



CRM TECHNOLOGIES

SOUTH COAST PRIVATE CRM TECHNICAL REPORT

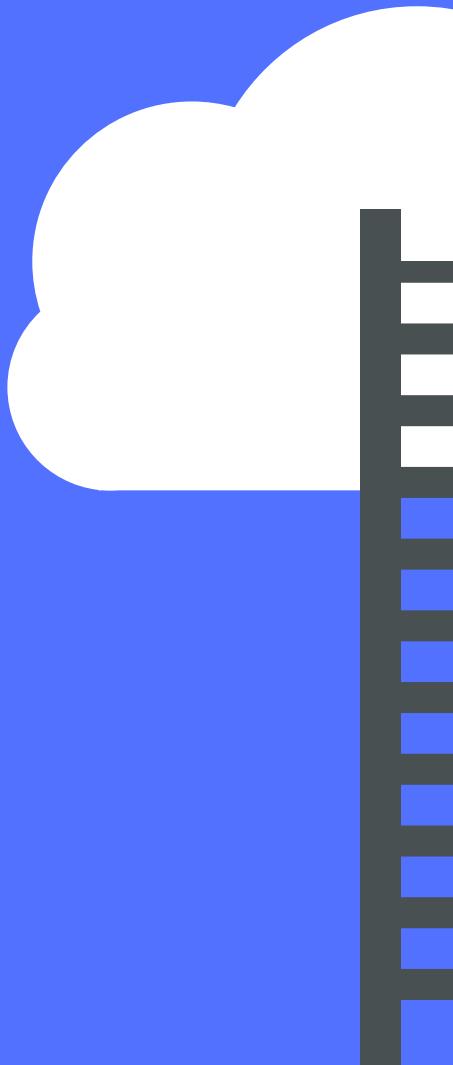
Prepared by:
CRM TECHNOLOGIES Project Team

May 2021

CONTENTS

Table of Contents

Project Overview	1
Project Team	2
Scope Management	3
Iteration Management	5
System Design	6
Requirements Traceability	10
Requirements Summary	14
User access	15
Cross-system interaction	16
Development Tools	21
Information Architecture	24
Post project review	29
Process improvement	30
Project Acceptance Criteria	31
Transition Plan	32
Lessons Learned	33
Future Product Considerations	34
System Installation	35
Data dictionary	36
Style Guide	38



PROJECT OVERVIEW

South Coast Private Hospital

South Coast Private Hospital (SCPH) CRM is a web application to manage SCPH's medical providers as part of the business development and marketing operations of the organisation. The system has been designed to capture key provider details including contact information, provider interactions and manage communication distribution lists. These functionalities have been complemented by a dashboard which displays key CRM metrics and provider information to the user.

Problem Statement

Digital transform work processes associated with Marketing and Business Development activities, specifically the management of prospective and current provider information.

South Coast Private Hospital previously managed its provider information in manual form via a number of Excel Spreadsheets. This resulted in hours of lost productivity due to poor data maintenance and provider information being scattered over various versions of the spreadsheets.

Previous processes saw Marketing and Business Development staff extract provider information from internal systems, add the information to the current version of the spreadsheet and subsequently use this information to guide business development activities. The information in the spreadsheet offers little insight into the history of provider engagement activities adding to the productivity issues experienced by the Marketing and Business Development team.

PROJECT TEAM

Riley Ross

Project Leader and Business Analyst Requirements management, project coordination, cloud architecture design and deployment. Work closely with all project members.

Harrison Norris

User interface design and related frontend development activities. Work closely with Backend Developers to ensure the frontend is fully functional with the backend

Sarah Swyer

Database Developer Database design and related database development activities. Work closely with Backend Developers to ensure the database is fully functional with the backend.

Keenan Weekes

Backend design and related backend development activities, including the development of a RESTful API. Work closely with the Frontend Developers and Database Developer to ensure the backend is fully functional with both the frontend and database.

Albert Iannelli

Backend design and related backend development activities, including the development of a RESTful API. Work closely with the Frontend Developers and Database Developer to ensure the backend is fully functional with both the frontend and database.

Bradley Rusak

Security and Business Analyst Security analysis and related development activities including, login/user authentication, user roles management and multi-factor authentication

Dean Torijovski

Business Analyst Documentation support and functional user requirements test management support

SCOPE MANAGEMENT

Project Name	Current and Prospective Customer Relationship Management (CRM) Database Tool	
Summary Statement	<p>The scope of this project is to facilitate the delivery of a usable CRM database management tool to South Coast Private Hospital. The tool will replace the existing business processes for Doctor, iCare and ADF Case manager provider information management. The project aims to reduce manual data error issues and administrative time spent by the Business Development and Marketing unit on managing current and prospective provider information. The project aims to be completed within the allocated time frame of the CSIT321 subject.</p>	
Objective	<ul style="list-style-type: none">Minimise the manual process handling and governing management of South Coast Private Hospital provider information on excel by 95% (Exclusive of reporting and analytics)Increase business process management efficiency of provider data by 75%Integrate product into a cloud based maintainable environmentCompletion of product by May 2021	
Sponsor	Julia Kolevski Business Development & Marketing Manager SCP	
Project Manager	Riley Ross	
Start/End Dates	Start Date : 3rd August 2020	End Date : 24th May 2021
Budget	AWS compute cost, AWS Route 53 domain register and Google Developer API	
In Scope	<p>In Scope</p> <ul style="list-style-type: none">Minimise the manual process handling and governing management of South Coast Private Hospital provider information on excel by 95% (Exclusive of reporting and analytics)Increase business process management efficiency of provider data by 75%Integrate product into a cloud based maintainable environmentCompletion of product by May 2021	

SCOPE MANAGEMENT

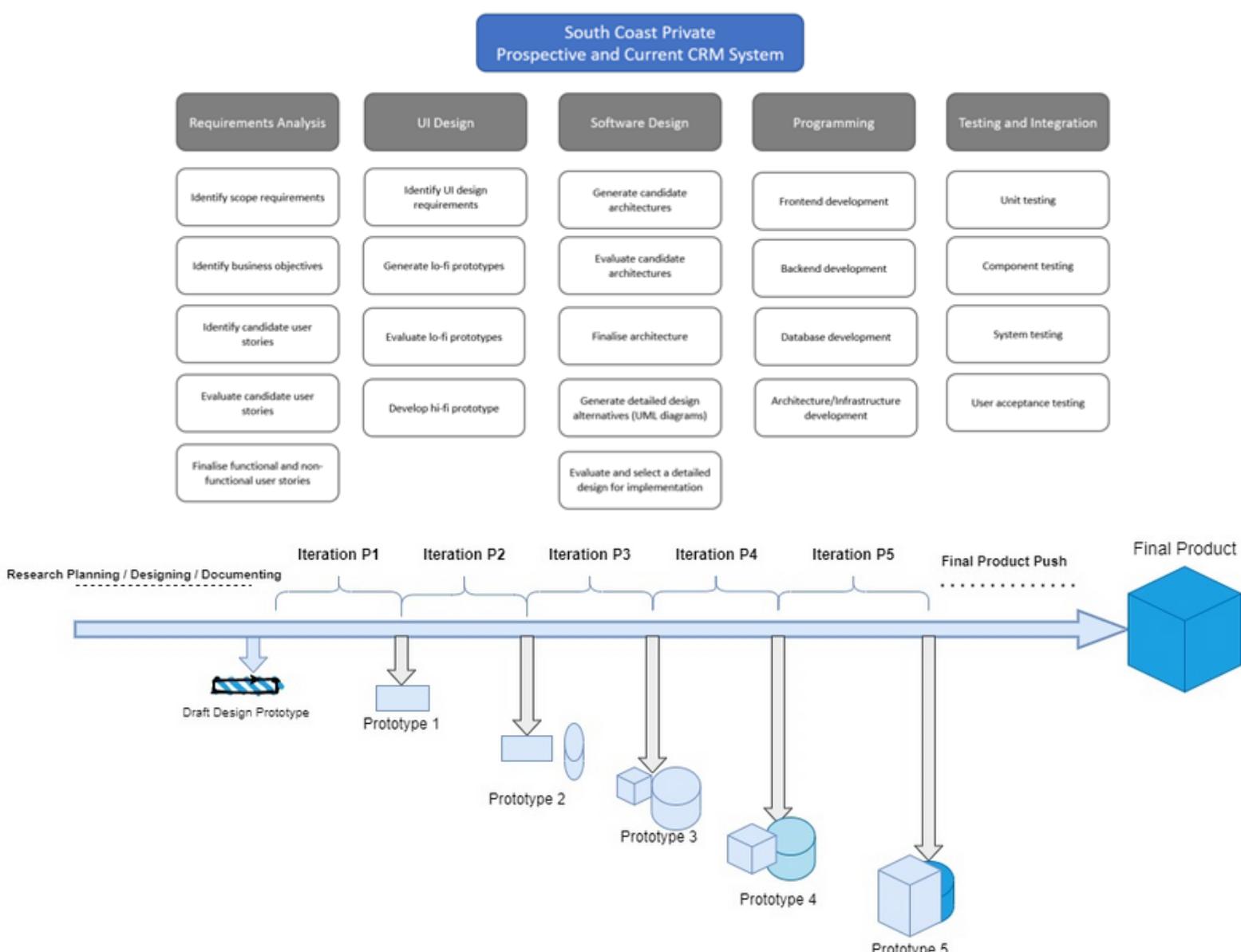
Out of Scope	<p>Out of Scope</p> <ul style="list-style-type: none">• Integration requirements of the current system• Integration development within an existing CRM software product• Management of sensitive medical information• Deployable mobile application
Stakeholders	<ul style="list-style-type: none">• Supervisor: Dr Mark Freeman• Sponsor: Julia Kolevski South Coast Private• South Coast Private CEO: Kim Capp• Healthecare CIO: Aaran Stent• South Coast Private Business Development and Marketing unit Assistants/Interns• Project Team Student Group
Constraints	<p>Business Constraints</p> <ul style="list-style-type: none">• Funding: Product funding is limited to open source capabilities with the exception of domain cost and google development api• 10 Month Project Timeline• Cost of ownership: Maintenance of deployed product constrained to restricted knowledge in troubleshooting domain <p>Technical Constraints</p> <ul style="list-style-type: none">• Technical Constraints• Web Based Solution• Maintainability: Product must have minimal maintenance• Software suites constrained to free licensing and community version• Cloud data center solution contained within Australia only• Locked internal systems within South Coast Private Policy
Technology	<p>Technology Used:</p> <ul style="list-style-type: none">• Amazon Web Services• Google Developer API• React• Material-UI• Trello• Slack• Discord• Outlook

