Final Year Project Proposal

TU858/4

ZakatIQ - The Smart Islamic Finance Assistant

Wissam Hadjarab
C21404706

School of Computer Science
TU Dublin – City Campus

03/10/2025

Table of Contents

Contents

Table of Contents i

Declaration ii

Summary 1

Background (and References) 1

Proposed Approach 2

Deliverables 3

Project Schedule .... Error! Bookmark not defined.

Technical Requirements 4

Conclusion 5

References R5

Appendix A: First Project Review A1

Appendix B: Second Project Review B1

Appendix C: Prompts Used with ChatGPT C1

Appendix D: D1

Declaration

I hereby declare that the work described in this dissertation is, except where otherwise stated, entirely my own work and has not been submitted as an exercise for a degree at this or any other university.

Signed:

Wissam Hadjarab

03/10/2025

Summary

This project *ZakaatIQ*, is an AI-powered Islamic financial assistant designed to help Muslims manage their wealth in a halal (permissible by Islamic law) and ethical way. The app goes beyond traditional Zakat (Charity) calculators by using machine learning to predict Zakat eligibility based on a user's financial data (income, savings, debts, gold, etc.) and personal lifestyle factors like marital status, number of dependents, and employment.

The system will also use AI-based forecasting to estimate users' future income and savings, helping them plan ahead while avoiding interest-based (riba) savings. It will alert users if their financial profile conflicts with Islamic finance principles and offer halal suggestions (e.g., avoiding interest accounts, planning charitable giving).

Users will be able to securely track their net assets, view personalized dashboards, and access verified charity donation links to fulfill their Zakat directly through the app. All sensitive information will be encrypted, and security best practices (e.g., hashed passwords, encrypted storage) will be used throughout.

By combining AI, ethical finance, secure development, and Islamic values, ZakaatIQ offers a smart, modern solution for Muslims seeking financial clarity, spiritual accountability, and long-term planning — all in one user-friendly platform.

Background (and References)

Zakat is one of the key pillars for Islam and an obligation upon every Muslim who are eligible to pay under a certain threshold / criterion. All though its importance is very prominent in Islam, many people struggle to understand whether they qualify to pay Zakat, how much and how to calculate it correctly. There are basic calculators online which provide a one-time calculation with no data tracking, intelligence or security and are liable to give incorrect calculations, e.g. inputting uses value in gold, property etc that cannot be inputted into a calculator. They also don't take into consideration a person's lifestyle – whether they are married, have children, expenses etc.

There is currently no application or app / webapp that incorporates AI, Islamic financial rules, personal financial tracking and secure donation options in one platform. There are tools like Mint or YNAB that offer financial tracking but don't align with Islamic ethics as they are interest based. There are some apps like Zakatify (USA Platorm) that focus on donations but not eligibility prediction or financial guidance for Muslims.

Research in financial machine learning supports the use of algorithms like decision trees, linear regression and more for income and savings forecast. These models are already widely used in finance tech and can be adapted to suit different ethical applications. There are different frameworks such as Flask that help with security with encryption libraries to ensure data privacy which is crucial when dealing with sensitive financial information.

This project builds on existing technologies, combining and adapting them to tailor to the Muslim community, offering smarter, secure and a more personalized way to manage their finances to fulfil their religious obligations.

Proposed Approach

My plan to complete this project is to break it down into 5 key stages – Research, requirements gathering, analysis and design, implementation and testing/evaluation. I am planning so that each phase will bring me closer from an idea to a fully functioning web application.

1) **<u>Research:</u>**
   The first step into research for this project will be doing an in-depth analysis on Islamic finance to gain a clear understanding on the ruling for Zakat, interest etc in Islam. I'll study how eligibility is determined (Nisab, types of assets etc) and how different life circumstances can affect financial obligations. I am going to explore already existing Zakat calculators, Islamic tools and academic research on machine learning in finance, more specifically income predictions and savings forecasting.
   I will also be researching relevant security frameworks for handling sensitive user information to ensure privacy and GDPR compliance. I also plan to conduct research on the Muslim community – who this app is tailored for – and gather more intel as to what would be beneficial in the app and what their needs are.

