',
                    max_length=1, choices=RISK_CHOICES)
       esg_level = models.CharField(verbose_name='ESG concern level',
                   max_length=1, choices=ESG_CHOICES)
       ...
```

Listing 4 Models.py: Create Django model for User

Using Django ORM settings, this model translates to a database table named *esgrs_userprofile*, where the id field is added automatically and serves as a primary key. Neither budget, nor investment time are allowed to be negative, so they are created as positive integer fields which can store up to 4 bytes of positive integer values.

Another Django shortcut is using ModelForm class to rapidly derive forms that map to the Django models (List. 5). It is not necessary to define the field types in the form, after having defined them in the model class.

```
forms.py
...
class UserProfileForm(ModelForm):
    class Meta:
        model = UserProfile
        fields = '__all__'
```

Listing 5 Forms.py: Create User model form

This creates a form, that can be plugged in an HTML template, controlled by a view, to serve as a user interface. If data from the user is valid, an entry is saved in DB table *esgrs_userprofile*.

In Fig. 8, a snapshot of the requirement elicitation user interface is shown, as requested in US1a. Risk tolerance level and ESG concern level are implemented as drop-down lists of pre-defined entries.

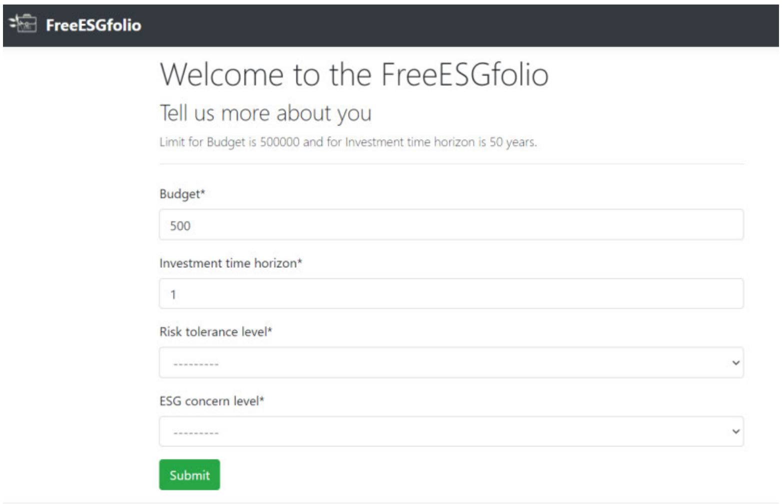


Figure 8 User interface: Requirements elicitation. Source: Own representation

ETF ESG profile data

To build the ESG profile of the ETFs chosen for the application, data was collected from MSCI ESG Fund Ratings website (MSCI ESG Research, n/d). This API is not open-source, and there is no web scraper available to use for the ESG Fund Ratings website, so at this phase of development data must be extracted manually from the website. Building a web scraper from scratch would cost a lot of development time, since most of the data is shown in the format of graphics and pictures, rather than text.

Prior to loading the data, a Django model for the ETF is created in analogy to the user profile model. This model is translated to the database table *esgrs_etfprofile*.

Information extracted to build the ETF ESG profile is illustrated in Table 4. Each attribute represents a field name for the model and a column for the database table. In the table, attributes have been categorized according to whether the type of data they represent is general asset information, or belongs to any of the rating pillars: E, S, or G.

Table 4 ETF profile attributes. Source: Own representation

Attribute name	***Category***	***Description***
ticker	General	Unique identifier assigned to a trading asset
name	General	Official name of the asset
type	General	Asset class: ‘EQ’ for Equity and ‘FI’ for fixed income
rating	General	Overall ESG rating assigned from MSCI
green_based_rev	E	Part of total revenue that is generated from alternative energy
fossil_fuel_based_rev	E	Part of total revenue that is generated from alternative energy

waci	E	Weighted average carbon intensity, based on the weighted average carbon intensity in tonnes per USD million sales waci = tCO_2e/ USD m sales
wabi	G	Weighted average percentage of independent board of directors
wabd	G	Weighted average percentage of women on boards
ungc_viol	S	Number of fund holdings that are in violation of the UNGC principles (UNGC, n/d)
sev_contro	S	Number of fund holdings that are facing very severe controversies related to the Organisation for Economic Co-operation and Development (OECD) Guidelines (OECD, 2011)

Both models are migrated to *db_esg* using migration scripts (List. 6).

```
...\mysite>python manage.py makemigrations
.
...\mysite>python manage.py migrate
```

Listing 6 Migrate changes to database

ETF price data

ETF price data is extracted using yfinance library from Yahoo Finance. This library has become popular as a substitution for Yahoo Finance historical data API, after it was decommissioned a few years ago (Aroussi, 2019). It downloads historical asset data from Yahoo Finance and offers it in form of pandas dataframes. Pandas is a pythonic open-source tool that is used in data analysis to read and manipulate large data (pandas, n/d). Processing the data in pandas dataframes, facilitates the usage of high volumes of data.

In *data_retrieve.py*, tickers of selected ETFs are passed as arguments and available historical, daily 'adjusted closing' price data of a period of five years is downloaded, transformed, and saved to a CSV file in the *data* directory. Using adjusted closing price data makes it easier for investors to compare performance between assets (Ganti, 2020).

After the models are created and the ESG data has been loaded in the database table, the development process continues with implementing the business logic and defining which data to present to the user.

4.3.3 ESG Screening

The screening of the assets based on user's ESG concern level is done using a method named *screen_etfs* in *rec_utils.py*. After the user has submitted the form with their requirements, one Read operation is performed on the last user profile created, to read the ESG concern level they entered and one other is performed on the ETF database table, to retrieve the tickers of the ETFs that have the appropriate ESG rating, based on the screening rules. To do this, Django QuerySet method *exclude* is applied, together with the filter keyword *__in* (List. 7):

- *exclude(**kwargs)* – Returns a new QuerySet containing objects that do not match the given lookup parameters.
- *__in* – is used in combination with a query term, to filter items according to a list of values for an attribute (Django, n/d)
- *excludes* – the list of ratings that the ETFs are not allowed to have. E.g for ESG concern level = 'low', *excludes* = ['CCC']

```
etfs = EtfProfile.objects.exclude(rating__in=excludes)
```

Listing 7 Screen ETFs according to ESG requirements

The tickers of the remaining funds, after the exclusion, are passed as argument to a DataRetrieve instance, that downloads ETF historical price data.

4.3.4 Portfolio Selection

The application logic for portfolio construction has been encapsulated in *port_constructor.py*. Class PortConstructor uses price data to create portfolios based on Modern Portfolio Theory.

PyPortfolioOpt

To apply MPT to select and construct portfolios, the PyPortfolioOpt Python package, introduced by Martin (2021) proves useful. This open-source package offers financial portfolio selection and optimization methods and is currently in use in industry and in academia. PyPortfolioOpt is the first known library-like implementation of general portfolio optimization methods (Martin, 2021). The implementation of these techniques in the ESG RS is based on the official documentation and on the examples offered by the author.

MPT

MPT is technically composed from Markowitz' Portfolio Selection Theory also known as mean-variance portfolio optimization and William Sharpe's theory of financial asset price formation, otherwise known as Capital Asset Pricing Model (CAPM) (Veneeva, 2006). It is ultimately an investment framework for the selection of assets and construction of investment portfolios, which aims to maximize expected return, while simultaneously minimizing risk. In its core concept, it relies on the idea of "not putting all eggs in one basket", thus benefiting from the advantages of portfolio diversification (Fabozzi, et al., 2002).