ITERATION MANAGEMENT

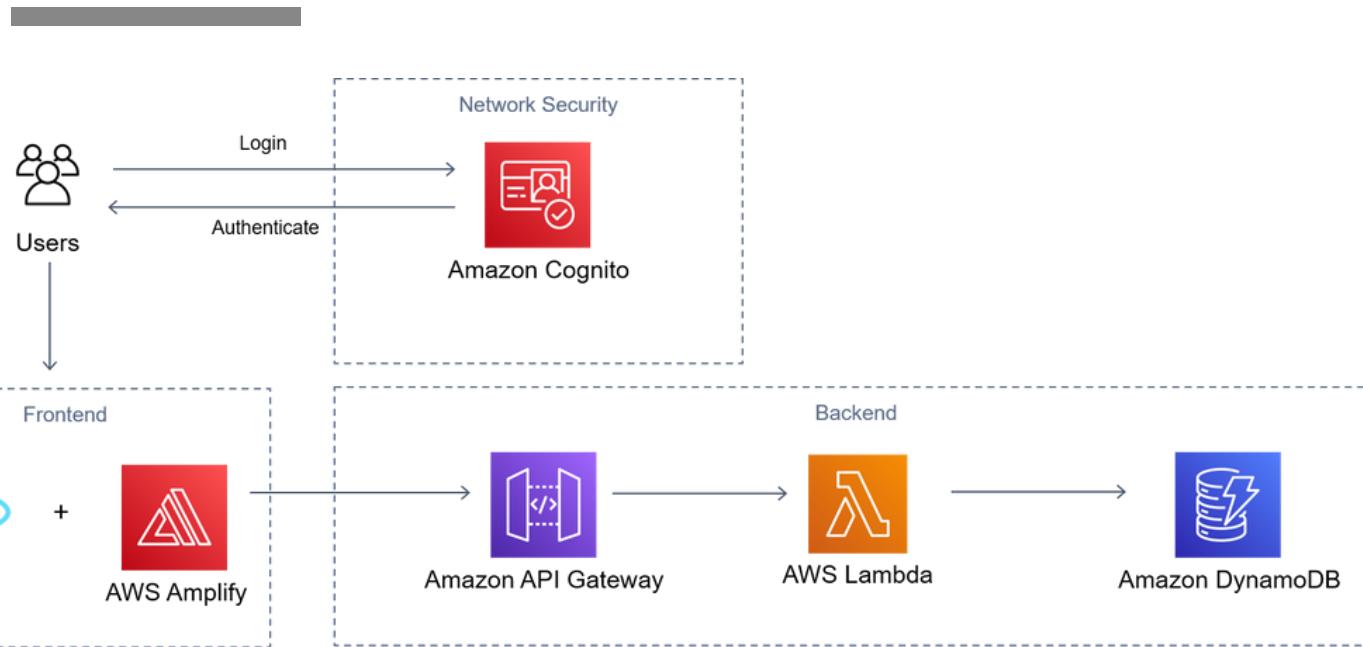
The project was developed in three core work packages that included the iterative development of the CRM Database Page, Dashboard Page and Settings functionalities. Each iteration built on the next and were used as key checkpoints to review the current design and functionality with the Client.

After each checkpoint the Client's feedback was gathered and incorporated into the upcoming development work, until a final product was delivered.

The project team utilised an agile approach to iteration management due to the flexibility it provided to pull outstanding development tasks into a 'Doing' state when resources became available to complete them.



SYSTEM DESIGN



Pipeline

The pipeline begins with the user interacting with the React framework which was used to create the fully functional and interactive user interface and user experience. When the user logs in Amazon Cognito is used to provide a secure authentication of the user's login. Amazon Cognito also provides the tools necessary to manage all users, create them, give them permissions and manage their securely stored passwords.

AWS Amplify is an AWS platform that consists of a number of different pre-configured AWS micro-services allowing for the efficient development of web applications. We have primarily used it to build and host the react frontend.

The Amazon API Gateway is the intermediary between the frontend services and the backend services. The API Gateway intercepts all incoming requests, ensures they are in correct format and have appropriate authorization and sends them to the AWS Lambda functions to process the event. It also receives feedback from the AWS Lambda and sends the requested object's state back to the frontend of the application for display within the UI.

The AWS Lambda serverless architecture is responsible for performing all the functionality of the DynamoDB database. There are many Lambda functions including full CRUD functionality. It communicates to the database by performing the operations stated in the functions and also sends database data back through the pipeline to AWS API Gateway to be sent further through to the frontend. AWS DynamoDB is the NoSQL database holding all of the CRM's database tables and is integral to search and database functionality provided to the user in the user interface. It receives operations from the AWS Lambda and sends the requested data back to it.

PROVIDER TABLE

API Endpoint: GET /provider-table

Lambda: Read_All_Provider_Table

Inputs: None.

Operations: Retrieves an array of all the provider records stored in the database.

Returns: Status code 200 returned if successful and the response body contains the array of provider records.

API Endpoint: POST /provider-table/provider

Lambda: Create_New_Provider

Inputs: Takes "Group" input for the Cognito user pool group and "Provider" object for adding provider record to the database, the object contains attributes: Title, First Name, Last Name, Postcode, Provider Number, Street, Suburb, State, Group, Practise, Position, Active, Email, Fax, and Phone.

Operations: Generates a unique ID attribute and assigns it to the new user. Using the current date and time, assign the provider record an attribute "CreationDate".

Sends a google maps API request to get the latitude and longitude attributes to assign to a provider record. If the user belongs to any group, increment the groups table counter. Add an empty record in the group table with "ID" attribute set as the new user ID. Add user record to the database.

Returns: Status code (200 if successful or 403 if user does not have sufficient permissions). Response body contains the new user ID, creation date of the user, latitude, and longitude.

API Endpoint: PATCH /provider-table/provider

Lambda: Update_Provider

Inputs: Takes "Group" input for the Cognito user pool group, "Parameter" which contains an array of the provider attributes to be updated, "Value" which contains the new value that the parameters will be changed to and an "ID" attribute which provider record to update.

Operations: Updates the provider record specified by the "ID" attribute using the data from "Parameter" and "Value". If provider attributes street, suburb, state, or postcode are updated then the latitude and longitude attributes are updated as well. Records the update in the History table.

Returns: Status code (200 if successful or 403 if user does not have sufficient permissions). Response body contains the newly updated provider record.

API Endpoint: DELETE /provider-table/{id}

Lambda: Delete_Provider

Inputs: Takes "Group" input for the Cognito user pool group. "ID" attribute is passed in the endpoint path.

Operations: Deletes the provider record from the provider database table, using the "ID" attribute. Deletes the provider from the history table. Decrements the group counts in the groups table if the provider was in any groups.

Returns: Status code (200 if successful or 403 if user does not have sufficient permissions). Response body contains a success or failure message.

HISTORY TABLE

API Endpoint: GET /history-table/{user-id}	Lambda: Read_History_Record
---	------------------------------------

Inputs: The "ID" attribute is passed in the endpoint path.

Operations: Retrieves the history record of a user in the history table.

Returns: Status code 200 returned if successful and the response body contains the user's history record.

API Endpoint: PATCH /history-table/{user-id}	Lambda: Update_History_Record
---	--------------------------------------

Inputs: Takes "Group" input for the Cognito user pool group. "ID" attribute is passed in the endpoint path.

Operations: Updates the history record specified by the "ID" in the database.

Returns: Status code (200 if successful or 403 if user does not have sufficient permissions). Response body contains a success or failure message.

GROUP TABLE

API Endpoint: GET /groups-table	Lambda: Read_All_Group_Table
--	-------------------------------------

Inputs: None.

Operations: Retrieves an array of all the group records stored in the database.

Returns: Status code 200 returned if successful and the response body contains the array of group records.

API Endpoint: POST /groups-table/{name}	Lambda: Create_New_Group
--	---------------------------------

Inputs: Takes "Group" input for the Cognito user pool group. "groupDetails" specifies the name and number of the group record to be created.

Operations: Creates a group record in the group database table with the "groupDetails" input.

Returns: Status code (200 if successful or 403 if user does not have sufficient permissions). Response body contains a success or failure message.

API Endpoint: DELETE /groups-table/{name}	Lambda: Delete_Group
--	-----------------------------

Inputs: Takes "Group" input for the Cognito user pool group. "name" attribute is passed in the endpoint path, this specifies the name of the group that will be deleted

Operations: Deletes the group record from the groups table in the database. Goes through each record in the provider table and removes the group from each provider record.

Returns: Status code (200 if successful or 403 if user does not have sufficient permissions).

COGNITO USER POOL TABLE

API Endpoint: GET /cognito-user-pool	Lambda: Cognito_Get_Users
--------------------------------------	---------------------------

Inputs: None

Operations: Retrieves an array of all the cognito users stored in the cognito user pool along with each users' permissions and a Boolean if MFA has been set up or not.

Returns: Status code (200 if successful or 403 if user does not have sufficient permissions). Response body contains the array of cognito user pool records.

API Endpoint: PATCH /cognito-user-pool/{username}	Lambda: Cognito_Update_Group
---	------------------------------

Inputs: Takes "Group" input for the Cognito user pool group. "NewCognitoGroup" specifies the name of the group that the user will be added to. "username" specifies the cognito user that will be updated.

