WEST UNIVERSITY OF TIMIŞOARA
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE
BACHELOR STUDY PROGRAM: Informatică

BACHELOR THESIS

SUPERVISOR:
Conf. Dr. Micotă Flavia

GRADUATE:
Stănculete Andreea

TIMIŞOARA
2024

WEST UNIVERSITY OF TIMIŞOARA
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE
BACHELOR STUDY PROGRAM: Informatică

Smart Budgeting

SUPERVISOR:
Conf. Dr. Micotă Flavia

GRADUATE:
Stănculete Andreea

TIMIŞOARA
2024

Abstract

In the digital world in which we live in today, one of the prime causes for poor financial decisions is the lack of financial literacy. Although there are plenty of websites and resources who aim to help people understand and improve their financial skills, many of them are either inaccessible due to the geographic location, or are not intuitive and hard to understand. In this paper, we propose a web application aimed to help people better manage their finances by easing their work through automatic receipt scanning and periodical incomes, and by helping them understand their cash flow through personalized insights.

What we achieved is an application with an intuitive interface, which is both aesthetically pleasing and easy to navigate and to understand. Unlike the current state-of-the-art, our solution is not impeded by geographical constraints and provides personalized insights periodically in order to help the end-user better understand their finances.

Rezumat

În lumea digitală în care trăim astăzi, lipsa de educatie financiară este una dintre cauzele principale pentru care oamenii nu își ating scopurile financiare. Deși există o mulțime de site-uri web și resurse care își propun să ajute oamenii să înțeleagă și să își îmbunătățească abilitățile financiare, multe dintre ele sunt fie inaccesibile din cauza locației geografice, fie nu sunt intuitive și sunt greu de înțeles. În acest articol, propunem o aplicație web menită să ajute oamenii să își administreze mai bine bugetul, simplificând activitățile lor prin utilizarea de venituri si cheltuieli periodice, scanarea automată de bonuri fiscale, și prin ajutarea lor să își înțeleagă fluxul de bani prin sfaturi personalizate.

Ceea ce am realizat este o aplicație cu o interfață intuitivă, care este atât plăcută estetic, cât și ușor de navigat și de înțeles. Spre deosebire de starea actuală a tehnologiei, soluția noastră nu este împiedicată de constrângerile geografice și oferă periodic rapoarte personalizate pentru a ajuta utilizatorul să își înțeleagă mai bine bugetul.

Contents

Chapter 1

Introduction

In a world of digitalization, more and more people find themselves in stressful situations as the result of improper financial decisions. Social media platforms along with modern digital technology have transformed the way we perceive time and money, by creating a sense of urgency and overwhelm in people's lives. The online environment facilitates spending money through effortless shopping, digital payment methods and personalized advertisements [AA23]. The fear of missing out on experiences and owning certain items are prime examples of reasons why people often find themselves in difficult situations, which could have been easily avoided through a better management of their finances.

Our well-being as individuals is tightly connected to our financial situation [TK23]. The better decisions we take when managing our finances, the less anxiety and unease we feel. For this, financial literacy has become a requirement not only to receive the most benefits from our money and investments, but also to achieve one's objectives.

In regards with the advanced technology we have these days, there have been studies on the association of financial and digital literacy [ACL21]. It is no longer sufficient to simply understand basic financial skills to be financially stable. Habits such as budgeting, saving money, and investing need to be correlated with the use of financial technologies to improve spending.

1.1 Motivation

In the light of the above, it's only natural that financial tracking applications have become a necessity to many people. When used well, they are practical tools that can help a person understand their flow of money, highlight areas where expenses can be reduced and avoid miscalculations.

Most of the time, these applications come with numerous facilities which reduce the amount of time and effort that a user must invest into keeping a budget. Yet, as the way we perceive time has changed, the approach which such applications employ should change as well.

More personalized recommendations tailored to each user's financial situation and goals can help optimize spending, and highlight saving opportunities and investment strategies. Innovative tools such as receipt scanning and automatic expense categorizing can improve the overall user experience and smoothen the financial management process.

When deciding on which application works best for you, several factors must be taken into consideration, namely:

1. The application's user interface
2. How flexible it is
3. The amount of workload required

1.2 The market

Top applications keep a simple interface while putting a lot of features on the table. They support a large variety of currencies and allow their user to automate their cash-flow, keeping the effort on their side to a minimum. In the figure 1.1, we have analysed several applications which are currently dominating the market.

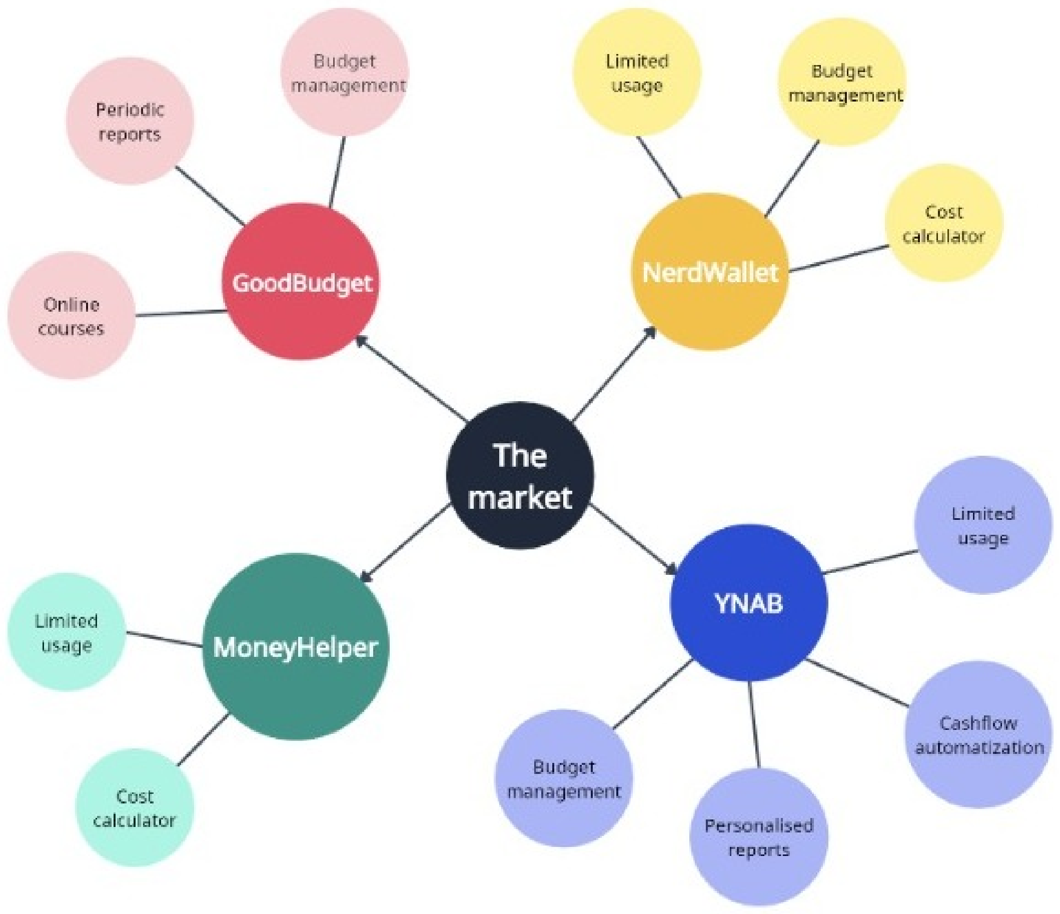
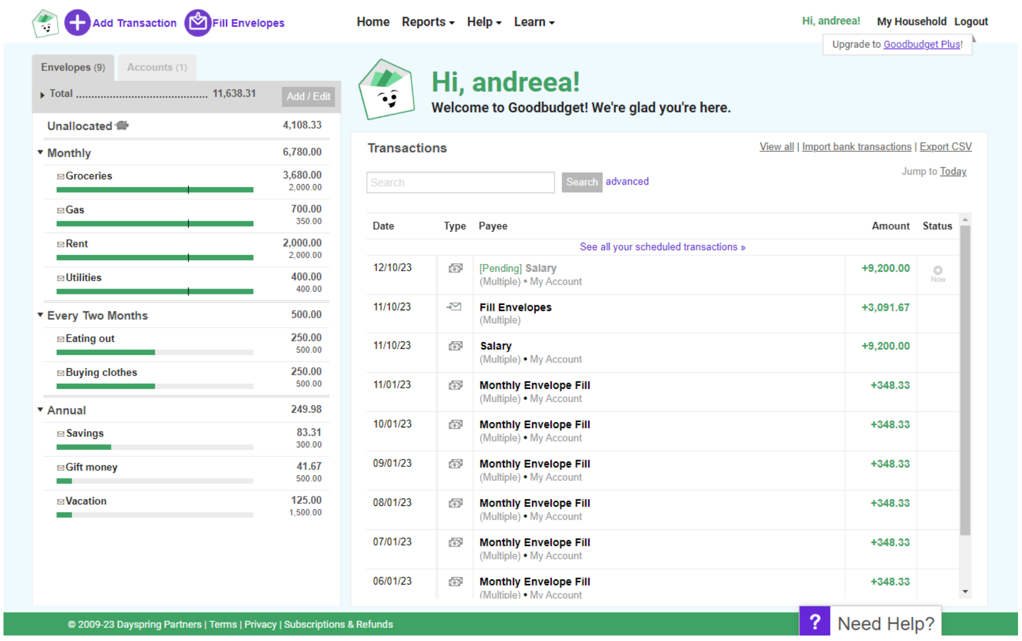


Figure 1.1: Similar solutions

The main flaw of these applications is their limited usage. Most of them either support a narrow amount of currencies, either are restricted to several geographic regions. This prevents a large portion of the population from using them. Furthermore, a great majority offer little to no automation, requiring the user to manually input everything.

1.2.1 GoodBudget

GoodBudget [Goo] is a money manager application based on an envelope system: each expense has an envelope assigned in which money will be placed monthly and a person is allowed to only use the allocated sum when paying for an expense.


Add Transaction
Fill Envelopes
Home
Reports
Help
Learn
Hi, andreea!
My Household
Logout
Upgrade to Goodbudget Plus!
Hi, andreea!
Welcome to Goodbudget! We're glad you're here.
Envelopes (9)
Accounts (1)
Add / Edit
Transactions
View all | Import bank transactions | Export CSV
Search
Search
advanced
Jump to Today
See all your scheduled transactions »
© 2009-23 Dayspring Partners | Terms | Privacy | Subscriptions & Refunds
?
Need Help?