MPT argues that any given investments' risk and return characteristics need to be considered in how it affects the overall portfolio's risk and return, rather than to be viewed in a stand-alone basis. In MPT, risk is synonymous with the volatility of a portfolio, which represents the amount of risk that is related to fluctuations in the price of an asset. The set of portfolios that offer the highest return for a given level of risk form the Efficient Frontier (Mangram, 2013; Hayes, 2021; Ganti, 2022). CAPM describes the relationship between risk and expected return and introduces the Sharpe ratio, a measure of the excess return that comes with added risk (Kenton, 2022). MPT in combination with CAPM aims to find the portfolio on the Efficient Frontier that maximizes the Sharpe ratio. Essentially, it calculates the most rewarding portfolio in terms of return, given the level of risk.

To select the assets for the portfolio, MPT is used, while certain methods to reduce systematic error are applied.

Sample Covariance Matrix vs Ledoit-Wolf Shrinkage

To determine what assets to include in the portfolio, it is necessary to determine the relationship between the assets' prices. A positive covariance means that assets generally move in the same direction, whereas a negative covariance indicates an opposite direction. To reduce overall risk, MPT uses historical asset prices to calculate covariance and selects assets that show a negative covariance (Investopedia, 2020). In their paper, Ledoit and Wolf (2003) propose an alternative method to improve the calculation of covariance between assets prices, which aims to reduce error when estimating the sample covariance matrix. This method is called shrinkage and tends to pull the most extreme values towards more central values, thus reducing systematic error in the calculations.

PyPortfolioOpt method:

- *ledoit_wolf(shrinkage_target='constant_variance')* – calculates the shrinkage estimate for a particular shrinkage target. Returns shrunk sample covariance matrix.

Mean historical returns vs CAPM returns

According to Martin (2021), CAPM returns tend to be slightly more stable than mean historical returns, resulting in less estimation error.

PyPortfolioOpt method:

- *pypfopt.expected_returns.capm_return(prices,market_prices=None,returns_data=False, risk_free_rate=0.02, compounding=True, frequency=252, log_returns=False)* – compute a return estimate using the CAPM. Returns annualised return estimate.

The code snippet in List. 8 shows the use of these two methods to select which assets to include in the portfolio.

```
port_constructor.py
...
# Method to create Portfolio based on user budget and risk level

def create_port(self, budget, risk_level):
...
        # Calculate expected CAPM returns
        ex_returns = expected_returns.capm_return(prices)

        # Calculate cov matrix using Ledoit - Wolf shrinkage
        cov_matrix = risk_models.CovarianceShrinkage(prices).ledoit_wolf()
        ...
```

Listing 8 Port_constructor.py: Calculate returns

4.3.5 Portfolio Optimization

Constructing a portfolio with optimal risk-return trade-off is only the first step in offering a personalized recommendation. What also needs consideration, is the risk aversion of the investor, which they have expressed using the UI. Based on the model of the recommender system, two general risk tolerance levels are possible: low-moderate and moderate-high. Practically this has been translated in choosing how to optimize a portfolio. An investor with low-moderate risk profile would rather prefer a portfolio with low risk, therefore the portfolio with the minimum volatility is advised. On the other hand, the investor with moderate-high risk profile, is more interested in the highest rewarding portfolio, therefore a recommendation for the portfolio that maximizes the Sharpe ratio is offered.

To find the optimal asset combination for any given level of risk, it is necessary to construct the Efficient Frontier. PyPortfolioOpt offers different methods to calculate the Efficient Frontier based on the vector of expected returns and a covariance matrix. In this work, the general Efficient Frontier module is used:

- *class pypfopt.efficient_frontier.EfficientFrontier(expected_returns,cov_matrix, weight_bounds=(0, 1), solver=None, verbose=False, solver_options=None)* – An EfficientFrontier object that contains multiple optimization methods that can be called with various parameters. Returns an array of weights, corresponding to the selected assets.

Minimum Volatility Portfolio

For investors with low-moderate risk, the portfolio is constructed as a global minimum volatility portfolio (GMVP) (List. 9). Because returns are hard to predict, but volatilities and covariances tend to be stable, the Efficient Frontier only uses the covariance matrix as a parameter to calculate an optimal asset combination. Some research shows that these types of portfolios often outperform the traditional mean-variance portfolios for out of sample examples (Wixom, n/d). GMVP is a special case of MPT that results either if all assets of the mean-variance portfolio have the same expected return, or the investor has infinite risk aversion (Reh, Krüger & Liesenfeld, 2020).

PyPortfolioOpt method:

- *Min_volatility()* - minimizes volatility. Returns asset weights for the volatility minimizing portfolio.

```
port_constructor.py
...
def create_port(self, budget, risk_level):
...
    # Risk level low-moderate, Global min variance Portfolio
        if risk_level == 1:
            # No need to provide expected returns in this case
            ef_1 = EfficientFrontier(None,cov_matrix,weight_bounds=(0, 1))
            ef_1.min_volatility()
            weights = ef_1.clean_weights() # format weights
 ...
```

Listing 9 Port_constructor.py: Construct GMVP

Optimal Portfolio

For investors with moderate-high risk tolerance level, the portfolio in the Efficient Frontier that maximizes Sharpe ratio is calculated (List. 10). The risk-free rate usually represents the yield received for investing in a U.S. government issued treasury security with a maturity of 10 years. For the calculation of the Sharpe ratio, it is left at a default 2% (two percent), which is in the range of the last year's values (Yahoo! Finance, n/d).

PyPortfolioOpt method:

- *Max_sharpe()* – maximizes the Sharpe ratio. Returns the asset weights for max Sharpe portfolio.

```
port_constructor.py
...
def create_port(self, budget, risk_level):
...
    # Risk level low-moderate, Global min variance Portfolio
     elif risk_level == 2:
            ef_2 = EfficientFrontier(ex_returns, cov_matrix)
            ef_2.max_sharpe(risk_free_rate=0.02)  # default
            weights = ef_2.clean_weights() # format weights
...
```

Listing 10 Port_constructor.py: Construct maximum Sharpe portfolio

Asset Allocation

After the weight of each asset has been determined, some post-processing of the optimal weights needs to be applied, to make the weights usable in real life situations. For example, if the optimal weight is allocating $1000 of the budget to an asset, but according to the asset's current price, say $133/share, there is no exact number of shares that can be purchased with exactly $1000, another allocation needs to happen, to adjust to the discrete characteristic of the problem (Martin, 2021). To perform a discrete allocation of the assets, a greedy algorithm is used.

As explained by Martin (2021), the algorithm runs in two rounds. In the first round, the maximum number of shares for each asset is bought, without exceeding the optimal weight. In the second round, the weights of the allocated assets are recalculated, then the deviation of these new weights from the former weights is estimated. The assets with the highest deviation are purchased first with the money left over, until there are no more purchases available. The remainder after the allocation is kept in cash.

PyPortfolioOpt class and methods:

- *Class pypfopt.discrete_allocation.DiscreteAllocation(weights, latest_prices, total_portfolio_value=10000, short_ratio=None)* – Generates a discrete portfolio allocation from continuous weights. Returns a python dictionary of ticker symbols with the respective number of share that can be bought.
- *greedy_portfolio()* – uses the aforementioned greedy algorithm.

Example code of discrete allocation in the portfolio construction is shown in List. 11.

```
port_constructor.py
...

def create_port(self, budget, risk_level):
       ...
       # Prices as of the day you are allocating
       latest_prices = self.prices.iloc[-1]
       # Discrete allocation using user budget
       da = DiscreteAllocation(weights, latest_prices,
                                total_portfolio_value=budget)
       alloc, leftover = da.greedy_portfolio(verbose=True)
...
