Joshua Monaghan

www.joshmonaghan.com

Software Engineering

PWA Scope

Table of Contents

[**Client and Problem:** 3](#_Toc193791891)

[**Why is the project a PWA:** 3](#_Toc193791892)

[Cross-Platform Compatibility 3](#_Toc193791893)

[No App Store Dependency 3](#_Toc193791894)

[Automatic Updates 3](#_Toc193791895)

[Security 4](#_Toc193791896)

[**Social, Legal and Ethical Implications:** 4](#_Toc193791897)

[Social Implications 4](#_Toc193791898)

[Customer Privacy 4](#_Toc193791899)

[Trust and Transparency 4](#_Toc193791900)

[Loss of Human Interaction 4](#_Toc193791901)

[Legal Implications 4](#_Toc193791902)

[Data Protection Laws 4](#_Toc193791903)

[Customer Consent & Data Usage 4](#_Toc193791904)

[Intellectual Property Concerns 5](#_Toc193791905)

[Ethical Implications 5](#_Toc193791906)

[Data Ownership 5](#_Toc193791907)

[Marketing Practices 5](#_Toc193791908)

[Customer Interactions 5](#_Toc193791909)

[**Project Requirements**: 6](#_Toc193791910)

[Client Interactions 6](#_Toc193791911)

[First Interaction with Client 6](#_Toc193791912)

[First Requirements Interaction 6](#_Toc193791913)

[Second Requirements Interaction 7](#_Toc193791914)

[Application Requirements 7](#_Toc193791915)

[**Security:** 9](#_Toc193791916)

[Common Web Vulnerabilities 9](#_Toc193791917)

[Security Principles 9](#_Toc193791918)

[PWA Security Implemented 9](#_Toc193791919)

[**Storyboards:** 9](#_Toc193791920)

[**Data Dictionaries:** 9](#_Toc193791921)

[**Gantt Chart:** 9](#_Toc193791922)

[**Research and Planning:** 9](#_Toc193791923)

[Web Frameworks 9](#_Toc193791924)

[Tools 9](#_Toc193791925)

[Development Approach 9](#_Toc193791926)

[Logbook 10](#_Toc193791927)

# **Client and Problem:**

**Client:** Trevor Monaghan, middle aged man who has expertise’s in designing and building Progressive Web App and accounting. He understands what a PWA is and what is needed to design and build one which will make the requirements straightforward and detailed.

He is the owner of **Climax Valuations and Forensics**, a business located in Belmont NSW, specialising in business valuation and forensic accounting reports, predominantly for court matters such as divorce, shareholder disputes and economic damages. He has no accessibility requirements, but they will be included because the project is SaaS allowing multiple users to use it.

Trevor is seeking a PWA solution to manage all his client’s details, files, valuation reports and communication. His current Client Relationship Manager (CRM) is increasing their pricing which is why he needs a free solution to ensure his overheads are kept to a minimum. This will help him have easier access to his client's needs and details while also making him more productive and efficient.

# **Social, Legal and Ethical Implications:**

## Social Implications

### Customer Privacy

Customer privacy is an essential requirement for all projects, ensuring that all user’s information and data is secure and only accessible to authorised individuals. Australia’s Privacy Act 1988 defines how user’s information is handled and stored, ensuring that users can feel safe and secure on the web. In the CRM project, customer privacy is the biggest concern because of

### Trust and Transparency

Trust and transparency are very important in web development, being transparent about how and why their data is stored and handles allows.

### Loss of Human Interaction

Asdasd

## Legal Implications

### Data Protection Laws

asdasd

### Customer Consent & Data Usage

asdsad

### Intellectual Property Concerns

Asdasd

## Ethical Implications

### Data Ownership

### Marketing Practices

### Customer Interactions

# **Why is the project a PWA:**

A PWA (Progressive Web App) is an efficient and easy using way of sourcing software on the internet. The following reasons why I’ve selected a PWA as the most suitable choice for the project:

## Cross-Platform Compatibility

PWAs are web-based applications that combine websites and apps together on the internet, because they are on the web this allows anyone to access the application given that they have a web browser. This enables faster development and removes the need for multiple different versions of the application to be made to suit the devices requirements. This is crucial for the project; it removes the need for multiple versions of the application to be made which saves time.