2) **<u>Requirements Gathering:</u>**
   I will create basic user personas based on different user groups
   - Single working adult
   - Married user with children and savings
   - Self-employed business owner
   - Someone eligible to receive Zakat due to low income or debt

   I will identify what the user needs are for each specific user group and what they could need from the app – including features like financial input forms, Zakat predictions, savings forecast and secure donation. Based on this, I'll define core functional / non-functional requirements.

3) **<u>Analysis and Design:</u>**
   Once I have the requirements done, I plan to design the architecture of the system. I plan to have the project following modular design with components for:
   - User registration + authentication
   - Financial data input and lifestyle
   - Zakat eligibility prediction (ML)
   - Ai forecasting for savings/income
   - Secure charity donation links
   - Personal dashboard with graphs and insights

   I plan to design a simple, responsive UI using Flask for now with bootstrap. For backend, I am looking to use Python and PostgreSQL to store encrypted financial data and wireframes to create the map of user flow on how the app will run.

4) **Implementation:**
For the first steps of implementation, I will begin to set up the backend and database. I plan to create and execute a secure login user interface, using password hashing and session management. Then, I can build the financial tracking element and integrate it with Zakat calculator-based rules.
Next, I will be looking to train and implement a machine learning model (decision tress etc) to predict Zakat eligibility using data. I would also hope to implement an AI forecasting model by then (like linear regression) to estimate income/savings trends over time.
Lastly, I will add the user interface along with the dashboard and link the system to verified trusted charity payment pages to allow the user to securely pay the charity.

5) **Testing/Evaluation:**
I plan to test each component/ module individually, testing its full functionality and then the system. The ML models will also be evaluated using accuracy, precision, recall and F1-score (will research more about these methods so I can implement correctly). I will also be conducting several user testings on different user groups to get feedback on usability, accuracy and clarity on Zakat predictions.
Security will be tested to ensure user data is encrypted and not expose. I will be documenting all these results and evaluate how well the app meets original goals and what needs to be changed/fixed.

Deliverables

Project Deliverables:

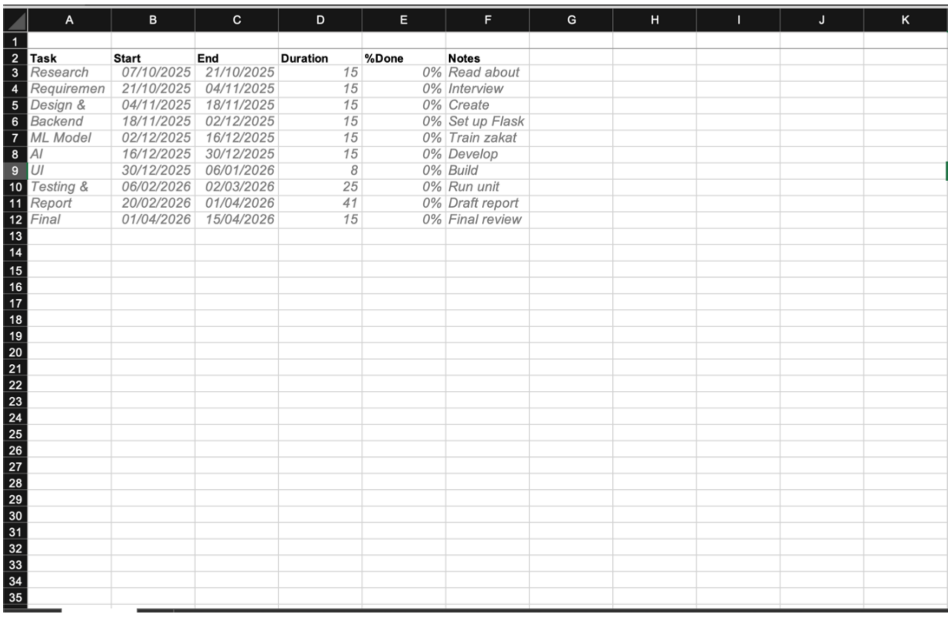
1) ZakaatIQ Web Application
A fully functional AI-powered web app that predicts Zakat eligibility, tracks the users' finances, forecasts their income/savings and integrate it with secure charity payment links.
It will include:
- User authentication (log in) and encrypted financial data storage
- ML based Zakat prediction model
- AI forecasting model for income/savings
- Lifestyle Zakat advice
- Secure dashboard with personalized insights and graphs for user
- Verified trusted charity donation links

