CovidTracker Sprint 4

By

Andre Ibrahim - 40132881

Dan Raiu - 40108722

Daren Kafafian - 40100511

Domenic Seccareccia - 40063021

Ejazali Rezayi - 40101892

Jason Gerard - 40079266

Khagik Chris Astor - 40099665

Lucas Blanchard - 40060670

Rafi Stepanians - 40108731

A report submitted in partial fulfillment of the requirements of SOEN 390

Concordia University

April 6, 2022

TABLE OF CONTENTS

LIST OF TABLES **5**

LIST OF FIGURES **8**

1.0 INTRODUCTION **13**

1.1 Positioning 13

1.1.1 Problem Statement 13

1.1.2 Product Position Statement 14

2.0 PROJECT DESCRIPTION **15**

2.1 Stakeholders 16

2.1.1 Stakeholder Roles 17

3.0 REQUIREMENTS **20**

3.1 User Stories 20

3.2 Backlog 21

4.0 RELEASE PLANNING **22**

4.1. Sprint 4 22

4.1.1 Summary 22

4.1.2 Retrospective 23

4.2 Sprint 5 24

4.2.1 Planning 24

5.0 SOFTWARE ARCHITECTURE **25**

5.1 Stakeholder Concerns 26

5.2 Viewpoints 27

5.2.1 Context Viewpoint 27

5.2.1.1 System Context Diagram 27

5.2.1.2 Use Case Diagrams 28

5.2.1.2.1 User 28

5.2.1.2.2 Administrator 29

5.2.1.2.3 Patient 30

5.2.1.2.4 Doctor 31

5.2.1.2.5 Health Official 32

5.2.1.2.6 Immigration Officer 33

5.2.2 Functional Viewpoint 33

5.2.2.1 Component Diagrams 33

5.2.3 Information Viewpoint 36
5.2.3.1 Domain Model 36
5.2.4 Development Viewpoint 38
5.2.4.1 Status Management 38
5.2.4.2 Patient Management 41
5.2.4.3 Messaging System 44
5.2.4.4 Dashboards 47
5.2.4.5 Contact Tracing 48
5.2.4.6 Appointments 50
5.2.5 Deployment Viewpoint 52
5.2.5.1 Deployment Diagram 52
5.3 Tech Stack 53
5.3.1 Presentation Tier 53
5.3.2 Application Tier 53
5.3.3 Data Tier 54
5.3.4 Deployment 54
5.4 External Libraries 55
5.4.1 Vuexy 55

6.0 RISK ASSESSMENT AND MANAGEMENT PLAN 57

7.0 USER INTERFACE DESIGN 60
7.1 Personas 60
7.2 Supported Devices 63
7.3 UI Mockups and Prototypes 67
7.3.1 Sign Up 68
7.3.2 Sign In 71
7.3.3 Sign Out 72
7.3.4 Add a Role 74
7.3.5 Assign Patient to Doctor 76
7.3.6 Define Status Report 78
7.3.7 Status Report 80
7.3.8 Number of Patients Assigned to a Doctor 82
7.3.9 Patient List 84
7.3.10 Status Reports 89
7.3.11 Add Test Result 92
7.3.12 Status Report Inbox 94
7.3.13 Status Report Details 96

7.3.14 Test Results 100
7.3.15 Test Result Details 107
7.3.16 Chat 114
7.3.17 Book Appointment 118
7.3.18 Appointments 121
7.3.19 Add Location 126
7.3.20 Contact Tracing 128
7.3.21 Contact Tracing Contacts 131
7.3.22 Dashboard 133

8.0 TESTING PLAN AND REPORT 146
8.1 Unit Tests 146
8.1.1 Client 146
8.1.2 Server 146
8.2 Integration Tests 147
8.2.1 Client 147
8.2.2 Server 147
8.3 Acceptance Tests 147
8.4 System Tests 169
8.5 Test Code Coverage 185
8.5.1 Client 185
8.5.2 Server 185

9.0 DEFECT TRACKING AND REPORT 187

10.0 QUALITY MEASUREMENTS 188
10.1 Metrics Used 188
10.2 Cause of Results 188
10.3 Improving the Results 189

APPENDIX A: TEAM COLLABORATION AND COMMUNICATION 190
A.1 Collaboration 190
A.2. Communication 190
A.3 Tools 191

APPENDIX B: GLOSSARY 192

REFERENCES 193

LIST OF TABLES

Table 1 Problem Statement 13

Table 2 Product Position Statement 14

Table 3 Sprints Schedule 15

Table 4 Supplementary Information 25

Table 5 Stakeholder Concern Traceability Matrix 26

Table 6 User Account Credentials for CovidTracker 55

Table 7 Risk Analysis Table 57

Table 8 Acceptance Test for COV-42 148

Table 9 Acceptance Test for COV-48 149

Table 10 Acceptance Test for COV-52 150

Table 11 Acceptance Test for COV-85 150

Table 12 Acceptance Test for COV-26 151

Table 13 Acceptance Test for COV-95 152

Table 14 Acceptance Test for COV-25 153

Table 15 Acceptance Test for COV-27 154

Table 16 Acceptance Test for COV-107 154

Table 17 Acceptance Test for COV-112 155

Table 18 Acceptance Test for COV-124 156

Table 19 Acceptance Test for COV-123 156

Table 20 Acceptance Test for COV-111 157

Table 21 Acceptance Test for COV-157 157

Table 22 Acceptance Test for COV-113 158

Table 23 Acceptance Test for COV-115 159

Table 24 Acceptance Test for COV-114 159
Table 25 Acceptance Test for COV-121 160
Table 26 Acceptance Test for COV-122 161
Table 27 Acceptance Test for COV-108 161
Table 28 Acceptance Test for COV-171 162
Table 29 Acceptance Test for COV-119 163
Table 30 Acceptance Test for COV-116 163
Table 31 Acceptance Test for COV-172 164
Table 32 Acceptance Test for COV-169 165
Table 33 Acceptance Test for COV-126 166
Table 34 Acceptance Test for COV-120 166
Table 35 Acceptance Test for COV-222 167
Table 36 Acceptance Test for COV-223 168
Table 37 Acceptance Test for COV-125 168
Table 38 System Test for COV-42 170
Table 39 System Test for COV-48 170
Table 40 System Test for COV-52 171
Table 41 System Test for COV-85 171
Table 42 System Test for COV-26 172
Table 43 System Test for COV-95 172
Table 44 System Test for COV-25 173
Table 45 System Test for COV-27 174
Table 46: System Test for COV-107 174
Table 47 System Test for COV-112 175
Table 48 System Test for COV-124 175
Table 49 System Test for COV-123 175

Table 50 System Test for COV-111 176

Table 51 System Test for COV-157 176

Table 52 System Test for COV-113 177

Table 53 System Test for COV-115 177

Table 54 System Test for COV-114 178

Table 55 System Test for COV-121 178

Table 56 System Test for COV-122 179

Table 57 System Test for COV-108 179

Table 58 System Test for COV-171 180

Table 59 System Test for COV-119 180

Table 60 System Test for COV-116 181

Table 61 System Test for COV-172 182

Table 62 System Test for COV-169 182

Table 63 System Test for COV-126 183

Table 64 System Test for COV-120 183

Table 65 System Test for COV-222 184

Table 66 System Test for COV-223 184

Table 67 System Test for COV-125 185

Table 68: Test Coverage for Each Sprint 189

LIST OF FIGURES

Figure 1 Project Roadmap 21

Figure 2 Sprint 4 Burndown Chart 22

Figure 3: Sprint 4 Retrospective Report 23

Figure 4: System Context Diagram of CovidTracker 27

Figure 5 Use Case Diagram of User 28

Figure 6 Use Case Diagram of Administrator 29

Figure 7 Use Case Diagram of Patient 30

Figure 8 Use Case Diagram of Doctor 31

Figure 9 Use Case Diagram of Health Official 32

Figure 10 Use Case Diagram of Immigration Officer 33

Figure 11 Component Diagram of CovidTracker 34

Figure 12 Architecture Component Diagram of CovidTracker 35

Figure 13 UML Domain Model Class Diagram of CovidTracker 37

Figure 14 Activity Diagram of Define Status Report 38

Figure 15 Activity Diagram of View Status Report 39

Figure 16 Sequence Diagram of Submit Status Report 40

Figure 17 Activity Diagram of Add Test Result 41

Figure 18 Sequence Diagram of View Test Results 42

Figure 19 Sequence Diagram of View Patient List 42

Figure 20 Sequence Diagram of Prioritize Patient 43

Figure 21 Sequence Diagram of Send Message 44

Figure 22 Sequence Diagram of Send Prioritized Message 45

Figure 23 Sequence Diagram of View Messages 46

Figure 24 Sequence Diagram of View Dashboard 47

Figure 25 Activity Diagram of Contact Trace Patient 48

Figure 26 Sequence Diagram of Add Location Report 49

Figure 27 Sequence Diagram of Book Appointment 50

Figure 28 Sequence Diagram of View Appointments 51

Figure 29: Deployment Diagram of CovidTracker 52

Figure 30 Patient Persona 60

Figure 31 Doctor Persona 61

Figure 32 Health Official Persona 61

Figure 33 Immigration Officer Persona 62

Figure 34 Administrator Persona 62

Figure 35 Safari Web Browser Interface Elements 63

Figure 36 Google Chrome Web Browser Interface Elements 64

Figure 37 Apple iPhone 8 Buttons and Safari Web Browser Interface Elements 65

Figure 38 Apple iPhone 11 Buttons and Safari Web Browser Interface Elements 66

Figure 39: Apple iPad Pro Buttons and Safari Web Browser Interface Elements 67

Figure 40 Sign Up Desktop & Tablet UI Mockup 69

Figure 41 Sign Up Mobile UI Mockup 70

Figure 42 Sign In Desktop & Tablet UI Mockup 71

Figure 43 Sign In Mobile UI Mockup 72

Figure 44 Sign Out Desktop & Mobile UI Mockup 73

Figure 45 Sign Out Mobile UI Mockup 73

Figure 46 Add a Role Desktop & Tablet UI Mockup 75

Figure 47 Add a Role Mobile UI Mockup 75

Figure 48 Assign Patient to Doctor Desktop & Tablet UI Mockup 76

Figure 49 Assign Patient to Doctor Mobile UI Mockup 77

Figure 50 Define Status Report Desktop & Tablet UI Mockup 79

Figure 51 Define Status Report Mobile UI Mockup 79

Figure 52 Status Report Desktop & Tablet UI Mockup 80

Figure 53 Status Report Mobile UI Mockup 81

Figure 54 Number of Patients Assigned to a Doctor Desktop & Tablet UI Mockup 82

Figure 55 Number of Patients Assigned to a Doctor Mobile UI Mockup 83

Figure 56 Patient List (Doctor & Health Official) Desktop & Tablet UI Mockup 85

Figure 57 Patient List (Doctor & Health Official) Mobile UI Mockup 86

Figure 58 Patient List (Immigration Officer) Desktop & Tablet UI Mockup 87

Figure 59 Patient List (Immigration Officer) Mobile UI Mockup 88

Figure 60 Status Reports (Patient) Desktop & Tablet UI Mockup 89

Figure 61 Status Reports (Patient) Mobile UI Mockup 90

Figure 62 Status Reports (Doctor/Health Official) Desktop & Tablet UI Mockup 91

Figure 63 Status Reports (Doctor/Health Official) Mobile UI Mockup 91

Figure 64 Add Test Result Desktop & Tablet UI Mockup 92

Figure 65 Add Test Result Mobile UI Mockup 93

Figure 66 Status Report Inbox Desktop & Tablet UI Mockup 94

Figure 67 Status Report Inbox Mobile UI Mockup 95

Figure 68 Status Report Details (Patient) Desktop & Tablet UI Mockup 97

Figure 69 Status Report Details (Patient) Mobile UI Mockup 97

Figure 70 Status Report Details (Doctor/Health Official) Desktop & Tablet UI Mockup 98

Figure 71 Status Report Details (Doctor/Health Official) Mobile UI Mockup 99

Figure 72 Test Results (Patient) Desktop & Tablet UI Mockup 101

Figure 73 Test Results (Patient) Mobile UI Mockup 102

Figure 74 Test Results (Doctor/Health Official) Desktop & Tablet UI Mockup 103

Figure 75 Test Results (Doctor/Health Official) Desktop & Tablet UI Mockup 104

Figure 76 Test Results (Immigration Officer) Desktop & Tablet UI Mockup 105

Figure 77 Test Results (Immigration Officer) Mobile UI Mockup 106

Figure 78 Test Result Details (Patient) Desktop & Tablet UI Mockup 108

Figure 79 Test Result Details (Patient) Mobile UI Mockup 109

Figure 80 Test Result Details (Doctor/Health Official) Desktop & Tablet UI Mockup 110

Figure 81 Test Result Details (Doctor/Health Official) Mobile UI Mockup 111

Figure 82 Test Result Details (Immigration Officer) Desktop & Tablet UI Mockup 112

Figure 83 Test Result Details (Immigration Officer) Mobile UI Mockup 113

Figure 84 Chat (Doctor) Desktop & Tablet UI Mockup 115

Figure 85 Chat (Doctor) Mobile UI Mockup 116

Figure 85 Chat (Patient) Desktop & Tablet UI Mockup 117

Figure 86 Chat (Patient) Mobile UI Mockup 118

Figure 87 Book Appointment Desktop & Tablet UI Mockup 119

Figure 88 Book Appointment Mobile UI Mockup 120

Figure 89 Appointments (Doctor) Desktop & Tablet UI Mockup 122

Figure 90 Appointments (Doctor) Mobile UI Mockup 123

Figure 91 Appointments (Patient) Desktop & Tablet UI Mockup 124

Figure 92 Appointments (Patient) Mobile UI Mockup 125

Figure 93 Add Location Desktop & Tablet UI Mockup 127

Figure 94 Add Location Desktop & Tablet UI Mockup 128

Figure 95 Contact Tracing Desktop & Tablet UI Mockup 129

Figure 96 Contact Tracing Mobile UI Mockup 130

Figure 97 Contact Tracing Desktop & Tablet UI Mockup 132

Figure 98 Contact Tracing Mobile UI Mockup 132
Figure 99 Dashboard (User) Desktop & Tablet UI Mockup 134
Figure 100 Dashboard (User) Mobile UI Mockup 135
Figure 101 Dashboard (Administrator) Desktop & Tablet UI Mockup 136
Figure 102 Dashboard (Administrator) Mobile UI Mockup 137
Figure 103 Dashboard (Doctor) Desktop & Tablet UI Mockup 138
Figure 104 Dashboard (Doctor) Mobile UI Mockup 139
Figure 105 Dashboard (Health Official) Desktop & Tablet UI Mockup 140
Figure 106 Dashboard (Health Official) Mobile UI Mockup 141
Figure 107 Dashboard (Patient) Desktop & Tablet UI Mockup 142
Figure 108 Dashboard (Patient) Mobile UI Mockup 143
Figure 109 Dashboard (Immigration Officer) Desktop & Tablet UI Mockup 144
Figure 110 Dashboard (Immigration Officer) Mobile UI Mockup 145
Figure 111 Code Coverage Report of Server Side Code 186

1.0 INTRODUCTION

The purpose of this document is to give an overview of the problem, proposed solution, statement of scope, project description, product requirements, summary of work packages completed in sprint 4, a sprint 5 release plan, software architecture, risk assessment and management plan, user interface design, testing plan, defect tracking, and quality measurements. This document is targeted at all stakeholders of the system: product owners and development team..

1.1 Positioning

1.1.1 Problem Statement

The problem of	Lack of an easy to use tool to track, manage and coordinate the onset of positive COVID-19 patients on both a micro and macro level.
Affects	Patients, Doctors, Health Officials, Immigration Officers
The impact of which is	The inability for the government to properly handle and manage COVID-19 variants, hospital capacity and perform a safe reopening plan backed by data and science.
A successful solution would be	- A way to assign quarantine restrictions to positive COVID-19 patients - A way to monitor the status and symptoms of confirmed and unconfirmed patients with COVID-19 - Conduct contact tracing notify the people with whom COVID-19 patients have been in contact - Allow patients to update their COVID-19 status and symptoms - An easy way to arrange appointments between doctors and patients

Table 1: Problem Statement

1.1.2 Product Position Statement

For	Patients, Doctors, Health Officials, Immigration Officers and Administrators
Who	Manage, monitor and respond to COVID-19 related events and situations
CovidTracker	Is a responsive web application
Unlike	Covid Alert, ArriveCan
Our product	Is designed to ease the management and monitoring of COVID-19 across the province by contact tracing and notifying patients positive with COVID-19, allowing doctors to follow patients symptoms and arrange appointments with positive patients, assign quarantine restrictions and allowing patients to daily update their status and symptoms.

Table 2: Product Position Statement

2.0 PROJECT DESCRIPTION

The project has a couple of main lenses where all features are derived from such as patient management, status report management, contact tracing, a messaging system, a notification system, a QR code system, and a detailed authentication and authorization layer in front of the system to make sure sensitive info is not shown to the wrong user.

Agile is an iterative software development methodology allowing software teams to produce working software quickly, test it, get feedback on it, and then iterate in quick cycles. Agile is being used given its methodology and to ensure that the product owner's needs -progress and requirements - are being satisfied by the development team throughout the project development lifecycle.

Development is broken down into 5 total sprints and the schedule will be as follows:

Sprint	Date (mm/dd/yyyy)
1	1/12/2022 - 2/2/2022
2	2/3/2022 - 2/23/2022
3	2/24/2022 - 3/16/2022
4	3/17/2022 - 4/6/2022
5	4/7/2022 - 4/18/2022

Table 3: Sprint Schedule

2.1 Stakeholders

Project stakeholders consist of users, the development team and project owners. The various roles assigned to each user, development team member and project owners are as follows and subsequently described in the following section.

- Users
    - Patients
    - Doctors
    - Health Officials
    - Immigration Officers
    - Administrators
- Product Owner
    - Yann-Gaël Guéhéneuc
    - Minani Jean Baptiste
- Project Champion
    - Jason Gerard
- Organizational Management Team
    - Andre Ibrahim
    - Ejazali Rezayi
    - Dan Raiu
    - Daren Kafafian
    - Domenic Seccareccia
    - Jason Gerard
    - Khagik Chris Astor
    - Lucas Blanchard
    - Rafi Stepanians
- Analysts
    - Andre Ibrahim
    - Ejazali Rezayi
    - Dan Raiu
    - Daren Kafafian
    - Domenic Seccareccia
    - Jason Gerard
    - Khagik Chris Astor
    - Lucas Blanchard
    - Rafi Stepanians
- Designers
    - Domenic Seccareccia
- Developers (Front end & Back end)

    - Domenic Seccareccia
    - Dan Raiu
    - Daren Kafafian
    - Ejazali Rezayi
    - Khagik Chris Astor
    - Rafi Stepanians
    - Lucas Blanchard
    - Andre Ibrahim
    - Jason Gerard
- Testers
    - Andre Ibrahim
    - Ejazali Rezayi
    - Dan Raiu
    - Daren Kafafian
    - Domenic Seccareccia
    - Jason Gerard
    - Khagik Chris Astor
    - Lucas Blanchard
    - Rafi Stepanians

2.1.1 Stakeholder Roles

2.1.1.1 Users

Users refer to anyone that uses the software for the functionality that it provides. They have an interest in this project since they use it to accomplish some of their tasks. Users consist of Patients, Doctors, Health Officials, Immigration Officers and Administrators.

2.1.1.2 Product Owner

The product owner is accountable for maximizing the value of the software being developed. His interest is in the delivery of the project in a timely manner with all requirements completed.

2.1.1.3 Project Champion

The project champion is the main driving force of the project fielding all external inquiries and responses. As such, they have one of the largest and most direct stakes in the project.

2.1.1.4 Organization Management Team

The organizational management team organizes and plans the activities that achieve the company's established goals. They will do this by allocating time to build the project schedule from start to finish, allocate resources, and plan meetings to reach pre-established deadlines. They also have an interest in the project both financially and personally as they are also university students.

2.1.1.5 Development Team

The development team has the same interests as the organizational management team. However, their impact, contribution and stake in the project are different. The development team is primarily focused on executing the activities that result in the system being realized and in turn satisfying established goals.

2.1.1.6 Analysts

Analysts are responsible for accessing and researching market opportunities and gathering requirements. They translate requirements to specifications allowing designers to design a system around those needs, developers to satisfy those needs and testers to ensure all developed features work according to specifications. Their stake in the project revolves around how the data impacts the user.

2.1.1.7 Designers

The designers are responsible for the system design aspects such as architectural design, user interface (UI) design and user experience (UX) design. Their

stake in the project revolves around system accessibility, maintainability, upgradability and usability.

2.1.1.8 Developers

The developers must develop a system that satisfies the specifications outlined by the analysts and follows the designs - architectural and interface - specified by the designers. Their stake revolves around the implementation of the system features.

2.1.1.9 Testers

The testers are responsible for quality assurance (QA) during the development and deployment phases. They ensure each developed feature within a sprint passes all associated tests and satisfies the specifications outlined by the analysts. Their stake in the project revolves around user and system QA and quality control (QC).

3.0 REQUIREMENTS

The following requirements elicited from the product owners were turned into user stories and subsequently approved by the product owner. Each user story is associated with a corresponding EPIC, has a list of subtasks and a definition of done defining what must be completed in order for said user story to be considered as complete. Furthermore, the description section of each user story is broken down with the following information:

- Definition of done
- Requirements
    - UI Prototype
    - Front end
    - API (Optional)
    - Specifications (Google document attached containing all specifications associated to the user story)
    - Personas (accessible by)
- Acceptance tests
- System tests

Once the user interface design mockups are complete, a prototype is created allowing assigned developers to interact with the designs to get a better sense of how the feature is expected to be interacted by users. Associated links and attachments are then added in the comments section of the associated user story.

3.1 User Stories

For Sprint 4, **10 user stories, 6 tasks, and 0 bugs** have been elicited for a total of **83 user story points**.

See next page to view user stories for sprint 4.

T	Key	Summary	Story point estimate	P	Risk	Parent	Start date	Due	Status	Assignee(s)
	COV-223	As an Immigration Officer, I want to view the patients list, so I can be aware of test results and prioritize them	1		High	Status Management	26/Mar/22	27/Mar/22	DONE	Jason Gerard
	COV-222	As a Doctor, I want to define status report fields from the patient list page	2		Low	Status Management	26/Mar/22	27/Mar/22	DONE	Jason Gerard
	COV-172	As a Patient, I want to add the locations of where I have been during the day, so that I can be contact traced if I come in contact with someone that has tested positive with COVID-19	8		Medium	Contact Tracing	27/Mar/22	03/Apr/22	DONE	Jason Gerard
	COV-171	As a Health Official, I want to view a list of all patients who have tested positive in the last [x] days, so that I can contract trace them	5		Medium	Contact Tracing	27/Mar/22	03/Apr/22	DONE	Andre Ibrahim, Jason Gerard
	COV-169	As a Doctor, I want to view my appointments, so that I can schedule myself	5		Low	Appointments	20/Mar/22	26/Mar/22	DONE	Jason Gerard
	COV-126	As a Health Official, I want to contact trace who a Patient has been in contact with in the last [x] days, so that I can manage who is at risk	5		High	Contact Tracing	27/Mar/22	03/Apr/22	DONE	Andre Ibrahim, Jason Gerard
	COV-125	As a User, I want to a dashboard personalized for my role, so I can have an overview of the system	13		Low	Status Management	29/Mar/22	01/Apr/22	DONE	Jason Gerard
	COV-120	As a Patient, I want to mark my message with a priority level, so that my Doctor will view it quicker	5		Medium	Messaging System	23/Mar/22	27/Mar/22	DONE	Jason Gerard
	COV-119	As a Patient, I want to direct message my Doctor, so that I can ask them questions	8		High	Messaging System	19/Mar/22	23/Mar/22	DONE	Jason Gerard
	COV-116	As a Doctor, I want to book an appointment with a Patient, so that we can discuss their symptoms	8		Medium	Appointments	17/Mar/22	20/Mar/22	DONE	Jason Gerard
	COV-183	Create UI prototype for doctor "view graphs of status fields" widgets	1		Low	Status Management	24/Feb/22	02/Apr/22	DONE	Domenic Seccareccia
	COV-132	As a Doctor, I want to send a notification to a Patient, so that they are reminded of an upcoming appointment	2		Low	Appointments	27/Mar/22	02/Apr/22	DONE	Jason Gerard
	COV-131	As a Doctor, I want to be notified when a Patient updates their status more than once in a day, so that I can look into their status	2		Low	Status Management	27/Mar/22	02/Apr/22	DONE	Jason Gerard
	COV-130	Create generic SMS notification infrastructure	5		High	Notification System	26/Mar/22	27/Mar/22	DONE	Jason Gerard
	COV-129	Create generic email notification infrastructure	5		High	Notification System	26/Mar/22	27/Mar/22	DONE	Jason Gerard
	COV-117	Create generic direct messaging infrastructure	8		High	Messaging System	17/Mar/22	19/Mar/22	DONE	Jason Gerard

3.2 Backlog

The following figure is the view of the project roadmap at the end of sprint 4. See the next page to view the project backlog.

Figure 1: Project Roadmap

T	Key	Summary	Story point estimate	P	Risk	Parent	Start date	Due	Status	Assignee(s)
	COV-223	As an Immigration Officer, I want to view the patients list, so I can be aware of test results and prioritize them	1		High	Status Management	26/Mar/22	27/Mar/22	DONE	Jason Gerard
	COV-222	As a Doctor, I want to define status report fields from the patient list page	2		Low	Status Management	26/Mar/22	27/Mar/22	DONE	Jason Gerard
	COV-172	As a Patient, I want to add the locations of where I have been during the day, so that I can be contact traced if I come in contact with someone that has tested positive with COVID-19	8		Medium	Contact Tracing	27/Mar/22	03/Apr/22	DONE	Jason Gerard
	COV-171	As a Health Official, I want to view a list of all patients who have tested positive in the last [x] days, so that I can contract trace them	5		Medium	Contact Tracing	27/Mar/22	03/Apr/22	DONE	Andre Ibrahim, Jason Gerard
	COV-169	As a Doctor, I want to view my appointments, so that I can schedule myself	5		Low	Appointments	20/Mar/22	26/Mar/22	DONE	Jason Gerard
	COV-157	As a Doctor, I want to view a list of my Patients, so that I can easily navigate to their specific detailed views	8		Medium	Patient Management	02/Mar/22	05/Mar/22	DONE	Jason Gerard
	COV-126	As a Health Official, I want to contact trace who a Patient has been in contact with in the last [x] days, so that I can manage who is at risk	5		High	Contact Tracing	27/Mar/22	03/Apr/22	DONE	Andre Ibrahim, Jason Gerard
	COV-125	As a User, I want to a dashboard personalized for my role, so I can have an overview of the system	13		Low	Status Management	29/Mar/22	01/Apr/22	DONE	Jason Gerard
	COV-124	As a Patient, I want to view the details of a single COVID test result, so that I'm aware of my diagnosis	3		High	Patient Management	24/Feb/22	27/Feb/22	DONE	Andre Ibrahim, Jason Gerard
	COV-123	As a Patient, I want to view all my line item COVID test results, so that I'm aware of my diagnosis	5		Medium	Patient Management	27/Feb/22	02/Mar/22	DONE	Andre Ibrahim, Khachig Astor
	COV-122	As a Patient, I want to be able to generate a QR code for a lab test result, so that I can share it with others	2		High	QR Code System	27/Feb/22	03/Mar/22	DONE	Jason Gerard
	COV-121	As a Patient, I want to be able to generate a QR code for a status report, so that I can share it with others	2		High	QR Code System	27/Feb/22	03/Mar/22	DONE	Jason Gerard
	COV-120	As a Patient, I want to mark my message with a priority level, so that my Doctor will view it quicker	5		Medium	Messaging System	23/Mar/22	27/Mar/22	DONE	Jason Gerard
	COV-119	As a Patient, I want to direct message my Doctor, so that I can ask them questions	8		High	Messaging System	19/Mar/22	23/Mar/22	DONE	Jason Gerard
	COV-116	As a Doctor, I want to book an appointment with a Patient, so that we can discuss their symptoms	8		Medium	Appointments	17/Mar/22	20/Mar/22	DONE	Jason Gerard
	COV-115	As a Doctor, I want to mark a Patient's status update as "Reviewed", so that I can see which statuses I've already seen	8		High	Status Management	06/Mar/22	09/Mar/22	DONE	Jason Gerard
	COV-114	As a Doctor, I want to flag certain patients, so that their updates are prioritized over others	5		High	Patient Management	06/Mar/22	09/Mar/22	DONE	Jason Gerard
	COV-113	As a Doctor, I want to view a line item list of my patients with their most recent line item status update, so that I can keep track of any updates	13		Medium	Status Management	02/Mar/22	06/Mar/22	DONE	Jason Gerard
	COV-112	As a Patient, I want to view the details of a single status report of a Patient, so that I can view their progress at a point in time	3		Medium	Status Management	24/Feb/22	27/Feb/22	DONE	Jason Gerard
	COV-111	As a Patient, I want to view all my line item statuses, so that I can monitor my progress over time	5		Medium	Status Management	27/Feb/22	02/Mar/22	DONE	Jason Gerard, rafistep98
	COV-108	As a Patient, I want to update my status for the day after already submitting, so that my Doctor stays up to date	3		Low	Status Management	10/Mar/22	12/Mar/22	DONE	Jason Gerard
	COV-107	As a Health Official, I want to input COVID test results, so that I can report if a Patient tested positive or negative	8		High	Patient Management	24/Feb/22	26/Feb/22	DONE	Andre Ibrahim

T	Key	Summary	Story point estimate	P	Risk	Parent	Start date	Due	Status	Assignee(s)
	COV-95	As a Doctor, I want to define the status report fields for my Patients, so I can properly track them	8	=	High	Status Management	10/Feb/22	18/Feb/22	DONE	Jason Gerard, Khachig Astor
	COV-90	As an Administrator, I want the API to authorize users by role, so that access rights are managed	8	^^	High	Authentication and Authorization	04/Feb/22	08/Feb/22	DONE	Andre Ibrahim, Jason Gerard
	COV-85	As an Administrator, I want to assign a role to a user, so that I can manage access rights	13	^^	Medium	Authentication and Authorization	06/Feb/22	17/Feb/22	DONE	Jason Gerard, rafistep98
	COV-52	As a User, I want to be able to sign out, so that I can delete my session	5	=	High	Authentication and Authorization	28/Jan/22	30/Jan/22	DONE	Domenic Seccareccia, Jason Gerard, Khachig Astor
	COV-48	As a User, I want to be able to sign in, so that I can access my account	8	=	High	Authentication and Authorization	24/Jan/22	30/Jan/22	DONE	Daren, Domenic Seccareccia, Jason Gerard
	COV-42	As a User, I want to be able to sign up, so that I can access the apps features	13	^	High	Authentication and Authorization	17/Jan/22	31/Jan/22	DONE	Andre Ibrahim, Domenic Seccareccia, Ejazali Rezayi, Jason Gerard
	COV-27	As an Administrator, I want to view the number of Patients assigned to a Doctor, so that no Doctor has too many Patients	5	=	Low	Patient Management	13/Feb/22	14/Feb/22	DONE	Jason Gerard
	COV-26	As an Administrator, I want to assign a Patient to a Doctor, so that I can manage the Patients	5	^	Medium	Patient Management	09/Feb/22	20/Feb/22	DONE	Andre Ibrahim, Ejazali Rezayi
	COV-25	As a Patient, I want to submit my status, so that I can keep my Doctor updated	13	=	High	Status Management	14/Feb/22	20/Feb/22	DONE	Andre Ibrahim, Jason Gerard
	COV-213	Setup and configure application deployment	2	^	High	Project Infrastructure	12/Mar/22	13/Mar/22	DONE	Jason Gerard
	COV-212	Create UI prototype for "contact trace result" page	1	^^	Low	Contact Tracing	10/Mar/22	13/Mar/22	DONE	Domenic Seccareccia
	COV-211	Create UI prototype for "contact trace" patient list page	1	^^	Low	Contact Tracing	10/Mar/22	13/Mar/22	DONE	Domenic Seccareccia
	COV-210	Create UI prototype for "location report" page	1	^^	Low	Contact Tracing	10/Mar/22	13/Mar/22	DONE	Khachig Astor
	COV-195	Add more complex and large scale data generation scripts	1	^	High	Project Infrastructure	28/Feb/22	28/Feb/22	DONE	Jason Gerard
	COV-192	Create UI prototype for Doctor and Patient "view appointments" page	1	^^	Low	Appointments	24/Feb/22	07/Mar/22	DONE	Lucas Blanchard
	COV-190	Create UI prototype for Doctor "book an appointment" page	1	^^	Low	Appointments	24/Feb/22	07/Mar/22	DONE	Lucas Blanchard
	COV-188	Create UI prototype for patients "mark message priority level" button	1	^^	Low	Messaging System	07/Mar/22	14/Mar/22	DONE	Domenic Seccareccia
	COV-187	Create UI prototype for doctor and patients "chat" page	1	^^	Low	Messaging System	24/Feb/22	07/Mar/22	DONE	Domenic Seccareccia

T	Key	Summary	Story point estimate	P	Risk	Parent	Start date	Due	Status	Assignee(s)
☑	COV-183	Create UI prototype for doctor "view graphs of status fields" widgets	1	⏫	Low	Status Management	24/Feb/22	02/Apr/22	DONE	Domenic Seccareccia
☑	COV-182	Create UI prototype for doctor "mark patient as prioritized" button	1	⏫	Low	Patient Management	19/Feb/22	21/Feb/22	DONE	Domenic Seccareccia
☑	COV-175	Create UI prototype for doctor "mark patient status as reviewed" page update	1	⏫	Low	Status Management	19/Feb/22	21/Feb/22	DONE	Domenic Seccareccia
☑	COV-174	Create UI prototype for patient "generate QR code of covid test page" button	1	⏫	Low	QR Code System	19/Feb/22	21/Feb/22	DONE	Domenic Seccareccia
☑	COV-173	Create UI prototype for patient "generate QR code of status report page" button	1	⏫	Low	QR Code System	20/Feb/22	21/Feb/22	DONE	Domenic Seccareccia
☑	COV-166	Create UI prototype for a patient "view details of a single status report" page	1	⏫	Low	Status Management	19/Feb/22	21/Feb/22	DONE	Domenic Seccareccia
☑	COV-165	Create UI prototype for a patient "view details of a single test result" page	1	⏫	Low	Patient Management	19/Feb/22	22/Feb/22	DONE	Domenic Seccareccia
☑	COV-160	Create UI prototype for patient "view line item list of status reports for that patient" page	1	⏫	Low	Status Management	19/Feb/22	21/Feb/22	DONE	Domenic Seccareccia
☑	COV-156	Create UI prototype for doctor "view list of patients" page	1	⏫	Low	Patient Management	19/Feb/22	21/Feb/22	DONE	Domenic Seccareccia
☑	COV-155	Create UI prototype for patient "view line item list of covid test results for that patient" page	1	⏫	Low	Patient Management	19/Feb/22	21/Feb/22	DONE	Domenic Seccareccia
☑	COV-154	Create UI prototype for admin "view number of patients a doctor has" page	1	⏫	Low	Patient Management	12/Feb/22	12/Feb/22	DONE	Domenic Seccareccia
☑	COV-153	Create UI prototype for patient "submit covid test results" page	1	⏫	Low	Patient Management	19/Feb/22	22/Feb/22	DONE	Domenic Seccareccia
☑	COV-152	Create UI prototype for doctor "patient most recent line item status report" page	1	⏫	Low	Status Management	11/Feb/22	21/Feb/22	DONE	Domenic Seccareccia
☑	COV-151	Create UI prototype for patient "submit status report" page	1	⏫	Low	Status Management	13/Feb/22	13/Feb/22	DONE	Lucas Blanchard
☑	COV-150	Create UI prototype for doctor "define patient status report fields" page	1	⏫	Low	Status Management	12/Feb/22	13/Feb/22	DONE	Lucas Blanchard
☑	COV-149	Create UI prototype for admin "assign patient to a doctor" page	1	⏫	Low	Patient Management	08/Feb/22	12/Feb/22	DONE	Domenic Seccareccia
☑	COV-148	Create UI prototype for admin "add role" page	1	⏫	Low	Authentication and Authorization	08/Feb/22	12/Feb/22	DONE	Domenic Seccareccia
☑	COV-147	Setup integration tests	1	🔽	Low	Project Infrastructure	20/Feb/22	22/Feb/22	DONE	Jason Gerard
☑	COV-146	Setup a Dockerfile for the front end and integrate it with docker-compose	2	⏬	Low	Project Infrastructure	04/Feb/22	06/Feb/22	DONE	Jason Gerard
☑	COV-132	As a Doctor, I want to send a notification to a Patient, so that they are reminded of an upcoming appointment	2	🔽	Low	Appointments	27/Mar/22	02/Apr/22	DONE	Jason Gerard
☑	COV-131	As a Doctor, I want to be notified when a Patient updates their status more than once in a day, so that I can look into their status	2	🔽	Low	Status Management	27/Mar/22	02/Apr/22	DONE	Jason Gerard
☑	COV-130	Create generic SMS notification infrastructure	5	⏫	High	Notification System	26/Mar/22	27/Mar/22	DONE	Jason Gerard
☑	COV-129	Create generic email notification infrastructure	5	⏫	High	Notification System	26/Mar/22	27/Mar/22	DONE	Jason Gerard
☑	COV-117	Create generic direct messaging infrastructure	8	⏫	High	Messaging System	17/Mar/22	19/Mar/22	DONE	Jason Gerard
☑	COV-84	Create domain model diagram	2	🟰	Low	Documentation	29/Jan/22	30/Jan/22	DONE	Andre Ibrahim, Jason Gerard

T	Key	Summary	Story point estimate	P	Risk	Parent	Start date	Due	Status	Assignee(s)
☑	COV-76	Create architecture component diagram	2	=	Low	Documentation	24/Jan/22	28/Jan/22	DONE	Jason Gerard
☑	COV-38	Create initial software architecture document	3	˅	Low	Documentation	24/Jan/22	31/Jan/22	DONE	Andre Ibrahim
☑	COV-33	Update risk management plan and table	2	˅	Low	Documentation	26/Jan/22	29/Jan/22	DONE	Dan, Khachig Astor
☑	COV-30	Add testing framework to web server	2	=	Low	Project Infrastructure	14/Jan/22	15/Jan/22	DONE	Jason Gerard
☑	COV-7	Setup docker for web server	5	=	Low	Project Infrastructure	17/Jan/22	18/Jan/22	DONE	Domenic Seccareccia, Jason Gerard
☑	COV-6	Configure CI/CD pipeline for web server	2	=	Low	Project Infrastructure	15/Jan/22	17/Jan/22	DONE	Jason Gerard
☑	COV-5	Setup Jira project	1	^	Low	Project Setup	15/Jan/22	18/Jan/22	DONE	Domenic Seccareccia, Jason Gerard
☑	COV-4	Setup basic project for front end client	5	^^	Medium	Project Setup	22/Jan/22	23/Jan/22	DONE	Andre Ibrahim, Dan, Domenic Seccareccia, Khachig Astor, rafistep98
☑	COV-3	Setup basic web server	5	^^	Medium	Project Setup	13/Jan/22	15/Jan/22	DONE	Jason Gerard
☑	COV-1	Setup GitHub repository	1	^^	Medium	Project Setup	13/Jan/22	15/Jan/22	DONE	Jason Gerard

4.0 RELEASE PLANNING

This section covers a summary and retrospective for sprint 4 and sprint 5 planning.