```

Listing 11 Port_constructor.py: Perform discrete allocation

4.3.6 Recommendation results

To show the recommendation results to the user as requested in US1, a GUI has been built using HTML and Bootstrap (Appendix G.3). This interface must show results in an understandable, human-friendly format. Relevant information based on the user requirements table must be displayed to the user, to assure that the utility of the webpage for the user is being maximized. It is important that the GUI adheres to main principles of User Interface Design which state that the UI should be usable, likeable, enjoyable, communicate brand values and reinforce users' trust (Interaction Design Foundation, n/d). Some of the key elements used to display results are described below. The description of the elements follows the order of placement in the UI.

To display important information about the portfolio, Bootstrap cards are used. These are flexible containers that include options for text and images. Cards are used to separate important information from the rest of the webpage and associated with images, they serve to attract user attention toward key information about their portfolio. The choice of images seeks to make the UI enjoyable and user-friendly.

General Financial Information Card

In US3 and US4, the user requested to receive general financial information about the portfolio. To fulfil this requirement, a card containing general financial information such as five-year annualized return, total investment value at the end of the investment period and portfolio volatility, is placed at the top of the general information container.

Asset allocation Card

A visual representation of the asset allocation was requested by the user in US2a. After the asset allocation has taken place in the portfolio constructor class, a dictionary with asset tickers as keys and allocated budget as values, is used to plot the asset allocation in a Matplotlib pie chart. Matplotlib is a comprehensive python library for creating visualizations. Graphics created with Matplotlib are embeddable in GUIs and compatible with Django web applications (Matplotlib, n/d).

Matplotlib offers several renderers for interactive and non-interactive use. Since the asset allocation is happening on the fly, after the user has sent a browser request expressing their requirements, the weights are also being plotted on the fly, but the presentation on the webpage is static. For this reason, the backend used to save the chart in a PNG format, to have it rendered on the HTML results webpage as a static image is ‘AGG’. AGG backend of Matplotlib writes high quality images to a PNG file using the Anti-Grain Geometry engine (Matplotlib, n/d). Once the pie chart has been plotted, a PNG file of the chart is saved in the image directory, which is delegated as context to the HTML template, to be displayed on the webpage.

ESG Overall Rating Card

In US5, the user requested to see a quantified overall ESG performance of the portfolio. To fulfil this requirement, the MSCI letter rating ‘AAA to CCC’ is converted to a points scale of respectively ‘7 to 1’. The overall ESG performance of the portfolio is calculated as the average rating of the assets included in the portfolio. For different rating values, a different illustration is displayed, showing an award to the user. For example, if the portfolio scores more than six points on average, the highest award illustration is displayed, to appreciate the user’s concern about ESG issues.

Portfolio Information Table

The core element of the UI, which holds the most important information for the user, is the table containing the data of the recommended portfolio, as requested in US2. This table is strictly informative and does not represent real financial advice. The user can see which ETFs are being included in their personalized portfolio, their current price, the number of shares that the recommender system advises them to purchase, and the ESG rating of each asset. Lastly, it shows how much the purchase would cost (not considering any expense ratios) and how much money from the available budget is left after the discrete allocation has taken place. All this information is saved in variables in *views.py* and delegated to the templates to display on the UI.

Environmental Card

The card containing the environmental information of the portfolio displays values of average percentage green based revenue, meaning revenue that the holding included in the portfolio generated by goods and services using alternative energy sources, sustainable water and that contributed to preventing pollution. In addition, it displays the average percentage of fossil fuel-based revenue, including revenue created by thermal coal, oil and natural gas extraction or power generating activities using the aforementioned sources. Furthermore, it shows information about the average percentage of carbon intensity emissions, calculated as tones carbon dioxide CO2 per unit of million USD in revenue of assets included in the portfolio. These averages are calculated on the basis of MSCI data for each of the assets.

Social Card

The social information card contains two values: the highest number of holdings in the portfolio that have committed UNGC principles violations and the highest number of holdings to have been involved in very severe controversial business activities, resulting in the OECD guidelines violations. UNGC principles summarize a number of declarations on human and labour rights, environment and anti-corruption responsibilities that corporates should meet, to be considered sustainable businesses (UNGC, n/d). OECD aims to improve lives of people all around the world by making policies and encouraging cooperation between countries and organizations. Each year, the organization publishes a report on guidelines for multinational enterprises, setting a sort of code of conduct for corporations (OECD, 2011).

Governance Card

In the governance card, information on the independence of the board of directors and percentage of women in the boards of the corporations is displayed. Both values are averages derived from assets included in the portfolio. For special cases, such as bond ETFs, that represent e.g., treasury bonds, there is no data available for these two attributes, as they do not represent corporations. In this case the data for the asset is simply ignored, to not skew the results for the whole portfolio.

ESG Information Table

Another essential informative element of the recommender system is the table representing ESG data of the portfolio, as requested in US6. The table contains all the ESG information on the assets, color-coded for easier understanding. The data for this table has been saved in variables in views, analogue to the core portfolio information, and sent to the templates to render as context.

4.4 Workflow

To summarize the way the system works and how the elements interact with each other, a step-by-step workflow is described below:

1- User expresses requirements using the user interface homepage. The homepage is loaded using the get function view and user data is saved using the post function view.
2- The method select_etfs in rec_utils.py, screens the ETFs in the ETF database table and saves the list of tickers whose ESG rating matches the user's ESG level of concern.
3- An instance of DataRetrieve class is instantiated to download financial data for the selected ETFs. The financial data includes daily adjusted closing prices of the last five years, and it is outputted as a csv file. Not available data series are dropped.
4- Upon the user's selection of risk level, a portfolio construction method has been selected in rec_utils.py, create_portfolio() method.
5- In PortConstructor class, prices have been fed to the portfolio construction methods, to form the covariance matrix and to calculate expected returns. Based on the covariance between asset prices, the portfolio constructor decides which of the assets to include in the portfolio. Then, a convex optimizer defines the optimal combination of assets that create the Efficient Frontier. Weights are calculated based on the user's risk profile. For users with lower risk profile, weights of the global minimum variance portfolio are calculated, whereas for users with higher risk profile, the weights for the portfolio that maximizes Sharpe ratio are estimated.
6- Based on the user budget, the weights are post-processed, and a discrete allocation is performed.

7- Financial and ESG data about the assets contained in the portfolio is delegated to the template and rendered to the user. Necessary calculations are performed to summarize results about the recommendation made.

4.5 Quality assurance

Several measures were taken during the development process to produce a qualitative and performant application. First, the development was strictly guided by the user requirements defined in the functional requirements section, to avoid writing unnecessary lines of code that might affect performance in the long run. Second, special attention was dedicated to the modular architecture of the system, to ensure that the code is reusable and that changes in the future can be made easily. Third, unit tests were performed on views, models, and methods, to make sure that the components were acting as intended. Moreover, the system was tested after each incrementation, to ensure that the added functionalities were properly integrated in the system.

To enhance user experience, the user interface for requirements elicitation was designed using controlled input options, such as default values indicating starter budget and investment time, as well as drop-down menus to indicate levels of risk and ESG concern. On one hand, this shortens user time on the website, reducing the risk of overloading the website with HTTP requests. On the other hand, it reduces the risk of users landing on the error page, contributing to an improved user experience with the website.

4.6 Summary

In this chapter, the recommender system is developed in the form of a web application. In the requirements analysis, functional and non-functional requirements are formulated. These requirements guide the system design and the implementation phase.

Using the Django framework for building RESTful web applications, the project is laid out as a standard Django web application that uses a Model-View-Controller type of pattern. However, the business logic is separated in different classes, to avoid creating fat models or fat views. Moreover, the modularity of the project serves future development opportunities.