2) Machine Learning Models
One classification model (decision tree or logistic regression) – for Zakat eligibility
One regression model (Linear regression or Prophet) – for income/savings forecasting
■ Research and learn both and ML models, do not know much yet

3) Database System
Encrypted database using mainly PostgreSQL and SQLite to store individual users' financial data securely

4) Technical Documentation
   - System architecture
   - Code documentation – comments etc
   - API Design – if applied
   - Security methods – encryption, authentication etc
5) Final report
   This will entail all the project’s goals, design, implementation, evaluation and reflection throughout the course of making the entire application
6) Final demo and presentation
   PowerPoint presentation showcasing the project along with a live or recorded demo of the working final product

- Gantt Chart:



Task	Start	End	Duration	%Done	Notes
Research	07/10/2025	21/10/2025	15	0%	Read about
Requiremen	21/10/2025	04/11/2025	15	0%	Interview
Design &	04/11/2025	18/11/2025	15	0%	Create
Backend	18/11/2025	02/12/2025	15	0%	Set up Flask
ML Model	02/12/2025	16/12/2025	15	0%	Train zakat
AI	16/12/2025	30/12/2025	15	0%	Develop
UI	30/12/2025	06/01/2026	8	0%	Build
Testing &	06/02/2026	02/03/2026	25	0%	Run unit
Report	20/02/2026	01/04/2026	41	0%	Draft report
Final	01/04/2026	15/04/2026	15	0%	Final review

Technical Requirements

To build and deliver a fully functioning web app, I will need the following technical components will be required in relation to hardware, software and infrastructure:

Hardware:

- MacBook Air M2 (macOS)
  This will be my main machine that I will use to develop the different features, suitable for coding, testing and running local servers.
  Has internet access for installing packages, deploying web app and using cloud services
- Second device
  I will use a secondary device like smartphone for testing responsive design and UI cross different screen sizes and OS

Software development tools:

- Python: This language is versatile which will be critical for my deployment of machine learning models, web development and scripting. Core language for backend logic and financial data processing

- Flask: Lightweight web frameworks for Python that gives full control over backend routes, session-based management (login/ register) and templates. Used to create the UI for login financial input, predictions, dashboards and charity links
- VSC: Lightweight code editor that supports all languages – including Python and Flak extensions. Main environment for writing and testing code
- Scikitlearn: Trusted, open-source Python library for machine learning. Supports classification (decision trees, logistic regression) and regression tasks which will be used for Zakat eligibility prediction and forecasting. Will be used to deploy ML models based on users' finances and lifestyle data – research more don't know about the tool
- Prophet / statsmodels (research both): Prophet – tool used for time series forecasting income and savings trends. Needed to predict how much users are likely to save or earn in the future based on patterns
- Pandas, Numpy + Matplotlib: Core python libraries for data analysis and visualization with basic graphs. Used for data input processing, ML engineering and visualizations in the dashboard.
- Cryptography (Fernet – research): Enables secure encryption and decryption of sensitive financial data. Ensures all user financial data and login credentials are stored securely in line with GDPR and ethical principles
- PostgreSQL – Scalable relational database to store encrypted user data, including income, savings and Zakat calculations

Conclusion

This project aims to develop a functioning and secure web application that helps Muslims manage their finances in a way that aligns with Islamic values. It incorporates machine learning to predict Zakat eligibility, AI- based incomes + savings forecasting and financial advice for fulfilling Zakat obligations.

Using Python and Flask and strong encryption along with data protection practices, the app offers a personalized dashboard that allows user to track their financial status over time. Its purpose is to help users fulfil their religious duty but to also promote financial awareness. By combining Islamic finance, AI cybersecurity in one application, the project aims to address a real-world gap in the market and offer a solution tailored to the Muslim community.