Operations: First removes the user from any previous groups they are a part of. Adds the user to the cognito group specified in the input.

Returns: Status code (200 if successful or 403 if user does not have sufficient permissions). Response body contains a success or failure message.

API Endpoint: DELETE /cognito-user-pool/{username}	Lambda: Cognito_Delete_User
--	-----------------------------

Inputs: Takes "Group" input for the Cognito user pool group. "username" specifies the cognito user that will be deleted.

Operations: Deletes the specified user from the user pool.

Returns: Status code (200 if successful or 403 if user does not have sufficient permissions). Response body contains a success or failure message.

API Endpoint: DELETE /cognito-user-pool/{username}/mfa	Lambda: deactivate-mfa
--	------------------------

Inputs: Takes "Group" input for the Cognito user pool group. "username" specifies the cognito user will have their MFA removed.

Operations: Deactivates the specified users MFA.

Returns: Status code (200 if successful or 403 if user does not have sufficient permissions). Response body contains a success or failure message.

REQUIREMENTS TRACEABILITY

FUNCTIONAL REQUIREMENTS

Requirement	Status	Test ID/s
1 The system shall store provider information including the type of provider, their name, position, organisation, contact details, address information and provider numbers (where available).	<input checked="" type="checkbox"/>	1.1, 1.2, 1.3
2 The system shall be capable of displaying a list of provider information to the user.	<input checked="" type="checkbox"/>	2.1, 2.2, 2.3
3 The system shall allow the user to search provider information by provider name and provider organisation.	<input checked="" type="checkbox"/>	3.1, 3.2, 3.3
4 The system shall allow the user to filter provider information by provider type, inpatient/outpatient provider, current/prospective provider and active/inactive provider.	<input checked="" type="checkbox"/>	4.1, 4.2, 4.3, 4.4, 4.5, 4.6
5 The system shall allow the user to add a single new provider to the system via manual input.	<input checked="" type="checkbox"/>	5.1, 5.2, 5.3
6 The system shall allow the user to upload an excel spreadsheet containing provider information in order to add a group of new providers to the system.	<input checked="" type="checkbox"/>	6.1, 6.2, 6.3
7 The system shall be capable of detecting and restricting the input of duplicate records.	<input checked="" type="checkbox"/>	7.1, 7.2, 7.3
8 The system shall be capable of exporting all provider information to an excel spreadsheet for the user to download.	<input checked="" type="checkbox"/>	8.1, 8.2, 8.3
9 The system shall allow the user to edit all provider information, including toggling provider status i.e. current or prospective / active or inactive. Note: providers cannot be deleted from the system, only marked as inactive.	<input checked="" type="checkbox"/>	9.1, 9.2, 9.3

FUNCTIONAL REQUIREMENTS

Requirement	Status	Test ID/s
10 The system shall be capable of tracking information related to provider visits, including the date and time of the visit, attendees, location and meeting notes.	<input checked="" type="checkbox"/>	10.1, 10.2, 10.3
11 The system shall be capable of tracking individual and group correspondence to providers, including type of correspondence (invitation, event notification, email, fax, group, meeting request), recipient, date and description.	<input checked="" type="checkbox"/>	11.1, 11.2, 11.3
12 The system shall be capable of tracking the provision of marketing collateral to providers, including type of collateral, recipient and date.	<input checked="" type="checkbox"/>	12.1, 12.2, 12.3
13 The system shall allow the user to create distribution groups to assist with the manual distribution of correspondence and its subsequent tracking in the system.	<input checked="" type="checkbox"/>	13.1, 13.2, 13.3
14 The system shall allow the user to view a profile for each provider that displays the history of all correspondence, visits, provision of marketing materials and distribution groups the provider belongs to.	<input checked="" type="checkbox"/>	14.1, 14.2, 14.3
15 The system shall be capable of reporting the number of times a provider has referred patients to the hospital.	<input checked="" type="checkbox"/>	15.1, 15.2, 15.3
16 The system shall be capable of visually displaying a map to the user that contains a marker for each provider in the system.	<input checked="" type="checkbox"/>	16.1, 16.2, 16.3
17 The system shall display the percentage breakdown of providers in NSW vs ACT, as well as key regions within these locations.	<input checked="" type="checkbox"/>	17.1, 17.2, 17.3
18 The system shall allow the user with administrator privileges to set-up and delete system users.	<input checked="" type="checkbox"/>	18.1, 18.2, 18.3, 18.4, 18.5, 18.6
19 The system shall allow the user with administrator privileges to reset passwords on behalf of other users.	<input type="checkbox"/>	Revised Scope

REQUIREMENTS TRACEABILITY

NON-FUNCTIONAL REQUIREMENTS

Requirement	Status	Test ID/s
1 The system shall accommodate the current space requirements needed to maintain CRM data and have the capability to expand automatically as CRM data grows.		This has been configured using AWS DynamoDB elastic storage allocated to store records.
2 The system shall have a user-friendly interface design that decreases the learning curve for a new user and replaces the need for a user manual.		2.1
3 The system shall be backed-up automatically to prevent the loss of data. Back-ups should take place no less than once per week.		Back up configuration with AWS DynamoDB.
4 The system shall implement a responsive design so the system can be viewed and used from a desktop computer or mobile device.		4.1, 4.2
5 The system shall be protected through the implementation of Multi-Factor Authentication and best-practice username and password protections to secure confidential data stored within the system.		5.1, 5.2, 5.3, 5.4, 5.5
6 The system shall be hosted external to SCPH's current IT infrastructure to enable access to the system when working inside and outside of SCPH offices.		6.1
7 The system shall have the ability for data to be uploaded to the system in a format that closely aligns with output from other internal systems including CyberQuery and BIJIB. Note: full integration with these systems is not required.		7.1, 7.2
8 The system shall be implemented in such a way to minimise maintenance of the code base or infrastructure post implementation.		8.1
9 The system shall implement confidential data standards as set out in Prudential Standard CPS 234 -Information Security		The team has, to the best of our knowledge and ability, complied with this standard

NON-FUNCTIONAL REQUIREMENTS

Requirement	Status	Test ID/s
10 The system should have an appropriate user authentication process.		10.1, 10.2, 10.3, 10.4, 10.5
11 The system should have appropriate access controls in place to determine what a specific user can do.		11.1, 11.2
12 The system shall be available to its users during normal working hours (Monday to Friday, 8.30am to 5.30pm). Downtime within normal working hours should not exceed 10 seconds in any one day.		12.1
13 The system shall allow a user to search and retrieve a single CRM record e.g. Doctor profile. The response time of this transaction should not exceed 1 second.		13.1
14 The system shall allow a user to conduct an advanced search e.g. all Doctors that have been visited in the last 12 months. The response time of this transaction should not exceed 2 seconds.		14.1

REQUIREMENTS SUMMARY

As highlighted by the requirements traceability matrix, all of the functional and non-functional requirements were successfully implemented and achieved by the team. This was done by configuring various services (most prominently, Amazon Web Services) and programming. Additionally, quality assurance has occurred through continuous testing.

Changes to the requirements and their implementation were minimal. For efficiency and simplicity, it was decided that all users can reset their own passwords, instead of relying on a user with administrative privileges to reset their passwords. Additionally, record and group counts were added to the dashboard to provide users with a statistical summary of the contents of the database. Furthermore, after the process of identifying requirements, the client decided that functionality to delete records was required, so this was implemented. For ease of searching and to provide users with more search options, functionality was added to allow users to search for providers based on any or all of their attributes. Functionality to add and delete filter groups was also added to account for any future group changes. Finally, to provide a form of organising records, functionality to sort records by their attributes was added. Overall, changes to the requirements and their implementation were minimal, but carried out when necessary to improve the system.

Overall, through our work over the last year, the team has been able to deliver a CRM system that, in its current state, has various useful features. User authentication mechanisms have been implemented to provide security. A dashboard providing users with statistical and factual information about the contents of the database has been included. Functionality allowing users to, among other things, add and delete records, search for records and edit records has been provided. Finally, various settings are available to the user (for example, reset password). In conclusion, through our work over the last year, the team has been able to implement and deliver an effective CRM system.

USER ACCESS

Three users were identified as key users of the CRM each requiring a set of individual role-based permissions. Each user is required to log into the CRM and verify their identity utilising multi-factor authentication. Within the application user actions are restricted based on the role the user holds. User access permission are outlined within the table below.

	admin	read/write	read
View Providers	✓	✓	✓
Search Providers	✓	✓	✓
Filter Providers	✓	✓	✓
Add Providers	✓	✓	
Add Filter Group	✓	-	-
Delete Filter Group	✓	-	-
Export Records	✓	✓	✓
Email Providers	✓	✓	✓
Fax Providers	✓	✓	✓
Edit Provider	✓	✓	-
Mark Provider Active/Inactive	✓	-	-
View Record History	✓	✓	✓
Edit Other History	✓	✓	-
View Dashboard	✓	✓	✓
Add New User	✓	-	-
Change Own Password	✓	✓	✓
Change Default Database Display (inactive providers)	✓	✓	✓
Change a User's permission group	✓	-	-
Delete a User	✓	-	-
Reset a User's MFA	✓	-	-

User identities and role permissions are managed via the AWS Cognito micro-service. The CRM's Settings page provides functionalities for the Admin account user to manage Cognito based interactions via the user interface.

CROSS-SYSTEM INTERACTION

How the system interacts with other existing systems

Microsoft Excel and Text File programs

The problem area surrounding our project involves our client who was working with a slow and disjointed Microsoft Excel process wanted a new streamlined CRM system to replace her old workflow. Due to this our program has high interoperability with Microsoft Excel and Text Files in CSV format. The CRM allows the client to import a Microsoft CSV file for the allowance of mass creation or modification of this data. The CRM also allows the mass extraction of provider records by allowing the exporting of these records as a CSV file which can later be opened in Microsoft Excel/Text files. Through this workflow pipeline our system interacts and provides compatibility with Microsoft Excel and other Text File programs.