## No App Store Dependency

Since PWAs are on the web this removes the need for the application to be downloaded and installed on the device. This streamlines user to access the application allowing for quick access given the user has an internet connection. In relation to the CRM, this allows users quick and easy access to the project without the need for downloads or updates.

## Automatic Updates

Unlike software installed on a device, PWAs can push updates to the application without the user even realising, allowing bug fixes or new features to be added to the PWA without user’s releasing. This accelerates the user’s productivity and accessibility to the application while ensuring new features are automatically updated when a user revisits the site.

## Security

PWAs use security by design to ensure that user’s information is secure for unauthorised personals. Since the application is on the web, HTTPS encrypts all data that’s received and sent ensure man-in-the-middle attacks are avoided. Everything in the database is also stored on a secure server, not on the user’s device, which increases the security for all users. The PWA will be constructed with all common web security protocols, ensuring the project is secure for the users.

|  |  |
| --- | --- |
| Benefits | Disadvantages |
| Cross-Platform Compatibility | Limited Device Features |
| No App Store Dependency | iOS Limitations |
| Automatic Updates | Performance Constraints |
| Security | Limited Multi-Tasking |

# **Project Requirements**:

The project is a CRM (Customer Relationship Manager) which allows users to track their quotes, leads, invoices and customer contact information. Trevor, who is the client, currently uses a CRM but needs another one because of the following reasons:

**Pricing**Trevor uses an application called Xero; with this they use to provide a free CRM that he has been using for years. Currently Xero have changed that and are charging a fee to use that feature which is why he needs this project.

**Features**Xero’s CRM doesn’t have all the features that would boost workflow, it does provide basic features but there’s no reporting. The PWA which I’m developing will have multiple features and will include reporting to boost production and experience.

## Client Interactions

### First Interaction with Client

The initial interaction with the client was on the 15/03, the phone call talked about expectations and requirements the project would achieve. After the brief call, I understood what the project would look like and what it should accomplish. A follow up email from the client should be sent in the upcoming days with a draft requirement attached.

### First Requirements Interaction

A screenshot of a computer

AI-generated content may be incorrect.

On the 18/03, the first requirements were received from the client with brief expectations and optional requirements that could be added to the project. After analysing the requirements requested, most requirements could be met in the timeframe assigned but some optional requirements won’t be feasible, like:

* **Stripe Integration**
* **Xero API**

Every other requirement should be met and a follow up email revolving around the updated requirements will be sent in the upcoming days.

[*Initial Requirements for Software by Climax Valuations and Forensics Pty Ltd*](file:///C:\Users\joshm\Documents\SoftwareEngineering\Y12\Assessments\Major-Project\SoftwareEngineering-Major-Project\PWA%20Scope%20and%20Requirements\Initial%20Request%20for%20Software%20by%20Climax%20Valuations%20and%20Forensics%20Pty%20Ltd.pdf)

### Second Requirements Interaction

After reviewing the requirements requested, I will be using all requirements except: Stripe Integration and Xero API. The new requirements were sent through email to the client, this will allow him to review my requirements and give feedback.

## Application Requirements

The application has specific requirements that need to be met for the project to be considered ready for production. Here are the following requirements that need to be met:

**Companies**

* Ability to store company details, including business name, address, industry, size, and contact details.
* Each company must have at least one primary contact.
* Categorisation of companies as prospect or customer.
* Ability to assign internal users to manage the relationship of companies.

**Contacts**

* Ability to store individual contact details, including name, job title, email, phone number, and associated companies.
* A contact can be associated with one or more companies or exist independently without a link to any companies.
* Ability to mark a contact as a primary contact for a specific company.

**Leads**

* Leads are created when there may be a current or future opportunity to provide a quote to a company.
* Each lead must be associated with a company and may also be linked to a specific contact.
* Status tracking for leads (e.g., In Progress, Won, Lost).
* Ability to assign internal users to manage leads.