To implement functionalities, numerous Python packages and libraries are used, including yfinance, NumPy, pandas, PyPortfolioOpt, and MatplotLib. The GUI is implemented using Bootstrap and HTML. The recommendation results are represented according to UI design principles. Quality assurance measures are taken to check the performance of the application.

5 Assessments

5.1 Evaluating the quality of the recommendation

To evaluate the quality of the recommendation, example recommended portfolio returns are benchmarked against Standard & Poor's (S&P) Target Risk Indices (Table 5), commonly used to assess the performance of different types of portfolio strategies. Such strategies include conservative, moderate, aggressive, and very aggressive asset allocation strategies. Starting from January 2021, S&P introduced a new ESG aligned target risk index series. S&P ESG Equity Target Risk Moderate Index (S&P Equity MI) measures performance of moderate stock-bond allocations to fixed income, while using ESG-themed equities to increase possibilities for higher returns. S&P ESG Equity Target Risk Aggressive Index (S&P Equity AI) measures performance of aggressive allocations to ESG-themed equities, while including small allocations of fixed income (S&P, 2021). These indices are used to benchmark portfolios created as recommendation for investors with low-moderate risk tolerance and moderate-high risk tolerance levels. The global minimum variance portfolios created for users with lower risk tolerance, usually include more fixed income assets, since they tend to have lower volatility rates, therefore they are benchmarked against the Moderate Index. Whereas maximum Sharpe ratio portfolios created for users with higher risk tolerance, generally include more equity type assets, therefore they are benchmarked against the Aggressive Index. As comparison values, the five-year annualized return is used. Budget value used is 1000 USD and investment time horizon covers five years.

Table 5 Results comparison to benchmark. Source: Own representation

Risk level	***ESG level***	***Benchmark***	***5-year price return annualized***	
			Benchmark	Portfolio
Low-moderate	low	S&P ESG Equity MI	4.12%	2.3%

(GMVP)	moderate	S&P ESG Equity MI	4.12%	2.3%
	high	S&P ESG Equity MI	4.12%	2.4%
Moderate-high (Max Sharpe)	low	S&P ESG Equity AI	7.87%	9.0%
	moderate	S&P ESG Equity AI	7.87%	8.2%
	high	S&P ESG Equity AI	7.87%	9.7%

As it can be seen in Table 5, the global minimum variance portfolios suggested from the recommender system to users with lower risk tolerance, seem to underperform the benchmark, for all levels of ESG screening. However, these portfolios have the lowest volatility, meaning that they trade off return, to offer a lower risk. On the other hand, the maximum Sharpe ratio portfolios, seem to outperform the benchmark, for all levels of ESG screening. Moreover, both types of portfolios show a higher return for the highest level of ESG screening, indicating that for the isolated asset universe chosen, the highest level of ESG screening, also offers the portfolios with the best financial performance.

5.2 Evaluating the utility of the advisor

A research paper by Rühr (2020) on user preferences in relation to digital financial advisors indicates that attributes such as automation, control and transparency substantially contribute to the overall utility of the system. The research explores how the utility of the application changes with increasing levels of automation, starting from low levels of automation where algorithms only support the human advisor, moving to human advisor supervised algorithms and ending with full automation with no influence of human advisors. Furthermore, it explores the relationship control-utility by researching changes in utility with increasing user control. Three levels of control are considered, starting from no control in the decisions of the robo-advisor, moving to a medium level of control where user can determine their risk profile, and ending with high control over risk profile and portfolio creation. For the attribute of transparency three levels are also defined. They start with no transparency, meaning no information on functionality is given to the user, proceed to the level where explanations on the relevance of data inputs is given and end with high levels of transparency, meaning the digital advisor provides real-time explanations on the relevance of data inputs and processes (Rühr, 2020).

Based on the categorization of these attributes in the conducted research (Rühr, 2020), the ESG recommender system has a full level of automation, meaning the algorithms conduct the entire decision process with no influence of human advisors. The user can adjust their risk profile and can influence the portfolio creation, by changing their set of requirements and in a more advanced usage, by voluntarily providing external ETF data to the recommender system. As far as transparency level goes, the user receives real-time explanation on the relevance of the data inputs and the process in which the recommendation was created. Therefore, the ESG RS is characterized by the highest levels of automation, control and transparency attributes.

The research by Rühr (2020) concludes that users value higher levels of automation significantly more compared to low levels, even though moving from advanced to full levels of automation slightly decreases utility. Also, highest levels of control and transparency significantly increase overall utility of the system. Based on these findings, the overall utility of the developed recommender system, evaluated from the perspective of a digital financial advisor, is arguably high. However, there are limitations on evaluating the utility of the recommender system when seen from a sustainability perspective, given the current absence of a comparable evaluation framework.

5.3 Summary

In this chapter, the recommendations' quality is evaluated by benchmarking the returns of the recommended portfolio against S&P ESG Target Risk indices. While the GMVPs underperform the benchmark, the maximum Sharpe portfolios seem to outperform the benchmark for all levels of ESG screening. In addition, the usability of the ESG recommender system as a digital financial advisor is evaluated using the framework introduced by Rühr (2020). The ESG recommender system scores high in usability, due to its automation, control, and transparency levels.

6 Conclusions

6.1 Project summary

With the goal of producing a sustainable financial advisory tool, this thesis takes a deep dive in a number of complex topics such as ESG investing, recommender systems and portfolio construction theories. First and foremost, it depicts the recognized importance of shifting from traditional investing strategies to ESG and socially responsible approaches, to steer the economy toward sustainable practices that not only benefit the future generations well-being, but also secure positive returns on investment for todays' investors. By dissecting the different ESG related approaches of digital advisor platforms and those of ESG rating agencies, this work points out the discrepancies between different providers and offers an original approach to ESG investing supported from best-practices and state-of-the-art literature. This theoretical background serves as a foundation for a recommender system design and is followed by extensive research during the development phase, to support product design and implementation decisions.

Apart from pure technical details of implementation from a computer scientist perspective, this work gives insight on portfolio construction theories and methods and illustrates their implementation in the ESG recommender system, to also offer a deeper understanding of the financial aspects of the implementation. By doing so, it seeks to achieve transparency of the way the system works.

Ultimately, the thesis provides a critical perspective on the results of the project and the quality of the product. Most importantly, it seeks to recognize its own limitations as a project in the scope of a bachelor thesis, by simultaneously recognizing the necessity of a cooperation with professionals from the finance domain, to enhance product quality.

6.2 Limitations and further development

The ESG recommender system is built to be a fully automated digital financial advisor, nonetheless, the quality of the recommendations highly depends on the initial asset universe that is evaluated from the system. For this reason, certain limitations apply. Knowledge-based recommender systems' reliance on the knowledge base, requires the expert knowledge in the field, to build the knowledge base and to be able to evaluate the quality of the recommendations. Since expert knowledge was not available to acquire in the scope of this project, the knowledge base of the ESG RS offers only two levels of risk aversion to choose from, that translate in two different portfolio construction strategies with focus either on opting for the lowest risk or for the highest return for a given risk. It is however possible to use MPT, to assume more than two risk targets and to find the optimal portfolio for that particular risk target. Yet, such assumptions would be incorrect in the absence of domain expert knowledge. In turn, this limitation can be seen as a further development scenario. If given, expert knowledge can be incorporated in two substantial parts of the recommendation process, first being the selection of the asset universe to begin with, and second, in optimizing the portfolio construction strategies based on user profile. This thesis mainly provides a tool that has the ability to produce a qualitative recommendation after such knowledge base has been created, and for this reason, it offers the means to easily make these changes if needed.

