

Project Research Document

Household budgeting web app

X00175685 – Rachel Ring

## **Section 1: Detailed Discussion**

The project's core objective is to create a user-friendly household budgeting web application. This web-based platform will enable users to gain insights into their income and expenditures within Ireland, empowering them to manage their finances effectively. It is important to note that this application will not have any access to users' bank accounts

**Data Visualization:** The web app will provide a comprehensive dashboard showcasing data related to household income, expenditures, and savings throughout Ireland. These insights will be derived from open-source APIs provided by the government of Ireland.

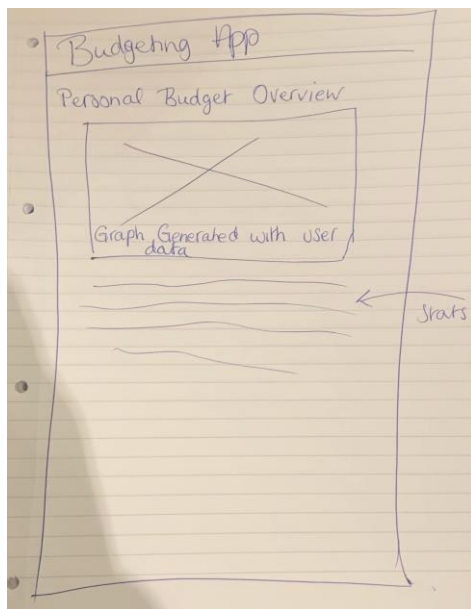
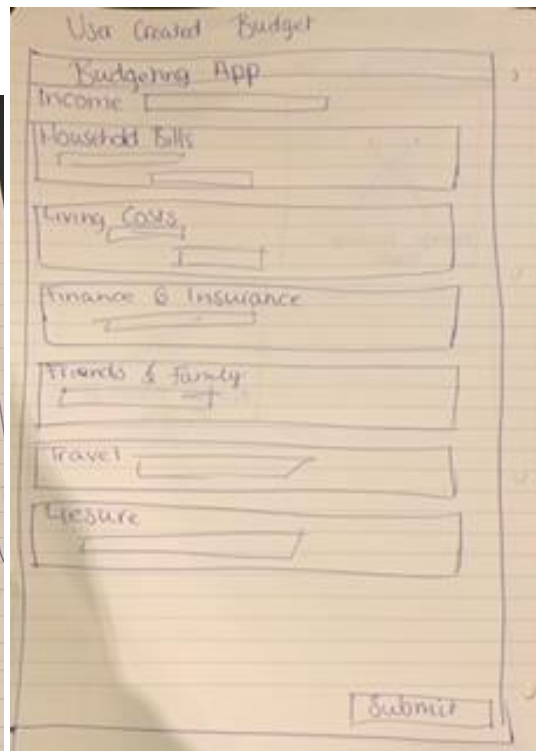
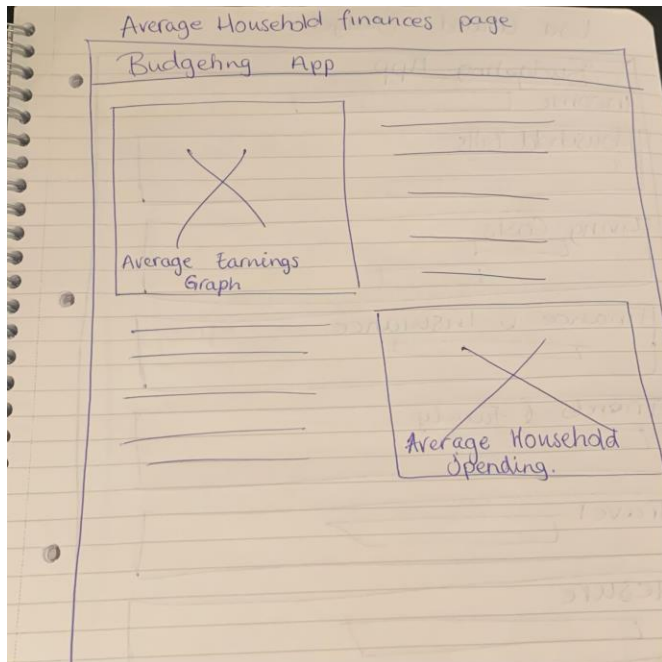
**Personal Budgeting:** Users will have the ability to input their income and expenses, enabling them to create a clear overview of their budget and spending habits. The aim is to simplify the budgeting process, offering a user-friendly experience that eases the potential overwhelm associated with managing finances.

