**Bachelor of Computing System and Graduate DIploma in Computing**

**Unitec New Zealand**

5 Nov 2023

Assignment

Cloud Design and development

Table of Contents

[Introduction: 2](#_Toc1908367093)

[Vision Statement 3](#_Toc634047369)

[Product Backlog 4](#_Toc109824000)

[User stories 4](#_Toc444444533)

[Requirements and Specifications 5](#_Toc1153211714)

[Scope and Prototype 6](#_Toc1513391029)

[Application Architecture 8](#_Toc916496498)

[User Interfaces – 1. Patient Signup and Login 10](#_Toc528030954)

[2. Patient Add/Edit/Save/Delete Schedule. 11](#_Toc540654701)

[3. Patient Submit their medication intake. 12](#_Toc237288210)

[4. Patient View their medication intake history 12](#_Toc671845451)

# Introduction:

Managing medication can be a complex task, especially for people who need to take it regularly. This challenge extends to those responsible for their care. In response, we introduce the Everdose Application—a mobile application designed to tackle the intricacies of medication adherence.

Taking medicine isn't always straightforward. For individuals with chronic conditions or specific treatment plans, keeping track of different medications, dosages, and schedules can be confusing. This complexity is even more daunting for caregivers, who want to ensure their loved ones receive the best care possible.

Everdose is like a personal assistant for medicine. It keeps track of when to take each medicine, how much to take, and even reminds you so you don't forget. This means less worry and more confidence in knowing you're getting the right care.

By providing clear schedules, detailed instructions, and timely reminders, Everdose turns the sometimes-complicated task of taking medicine into a routine that's easy to follow. This not only reduces stress but also helps the treatment work better.

# Vision Statement

Our vision is to empower patients and caregivers by providing a user-friendly, reliable, and secure medication management platform. We strive to ensure patients never miss a dose, while enabling caregivers to offer timely support. Together, we enhance healthcare adherence and peace of mind

# Product Backlog

The product blog consists of user stories from patient and caregiver, the user stories which achieves the essential functionality of the app is considered high as well as dependency.

## User stories

|  |  |  |
| --- | --- | --- |
| ID | Description | Priority |
|  | As a patient,  I want to be able to register an account with login details,  So that only registered authorized users can see or update my medication schedules. | High |
|  | As a patient,  I want to be able to set up a schedule for my prescribed medication,  So that I can keep track of all my medications. | High |
|  | As a patient,  I want to be able to view my medication history in the app,  So that I can see whether I've missed any doses. | Medium |
|  | As a patient,  I want to be able to receive reminders for when I need to take medications,  So that I don't forget to take them when I need them. | High |
|  | As a patient,  I want to be able to edit or delete my schedule,  So that I can keep my schedules up to date to what I’ve been subscribed. | Medium |
|  | As a caregiver,  I want to be able to sign up and login and view one or more patient(s) schedule,  So, I can track patients assigned to me. | High |
|  | As a patient,  I want to be able to grant access to a caregiver,  So that they can provide adequate care. | High |
|  | As a caregiver,  I want to be able to view any patient's dose list in the app,  So that I can make sure they're taking the right medication at the right time. | Medium |
|  | As a caregiver,  I want to receive notifications when the patient misses a medication dose,  So that I can remind them to take it. | High |
|  | As a caregiver,  I want to be able to view the patient's medication Intake history.  So that I can see if there are any patterns or issues that need to be addressed. | Medium |
|  | As a caregiver,  I want to be able to view a patient's profile in the app,  So that I can access their contact details, DOB and NHI number | Medium |

## Requirements and Specifications

**User Story 1**

* Patient/Carer can register in the app and login successfully.
* An unregistered user cannot login and use the app.
* The app displays relevant error messages for unregistered users.
* User can change password or reset password (forgot [password)

**User Story 2**

* Patients can add a schedule of medications prescribed including time, day, dose, and frequency.
* The app displays a list of medications prescribed to patients.
* Patients can update schedule of medications prescribed and the apps updates the list and saves.
* Patients can delete the schedule of medications prescribed and the apps updates and saves.
* All data from schedules are stored in the medication schedule database.