Another limitation that was faced during the development, was the lack of a free API for ESG data extraction. During the development, this process was performed manually, which in a real-life case scenario would be unproductive, prone to errors and not flexible to changes. The automatic extraction of ESG data is crucial for a future use of the system. Also, by automating the process of ESG data extraction, more than one source can be used to perform the evaluations, thus making them less biased. The automation can be supported by a web scraping tool if the websites where the data needs to be extracted allow it. The development of a web scraping tool for ESG data from different websites can be a topic of further research.

In future development scenarios, the ETF ESG screener component of the recommender system can be extended to screen the ETFs based on the most important pillar for the user (E, S or G), since users can have higher concerns about environmental issues, but not share the same interest in social justice issues or vice versa. Going further it can be developed to exclude ESG issuers that are particularly emitting high amounts of carbon dioxide in the atmosphere, or those which do not encourage diversity in their boards of directors. Moreover, it can be designed to advertise sustainable business practices and urge investors to engage more with these topics.

Lastly, the system can serve as a starting point for the development of a hybrid recommender system. Knowledge-based and collaborative-based filtering methods can be combined to offer product recommendations to users with similar profiles, who are undecided on how they feel about environmental issues, thus encouraging them to consider investing more sustainably, like their peers are already doing.

References

Adams, C. A. & Abhayawansa, S., 2021. Connecting the COVID-19 pandemic, environmental, social and governance (ESG) investing and calls for 'harmonisation' of sustainability reporting. *Critical Perspectives on Accounting*, p. 102309.

Antoncic, M., Bekaert, G., Rothenberg, R. V. & Noguer, M., 2020. Sustainable Investment - Exploring the Linkage between Alpha, ESG, and SDG's. *SSRN,* August.p. 34.

Aroussi, R., 2019. *Reliably download historical market data from with Python: Aroussi.* [Online]
Available at: https://aroussi.com/post/python-yahoo-finance
[Accessed 18 February 2022].

Au, C.-D., Klingerberger, L. & Svoboda, M. F. E., 2021. Business Model of Sustainable Robo-Advisors: Empirical Insights for Practical Implementation. *Sustainability,* 18 November.13(23).

Bender, J. et al., 2018. A Blueprint for integrating ESG into Equity Portfolios. *The Journal of Investment Management*, 16(1), p. 20.

Bender, J., Sun, X. & Wang, T., 2017. Thematic Indexing, Meet Smart Beta! Merging ESG into Factor Portfolios. *The Journal of Index Investing Winter*, 8(3), pp. 89-101.

Beraldi, p., Violi, A. & De Simone, F., 2011. A decision support system for strategic asset allocation. *Decision Support Systems,* 51(3), pp. 549-561.

Bermudez, J. D., Segura, J. V. & Vercher, E., 2007. *A fuzzy ranking strategy for portfolio selection applied to the Spanish stock market.* London, UK, IEEE, pp. 1-4.

Betterment, n/d. *How Betterment Works.* [Online]
Available at: https://www.betterment.com/how-it-works

Bhosale, S., Patil, M. & Patil, M., 2015. Sqlite: Light database system. *International Journal of Computer Science and Mobile Computing,* 44(4), pp. 882-885.

Bosson, H., 2016. Shorting the Devil. *Cornell Business Review*, 20 10.

Bouraga, S., Jureta, I., Faulkner, S. & Herssens, C., 2014. Knowledge-Based Recommendation Systems: A survey. *International Journal of Intelligent Information Technologies,* 10(2), pp. 1-19.

Bruder, B., Cheikh, Y. & Deixonne, F., 2019. Integration of ESG in asset allocation. *SSRN,* 25 October, Issue 3473874, p. 29.

Cerf, V. G., 2016. Apps and the web. *Communications of the ACM,* 59(2), p. 7.

Chen, J., 2022. *Exchange-Traded Fund (ETF).* [Online]
Available at: https://www.investopedia.com/terms/e/etf.asp

de Franco, C., 2020. Practical Applications of ESG Controversies and Their Impact on Performance. *Practical Applications,* 8(2), pp. 1-6.

Django, n/d. *Applications: Django Documentation.* [Online]
Available at: https://docs.djangoproject.com/en/4.0/ref/applications/
[Accessed 18 February 2022].

Django, n/d. *FAQ General: Django.* [Online]
Available at: https://docs.djangoproject.com/en/dev/faq/general/#django-appears-to-be-a-mvc-framework-but-you-call-the-controller-the-view-and-the-view-the-template-how-come-you-don-t-use-the-standard-names
[Accessed 1 February 2022].

Django, n/d. *Making queries: Django Documentation.* [Online]
Available at: https://docs.djangoproject.com/en/4.0/topics/db/queries/#the-pk-lookup-shortcut
[Accessed 27 February 2022].

Django, n/d. *Models: Django Documentation.* [Online]
Available at: https://docs.djangoproject.com/en/4.0/topics/db/models/
[Accessed 16 January 2022].

Earthfolio, n/d. *How It Works, ESG Screens: Earthfolio.* [Online]
Available at: https://www.earthfolio.net/sustainability/esg-screens/
[Accessed 18 January 2022].

Eccles, R. G. & Stroehle, J. C., 2018. Exploring social origins in the construction of ESG measures. *SSRN 3212685.*

Eccles, R. & Stroehle, J., 2018. Exploring social origins in the construction of ESG measures. *SSRN 3212685.*

Ekstrand, M. D., Riedl, J. T. & Konstan, J. A., 2010. Collaborative Filtering Methods. In: *Collaborative Filtering Recommender Systems, Foundations and Trends in Human-Computer Interaction.* Delft: now Publishers Inc., pp. 81-173.

Escrig-Olmedo, E., Fernández-Izquierdo, M. Á. & Ferrero-Ferrero, I., 2019. Rating the Raters: Evaluating how ESG Rating Agencies Integrate Sustainability Practices. *Sustainability,* 11 February, 11(3), p. 915.

Esty, D. & Cort, T., 2020. *Values at Work: Sustainable Investing and ESG Reporting.* 1 ed. s.l.:Palgrave Macmillan, Cham.

Fabozzi, F. J., Gupta, F. & Markowitz, H. M., 2002. The Legacy of Modern Portfolio Theory. *The Journal of Investing Fall* , 11(3), pp. 7-22.

Fasanghari, M. & Montazer, G. A., 2010. Design and implementation of fuzzy expert system for Tehran Stock Exchange portfolio recommendation. *Expert Systems with Applications,* 37(9), pp. 6138-6147.

Felfernig, A., F. G., Jannach, D. & Zanker, M., 2015. Constraint-based recommender systems. In: *Recommender systems handbook.* Boston, MA: Springer, pp. 161-190.

Felfernig, A., Isak, K., Szabo, K. & Zachar, P., 2007. *The VITA financial services sales support environment.* Menlo Park, CA; Cambridge, MA; London, AAAI Press; MIT Press, p. 1962.

FNG, 2021. *Marktbericht Nachhaltige Geldanlagen 2021, Deutschland, Österreich & die Schweiz,* Berlin: Forum Nachhaltige Geldanlagen e.V..

Freshfields Bruckhaus Deringer, 2005. *A legal framework for the integration of environmental, social and governance issues into institutional investment,* s.l.: Freshfields Bruckhaus Deringer.

Fürnkranz, J., 2013. *Rule-based Methods: Springer.* [Online]
[Accessed 2 February 2022].

Gaikwad, S. S. & Adkar, P., 2019. A Review Paper on Bootstrap Framework. *Iconic Research and Engineering Journals,* 2(10), pp. 341-351.

Ganti, A., 2020. *Adjusted Closing Price: Investopedia.* [Online]
Available at:
https://www.investopedia.com/terms/a/adjusted_closing_price.asp#:~:text=The%20main%20advantage%20of%20adjusted,to%20lose%20half%20their%20money.