4.1. Sprint 4

4.1.1 Summary

Sprint 4 mainly focused on delivering user stories in the Notification System, Messaging System, Contact Tracing, and Appointments epics. Doctors and patients can directly communicate through the application. Doctors can book appointments with their assigned patients. Health officials can contact trace and notify individuals that they must self-quarantine after coming into contact with a positive patient. Users have access to a personalized dashboard that displays various COVID-19 and system related information. E-mail and SMS notifications have also been integrated in various system user flows.

Project velocity after Sprint 4: 72 User Story Points

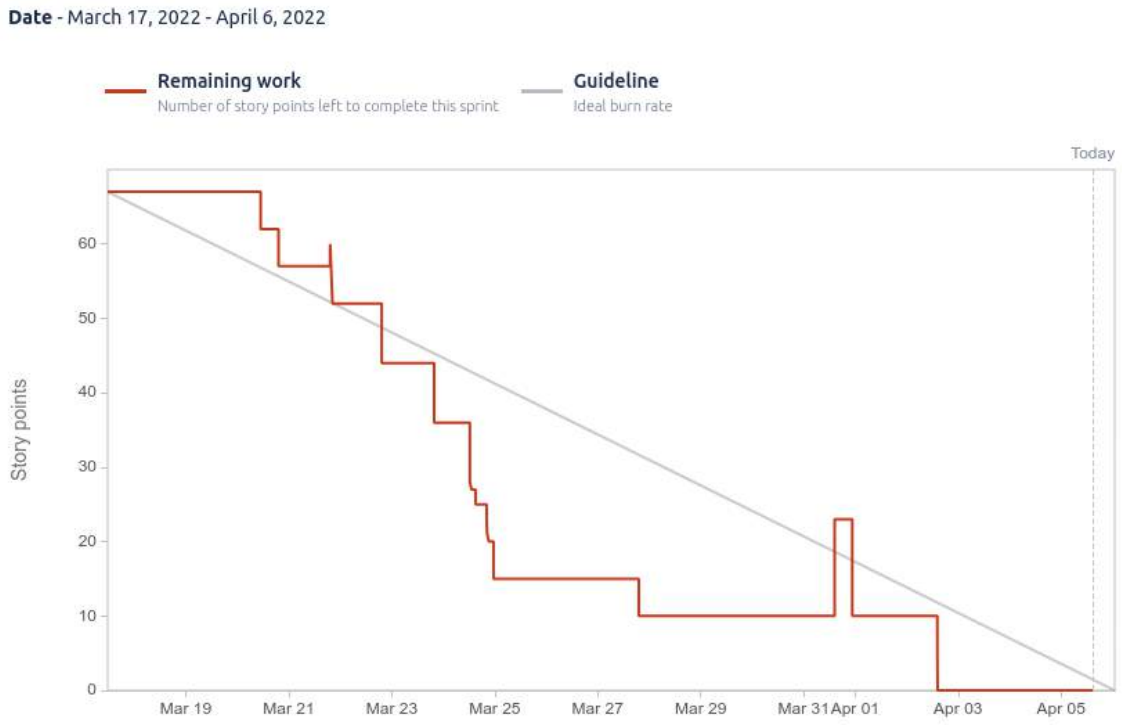


Figure 2: Sprint 4 Burndown Chart

T	Key	Summary	Parent	Description

	COV-223	As an Immigration Officer, I want to view the patients list, so I can be aware of test results and prioritize them	Status Management	**Definition of done:** An Immigration Officer should be able to view a list of patients in the system but only their identifiable information and test results.
				
				Requirements:
				
				Front end:
				• update patient list page to have dynamic menu item visibility depending on the user role
				• Relevant pages
				  ◦ Patient list
				  ◦ Patient test results
				  ◦ Test result details
				
				Back end
				• Update any API endpoint related to the pages to allow access for immigration officers
				• No new endpoints should need to be created
				• The immigration officers will have the same data as the health officials
				
				Specifications:
				
				https://docs.google.com/document/d/1Gz1aIwcxNxBtoL-YoTgScYyQz4NLlRsK9bDcCVIY88I/edit?usp=sharing
				
				Personas (accessible by):
				• Immigration Officer

	COV-222	As a Doctor, I want to define status report fields from the patient list page	Status Management	**Definition of done:** A doctor should be able to define the status report fields for a patient without having to manually input their patient id **Requirements:** **Front end** • integrate the define status report fields page with the patient list • Remove the navbar link for the old page • Update page route to include the patient id so we can prefill it in the API **Back end** • API should be functioning and normal no back end work should be required **Specifications:** https://docs.google.com/document/d/1tvUcHNK6HVxYmF53Q3Z6MDn392bciqL1_rnrMd0CaAE/edit?usp=sharing **Personas (accessible by):** • Doctor

T	Key	Summary	Parent	Description

	COV-172	As a Patient, I want to add the locations of where I have been during the day, so that I can be contact traced if I come in contact with someone that has tested positive with COVID-19	Contact Tracing	**Definition of done:** A patient should be able to add a location report multiple times a day and select the date they were at the location.
				
				Requirements:
				
				Front end:
				• New page with a form to input address
				• On submit
				  ◦ Show toast
				  ◦ Reset form
				
				API:
				• endpoint to add new location report
				  ◦ POST to `/location_report`
				  ◦ Body
				  ▪ address fields
				  ◦ Returns status code 201
				
				Database:
				• new table called `location_reports`
				  ◦ columns
				  ▪ patientId FK to `patients` table
				  ▪ addressId FK to `addresses` table
				
				Specifications
				
				https://docs.google.com/document/d/1lSsk4lrbPKPlLSRoreY4N2zxP28TSOMEPwRneQQrQ_s/edit?usp=sharing
				
				Personas (accessible by):
				• User
				• Patient

	COV-171	As a Health Official, I want to view a list of all patients who have tested positive in the last [x] days, so that I can contract trace them	Contact Tracing	**Definition of done:** A Health Official should be able to view a list of positive patients and filter the list by a date range.
				
				Requirements:
				
				Front end:
				• Page with data table of patients
				• Date range picker
				
				API:
				• endpoint to get patients who's last test was positive in date range
				  ◦ GET to {{/patients?status=postive&testDateFrom=
				Unknown macro: {from}
				
				&testDateTo=
				Unknown macro: {to}
				
				}}
				
				  ▪ Body
				  ▪ None
				  ▪ Returns
				  ▪ status 200
				  ▪ List of patients who's last test was positive in that range
				
				Specifications:
				
				https://docs.google.com/document/d/1HtXigREozTG4axqx8wqH1Z1z3EBVayhsiVE9qYLYLDg/edit?usp=sharing
				
				Personas (accessible by):
				• Health Official

T	Key	Summary	Parent	Description

	COV-169	As a Doctor, I want to view my appointments, so that I can schedule myself	Appointments	**Definition of done:** A doctor or patient should be able to view their appointments.
				
				Requirements:
				
				Front end:
				
				• Adds a new page + table with a list of their appointments
				
				Back end:
				
				• Endpoint for a user to get their appointments
				  ◦ GET `/appointments`
				  ◦ Add route verification for “is patient or doctor”
				  ◦ Returns
				  ▪ status 200
				  ▪ List of appointments for that user
				
				Specifications:
				
				https://docs.google.com/document/d/1X70ONaP0-sTX_2TIh5iEoOjcD_81YTvXrnqGSwe0wj0/edit?usp=sharing
				
				Personas (accessible by):
				
				• Doctor
				• Patient

	COV-126	As a Health Official, I want to contact trace who a Patient has been in contact with in the last [x] days, so that I can manage who is at risk	Contact Tracing	**Definition of done:** A Health Official should be able to select a given patient and contact trace them to view all the patients they have been in contact with over a given date range.
				
				Requirements:
				
				Front end:
				
				• Table containing patients who have been “contact traced” in time range specified
				• Icon by each row to notify user
				  ◦ Should call notification endpoint
				
				API:
				
				• endpoint to get list of patients who have been at the same location on the same calendar day as the target patient within the specified time range
				  ◦ GET to {{/patients?traceTarget=
				  Unknown macro: {patientId}
				
				  &testDateFrom=
				  Unknown macro: {from}
				
				  &testDateTo=
				  Unknown macro: {to}
				
				  }}
				
				Specifications:
				
				https://docs.google.com/document/d/13BqHRJvUiirZGn8sqP2ZCnbrozDp7rPpKgufjNEc26A/edit?usp=sharing
				
				Personas:
				
				• Health Official

	COV-125	As a User, I want to a dashboard personalized for my role, so I can have an overview of the system	Status Management	**Definition of done:** All users should be able to view graphs of data points from the list of most recent statuses and time series graphs for a single patient. **Requirements:** **Front end:** **Specifications:** https://docs.google.com/document/d/1k7mehy-0PiTf1Auaq4lAyteuvj1hDZD2GBbgrLWD2c4/edit?usp=sharing **Personas (accessible by):** • Administrator • Patient • Doctor • Health Official • User • Immigration Officer

T	Key	Summary	Parent	Description
	COV-120	As a Patient, I want to mark my message with a priority level, so that my Doctor will view it quicker	Messaging System	**Definition of done:** A patient should be able to mark their message as “priority” before sending so that doctor knows its important. **Requirements:** **Front end:** • This is augmenting the chat page send bar for the patient only. **Back end:** • Update POST messages endpoint to support a isPriority boolean flag **Database** • new column in the `messages` table for `is_priority BOOLEAN NOT NULL` **Specifications:** https://docs.google.com/document/d/1voWDcepvfrUwEOI8nGmXi7_eMVITCqjE7uvvIOIc7nA/edit?usp=sharing **Personas (accessible by):** • Doctor • Patient
	COV-119	As a Patient, I want to direct message my Doctor, so that I can ask them questions	Messaging System	**Definition of done:** A patient should be able to message their doctor through the website to ask them questions and the doctor should be able to respond. **Requirements:** **Front end:** • Should implement the chat page • Patient should see 1 chat available (doctor) • Doctor should see many chats available (all their patients) **Specifications:** https://docs.google.com/document/d/1-eOp7mbc4g_DPHw_aCJXns9vgfiEH8UbNSiT8Wksc9A/edit?usp=sharing **Personas (accessible by):** • Patient • Doctor

T	Key	Summary	Parent	Description

	COV-116	As a Doctor, I want to book an appointment with a Patient, so that we can discuss their symptoms	Appointments	**Definition of done:** A doctor should be able to book an appointment with a patient so that they can discuss their symptoms more in depth.
				
				Requirements:
				
				Front end:
				• Adds a new form for the doctor to book an appointment with one of their patients
				
				Back end:
				• Endpoint for a doctor to book an appointment
				  ◦ POST `/appointments`
				  ◦ Body
				  ▪ patientId
				  ▪ date
				  ▪ start
				  ▪ end
				  ▪ address
				  ◦ Add route verification for “is patient of doctor”
				  ◦ Returns status 201
				
				Database
				• new table `appointments` with columns
				  ◦ doctor_id
				  ◦ patient_id
				  ◦ start_date
				  ◦ end_date
				  ◦ address_id
				
				Specifications
				
				https://docs.google.com/document/d/1XSpuQnlHnWnvODFoQV5cM6PysjhK6EvAZfMwrX7skF4/edit?usp=sharing
				
				Personas (accessible by):
				• Doctor

	COV-183	Create UI prototype for doctor "view graphs of status fields" widgets	Status Management	**UI Prototype:**
				• Users/Patients/Immigration Officer/Health Official/Doctor/Admin
				  ◦ # of total covid cases (in users province): Stats card
				  ◦ # of confirmed cases per day (in users province): Graph
				  ◦ # of total confirmed cases by age (in users province): Pie chart
				  ◦ # of current cases (latest test result = positive) (in users province): Graph
				• Health Official
				  ◦ # of status reports in last 24 hours: Stats card
				  ◦ Symptom in last 24 hours (primary symptoms: Bar chart/pie chart
				  ◦ # of total assigned patients across all doctors: Stats card
				• Admin
				  ◦ Patients per doctor: Stats card
				  ◦ # of total assigned patients across all doctors: Stats card
				• Patient
				  ◦ # of appointments today: Stats card
				• Doctor
				  ◦ # of unread status reports: Stats card
				  ◦ Symptoms in last 24 hours (primary + secondary symptoms): Bar chart/pie chart
				  ◦ # of new patients assigned for the day: Stats card
				  ◦ # of total assigned patients: Stats card
				  ◦ # of appointments today: Stats card

	COV-132	As a Doctor, I want to send a notification to a Patient, so that they are reminded of an upcoming appointment	Appointments	**Requirements:** **Back end:** • When an appointment is successfully saved in the database the notification service should be called to send a notification to the patient • The subject of the email should contain the patients name, doctors name, mention the appointment • The body should contain all other relevant info about the appointment

T	Key	Summary	Parent	Description
☑	COV-131	As a Doctor, I want to be notified when a Patient updates their status more than once in a day, so that I can look into their status	Status Management	**Requirements:** **Back end:** • When a patient makes more than one status report in a single calendar day a notification should be sent to the doctor • The subject of the notification should contain the patients name and mention the status update • The body should contain all other relevant info about the patient and status update

☑	COV-130	Create generic SMS notification infrastructure	Notification System	**Requirements:**
				
				API:
				
				• Endpoint to send an sms notification to a user
				  ◦ This will mainly be used through the service interface by other services not by the front end
				  ◦ POST `/notifications/sms`
				  ◦ Body
				  ▪ toUserId: number → userid notification is being sent to
				  ▪ smsBody: string
				  ◦ Returns status code 201

☑	COV-129	Create generic email notification infrastructure	Notification System	**Requirements:**
				
				API:
				
				• Endpoint to send an email notification to a user
				  ◦ This will mainly be used through the service interface by other services not by the front end
				  ◦ POST `/notifications/email`
				  ◦ Body
				  ▪ toUserId: number → userid notification is being sent to
				  ▪ emailSubject: string
				  ▪ emailBody: string
				  ◦ Returns status code 201

☑	COV-117	Create generic direct messaging infrastructure	Messaging System	**Requirements:**
				
				API:
				
				• Get users messages
				  ◦ GET `/messages`
				  ◦ Route verification for “patients and doctors only”
				  ◦ Get messages associated with userId in jwt token
				  ◦ Returns
				  ▪ Status code 200
				  ▪ List of messages
				• Send a message
				  ◦ POST `/messages`
				  ◦ Body
				  ▪ toUserId: number → userid message is being sent to
				  ▪ messageBody: string
				  ◦ Route verification for “patients and doctors only”
				  ◦ save message in db associated with userid in jwt
				  ◦ Returns status code 201
				
				Database:
				
				• new table called `messages` with columns
				
				  ◦ from_user_id
				  ◦ to_user_id
				  ◦ message_body
				  ◦ created_on

4.1.2 Retrospective

View the report of the sprint 4 retrospective meeting below.

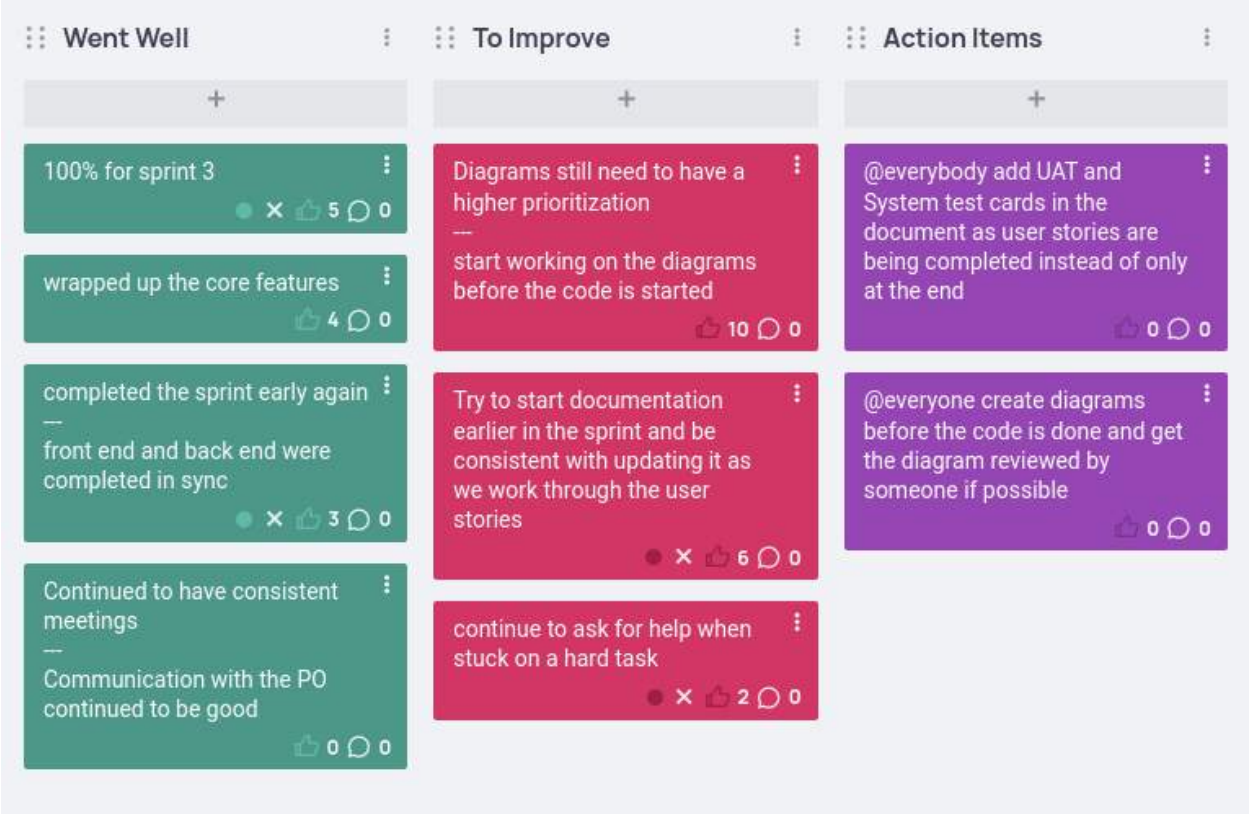


Figure 3: Sprint 4 Retrospective Report

4.2 Sprint 5

4.2.1 Planning

Sprint 5 has no features planned as the team managed to deliver all committed features by the end of sprint 4. The dashboard feature which was originally planned to be implemented in sprint 5 was completed in sprint 4. As a result, sprint 5 will focus on testing, bug fixing, and any improvements as the team and Project Owner see fit. Given the uncertainty of tasks, no user story point commitment will be made beforehand. Because of this there is no generated line item or detailed report for the Sprint 5 user stories and tasks.

5.0 SOFTWARE ARCHITECTURE

This section provides an overview of the system to be built using both a domain model and a component diagram depicting and describing the chosen design decisions of the system.

Date Issued	January 11, 2022
Status	Sprint completed
Authors	Jason Gerard, Andre Ibrahim, Domenic Seccareccia
Reviewers	Domenic Seccareccia, Jason Gerard
Scope	The domain model covers the domain of the application, the component diagram covers the entire system in development, the use case diagrams cover the various activities each user can accomplish.
Context	This is the third sprint for the web application “CovidTracker”. Diagrams will be expanded and improved over each sprint iteration.

Table 4: Supplementary Information

5.1 Stakeholder Concerns

Stakeholder concerns associated with CovidTracker are depicted in the following Stakeholder Concern Traceability Matrix. Only stakeholders that have a concern impacted by the systems architecture are present in this table.

	Developer	Project Champion	Testers	Product Owner	User
System failure					
Security breach					
Unscalable architecture					
Tightly coupled layers					
System complexity					
Longer development time					

Table 5: Stakeholder Concern Traceability Matrix

5.2 Viewpoints

The following 5 viewpoints have been identified to describe CovidTracker:

- Context
- Functional
- Information
- Development
- Deployment

Each section below explains what each viewpoint covers and how it relates to the system as well as the stakeholders that should be concerned with each viewpoint. Diagrams are used to depict how the system looks from each viewpoint.

5.2.1 Context Viewpoint

The context viewpoint describes the relationships, dependencies, and interactions between the system and its environment (people, systems, and external entities with which it interacts). This viewpoint concerns all project stakeholders.

5.2.1.1 System Context Diagram

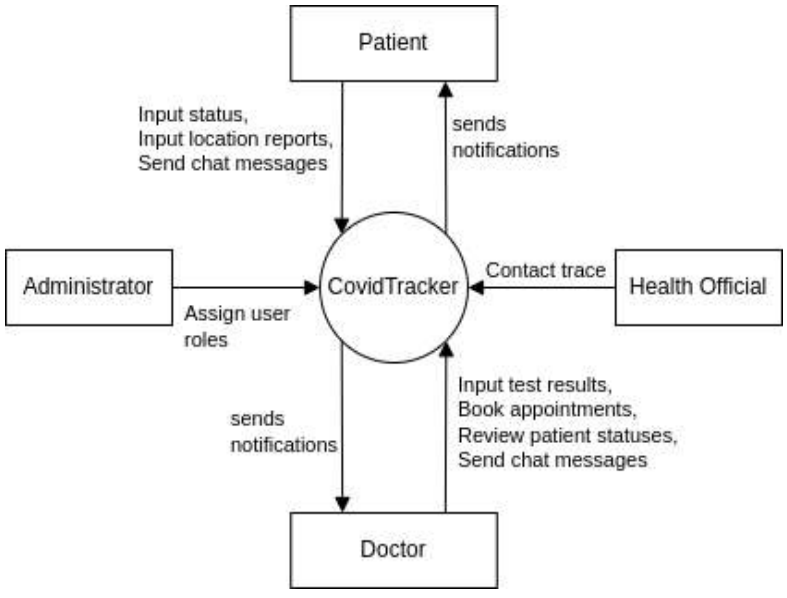


Figure 4: System Context Diagram of CovidTracker

5.2.1.2 Use Case Diagrams

The use case diagrams representing all users that can access CovidTracker and their associated activities are represented in the following sections. The diagrams can be viewed in draw.io through this link:

- Use Case Diagrams of CovidTracker

5.2.1.2.1 User

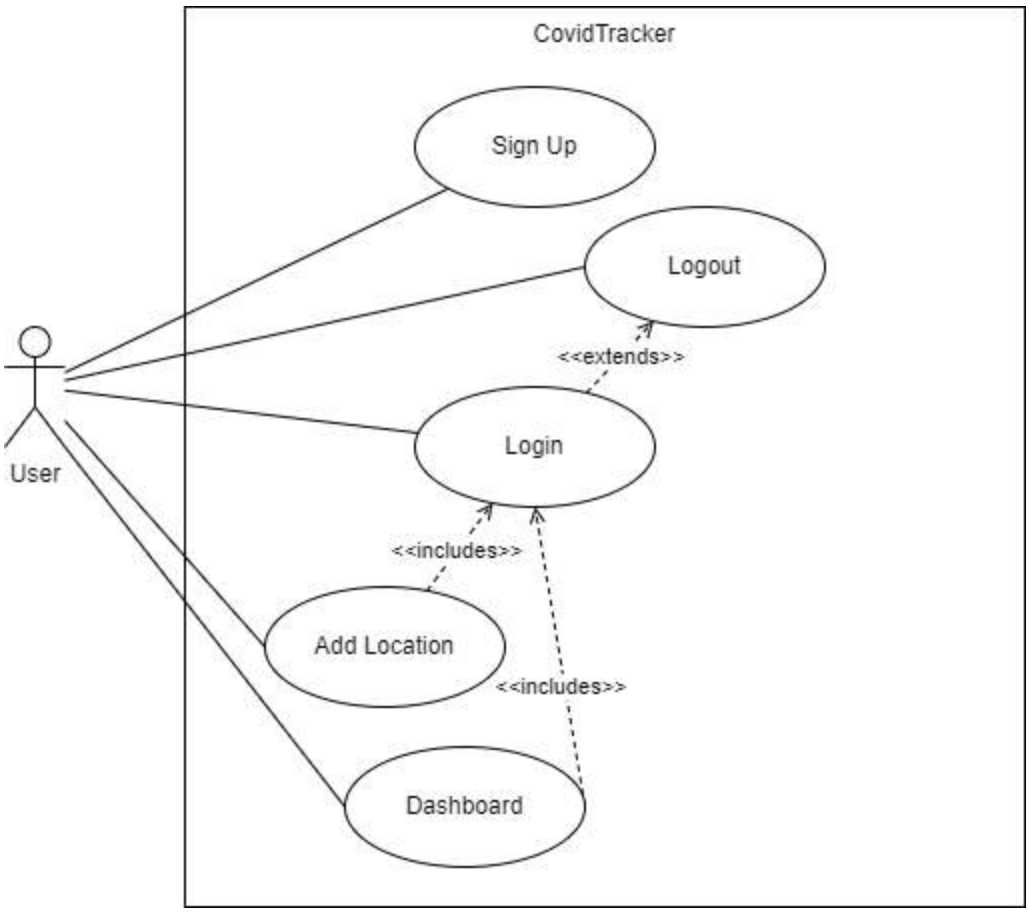


Figure 5: Use Case Diagram of User

5.2.1.2.2 Administrator

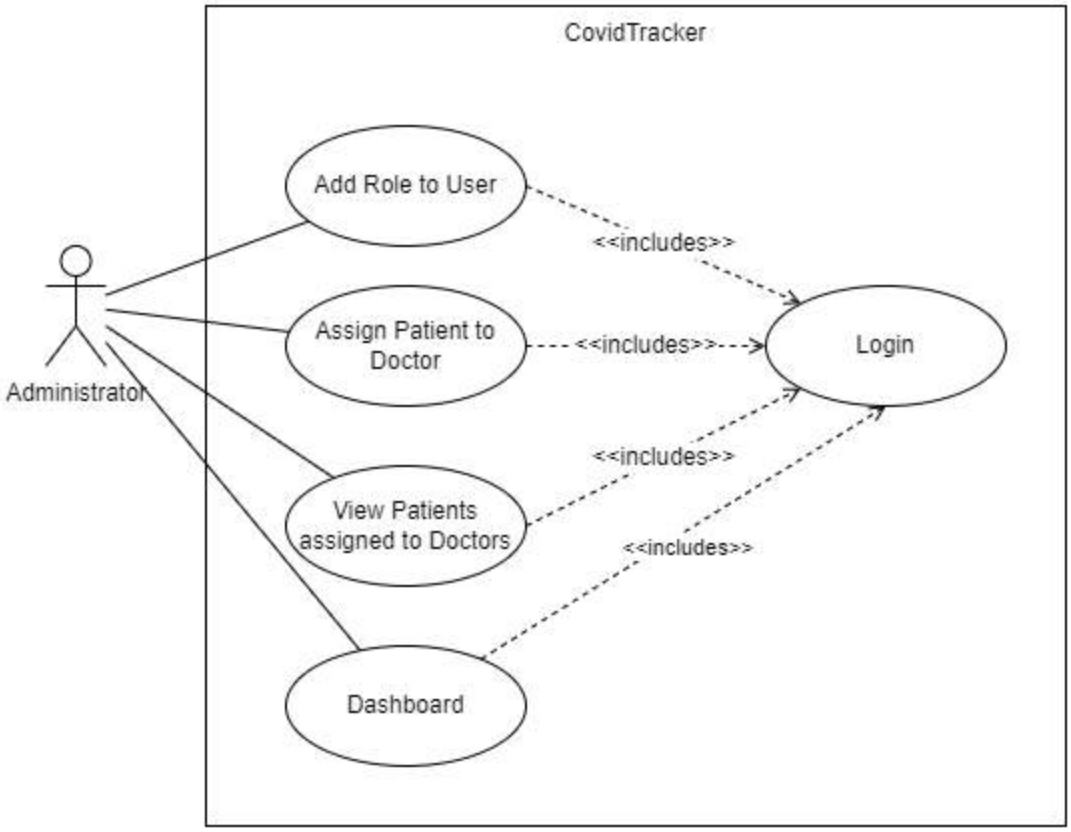


Figure 6: Use Case Diagram of Administrator

5.2.1.2.3 Patient

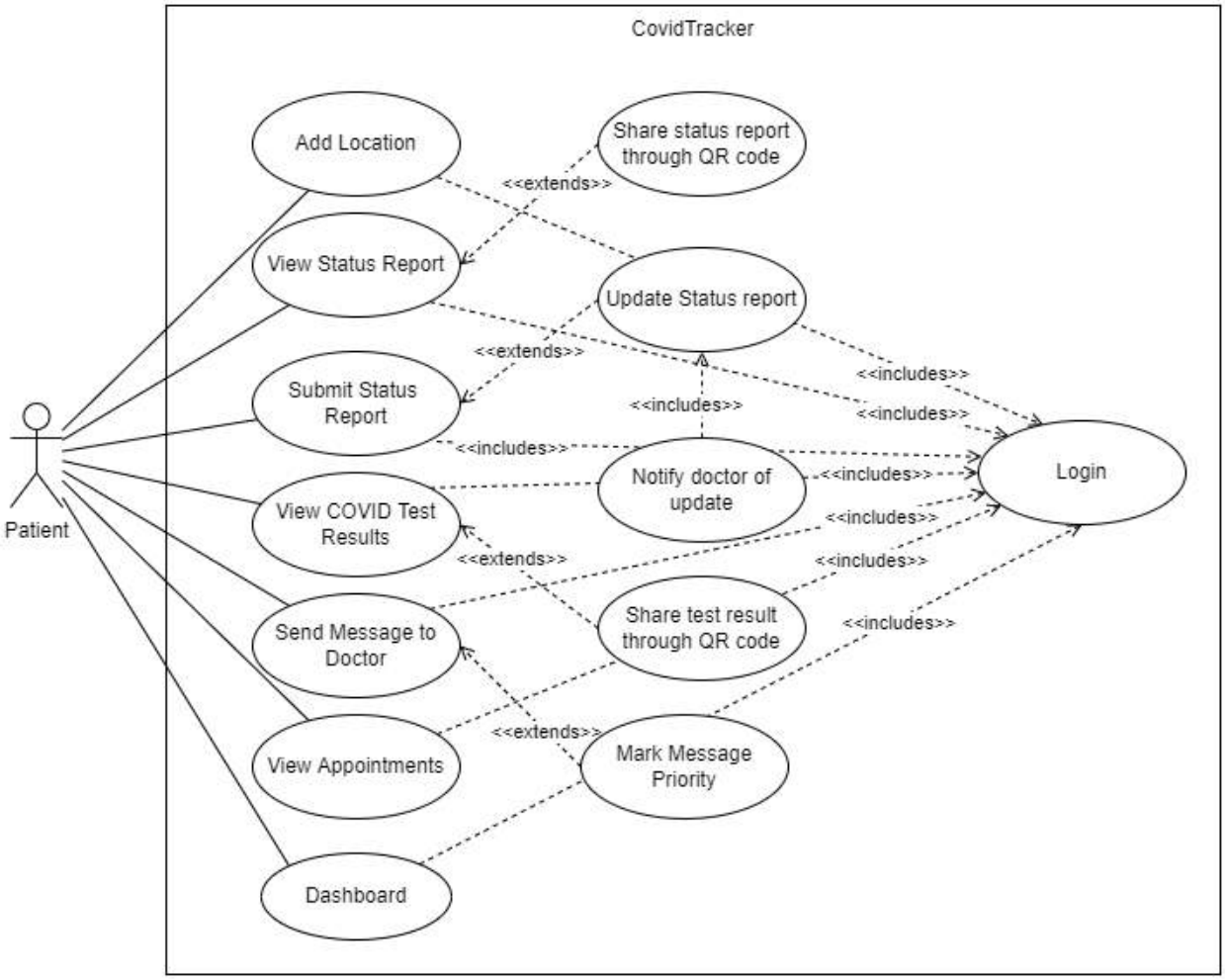


Figure 7: Use Case Diagram of Patient

5.2.1.2.4 Doctor

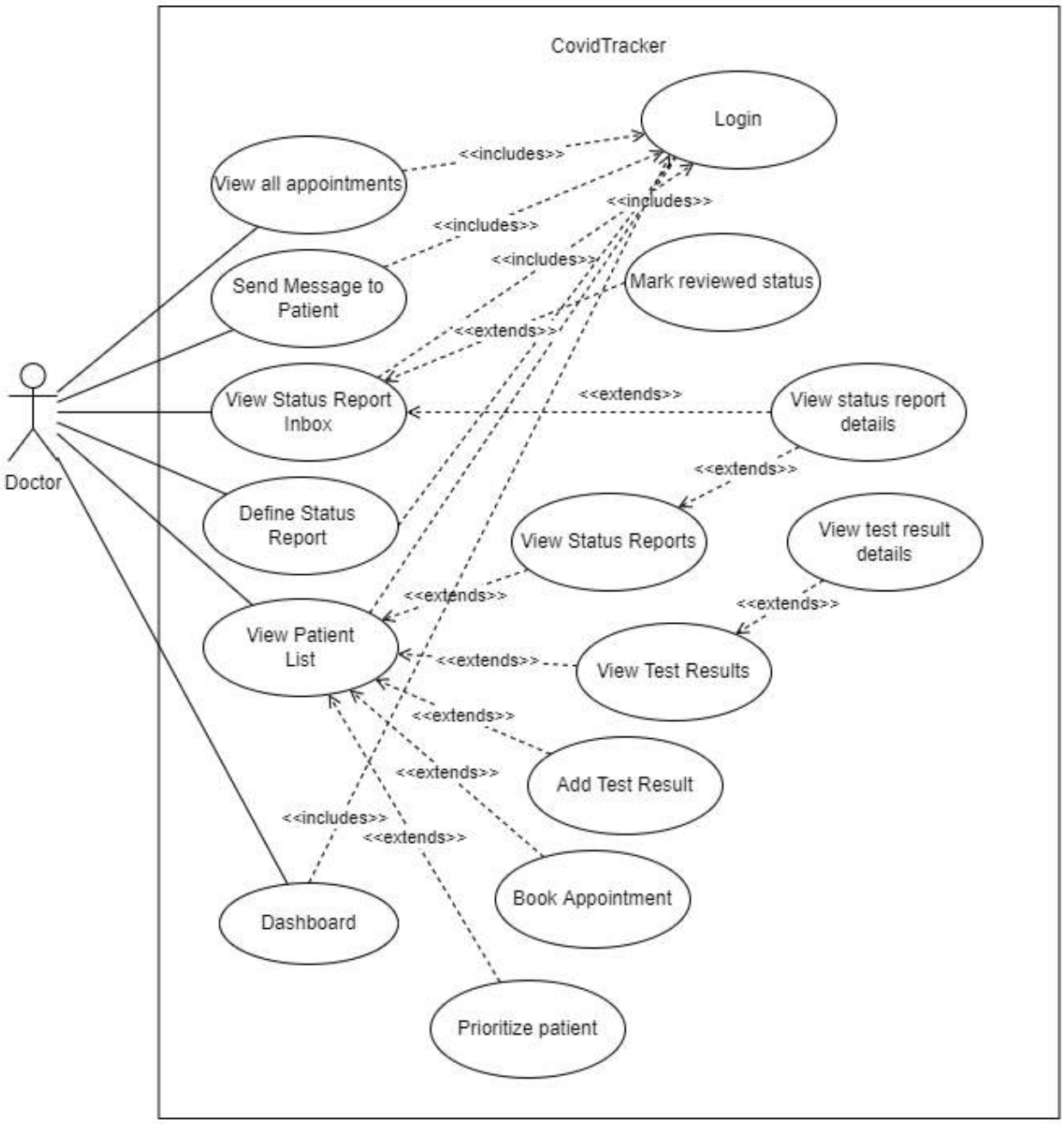


Figure 8: Use Case Diagram of Doctor

5.2.1.2.5 Health Official

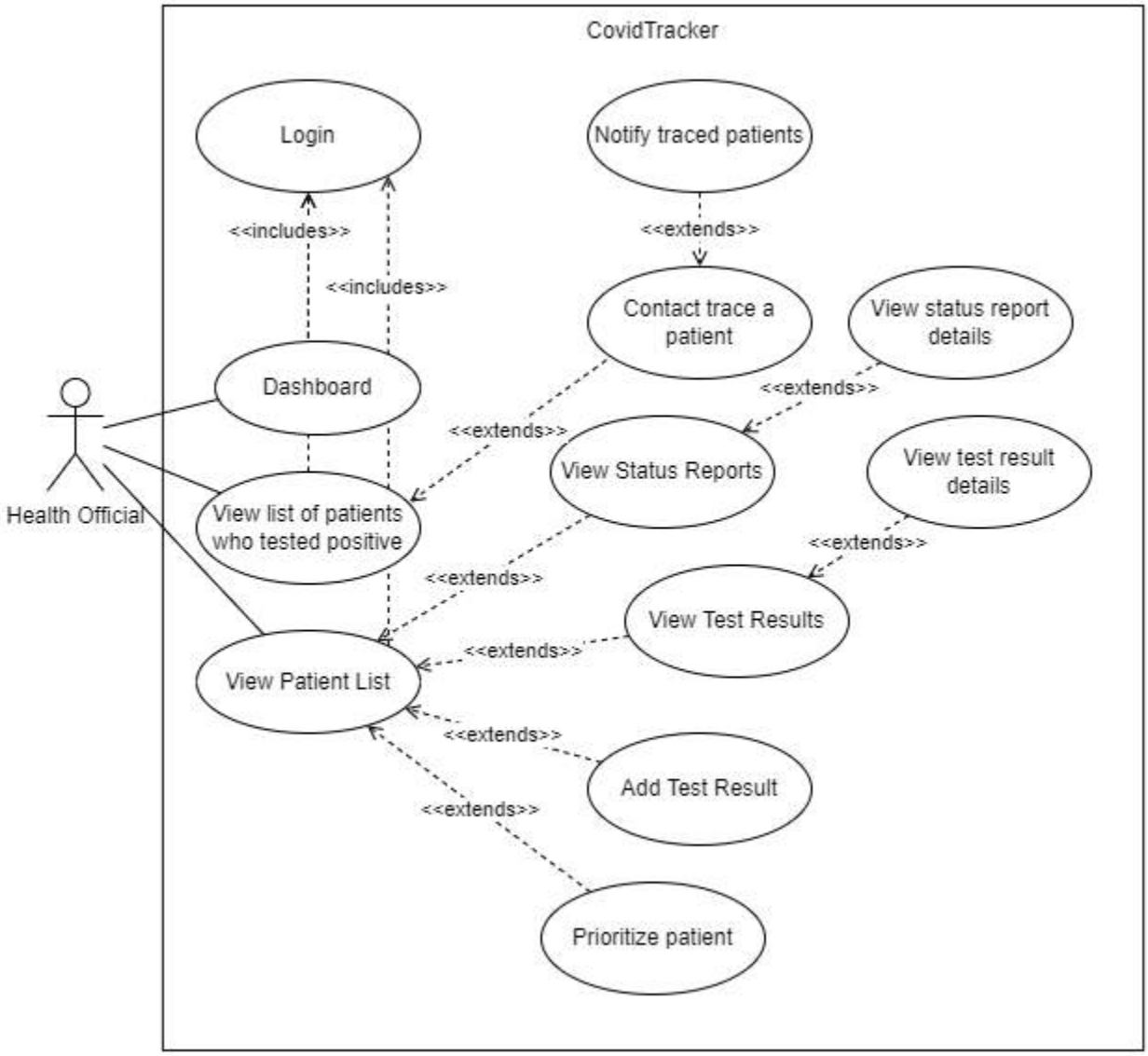


Figure 9: Use Case Diagram of Health Official

5.2.1.2.6 Immigration Officer

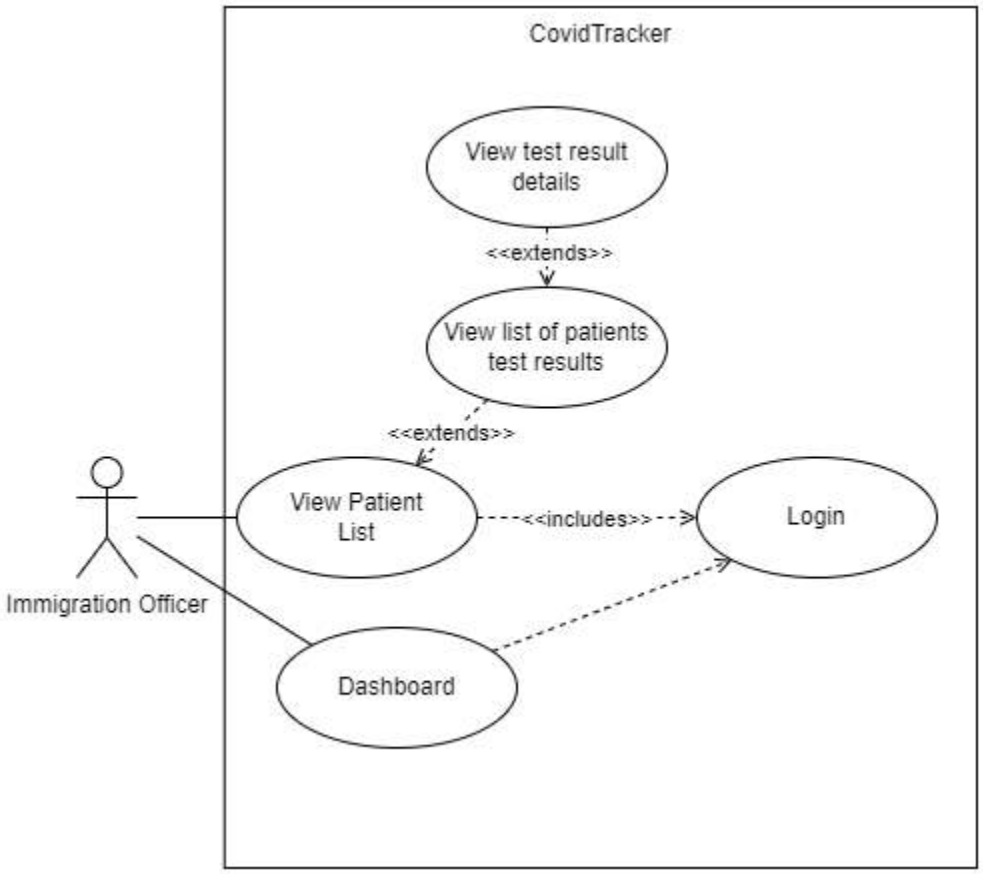


Figure 10: Use Case Diagram of Immigration Officer

5.2.2 Functional Viewpoint

The functional viewpoint describes the system's architectural elements and their associated responsibilities, interfaces, and primary interactions. This viewpoint concerns all the project's stakeholders.

5.2.2.1 Component Diagrams

There are two UML component diagrams for CovidTracker (see Figures 11 and 12) each describing the layers of the system at a different level of abstraction as well as the components within each layer and their relationships. The component diagram (Figure 11) shows the service level components, their required and provided interfaces,

and how they interact. The architecture component diagram (Figure 12) displays the general architecture of the monolithic server and thin client. The diagrams can be viewed in draw.io through this link:

- Component Diagram of CovidTracker

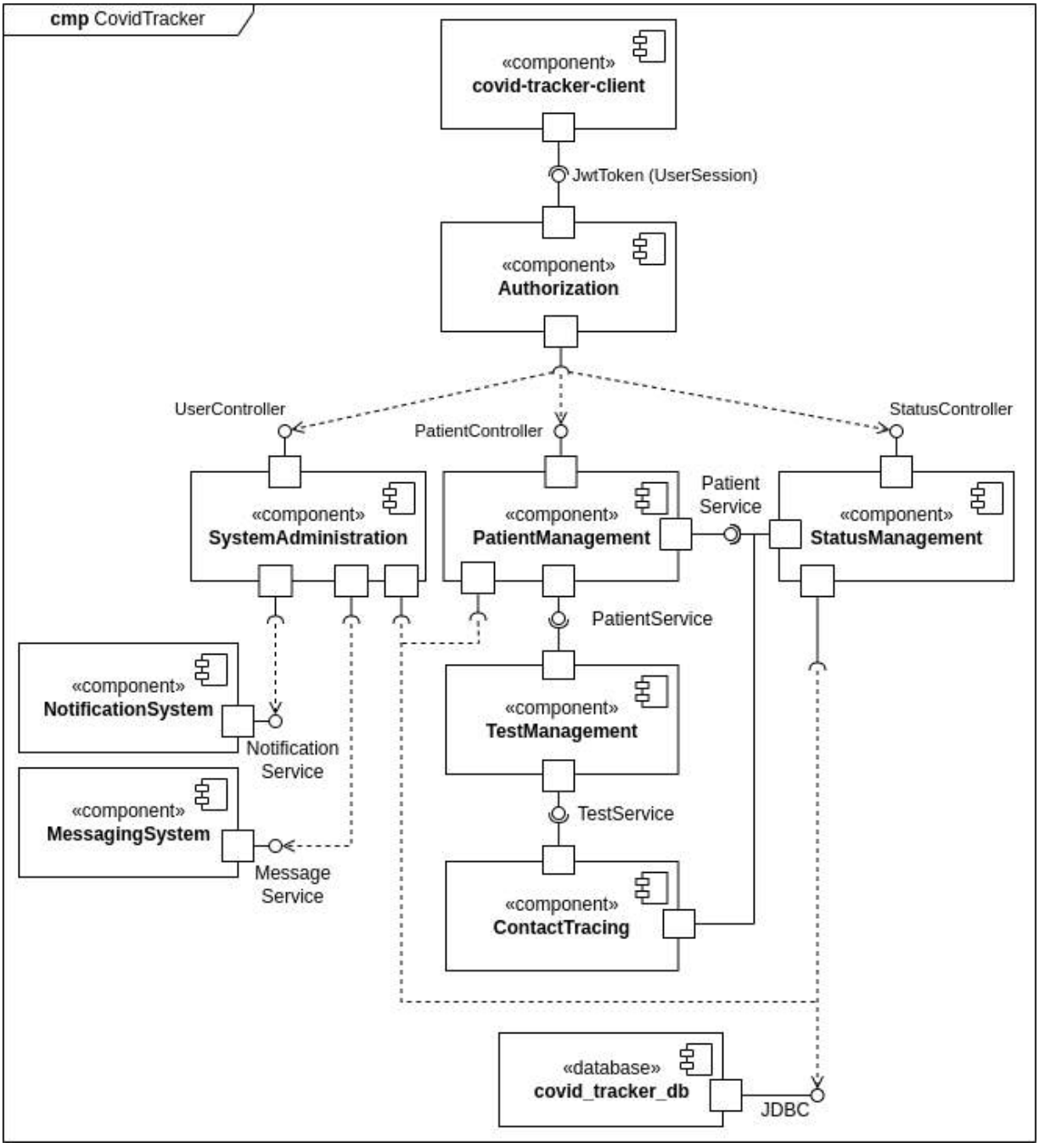


Figure 11: Component Diagram of CovidTracker

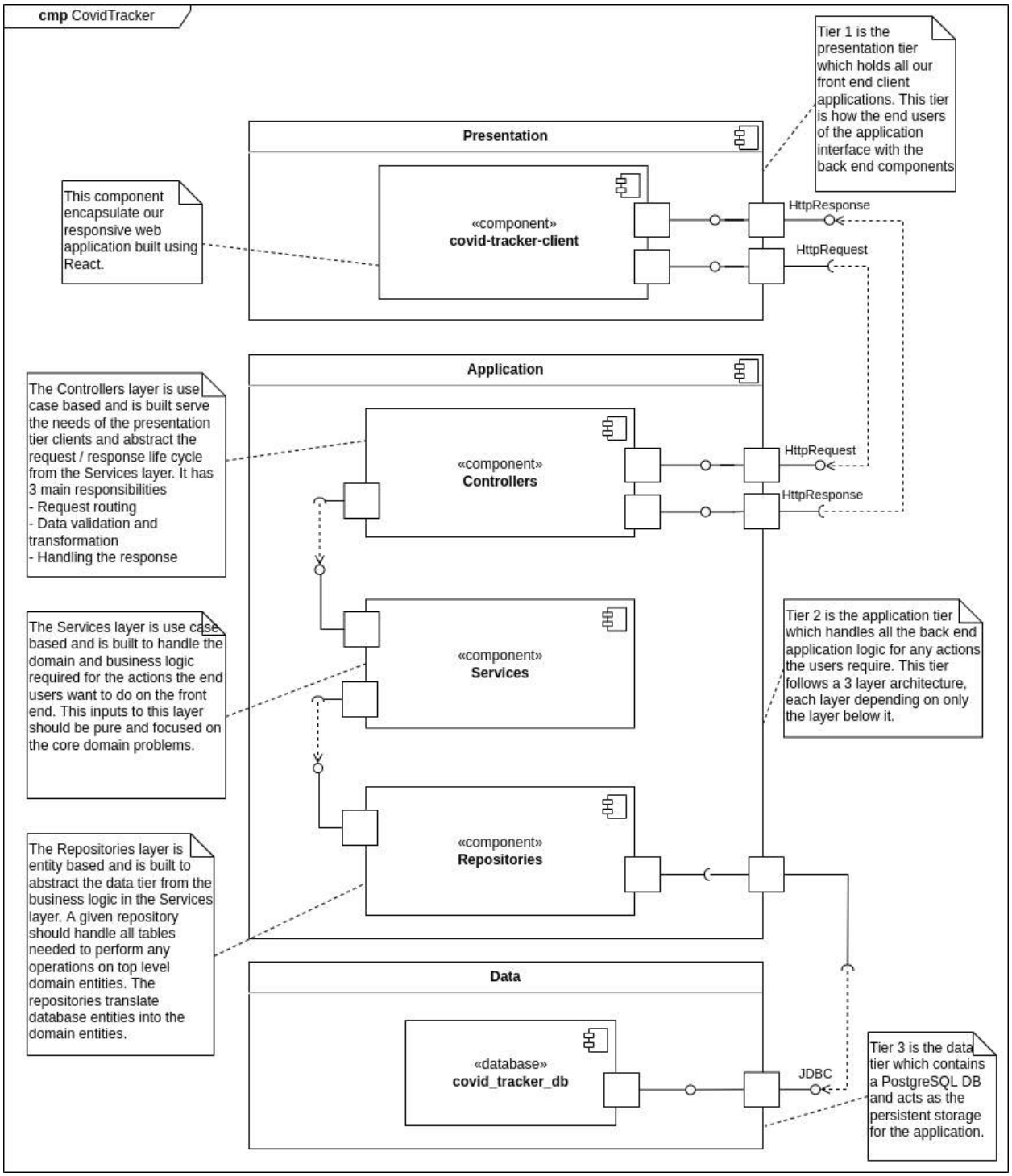


Figure 12: Architecture Component Diagram of CovidTracker

The architecture of the system follows a 3-Tier architecture with the middle application tier using a layered architecture. The 3 tiers include the presentation tier, application tier and data tier. The presentation tier is the front end of the application. The application tier contains our web server which handles all the business logic of our application. Lastly, the data tier is the persistent storage layer for the application. The application tier has 3 layers which include: controller, service, and repository layers that allows the system to have low coupling and high cohesion. We utilize dependency injection to further separate the concerns between our layers. The standard flow of data starts at the presentation tier (the frontend) where a HTTP request is made to the application tier. The controller layer handles all routing, passing the data to the service layer, then to the repository layer to convert to schema form and make the JDBC connection with the database tier to persist the data.

5.2.3 Information Viewpoint

The information viewpoint describes the way that the system stores, manipulates, and manages the interactions between the complex data structures that form the domain of the system. The stakeholders for this viewpoint are developers, testers, and system analysts.

5.2.3.1 Domain Model

The UML domain model for CovidTracker describing all system entities, relationships and associations is represented by the UML domain model diagram seen in Figure 13. The diagram can be viewed in draw.io through this link:

- UML Domain Model Class Diagram of CovidTracker

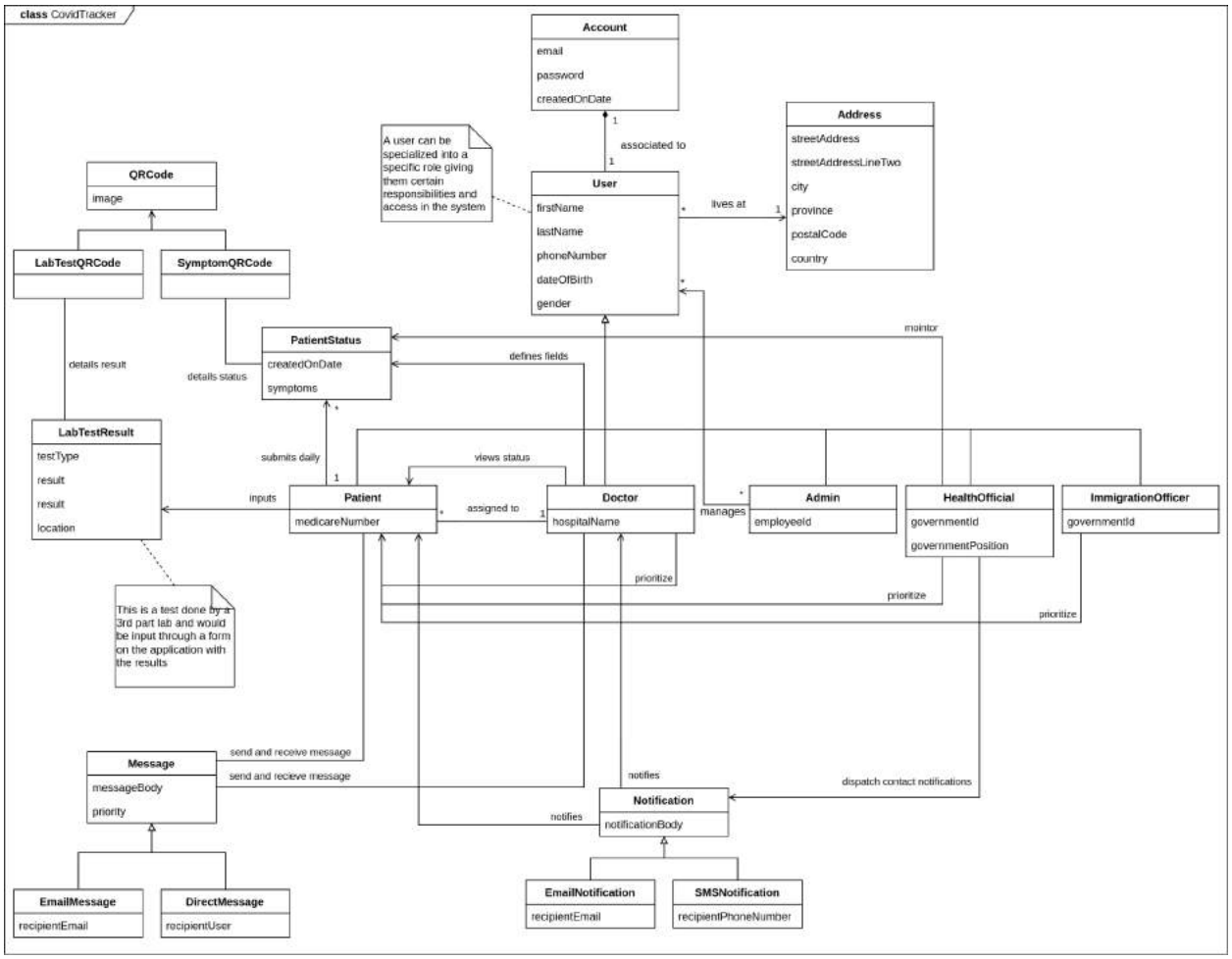


Figure 13: UML Domain Model Class Diagram of CovidTracker

A few key decisions were made during the creation of the Domain Model. The user entity is split into 2 separate entities: person and account. A user has a single account and then can have a specialized role through inheritance (i.e. Doctor, Admin, Patient etc.). This encapsulates the core attributes of a role in the user and keeps it modular from the account itself which gives us a lot of flexibility when designing the authentication and authorization system.

5.2.4 Development Viewpoint

The development viewpoint describes the lower level implementation details of the core system features. These diagrams depict the structure and dependencies of various classes and services when fulfilling the applications features and use cases. The stakeholders for this viewpoint are developers and testers.

5.2.4.1 Status Management

Define Status Report

The following activity diagram describes the activity of a doctor defining the status report fields that a patient must fill up daily after testing positive for COVID-19. Further information can be found in section 7.3.6 Define Status Report.

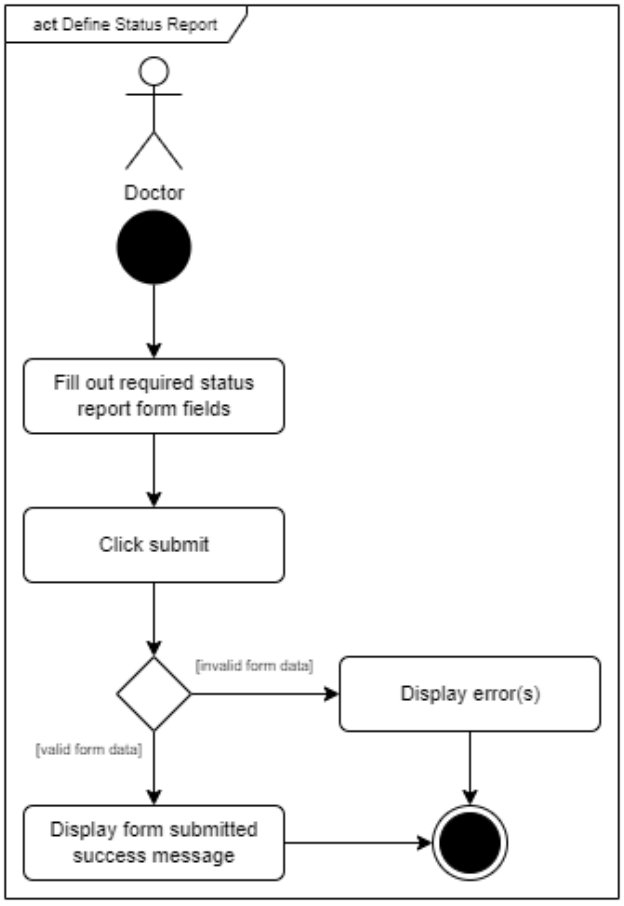


Figure 14: Activity Diagram of Define Status Report

View Status Report

The following activity diagram describes the activity of a doctor viewing patient status reports, sharing the associated QR code when a second opinion is needed, and marking said status report as reviewed.

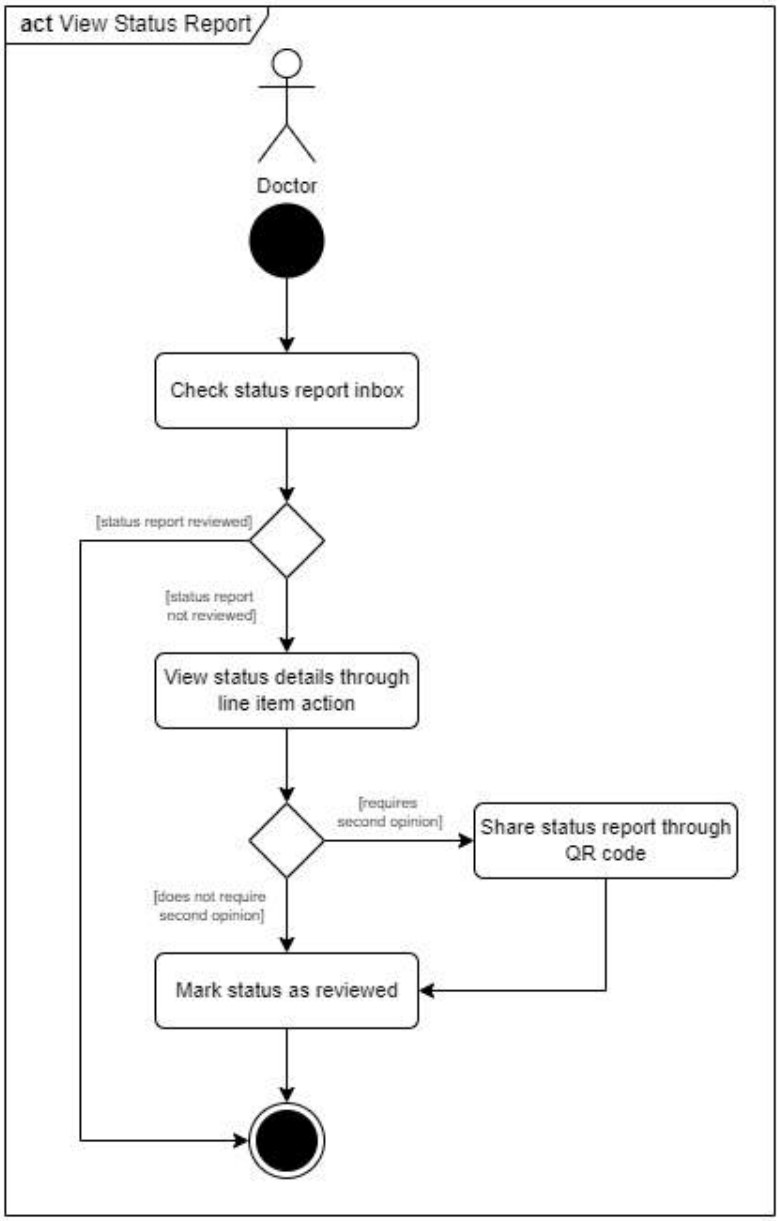


Figure 15: Activity Diagram of View Status Report

Submit Status Report

The following sequence diagram describes the various system object interactions during the submit status report activity. Further information can be found in section 7.3.7 Status Report.

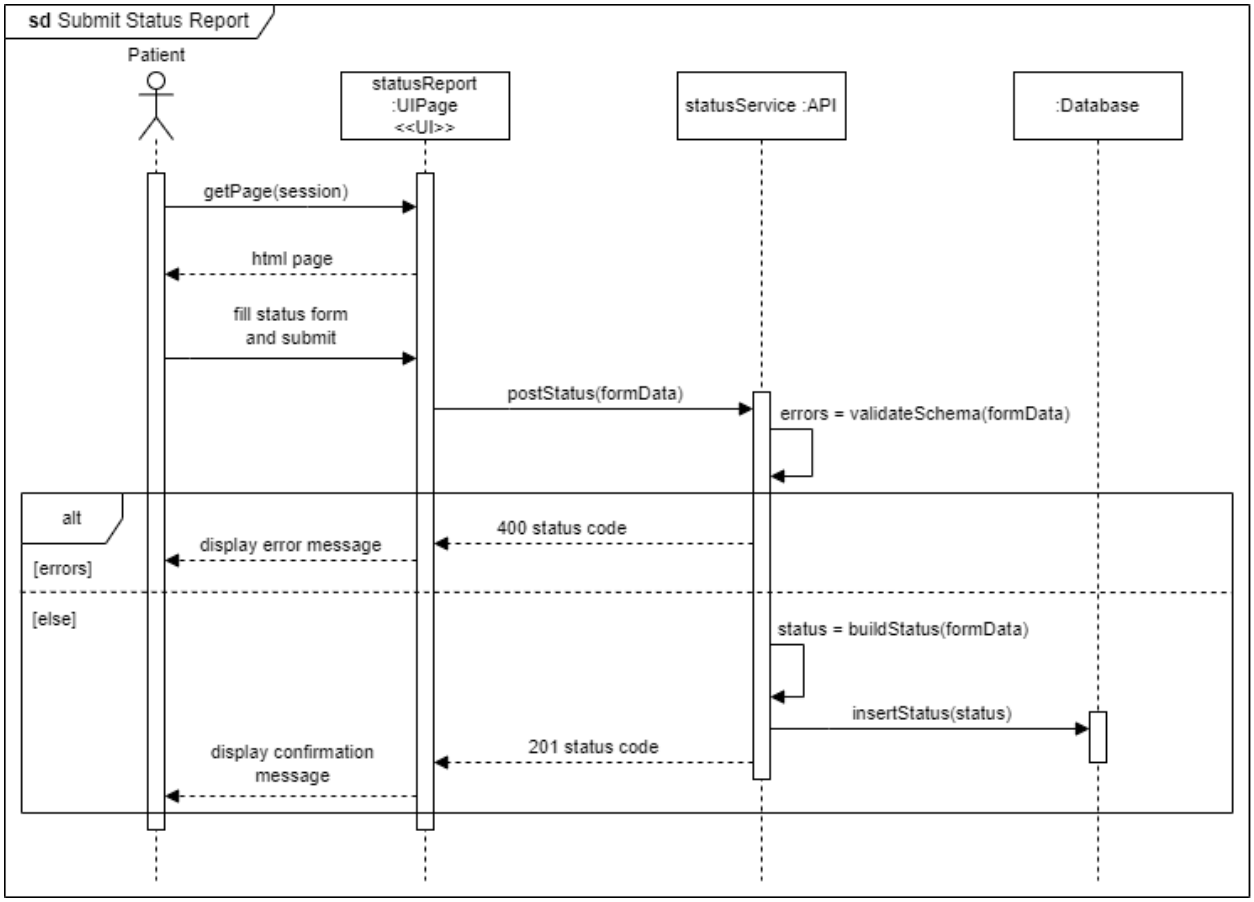


Figure 16: Sequence Diagram of Submit Status Report

5.2.4.2 Patient Management

Add Test Result

The following activity diagram describes the activity of a doctor adding a test result to a patient's records after testing positive for Covid-19.

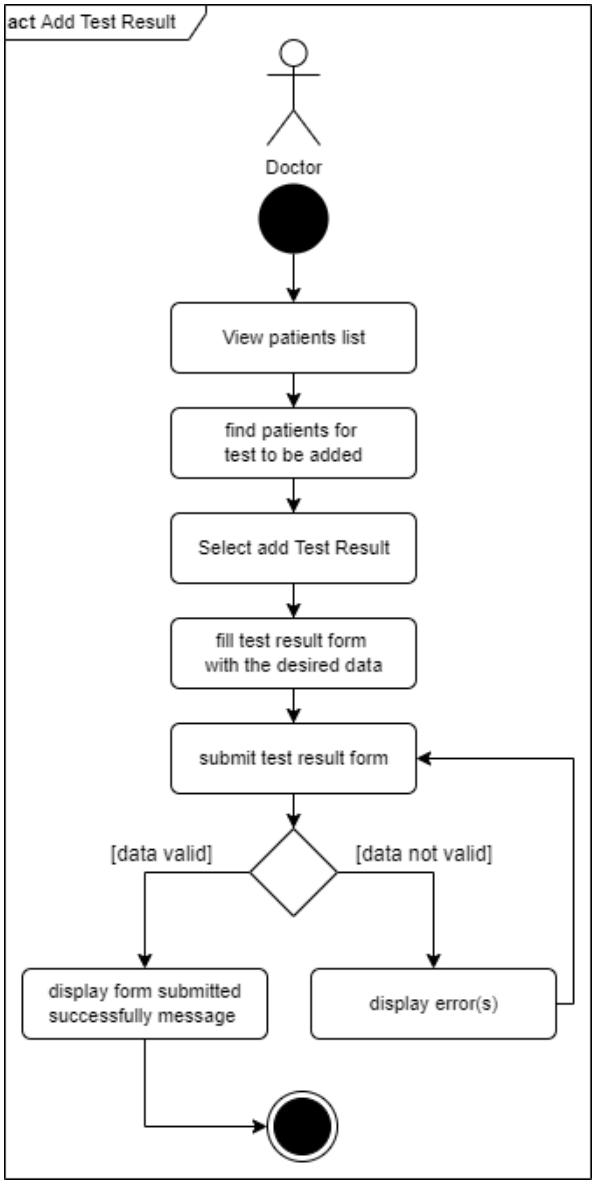


Figure 17: Activity Diagram of Add Test Result

View Test Results

The following sequence diagram describes the various system object interactions during the view test results activity. Further information can be found in section 7.3.14 Test Results.

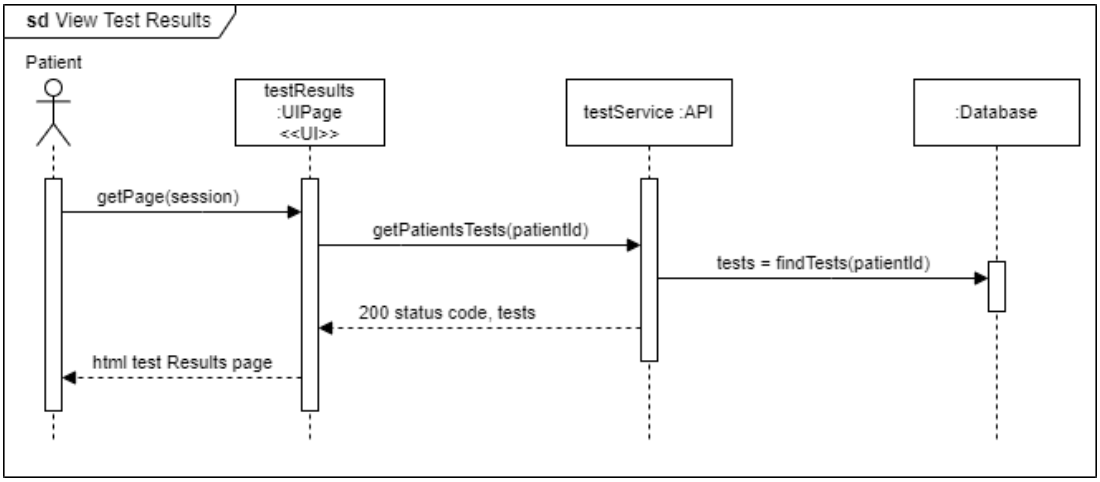


Figure 18: Sequence Diagram of View Test Results

View Patient List

The following sequence diagram describes the various system object interactions during the view patient list activity. Further information can be found in section 7.3.9 Patient List.