**Quotes**

* Quotes are generated for leads when a pricing estimate is provided.
* Standard fields for quotes, including quote number, date, associated company, contact, and lead etc.
* Ability to add line items to quotes, including description, quantity, price per unit, and tax code. Software needs to calculate and display the total and any tax.
* Ability to amend and resend quotes.
* Status tracking for quotes (e.g., Draft, Sent, Accepted, Rejected).

**Invoices**

* An invoice is created when a quote is won
* The invoice will initially contain the same line items as the corresponding quote but must be editable so that it can be different to the quote.
* Standard fields for invoices, including invoice number, issue date, due date, and payment terms.
* Ability to add and edit line items like quotes.
* Status tracking for invoices (e.g., Draft, Sent, Partly Paid, Fully Paid).

**Payments**

* Ability to record one or more payments against an invoice.
* Calculation of outstanding balance on each invoice.
* Automatically change the status to paid when payments are equal to the invoice total.

**Reporting**

* Users must be able to generate reports for Companies, Contacts, Leads, Quotes, and Invoices.
* Reports should be filterable by date, status, company, or other relevant fields.
* Ability to export reports to formats such as CSV or PDF.

**User Management & Security**

* Each business subscriber will have its own internal users.
* Role-based access control to restrict functionalities based on user roles.
* 2-factor authentication for all users.
* Secure login system with password recovery options.

**Additional Considerations**

* The CRM will be web-based and must support access from different devices.
* The user interface should be intuitive and easy to navigate.
* The system should be scalable to handle multiple businesses efficiently if we decide to make it a SaaS product.
* Data security and encryption should be implemented for sensitive information.

**Boundaries**

This project is to be built over a 10-week span, so time management is crucial so not all requirements asked by the client can be met. Boundaries are implemented to stop scope creep and to ensure projects stay on task with the assigned timeframe. Nothing outside of the scope will be integrated unless required for the project.

# **Security:**

## Common Web Vulnerabilities

## Security Principles

## PWA Security Implemented

# **Storyboards:**

# **Data Dictionaries:**

## SQL Tables

### Users

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Variable | Data Type | Format for Display | Size in Bytes | Description | Example | Validation |
| UserID | Char (32) | Alphanumeric and all lower case. | 32 Bytes | A UUID value that is used as a primary key for every user. | a81bc81bdead4e5dabff90865d1e13b1 | Ensure that the UUID length and characters match the criteria. |
| FName | Varchar (50) | Plain Text | Up to 50 Bytes | Stores the first name of the user. | Josh | Validates that the user’s input is only letters and is the correct length. |
| LName | Varchar (50) | Plain Text | Up to 50 Bytes | Stores the last name of the user. | Monaghan | Validates that the user’s input is only letters and is the correct length. |
| DOB | Date | DD MM YYYY | 3 Bytes | Stores the user’s date of birth. | 11-11-2008 | Validates that the year is appropriate, and the date and month are valid. |
| Email | Varchar (80) | All lowercase | Up to 80 Bytes | Stores the users email address. | joshmono2008@outlook.com | Validates the email address is valid. |
| PhoneNum | Varchar (15) | +CC-XXX-XXX-XXXX | Up to 15 Bytes | Stores the users Phone Number. | +61 490 767 436 | Validates that the phone number has a valid country code and local number. |
| Password | Varchar (250) | Alphanumeric with special characters allowed | Up to 250 Bytes | Stores a encrypted version of the users password. | $argon2i$v=19$m=16,t=2,p=1$WXZwbGpjUEdLUEZBeFlEUw$U/E9CHQXKaTReVVeZRvrmQ | Validates that the encrypted password is valid. |
| SecretKey | Varchar (16) | Alphanumeric and all upper case. | 16 Bytes | Stores a secret key used for Two Factor authentication. | QAAL5GI2H3XIRFHB | Validates that the secrecy key is 16 characters long and is valid. |
| FirmID | Char (32) | Alphanumeric and all lower case. | 32 Bytes | Many to many link to the users firms. | a81bc81bdead4e5dabff90865d1e13b1 | Validates that the link to the firms exists. |
| Admin | Bit | Yes/No | 1 Byte | Determines if the user is an Admin | False | Validate that the user is allowed to be an admin. |