Total	11,638.31	
Unallocated	4,108.33	
Monthly	6,780.00	
Groceries	3,680.00	2,000.00
Gas	700.00	350.00
Rent	2,000.00	2,000.00
Utilities	400.00	400.00
Every Two Months	500.00	
Eating out	250.00	500.00
Buying clothes	250.00	500.00
Annual	249.98	
Savings	83.31	300.00
Gift money	41.67	500.00
Vacation	125.00	1,500.00

Date	Type	Payee	Amount	Status
12/10/23		[Pending] Salary (Multiple) • My Account	+9,200.00	Now
11/10/23		Fill Envelopes (Multiple)	+3,091.67	
11/10/23		Salary (Multiple) • My Account	+9,200.00	
11/01/23		Monthly Envelope Fill (Multiple) • My Account	+348.33	
10/01/23		Monthly Envelope Fill (Multiple) • My Account	+348.33	
09/01/23		Monthly Envelope Fill (Multiple) • My Account	+348.33	
08/01/23		Monthly Envelope Fill (Multiple) • My Account	+348.33	
07/01/23		Monthly Envelope Fill (Multiple) • My Account	+348.33	
06/01/23		Monthly Envelope Fill (Multiple) • My Account	+348.33	

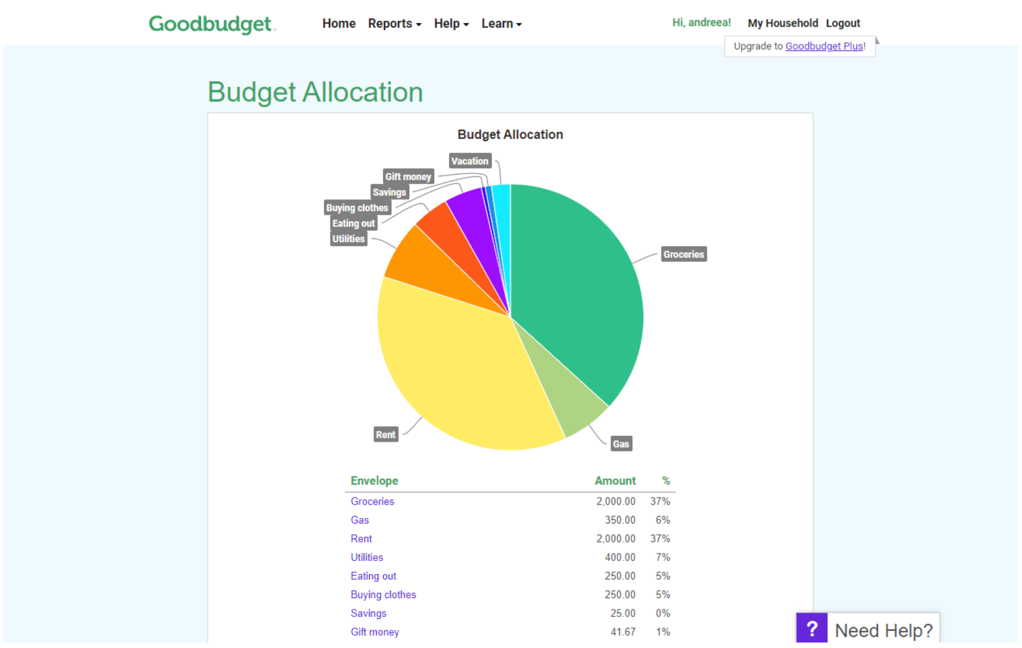
Figure 1.2: GoodBudget - envelopes and transactions

A good aspect of the application is the fact that the incomes and expenses can be automatized and will be added monthly as needed.

When it comes to the layout, GoodBudget does not have a user-friendly interface and setting up a budget is tough without watching their tutorials. Updating and deleting incomes is a troublesome process, as there are too many buttons and pages, but insufficient explanations.

Furthermore, the envelope system will not restrain the user from making unnecessary purchases. It only allows a user to track their expenses and understand their budget through different reports and diagrams.

Overall, the application is too complicated and confusing.



Envelope	Amount	%
Groceries	2,000.00	37%
Gas	350.00	6%
Rent	2,000.00	37%
Utilities	400.00	7%
Eating out	250.00	5%
Buying clothes	250.00	5%
Savings	25.00	0%
Gift money	41.67	1%

Figure 1.3: GoodBudget - budget allocation

1.2.2 NerdWallet

NerdWallet [Ner] offers a wide range of courses and information on topics such as: credit cards, loans, managing money, investing and funding a business. It is a great tool to keep track of your cash flow and has a user-friendly interface.

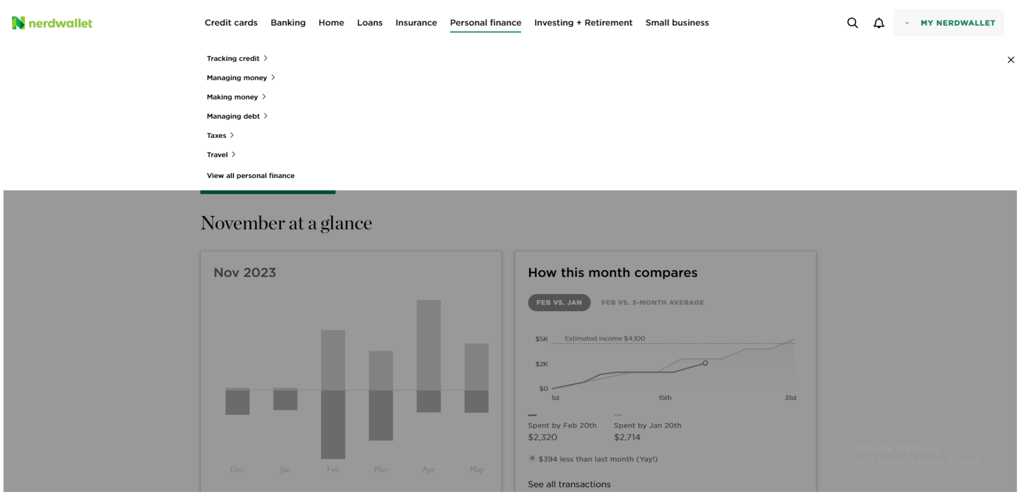
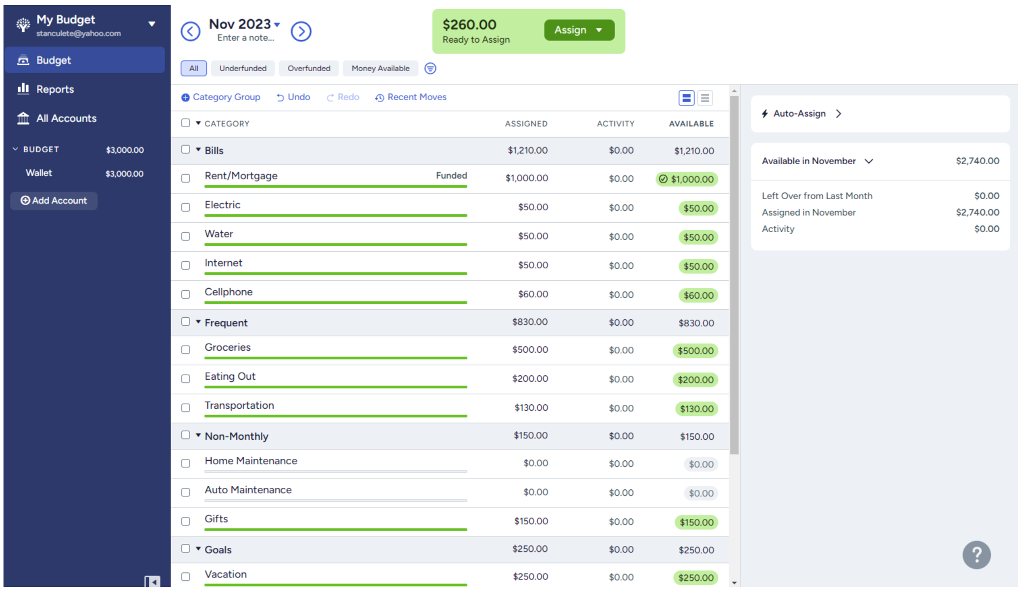


Figure 1.4: NerdWallet

The only and very important downside of it is that the application requires you to have a USA phone number. Therefore, it is not accessible to everyone.

1.2.3 YNAB

YNAB - You Need a Budget [YNA], is a personal budgeting application based on the envelope system. You input your income and start assigning money to different expenses: bills, groceries, vacation, hobbies. The application has a user-friendly interface with a responsive design, and it's very easy to use.



CATEGORY	ASSIGNED	ACTIVITY	AVAILABLE
Bills	$1,210.00	$0.00	$1,210.00
Rent/Mortgage (Funded)	$1,000.00	$0.00	$1,000.00
Electric	$50.00	$0.00	$50.00
Water	$50.00	$0.00	$50.00
Internet	$50.00	$0.00	$50.00
Cellphone	$60.00	$0.00	$60.00
Frequent	$830.00	$0.00	$830.00
Groceries	$500.00	$0.00	$500.00
Eating Out	$200.00	$0.00	$200.00
Transportation	$130.00	$0.00	$130.00
Non-Monthly	$150.00	$0.00	$150.00
Home Maintenance	$0.00	$0.00	$0.00
Auto Maintenance	$0.00	$0.00	$0.00
Gifts	$150.00	$0.00	$150.00
Goals	$250.00	$0.00	$250.00
Vacation	$250.00	$0.00	$250.00

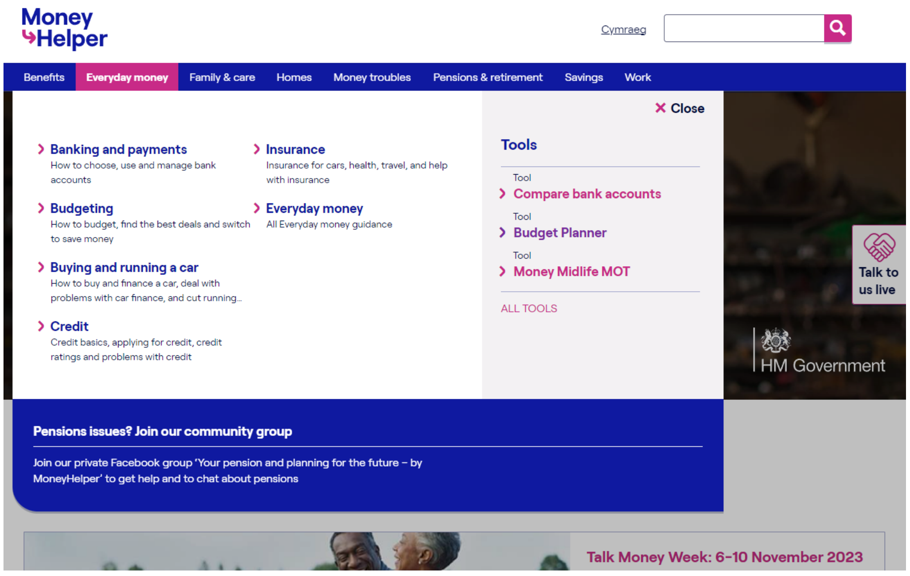
Available in November	$2,740.00
Left Over from Last Month	$0.00
Assigned in November	$2,740.00
Activity	$0.00

Figure 1.5: YNAB - budget

The user can automatically asign money to each category and all the incomes can be automatized. Therefore, it can become a practical tool to keep track of your spending, but in order for it to be effective the user has to buy a subscription and constantly use it.