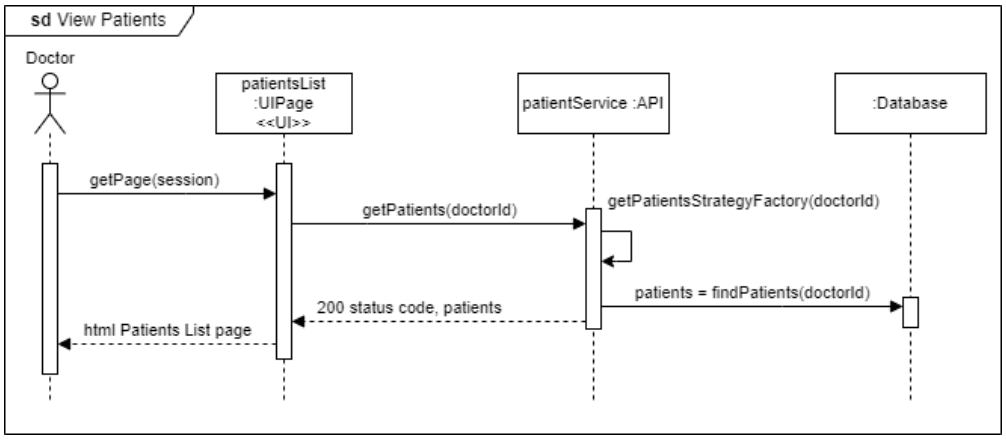


Figure 19: Sequence Diagram of View Patient List

Prioritize Patient

The following sequence diagram describes the various system object interactions during the prioritize patient activity. Further information can be found in section 7.3.9 Patient List.

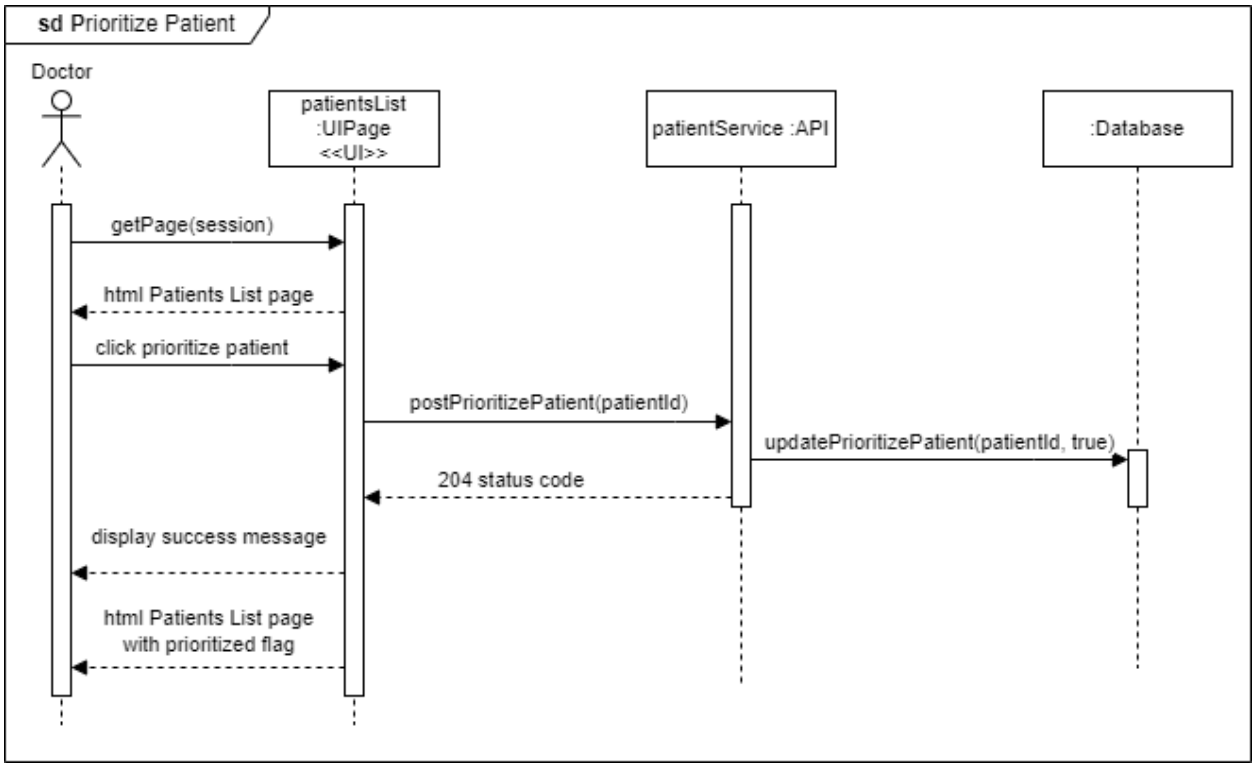


Figure 20: Sequence Diagram of Prioritize Patient

5.2.4.3 Messaging System

Send Message

The following sequence diagram describes the various system object interactions during the send message activity. Further information can be found in section 7.3.16 Chat.

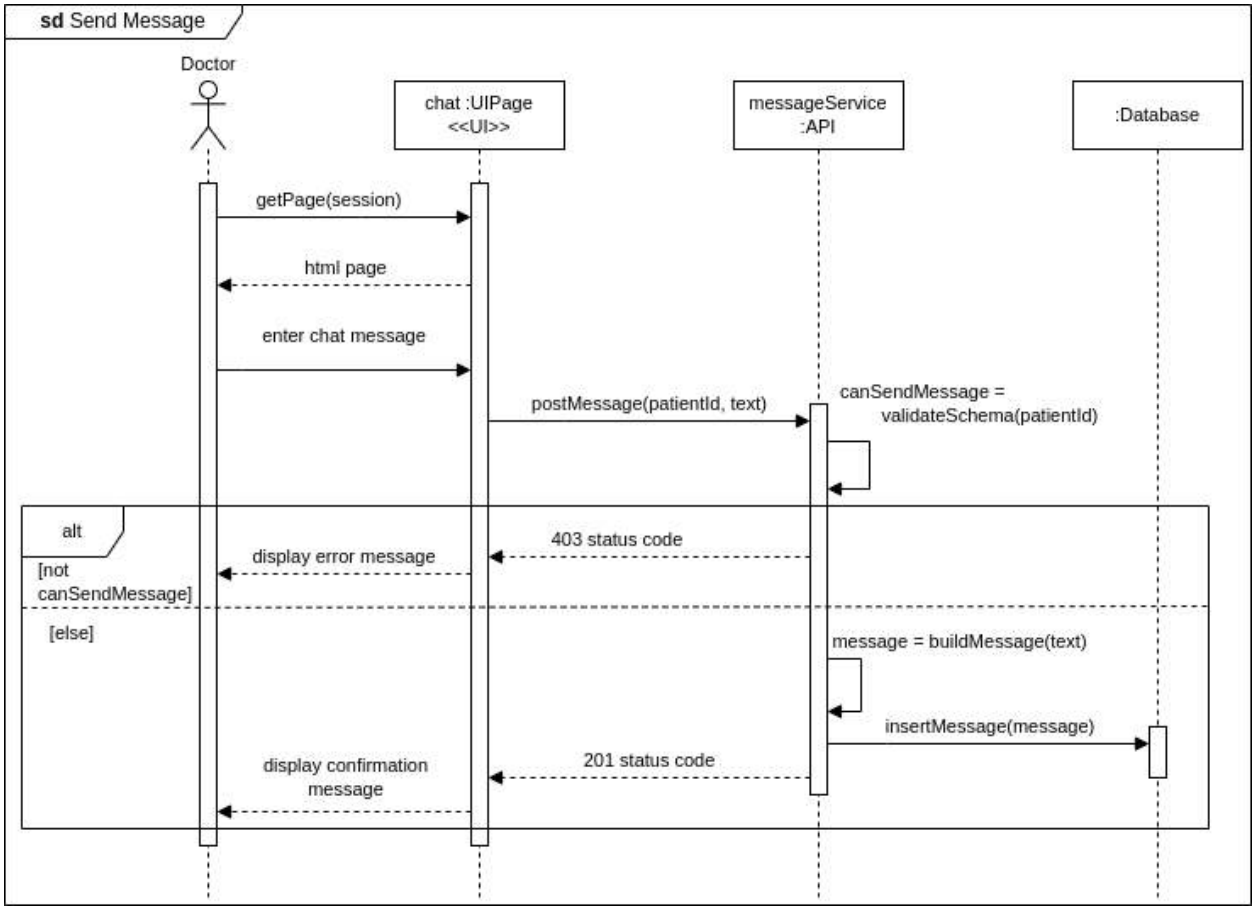


Figure 21: Sequence Diagram of Send Message

Send Prioritized Message

The following sequence diagram describes the various system object interactions during the send a prioritized message activity. Further information can be found in section 7.3.16 Chat.

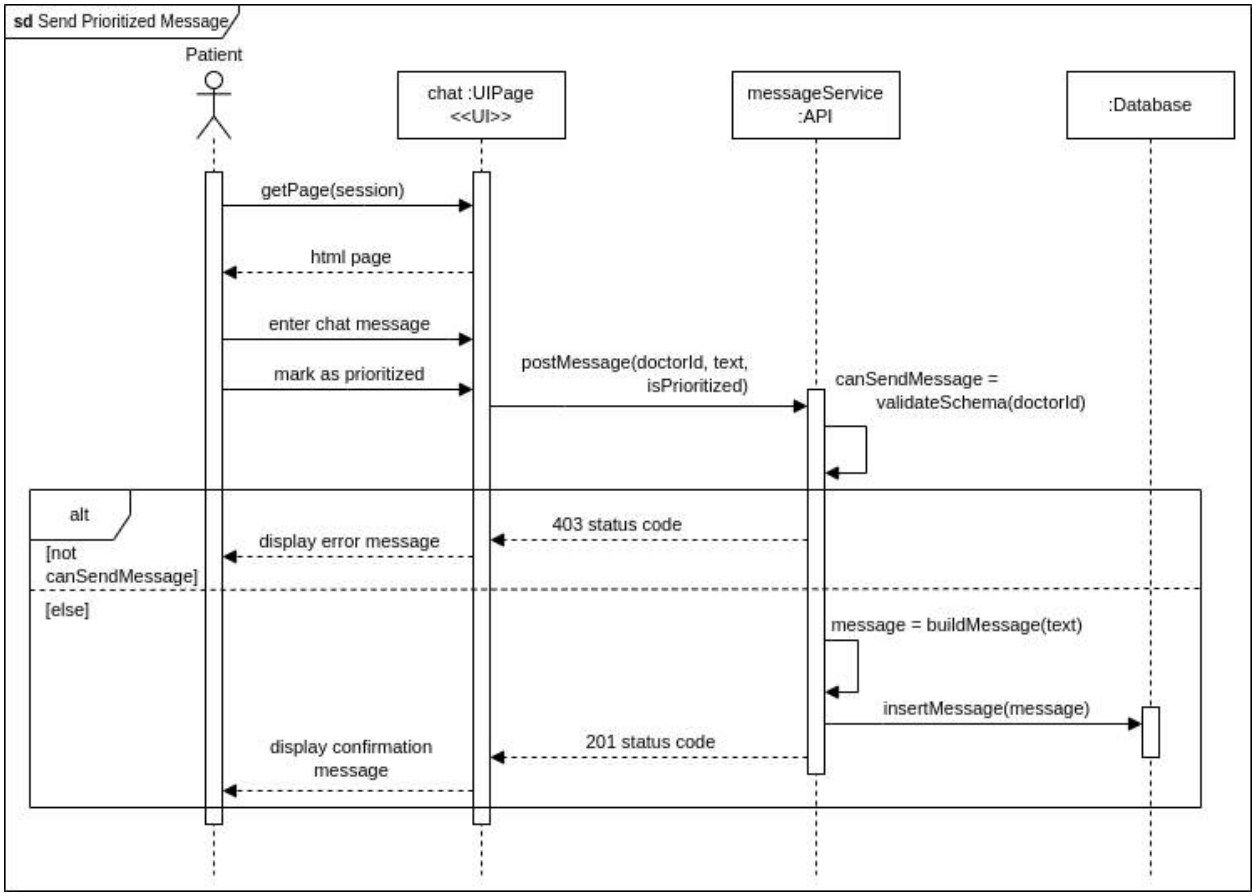


Figure 22: Sequence Diagram of Send Prioritized Message

View Messages

The following sequence diagram describes the various system object interactions during the view messages activity. Further information can be found in section 7.3.16 Chat.

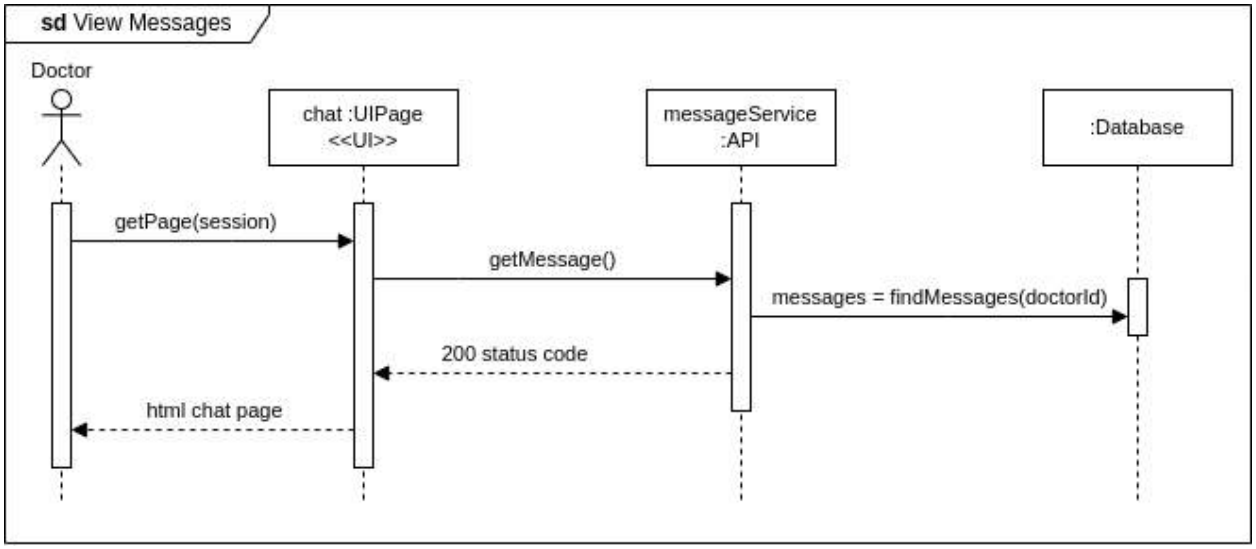


Figure 23: Sequence Diagram of View Messages

5.2.4.4 Dashboards

View Dashboard

The following sequence diagram describes the various system object interactions during the view dashboard activity. Further information can be found in section 7.3.16 Chat.

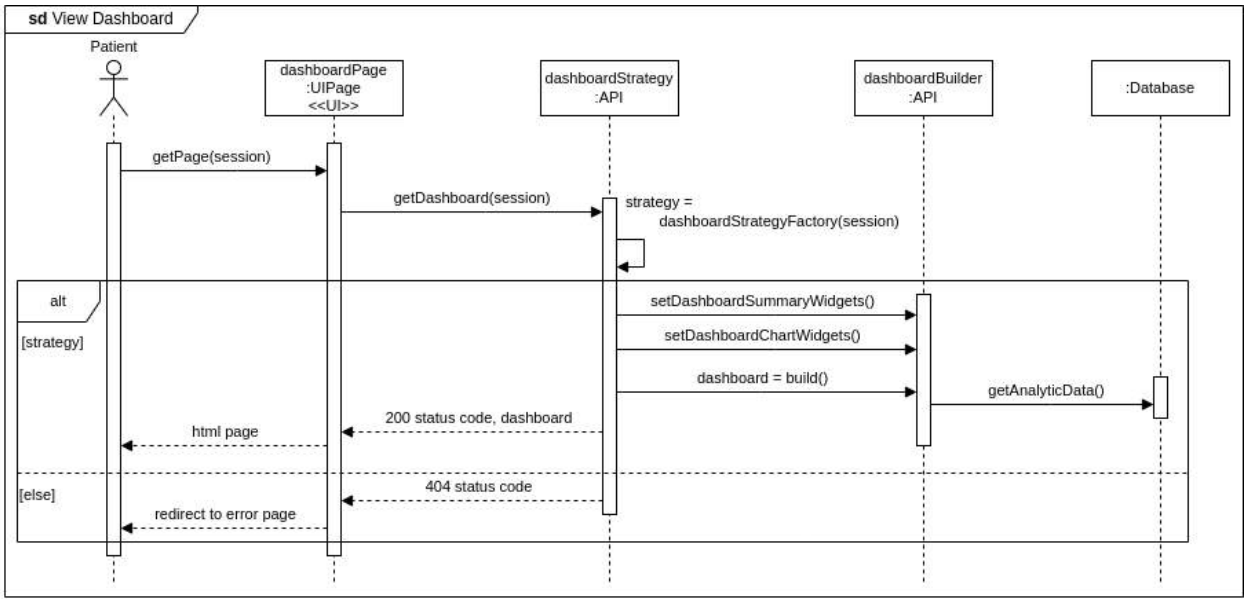


Figure 24: Sequence Diagram of View Dashboard

5.2.4.5 Contact Tracing

Contact Trace Patient

The following activity diagram describes the activity of a health official contact tracing and notifying individuals to self-quarantine because they were in contact with a positive patient.

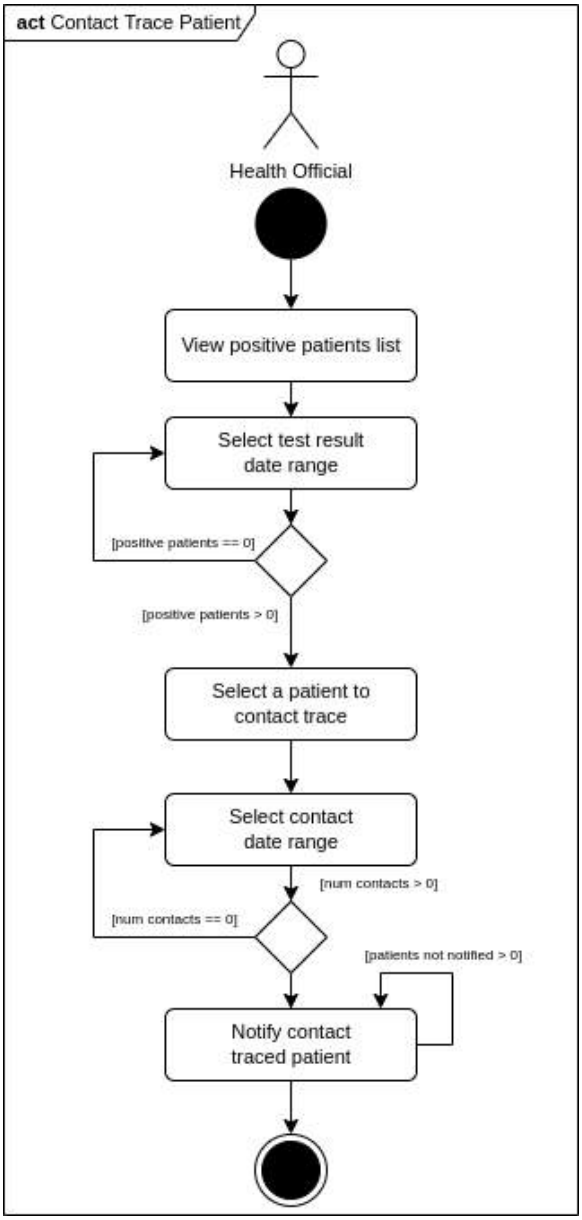


Figure 25: Activity Diagram of Contact Trace Patient

Add Location Report

The following sequence diagram describes the various system object interactions during the add location report activity. Further information can be found in section 7.3.19 Add Location.

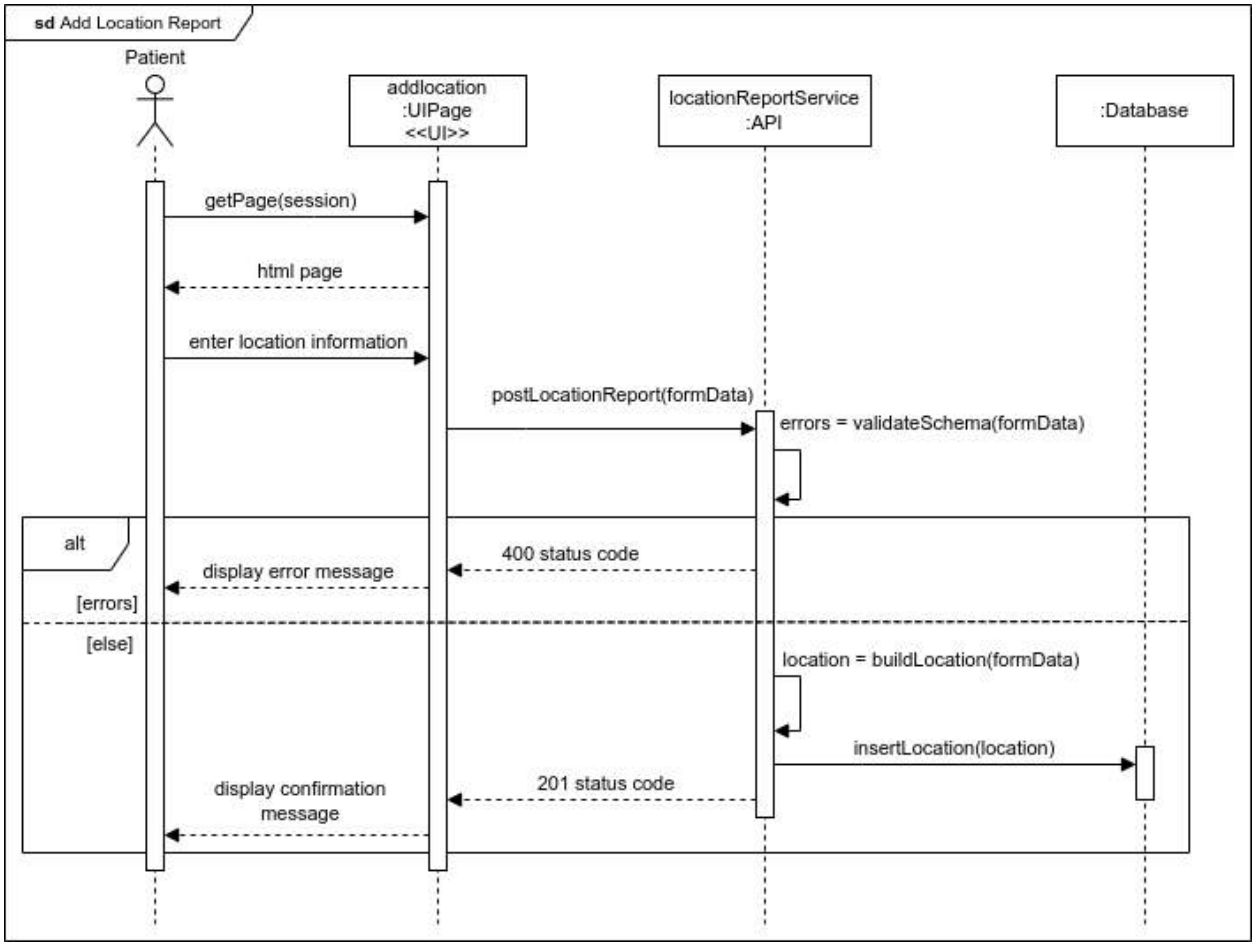


Figure 26: Sequence Diagram of Add Location Report

5.2.4.6 Appointments

Book Appointment

The following sequence diagram describes the various system object interactions during the book appointment activity. Further information can be found in section 7.3.17 Book Appointment.

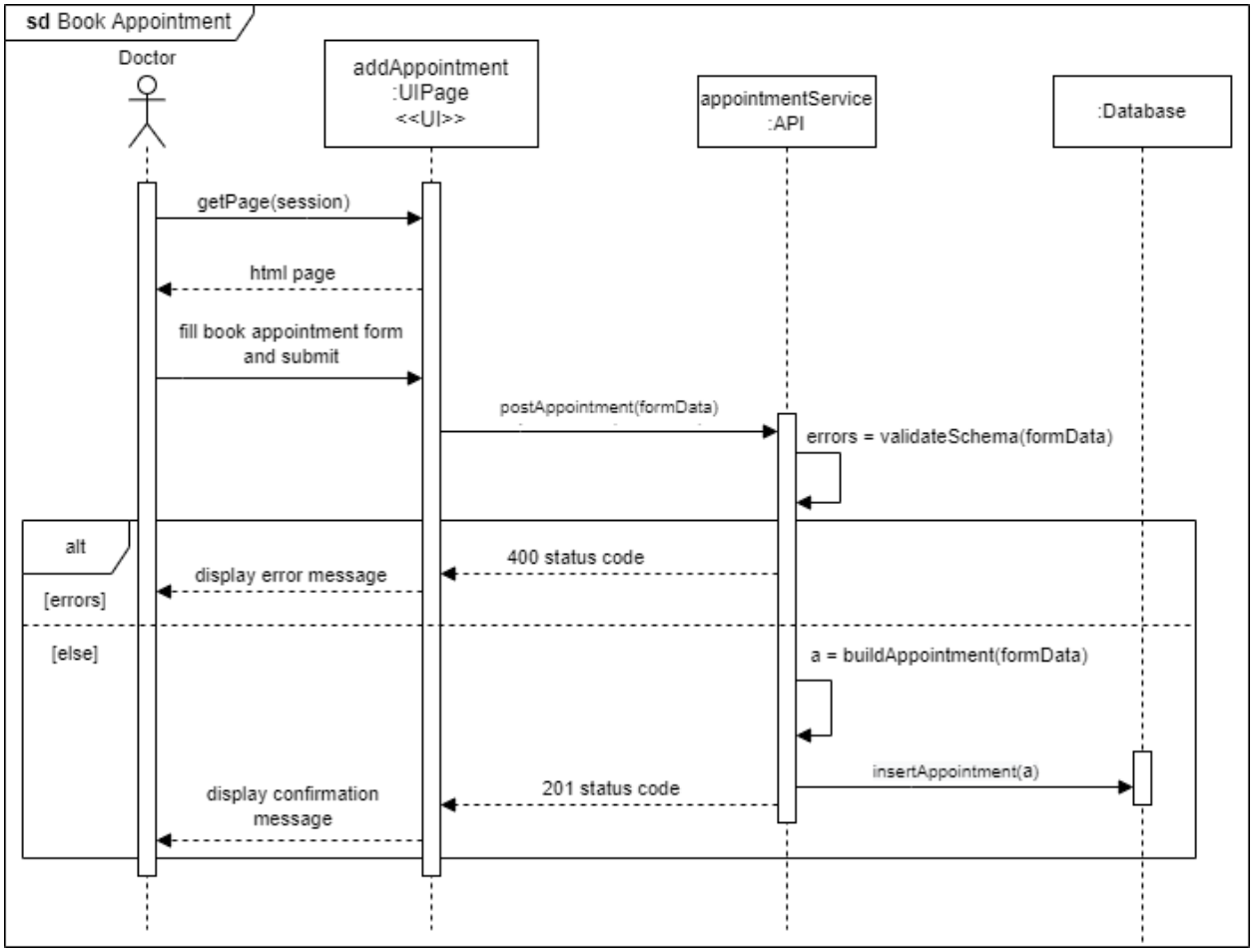


Figure 27: Sequence Diagram of Book Appointment

View Appointments

The following sequence diagram describes the various system object interactions during the view appointment activity. Further information can be found in section 7.3.18 Appointments.

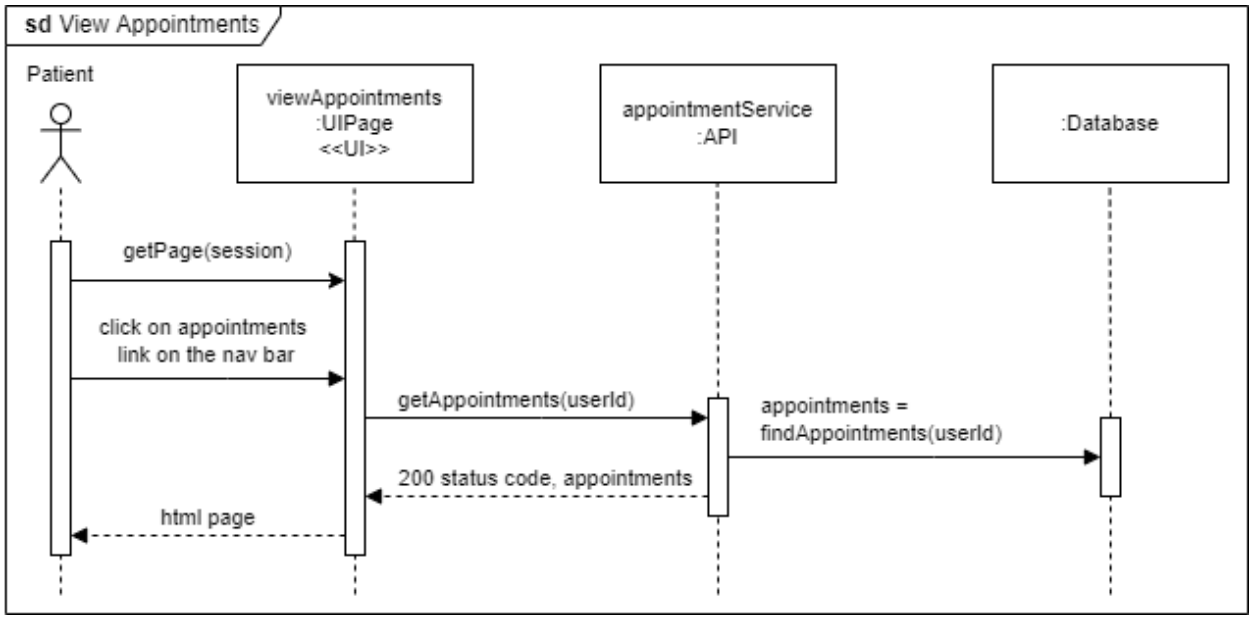


Figure 28: Sequence Diagram of View Appointments

5.2.5 Deployment Viewpoint

The deployment viewpoint describes the environment into which the system will be deployed, including the dependencies the system has on its runtime environment. The stakeholders for this viewpoint are developers and testers who focus on the infrastructure and deployment of the system.

5.2.5.1 Deployment Diagram

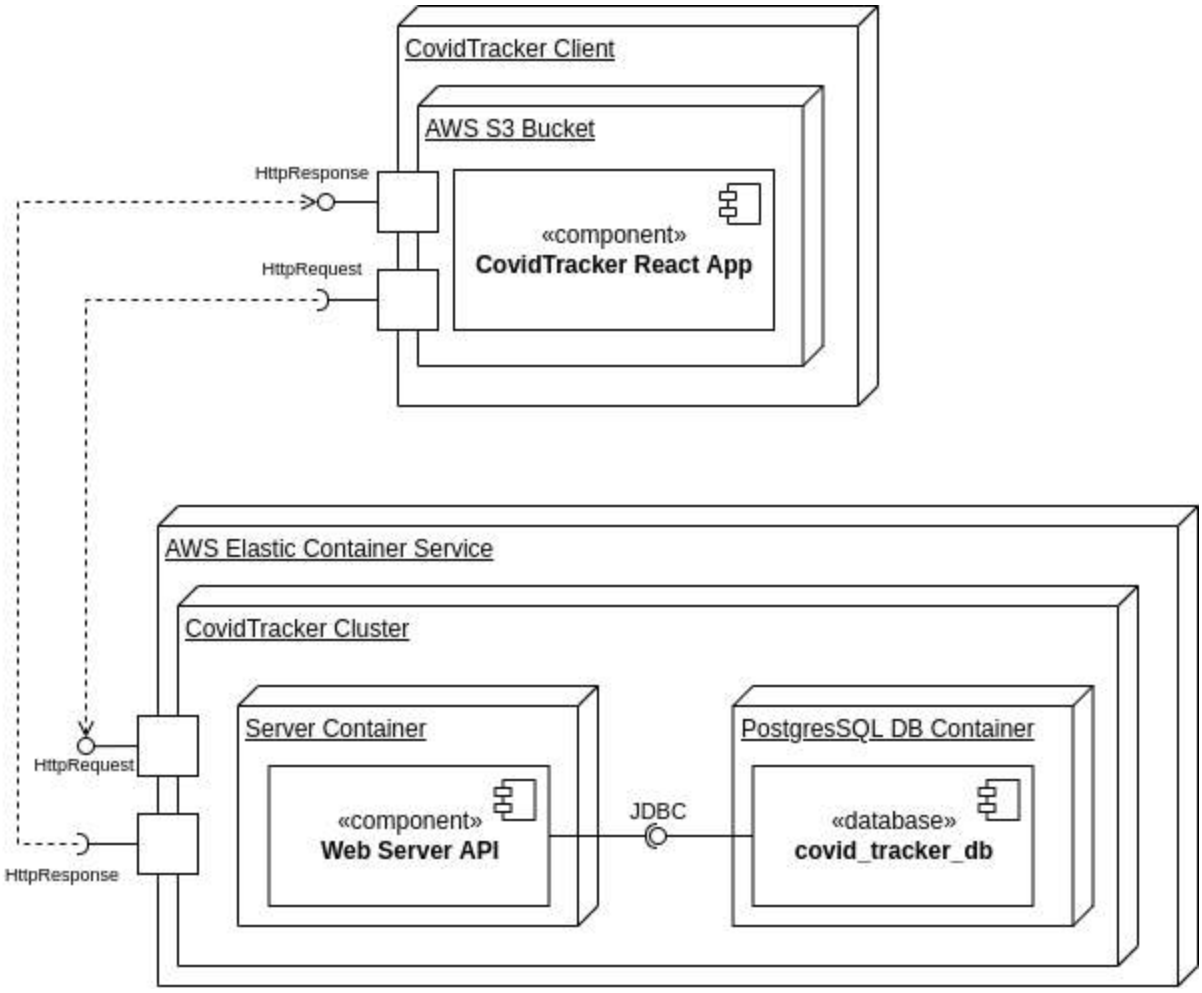


Figure 29: Deployment Diagram of CovidTracker

5.3 Tech Stack

5.3.1 Presentation Tier

The client is encapsulated by the presentation tier and currently contains a single front end service, built using JavaScript and React, which is the primary way a client can interact with the system.

React was chosen because it is one of the largest and most tested front end frameworks on the open source market. Other competitors include Vue.JS and Angular. Angular is outdated and leans towards a thick client approach which is mostly considered an anti-pattern with front end web applications as they can become slow and hard to maintain overtime. While Vue.JS is a great framework, React ended up being chosen simply because more people on the team had experience with it.

The front end also uses SCSS which is a CSS preprocessing language that allows us to extend what can normally be done with CSS such as adding variables and mixins.

The client will be running in its own docker container which will allow it to be easily productionalized and deployed while also being compatible on all our developers' local machines.

5.3.2 Application Tier

The application tier consists of a single web server accepting HTTP requests from the front end client. This monolithic service is built using TypeScript and runs inside a docker container. TypeScript was chosen due to its clean syntax and advanced type system. As a result of TypeScript being transpiled in JavaScript, all JavaScript libraries work with TypeScript as well. This means we can also take advantage of the vast open source libraries built over the years by the web service oriented JavaScript community. Since we wanted to use a statically typed language to make it easier to model our domain in code and build clean type hierarchies, Python and JavaScript were out. Java was also considered however even with recent advancements aiming to improve the language it is still verbose and extremely opinionated in its packages with Java Spring being a perfect example. There was simply a desire to choose a language that was more flexible.

The web framework the service uses is Restify paired with the dependency injection package called Inversify. Restify was chosen because it is a simple framework that is not opinionated and allows us to only use and build what is needed. Thus abstracting away a lot of the boiler plate while still providing the ability to customize the service as needed. Other alternatives include express which has great 3rd party library support due to its popularity.

The web service uses an NPM package called pg which is a non-blocking client for PostgreSQL that provides the ability to write native SQL statements and abstract away the database connection pooling and type coercion. The main alternative for database interaction would be to use an object relational mapper (ORM), a popular one being TypeORM, that abstract away the entire concept of SQL statements and schema and leave a simple API for the application developer to work with. In our team's experience ORMs only make it easy to do the simple stuff which is easy to do anyways. When it comes to the complex parts you have to write the SQL by hand anyways so it's best to have full control from the start.

5.3.3 Data Tier

The data tier is currently using a single PostgreSQL database to store all the needed relational data. PostgreSQL was chosen as it is the most modern and maintained open source relational database. There is a large community that creates many libraries for all different types of languages and so many people on the team were already familiar with its standard SQL syntax. Other alternatives include MySQL and Oracle Database but these database systems lack some of the advanced features and polish PostgreSQL has. The PostgreSQL instance our application uses runs inside its own docker container to make setup quick and easy for the development team.

5.3.4 Deployment

The system is deployed in two parts using Amazon Web Services (AWS) cloud deployment services.

The first part consists of deploying the backend API and database. This is done through registering the respective docker container images in Elastic Container Registry (ECR). From there a new Elastic Container Service (ECS) task is started in a cluster for the application. The API and database are running in the same task on the same cluster.

