Case study on Well Home Loans

Introduction

Well Home Loans is an online non-bank home loan provider which creates simple financial products to help everyday Australian's meet their lending goals. User can come and explore various loan schemes, apply online or reach our customer service for more details. Admin has the role of configuring products, recommend product to customers, verify documents and reach out to the customers when needed.

In this document, we will cover various modules of Well Home Loans, functional flow of loan interview process, in-depth description of technical architecture and different methodologies followed during the development process.

Modules

Well Home Loans has three modules as: Marketing, Loan Application and Staff Portal having their respective websites.

1. Marketing

Customers can go through various products in the marketing website of Well Home Loans. The site also provides many essential features like loan comparer, home loan calculator, purchase calculator, refinance calculator etc. which help them to take a step closer towards purchasing or refinancing a property. Free property reports and blogs ameliorate customers to stay up-to-dated with financial market and ever-changing market trends.

Fig 1: Marketing site features

Technology Used: Following technologies are used in Marketing website:

- MVC
- AngularJS 1.6
- Entity Framework

2. Loan Application

Loan application portal handles end to end loan application scenarios for customers. This module is intelligently designed to collect their necessary information in a series of steps by asking customers the right set of questions. Customers can select a product of their choice, view the loan amount they are eligible for and submit their application for further processing.

Customer information includes personal data, property they are looking out for, assets, liabilities, current and previous employment, income, expenses, credit history etc. Based on these details, the business logic computes and displays the suitable products matching their profiles.

Technology Used: Following technologies are used in Loan Application website:

- WebAPI
- LESS
- AngularJS 1.6
- Entity Framework

3. Staff Portal

Staff portal is designed to be used by the admin for monitoring the progress of an applications and managing various component that forms the core of loan processing module. These components include brand, product with its attributes, interest rates, policies, conditions, default fees and cashbook.

Admin checks the failed policy rules for the loan applications and provides valuable suggestions to the customers making it easier for them to get a loan.

Technology Used: Following technologies are used in Loan Application website:

- WebAPI
- LESS
- AngularJS 1.6
- Entity Framework

Functional Flow

Customers can start a loan application by accepting certain consents and providing their email-id. A confirmation mail is sent to their email-id along with a link to set their password.

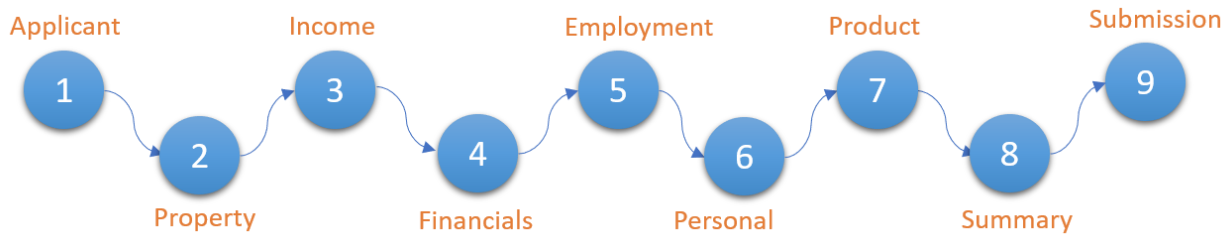

Fig 2: Functional flow of loan Application module

Once the account is successfully created, they are required to provide personal and financial details in the following order:

- Borrower Information:
  - Name
  - Marital status
  - Dependents
  - Credit history health
- Property Information:
  - Property type: purchase or refinance
  - Property address
  - Deposit amount
- Income Information:
  - Current and Previous Financial Income
- Financial Information:
  - Assets
  - Liabilities
  - Expenses
- Employment Information:
  - Current and previous employment details
- Personal Information:
  - Date of birth
  - Residential status
  - Phone numbers
  - Current and previous address
- Product Information:
  - Product selection
  - Customizing the product

On completion of filling all the required data, they can view the summary of the information provided by them. After reviewing, they can submit the loan application for further processing.

Technical Architecture

Loan Application

The front end of Loan application uses the following technologies:

- HTML5 and bootstrap
- LESS for styling
- Grunt for java script task runner
- Bower for package manager
- AngularJS for front end binding and making http call to Web API.