### Firm

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Variable | Data Type | Format for Display | Size in Bytes | Description | Example | Validation |
| FirmID | Char (32) | Alphanumeric and all lower case. | 32 Bytes | A UUID value that is used as a primary key for every firm. | a81bc81bdead4e5dabff90865d1e13b1 | Ensure that the UUID length and characters match the criteria. |
| Name | Varchar (50) | Plain Text | Up to 50 Bytes | Stores the firm’s name. | Apple | Validates that the user’s input is only letters and is the correct length. |
| Email | Varchar (80) | All lowercase | Up to 80 Bytes | Stores the firms email address. | business@business.com | Validates the email address is valid. |
| PhoneNum | Varchar (15) | +CC-XXX-XXX-XXXX | Up to 15 Bytes | Stores the users Phone Number. | +61 490 767 436 | Validates that the phone number has a valid country code and local number. |

### Company

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Variable | Data Type | Format for Display | Size in Bytes | Description | Example | Validation |
| CompanyID | Char (32) | Alphanumeric and all lower case. | 32 Bytes | A UUID value that is used as a primary key for every user. | a81bc81bdead4e5dabff90865d1e13b1 | Ensure that the UUID length and characters match the criteria. |
| Name | Varchar (50) | Plain Text | Up to 50 Bytes | Stores the firm’s name. | Apple | Validates that the user’s input is only letters and is the correct length. |
| Contact | Char (32) | Alphanumeric and all lower case. | 32 Bytes | A UUID value that is used as a primary key for every Contact. This field is a many to many so a company can have many contacts. | a81bc81bdead4e5dabff90865d1e13b1 | Ensure that the UUID length and characters match the criteria. |

### Quote

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Variable | Data Type | Format for Display | Size in Bytes | Description | Example | Validation |
| QuoteID | Char (32) | Alphanumeric and all lower case. | 32 Bytes | A UUID value that is used as a primary key for every quote. | a81bc81bdead4e5dabff90865d1e13b1 | Ensure that the UUID length and characters match the criteria. |
| Pricing | Money | Currency | 8 Bytes | Stores multiple prices for the quote. | $12500 | Validates that the input is a valid currency. |
| Contact | Char (32) | Alphanumeric and all lower case. | 32 Bytes | A UUID value that is used as a primary key for every Contact. This field is a many to many so a quote can have many contacts. | a81bc81bdead4e5dabff90865d1e13b1 | Ensure that the UUID length and characters match the criteria. |
| Notes | Varchar (300) | Plain Text | 300 Bytes | Store’s notes about the quote | Ring company. Email Joe. | Validates that the text is a valid format. |
| Item | Char (32) | Alphanumeric and all lower case. | 32 Bytes | A UUID value that is used as a primary key for multiple items related to the quote. One to Many. | a81bc81bdead4e5dabff90865d1e13b1 | Ensure that the UUID length and characters match the criteria. |
| LeadID | Char (32) | Alphanumeric and all lower case. | 32 Bytes | A UUID value that is used as a primary key linking to the lead. Optional Field. | a81bc81bdead4e5dabff90865d1e13b1 | Ensure that the UUID length and characters match the criteria. |

### Invoice

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Variable | Data Type | Format for Display | Size in Bytes | Description | Example | Validation |
| InvoiceID | Char (32) | Alphanumeric and all lower case. | 32 Bytes | A UUID value that is used as a primary key for every invoice. | a81bc81bdead4e5dabff90865d1e13b1 | Ensure that the UUID length and characters match the criteria. |
| Pricing | Money | Currency | 8 Bytes | Stores multiple prices for the invoice. | $12500 | Validates that the input is a valid currency. |
| Contact | Char (32) | Alphanumeric and all lower case. | 32 Bytes | A UUID value that is used as a primary key for every contact. This field is a many to many so a invoice can have many contacts. | a81bc81bdead4e5dabff90865d1e13b1 | Ensure that the UUID length and characters match the criteria. |
| Notes | Varchar (300) | Plain Text | 300 Bytes | Store’s notes about the invoice. | Ring company. Email Joe. | Validates that the text is a valid format. |
| Item | Char (32) | Alphanumeric and all lower case. | 32 Bytes | A UUID value that is used as a primary key for multiple items related to the invoice. One to Many. | a81bc81bdead4e5dabff90865d1e13b1 | Ensure that the UUID length and characters match the criteria. |
| QuoteID | Char (32) | Alphanumeric and all lower case. | 32 Bytes | A UUID value that is used as a primary key linking to the quote. Optional Field. | a81bc81bdead4e5dabff90865d1e13b1 | Ensure that the UUID length and characters match the criteria. |