The second part is for the client which is a single page application (SPA) built using React. This allows us to build the front end into static HTML, CSS, and JavaScript files along with the rest of our static assets. This content is pushed into a public static web hosting enabled bucket on AWS Simple Storage Service (S3). From there, clients can access the website through the publicly accessible S3 bucket URL.

The application can be reached at the following link:

- http://covid-tracker-client-bucket.s3-website.us-east-2.amazonaws.com/

Note to reader: To save costs the backend and database are scaled down to 0 instances in the cluster when not in use. If you are trying to access the website let Team 17 know on slack and we can spin up an instance.

The user account credentials needed to access the various features in CovidTracker are represented in the following table:

User Role	**Email**	**Password**
Patient	Rebecca_Feeney@hotmail.com	Test123!
Admin	admin@test.com	Test123!
Doctor	doctor@test.com	Test123!
Health Official	health_official@test.com	Test123!
Immigration Officer	immigration_officer@test.com	Test123!

Table 6: User Account Credentials for CovidTracker

5.4 External Libraries

5.4.1 Vuexy

Vuexy is a user interface templating library used for building user interface components and layouts in CovidTracker. Vuexy provides a fast and easy approach for building responsive web applications as it is built with a mobile first mindset. As such, this ensures that any element, component or feature built with the corresponding set of

tools provided results in a responsive application from the onset. Without such a library, more painstaking time would be required to ensure that all user interface components function as expected regardless of platform. As such, developers can spend more time focusing on device compatibility and fine tuning the overall user experience.

Vuexy's underlying architecture is built on various external React libraries and Bootstrap 5, thus ensuring vast compatibility and various ways to build elements. Developers are free to decide whether or not using a pre-built Vuexy component is a better option instead of either using a basic Bootstrap 5 component or building it from scratch.

Unfortunately, Vuexy does have some major disadvantages. The main one is the sheer complexity of the library. While one would assume it is simple in nature whereby you only need to search and find the relevant components, attempting to find, decipher and then import the associated code is rather complicated. There is an extreme amount of interconnectivity between the provided files that result in a less than ideal amount of time being spent understanding what is necessary for a single component to work. Secondly, due to the interconnectivity of files, it is a rather tedious process to delete any non relevant files given the complex references between all files and displayed errors if not removed properly. Lastly, while the documentation is rather informative and thorough, there is more to be desired in regards to further explanation of certain components and elements.

While other user interface templating libraries were researched - CoreUI, Fuse and Isomorphic -, Vuexy was decided as the go to for two reasons. The first one was due to the variety of pre-built elements and components provided. For example, with the inclusion of layouts, forms, authentication, localization, charts, graphs and interactive data table components, developers do not have to spend many hours building these complicated UI components once a level of understanding is achieved on how to use them. Secondly, Figma UI design files are included, a rarity for these types of libraries. The inclusion of these files are extremely helpful during UI mockup design as the designers do not have to recreate all the elements from the template from scratch. Thus, allowing designers to spend more time focusing on the overall user interface and experience. This also helps ensure both the UI mockups and developed interfaces are a 1 for 1 match.

6.0 RISK ASSESSMENT AND MANAGEMENT PLAN

The risk assessment and management plan for CovidTracker is depicted and described in the following table. Two risks, R-13 and R-14, were added and resolved in sprint 4.

Risk ID	Description	Resolved in Sprint	Strategy and Effectiveness	Probability	Impact
R-1	Computers can crash causing us to lose our work	1	We decided to store our work in the cloud. This strategy has proven successful thus far.	Low	Severe
R-2	Database crashing			Low	Severe
R-3	Database leak and all the patient's medical record is stolen			Low	Severe
R-4	Not having the same versions of software in our systems	1	We decided to use docker to ensure every member is on the same version while working on the project.	Low	Minor
R-5	Changes in the law preventing us from using GPS tracking			Medium	Severe

R-6	Code being pushed to the main without validation	1	A CI/CD pipeline is used in Github and at least one code approval is needed before any code code is merged in the main branch. This ensures the author of the code cannot just merge the code without approval.	Low	Moderate
R-7	Timeline estimates unrealistic	1	Playing poker was used. Every member took a vote on each task for a realistic timeline and we took the average length as the final length for a task.	Medium	Moderate
R-8	Project team availability	1	Every member dedicated a certain amount of time for this class from the beginning of the semester.	Low	Moderate
R-9	Weak user participation (if no one decides to use the website)			Medium	Moderate
R-10	A user having access to page/feature outside the scope of their role	2	The front end and back end of the application restrict users access to certain pages based on their role which is	Medium	Moderate

			authenticated through the JWT.		
R-11	Users not properly reporting their locations	3	The strategy for this risk is acceptance. While this mitigation strategy is not very effective it is necessary to preserve the privacy of the application's users.	Medium	Severe
R-12	User attempting a cross site scripting (XSS) software attack through the chat textbox.			Low	Sever
R-13	Dashboard information isn't displaying correct	4	Implement unit and integration tests to make sure dashboard info correctly reflects the analytics.	Low	Minor
R-14	Patient is double notified during a contact trace	4	The strategy for this risk is acceptance because notifying a patient twice to self quarantine if they are in contact with a positive patient twice is not a major issue.	Low	Minor

Table 7: Risk Analysis Table

7.0 USER INTERFACE DESIGN

All personas, supported devices, UI and user flow mockups, and interactive prototypes are depicted and described in the following sections.

7.1 Personas

CovidTracker is accessible by the following five user personas: Patients (see Figure 30), Doctors (see Figure 31), Health Officials (see Figure 32), Immigration Officers (see Figure 33) and Administrators (see Figure 34). Each persona is considered to be representative of a certain archetype within the general demographic.

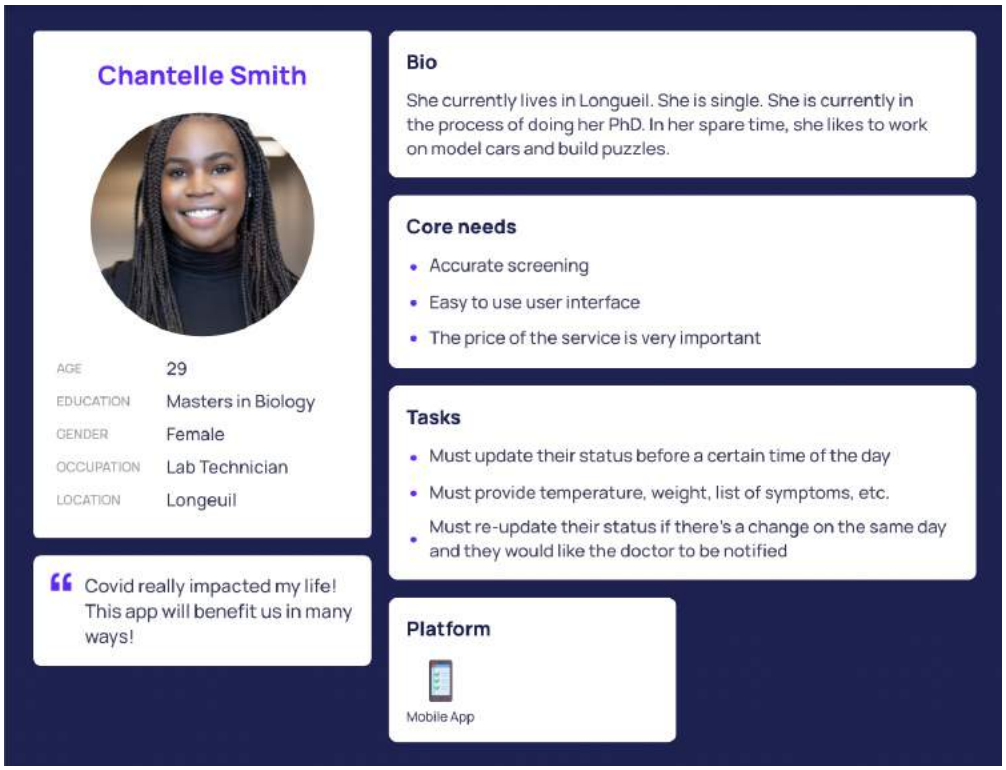


Figure 30: Patient Persona

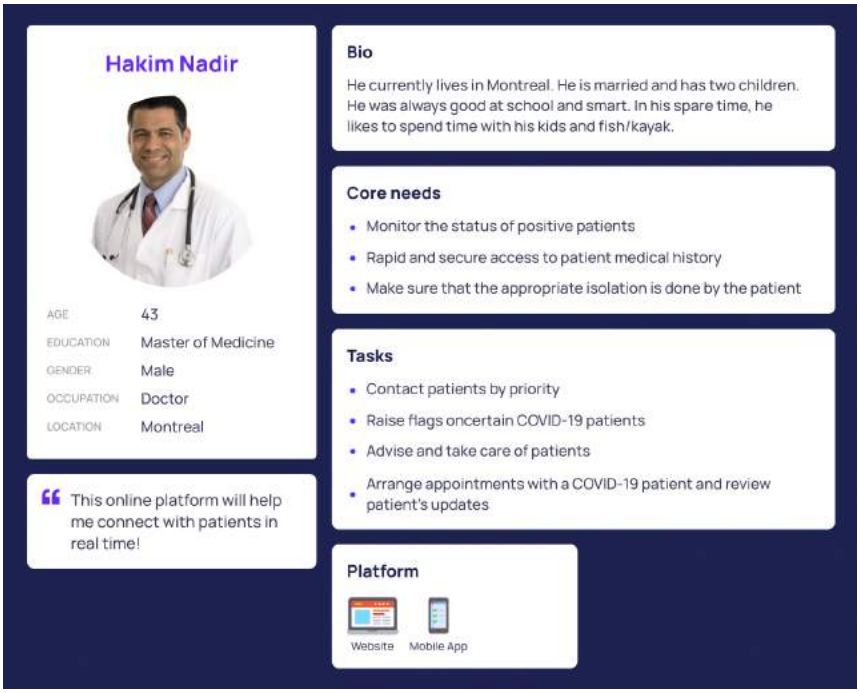


Figure 31: Doctor Persona

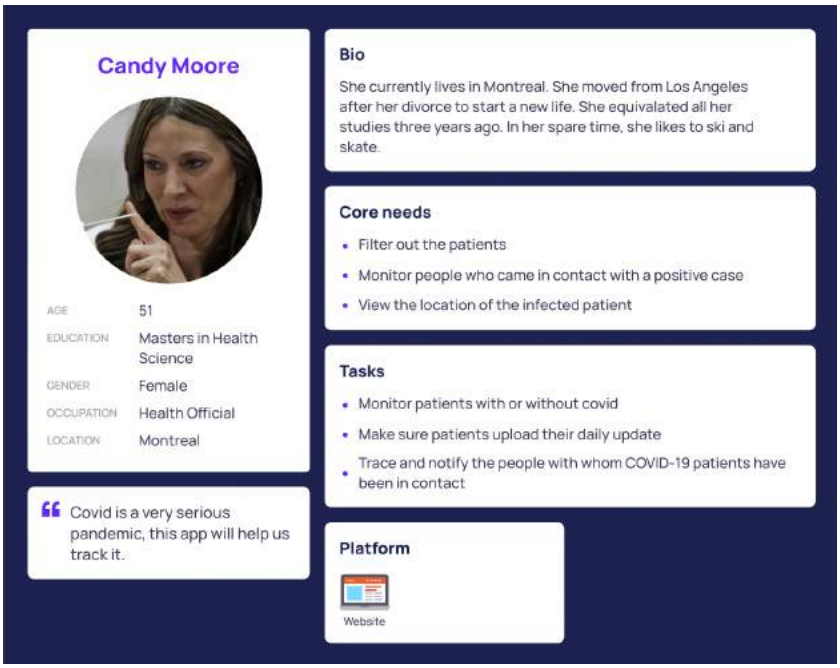


Figure 32: Health Official Persona

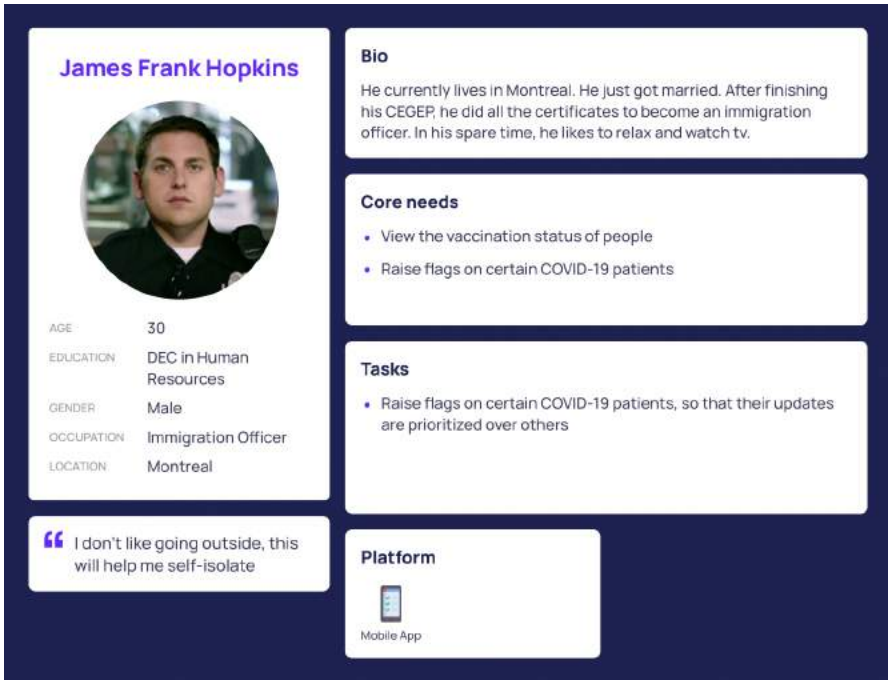


Figure 33: Immigration Officer Persona

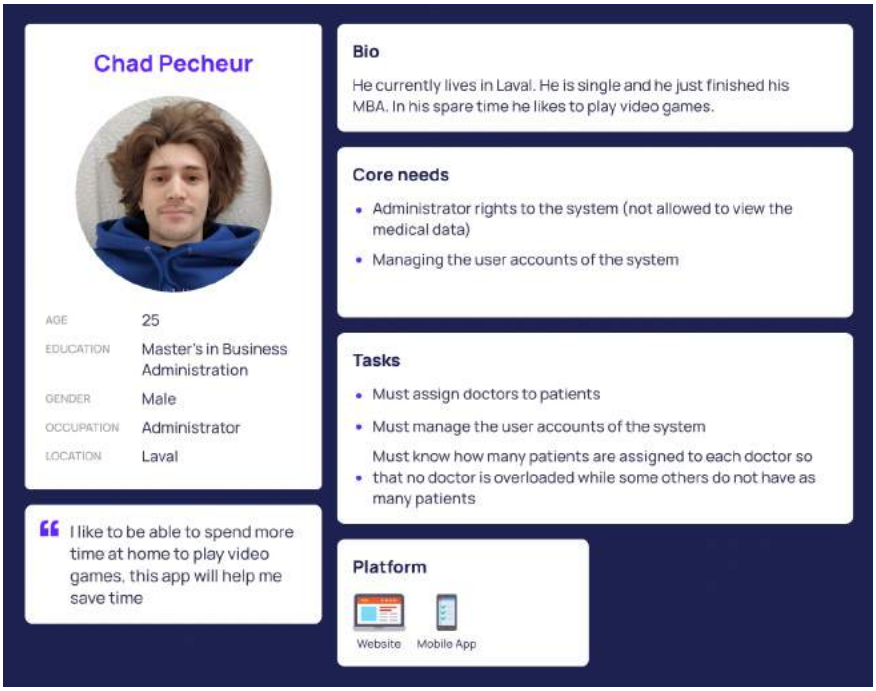


Figure 34: Administrator Persona

7.2 Supported Devices

CovidTracker currently supports desktop and mobile platforms. More specifically, regardless of desktop device, all desktop based web browsers are supported. Likewise, regardless of mobile and tablet devices, all associated web browsers are supported. Figures have been provided below describing the various physical and virtual interface elements present on some of the supported devices.

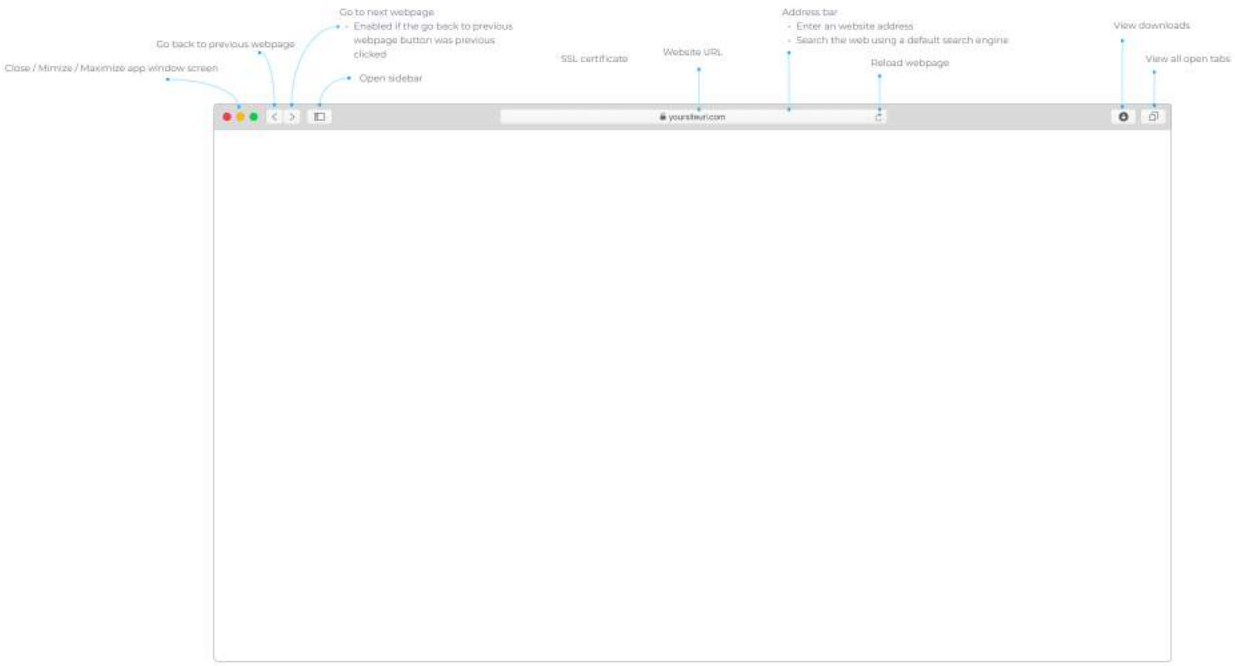


Figure 35: Safari Web Browser Interface Elements

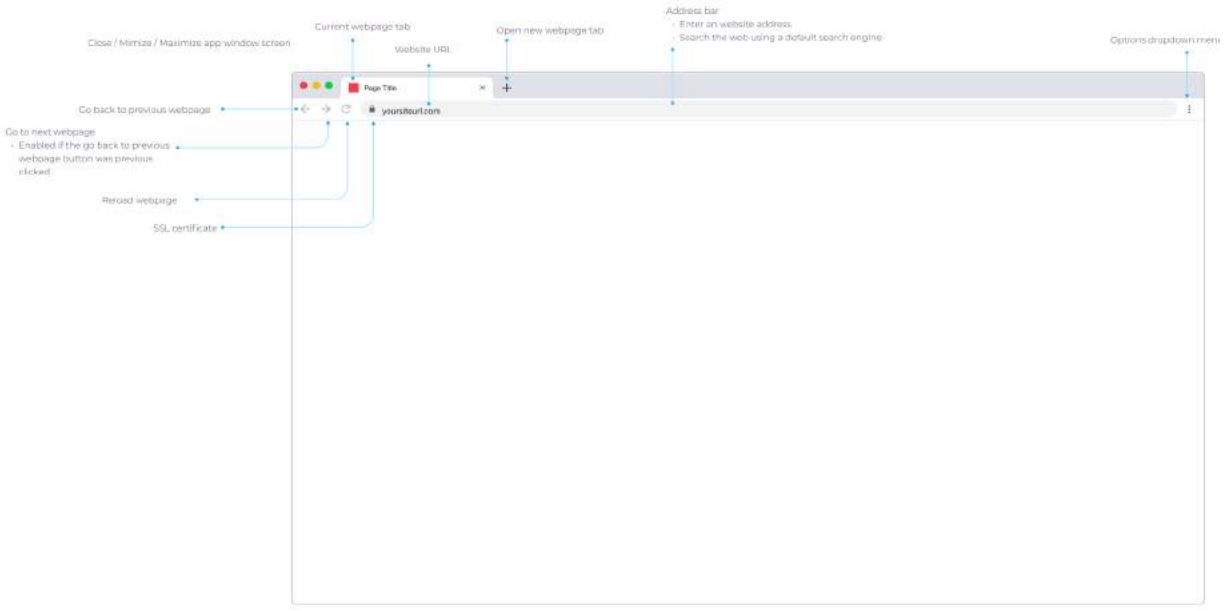


Figure 36: Google Chrome Web Browser Interface Elements

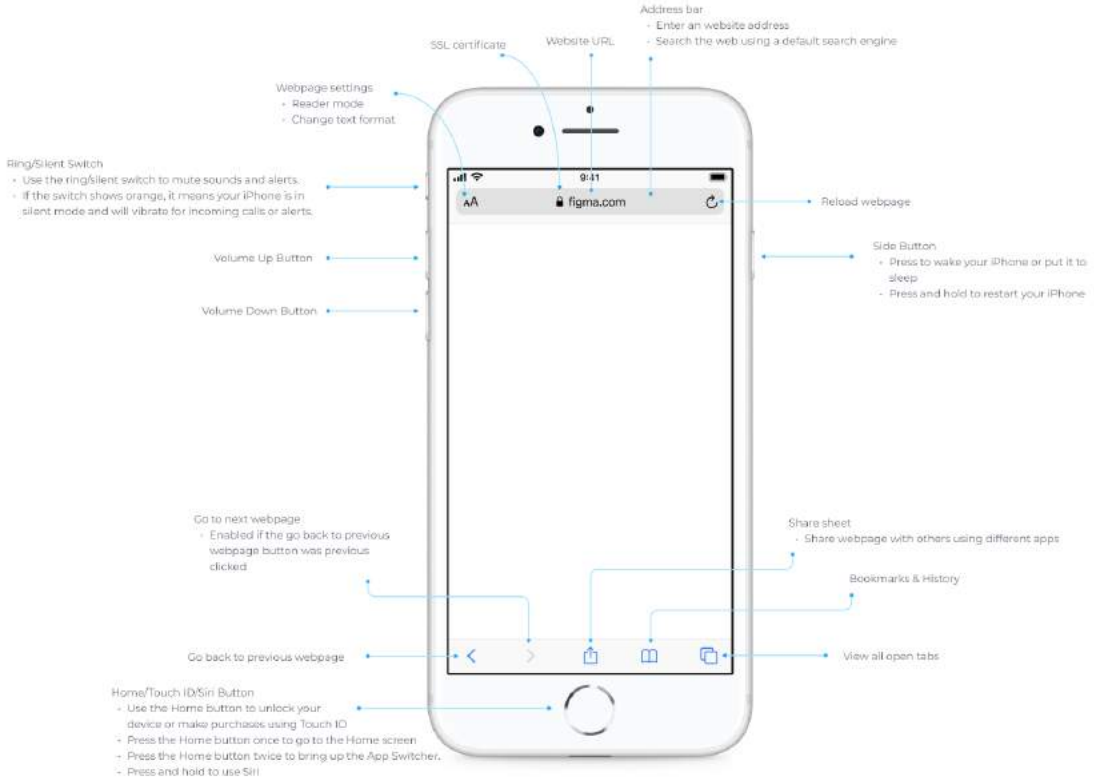


Figure 37: Apple iPhone 8 Buttons and Safari Web Browser Interface Elements

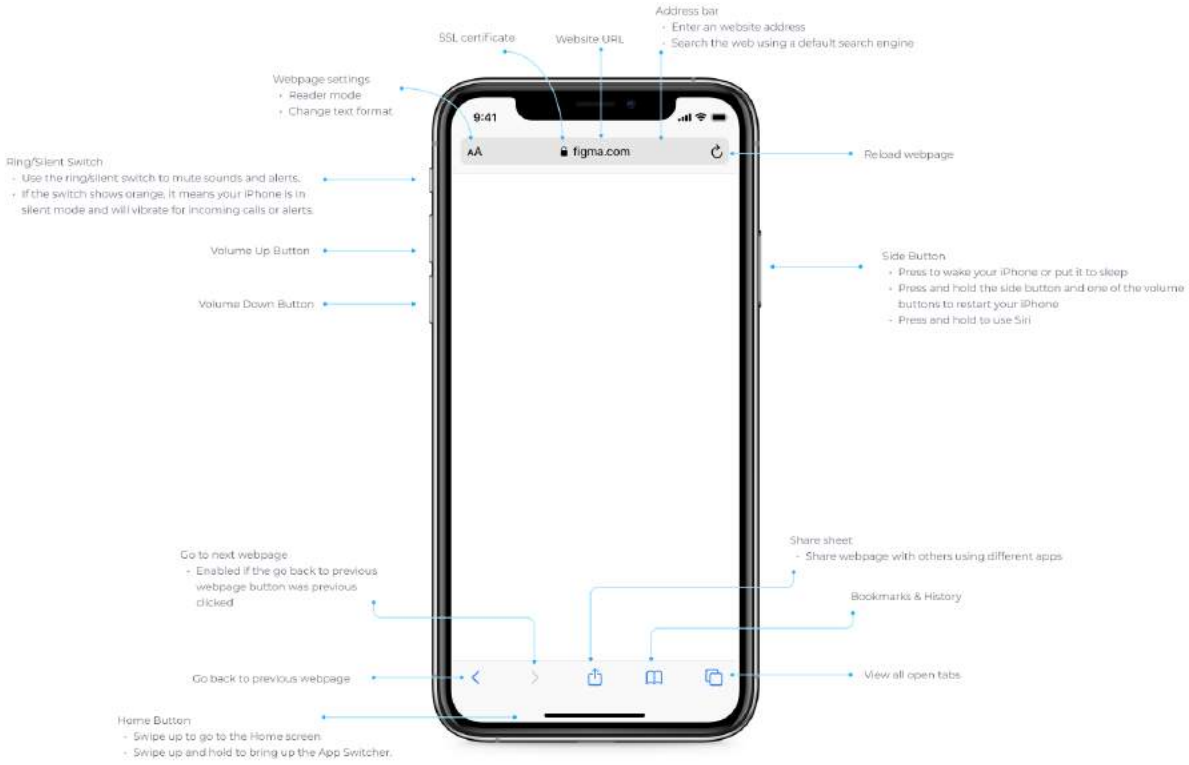


Figure 38: Apple iPhone 11 Buttons and Safari Web Browser Interface Elements

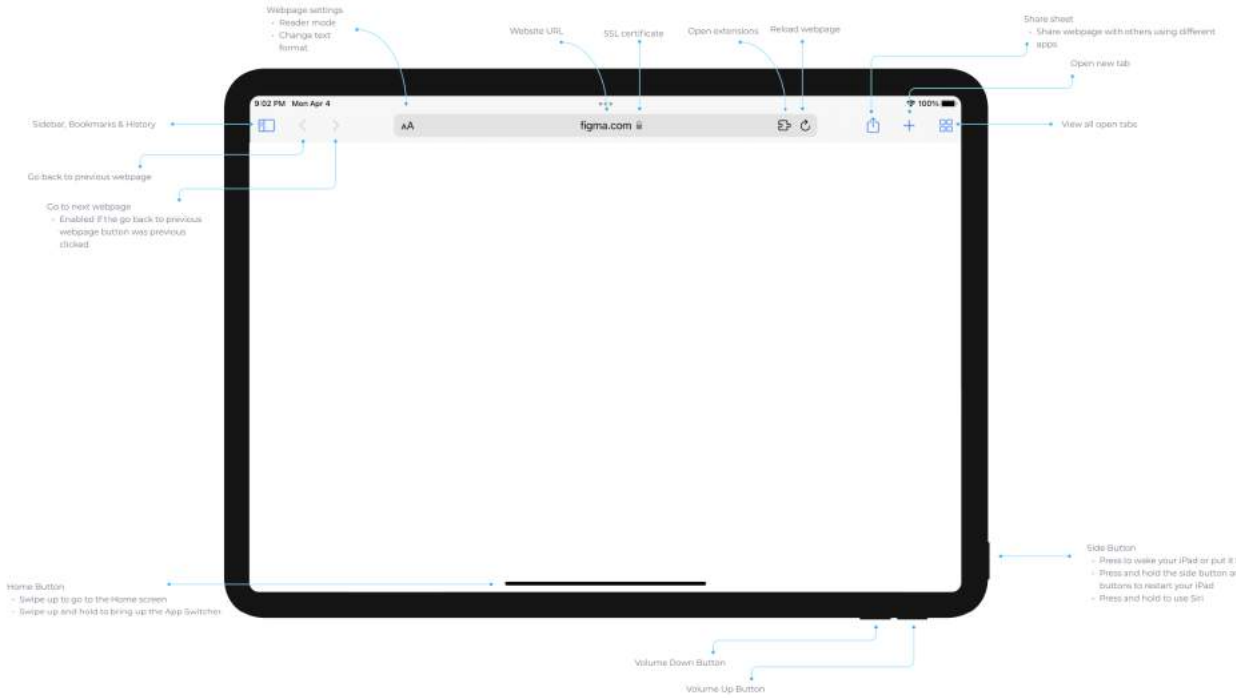


Figure 39: Apple iPad Pro Buttons and Safari Web Browser Interface Elements

7.3 UI Mockups and Prototypes

All UI mockups and associated interactive prototypes are created in Figma. The Figma is organized with the following pages: Components, Personas, Supported Devices, Research, Drafts and UI. The Components page contains all reusable UI elements - logo, form elements, buttons, etc. - which designers might need to use when designing the various mockups. The Personas page contains all the personas information, as discussed in section 7.1 Personas. The Supported Devices page contains information about the various supported devices the application currently supports, as discussed in section 7.2 Supported Devices. The Research page is where various website links, ideas and snippets that one might have come across reside for possible future reference. The Drafts page contains UI mockups or elements that were either discarded or partially worked on. The UI page contains the finalized UI mockups and their associated interactive prototype.

A set of user interface (UI) mockups and interactive prototypes are created for each corresponding user story. The mockups are broken down into groups based on the platform they represent - desktop, tablet and mobile - resulting in a platform specific accessible and ease of use user experience. Each set of mockups are organized in the following manner: the first row describes the user flow steps for the associated user story and each subsequent row below represents various states a particular interface in any given column can have. User flow steps proceed from left to right (start to finish) while each child mockup in a given column can depict one of the following states: active, filled or error. Subsequently, once all the mockups are completed, an interactive prototype is created.

7.3.1 Sign Up

COV-42 - As a User, I want to be able to sign up, so that I can access the apps features

An active user account is required to interact with all features in CovidTracker. As such, if the user does not have an account, they must first sign up to create an account. The sign up page is reached by a clear visible link at the bottom of the sign in page. There are two separate steps that must be completed during the sign up process: a user must fill in their personal information and secondly account information. Context awareness is provided to the user by way of a wizard at the top of the form highlighting the associated icon and text corresponding to the step the user is currently in during the sign up process. Such an element also helps users have a clear visible guideline regarding how many steps are required to be completed. The UI does not adjust based on persona. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 40 and 41. All UI mockups, user flow and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Sign Up / Desktop & Tablet
- UI and User Flow Mockup - Sign Up / Mobile
- ▸ Prototype - Sign Up / Desktop & Tablet
- ▸ Prototype - Sign Up / Mobile

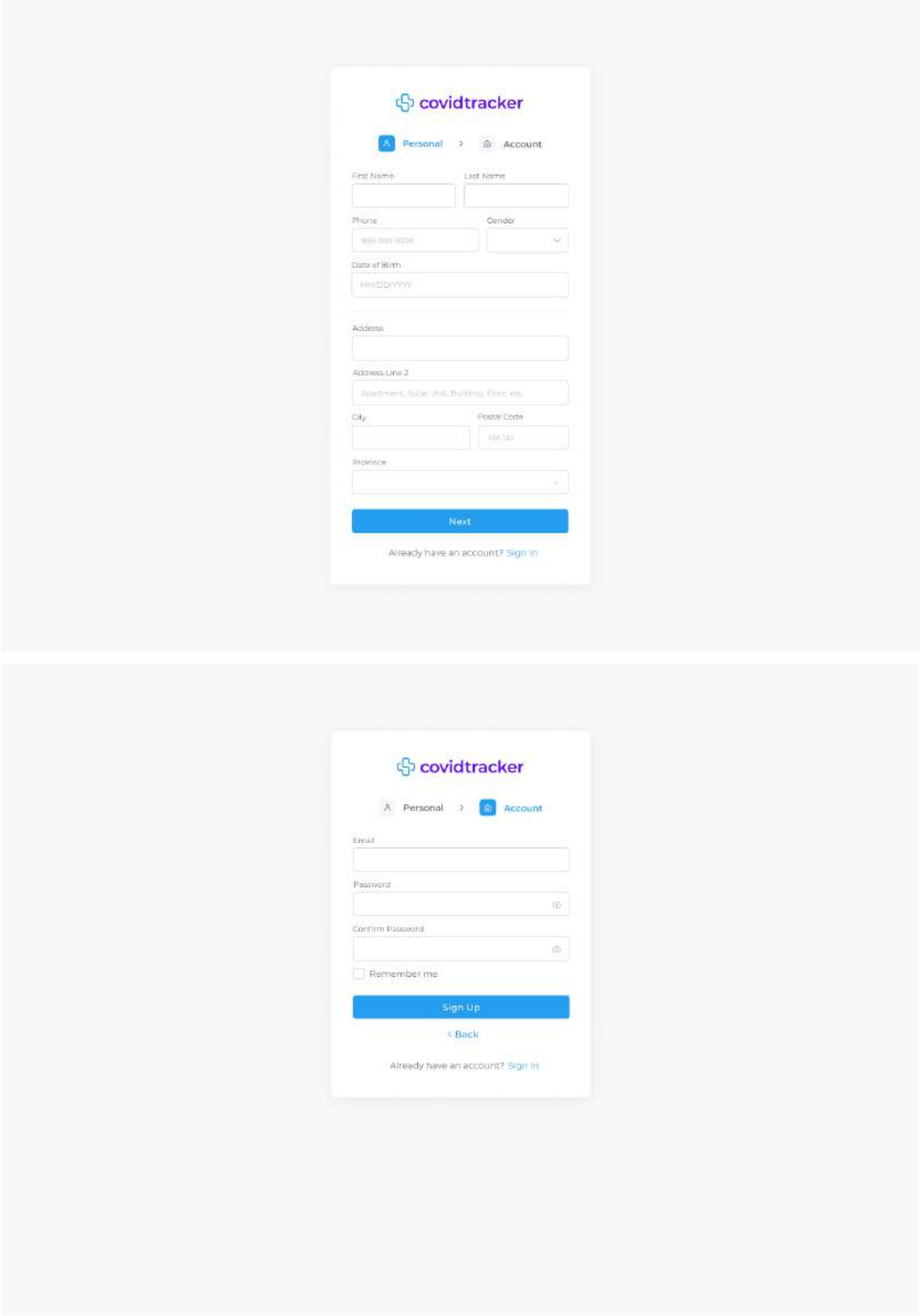


Figure 40: Sign Up Desktop & Tablet UI Mockup

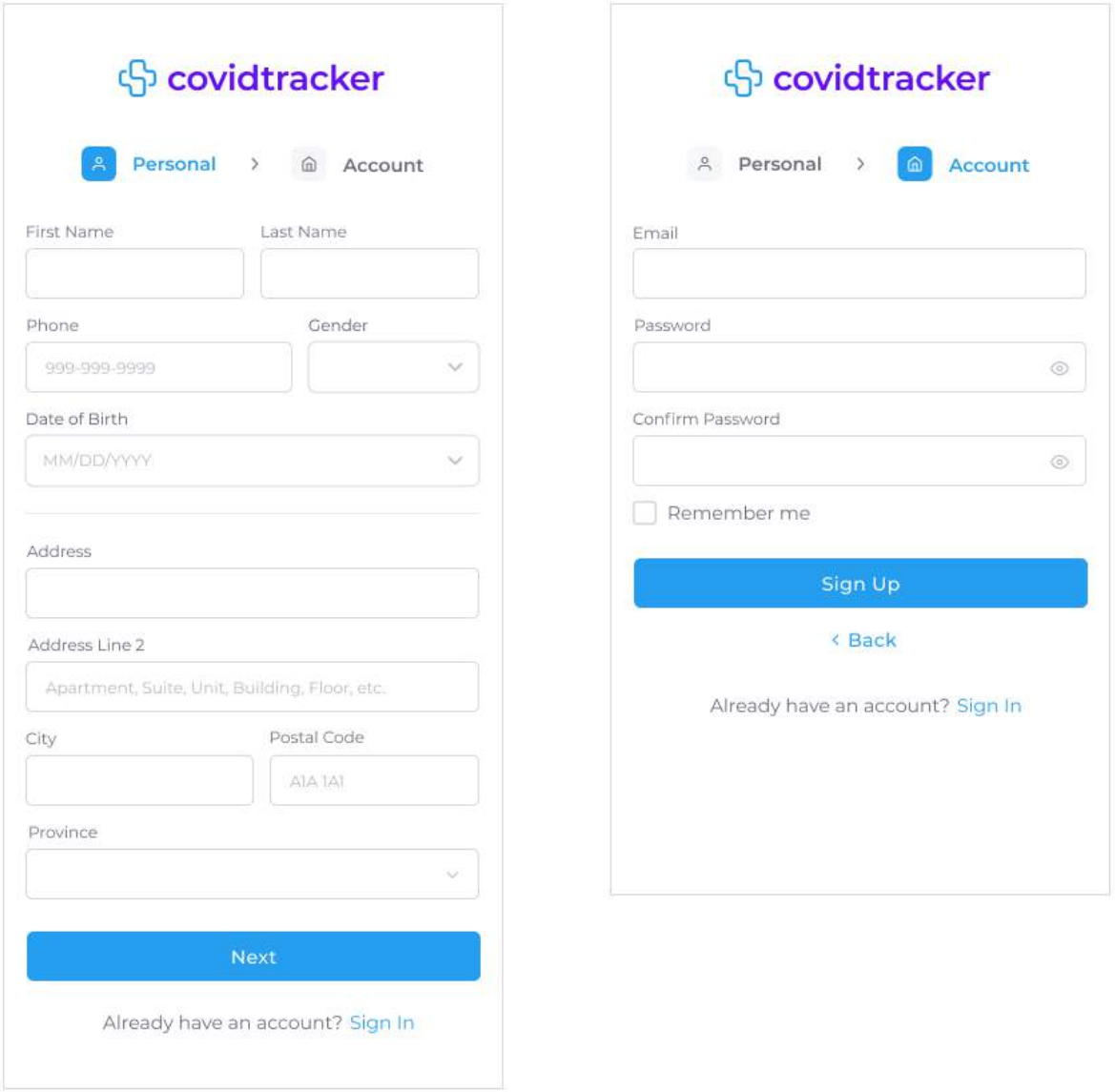


Figure 41: Sign Up Mobile UI Mockup

7.3.2 Sign In

COV-48 - As a User, I want to be able to sign in, so that I can access my account