**User Story 3**

* The app displays medication name, dosage, and schedule and which has been taken and which are pending.
* The app displays date and time scheduled in each history.
* The app should display and highlight missed doses.
* The patient can mark a specific history as taken and the app updates the list and saves.
* Any changes from history are stored in the medication schedule database.

**User Story 4**

* The app displays a reminder on a patient’s phone, 15 minutes prior to the medication is due.
* The reminder displays times (morning, afternoon, dinner, or bedtime medication) of the medication due.
* The reminder lists all the medication due for that specific time.

**User Story 5**

* Patients can select and edit the details (medication name, dosage, time, day, frequency) of an existing schedule in the app.
* The edited details are saved in the medication schedule database.
* The app displays an updated schedule list, reflecting the changes made by the patient.
* Patients can select and delete an existing schedule in the app.
* The deleted schedule is removed from the medication schedule database.
* The app displays an updated schedule list, excluding the deleted schedule.

**User Story 6**

* The app should validate the user's credentials (username and password) against the stored data.
* The app should check if account exist or not, appropriate message displayed for account doesn't exist.
* Appropriate error messages should be displayed for invalid login attempts (e.g., incorrect username or password).
* Clear instructions or hints should be provided to users if their login fails.
* Once logged in, a list of patients is displayed sorted by the whom the medication is due next.
* Each patient's schedule should show relevant details such as schedule times, dates, and any other necessary information.
* If no patients are assigned to my account, a message should be displayed indicating that there are no schedules to view.

**User Story 7**

* Patient can access a settings or profile page where he can manage access permissions.
* On the settings or profile page, there should be an option to grant access to a caretaker.
* When selecting the option to grant access, the patient should be able to provide the caretaker's email address or username.
* After entering the caretaker's email address or username, the system should send an invitation to the caretaker.
* The invitation should include a clear message explaining that I, as the patient, he/she is granting access to their health information and schedules.
* When the caretaker receives the invitation, they should be able to click on the provided link to accept it.
* Once the caretaker accepts the invitation, they should be granted access to the patient's schedule and relevant health information.
* The patient should have the ability to manage and revoke access permissions for caretakers at any time.
* If the patient revoke accesses the caretakers can no longer access patients’ data.

**User Story 8**

* Caregiver can view the correct dose plan for any selected patient.
* The dose plan displays the correct medication name, dosage, interval, and method for the selected patient.
* The dose plan displays all the medications assigned to the patient’s schedule. No additional medications are displayed.
* Caregiver can't edit or delete the dose list. They can only view it.

**User Story 9**

* The app should check whether a patient took a scheduled dose or not.
* The app notifies a caregiver about the missed doses after 60 minutes.
* The app can only send notifications to the caregivers who are in charge of the patient.
* the notification shows who missed dose, when scheduled, details of dosage.
* The app should support both online and offline notification.
* Online notification should be app notification on a screen.
* offline notification should be text message.

**User Story 10**

* Caregiver can view patient's medication history in the app, upon authorization.
* The app displays medication name, dosage, and schedule, indicating whether the dose has been taken or is pending for each patient.
* The app displays date and time scheduled in each patient's history.
* The app should highlight missed doses in the patient's history.
* Caregiver can't edit or delete the medication history. They can only view it.

**User Story 11**

* Caregiver can view the correct profile for any selected patient.
* The profile displays the correct patients name, mobile no, email address, DOB and NHI Number
* Caregiver can't edit or delete the profile. They can only view it.

# Scope and Prototype

**Scope**

The scope of the application aims to facilitate medication management for the patients and caregivers. Users can log in, view a calendar, add, edit medication schedules, receive reminders, and submit medication intake status. Caregivers have dashboard access, enabling them to monitor patients’ schedules and receive alerts for missed dosses and access patient history. The system emphasizes user-friendly interfaces and secure authentication. Caregivers can be granted or revoked access. The application prioritizes patient privacy, timely medication reminders, and seamless caregiver- patient collaboration.