**Expense Categories:** Spending will be categorized into key areas, including Household bills, Living costs, Finance & Insurance, Family & Friends, Travel, and Leisure. These values will be inputted by the user.

**User Accounts:** To enhance the personalization of the experience, users can create accounts within the web app. This feature allows them to not only view their budget information as needed. There will be two types of users for this web app; guest and a registered user. A registered user will need to sign into their account to view their budget. A guest user will be able to create a household budget but will not be able to save it. All users can view the statistics dashboard.

**Project Scope:** The app will exclusively display data sourced from open-source APIs, ensuring data transparency and real-time updates as provided by the APIs. The application will not collect financial data from users. It is designed to offer insights into spending habits and will not establish any connections with bank accounts.

**Outline of the budget planner UI:** A user will be able to fill out the budgeting form. This form will have multiple sections in it. The user will be asked to log in their income, Household bills, Living costs, Finance & Insurance, Family & Friends, Travel, and Leisure. After the user has entered these details, they will be shown a chart to show where their money is going and how much money they have left over. A registered user can save and view their old budget.



## Section 2: Existing Applications in this domain

- **NerdWallet:** <https://www.nerdwallet.com/article/finance/budget-worksheet>

Comparison: NerdWallet offers a quick Budget Planner spreadsheet on their website. This tool allows users to break down their income and expenses using a form. While the form has a lot of fields, this layout could be beneficial for our user's personal budgeting needs.

Monthly Breakdown: NerdWallet's tool conveniently breaks down expenses on a monthly basis, providing users with an organized approach.

- **Google sheets**

Google sheets has a few good budgeting templates for households. The templates are quite simplistic, easy to follow and very user friendly. This web app should be easy and very user friendly and will probably take inspiration from these templates

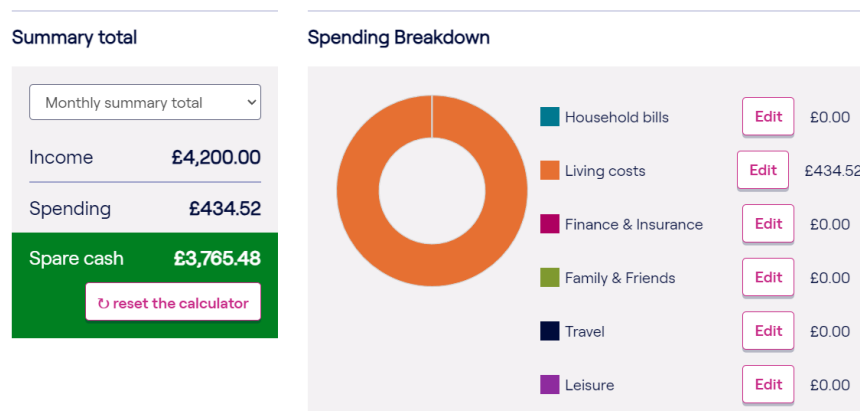
- **CSO simple statistics:**  
<https://www.cso.ie/en/statistics/housingandhouseholds/householdbudgetsurvey/>

I like how they quickly show all the information needed in a way readable to most users. However, I think the dashboard section of my web app will be a little bit more with graphs and less pictures. This link is not an application but a statistical analysis of the data from some of the APIs that are listed as potential data sources.

- **Money Helper:** <https://www.moneyhelper.org.uk/en/everyday-money/budgeting/use-our-budget-planner>

This website lays out a great budget planner and groups expenses and income into very effective groups. This web app would be the gold standard that this project would aim to achieve. However, with the time restrictions placed on this project it is not possible to recreate this budget planner. It would still be an effective source to use to model the budget planner around.

The breakdown part of their UI is fantastic. This project will strongly take after Money Helper UI:



*Money Helper Budget Summary.*

### Section 3: Platform, Technologies and Libraries

The client side of the app with Angular and Typescript. The backend, server side, will be written in Golang.