1.2.4 MoneyHelper

MoneyHelper [Mon] is a cost calculator for everyday use. It offers information on budgeting, savings, insurance, debt, retirement and has many tools such as: budget planner, mortgage, divorce and pension calculator.



Banking and payments
How to choose, use and manage bank accounts

Insurance
Insurance for cars, health, travel, and help with insurance

Budgeting
How to budget, find the best deals and switch to save money

Everyday money
All Everyday money guidance

Buying and running a car
How to buy and finance a car, deal with problems with car finance, and cut running...

Credit
Credit basics, applying for credit, credit ratings and problems with credit

Tool
Compare bank accounts

Tool
Budget Planner

Tool
Money Midlife MOT

ALL TOOLS

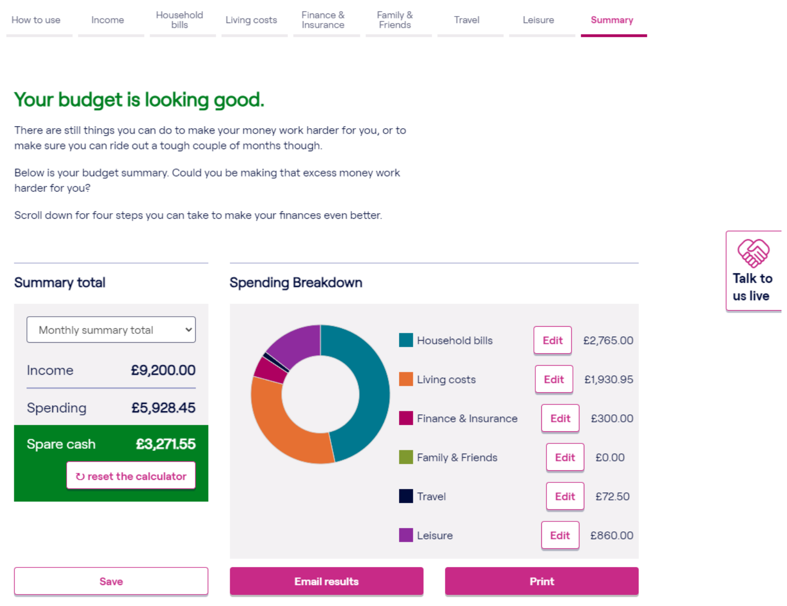
Pensions issues? Join our community group

Join our private Facebook group 'Your pension and planning for the future – by MoneyHelper' to get help and to chat about pensions

Talk Money Week: 6-10 November 2023

Figure 1.6: MoneyHelper

One of its tools, the budget planner, can help you understand your cash flow: where your money are going and how to better save them. All you have to do is fill in all the data and they will generate a summary of your expenses.



Your budget is looking good.

There are still things you can do to make your money work harder for you, or to make sure you can ride out a tough couple of months though.

Below is your budget summary. Could you be making that excess money work harder for you?

Scroll down for four steps you can take to make your finances even better.

Summary total

Income	£9,200.00
Spending	£5,928.45
Spare cash	£3,271.55

Spending Breakdown

🟦 Household bills	Edit	£2,765.00
🟧 Living costs	Edit	£1,930.95
🟥 Finance & Insurance	Edit	£300.00
🟩 Family & Friends	Edit	£0.00
⬛ Travel	Edit	£72.50
🟪 Leisure	Edit	£860.00

Figure 1.7: MoneyHelper - budget planner

The downside of it is that all information are specific to UK, but the tools can still be used by anyone as long as they ignore the currency.

1.3 Proposed solution

Smart Budgeting is a financial-tracking application which aims to overcome the existing flaws in today's market. The main feature which makes it stand out among other applications is the large amount of automatizing put at the user's disposal: from search engines tailored to the customer's needs to receipt scanning and categorizing.

Another unique characteristic of our application represents the reports given to our users. They are highly customizable and highlight all the areas where money can be saved.

We have extracted the key features of each application and compared them in the table bellow. As it can be noticed, our solution covers every major area while remaining accesible to everyone.

Application	Automatisations	Reports	Search engine	Region specific
GoodBudget	X	X	-	-
NerdWallet	X	X	-	X
YNAB	X	X	-	-
MoneyHelper	-	X	-	X
SmartBudgeting	X	X	X	-

1.4 Further reading

In the next chapter, called "**Our Solution**" we will provide an overview of the implementation details and the functionalities of the solution proposed earlier. In the following chapter, we will detail the implementation process of the application, while outlining the key features which make it stand out. Lastly, we will summarise the achieved results and emphasize the directions for improvement and further development.

Chapter 2

Our solution

In the previous chapter, we have defined the key characteristics of an ideal financial tracking application. In the following sections we will provide an overview on how these characteristics would map onto concrete use-cases and we will detail how the application must behave in both expected and unexpected scenarios in order to provide a smooth user experience.

2.1 Functionalities

The user must first create and login into their account in order to use the application. Afterwards, they will be able to manage their finances by adding incomes and expenses, generate reports based on their data and search for cheaper products when making a purchase.

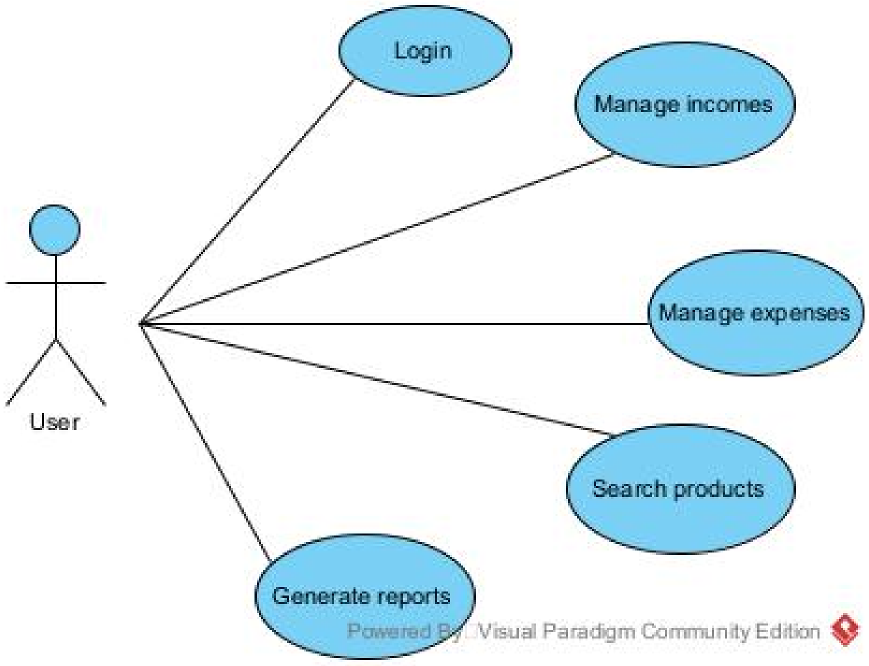


Figure 2.1: Use case diagram

A key aspect which must be taken into account during the implementation of the application is which of the above-mentioned functionalities depend on the user input, and which need to happen automatically. While most of them will make use of a dedicated interface, the reports need to be available to the user 24/7. This is why, on top of the displayed information, the reports should be provided to the user via an email attachment, so that they can be downloaded and stored permanently.

2.1.1 Account registration

The user writes into the registration form data such as username, email and password, which will be all validated. If there is already a user with the same username, an error is displayed. Otherwise, a new account is created and the user is logged in.

Precondition: The account does not exist.

Postcondition: The account details are stored in the system.