Fax and Email programs

Our system also interacts with email and fax systems by allowing the exportation of mass email address and fax addresses to be copied to clipboard by the client and used to send emails and faxes in any email or fax program of their choice. This system interaction is important to South Coast Private Hospital due to their need to target and message many different providers in a single distribution. These messages could involve medical discussion, educational event invitations, information for certain distribution groups or for providers in certain areas.

Due to the closed system architecture of South Coast Private our interaction with their systems is restricted in many areas due to the confidential data and corporate hierarchy that is possessed by South Coast Private and their parent company Healthcare.



CROSS-SYSTEM INTERACTION

Libraries and APIs

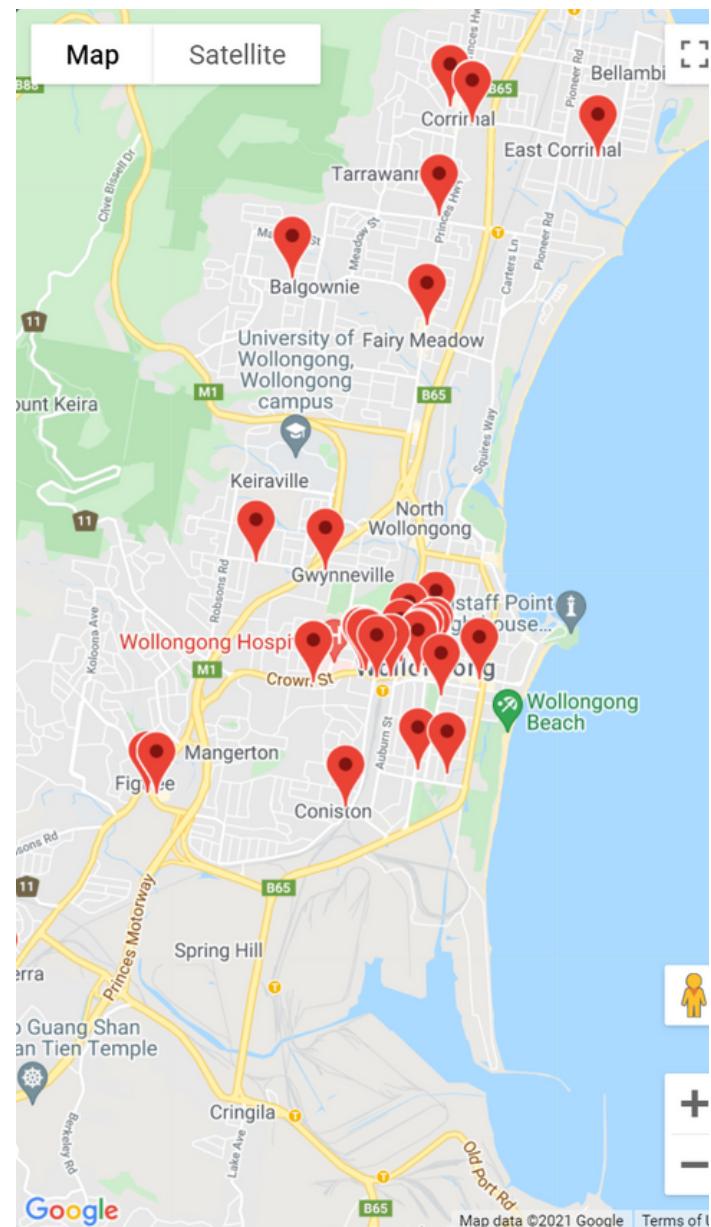
Google Maps

An important use case to the client involved a marker-based overlay of where South Coast Private's providers reside on a map. This allows the client to view the location clustering of their client base, and view details of each provider by clicking the marker on the map to display the provider practice and address.

The client also viewed that maps use case a as a great tool to see areas where they did not have any clients situated and therefore could use marketing and communication techniques to attempt to recruit clients in these untouched areas.

The Google Maps API allowed the implementation to view, navigate, interact with and provide annotations onto a map in the CRM system. We used this to perform Geocoding on medical practice street addresses to turn the address into latitude and longitude coordinates. These coordinates then allowed us to place map markers and annotations at various addresses to allow user interaction with the map interface. As an example, the client navigates the Google maps interface on the Dashboard page of the CRM, which displays markers of where each of the providers/clients and medical practices are geographically located. There are many markers on the map around Wollongong, each clickable for more information on the practice. The client notices that the area of Corrimbal is completely devoid of markers and therefore South Coast Private Hospital have no providers who have referred patients to the hospital or prospective providers in that area.

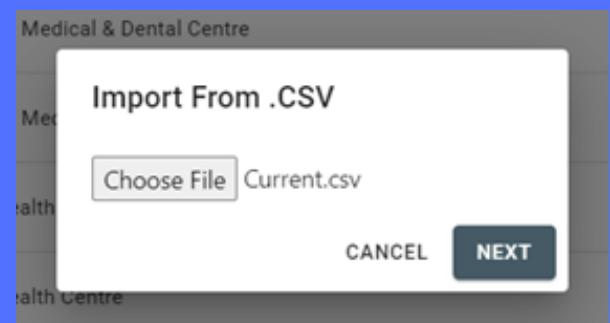
Our client then may actively search out doctors and medical practices in that area in an attempt to expand their client base and allow the access of their services to more public members in Corrimbal.



CROSS-SYSTEM INTERACTION

CSV Export/Import

The export-to-csv API allowed our program to implement a group of core use cases to the system for the client. Using this API we were able to allow the client to mass import and export selected providers to a CSV/Excel file to then be retrieved for external use in other systems such as email and fax.