**Prototype**

The prototype features a streamlined user interface with login/signup screens, a calendar based home page and an intuitive schedule creation and editing process. Features of the Prototype are listed below.

1. Authentication
   1. The application has a user-friendly login and sign-up screens with secure authentication.
   2. It has different login options for caregivers and patients.
2. Home Page
   1. The app offers a calendar view for quick date reference.
   2. It also has an Add Schedule button for seamless schedule creation.
3. Add/Edit Schedule
   1. An interactive interface for entering the schedule (title, date, frequency)
   2. Options for setting medication reminders, specifying dose and adding notes.
   3. Save and discard buttons ensure easy schedule management.
4. Schedule Details
   1. Clickable schedules on the home page for detailed views.
   2. Edit and delete options within the schedule details for quick adjustments.
5. Medication Reminder
   1. Push notifications provide timely reminders for medication intake.
   2. The "Submit" button allows users to confirm they've taken the prescribed dose.
6. Caregiver Access
   1. User-friendly interface to add caregiver contact details.
   2. Grant and revoke access options, ensuring control over caregiver interactions.
7. Caregiver Dashboards
   1. Clear dashboard displaying a list of associated patients for the caregiver.
   2. Quick access to individual patient schedules for efficient monitoring.
8. Alert History
   1. Instant alerts notify caregivers of missed doses.
   2. Accessible patient history and dose plans provide comprehensive insights.
9. Overall Design
   1. Consistent and visually appealing design for a seamless user experience.
   2. Wireframes visually represent each screen, emphasizing simplicity and clarity.

# Application Architecture

This architectural design outlines a medication tracking application using Amazon Web Services (AWS). It employs AWS services for user management, data storage, notifications, security, and scalability. The design emphasizes secure patient management, caregiver support, medication reminders, and data analytics, ensuring a robust and flexible solution for healthcare needs.

**1. User Interface (UI):**

* Develop the user interface as a web application.
* Host the UI using Amazon S3 for static web content or AWS Elastic Beanstalk for web applications.

**2. Authentication and Authorization:**

* Use Amazon Cognito for user authentication, including OAuth 2.0 and JWT support.
* Implement fine-grained access control with AWS Identity and Access Management (IAM).

**3. Patient Management:**

* Store patient profiles and caregiver access permissions in Amazon DynamoDB, a NoSQL database for flexibility and scalability.

**4. Medication Schedule Management:**

* Store medication schedules in DynamoDB.
* Create a RESTful API using AWS API Gateway and AWS Lambda for interaction with schedules.

**5. Medication Reminder Service:**

* Use AWS Simple Notification Service (SNS) for sending reminder notifications.
* Create a scheduled AWS Lambda function to send reminders.

**6. Caregiver Management:**

* Store caregiver profiles and relationships in DynamoDB.
* Implement RESTful API for caregiver management using API Gateway and Lambda.

**7. Data Storage:**

* Use Amazon RDS (Relational Database Service) for storing structured data, such as user profiles, caregiver information, and historical data.
* Use Amazon DynamoDB for unstructured data and high-velocity data storage.

**8. Patient-Caregiver Communication:**

* Use AWS Simple Notification Service (SNS) for sending notifications to caregivers.
* Configure email and SMS notifications using Amazon SES and Amazon SNS, respectively.

**9. License Agreement Management:**

* Store license agreement acceptance status in DynamoDB.
* Implement logic to display the agreement in the UI.

**10. Security Layer:**

* Leverage AWS Web Application Firewall (WAF) to protect against web application attacks.
* Use AWS Certificate Manager for SSL/TLS certificates.
* Secure APIs using API Gateway with built-in security features.

**13. Web Clients:**

* Use AWS Mobile SDK for seamless integration with AWS services on mobile platforms.