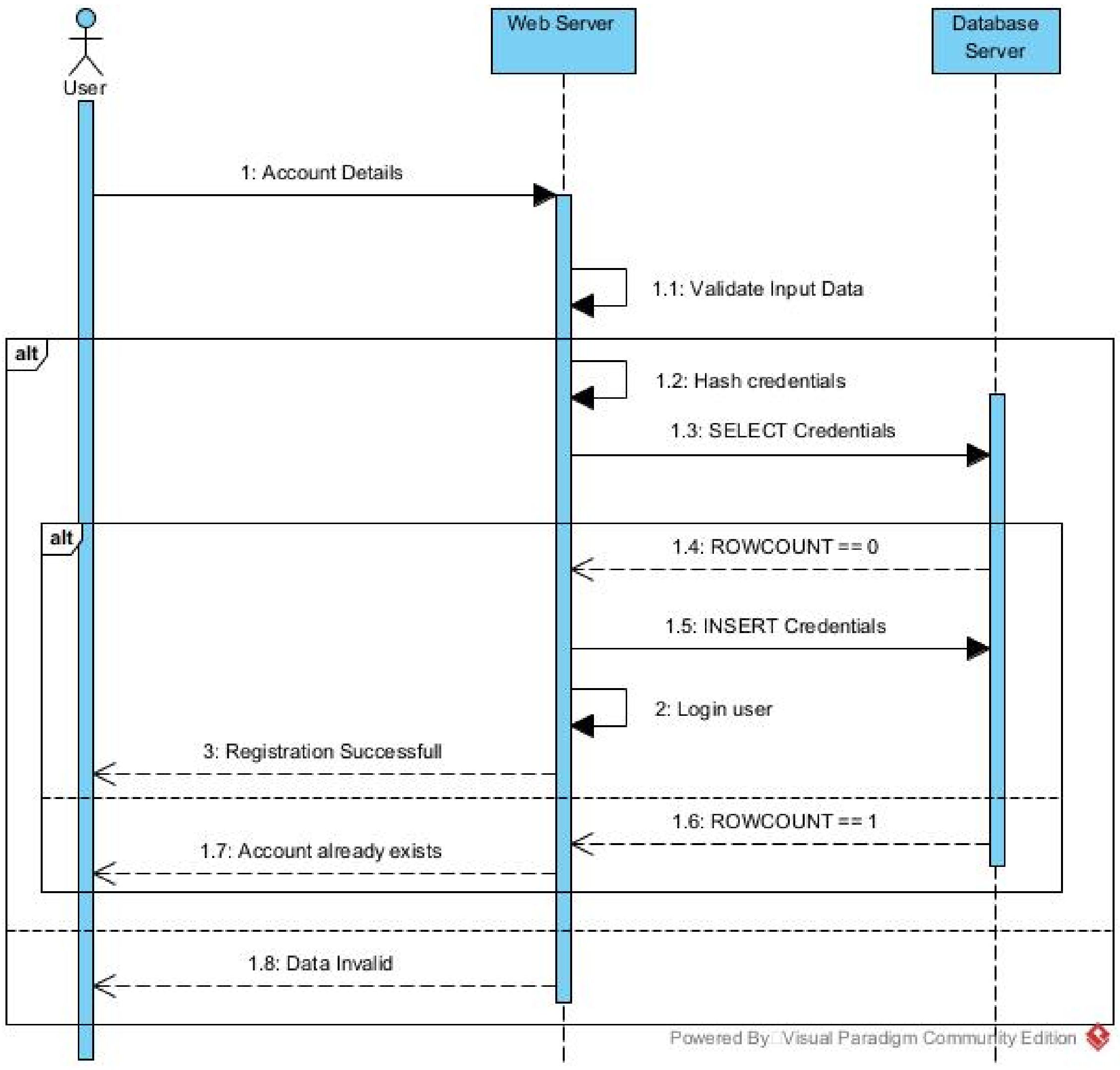


Figure 2.2: Creating a new account

Alternate use-case 1: An account already exists. In this case, the website will display an error notifying the user, that the chosen username is already taken.

Alternate use-case 2: The provided data is invalid. In this case, the website will display an error notifying the user, that the entered data is invalid.

2.1.2 Account authentication

The user writes into the authentication form their username and password. This data will be validated against the list of existing accounts. Upon a successful authentication, any missing periodical incomes or expenses will be added to the database and the reports will be updated accordingly.

Precondition: The account exists.

Postcondition: The user is logged in and all its data it up to date.

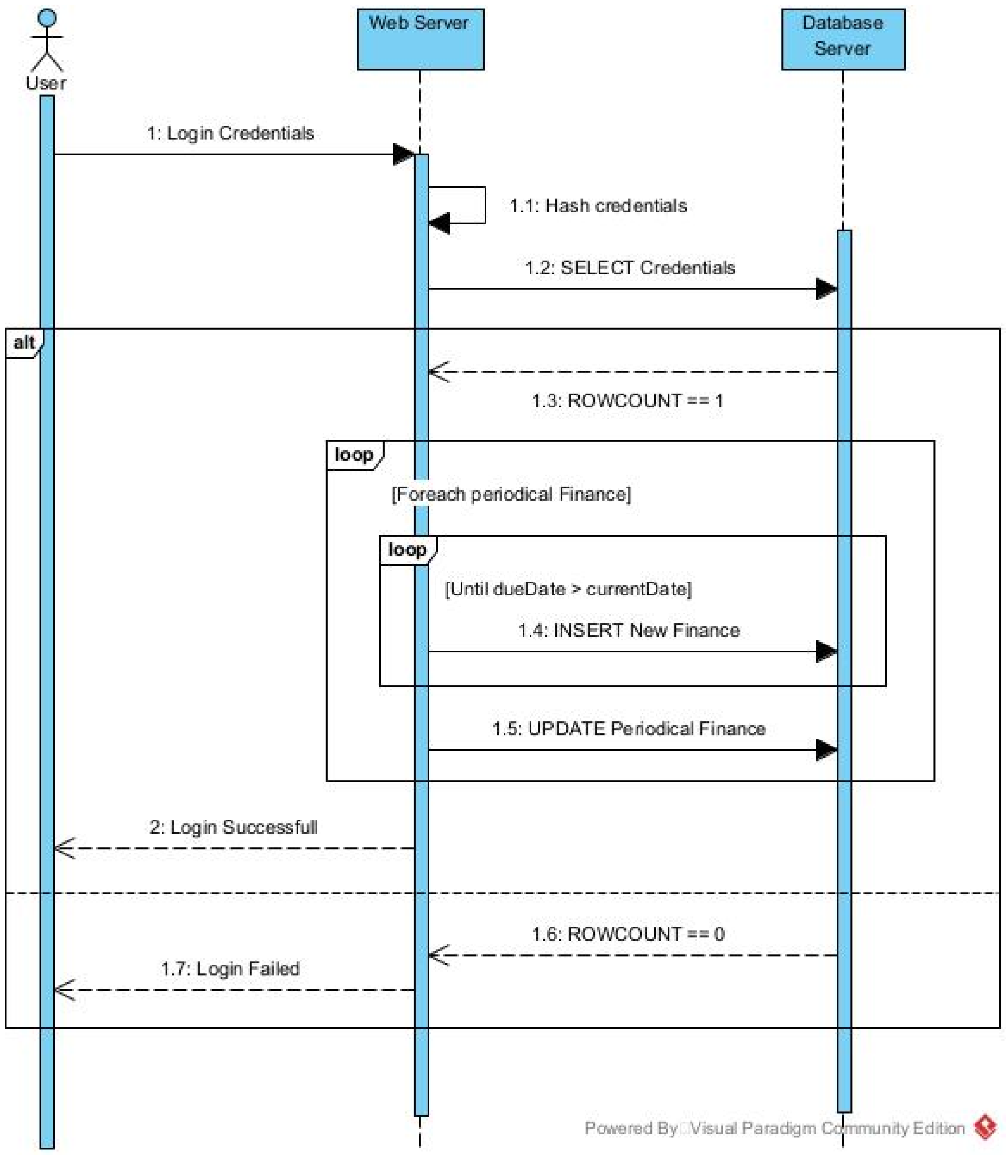


Figure 2.3: Login into account

Alternate use-case 1: The authentication fails. If no user account corresponds to the inputted credentials, the authentication request will be denied and the website will display an error.

2.1.3 Manage incomes

The user chooses to add an income through the application form. They insert details such as the type of income, amount, periodicity and select a currency. If the user marks the current income as being periodic, a scheduler is used to make the income recurrent.

Precondition: The user is logged in.

Postcondition: The Finances table has been updated.

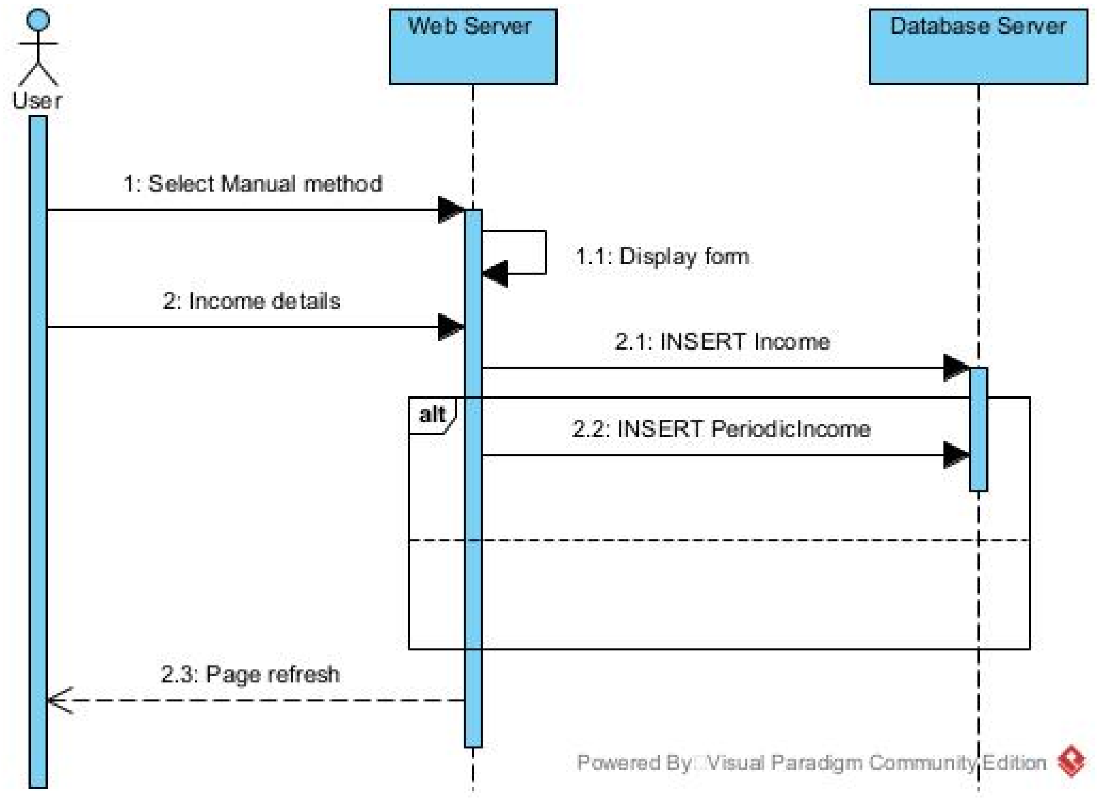


Figure 2.4: Adding an income

The other operations included in this use-case are income deletion and update. Given the simplicity of both operations, they have not been detailed individually. The flow of events for both operations is identical to the one depicted above, the only difference being in the final operation performed on the databse (DELETE or UPDATE instead of INSERT).

2.1.4 Manage expenses

There are several ways for a user to update their expenses. Firstly, they can insert custom amounts via a dedicated form. However, that process is long and tedious, which is why the user can upload several receipt images. Then, the expenses will be extracted automatically from each receipt uploaded.

When dealing with receipt scanning, the application offers support for two methods of receipt scanning: by photographing an actual receipt, and by uploading receipt images in bulk.

From a design point-of-view, each separate method of adding expenses consists its own use-case. This is why in the paragraphs below, all of the three aforementioned methods have been detailed individually.

Manually via a dedicated form

The user chooses to add an expense through the application form. They insert details such as the type of expense, amount, periodicity and select a currency. If the user marks the current expense as being periodic, a scheduler is used to make the expense recurrent.

Precondition: The user is logged in.

Postcondition: The Finances table has been updated.