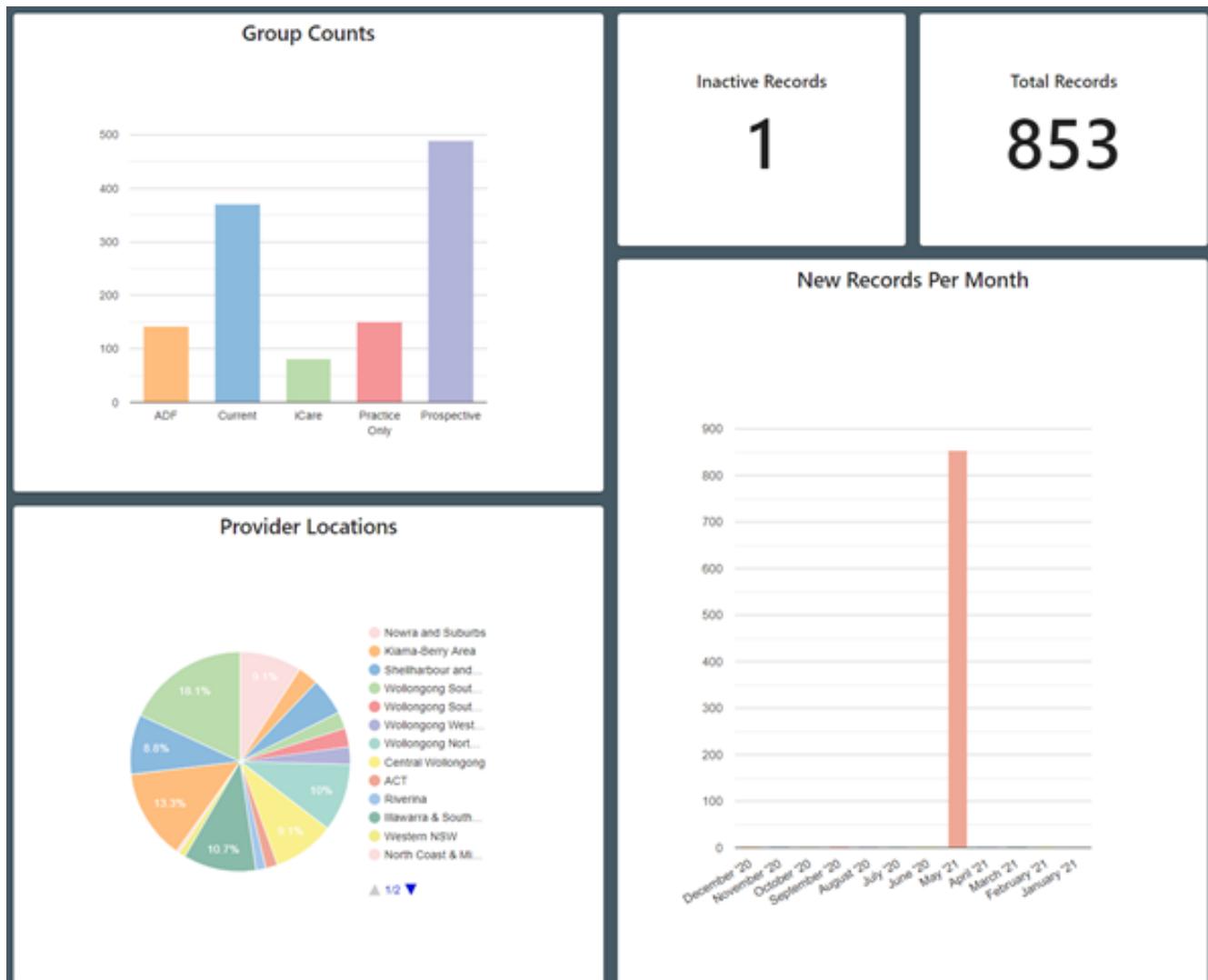
CSV Export

First Name	Last Name	Provider Number	Practice	Position	Address	Suburb	State	Phone
Gregory	A			Care Manager				
Dr. Gary	Aaron	217382YF	Australian Menopause Centre		436-438 Burwood St	Burwood	NSW	2106
Dr. Shireen	Allister	229420CT	Sheffardside Medical Centre		2-6 College Ave	Oakfield	NSW	2106
Dr. Angele	Aloham	462740BK	Castle Hill Medical Centre		4-16 Terminus Street	Castle Hill	NSW	2106
Dr. Eric	Acosta		Robertson Doctor		28 Hoddle St	Robertson	NSW	2106
Dr. S	Allan	42409679K	Milperra Healthcare Centre		63 Bivond Road	Milperra	NSW	2106
Dr. Javed	Ahmed	46036771W	CampbellBrown Medical & Dental Centre		296 Queen Street	Campbelltown	NSW	2106
Dr. Hussain	Ahmed	3790074J	CampbellBrown Medical & Dental Centre		Queen Street	Campbelltown	NSW	2106
Dr. Alfiya	Ahmed		Holsworthy Healthworks		Woolworths Bankstown	Woolworths Bankstown	NSW	2106
Dr. Afzal	Ahmed		Holsworthy Healthworks		Woolworths Bankstown	Woolworths Bankstown	NSW	2106
Dr. G.A.	Ajami	505874CX	Wanawring Medical Centre		43 King Street	Wanawring	NSW	2106
Dr. Sharmin	Alim	415380KL	CampbellBrown Medical & Dental Centre		296 Queen Street	Campbelltown	NSW	2106
Dr. Adm	Al Khamees	34252594F	Better Care Medical Centre		88-90 Princes Highway	Fairy Meadow	NSW	2106
Dr. Gays	Al Sheer	437750UK	Milton Family Medical Practice		141 Princes Highway	Milton	NSW	2106
Dr. Randa	Al-Hajri	3174878A	CampbellBrown Medical & Dental Centre		296 Queen Street	Campbelltown	NSW	2106
Toni	Albridge							
Dr. Amy	Alexander	5468002J	Engadine Family Medical Practice		3 Station Street	Engadine	NSW	2106
Dr. Chris	Alexander	2071718K	Bronx Street Medical Practice		Suite 1, 70 Bronx Street	Bronx	NSW	2106

CROSS-SYSTEM INTERACTION

Google Charts

Google Charts is an open source library that allowed us to create beautiful graphical representation of data on the system. This was important for one of our core use cases where the client requested a tiled layout with graphical representation of provider stats on the home page. The stats provided at the request of the user are number of providers in each group which is represented as a bar graph, inactive and total record which are regular tiles, provider locations which is a pie and new records per month which is also represented as a bar graph.



CROSS-SYSTEM INTERACTION

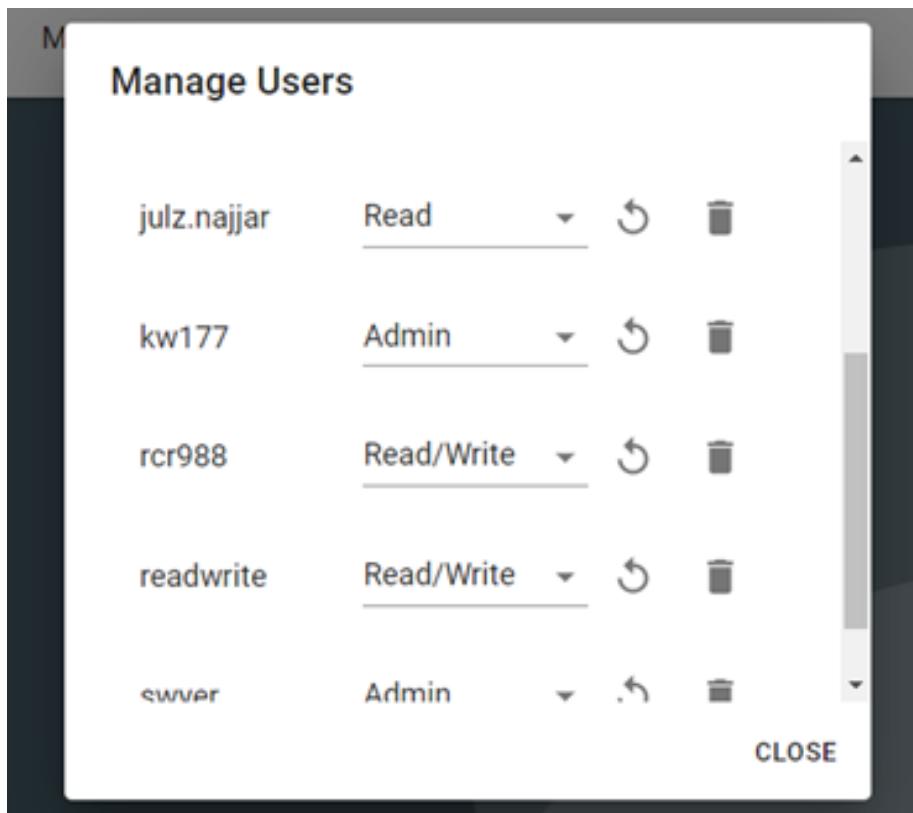
React libraries and core framework functionality

The react framework is an open-source library that we used to build our beautiful user interface. The main advantages with react is its ability to create reusable UI components, the large and robust plugin availability and its open source libraries that can be imported and used to help with any task. A small list of open source plugins used is as follows:

- AWS amplify plugin which allows our application to begin to connect and communicate with the AWS pipeline in the backend of the application.
- Web vitals is a plugin allowed us during testing to capture useful metrics about our applications performance and real time analytics.

The react framework allows the importation of testing libraries that provide APIs to perform testing on react components. Deploying these libraries and performing intricate testing on the elements of the react framework and network is vital to providing the user with a bug free and high-quality user experience.

Icon libraries such as @material-ui/core and @material-ui/icons are used in our project to provide general and universal icons that are familiar to the user and pair with a functionality that the user can click and interact with to perform a task. For example, the delete group trashcan icons allows the user to click the icon to delete a group.



DEVELOPMENT TOOLS

	Type	Details
	Material UI	Popular React UI Framework consisting of prebuilt components which allow the fast development of web applications.
	Frontend Framework	React a JavaScript Framework has been selected to develop the CRM frontend. React provides a component based structure that allows for component reuse making the frontend code base easier to maintain as it grows.
	AWS Amplify	A pre-configured AWS platform consisting of many of AWS's popular microservices. Amplify allowed the project to be developed in line with CI/CD principles.
	AWS API Gateway	RESTful API which allowed the frontend of the web application to communicate with AWS backend microservices.
	AWS Lambda	AWS event driven serverless computing platform, responsible for processing events received on the API Gateway and communicating with the DynamoDB.
	AWS DynamoDB	NoSQL data responsible for the persistent data storage requirements of the project.

AWS Services

Due to our program which harnesses the power of many AWS services in the AWS collection, a lot of our work is performed inside each individual AWS micro-service's dashboard.

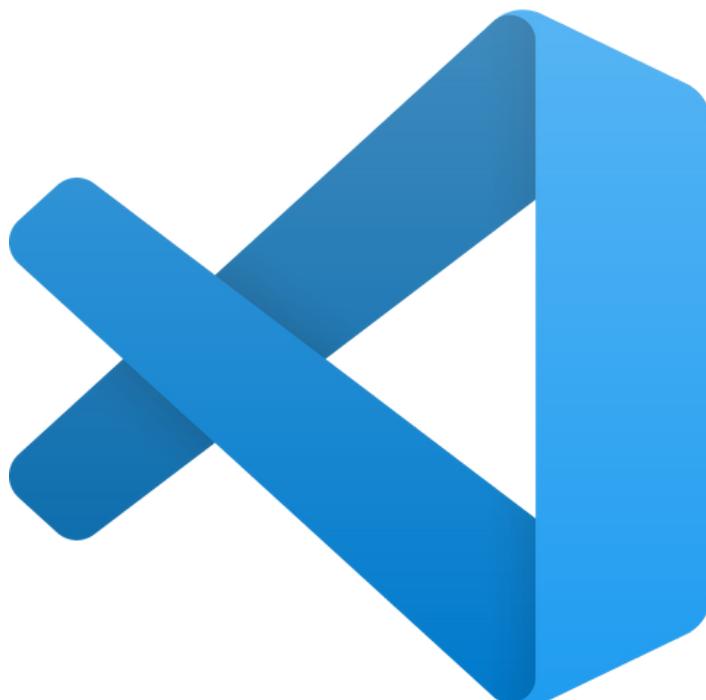
These environments allowed us to access the full functionality of the AWS micro-services and their compatibility with other closely linked AWS services in our pipeline. For AWS Lambda our serverless backend architecture, we were provided with a full development environment to write python lambda functions, testing to see connectivity and compatibility issues and the searching and linking of AWS services of the API gateway and AWS DynamoDB. We were simply able to create the API Gateway in its own AWS dashboard, work on it, configure its settings and then link to it through a simple drop-down menu in the AWS Lambda environment. Each AWS dashboard gave us the full functionality and potential for the service that we were using and implementing.

The below screenshot is an example of the AWS Lambda serverless environment which provides many functionalities such as an environment to write code, test it, configure it and perform version control. It also provides an at a glance view of the connected API Gateway and further AWS Lambda functionalities on the sidebar menu.

DEVELOPMENT TOOLS

Visual Studio Code

Visual Studio code is a robust and large-scale development environment. The front-end developers used this development environment due to its great React framework capabilities. It allows the downloading of any plugin, the cloning and pushing of our GitHub repositories and most importantly it integrates with a node.js webserver that allowed us during development to run and test our react project on a local development environment.



DEVELOPMENT TOOLS

GitHub

Continuous Integration Continuous Deployment principles were implemented during the development of the project through the use of a shared GitHub repository. This repository was integrated with AWS's Amplify platform which enabled the build and continuous deployment of the web application when code commits were integrated.

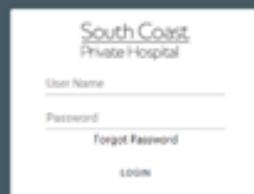
Continuous Integration Continuous Deployment principles ensured continued quality of the code base throughout the project and allowed regular builds to the production environment.



INFORMATION ARCHITECTURE

The Sign in page

The sign in page provides a minimalistic interface with text field hints to guide the user to the correct text fields and implemented their credentials. There is an option to recover your password based off the user's username and the only button on the page which is for login is self-explanatory. If the username or password is incorrect eye catching red text is generated to display to the user what they have done wrong and allows them to try again.