The client-side will leverage Angular and Typescript for the user interface and will follow a single-page application architecture for a seamless user experience. The client side will communicate with the server side through APIs defined on the server. Angular Material will be utilized to maintain a uniform look and feel throughout the web app. Accessibility features are integrated into Angular Material Components. Accessibility is a concern for all users, be it that the user is visually impaired, hard of hearing, colorblind, or any user that uses assistive technology, it is important that the user can traverse throughout the application with these assistive technologies. Angular material provides default accessibility features (API found [here](#)). The Angular version used will be 16.2.7. and the following extensions will be used in the development process; Angular Language Service v16.1.8 (documentation [here](#)), Angular 1 JavaScript and TypeScript Snippets: v0.2.1 (documentation [here](#)). Auth0 will be integrated for user authorization.

The server side will be developed using Golang version go1.20.2. We will use Golang's net/http package to create a web server. Golang has many useful libraries and there is an abundance of documentation on how to use go for Web development.

User data and budgets will be stored in a MySQL database hosted on Google Cloud SQL. It was chosen for its compatibility with the Golang environment.

Testing is very important to ensure that the application will work as expected and to ensure quality before deployment. Golang has a testing package that will be used for writing unit tests and the documentation can be found [here](#). Angular and Typescript can be tested using the jasmine testing framework and the documentation can be found [here](#). Jasmine tests are run using the Karma test runner and documentation can be found [here](#). Unit tests will be written and run each time a new piece of functionality is developed to ensure quality and to try to avoid technical debt by having code that is untested. It is important to use more than one type of testing. Selenium can be used to perform web functional testing and automate it.

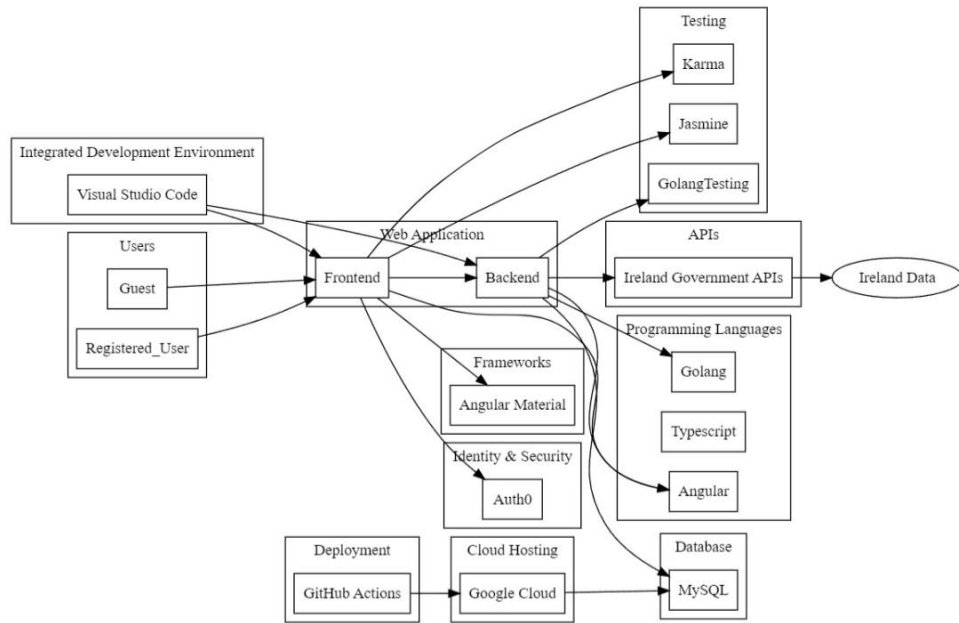
The web app will be deployed on Google Cloud using GitHub actions for automated deployments.

This project will be developed in Visual Studio Code version 1.77.

#### APIs

<https://data.cso.ie/table/EHQ03> - this API shows data for the average earnings of a person working in Ireland. I could process it and display graphs to show the user what the average is nationally.

[https://ws.cso.ie/public/api.restful/PxStat.Data.Cube\\_API.ReadDataset/HS236/JSON-stat/2.0/en](https://ws.cso.ie/public/api.restful/PxStat.Data.Cube_API.ReadDataset/HS236/JSON-stat/2.0/en) - This API shows average weekly household spending.