**14. High Availability and Load Balancing:**

* Implement AWS Elastic Load Balancing (ELB) to distribute traffic and ensure high availability.

**15. Backup and Disaster Recovery:**

* Implement regular backups using AWS Backup for critical data.
* Set up disaster recovery using AWS Disaster Recovery services.

**16. Monitoring and Logging:**

* Use Amazon CloudWatch for application and infrastructure monitoring.
* Configure AWS CloudTrail for audit and log trail.

# User Interfaces – 1. Patient Signup and Login

A screenshot of a cell phone

Description automatically generatedA screenshot of a login screen

Description automatically generatedA screenshot of a sign up form

Description automatically generatedA blue circle with a white pill and white text

Description automatically generated

Patient accepts the Patient License Agreement by Pressing the “Accept” button. Then system directs to Login Screen

A splash Screen while app is loading.

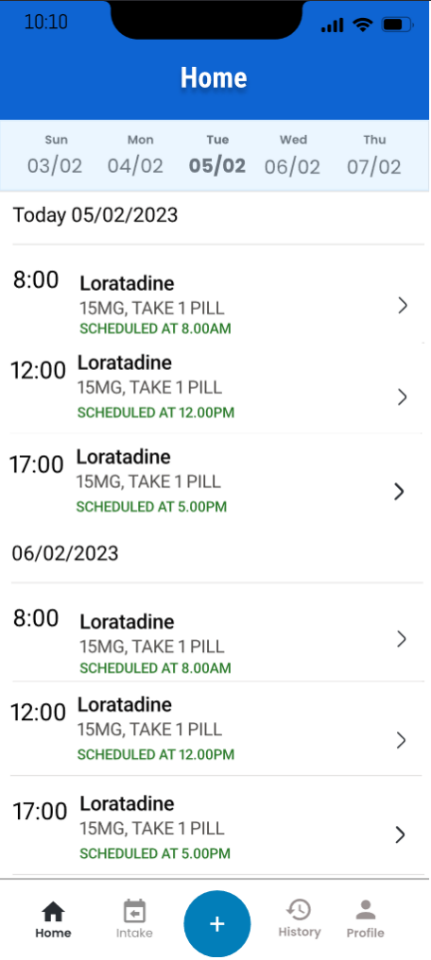
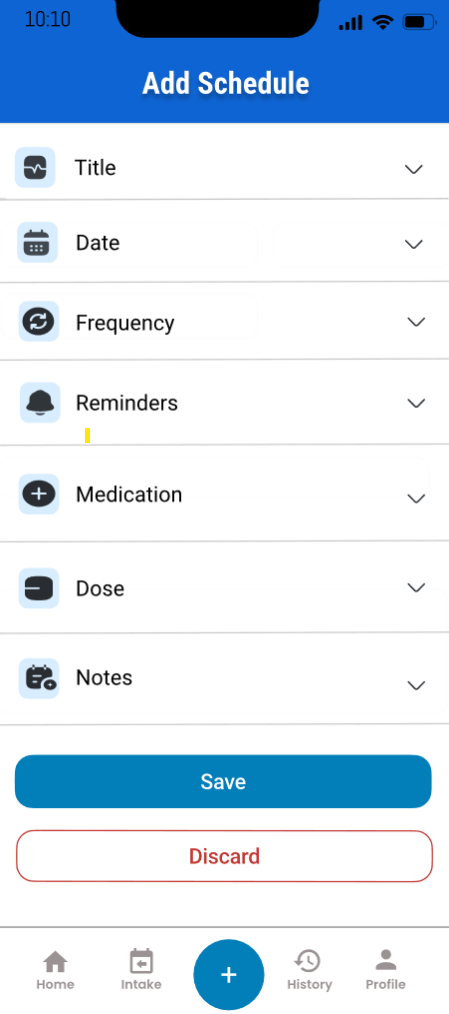
If the patient already has an account, they can click on “Login label this will take patient to Login screen.

Patient presses the signup button. System validates the details and directs Patient to License Agreement Screen.