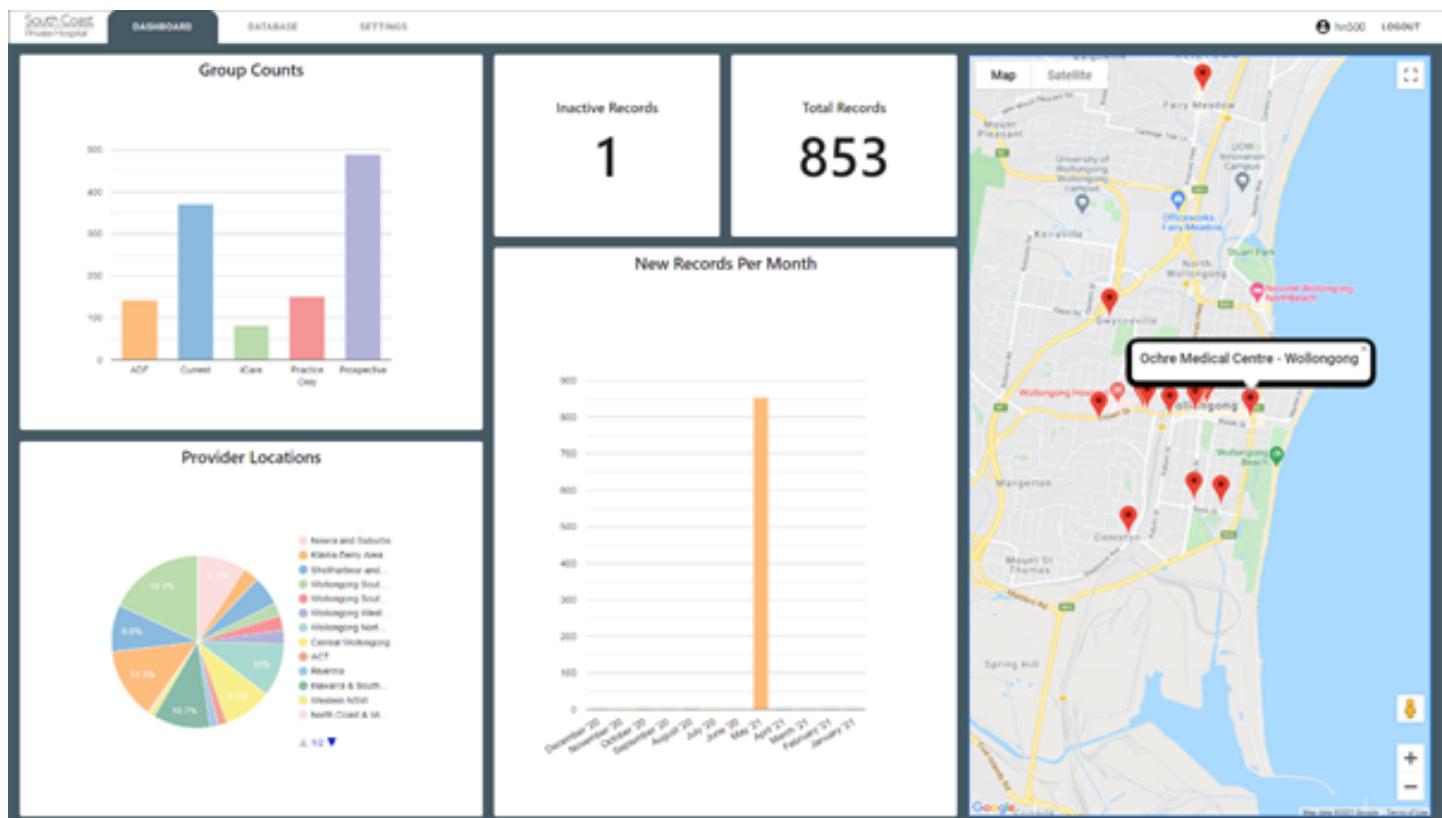
INFORMATION ARCHITECTURE

Main Dashboard

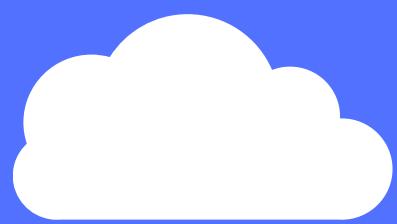
Immediately presented once logged in is the colourful collection of tile layouts and a Google map interface. This main dashboard provides an at-a-glance view of system statistics all fully labelled with headings, relevant chart variables and colour orientations.

The Google Maps section of the main Dashboard functions the same as the Google Map's app. The map is fully interactable and allows satellite view, zooming in and out and full screen all of which are usable by the relevant self-explanatory icons. In our case the Google Map has bright red/orange clickable annotations that then provide textual information about the location and its providers.

The tab-like menu provides instructions displaying where you will go if you click one of the tabs and provides a quick and easily accessible way to move to different tabs. This dashboard stays present during the entirety of the systems use.



INFORMATION ARCHITECTURE



Database tab

The list of search filters make up the entirety of the side menu, each search condition is either labelled with a text label, heading, button text or textfield hint so there is no doubt for the user on what UI component does what.

South Coast Private Hospital		DASHBOARD	DATABASE	SETTINGS	kw177	LOGOUT		
Search		<input type="checkbox"/> Title	First Name	<input type="checkbox"/> Last Name	Provider Number	Practice	Position	Address
First Name		<input type="checkbox"/> Dr	Soheir	<input type="checkbox"/> Abadier	229622CT	Shellharbour Medical Centre		2-6 College Ave
Last Name		<input type="checkbox"/> Dr	Angelie	<input type="checkbox"/> Abhaham	462745DX	Castle Hill Medical Centre		4-16 Terminus Street
Provider Number		<input type="checkbox"/> Dr	Eric	<input type="checkbox"/> Acevedo		Robertson Doctor		28 Hoddle St
Practice	MORE	<input type="checkbox"/> Dr	S	<input type="checkbox"/> Afrin	4240929H	Mittagong Healthcare Centre		63 Bowral Road
Filter		<input type="checkbox"/> Dr	Hussain	<input type="checkbox"/> Ahmed	2700074J	Campbelltown Medical & Dental Centre		296 Queen Street
ADF	<input type="checkbox"/>	<input type="checkbox"/> Dr	Jovad	<input type="checkbox"/> Ahmed	4603671W	Campbelltown Medical & Dental Centre		296 Queen Street
iCare	<input type="checkbox"/>	<input type="checkbox"/> Dr	G A	<input type="checkbox"/> Ajam	035874CX	Warrawong Medical Centre		43 King Street
MORE	<input type="checkbox"/>	<input type="checkbox"/> Dr	Sharmin	<input type="checkbox"/> Akhter	415383KL	Campbelltown Medical & Dental Centre		296 Queen Street
Action	<input type="checkbox"/> ADD	<input type="checkbox"/> Dr	Azim	<input type="checkbox"/> Al Khemesy	242525HF	Better Care Medical Centre		88-90 Princes Highway
EMAIL SELECTION		<input type="checkbox"/> Dr	Qays	<input type="checkbox"/> Al Shaer	437758UK	Milton Family Medical Practice		141 Princes Highway
FAX SELECTION		<input type="checkbox"/> Dr	Randa	<input type="checkbox"/> Al-hajali	2174518A	Campbelltown Medical & Dental Centre		296 Queen Street
Manage		<input type="checkbox"/> Dr	Chris	<input type="checkbox"/> Alexander	201519BX	Bowral Street Medical Practice		Suite 1, 70 Bowral Street

INFORMATION ARCHITECTURE

There are also many small popup ups indicated by the buttons on the side menu that allow the application of functional use cases such as, adding a new record, importing from csv and email selection. Textfield hints and text labels are used to allow the user to navigate these smaller popups.

The column titles at the top provide information about what kind of data is in the corresponding column. Then, in each row is each provider and their details. The user is able to track and highlight any record with the checkbox otherwise they can easily track each separate provider and the columns for that record.