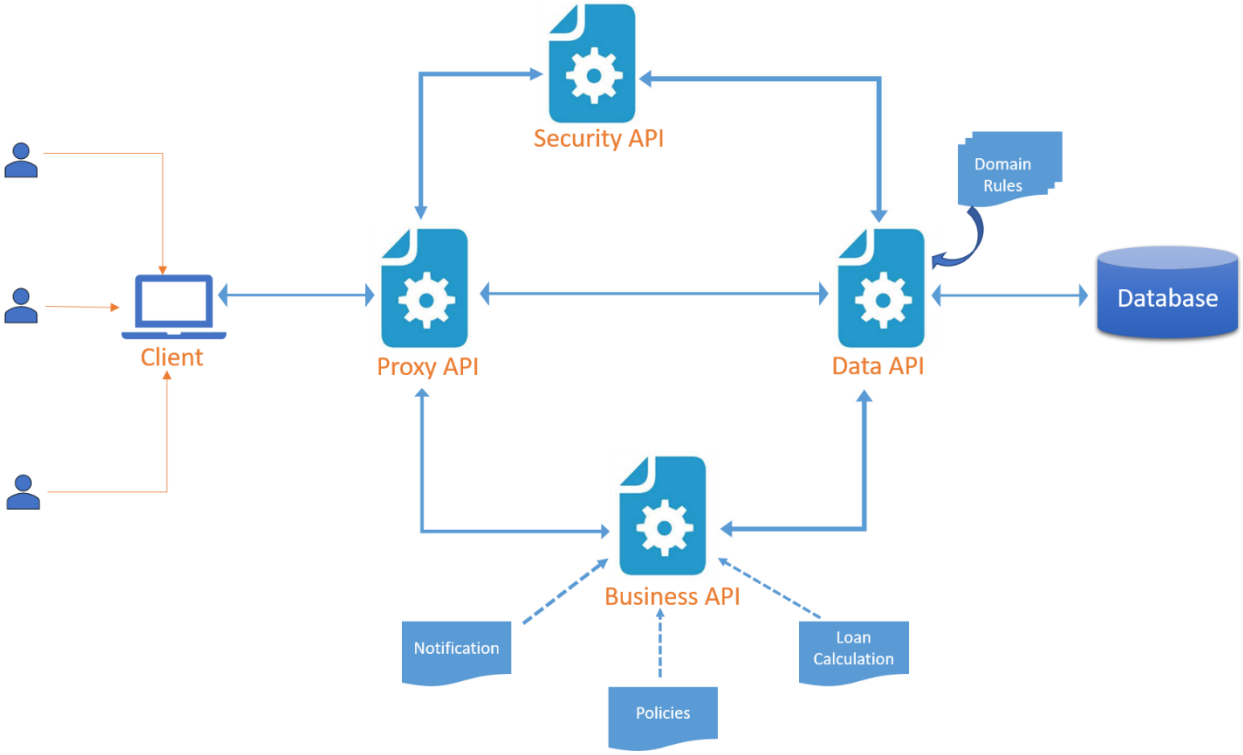

Fig 3. Architecture diagram

On receiving a request from the client, AngularJS makes a call to proxy-WebAPI. Each request has a token associated with it. This token is used to authenticate the user. This authentication process is taken care by Security-WebAPI component. Once authenticated, proxy-WebAPI then sends another request to data-WebAPI which gets relevant data from the database and sends back the response in json format. On receiving the response from data-WebAPI, proxy-WebAPI creates another response with relevant data required to render the UI and sends it back to the client as json payload.

Development Methodology

We adhere to the following practices throughout our product development cycle.

Code Review

- Peer review of the code
- Followed checklists for code review

Release Process

- Followed continuous deployment and integration process
- Monitored successful completion of integration tests for each deployment
- Manual deployment to all environment

Project Management

- TFS for source code management
- VSTS for project management
- Agile sprints scrum model
- Yammer for team communication and Q&A

Testing

- Manual testing of WebAPIs using Postman
- Automation testing using Selenium

Team Information

Team Members @costar

- Satya Sai
- K Rajesh Kumar
- Sarvani Yakali
- Farheena