A non logged in user is automatically redirected to the sign in page when trying to access the CovidTracker website. The UI does not adjust based on persona. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 42 and 43. All UI mockups, user flow and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Sign In / Desktop & Tablet
- UI and User Flow Mockup - Sign In / Mobile
- ▸ Prototype - Sign In / Desktop & Tablet
- ▸ Prototype - Sign In / Mobile

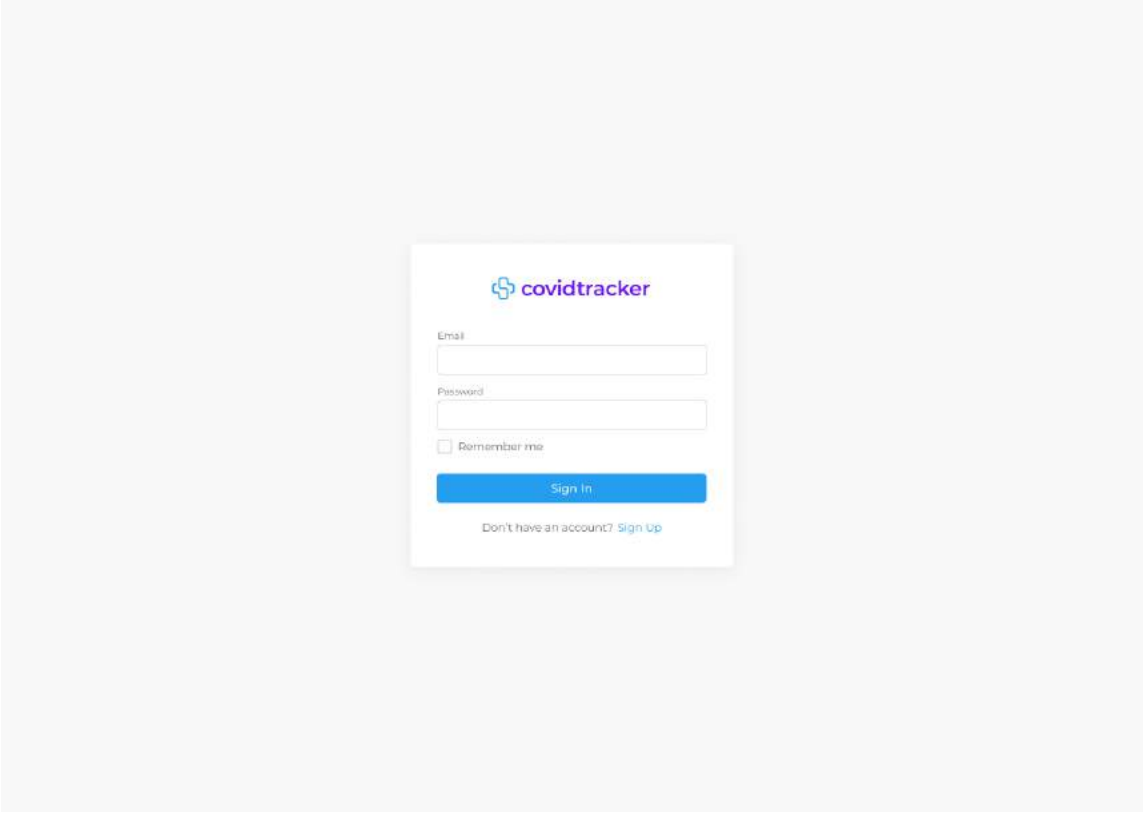


Figure 42: Sign In Desktop & Tablet UI Mockup

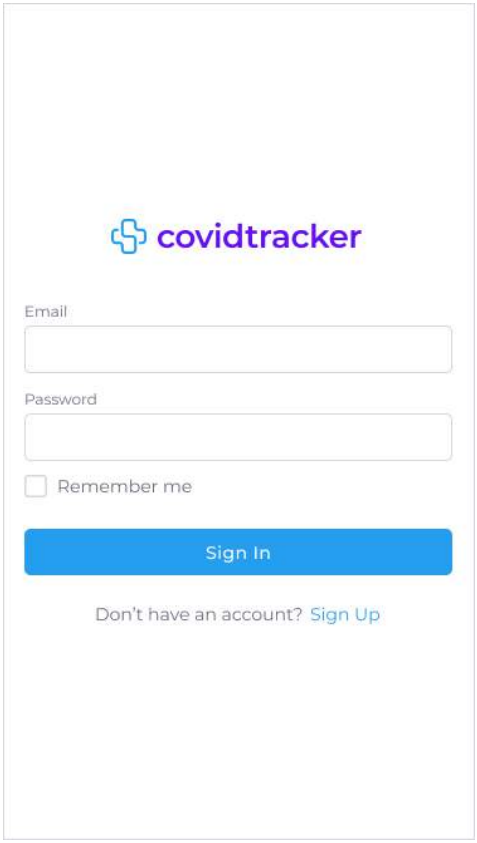


Figure 43: Sign In Mobile UI Mockup

7.3.3 Sign Out

COV-52 - As a User, I want to be able to sign out, so that I can delete my session

A user is only able to sign out of their account if they are currently signed in. The UI does not adjust based on persona. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 44 and 45. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Sign Out / Desktop & Tablet
- UI and User Flow Mockup - Sign Out / Mobile
- ▸ Prototype - Sign Out / Desktop & Tablet
- ▸ Prototype - Sign Out / Mobile

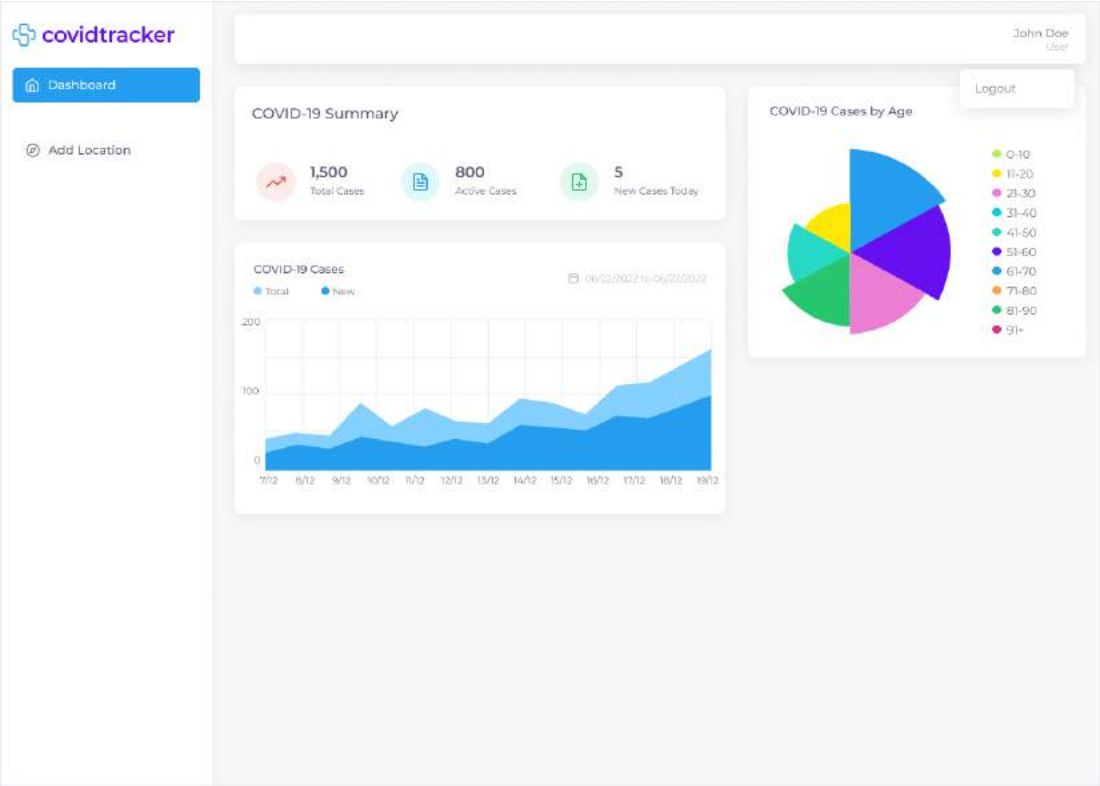


Figure 44: Sign Out Desktop & Tablet UI Mockup

Figure 45: Sign Out Mobile UI Mockup

7.3.4 Add a Role

COV-85 - As an Administrator, I want to assign a role to a user, so that I can manage access rights

An administrator is able to assign a role to a user using their user id. A user can be assigned one of the following roles: Patient, Doctor, Health Official, Immigration Officer, or Administrator. Once assigned a role, a user has access to certain functionalities deemed appropriate for said role in CovidTracker. A user cannot be assigned more than one role at any time. The UI is only accessible by the Administrator persona and does not adjust based on persona. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 46 and 47. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Add a Role / Desktop & Tablet
- UI and User Flow Mockup - Add a Role / Mobile
- ▸ Prototype - Add a Role / Desktop & Tablet
- ▸ Prototype - Add a Role / Mobile

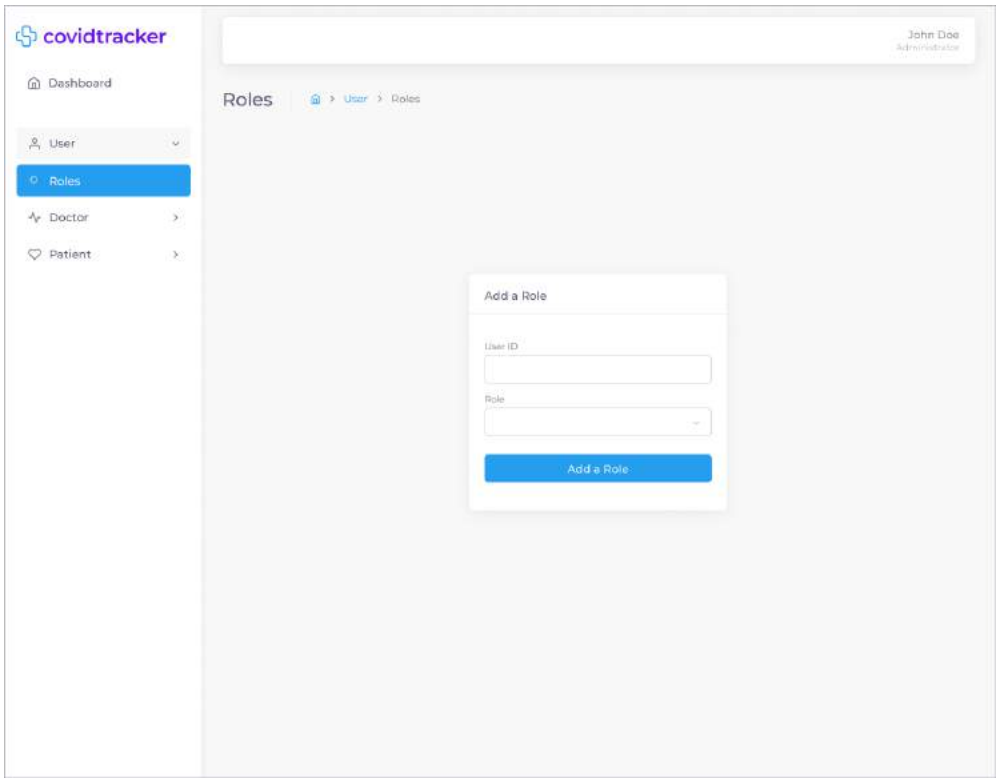


Figure 46: Add a Role Desktop & Tablet UI Mockup

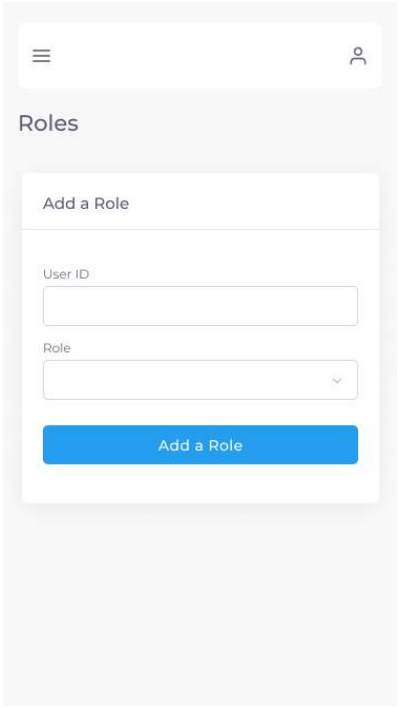


Figure 47: Add a Role Mobile UI Mockup

7.3.5 Assign Patient to Doctor

COV-26 - As an Administrator, I want to assign a Patient to a Doctor, so that I can manage the Patients

An administrator is able to assign a patient to a doctor using both the patient and doctor ids, respectively. A patient can only be assigned to a single doctor at a given time. The UI is only accessible by the Administrator persona and does not adjust based on persona. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 48 and 49. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Assign Patient to Doctor / Desktop & Tablet
- UI and User Flow Mockup - Assign Patient to Doctor / Mobile
- ▸ Prototype - Assign Patient to Doctor / Desktop & Tablet
- ▸ Prototype - Assign Patient to Doctor / Mobile

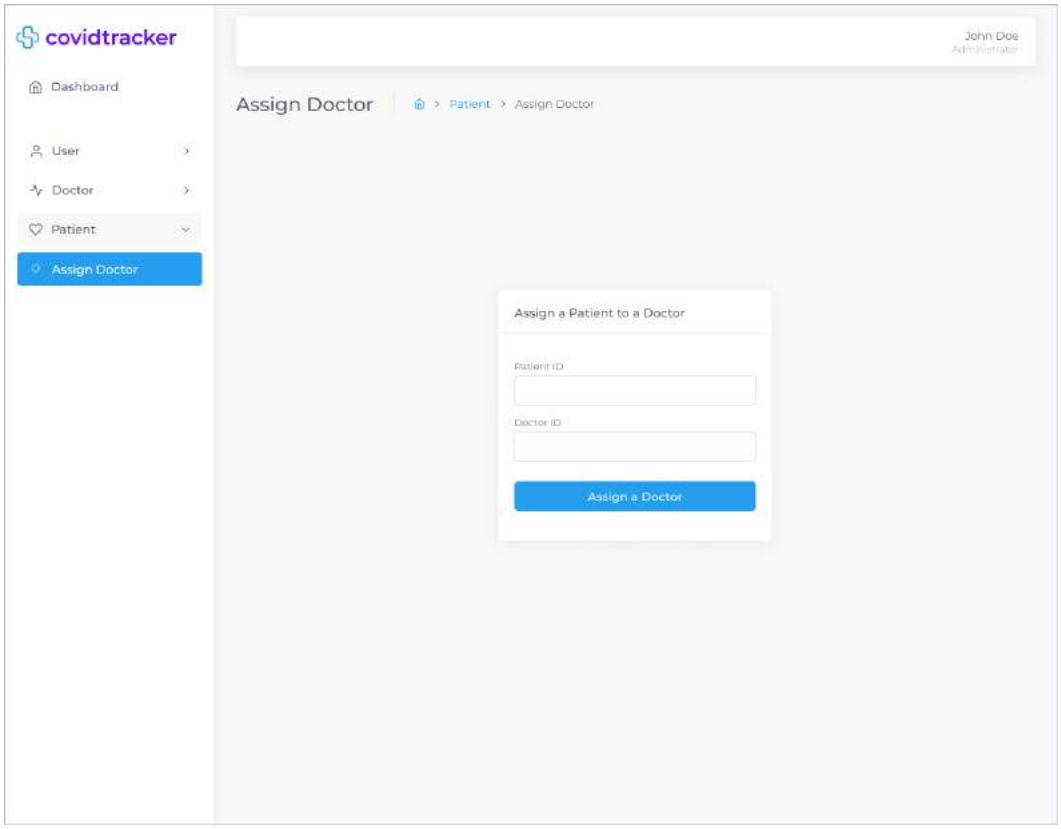


Figure 48: Assign Patient to Doctor Desktop & Tablet UI Mockup

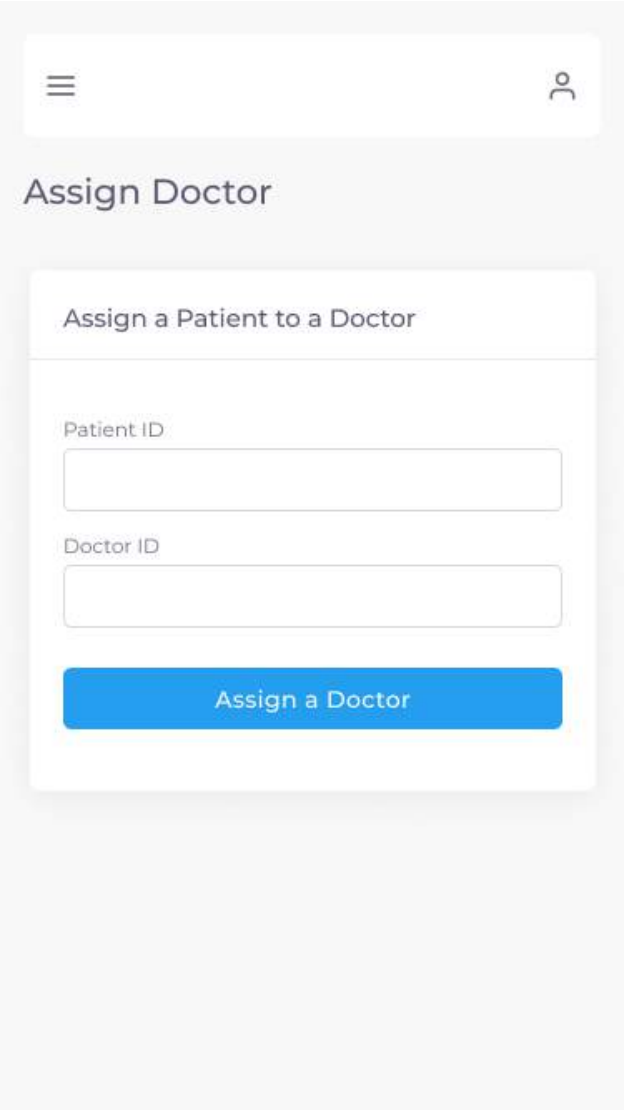


Figure 49: Assign Patient to Doctor Mobile UI Mockup

7.3.6 Define Status Report

COV-95 - As a Doctor, I want to define the status report fields for my Patients, so I can properly track them

A doctor is able to assign a unique status report to each assigned patient. The “Define Status Report” form allows a doctor to choose specific status report fields which must be filled up by the assigned patient of said report. The fields are classified as either general or symptoms (primary and secondary). General fields are pre-selected for the doctor as they are mandatory whereas the symptom fields are up to the doctors discretion. A status report is unique for each assigned patient. In other words, certain fields might not need to be filled up by certain patients compared to others. This page can be accessed from the Patient List by selecting the “Defined Status Report” option found in the more options dropdown for a given patient as described in section 7.3.9 Patient List. The UI is only accessible by the Doctor persona and does not adjust based on persona. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 50 and 51. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Define Status Report / Desktop & Tablet
- UI and User Flow Mockup - Define Status Report / Mobile
- ▸ Prototype - Define Status Report / Desktop & Tablet
- ▸ Prototype - Define Status Report / Mobile

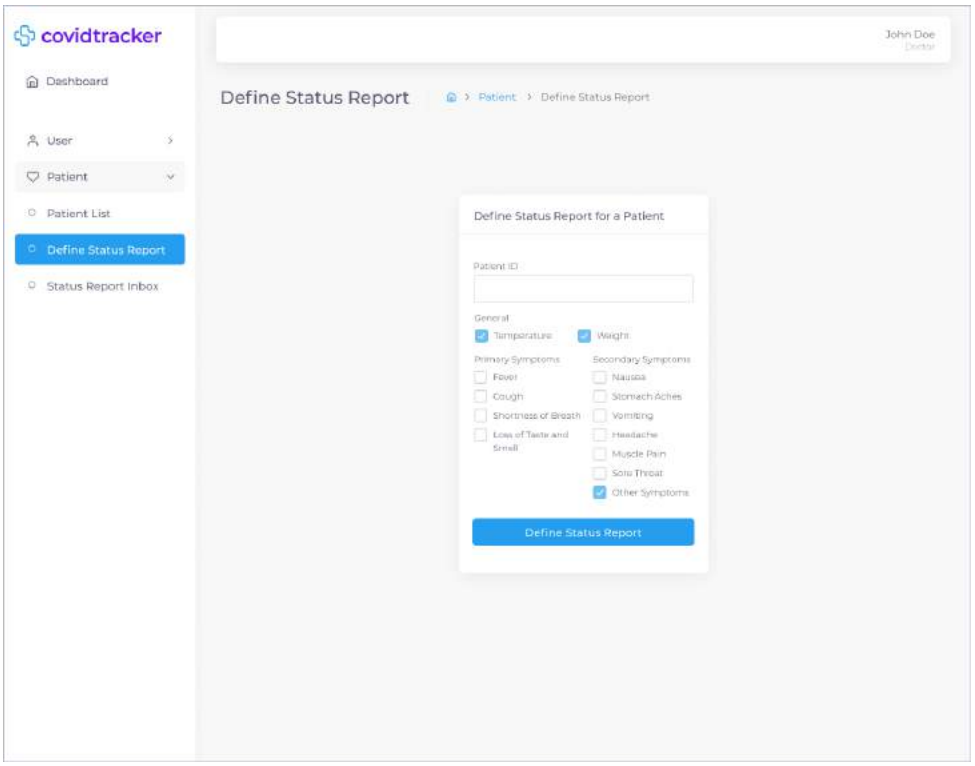


Figure 50: Define Status Report Desktop & Tablet UI Mockup

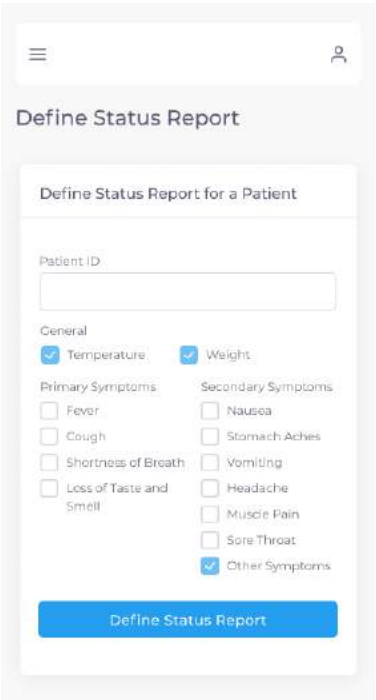


Figure 51: Define Status Report Mobile UI Mockup

7.3.7 Status Report

COV-25 - As a Patient, I want to submit my status, so that I can keep my Doctor updated

Once a patient is assigned a status report by their doctor due to a positive test result, they must fill up their report daily until no longer required to. The status report form fields displayed are adjusted per patient based on the chosen options defined by the doctor as described in section 7.3.6 Define Status Report. Therefore, the status report that a patient must fill out will only display the fields the doctor selects. For example, if a doctor deems secondary symptoms as unnecessary to be filled up by a patient then said patient will not see that option in their daily status report. The UI is only accessible by the Patient persona. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 52 and 53. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Status Report / Desktop & Tablet
- UI and User Flow Mockup - Status Report / Mobile
- ▸ Prototype - Status Report / Desktop & Tablet
- ▸ Prototype - Status Report / Mobile

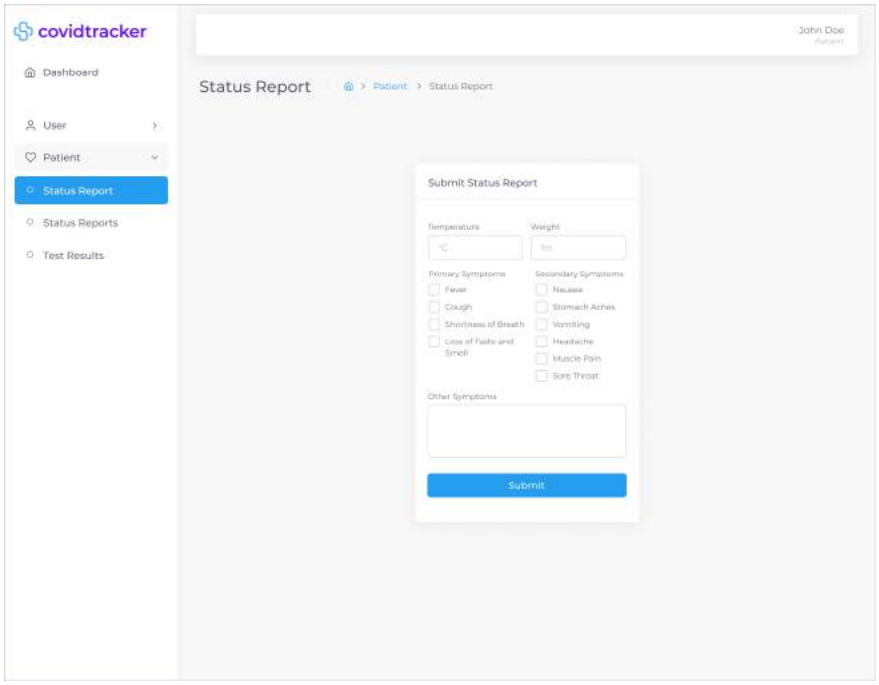


Figure 52: Status Report Desktop & Tablet UI Mockup

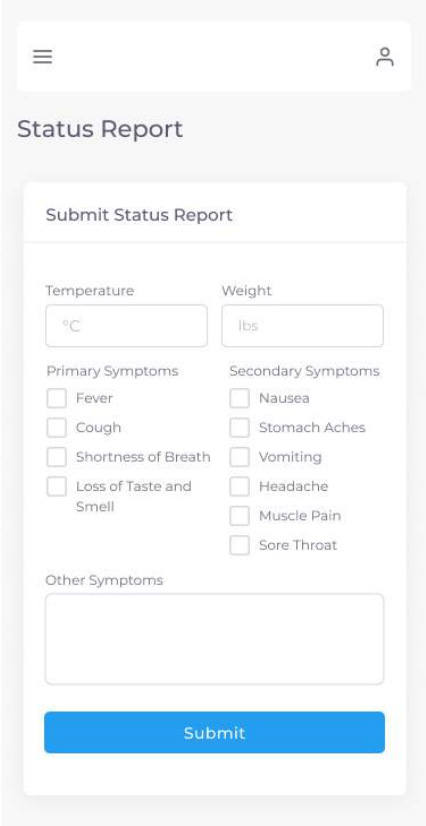


Figure 53: Status Report Mobile UI Mockup

7.3.8 Number of Patients Assigned to a Doctor

COV-27 - As an Administrator, I want to view the number of Patients assigned to a Doctor, so that no Doctor has too many Patients

An administrator is able to view a table containing all doctors and the number of patients assigned to them. There are also two information cards above the table describing the total number of patients assigned to all doctors and the number of assigned patients per doctor. The UI is only accessible by the Administrator persona and does not adjust based on persona. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 54 and 55. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Number of Patients Assigned to a Doctor / Desktop & Tablet
- UI and User Flow Mockup - Number of Patients Assigned to a Doctor / Mobile
- ▸ Prototype - Number of Patients Assigned to a Doctor / Desktop & Tablet
- ▸ Prototype - Number of Patients Assigned to a Doctor / Desktop & Tablet

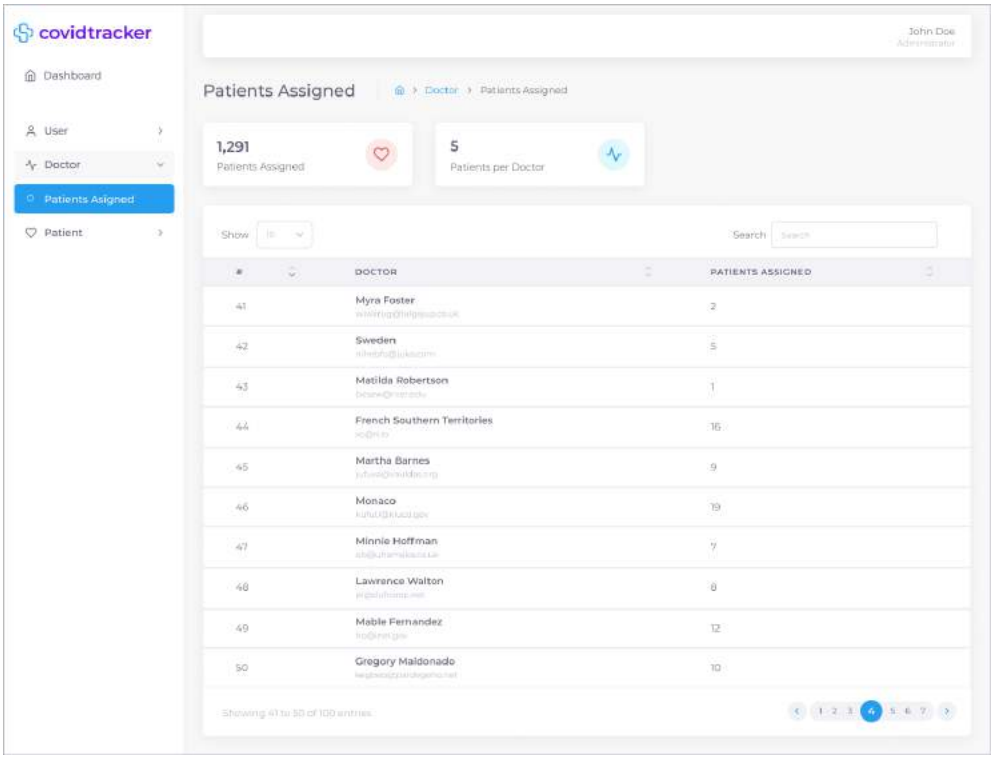


Figure 54: Number of Patients Assigned to a Doctor Desktop & Tablet UI Mockup

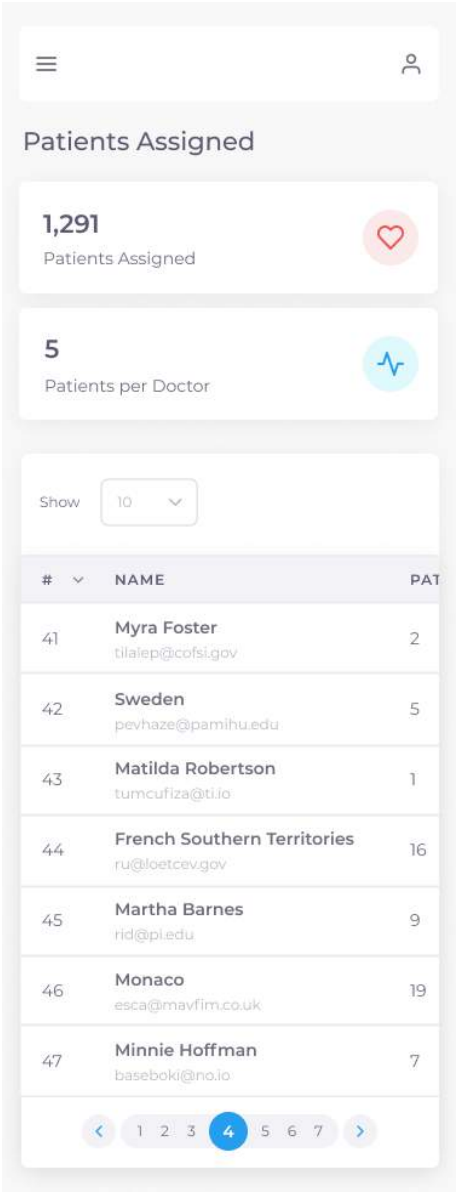


Figure 55: Number of Patients Assigned to a Doctor Mobile UI Mockup

7.3.9 Patient List

COV-114 - As a Doctor, I want to flag certain patients, so that their updates are prioritized over others

COV-157 - As a Doctor, I want to view a list of my Patients, so that I can easily navigate to their specific detailed views

COV-222 - As a Doctor, I want to define status report fields from the patient list page

COV-223 - As an Immigration Officer, I want to view the patients list, so I can be aware of test results and prioritize them

A doctor is able to view a table containing all patients they are assigned to by an administrator. A doctor can prioritize a patient by clicking the “flag” icon under the “Actions” column allowing said patient’s status reports to be prioritized over other patients. A doctor can also click the “more options” (three dots) icon and choose one of the following actions: add a test result for a patient, view a patient’s test results, view a patient's status reports, book an appointment, and define a status report. This page is also accessible to a health official and immigration officer. A health official can have the same abilities as a doctor would, such as prioritization, adding a test result, and viewing all test results and status reports but cannot, however, book an appointment with a patient. An immigration officer can only see patient test results. The only adjustment between all personas is the health official and immigration officer will see a list of all patients in the system and a doctor only sees their assigned patients. Therefore, the UI is only accessible by the Doctor, Health Official, and Immigration Officer personas. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 56 and 57 for the Doctor and Health official personas and Figures 58 and 59 for the Immigration Officer persona. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Patient List (Doctor/Health Official) / Desktop & Tablet
- UI and User Flow Mockup - Patient List (Doctor/Health Official) / Mobile
- ▸ Prototype - Patient List (Doctor/Health Official) / Desktop & Tablet
- ▸ Prototype - Patient List (Doctor/Health Official) / Mobile

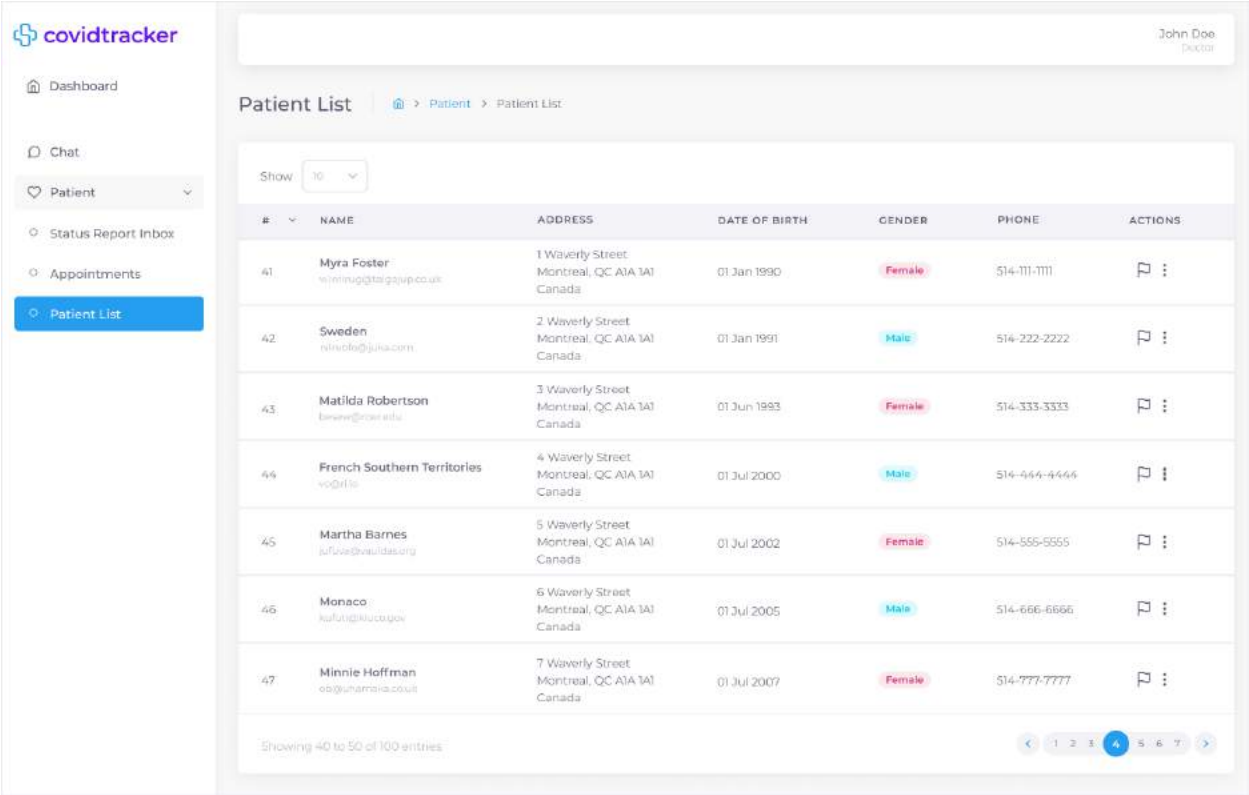


Figure 56: Patient List (Doctor & Health Official) Desktop & Tablet UI Mockup

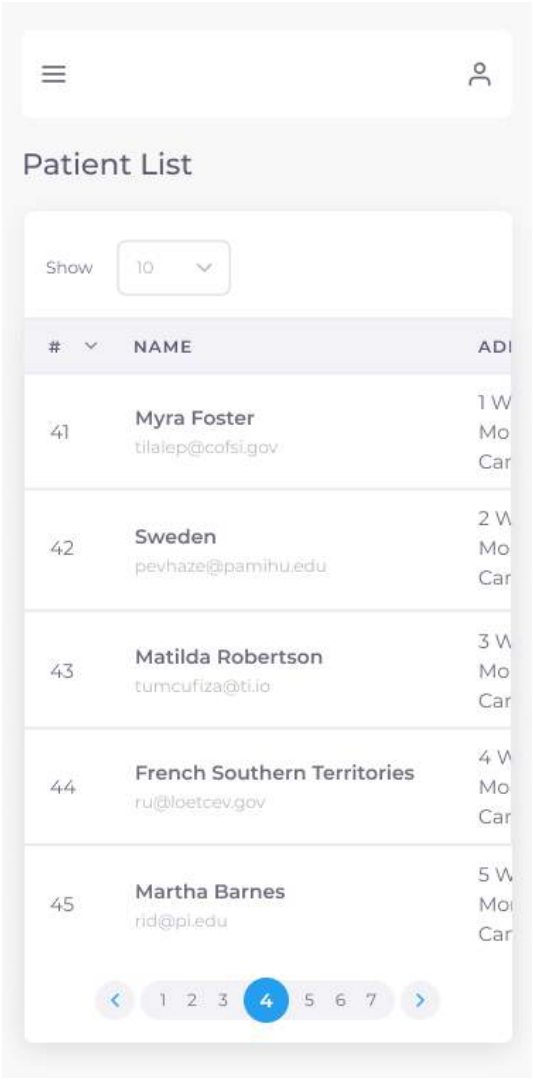


Figure 57: Patient List (Doctor & Health Official) Mobile UI Mockup

- UI and User Flow Mockup - Patient List (Immigration Officer) / Desktop & Tablet
- UI and User Flow Mockup - Patient List (Immigration Officer) / Mobile
- ▸ Prototype - Patient List (Immigration Officer) / Desktop & Tablet
- ▸ Prototype - Patient List (Immigration Officer) / Mobile

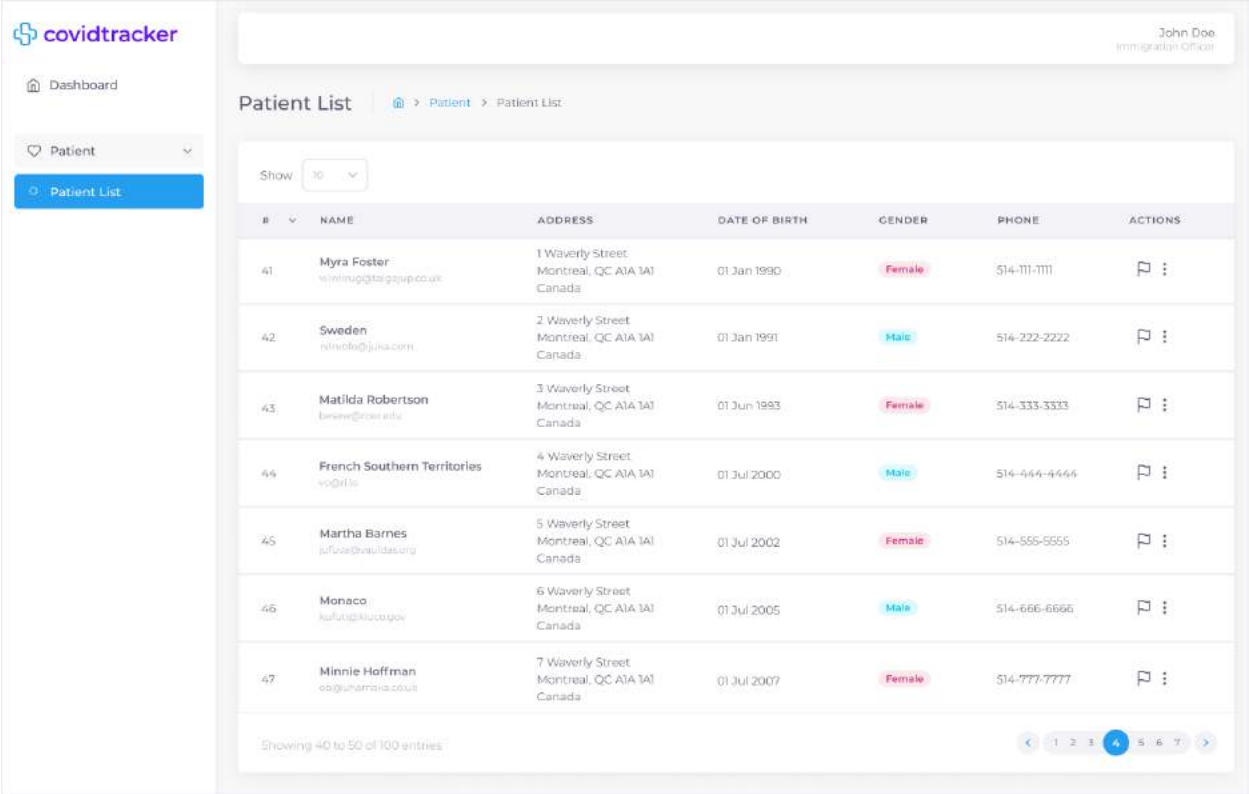


Figure 58: Patient List (Immigration Officer) Desktop & Tablet UI Mockup

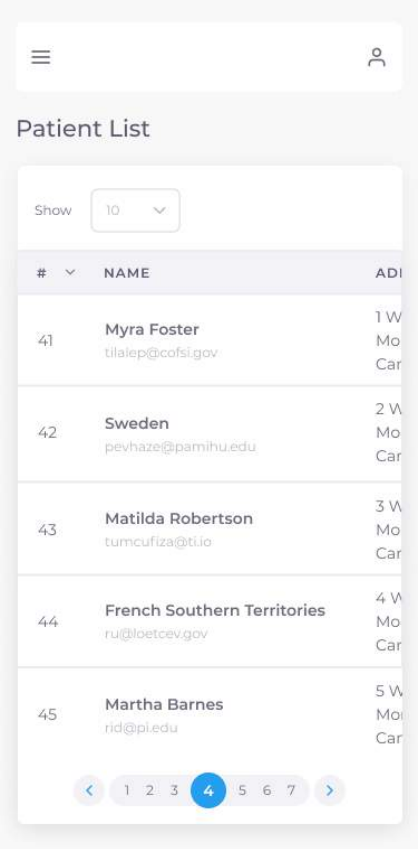


Figure 59: Patient List (Immigration Officer) Mobile UI Mockup

7.3.10 Status Reports

COV-111 - As a Patient, I want to view all my line item statuses, so that I can monitor my progress over time

A patient is able to view a table containing all the status reports submitted to their doctor. By clicking the “see details” (eye) icon under the “Actions” column, a patient would be able to see a full description of the status report as described in section 7.3.13 Status Report Details. A doctor and health official can also view a similar page upon selecting the “Status Reports” option found in the more options dropdown for a given patient within the Patient List page as described in section 7.3.9 Patient List. Therefore, the UI is only accessible by the Patient, Doctor and Health Official personas. The only UI element that adjusts based on the persona is the breadcrumb text. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 60 and 61 for the Patient persona and Figures 62 and 63 for the Doctor and Health Official personas. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Status Reports (Patient) / Desktop & Tablet
- UI and User Flow Mockup - Status Reports (Patient) / Mobile
- ▸ Prototype - Status Reports (Patient) / Desktop & Tablet
- ▸ Prototype - Status Reports (Patient) / Mobile

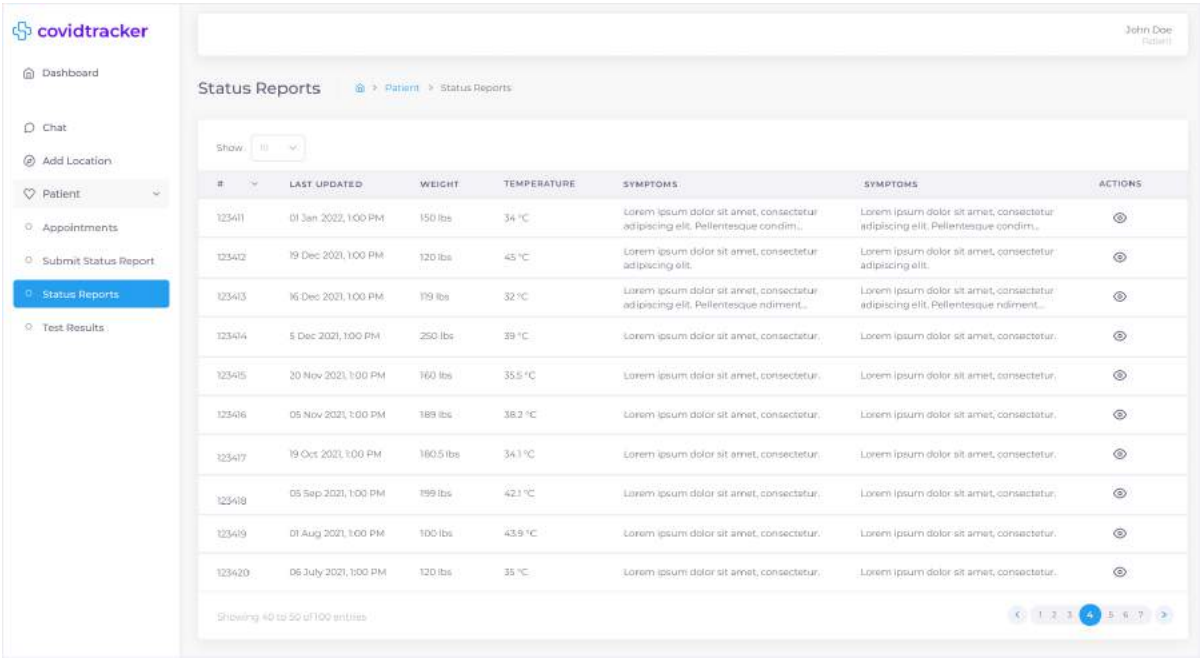


Figure 60: Status Reports (Patient) Desktop & Tablet UI Mockup

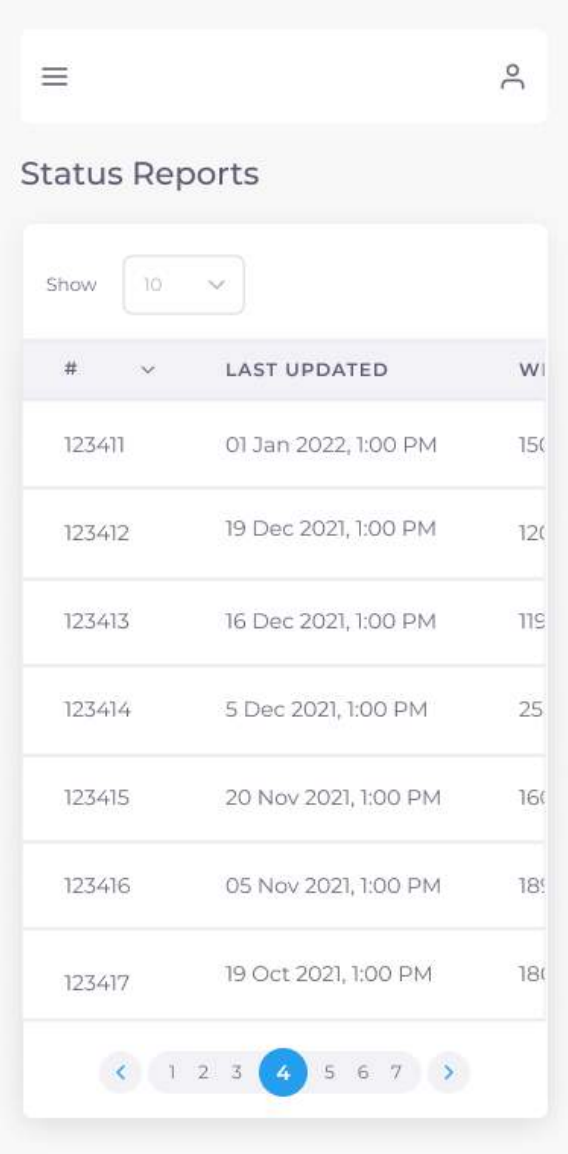


Figure 61: Status Reports (Patient) Mobile UI Mockup

- UI and User Flow Mockup - Status Reports (Doctor/Health Official) / Desktop & Tablet
- UI and User Flow Mockup - Status Reports (Doctor/Health Official) / Mobile
- ▸ Prototype - Status Reports (Doctor/Health Official) / Desktop & Tablet
- ▸ Prototype - Status Reports (Doctor/Health Official) / Mobile

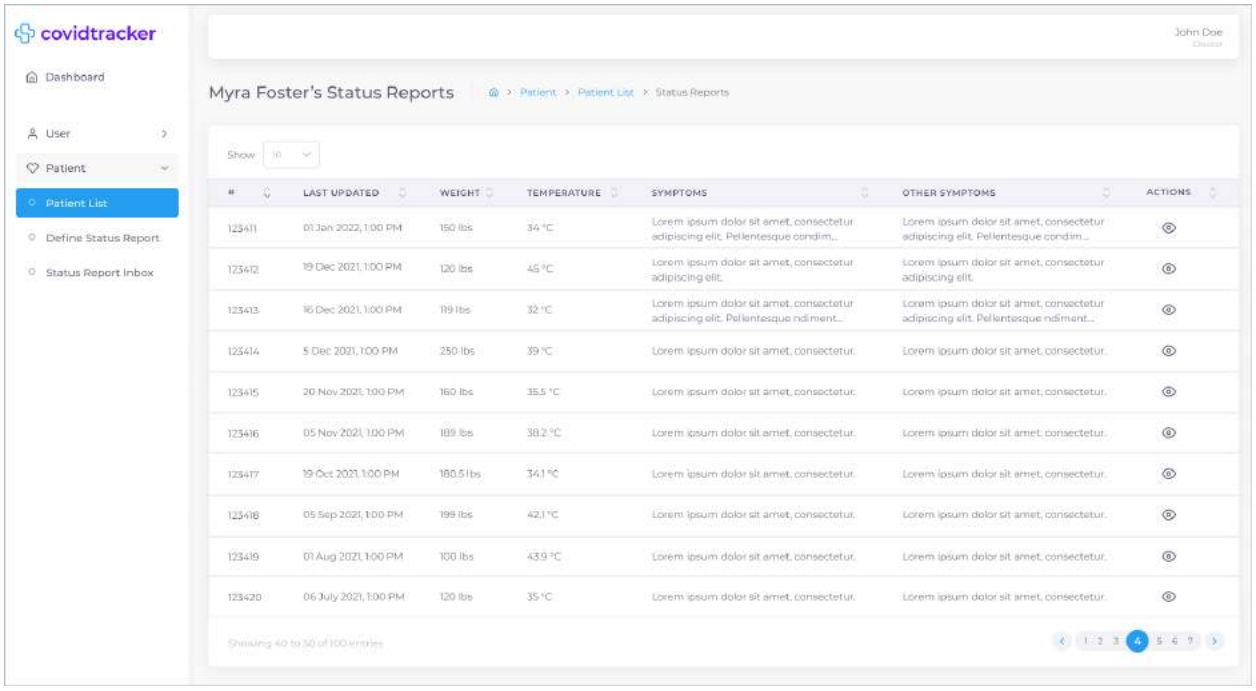


Figure 62: Status Reports (Doctor/Health Official) Desktop & Tablet UI Mockup

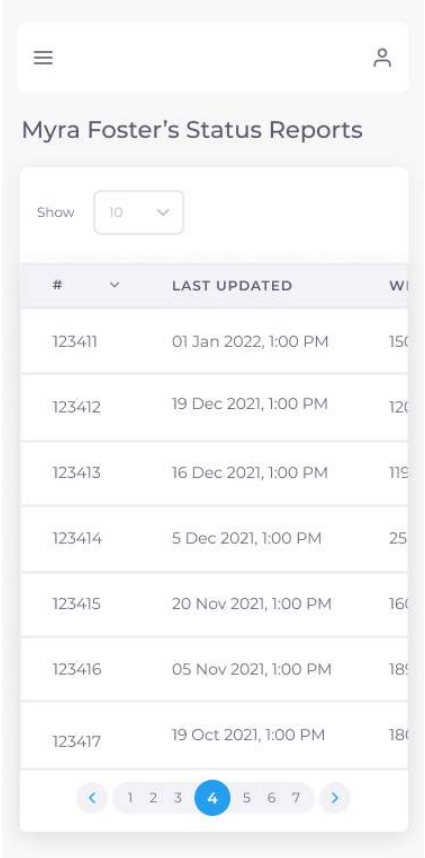


Figure 63: Status Reports (Doctor/Health Official) Mobile UI Mockup

7.3.11 Add Test Result

COV-107 - As a Health Official, I want to input my COVID test results, so that I can report if a Patient tested positive or negative

A health official and doctor are able to add a test result for a given patient. The following information must be provided: test result (positive or negative), type of test (antigen or PCR), date of test, and location of test. This page can be accessed from the Patient List by selecting the “Add Test Result” option found in the more options dropdown for a given patient as described in section 7.3.9 Patient List. The UI is only accessible by the Doctor and Health Official personas and does not adjust based on persona. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 64 and 65. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Add Test Result / Desktop & Tablet
- UI and User Flow Mockup - Add Test Result / Mobile
- ▸ Prototype - Add Test Result / Desktop & Tablet
- ▸ Prototype - Add Test Result / Mobile

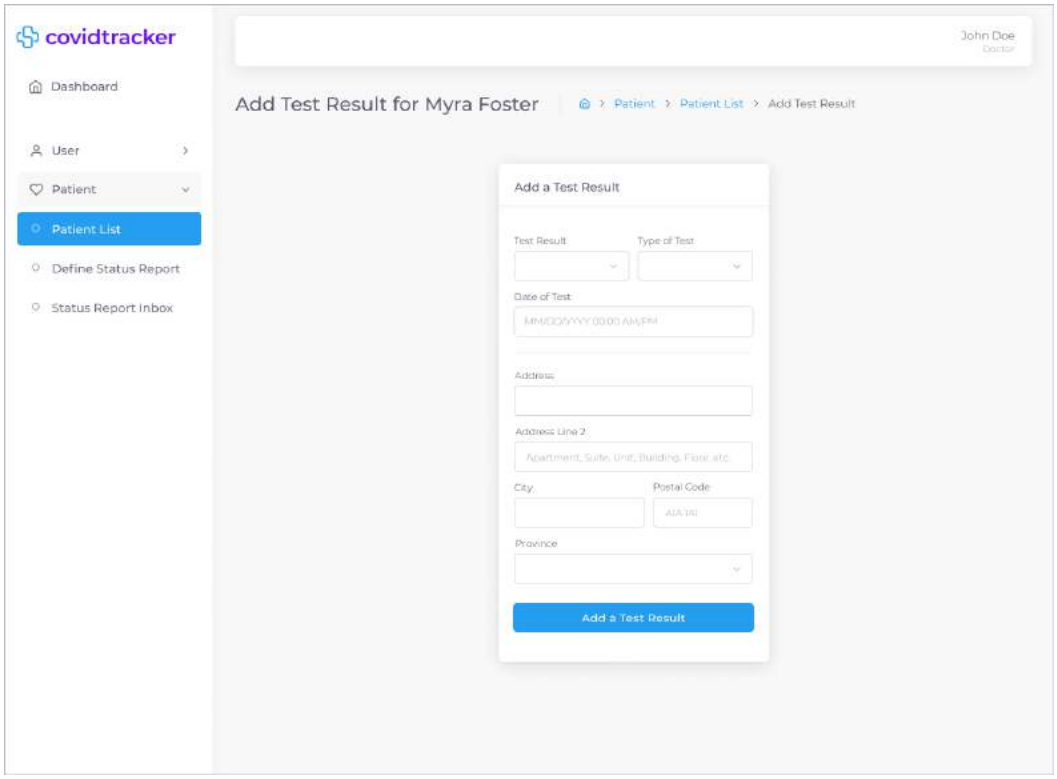


Figure 64: Add Test Result Desktop & Tablet UI Mockup

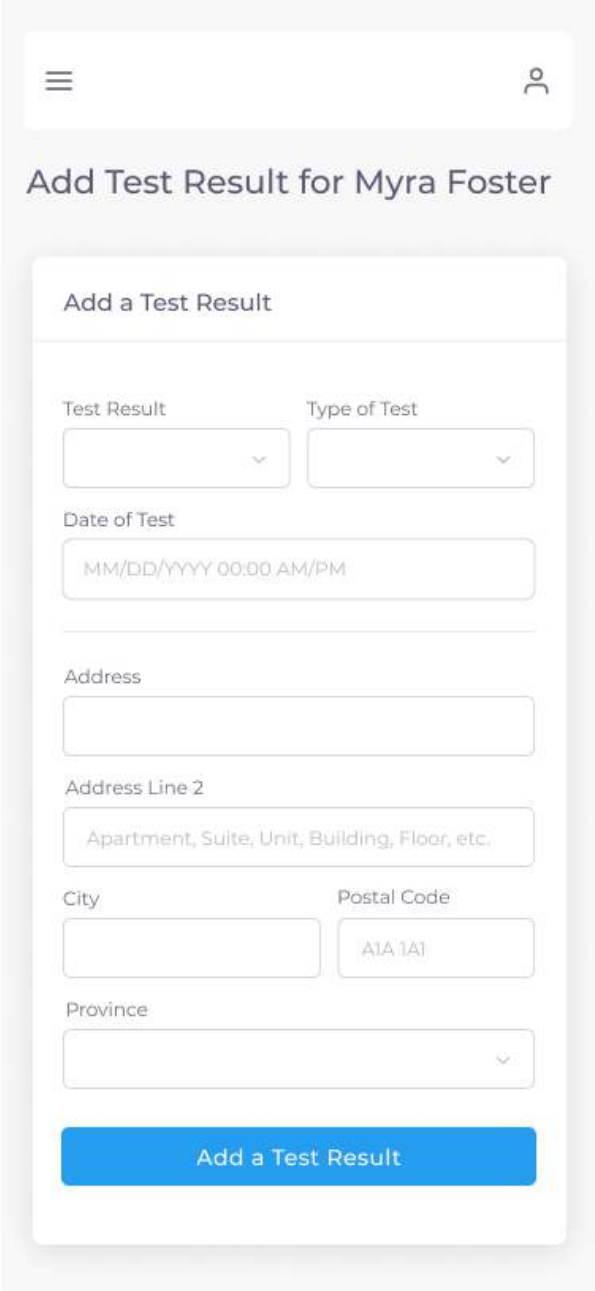


Figure 65: Add Test Result Mobile UI Mockup

7.3.12 Status Report Inbox

COV-113 - As a Doctor, I want to view a line item list of my patients with their most recent line item status update, so that I can keep track of any updates

COV-115 - As a Doctor, I want to mark a Patient's status update as "Reviewed", so that I can see which statuses I've already seen

A doctor is able to view a table containing all status reports submitted by all their assigned patients. A doctor can subsequently, press the checkbox to mark a status report as viewed or uncheck it to mark it as not viewed. A doctor can also click the “see details” (eye) icon under the “Actions” column to view a full description of the status report as described in section 7.3.13 Status Report Details. The UI is only accessible by the Doctor persona and does not adjust based on persona. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 66 and 67. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Status Report Inbox / Desktop & Tablet
- UI and User Flow Mockup - Status Report Inbox / Mobile
- ▸ Prototype - Status Report Inbox / Desktop & Tablet
- ▸ Prototype - Status Report Inbox / Mobile

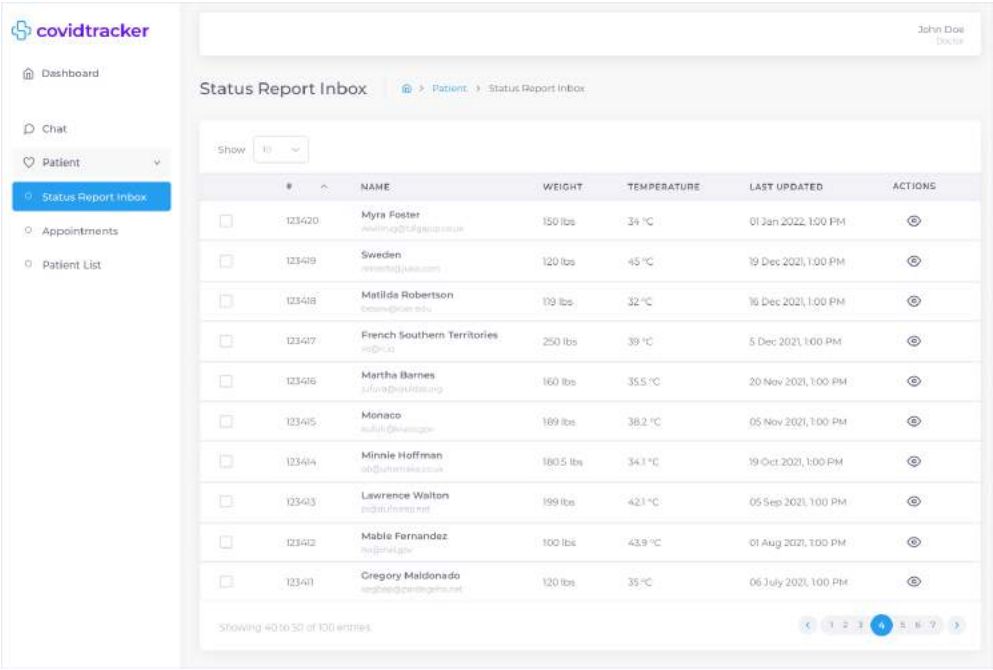


Figure 66: Status Report Inbox Desktop & Tablet UI Mockup

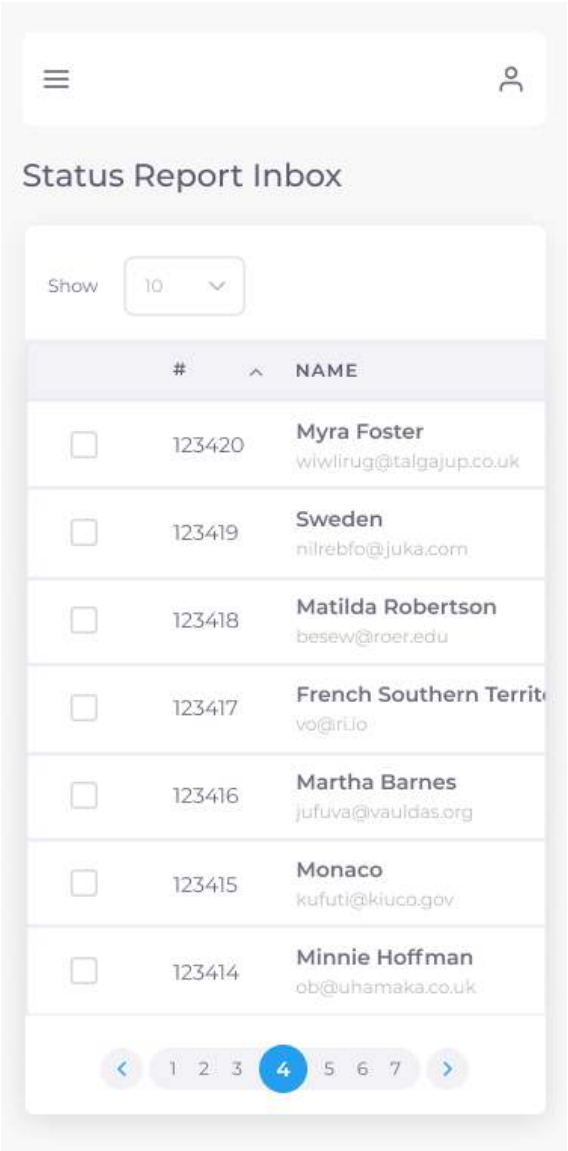


Figure 67: Status Report Inbox Mobile UI Mockup

7.3.13 Status Report Details

COV-112 - As a Patient, I want to view the details of a single status report of a Patient, so that I can view their progress at a point in time

COV-121 - As a Patient, I want to be able to generate a QR code for a status report, so that I can share it with others

A patient is able to view a full detailed description of a given status report and its associated QR code that can be used to easily share such information with either their doctor or a health official. Likewise, upon scanning such QR code, a doctor or health official will be redirected to a similar page. The UI is only accessible by the Patient, Doctor and Health Official personas. The only UI element that adjusts based on the persona is the breadcrumb text. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 68 and 69 for the Patient person and Figures 70 and 71 for the Doctor and Health Official personas. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Status Report Details (Patient) / Desktop & Tablet
- UI and User Flow Mockup - Status Report Details (Patient) / Mobile
- ▸ Prototype - Status Report Details (Patient) / Desktop & Tablet
- ▸ Prototype - Status Report Details (Patient) / Mobile

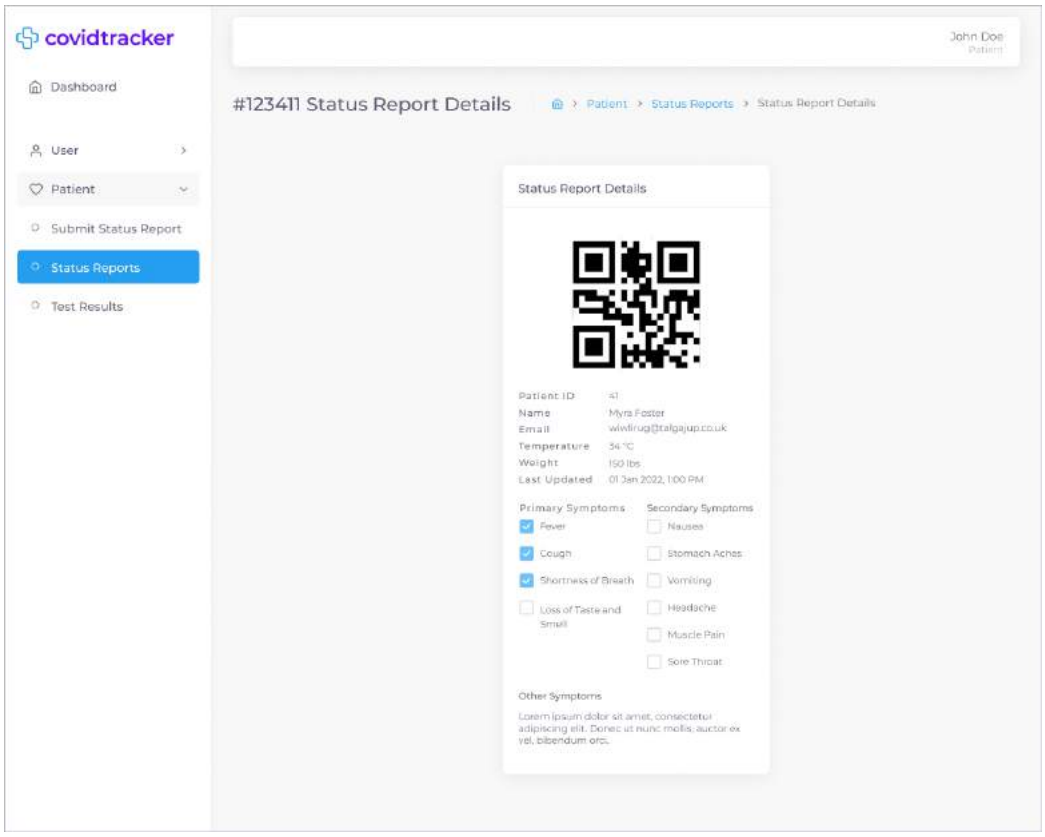


Figure 68: Status Report Details (Patient) Desktop & Tablet UI Mockup

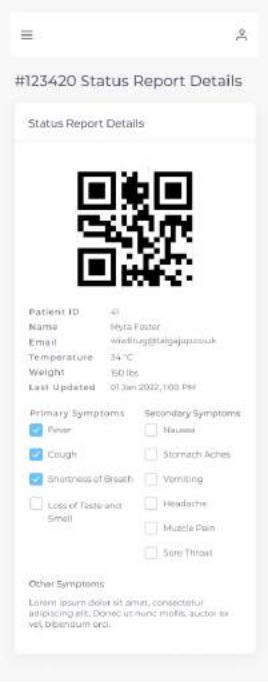


Figure 69: Status Report Details (Patient) Mobile UI Mockup

- UI and User Flow Mockup - Status Report Details (Doctor/Health Official) / Desktop & Tablet
- UI and User Flow Mockup - Status Report Details (Doctor/Health Official) / Mobile
- ▸ Prototype - Status Report Details (Doctor/Health Official) / Desktop & Tablet
- ▸ Prototype - Status Report Details (Doctor/Health Official) / Mobile