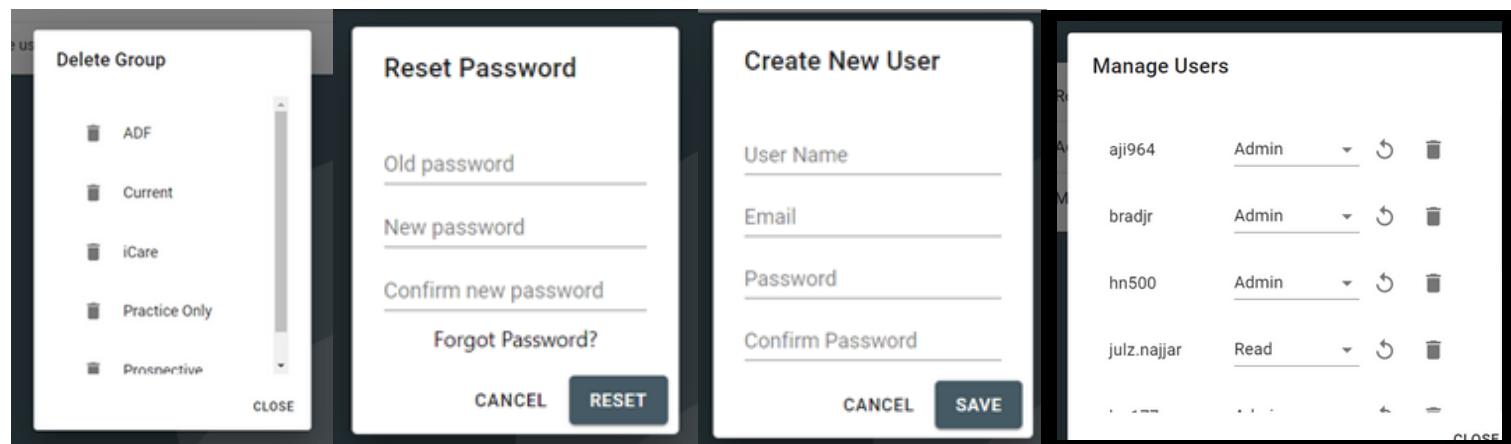
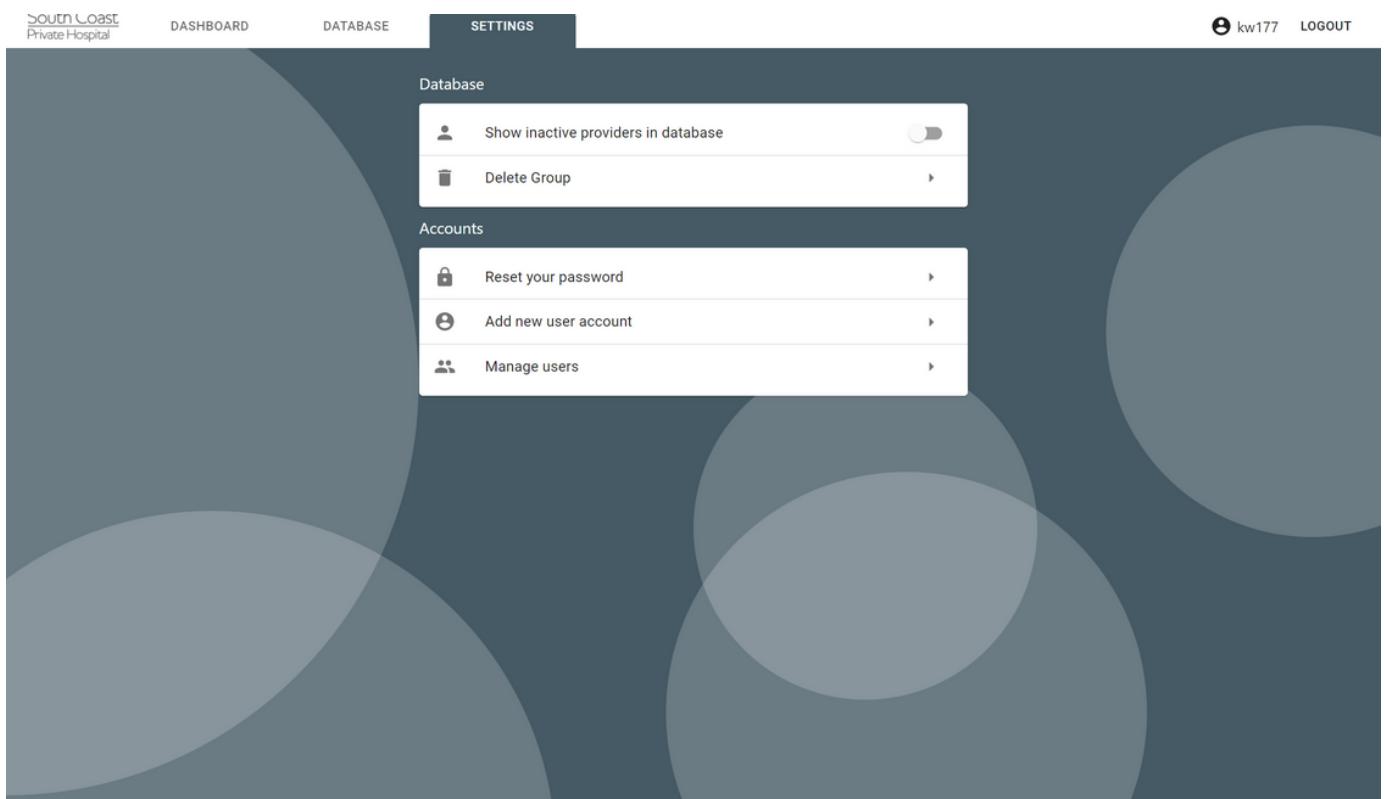
The screenshot shows a CRM application interface. On the left, there is a sidebar with various filters and actions. The main area has a table of providers with columns for Title, First Name, Last Name, Provider Number, Practice, Position, and Address. A modal window titled 'Add Record' is open in the center, containing fields for Title, First Name, Last Name, Provider Number, Email, Practice, Position, Address, Suburb, State, Postcode, Phone, and Fax. There are 'CANCEL' and 'NEXT' buttons at the bottom of the modal. The table data is as follows:

Name	Title	First Name	Last Name	Provider Number	Practice	Position	Address
Dr	Billie	Rajabali		409004FW	The Surgery At Jerra		Po Box 152
Dr	Oliver	Webber		5239481F	Family Medical Practice Health - Woonona		381 Princes Highway
Kate		Add Record					
Dr	Candice	Title	First Name	Last Name			436-438 Burwood St
Dr	Akbar	Provider Number	Email				130 Shoalhaven Road
Dr	Eric	Practice	Position				28 Hoddle St
Dr	Rachel	Address	Suburb				409 Great North Road
Dr	Jeff	State	Postcode				72-74 Park Rd
Dr	Esther	Phone	Fax				3a Bronte Road
Dr	Alison						409 Great North Road
Dr	Leslie	Pate	0095854K	Campbelltown Medical & Dental Centre			296 Queen Street
Dr	Yusufali	Khalfan	004979AF	Crown Medical Centre, Westfield Figtree			Shop 108-110/19 Princes Hig
Dr	Kevin	Ng	214545AL	Dee Why Medical Centre			Shop A, 1-5 Dee Why Pde

INFORMATION ARCHITECTURE

Settings Tab

The settings tab uses not only text to indicate the functionality for the component but also standardized icons that are familiar to all kinds of users throughout any software program. Clicking these user interface components also bring up some minimalistic popup menus providing instructions, text field hints and labels describing how to perform an action.



POST PROJECT REVIEW

Aspects	Expected Performance	Actual Performance	Reasons of Deviation
Scope	<ul style="list-style-type: none"> Project to be completed by May 31st 2021 Project scope to be achieved as per requirements analysis 	<ul style="list-style-type: none"> Project completed before 31st of May Scope has been achieved as per requirements summary 	No deviation was experienced between expected scope and performed scope. Clear scope definitions were made to ensure the team was not setting out to fail
Cost	Cost expectations were to be minimal using open-source frameworks and product tier levels to ensure minimised utility of cost.	<p>Cost of development was minimised to zero with only support costs factoring into budget such as:</p> <ul style="list-style-type: none"> Google Map API Domain cost 	Google Map API was preferred over open-source options for client's usability. Domain name cost was expected and needed for client to have a unique site name
Time	Scope of requirements to be complete by the submission due date of the project to the university	Project has been submitted to the university in the timeframe expected. Delivered to the client before their intended company strategy day end of May 2021	No deviation on timeframe to complete and deliver the project was experienced
Quality	Provider information is centralised into one database for maintenance. Duplication of data will no longer occur. Manual data entry is limited to requirements of singular entries as per data source pipeline.	Requirements summary has summarised the qualitative metrics that the project was intended to maintain as per the quality of the output product.	Additional user feedback as time goes on with use to identify if quality is still maintaining expected outcomes.

PROCESS IMPROVEMENT

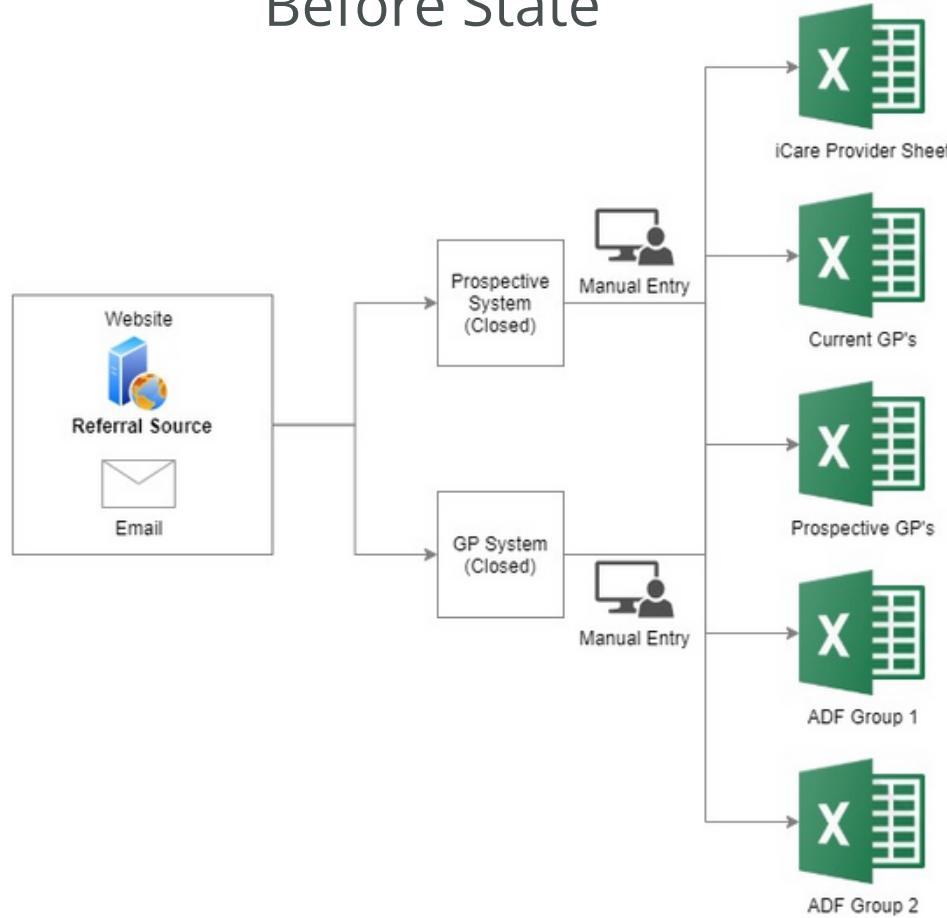
Before State

As part of the process discovery, the following process map was identified as the “before state” of the scope of elements that the client was using for provider data management. The process of taking provider information from its referral source would be conducted through two private closed systems that had limited interactions and heavy manual processes. Data was stored across multiple excel spreadsheets, which caused issues with version control of up-to-date information. Data duplication had no system constraints to prevent such cases. Visualisation of data was also limited. The goal of the project was to limit this process model to a singular flow where data could be stored and controlled in a centralised repository

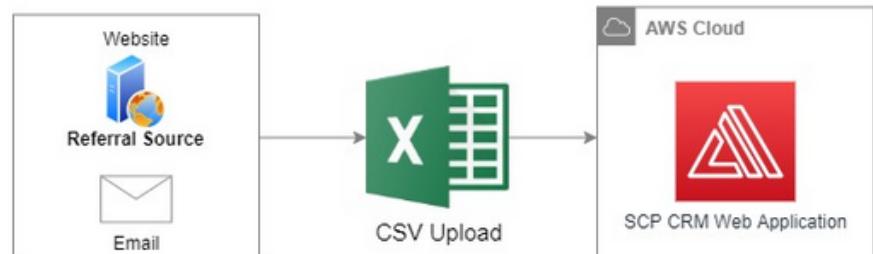
After State

The after state heavily reduces time spent on handling and controlling data. With a simple csv upload with manual entry for singular data events, the CRM solution reduces the pipeline down to one singular flow.. Information is centralised in one convenient database

Before State



After State



ACCEPTANCE CRITERIA

During the initial requirements analysis phase of the project a set of acceptance criteria were established to ensure the final product met the user's need.