[Accessed 29 January 2022].

Ganti, A., 2022. *Efficient Frontier: Investopedia.* [Online]
Available at: https://www.investopedia.com/terms/e/efficientfrontier.asp
[Accessed 28 February 2022].

Garcia-Crespo, A., Lopez-Cuadrado, J. L., Gonzalez-Carrasco, I. & Colomo-Palcaios, R. R.-M. B., 2012. SINVLIO: Using semantics and fuzzy logic to provide individual investment portfolio recommendations. *Knowledge-Based Systems,* Volume 27, pp. 103-118.

Giese, G. et al., 2019. Foundations of ESG Investing: How ESG Affects Equity Valuation, Risk, and Performance. *The Journal of Portfolio Management,* July, 45(5), pp. 69-83.

Gigli, A., Lillo, F. & Regoli, D., 2017. Recommender Systems for Banking and Financial Services. *RecSys Posters.*

González-Carrasco, I. et al., 2011. PB-ADVISOR: A private banking multi-investment porfolio.. *Information Science,* November, Volume 206, pp. 63-82.

Gravier, E., 2022. *The 5 best robo-advisors when you want to be hands off with your investments: CNBC.* [Online]
Available at: https://www.cnbc.com/select/best-robo-advisors/
[Accessed 01 January 2022].

Growney, n/d. *Homepage: Growney.* [Online]
Available at: https://growney.de/
[Accessed 15 January 2022].

Grzanna, M., 2020. Nachhaltige Robo Advisors: Automatisch grüner. *Süddeutsche Zeitung*, 9 Dezember.

Hauck, D., Voorhes, M. & Goldberg, G., 1983. *Two Decades of Debate: the controversy over U.S. companies in South Africa.* Washington, D.C.: Investor Responsibility Research Center, Inc.

Hayes, A., 2021. *Impact Investing With a Robo-Advisor.* [Online]
Available at: https://www.investopedia.com/impact-investing-with-a-robo-advisor-4775379
[Accessed 28 01 2022].

Hayes, A., 2021. *Volatility: Investopedia.* [Online]
Available at: https://www.investopedia.com/terms/v/volatility.asp
[Accessed 4 December 2021].

Highsmith, J. & Cockburn, A., 2001. Agile software development: the business of innovation. *Computer,* 34(9), pp. 120-127.

Hughes, A., Urban, M. A. & Wojcik, D., 2021. Alternative ESG Ratings: How Technological Innovation Is Reshaping Sustainable Investment. *Sustainability,* 13(6), p. 3551.

Institute for sustainable investing, 2021. *Morgan Stanely Homepage.* [Online]
Available at: https://www.morganstanley.com/ideas/sustainable-investing-sentiment-covid-19
[Accessed 22 01 2022].

Interaction Design Foundation, n/d. *User Interface Design: Interaction Design Foundation.* [Online]
Available at: https://www.interaction-design.org/literature/topics/ui-design
[Accessed 25 February 2022].

Investopedia, 2020. *How Is Covariance Used in Portfolio Theory?: Investopedia.* [Online]
Available at: https://www.investopedia.com/ask/answers/041315/how-covariance-used-portfolio-theory.asp#:~:text=Covariance%20is%20used%20in%20portfolio,overall%20risk%20for%20a%20portfolio.
[Accessed 29 January 2022].

Jennifer, B. et al., 2018. A Blueprint for Integrating ESG into Equity Portfolios. *The Journal of Investment Management,* June.16(1).

Jetbrains, n/d. *Pycharm: Jetbrains.* [Online]
Available at: https://www.jetbrains.com/pycharm/
[Accessed 10 February 2022].

Kaya, O. & von Martens, T., 2020. *German robo-advisors: March of the machines driving passive*, Frankfurt am Main: Deutsche Bank Research.

Kenton, W., 2022. *Capital Asset Pricing Model (CAPM): Investopedia.* [Online]
Available at: https://www.investopedia.com/terms/c/capm.asp
[Accessed 8 January 2022].

Kerber, R. & Jessop, S., 2021. *Analysis: How 2021 became the year of ESG investing,* s.l.: Reuters.

Klein, C. M., 2015. *Integrating ESG into the Fixed-Income Portfolio.* Frankfurt, Germany, CFA Institute.

Ledoit, O. & Wolf, M., 2004. Honey, I Shrunk the Sample Covariance Matrix. *The Journal of Portfolio Management,* 30(4), pp. 110-119.

Liawatimena, S. et al., 2018. Django Web Framework Software Metrics Measurement Using Radon and Pylint. *2018 Indonesian Association for Pattern Recognition International Conference (INAPR),* pp. 218-222.

Lucassen, G., Dalpiaz, F. & van der Werf, J. M. E. M., 2015. *Forging high-quality User Stories: Towards a discipline for Agile Requirements.* s.l., IEEE, pp. 126-135.

Luo, Y. & Ahuja, R., n/d. *Django Application Development with SQL and Databases: edX IBM Computer Science e-Course.* [Online]
Available at: https://www.edx.org/course/django-application-development-with-sql-and-databases
[Accessed 5 December 2021].

Madison, N. & Schiehll, E., 2021. The Effect of Financial Materiality on ESG Performance Assessment. *Sustainability,* 25 March, 13(7), p. 3652.

Mangram, M. E., 2013. A Simplified Perspective of the Markowitz Portfolio Theory. *Global Journal of Business Research,* 7(1), pp. 59-70.

Markowitz, H., 1952. Portfolio Selection. *The Journal of Finance,* 7(1), pp. 77-91.

Martin, R., 2021. *Post-processing weights: PyPortfolioOpt.* [Online]
Available at: https://pyportfolioopt.readthedocs.io/en/latest/Postprocessing.html
[Accessed 7 December 2021].

Martin, R., 2021. *Risk and return models: Github.* [Online]
Available at:
https://github.com/robertmartin8/PyPortfolioOpt/blob/102c593d54e979c1046ad5cf5d3e1a9c77dcd692/cookbook/1-RiskReturnModels.ipynb
[Accessed 28 February 2022].

Matplotlib, n/d. *Backends: Matplotlib Documentation.* [Online]
Available at: https://matplotlib.org/stable/users/explain/backends.html
[Accessed 26 February 2022].

Matplotlib, n/d. *Matplotlib: Visualization with Python.* [Online]
Available at: https://matplotlib.org/
[Accessed 26 February 2022].

MSCI ESG Research, 2020. *MSCI ESG Ratings,* s.l.: s.n.

MSCI ESG Research, n/d. *ESG Fund Ratings: MSCI.* [Online]
Available at: https://www.msci.com/our-solutions/esg-investing/esg-fund-ratings
[Accessed 2 March 2022].

Musto, C. et al., 2015. Personalized finance advisory through case-based recommender systems and diversification strategies. *Decision Support Systems,* Volume 77, pp. 100-111.

Nanda, S., B., M. & Tiwari, M., 2010. Clustering Indian stock market data for portfolio management. *Expert Systems with Applications,* 37(12), pp. 8793-8798.

Norrestad, F., 2022. *Share of professional investors increasing their ESG investments worldwide 2020-2021,* s.l.: Statista.

O'Connell, B. & Curry, B., 2021. *Investing Basics: What Is A Portfolio?.* [Online]
Available at: https://www.forbes.com/advisor/investing/portfolio/
[Accessed 7 12 2021].

OECD, 2011. *OECD Guidelines for Multinational Enterprises,* s.l.: OECD Publishing.

Oliphant, T., 2007. Python for Scientific Computing. *Computing in Science & Engineering, IEEE,* 9(3), pp. 10-20.