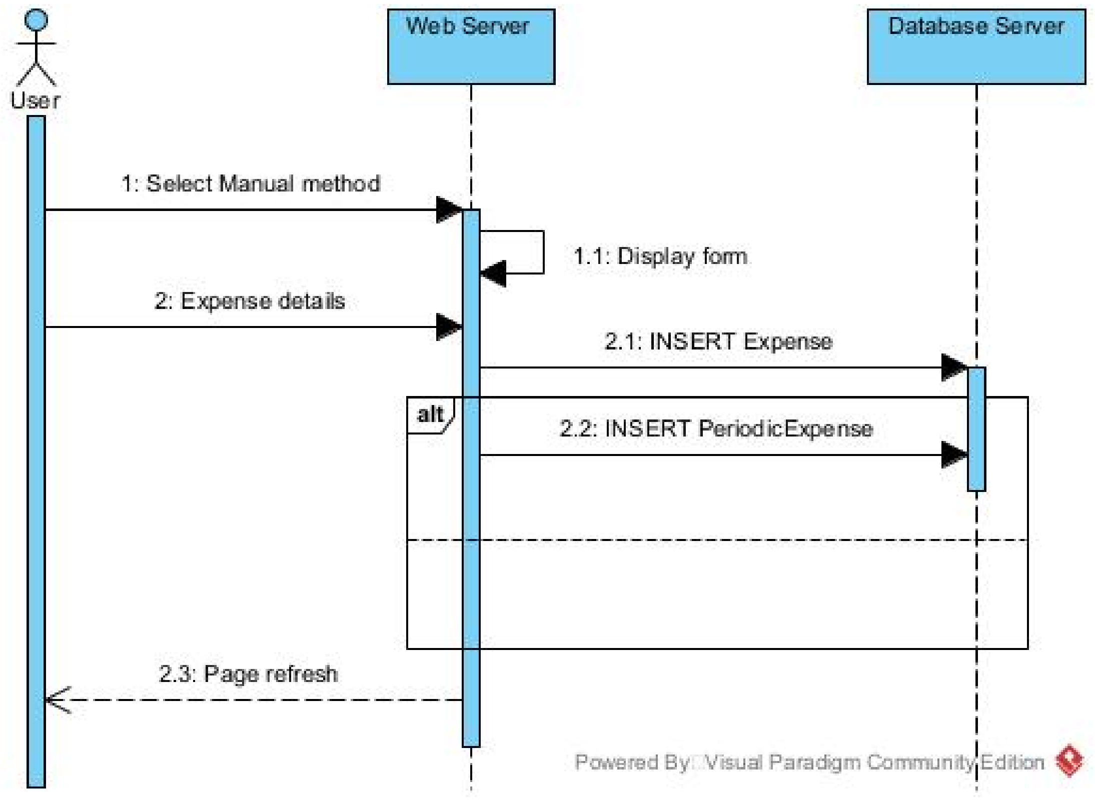


Figure 2.5: Adding an expense

The other operations included in this use-case are income deletion and update. Given the simplicity of both operations, they have not been detailed individually. The flow of events for both operations is identical to the one depicted above, the only difference being in the final operation performed on the databse (DELETE or UPDATE instead of INSERT).

Automatically via receipt scanning

The user chooses to take a picture of their receipt through the application. The file received through the camera feed is sent to a third party OCR service via a POST request, which returns all data from the receipt through a JSON object. The received information will be formatted and displayed to the user for confirmation. After the user has made the necessary corrections, the expense will be saved in the database.

Precondition: The user is logged in and camera access is allowed

Postcondition: The user's expenses have been updated with the data from the receipts.

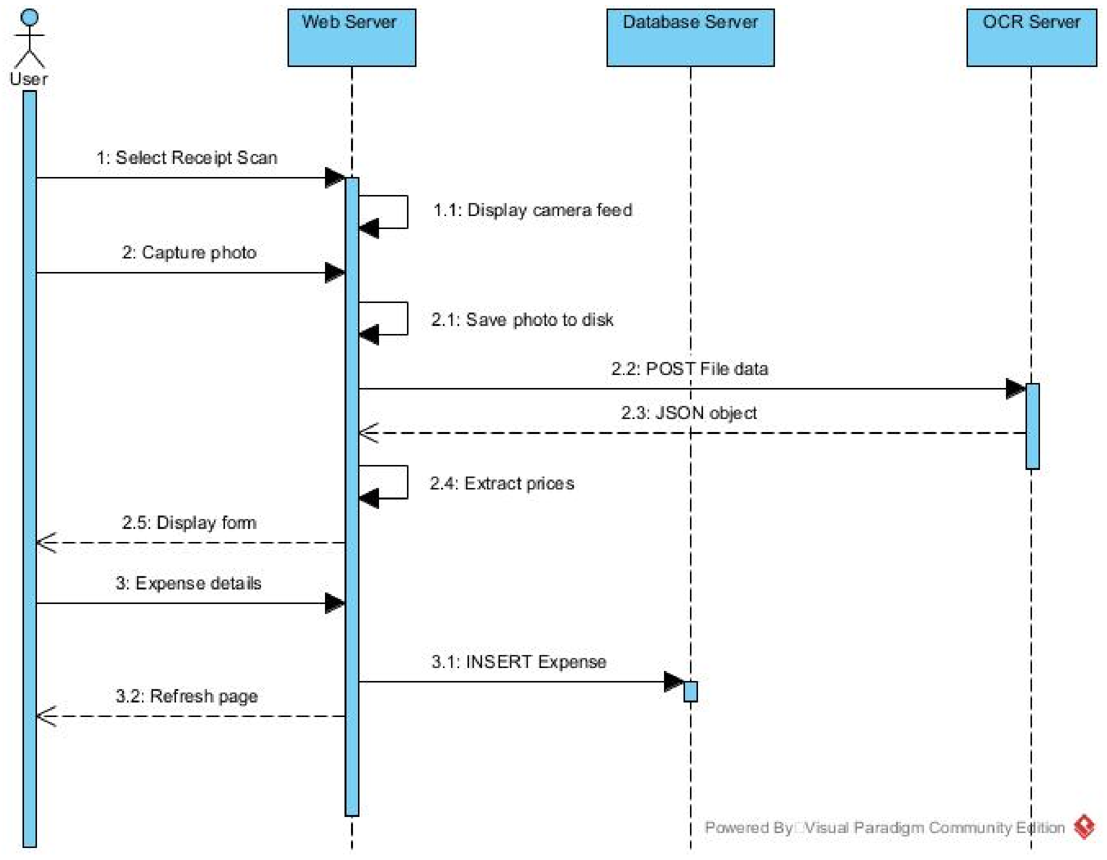


Figure 2.6: Scanning a receipt

Alternate use-case 1: The OCR server is unavailable. In this scenario, the application will return an empty form, and the user will have to input the expense manually.

Automatically via bulk upload

The user chooses to upload multiple files from their device, which are sent to a third-party OCR service via a POST request. The OCR service will return a JSON object containing the receipt data for all the files. The data will be formatted and displayed to the user for confirmation. After the user does the necessary adjustments, the expenses will be saved in the database.

Precondition: The user is logged in.

Postcondition: The user's expenses have been updated with the data from the receipts.

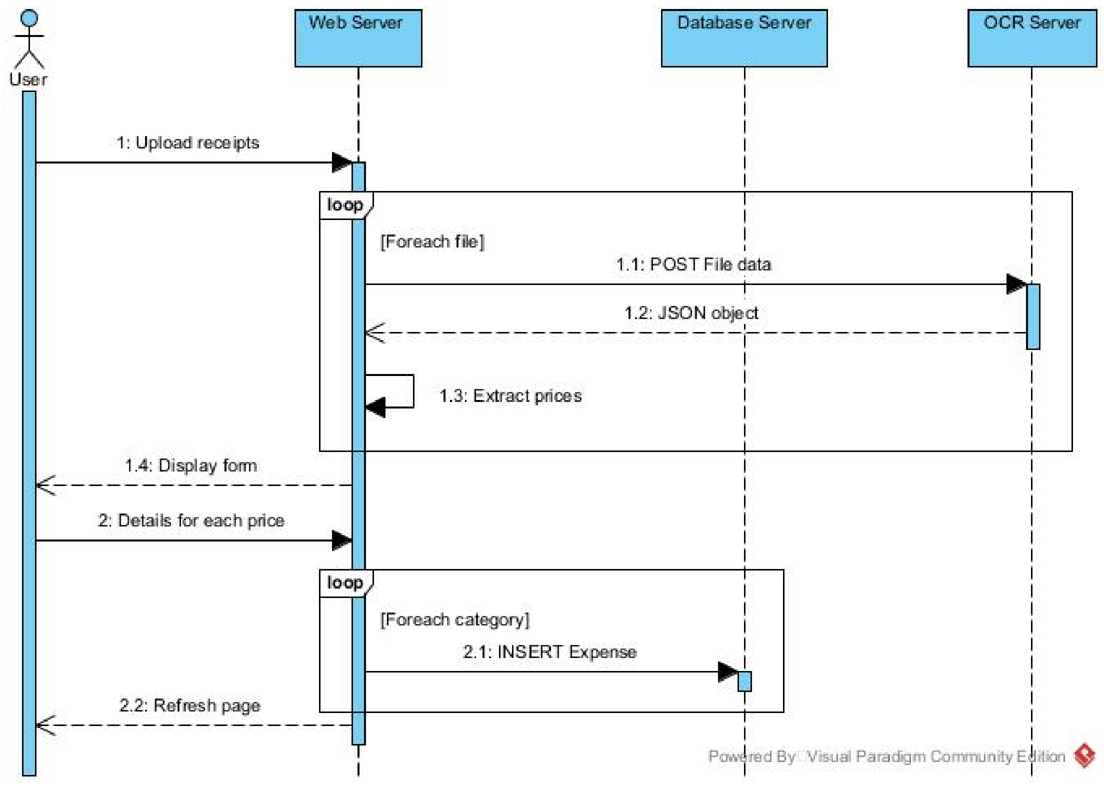


Figure 2.7: Scanning multiple receipts

Alternate use-case 1: The OCR server is unavailable. In this scenario, the application will return an empty form, and the user will have to input the expense manually.

2.1.5 Generate reports

The user asks for the generation of personalised reports. The user's finance data is retrieved from the database. Then, the data is grouped into categories and the income, expenses and overall balance charts are constructed in order to help the user identify the areas for improvement. All these charts and data will be put together in a Microsoft Excel file which will be attached to an email sent to the user.

Furthermore, the computed charts will be displayed in a dedicated interface at all times, and provide insights into the finance data gathered for the chosen month.