References

S. Aherne, *Using Machine Learning Techniques to Predict a Risk Score for New Members of a Chit Fund Group*, BSc Final Year Project, Technological University Dublin, 2020. [Online]. Available: https://arrow.tudublin.ie/scschcomdis/138/

N. Pereira, *Using Machine Learning Classification Methods to Detect the Presence of Heart Disease*, BSc Final Year Project, Technological University Dublin, 2021. [Online]. Available: https://arrow.tudublin.ie/scschcomdis/226/

F. Pedregosa et al., “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

Facebook Research, *Prophet: Forecasting at Scale*. 2017. [Online]. Available: https://facebook.github.io/prophet/

M. Grinberg, *Flask Web Development: Developing Web Applications with Python*, 2nd ed., O’Reilly Media, 2018.

Python Cryptography Authority, *cryptography: Python package for secure data*, 2024. [Online]. Available: https://cryptography.io/

Appendix A: First Project Review

Title: Using Machine Learning Classification Methods to Detect the Presence of Heart Disease

Student: Nestor Pereira

Description (brief): The project applies classification models to medical data to detect whether a patient has heart disease based on features like cholesterol, blood pressure etc. It compares different algorithms (decision trees etc) to identify the best predictor

What is complex in this project:

- Medical data often contain missing values, noise and varying scales – so preprocessing is complex
- Choosing and tuning multiple classification models
- Dealing with class imbalance (patients with disease vs healthy ones)
- Interpreting model outputs in a domain where false negatives/false positives have huge implications

What technical architecture was used:

- A data processing and ML pipeline (preprocess -> train/test -> validate)
- Comparison study of algorithms (decision tree, SVM)
- Likely Python / scikit-learn ecosystem

Explain key strengths and weaknesses of this project, as you see it.

Strengths:

- Direct comparison of algorithms gives insight into which works best for the problem
- Uses realistic medical, enhancing relevance
- Clear domain and well-understood evaluation metrics

Weakness:

- Not a full software solution – no UI or web application features
- Doesn't consider privacy, encryption or domain-specific rules (health care data regulation)
- May not include lifestyle inputs (which your project plans to)
- Its insights are limited to medical predictions, not financial behaviour or ethics

Appendix B: Second Project Review

Title: Using Machine Learning Techniques to Predict a Risk Score of New Members of a Chit Fund Group

Student: Sinead Aherne

Description (brief): This project is a supervised learning to predict a risk score for a new or existing member of a chit fund (collective savings/credit group). It aims to identify which members are likely to default, based on historical data

What is complex in this project:

- Handles imbalanced classification
- Uses multiple machine learning algorithms (SVM, decision trees)
- Feature engineering and factor reduction
- Incorporating social network analysis (links between members) as part of the risk score model

What technical architecture was used:

- Data driven ML pipeline
- Comparison of multiple
- Use of real historical data from a chit fund
- Python based ML stack

Explain key strengths and weaknesses of this project, as you see it.
Key strengths:

- Testing multiple models to find the best fit
- Real world data usage gives practical relevance
- Including social network / relational analysis
- Clear objective and domain

Weakness:

- Doesn't include a user facing application or interface
- No features related to data security, privacy, encryption or user profiling
- Doesn't account for domain-specific constraints
- The evaluation metrics may be impacted by data imbalance

Appendix C: Prompts Used with ChatGPT

- Would this project be okay to implement for TUD final year project along with Machine Learning and Security modules?

- Would I have to access bank accounts and extra security for that?

- How does this differ from just a regular zakat calculator?

- I need to add complexity my supervisor said it lacks complexity. I want to make it a financial app that's tailored towards Muslims and has features that would incorporate Islamic finance. I was thinking to use AI to predict how much they make and how much savings the user has so that they won't be paying interest with their money and a feature that gives secure authorized links to multiple charities that can pay their Zakat to, how will I go by implementing this?

- Could I also add a feature that will tailor to the user's lifestyle e.g. if they're single, married have kids etc?

- Can you explain further the software development tools and why I am choosing the options you have given me?

Appendix D: