Implementation

# Types

The main types created for this project were the Account and the Contact type.

The Account is the main object that holds data for the user’s account. This object is written to a firebase database as a JSON, and read back into the website to display information that the user can modify.

The Contact type is the base object used to construct a new contact. Contacts are stored as an array within the Account object. These hold data such as name, address and contact information.

# Methods

Many global and local-use methods were created to streamline the development of the project

* AddAccountToDatabase()
  + Takes an account object as a parameter, and adds it to the Firebase database.
* UpdateAccountOnDatabase()
  + The most useful method we created. Takes an account object as a parameter, and pushes all of the current updates to this object, to the corresponding entry in the database. Allows us to constantly update the account when any changes are made, so they can be reflected instantly on the webpage.
* OnLoginSuccessful()
  + Almost a callback method, which calls when the Account is logged in on the current webpage, which allows us to fill out the HTML data using the account object.

# Core Functionality

This web application is a simple, yet efficient contacts book. The user interface is designed to streamline the user’s experience, and minimise any confusion.

The application is capable of allowing the user to

* Register an Account
* Log in to the Account
* Add multiple contacts, with varied information
* Edit any contact
* Delete a contact
* Call a contact
* SMS a contact
* Email a contact
* Add multiply tags to a contact
* Create new tags
* Delete existing tags
* Delete all contacts in one click
* Find, and display any duplicate accounts

The application uses Firebase Database to store all account data for the user to allow access from any location.

# Testing

Testing was conducted by following a path similar to the Agile Methodology. This allowed us to consistently check the state of the application, and make any necessary changes. On top of this, it prevented any last minute situations that can arise from other testing approaches which require you to test all at once, at the end of the project.

# Team Contributions

## B00330925’s contributions

* Settings Page (HTML & Javascript)
* Firebase Functionality

## B00330156’s contributions

* Tags Page (HTML & Javascript)
* Firebase Functionality

## B00334101’s contributions

* Validation Page (HTML & Javascript)
* Settings Page (HTML & Javascript)
* Duplicates Page (HTML & Javascript)
* Firebase Functionality

## B00330023’s contributions

* Validation Page (Javascript)
* Duplicates Page (Javascript)
* Accounts Page (HTML & Javascript)
* Tags Page (HTML & Javascript)
* Firebase Functionality

## B00311466 ‘s contributions

* None