Precondition: The user is logged in.

Postcondition: The user receives an overview of their finances.

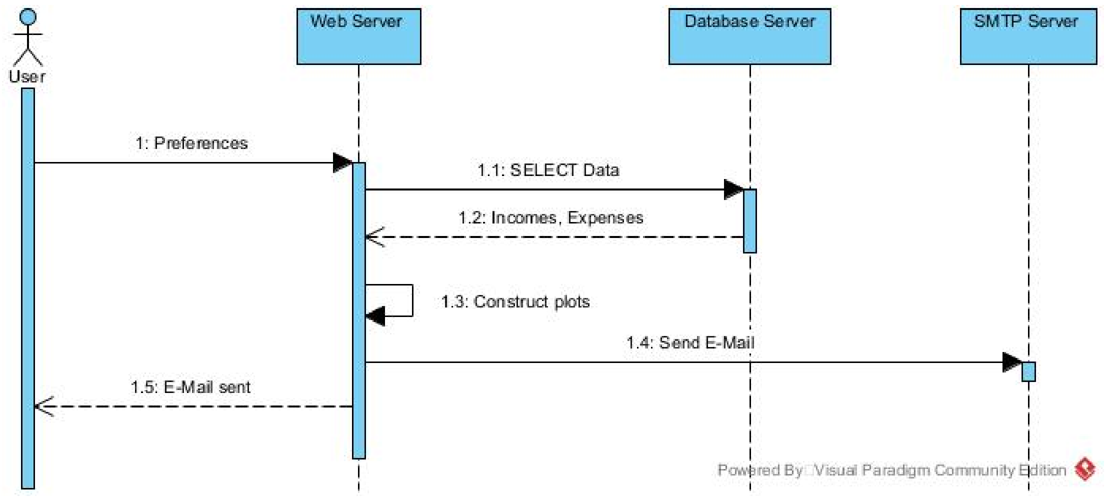


Figure 2.8: Generating reports

Alternate use-case 1: The SMTP server is not responding. In this scenario, no email will be sent to the user with the generated graphs.

2.1.6 Search for products

The user uses the search bar in order to search for cheaper products. The application will take the user input and feed it into a web-crawler which will search popular 3rd party shopping websites for the desired product(s). The crawler will extract from the page's HTML content the details of each product (name, price, hyperlink) and sort them in an ascending order by their price. The first couple of products will be then displayed to the user.

Precondition: The user is logged in.

Postcondition: The cheapest products found are displayed.

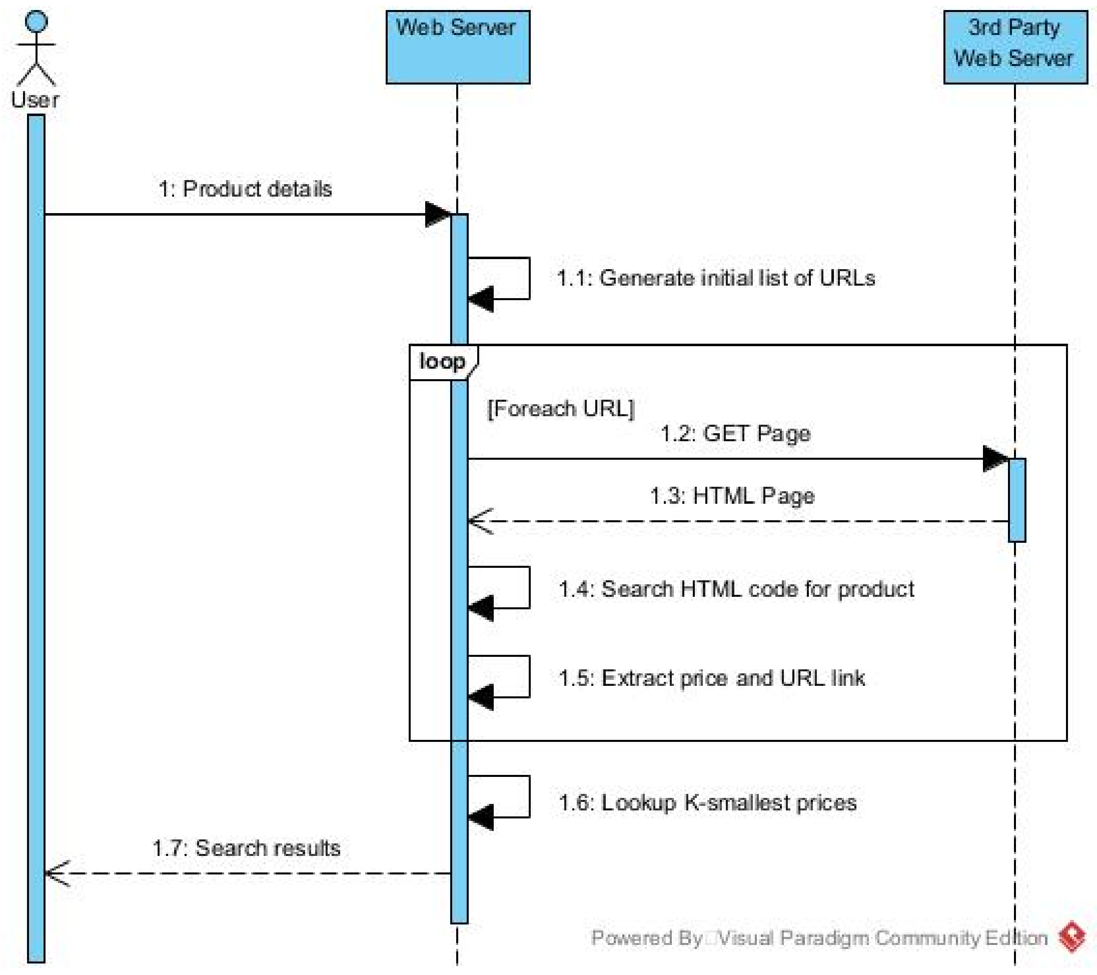

User
Web Server
3rd Party Web Server
1: Product details
1.1: Generate initial list of URLs
loop
[Foreach URL]
1.2: GET Page
1.3: HTML Page
1.4: Search HTML code for product
1.5: Extract price and URL link
1.6: Lookup K-smallest prices
1.7: Search results
Powered By Visual Paradigm Community Edition

Figure 2.9: Searching for products

Alternate use-case 1: The connection to the 3rd party web servers is denied. In this scenario, the process will be stopped and an empty list of products will be displayed to the user.

2.2 Database

The database is made out of seven tables. The User table is part of the Django Authentication package and it connects to the other three tables. There are several other tables which Django uses, but have not been displayed in the diagram bellow.

The *UserSettings* table stores data relevant to the user's use of the application: profile picture and default currency for their finances.

The *Currencies* and *Expenses* tables store the relevant categories that are used into the application. This way, whenever there is a need for a new expense or for a new currency, the respective data will be added straight into the database and the application will automatically update with the new changes.

The *Expenses* table is also connected to the *Advices* table, which stores suggestions for each expense that is used into the application. These recommendations are later given to the user via email, based on the data analysis performed for the current month.

The *Finances* and *PeriodicalFinances* tables store each user's income and expenses. The first table allows users to keep their finances in one place and see them in the main page after login. Meanwhile, the *PeriodicalFinances* table is used to keep all the periodical income and expenses for a better and easier access. Whenever a user will login into the application, this table will be checked for any due income and expenses.

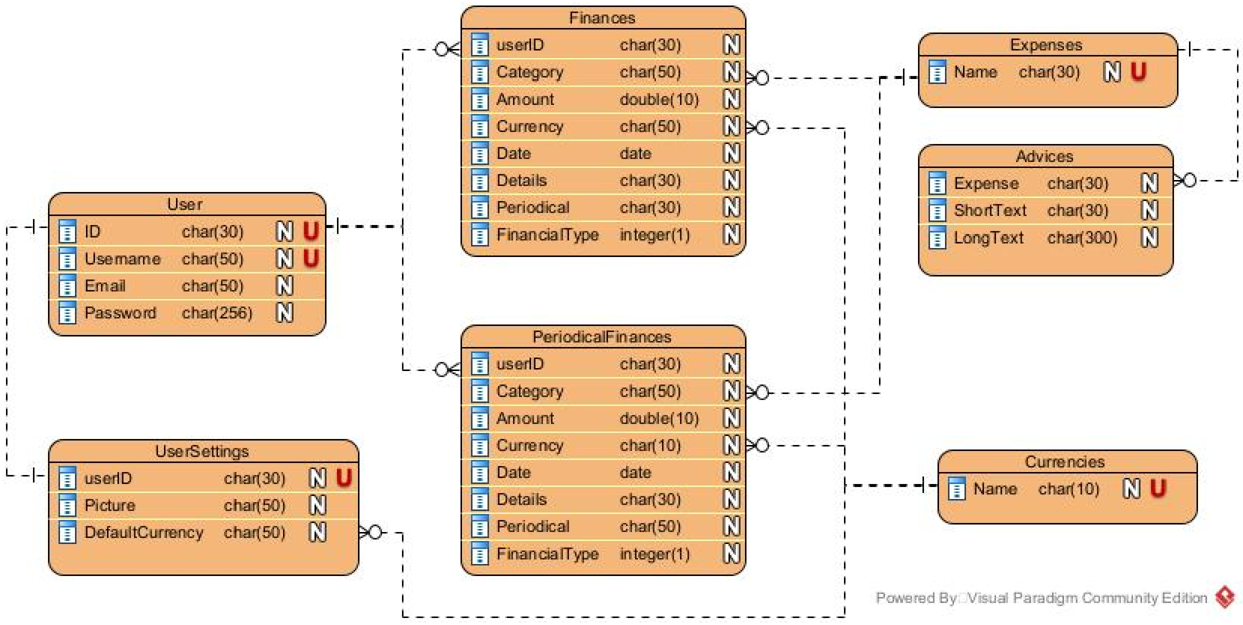


Figure 2.10: Database structure

Chapter 3

Implementation

In today's world, web applications are most sought after for their accessibility and how facile their implementation process is. Furthermore, the large number of free open-source frameworks that exist on the market have made developing web applications that much easier. These frameworks usually feature a modular architecture which facilitates code reusability, and implement security features that offer protection against malicious attacks.

When choosing a framework the developer must take into consideration the programming language they are comfortable with, the documentation, the flexibility and its features, as well as the application's target audience, goals and complexity.