Outlined in the table below is a summary of the criteria and their final acceptance status.

Acceptance Criteria	Status
View provider profile	✓
View all providers	✓
Filter providers	✓
Search for provider	✓
Add new provider	✓
Edit provider	✓
Export providers to excel	✓
Create distribution lists	✓
Edit distribution lists	✓
Edit distribution lists	✓
View distribution lists	✓
Add provider visit information	✓
Add provider communication	✓
Add provider marketing material	✓
View no. of referrals	✓
View map displaying providers	✓
View breakdown of provider locations	✓
Create system user	✓
Delete system	✓
Edit system user settings	✓

TRANSITION PLAN

The project team will conduct a two-step process in transitioning the digital assets created in this development across to the client South Coast Private Hospital. Step one will consist of the data transfer of the following digital assets:

Asset Transfer:

- GIT Hub Code Repository
- Development branches will be dropped with Main branch remaining.

Amazon Web Services

- AWS Cognito
- AWS API Gateway
- AWS Lambda
- AWS Amplify
- AWS Route 53
- IAM Development Users will be dropped.
- Account transfer form will be submitted to AWS Support by the client

Google Development

- Google Development API
- Account for API map integration created and transferred to clients.

Document transfer of all associated document artifacts created in project development through email.

Ongoing Support

As part of ongoing continuous engagement discussions with the client, the project team will be offering support options when it comes to the clients needs. As stated in the initial scope document of this project, it was intended that the project transition would need to allow the client to control complete maintenance of the product. This is still the expected outcomes of the deliverables with the exception of one team member offering their individual free-lancing services as discussed with the client.

It is recommended that this engagement be involved on a request-by-request basis with expected scope and deliverables to be set per discussions. Dependent on the impact of the request, it is recommended that the team member involved in these engagements propose the standard of expectations and deliverables for the client to expect based upon their individual availability.

Additional recommendation is based upon the understanding that the client intends to re-engage with the university for the CSIT321 subject for additional projects. Dependent on the request and making assumptions based upon South Coast Private Hospital's needs, it is suggested that the client consider a possible integration of a new project system to the South Coast Private CRM. Contained within the scope of future projects could include new capability in accordance with the client's needs and potential features that didn't fit within the required scope of the CRM development phase 1. Potential future product state opportunities have been outlined in the future product considerations section of this report.

LESSONS LEARNED

The following lessons learned evaluation has been produced through the performance of a project retrospect, a popular agile method of gaining retrospective information from the development team of how the project went. The insights gained from this retrospective brain storming session have been categorised against key project metrics:

Category	Issue Name	Problem/Success	Impact	Recommendation
Risk	Risk Management	A risk regarding experience in the AWS Cloud development was identified. Root AWS account was locked out of when an MFA device was unrecoverable. Root access was unable to fall back on phone verification as the number was the client's switchboard and not a direct number.	Inability to predict billing costs of development. The clients billing account needs to be recreated with a request to AWS	When working within AWS, it is recommended to completely understand the structure of the user account levels. By enabling IAM billing access to a secondary user, this enables the account to be completely managed to change contact information such as the contact number
Scope	Time/Planning Fallacy	Timeframe to available time ratio to develop the product. (Yearly time frame for development to fit within fulltime study)	Impact was minimal as the team was able to still fit the development within the required scope	Consistent weekly team check-ins to ensure continuous progression
Resources/Scope	Upskill	Teams' exposure and maturity to cloud technology was initially limited	Learning curve impacting proficiency in understanding product application	Conducting team learning sessions with encouragement for cloud entry certification
Development	Development Environment	Product was made available to client for user acceptance testing without a UAT environment setup	Pushing changes to production impact data that the client was testing within the system	Setup and deployment of UAT environment to run alongside production
Resources	Digital Teams	Due to the COVID-19 pandemic, the entire project was conducted through virtual communications	Engagement and information transfer required support channels to ensure consistency	Build standard communication channels along with consistent weekly engagement touch points



FUTURE PRODUCT CONSIDERATIONS

As part of the scope analysis, product components and ideas considered to be out of the scope of the project have been theorised into potential future product features. These concepts/features are part of bringing seamless improvement to the capability of the product in alignment with the needs of South Coast Private Hospital.

Outside of project scope (Where this product could go):

- API integration to Provider Health site for automotive data validation
- Mailing integration to a communication system
- Page refresh to maintain user session state.
- Dashboard reactivity and dynamic changes (User can create their own reports and move them around on the page)
- API Go fax integration.
- Improved data intake using AWS data cleaning tools on csv imports (S3 Bucket into data cleanse)
- Improved duplicate check throw backs as part of the csv import process
- Separate phone fields for mobile/office phones
- Move all http methods into utility functions.
- Formation on GoogleMap InfoWindow

SYSTEM INSTALLATION

Due to the cloud nature of the product, there is no required installation process for putting the project into production. A domain has been purchased and associated with the AWS Amplify application using AWS Route 53 that can be accessed at the following address:

<https://www.scphcrm.org/>
(Contact rcr988@uowmail.edu.au for access)

The site is supported and optimised for the google chrome web browser. To install the google chrome browser to your computer, follow the below link:
[Google Chrome - Download the Fast, Secure Browser from Google](#)

It is noted that this application does not support mobile browser access.

DATA DICTIONARY

Field Name	Data Type	Data Format	Field Size	Description	Example
Active	Boolean	True/False	1	Determine if provider is active in the system or not	True = Active False = Inactive
CreationDate	String	DD/MM/YY	8	Date the provider was created by the user	20/05/21
Display	Boolean	True/False	1	Determines if the doctor will be visible in the system or not	True = Visible False = Invisible
Email	String			The providers email address	James@daptomed.com
Fax	String			The providers fax address	john@6278444
FirstName	String			The first name of the provider	George
Group	List: String			The group the provider has been marked as apart of	iCare ADF Prospective
ID	String			The providers unique identifier in the data base	013a03fa-e97f-46e5-8a11-c8b7d9f8fe98
LastName	String			The providers last name	Smith
Latitude	Number	NN.NNNNNNN		The geographic latitude of the practice of the provider works	-34.33333331
Longitude	Number	NNN.NNNNNNN		The geographic longitude of the practice of the provider works	150.8476776
Phone	String	XX XXXX XXXX		The providers phone number	02 4295 1111
Postcode	String	XXXX		The providers work postcode	2500
Practice	String			The place where the provider works	Corrimal Medical Centre
ProviderNumber	String	NNNNNNNNX		The unique provider number of the doctor	1234567K

DATA DICTIONARY

Field Name	Data Type	Data Format	Field Size	Description	Example
Referrals	Number			The number of times the doctor has been referred to by South Coast Private	4
State	String			The state the doctor practices in	NSW
Street	String			The street the address of the medical practice where the doctor works	Shop 1, Medical Way, Princes Highway
Suburb	String			The suburb of the medical practice where the doctor works	Fairy Meadow
Title	String			The doctors title	Dr
Position	String			The position of the doctor at their medical practice	Registrar
CommsHistory	String			The communication history between the user and the doctor	"Julia: Hello how are you?" Paul: "Good thanks"
ID	String			Unique ID for the records history	01a03f-e97f-8a11-c678fdhdj
MerchHistory	String			History of merchandise transactions with the doctor	"Sold South Coast Private T-Shirt"
RecordHistory	List: String			History of operations performed on the providers record	"Thu May 10 11:43:42 2021 - User created new record"
VisitHistory	String			The visit history of the doctor to South Coast Private	"Paul visited for business meeting Wed May 9"
Name	String			The name of the provider group	ADF

STYLE GUIDE

Company Logos



CRM TECHNOLOGIES

Fonts

Houschka Light
Aa Bb Cc Dd Ee Ff Gg
Hh Ii Jj Kk Ll Mm Nn
Oo Pp Qq Rr Ss Tt Uu
Vv Ww Xx Yy Zz

Houschka Medium
Aa Bb Cc Dd Ee Ff Gg
Hh Ii Jj Kk Ll Mm Nn
Oo Pp Qq Rr Ss Tt Uu
Vv Ww Xx Yy Zz

Houschka Light
Aa Bb Cc Dd Ee Ff Gg
Hh Ii Jj Kk Ll Mm Nn
Oo Pp Qq Rr Ss Tt Uu
Vv Ww Xx Yy Zz

Main Colour 1:

White
R255 G255 B255
C0 M0 Y0 K0
#FFFFFF

Main Colour 2:

Grey Blue
R120 G144 B156
C0.23 M0.08 Y0 K0.39
#78909C