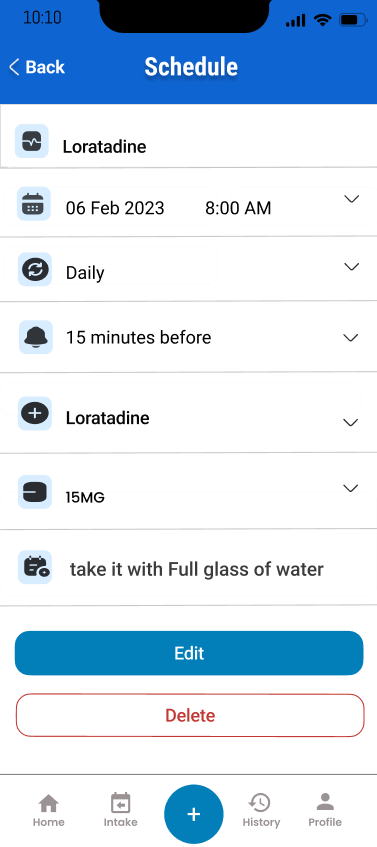
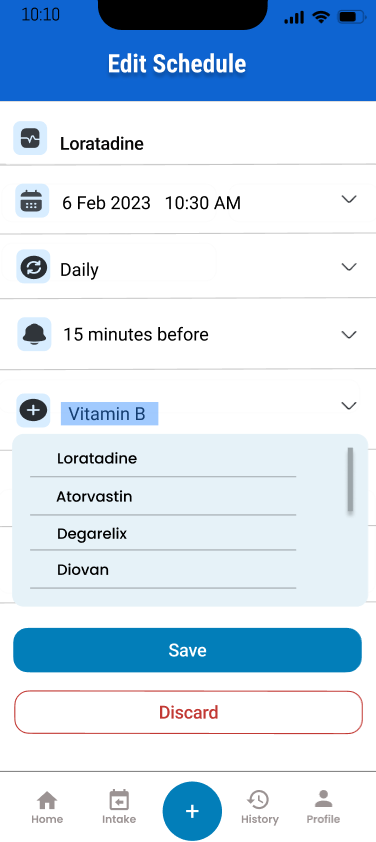
Patient enters their email address, phone number and password.

Patient enters their email and address and password and presses “Login” button. System validates the details and directs patient to “Home” Screen.

## 2. Patient Add/Edit/Save/Delete Schedule.



Existing Schedule for the day.



Patient clicks on “+” button to enter new schedule. The system directs to “Add Scheule” screen,

Patient enters the Title, Date, Frequency, Reminder time, Medication, Dose and Notes. Patient Presses the Save button to save the schedule. System saves the schedule and redirects to “Home Screen”.

Patient taps on existing schedule, and this edit screen opens.

Patient can edit Title, Date, Frequency, Reminder time, Medication, Dose and Notes. Patient Presses the Save button to save the schedule. System saves the schedule and redirects to “Edit Screen”.

Press edit button to edit the schedule. System directs to “Edit Schedule” screen.

Press “Save” to save the changes made. System directs to “Edit Screen”.

Press “Discard” to discard the changes made. System directs to “Edit Screen”.

Press delete button to delete the Schedule. System directs to home screen.

## Screens screenshot of a phone screen Description automatically generatedA close up of a blue and orange fabric Description automatically generated3. Patient Submit their medication intake.

Pressing submit button marks the medication as taken and message is displayed.

Medication Details, including submit button.

Patient receives the notification and tapes on it. System directs to submit intake screen.

A screenshot of a phone

Description automatically generatedScreens screenshot of a phone

Description automatically generated

Press close to return to Submit intake screen.

Medication marked as taken.

Submit intake screen can also be accessed through by tapping on this icon.

## 4. Patient View their medication intake history

## A screenshot of a phone Description automatically generated

Patient selects the date and history of medication intake displayed according to that.

Patient taps on History Icon button to view the history.