Considering these, it should be noted the Smart Budgeting web application is designed with the following objectives in mind:

- Intuitive user interface
- Receipt scanning
- Data export
- Money saving recommendations

3.1 Technologies

For the back-end development, we found Django[Djaa], a high-level web framework based on Python[Pyt], to be the best fit. It allows to separate the application's data into models, the user interface into templates and the logic flow into views. Thanks to Django, database operations can be done without directly dealing with SQL queries.

Each database table used for the application starts as a model, a Python class with fields and methods, and constructed at the request of the developer. This gives the chance to better plan for the data and to easily access or modify it. In views, class methods must be called in order to retrieve data and send them further to the templates to be handled. All content sent to the templates is dynamically displayed into tables and charts with the help of Django Template Language [Djab], a templating engine similar in structure to Jinja[Jin].

For an optimum design and user experience, we have chosen Bootstrap[Boo] as the main front-end framework. It simplified the process of creating responsive HTML pages and customizing the predefined elements to fit the desired theme.

3.2 Structure

For a clean navigation and understanding of the code, the files have all been separated into folders, based on categories and purpose. The figure below outlines the project's main file structure:

```
SmartBudgetingApp
    |--- models
    |--- static
    |--- templates
    |--- utils
    |--- views
```

Django's template language helped with the modularisation of the code. This way, each page is composed of multiple components, all with their own files and structure. The code can be reused over and over again, withing multiple pages.

3.3 Pages

An intuitive user interface has a different meaning for each type of application. For a financial tracking application, as a simplified flow is desired, the need to have as few pages as possible arises. The design of pages has been kept clean and simple.

The five main pages for the Smart Budgeting application, as well as the flow between them, can be observed in the following diagram.

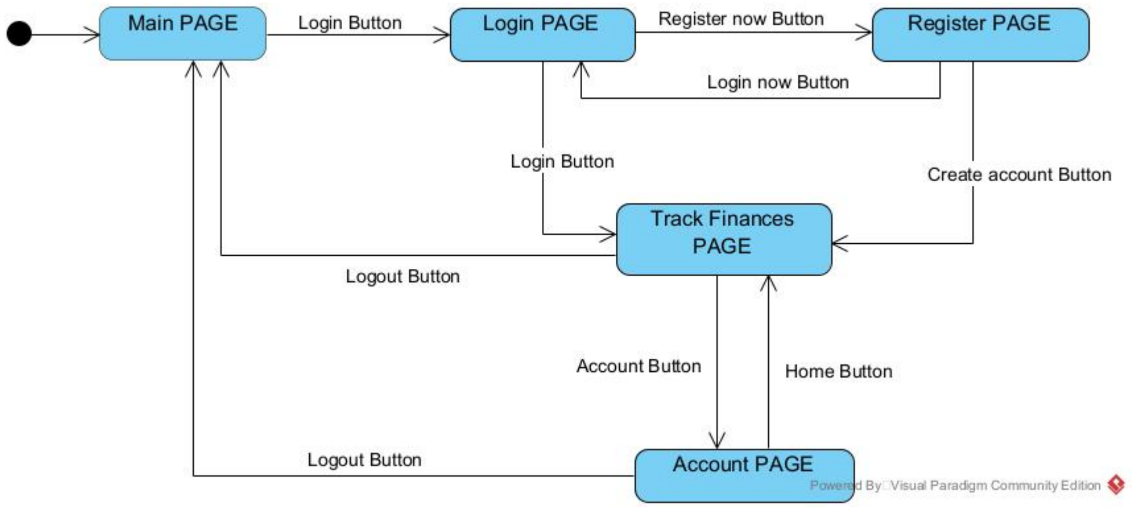


Figure 3.1: flow of pages

3.3.1 Main page

Upon opening the application, the user will be directed to the main page, which has as informative role. From this page they have the option to Login into their account.

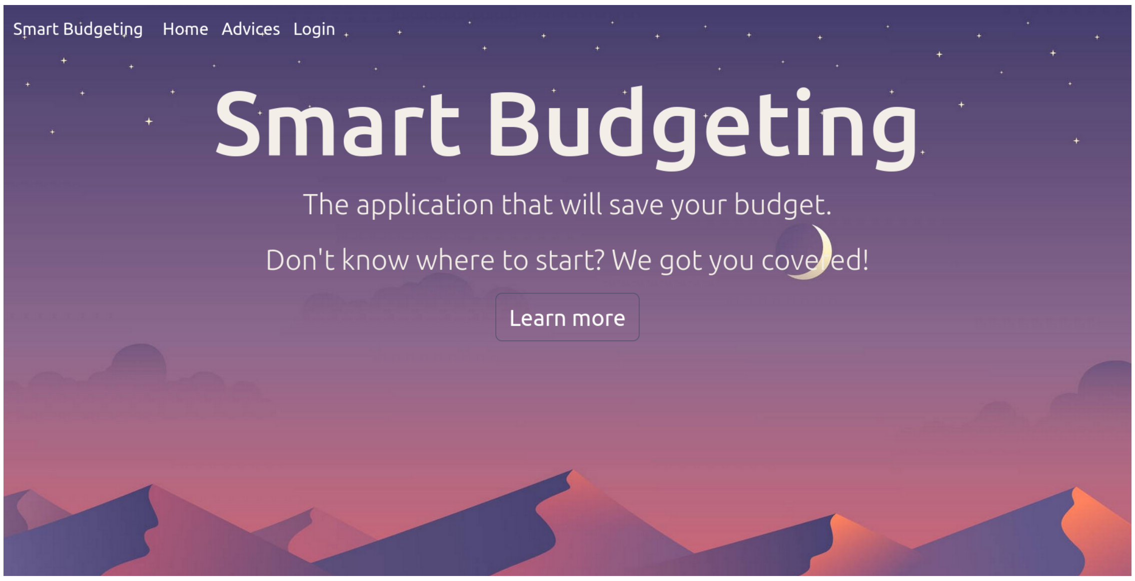

Smart Budgeting
Home
Advices
Login
Smart Budgeting
The application that will save your budget.
Don't know where to start? We got you covered!
Learn more

Figure 3.2: Main Page

3.3.2 Register and Login

The Login and Register pages are similar in structure. They have the purpose of allowing the user to create an account and use it to login into the application. After logging in, the user can access all functionalities the application has to offer.

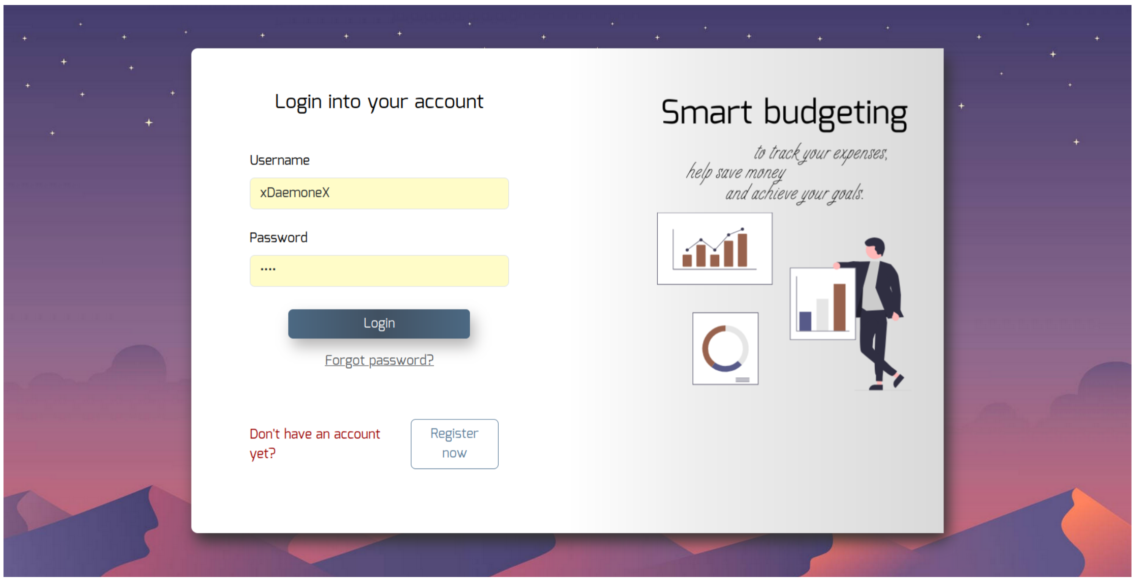
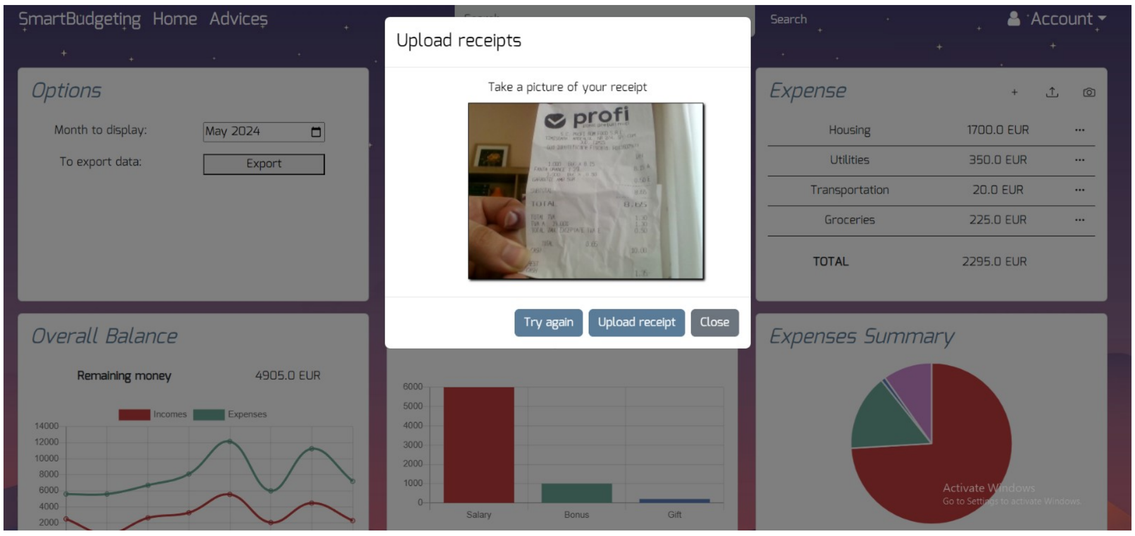


Figure 3.3: Login page

3.3.3 Finances tracking page

The main functionalities of the application all lie in the finances tracking page. It allows users to add sources of income, as well as expenses, through multiple forms.

While all incomes must be added manually, most expenses can be added by scanning receipts. This effortless approach saves the users time and energy.



