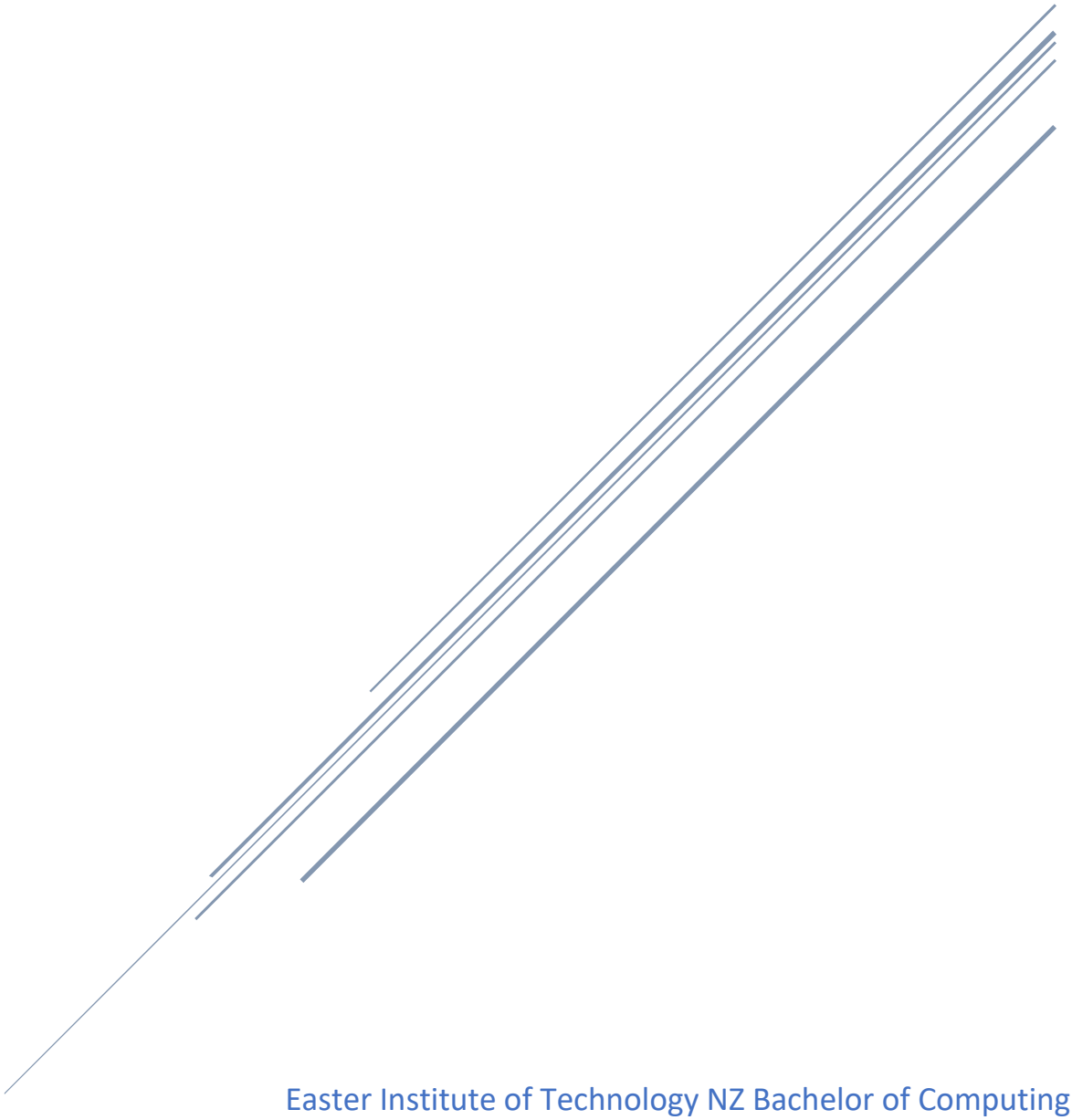
USER REQUIRED SPECIFICATION

Mortgage Calculator



Easter Institute of Technology NZ Bachelor of Computing Systems
ITPR7.508 Business Application Programming
John Jamieson

Katherine Mulder & Alex Borawski

Document Approvals

We are pleased to present this document for the approval of the mortgage calculator software project. The client has agreed to the scope of work for customizing and implementing the mortgage calculator software.

Sign-Off: Project Client

Approver Name	Title	Email	Signature	Date
John Jamieson	Software Client	jjamieson@eit.ac.nz		

Contents

Section 1. Introduction

1.1 Purpose

This user requirement specification aims to deeply analyze and think over the business issues and program specifications important for our project's development. Through this document, we want to explain the decisions we've made with our client and deal with any changes needed for the result.

Also, this document will keep things clear and accountable by explaining why we made each decision and if we changed anything during development.

1.2 Background information

The initial project started when the client provided a detailed draft Excel spreadsheet containing the desired features and calculations for the mortgage calculator. These requirements serve as the foundation for the development of the web application.

1.3 Subject Scope

The scope of this project is to provide a web application for managing mortgages. This includes an initial set-up allowing users to input essential details for creating new mortgages with flexibility for various types of loans. After the first mortgage has been established, users can update existing mortgage details to accommodate changes in financial circumstances. The system generates transaction reports, aiding users in tracking payment history and financial obligations, while also enabling mortgage editing to adjust as needed, with proper validation to maintain data integrity. We have also provided simple authentication features for users to create an account with a username and password.