Openko, P. V. et al., 2019. Problem of Choosing a DBMS in Modern Information System. *2019 IEEE International Conference on Advanced Trends in Information Theory (ATIT),* pp. 171-174.

Oracle, n/d. *What are RESTful Web Services: The Java EE 6 tutorial.* [Online]
Available at: https://docs.oracle.com/javaee/6/tutorial/doc/gijqy.html
[Accessed 13 February 2022].

Osborne, M., 2017. *Forbes: Brands Need To Step Up Their Game To Win Over Millennials.* [Online]

Available at: https://www.forbes.com/sites/forbesagencycouncil/2017/09/26/brands-need-to-step-up-their-game-to-win-over-millennials/?sh=20dd9db21b32
[Accessed 20 12 2021].

Oskar, n/d. *Anlagestrategie: Oskar.* [Online]
Available at: https://www.oskar.de/anlagestrategie/
[Accessed 15 January 2022].

pandas, n/d. *About pandas: pandas.* [Online]
Available at: https://pandas.pydata.org/
[Accessed 26 February 2022].

Reh, L., Krüger, F. & Liesenfeld, R., 2020. *Predicting the global minimum variance portfolio,* s.l.: Karlsruher Institut für Technologie (KIT), SSRN 3471216.

Ruggie, J. G. & Middleton, E. K., 2019. Money, Millennials and Human Rights: Sustaining 'Sustainable Investing'. *Global Policy,* 10(1), pp. 144-150.

Rühr, A., 2020. *Robo-Advisor Configuration: An Investigation of User Preferences and the Performance-Control Dilemma.* s.l., Research Papers, p. 94.

S&P Dow Jones Indices, 2021. *S&P Dow Jones Indices Introduces New ESG Aligned Target Risk Index Series: S&P Global.* [Online]
Available at: https://www.spglobal.com/spdji/en/index-launches/article/sp-dow-jones-indices-introduces-new-esg-aligned-target-risk-index-series/
[Accessed 28 February 2022].

Salampasis, D., 2017. Leveraging robo-advisors to fill the gap within the SRI marketplace. *Journal of Innovation Management,* 6(13).

Schanzenbach, M. M. & Sitkoff, R. H., 2020. ESG Investing: Theory, Evidence, and Fiduciary Principles. *Journal of Financial Planning,* p. 9.

Schmidt, A. B., 2020. Optimal ESG portfolios: an example for the Dow Jones Index. *Journal of Sustainable Finance & Investment.*

Sette, M. L., 2022. *Best Robo-Advisors: Investopedia.* [Online]
Available at: https://www.investopedia.com/best-robo-advisors-4693125
[Accessed 1 March 2022].

Shah, K., Slunke, A., Dongare, S. & Antala, K., 2017. *Recommender systems: An overview of different approaches to recommendations.* s.l., 2017 4th International Conference on Innovations in Information, Embedded and Communication Systems.

Sharma, M. et al., 2022. *A Survey and Classification on Recommendation Systems.* s.l., 2nd International Conference on Big Data, Machine Learning and Applications (BigDML 2021).

Stack Overflow, n.d. *2021 Developer Survey: Stack Overflow Insights.* [Online]
Available at: https://insights.stackoverflow.com/survey/2021#most-popular-technologies-webframe-prof
[Accessed 17 February 2022].

Sycara, K., Zeng, D. & Decker, K., 1998. Intelligent Agents in Portfolio Management. In: *Jennings N.R., Wooldridge M.J. (eds) Agent Technology.* s.l.:Springer, Berlin, Heidelberg, pp. 267-281.

Tepper, T. & Schmidt, J., 2022. *Best Robo-Advisors Of March 2022: Forbes Advisor.* [Online]
Available at: https://www.forbes.com/advisor/investing/best-robo-advisors/
[Accessed 1 March 2022].

UNGC, n/d. *The Ten Principles of the UN Global Compact: United Nations Global Compact.* [Online]
Available at: https://www.unglobalcompact.org/what-is-gc/mission/principles
[Accessed 27 February 2022].

van Meteren, R. & van Someren, M., 2000. Using Content-Based Filtering for Recommendation. *Proceedings of the machine learning in the new information age: MLnet/ECML2000 workshop,* May, Volume 30, pp. 47-56.

Veneeva, V., 2006. *Analysis of Modern Portfolio Theory: SimplySearch4it!.* [Online]
Available at: http://www.simplysearch4it.com/article/31242.html
[Accessed 27 January 2022].

Wealthfront, n/d. *Investing: Wealthfront.* [Online]
Available at: https://www.wealthfront.com/investing
[Accessed 18 January 2022].

Wealthfront, n/d. *Socially responsible ETFs: Wealthfront.* [Online]
Available at: https://www.wealthfront.com/explore/investment-categories/socially-responsible-etfs
[Accessed 20 January 2022].

Wealthsimple, n/d. *Homepage: Wealthsimple.* [Online]
Available at: https://www.wealthsimple.com/en-ca?uk_redirect=1&
[Accessed 18 January 2022].

Whelan, T., Atz, U. & Clark, C., 2021. *ESG and financial performance: Uncovering the relationship by aggregating evidence from 1000 plus studies published between 2015-2020,* s.l.: NYU Stern.

Wixom, D., n/d. *Introduction to Portfolio Risk Management in Python: datacamp Online Course.* [Online]
Available at: https://campus.datacamp.com/courses/introduction-to-portfolio-risk-management-in-python/univariate-investment-risk-and-returns?ex=1
[Accessed 10 February 2022].

Yahoo! Finance, n/d. *Treasury Yield 10 Years (^TNX): yahoo!finance.* [Online]
Available at: https://finance.yahoo.com/quote/%5ETNX?p=%5ETNX
[Accessed 1 March 2022].

Zatevakhina, A., Dedyukhina, N. & Klioutchnikov, O., 2019. *Recommender Systems - The Foundation of an Intelligent Financial Platform: Prospects of Development.* s.l., 2019 International Conference on Artificial Intelligence: Applications and Innovations (IC-AIAI), pp. 104-1046.

Zatevakhina, A., Dedyukhina, N. & Klioutchnikov, O., 2019. *Recommender Systems - The Foundation of an Intelligent Financial Platform: Prospects of Development.* s.l., 2019 International Conference on Artificial Intelligence: Applications and Innovations (IC-AIAI).

Zhang, Z. & Zhang, C., 2004. 7 Agent-Based Hybrid Intelligent System for Financial Investment Planning. In: *Agent-Based Hybrid Intelligent Systems. Lecture Notes in Computer Science.* s.l.:Springer, Berlin, Heidelberg, pp. 93-125.

Zibriczky, D., 2016. *Recommender Systems meet Finance: A literature review.* Bari, Italy, s.n.

G Appendix 1

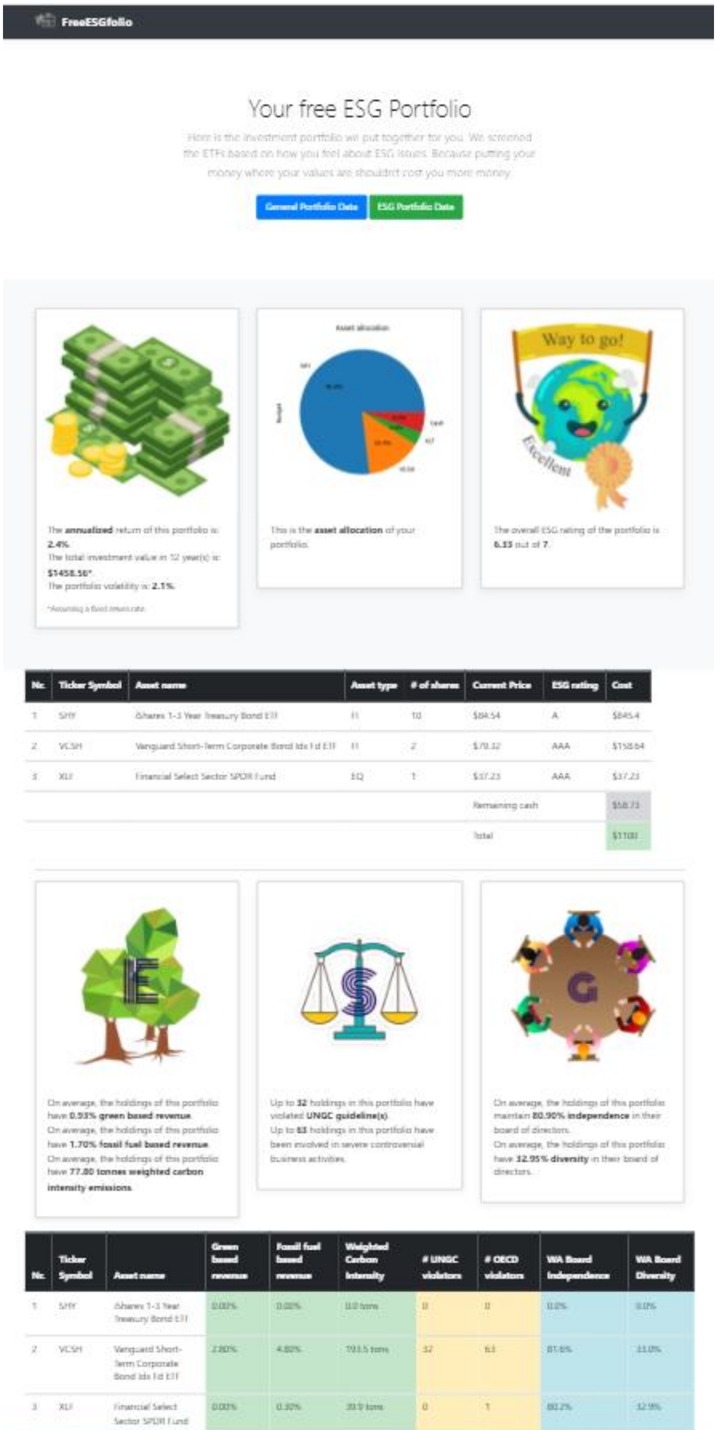
G.1 Table of ESG ETFs offered by digital advisor plattforms

ESG ETF Ticker	***Name***	***Platform***
BAFWX	Brown Advisory Sustainable Growth Fund Institutional Shares	Earthfolio
CGAFX	Calvert Green Bond Fund Class A	Earthfolio
ESGD	iShares ESG Aware MSCI EAFE ETF	Wealthfront
ESGE	iShares ESG Aware MSCI EM ETF	Betterment, Wealthfront
ESGU	iShares ESG Aware MSCI USA ETF	Betterment, Wealthfront
ESGV	Vanguard ESG US Stock ETF	Wealthsimple
GLDM	SPDR Gold MiniShares Trust	Wealthsimple
LTPZ	PIMCO 15+ Year US TIPS Index Exchange-Traded Fund	Wealthsimple
SPTL	SPDR Portfolio Long Term Treasury ETF	Wealthsimple
SUSA	iShares MSCI USA ESG Select ETF	Betterment
SUSB	iShares ESG Aware 1-5 Year USD Corporate Bond ETF	Wealthsimple
VSGX	Vanguard ESG International Stock ETF	Wealthsimple

G.2 Table of ETFs included in the recommender system

Ticker	*Name*	*Type*	*Rating*
AGG	iShares Core US Aggregate Bond ETF	FI	A
BND	Vanguard Total Bond Market Index Fund ETF	FI	A
EMB	iShares JPMorgan USD Emerging Markets Bond ETF	FI	BB
HYEM	VanEck Emerging Markets High Yield Bond ETF	FI	BB
IJR	iShares Core S&P Small-Cap ETF	EQ	BBB
IVV	iShares Core S&P 500 ETF	EQ	AA
IWM	iShares Russell 2000 ETF	EQ	BBB
JETS	US Global Jets ETF	EQ	BB
KSA	iShares MSCI Saudi Arabia ETF	EQ	B
PCY	Invesco Emerging Markets Sovereign Debt ETF	FI	BB
SHY	iShares 1-3 Year Treasury Bond ETF	FI	A
SPY	SPDR S&P 500 ETF Trust	EQ	AA
TIP	iShares TIPS Bond ETF	FI	A
VCIT	Vanguard Intermediate-Term Corp Bond Idx Fund ETF	FI	AA
VCSH	Vanguard Short-Term Corporate Bond Idx Fd ETF	FI	AAA
VGT	Vanguard Information Technology Index Fund ETF	EQ	AAA
VWOB	Vanguard Emerging Markets Govt Bd Idx ETF	FI	BB
VOX	Vanguard Communication Services Index Fund ETF	EQ	BB
XLC	Communication Services Select Sector SPDR Fund	EQ	BB
XLF	Financial Select Sector SPDR Fund	EQ	AAA

G.3 Results web page



FreeESGfolio

Your free ESG Portfolio

Here is the investment portfolio we put together for you. We screened the ETFs based on how you feel about ESG issues. Because putting your money where your values are shouldn't cost you more money.

General Portfolio Data | ESG Portfolio Data

The **annualized** return of this portfolio is: **2.4%**.
The total investment value in 12 year(s) is: **$1458.56***.
The portfolio volatility is: **2.1%**.

*Assuming a fixed return rate.

This is the **asset allocation** of your portfolio.

The overall ESG rating of the portfolio is **6.33** out of **7**.

Nr.	Ticker Symbol	Asset name	Asset type	# of shares	Current Price	ESG rating	Cost
1	SHY	iShares 1-3 Year Treasury Bond ETF	FI	10	$84.54	A	$845.4
2	VCSH	Vanguard Short-Term Corporate Bond Idx Fd ETF	FI	2	$79.32	AAA	$158.64
3	XLF	Financial Select Sector SPDR Fund	EQ	1	$37.23	AAA	$37.23
					Remaining cash		$58.73
					Total		$1100

On average, the holdings of this portfolio have **0.93% green based revenue**.
On average, the holdings of this portfolio have **1.70% fossil fuel based revenue**.
On average, the holdings of this portfolio have **77.80 tonnes weighted carbon intensity emissions**.

Up to **32** holdings in this portfolio have violated **UNGC guideline(s)**.
Up to **63** holdings in this portfolio have been involved in severe controversial business activities.

On average, the holdings of this portfolio maintain **80.90% independence** in their board of directors.
On average, the holdings of this portfolio have **32.95% diversity** in their board of directors.

Nr.	Ticker Symbol	Asset name	Green based revenue	Fossil fuel based revenue	Weighted Carbon Intensity	# UNGC violators	# OECD violators	WA Board Independence	WA Board Diversity
1	SHY	iShares 1-3 Year Treasury Bond ETF	0.00%	0.00%	0.0 tons	0	0	0.0%	0.0%
2	VCSH	Vanguard Short-Term Corporate Bond Idx Fd ETF	2.80%	4.80%	193.5 tons	32	63	81.6%	33.0%
3	XLF	Financial Select Sector SPDR Fund	0.00%	0.30%	39.9 tons	0	1	80.2%	32.9%

Erklärung zur selbstständigen Bearbeitung einer Abschlussarbeit

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst und nur die angegebenen Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.

[redacted]

Ort	Datum	Unterschrift im Original