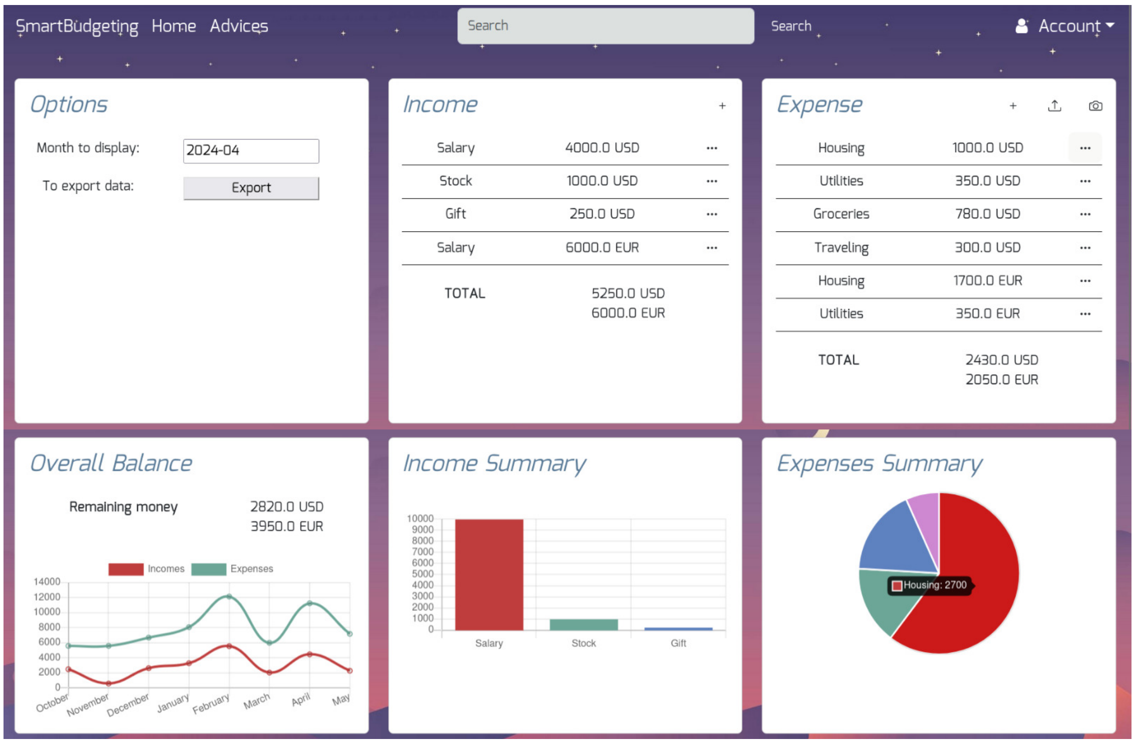
Month to display:	May 2024
To export data:	Export

Housing	1700.0 EUR
Utilities	350.0 EUR
Transportation	20.0 EUR
Groceries	225.0 EUR
TOTAL	2295.0 EUR

Remaining money	4905.0 EUR

Figure 3.4: Enter Caption

After being added, the incomes and expenses can be seen in this page very clearly in their respective sections. Based on this data, each user can see an overview of their finances through the corresponding charts.



Month to display:	2024-04
To export data:	Export

Salary	4000.0 USD
Stock	1000.0 USD
Gift	250.0 USD
Salary	6000.0 EUR
TOTAL	5250.0 USD 6000.0 EUR

Housing	1000.0 USD
Utilities	350.0 USD
Groceries	780.0 USD
Traveling	300.0 USD
Housing	1700.0 EUR
Utilities	350.0 EUR
TOTAL	2430.0 USD 2050.0 EUR

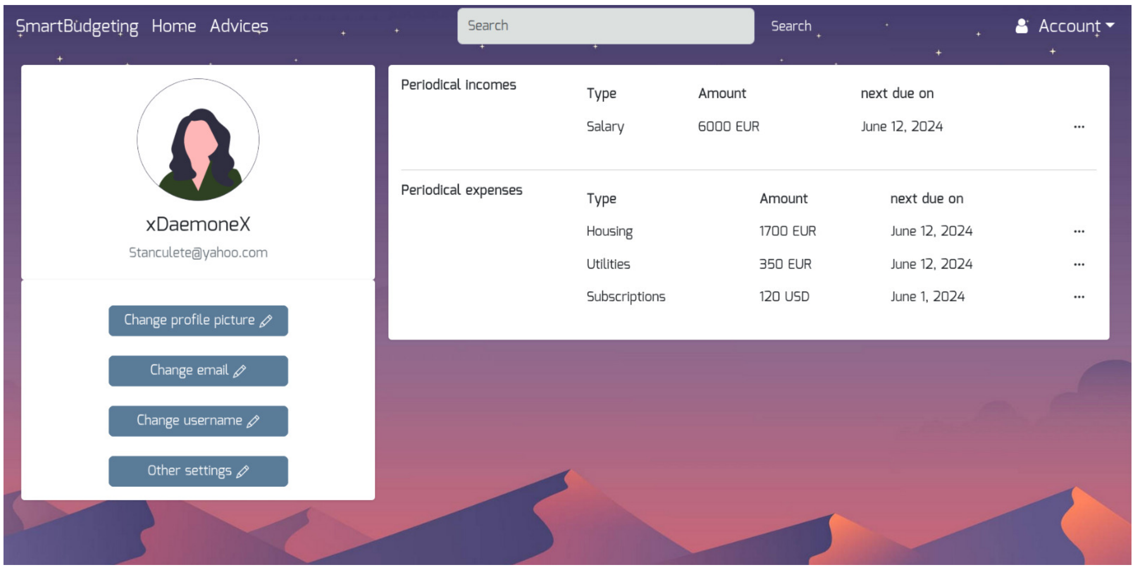
Remaining money	2820.0 USD 3950.0 EUR

Figure 3.5: Finances tracking page

At the end of the month, each user receives on their account email an excel export with all incomes and expenses for the current month, as well as advice on how to improve their finances.

3.3.4 Account page

From the account page, each user is able to check their periodical incomes and expenses, as well as edit or delete them. They can change their account information, such as: main currency, username, email.



Periodical incomes

Type	Amount	next due on	
Salary	6000 EUR	June 12, 2024	···

Periodical expenses

Type	Amount	next due on	
Housing	1700 EUR	June 12, 2024	···
Utilities	350 EUR	June 12, 2024	···
Subscriptions	120 USD	June 1, 2024	···

Figure 3.6: Account page

3.4 Testing and Results

Testing is an important step in the development process of an application. All the components, from pages to functionalities, have to be rigorously tested to reveal problems within the application and check for untreated cases.

The most attention has been paid to the following tasks:

- Implementation of a responsive application
- Correctly updating the user's periodical data
- Fully working functionalities
- Accurate data analysis

Upon testing, the application has been found to be fully responsive according to the user's screen. All data is shown correspondingly in their assigned space, and when there is not enough data, the application will state as much.

Regarding the update of the user's periodical data, a fault has been discovered. As the update only takes place when the user uses the login form, as long as the user stays connected, their periodical finances will not be revised. There have been found multiple solutions to this problem, like disconnecting the user when they close the application, but this would cause an inconvenience for them. The best approach might be adding an update button to the main page, so when the users notices a discrepancy, they can manually ask for their data to be updated.

All the relevant functionalities have been implemented. While most of them show no error or special cases, the same can not be said about the receipt scanning and the web crawler.

The free API used to scan the receipts only allows a few scans per day, so when it has reached its limit, it will throw an error. Of course, this can be fixed by using a paid API.

The web crawler works correctly, but from time to time the sites it uses will ask for a bot checker. That can not be helped as it is a third party.

Analysing the data has been challenging, but the task has been completed. When it comes to the expenses, the thresholds and advice can still be improved.

Overall, the objectives for this application have been met.

Chapter 4

Conclusions

To sum everything up, we have developed a financial tracking application which offers a new and straightforward way of keeping a budget by automatizing the majority of the processes involved. This is accomplished through periodical finances and automatic receipt scanning, simplifying the tasks involves and allowing users to set up regular monitoring of their finances. Additionally, users can easily export their financial data and receive personalized insights based on the user's spending patterns.

4.1 Future work

Nonetheless, there are numerous opportunities to further elevate the user experience and functionality of our application. Integrating a feature that allows users to link their credit cards directly with the application is a great way to automatically import all purchases, eliminating the need for manual entry and ensuring up to date expense tracking.

Moreover, a variety of detailed reports on incomes and expenses, combined with a more enhanced data analysis, would provide users with a more comprehensive view of their financial situation and offer deeper insights into their spending habits.

In terms of design improvements, while the current model is straightforward, it can be overwhelming and cluttered, especially as more features are added. To improve the user experience, functionalities can be divided into multiple pages and categorized by incomes, expenses, reports and settings. This multi-page layout would allow users to easily access and manage their finances without feeling overpowered by all the data shown.

In addition, a feature that allows currency exchange would significantly enhance its utility for users who deal with multiple currencies. It would benefit travelers and international business users by providing a higher versatile financial management tool across different currencies.

Overall, these design improvements and new features would not only make the application more user-friendly but also more powerful and adaptable to the evolving needs of its users.

Bibliography

[AA23] Masrullah Agustan-Firmansyah Rahmat Aditya Andi Arman, Mira. Financial literacy and assistance in compiling independent financial reports using expense iq money manager. *Technium Sustainability*, 4:7–12, 2023.

[ACL21] Josephine Kass-Hanna Angela C. Lyons. A methodological overview to defining and measuring“digital”financial literacy. 2021.

[Boo] Bootstrap. `https://getbootstrap.com/`.

[Djaa] Django. `https://www.djangoproject.com/`.

[Djab] Djangotemplatelanguage. `https://docs.djangoproject.com/en/5.0/ref/templates/language/`.

[Goo] Goodbudget. `https://www.goodbudget.com/`.

[Jin] Jinja. `https://pypi.org/project/Jinja2/`.

[Mon] Moneyhelper. `https://www.moneyhelper.org.uk/en`.

[Ner] Nerdwallet. `https://www.nerdwallet.com/?trk=nw_gn_6.0`.

[Pyt] Python. `https://www.python.org/`.

[TK23] Eusebio Scornavacca Tero Vartiainen Tiina Koskelainen, Panu Kalmi. Financial literacy in the digital age—a research agenda. *The journal of consumer affairs*, 57:507–528, 2023.

[YNA] Ynab - you need a budget. `https://www.ynab.com/`.