Above is the architecture design for the web app. It will be developed using Visual Studio Code. There are two types of users; a guest and a registered user. A registered user can create and keep a budget plan. A guest can only view the dashboard.

The user will view the web User Interface and interact with it. The client-side UI will connect to the server side using http.

## **Section 4: The risks**

### **Data Freshness**

Risk: Data from external APIs may not be up to date. This could affect the accuracy and relevance of the information presented to users.

Mitigation: The data APIs supplied by the website listed above, are provided by the government of Ireland and the API endpoint may be outdated next year. To mitigate this, the data should be regularly monitored, and APIs should be updated where possible.

### **Learning Curve**

Risk: lack of experience can sometimes be detrimental to a project's timeline. Learning and implementing new technologies may consume more time than anticipated.

Mitigation: Allow time for learning and practicing. Seek for help from lecturers and online sources.

### **Data Privacy and Security**

Risk: Handling user data and financial information requires strict adherence to data privacy regulations and security measures.

Mitigation: Prioritize data encryption, compliance with data protection laws (e.g., GDPR), and regular security assessments to protect sensitive user information. Data is encrypted at rest in GCP, and the explanation can be found [here](#).

## Bibliography

- Quicken (2023) *10 budget categories that belong in your plan: Quicken, Quicken Blog*. Available at: <https://www.quicken.com/blog/budget-categories/> (Accessed: 24 October 2023).
- Rob Bertman, C. (2023) *Budget categories: Long list or short list?, Family Budget Expert - Rob Bertman*. Available at: <https://familybudgetexpert.com/budget-categories/> (Accessed: 24 October 2023).
- Person (2023) *How to make a budget spreadsheet, Capital One*. Available at: <https://www.capitalone.com/learn-grow/money-management/how-to-make-budget-spreadsheet/> (Accessed: 24 October 2023).
- lizgravier\_ (2023) *Here are the best free budgeting tools of 2023, CNBC*. Available at: [https://www.cnbc.com/select/best-free-budgeting-tools/#1\\_NeOPKxIH](https://www.cnbc.com/select/best-free-budgeting-tools/#1_NeOPKxIH) (Accessed: 24 October 2023).
- Household budget survey - CSO - central statistics office (2023) CSO. Available at: <https://www.cso.ie/en/statistics/housingandhouseholds/householdbudgetsurvey/> (Accessed: 24 October 2023).
- Ramsey Solutions (2023) *How to track your expenses, Ramsey Solutions*. Available at: <https://www.ramseysolutions.com/budgeting/how-to-track-expenses> (Accessed: 24 October 2023).
- Developers (no date) *Data.Gov.IE*. Available at: <https://data.gov.ie/pages/developers> (Accessed: 24 October 2023).
- Author, L.M.J. and Lasse Martin JakobsenGuest AuthorView ProfileLasse Martin JakobsenGuest AuthorView Profile (2018) *Learn how to develop to-do app - Golang and Angular - Pt. 1 golang API, Auth0*. Available at: <https://auth0.com/blog/developing-golang-and-angular-apps-part-1-backend-api/> (Accessed: 24 October 2023).
- Create a connection using the cloud SQL GO connector | cloud SQL for postgresql | google cloud (no date) Google. Available at: <https://cloud.google.com/sql/docs/postgres/samples/cloud-sql-postgres-databasesql-connect-connector> (Accessed: 24 October 2023).
- HS236 - average weekly household expenditure (no date) Dataset. Available at: [https://data.gov.ie/en\\_GB/dataset/hs236-average-weekly-household-expenditure](https://data.gov.ie/en_GB/dataset/hs236-average-weekly-household-expenditure) (Accessed: 24 October 2023).
- Go for web development (no date) Go. Available at: <https://go.dev/solutions/webdev> (Accessed: 25 October 2023).
- (No date) *Selenium*. Available at: <https://www.selenium.dev/> (Accessed: 25 October 2023).
- Use our budget planner: Moneyhelper (no date) MaPS. Available at: <https://www.moneyhelper.org.uk/en/everyday-money/budgeting/use-our-budget-planner> (Accessed: 25 October 2023).