1.4 Project Excludes

Our project won’t include complex mortgage variations like adjustable-rate mortgages (ARMs) and advanced financial analyses like investment assessment or risk evaluation. The project will cover the development of the website and its functionalities, deployment onto live servers is not within its scope. Basic considerations for website visibility will be addressed, but extensive SEO services such as keyword research and link building are not included in the project. Post-project maintenance and updates, including ongoing software support, are not covered under this project's scope. Users are encouraged to seek legal and tax advice from professionals for matters related to mortgages, as such advice is not included in the project scope. Integration with external

systems beyond basic mortgage functionalities, such as banking or real estate databases, is not part of the project's focus.

Section 2. Analysis/ Reflection of Business Issues

In our discussions with the client, we gathered information regarding their expectations and structured how we were going to deliver the project within a timeline. We will combine elements of both Agile and Waterfall methodologies to deliver this project, leveraging its iterative and collaborative approach. This will help us adjust to changing needs by keeping communication open with our team and the client. With Agile, we can quickly react to shifts in priorities or new insights during development, making sure the result meets what the client wants. This flexibility also lets us take input from stakeholders along the way, leading to a better final product.

We will hold regular meetings with the client to keep them updated on our progress. In the event of any scope creep changes, we will mitigate them by carefully evaluating change requests and implementing a change management process to assess the impact of any proposed changes on our project scope, schedule, and budget. For example, if the client requests additional features to be included in the software, we will assess the feasibility and impact of these changes on the project. This includes identifying and discussing potential risks associated with the changes and brainstorming mitigation strategies with the client to ensure successful implementation.

If there are significant setbacks, including delays, which puts its successful completion at risk. We will analyze to identify the underlying issues contributing to the project's failure, discuss pinpoint areas for improvement and communicate with the client about project challenges and recovery plans. We will actively seek feedback from the client and each other to refine project management practices. The advantage of a small team size, with only two of us, is that we will leverage this for quickly implementing changes and iterating on our approach, driving ongoing success in future projects.

Section 3. Program Specifications

We are committed to developing a mortgage calculator using a programming language, based on the provided Excel spreadsheet. This tool aims to enable users to input mortgage details, track transactions, and visualize data to understand mortgage maturity. Users can manage multiple mortgages, monitor progress, and assess the impact of extra payments on loan balance and savings. The goal is to provide a simple calculation mortgage management platform for informed decision-making. Editing features ensure data accuracy, while visual representations like interactive charts

facilitate an understanding of payment trends and loan schedules. Personalization options, including payment reminders and communication preferences, will enhance the user experience. Detailed technical information will be outlined in the software design documentation, covering system architecture and implementation.

For the functional requirements, we will deliver an interface comprising login, account creation, user settings, main analysis page, mortgage/transaction creation, and data removal interfaces, with navigation between pages and clear presentation of information. Design straightforward entry pages for mortgages and transactions, including a clear description next to entry fields, and implement error handling for incorrect entries. In the analytical functionality, we will include estimated repayment calculation, payment breakdown, and mortgage maturity information. The system should analyze extra payments and reduced terms, with the ability to generate graphs and amortization tables based on mortgage state. Also, provides users with the ability to remove mortgages and transactions.

For the non-functional requirements, we will implement error checking to mitigate rounding errors and input out-of-range issues to prevent unauthorized access to other user's data, which is the most likely vulnerability. We will implement measures to thwart such unauthorized access. Another vulnerability is the potential for malicious data input. To counter this, extensive data validations, including type, length, and value checks, will be integrated throughout the application. Sensitive data will be encrypted to enhance security and remain inaccessible even to administrators.

While our software isn't aimed at mass scalability, it's important to build flexibility into the code for future feature additions. We'll adopt a modular code design with clear class definitions, facilitating efficient adjustments, and prioritize compatibility with broader frameworks to ensure integration into larger environments for future scalability. Also, the application's speed is important, especially when users want to see their analysis right after they log in. The biggest challenge we face is making the analysis load quickly. One way to speed things up is to use something called multi-threading, but for now, we'll focus on loading all the analyses as soon as a user logs in. This should make everything load much faster. After we finish developing the application, we'll ensure it's easy to maintain in the long run by organizing the code into modular sections, clearly defining each part's role. Users will receive detailed feedback after incorrect data entry, along with clear descriptions to clarify expectations in input and display areas. We'll develop the software to accommodate changes in usage patterns. By separating database access and web display functionalities, we'll facilitate effortless substitution or enhancement of usage methods.

Section 4. Conclusion

Our examination of the business issues underscores the importance of using project management methodologies to meet client expectations and complete projects successfully within the agreed timeline. By blending aspects of both Agile and Waterfall methodologies, we seek to benefit from Agile's collaborative approach while retaining Waterfall's structured planning, thus ensuring adaptability to changing requirements.

We are committed to developing a mortgage calculator that not only meets the functional and non-functional requirements outlined but also prioritizes user experience and data security. Through clear communication channels with the client and regular updates on project progress, we will mitigate scope creep and address any setbacks promptly, ensuring the project's successful completion.

Moving forward, our small team size will be an advantage, allowing us to implement changes quickly and iterate on our approach for ongoing success in future projects. With a focus on modular code design, efficient adjustments, and compatibility with broader frameworks, we aim to build flexibility and scalability into the application while maintaining fast performance and ease of maintenance.

Our commitment to good project management and technical excellence will make this project a success and lay a strong foundation for future work.