### Lead

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Variable | Data Type | Format for Display | Size in Bytes | Description | Example | Validation |
| LeadID | Char (32) | Alphanumeric and all lower case. | 32 Bytes | A UUID value that is used as a primary key for every lead. | a81bc81bdead4e5dabff90865d1e13b1 | Ensure that the UUID length and characters match the criteria. |
| Pricing | Money | Currency | 8 Bytes | Stores multiple prices for the lead. | $12500 | Validates that the input is a valid currency. |
| Contact | Char (32) | Alphanumeric and all lower case. | 32 Bytes | A UUID value that is used as a primary key for every contact. This field is a many to many so a lead can have many contacts. | a81bc81bdead4e5dabff90865d1e13b1 | Ensure that the UUID length and characters match the criteria. |
| Notes | Varchar (300) | Plain Text | 300 Bytes | Store’s notes about the lead. | Ring company. Email Joe. | Validates that the text is a valid format. |

### Contact

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Variable | Data Type | Format for Display | Size in Bytes | Description | Example | Validation |
| ContactID | Char (32) | Alphanumeric and all lower case. | 32 Bytes | A UUID value that is used as a primary key for every contact. | a81bc81bdead4e5dabff90865d1e13b1 | Ensure that the UUID length and characters match the criteria. |
| FName | Varchar (50) | Plain Text | Up to 50 Bytes | Stores the first name of the user. | Josh | Validates that the user’s input is only letters and is the correct length. |
| LName | Varchar (50) | Plain Text | Up to 50 Bytes | Stores the last name of the user. | Monaghan | Validates that the user’s input is only letters and is the correct length. |
| Email | Varchar (80) | All lowercase | Up to 80 Bytes | Stores the firms email address. | business@business.com | Validates the email address is valid. |
| PhoneNum | Varchar (15) | +CC-XXX-XXX-XXXX | Up to 15 Bytes | Stores the contact’s Phone Number. | +61 490 767 436 | Validates that the phone number has a valid country code and local number. |

## Python Data

### asdkajsn

# **[Gantt Chart](https://trello.com/invite/b/67e3ca4a811242d1c1808f4f/ATTIc56c8ed342cda1f8e72b03d6d1890e5cDB635A55/pwa-project):**

A screenshot of a project

AI-generated content may be incorrect.

# **Research and Planning:**

## Web Frameworks

## Tools

## Development Approach

# Logbook

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Task Achieved | Difficulties and Solutions | Ideas and Thoughts | Reflection |
| 15/03 | Identify what my project would be and who my client is. | Trying to find a project that would cover all areas in web development. | Unsure if the project is too large to finish in the time frame provided | I have a client and project which is a good starting point. |
| 16/03 | Create GitHub Repository. | None | None | None |
| 16/03 | Wrote about the client and the project. | Only problem was I had not received any requirements. | I knew that I needed requirements before I started the other parts. | Good start but need requirements asap. |
| 18/03 | Received Requirements | None | Now that I had the requirements, I was able to complete the rest of the report. | Need to start the rest of the report now. |
| 19/03 | Application Requirements and Boundaries |  |  |  |
| 25/03 | Client Interactions |  |  |  |
| 26/03 | Start on Data Dictionary |  |  |  |
| 26/03 | Gantt Chart | Difficulties logging in and using |  |  |
| 27/03 | Finish SQL Data Dictionary |  |  |  |
| 28/03 | Edit Why the project is a PWA |  |  |  |