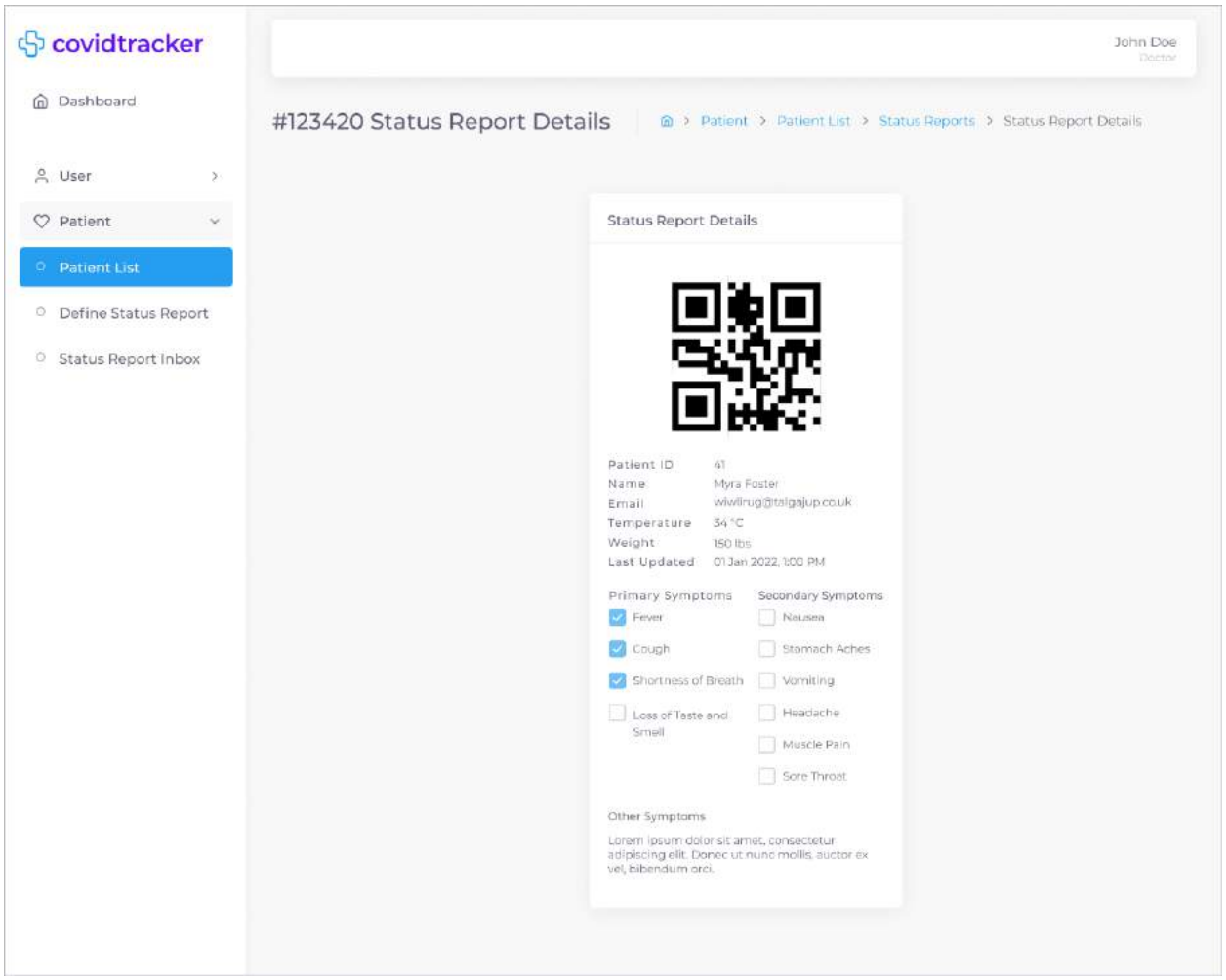


Figure 70: Status Report Details (Doctor/Health Official) Desktop & Tablet UI Mockup

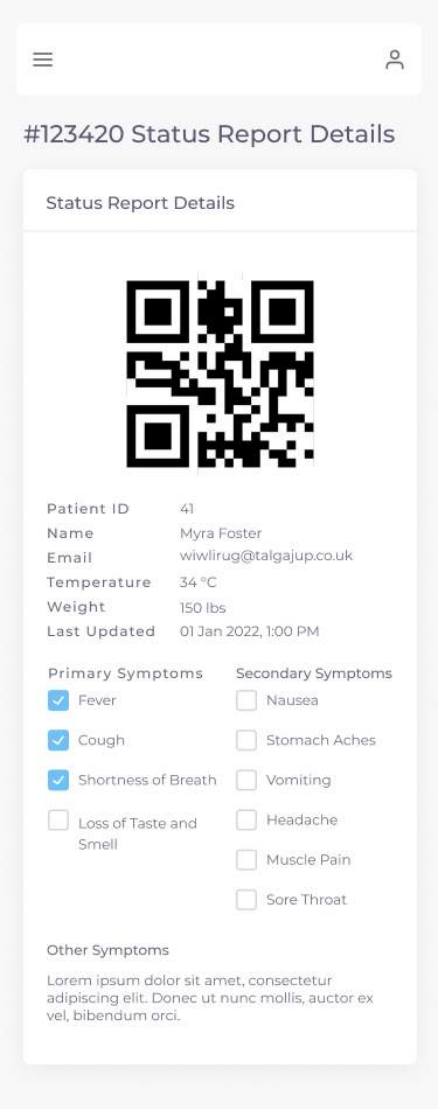


Figure 71: Status Report Details (Doctor/Health Official) Mobile UI Mockup

7.3.14 Test Results

COV-123 - As a Patient, I want to view all my line item COVID test results, so that I'm aware of my diagnosis

COV-223 - As an Immigration Officer, I want to view the patients list, so I can be aware of test results and prioritize them

A patient is able to view a table containing all their test results. By clicking the “see details” (eye) icon under the “Actions” column, a patient would be able to see a full description of the test result as described in section 7.3.15 Test Result Details. A doctor, health official, and immigration can also view a similar page upon selecting the “Test Results” option found in the more options dropdown for a given patient within the Patient List page as described in section 7.3.9 Patient List. Therefore, the UI is only accessible by the Patient, Doctor, Health Official, and Immigration Officer personas. The only UI element that adjusts based on the persona is the breadcrumb text. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 72 and 73 for the Patient person, Figures 74 and 75 for the Doctor and Health Official personas, and Figures 76 and 77 for the Immigration Officer persona. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Test Results (Patient) / Desktop & Tablet
- UI and User Flow Mockup - Test Results (Patient) / Mobile
- ▸ Prototype - Test Results (Patient) / Desktop & Tablet
- ▸ Prototype - Test Results (Patient) / Mobile

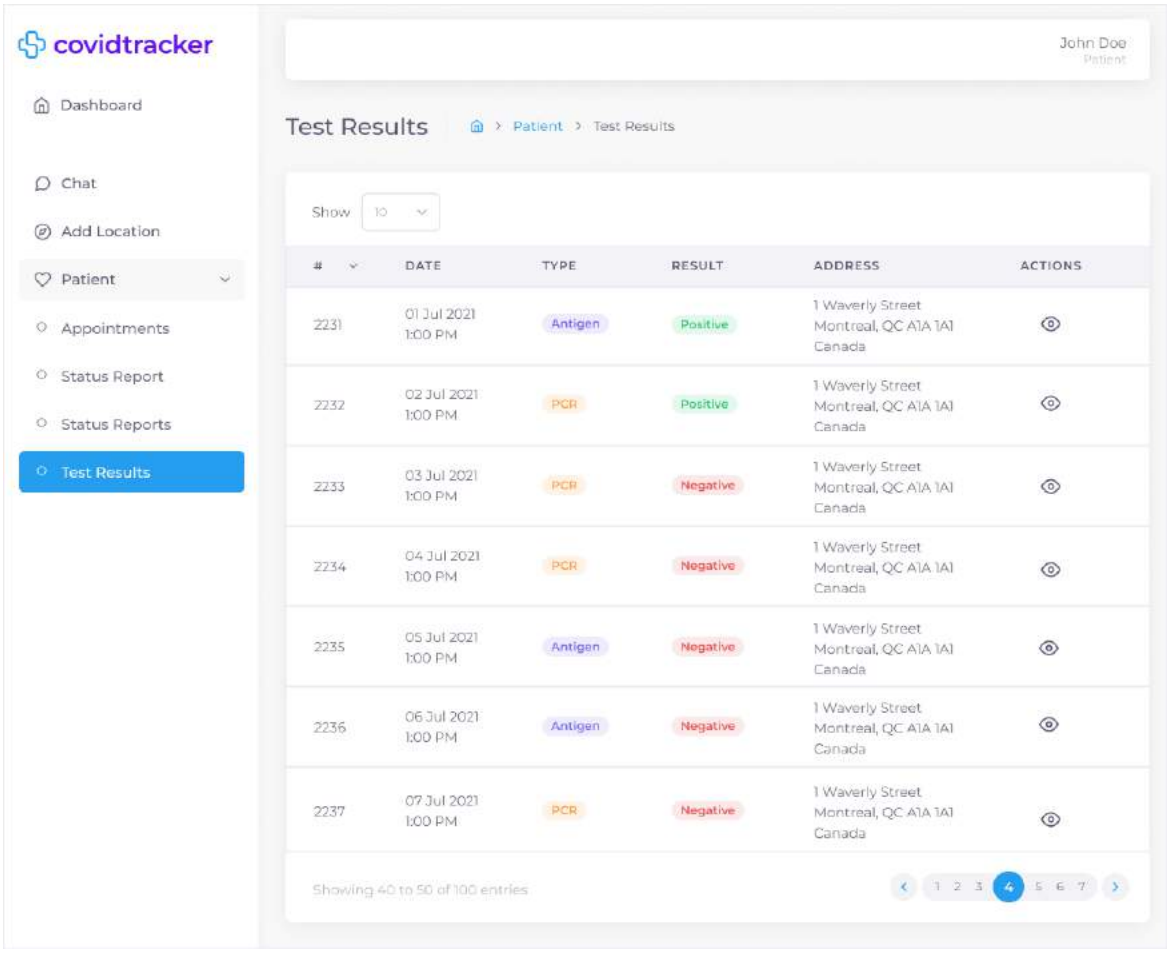


Figure 72: Test Results (Patient) Desktop & Tablet UI Mockup

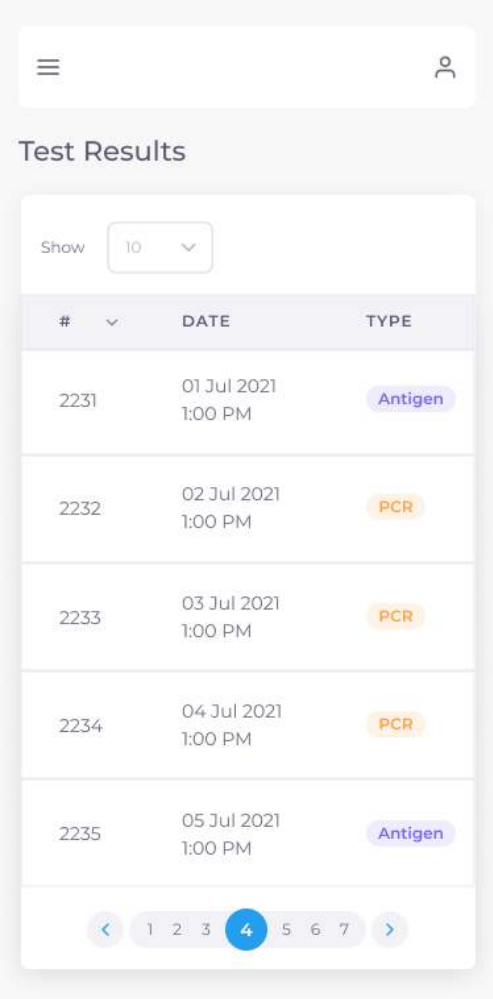


Figure 73: Test Results (Patient) Mobile UI Mockup

- UI and User Flow Mockup - Test Results (Doctor/Health Official) / Desktop & Tablet
- UI and User Flow Mockup - Test Results (Doctor/Health Official) / Mobile
- ▸ Prototype - Test Results (Doctor/Health Official) / Desktop & Tablet
- ▸ Prototype - Test Results (Doctor/Health Official) / Mobile

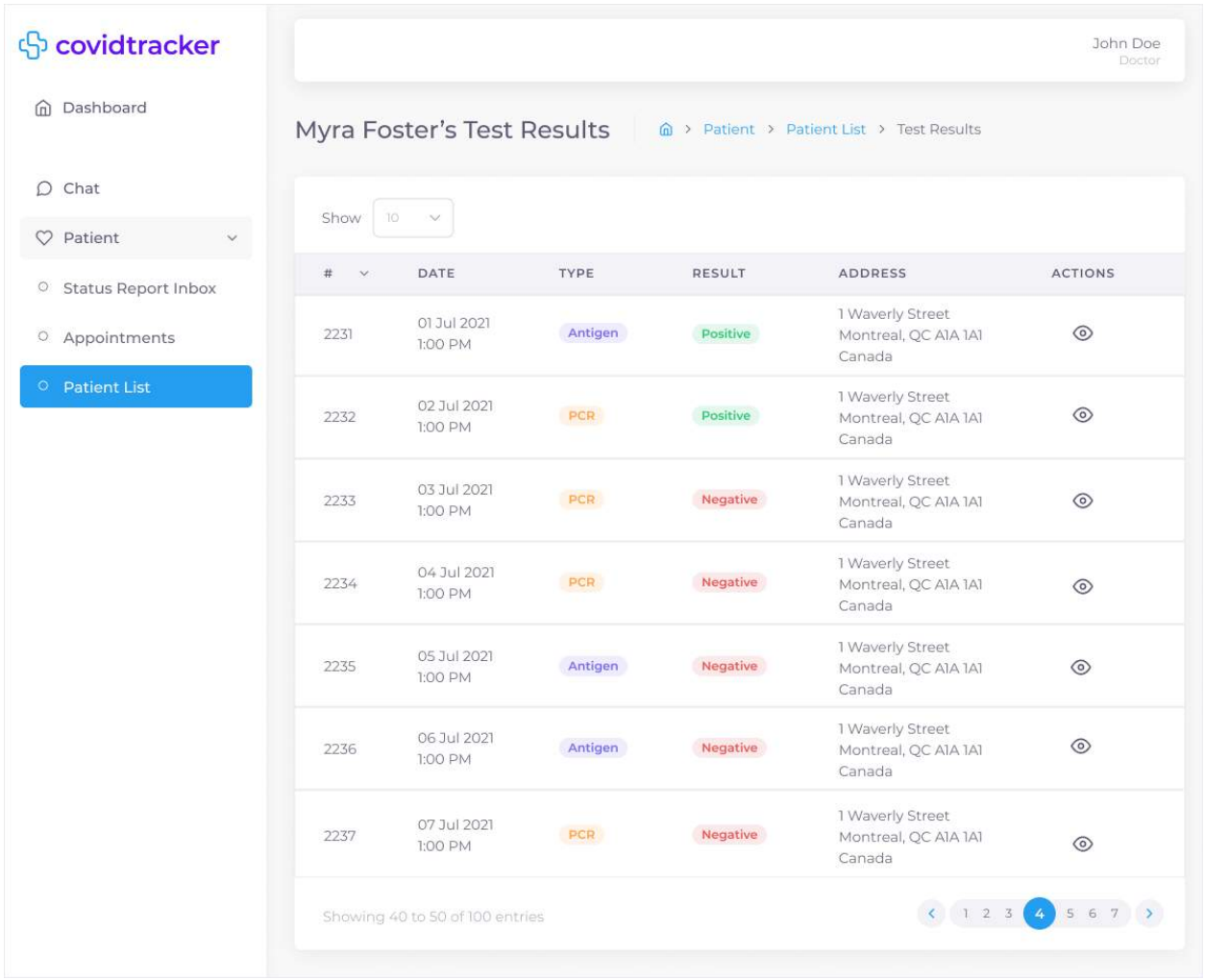


Figure 74: Test Results (Doctor/Health Official) Desktop & Tablet UI Mockup

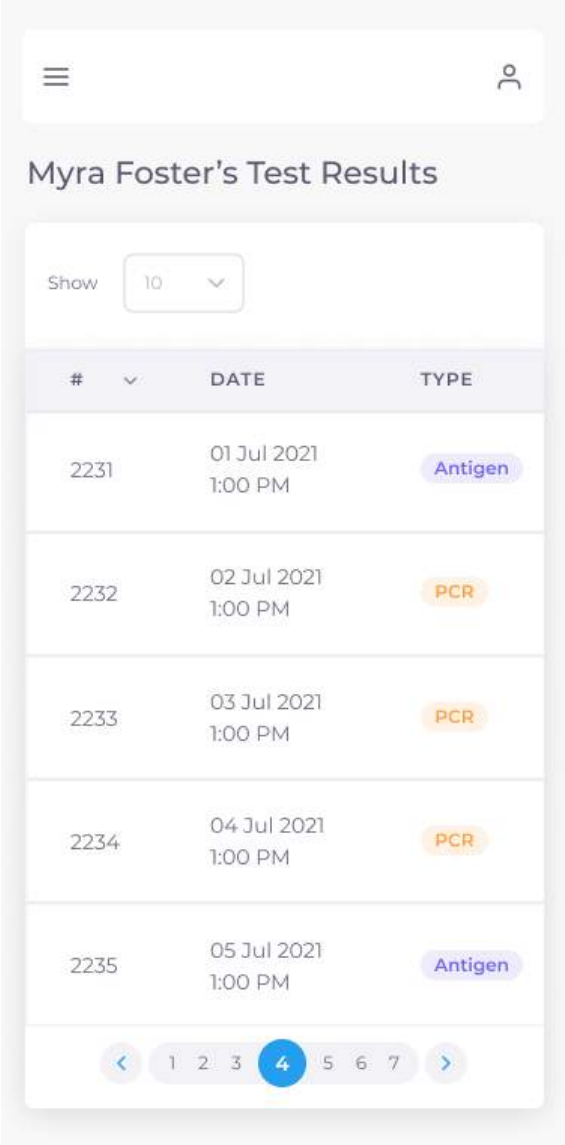


Figure 75: Test Results (Doctor/Health Official) Desktop & Tablet UI Mockup

- UI and User Flow Mockup - Test Results (Immigration Officer) / Desktop & Tablet
- UI and User Flow Mockup - Test Results (Immigration Officer) / Mobile
- ▸ Prototype - Test Results (Immigration Officer) / Desktop & Tablet
- ▸ Prototype - Test Results (Immigration Officer) / Mobile

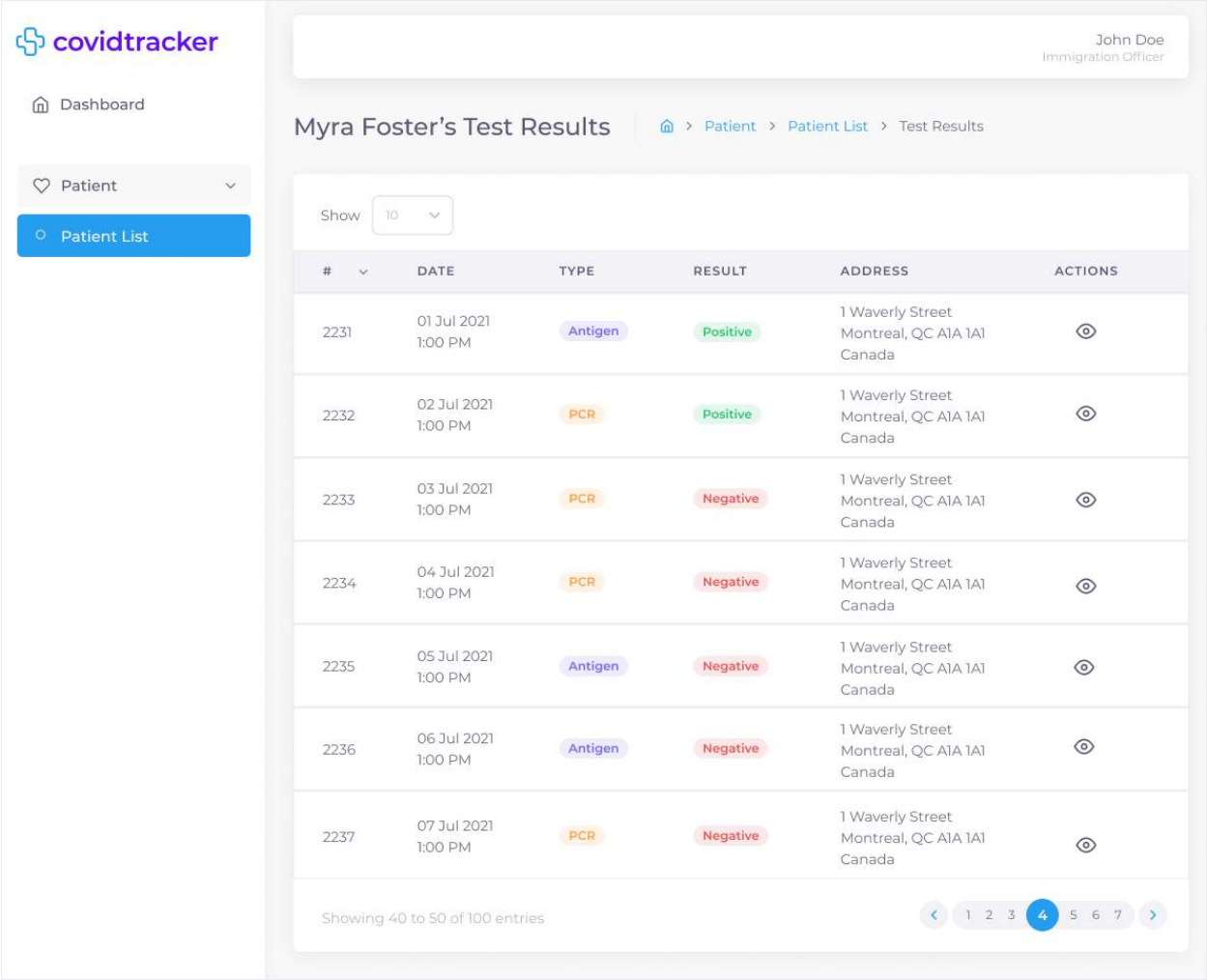


Figure 76: Test Results (Immigration Officer) Desktop & Tablet UI Mockup

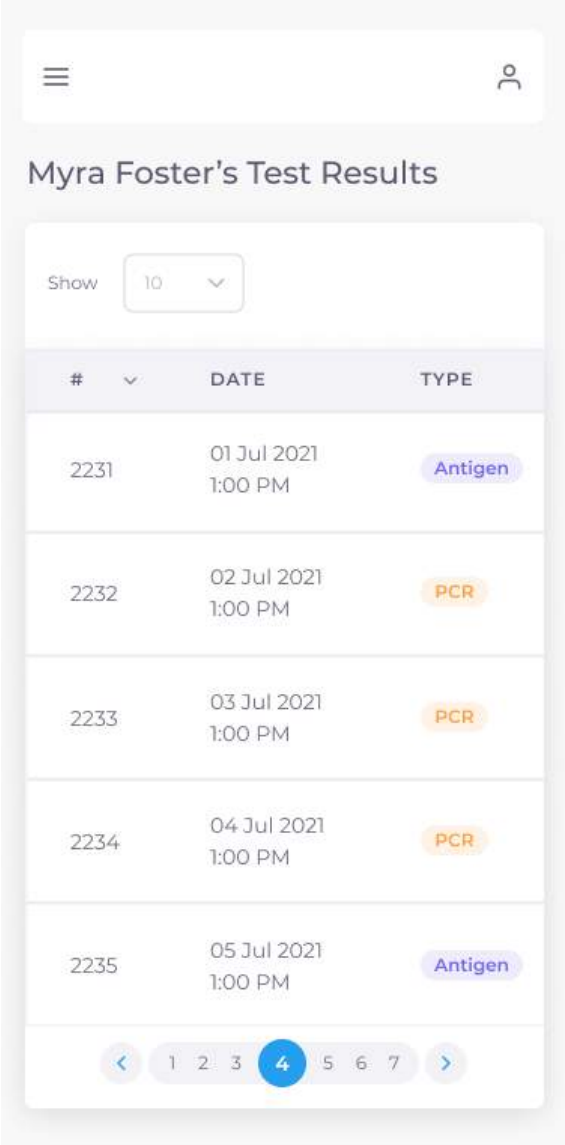


Figure 77: Test Results (Immigration Officer) Mobile UI Mockup

7.3.15 Test Result Details

COV-122 - As a Patient, I want to be able to generate a QR code for a lab test result, so that I can share it with others

COV-124 - As a Patient, I want to view the details of a single COVID test result, so that I'm aware of my diagnosis

COV-223 - As an Immigration Officer, I want to view the patients list, so I can be aware of test results and prioritize them

A patient is able to view a full detailed description of a given test result and its associated QR code (additional feature) that can be used to easily share such information with either their doctor or a health official. Likewise, upon scanning such QR code, a doctor or health official will be redirected to a similar page. The UI is only accessible by the Patient, Doctor, Health Official and Immigration Officer personas. The only UI element that adjusts based on the persona is the breadcrumb text. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 78 and 79 for the Patient person, Figures 80 and 81 for the Doctor and Health Official personas, and Figures 82 and 83 for Immigration Officer personas. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Test Result Details (Patient) / Desktop & Tablet
- UI and User Flow Mockup - Test Result Details (Patient) / Mobile
- ▸ Prototype - Test Result Details (Patient) / Desktop & Tablet
- ▸ Prototype - Test Result Details (Patient) / Mobile

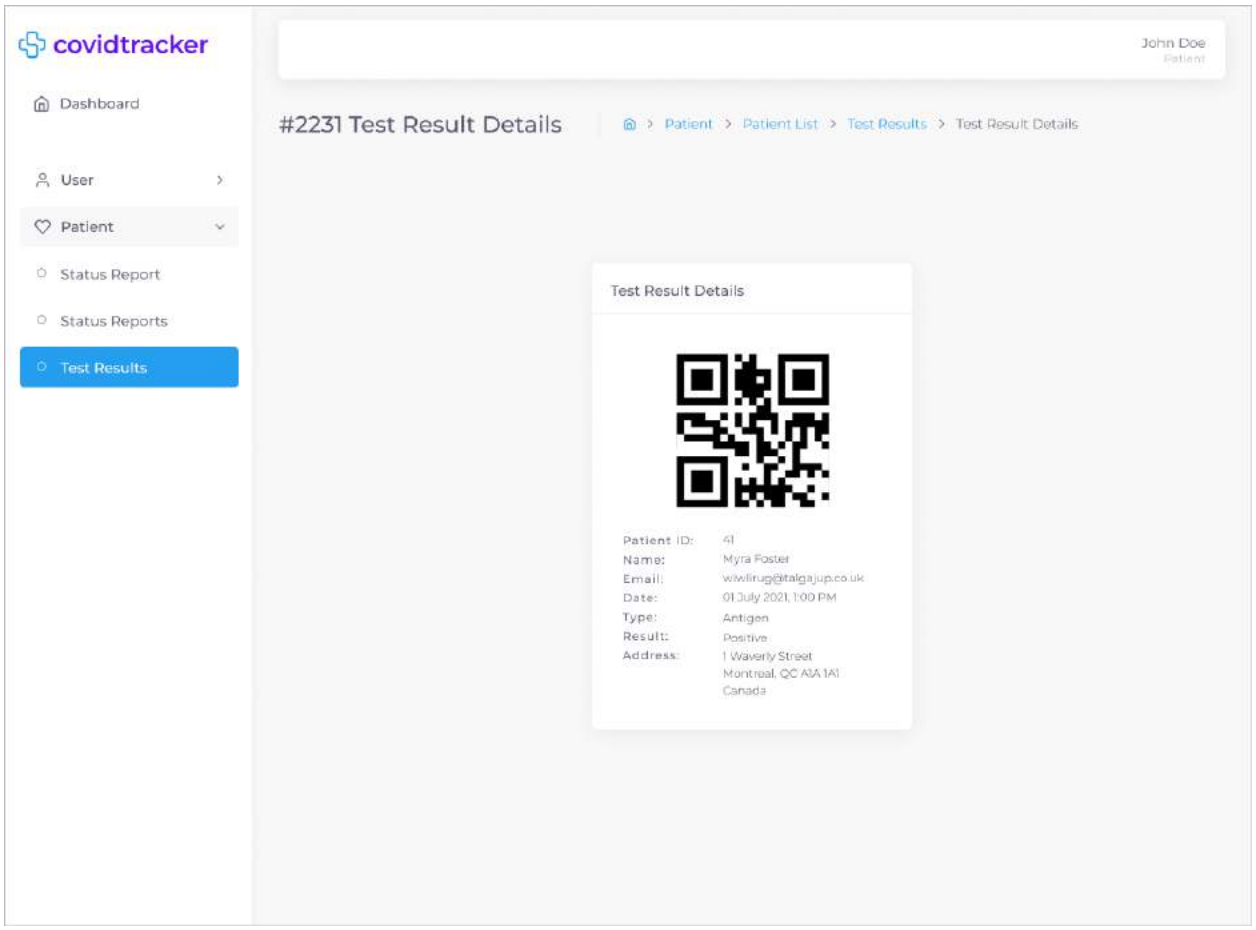


Figure 78: Test Result Details (Patient) Desktop & Tablet UI Mockup

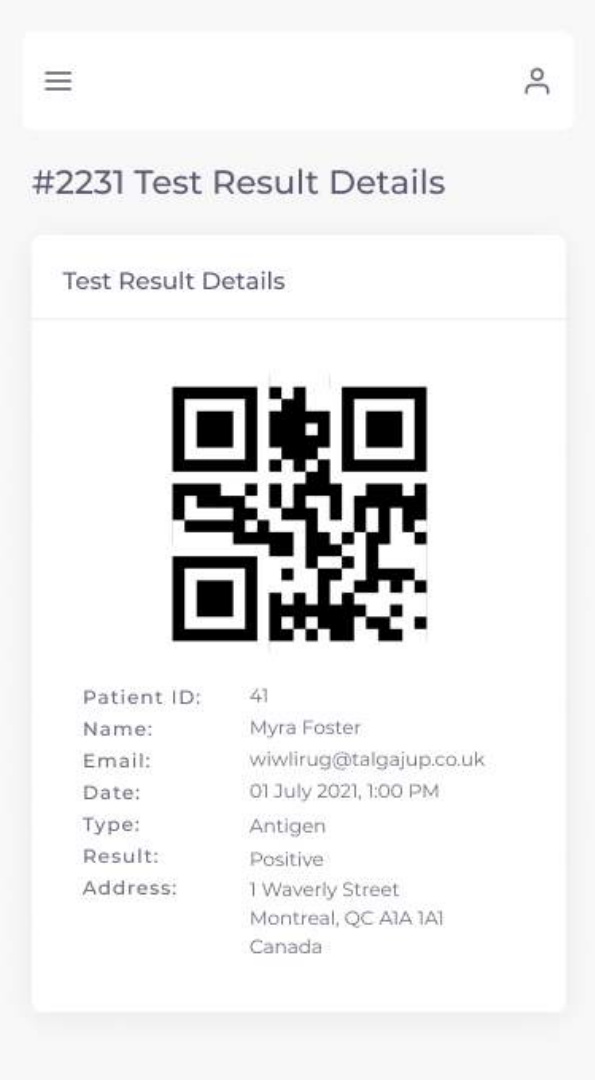


Figure 79: Test Result Details (Patient) Mobile UI Mockup

- UI and User Flow Mockup - Test Result Details (Doctor/Health Official) / Desktop & Tablet
- UI and User Flow Mockup - Test Result Details (Doctor/Health Official) / Mobile
- ▸ Prototype - Test Result Details (Doctor/Health Official) / Desktop & Tablet
- ▸ Prototype - Test Result Details (Doctor/Health Official) / Mobile

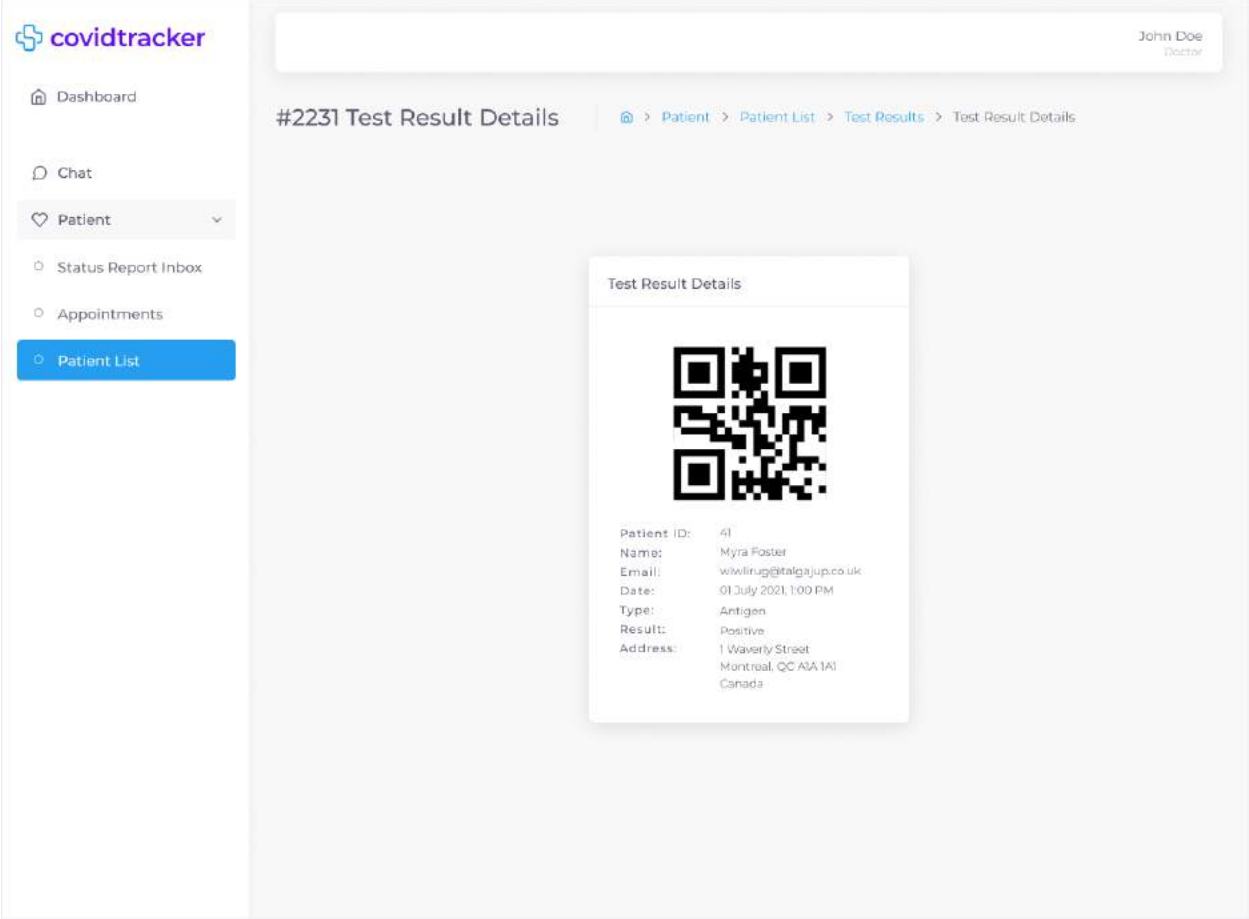


Figure 80: Test Result Details (Doctor/Health Official) Desktop & Tablet UI Mockup

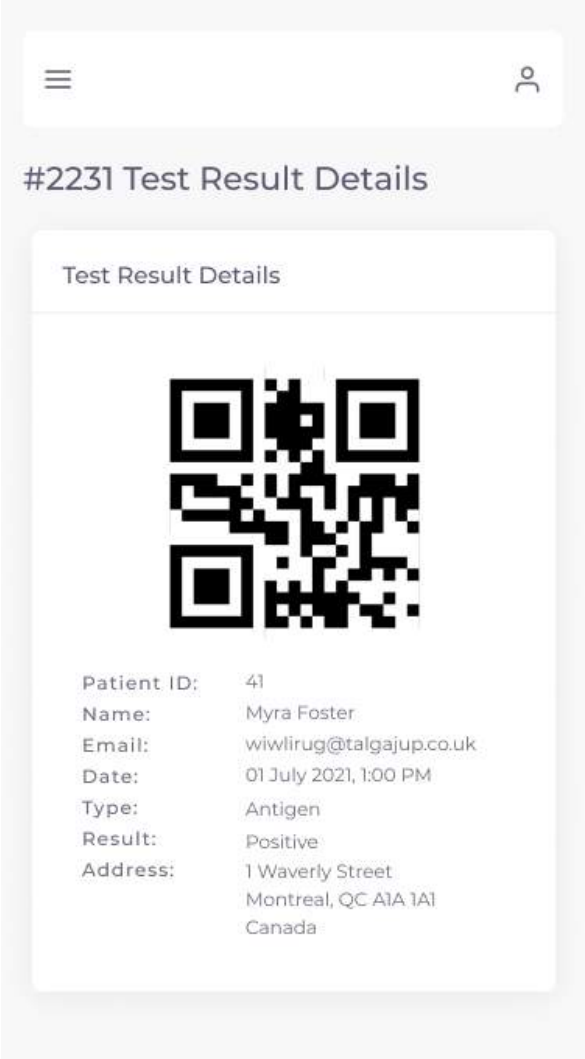


Figure 81: Test Result Details (Doctor/Health Official) Mobile UI Mockup

- UI and User Flow Mockup - Test Result Details (Immigration Officer) / Desktop & Tablet
- UI and User Flow Mockup - Test Result Details (Immigration Officer) / Mobile
- ▸ Prototype - Test Result Details (Immigration Officer) / Desktop & Tablet
- ▸ Prototype - Test Result Details (Immigration Officer) / Mobile

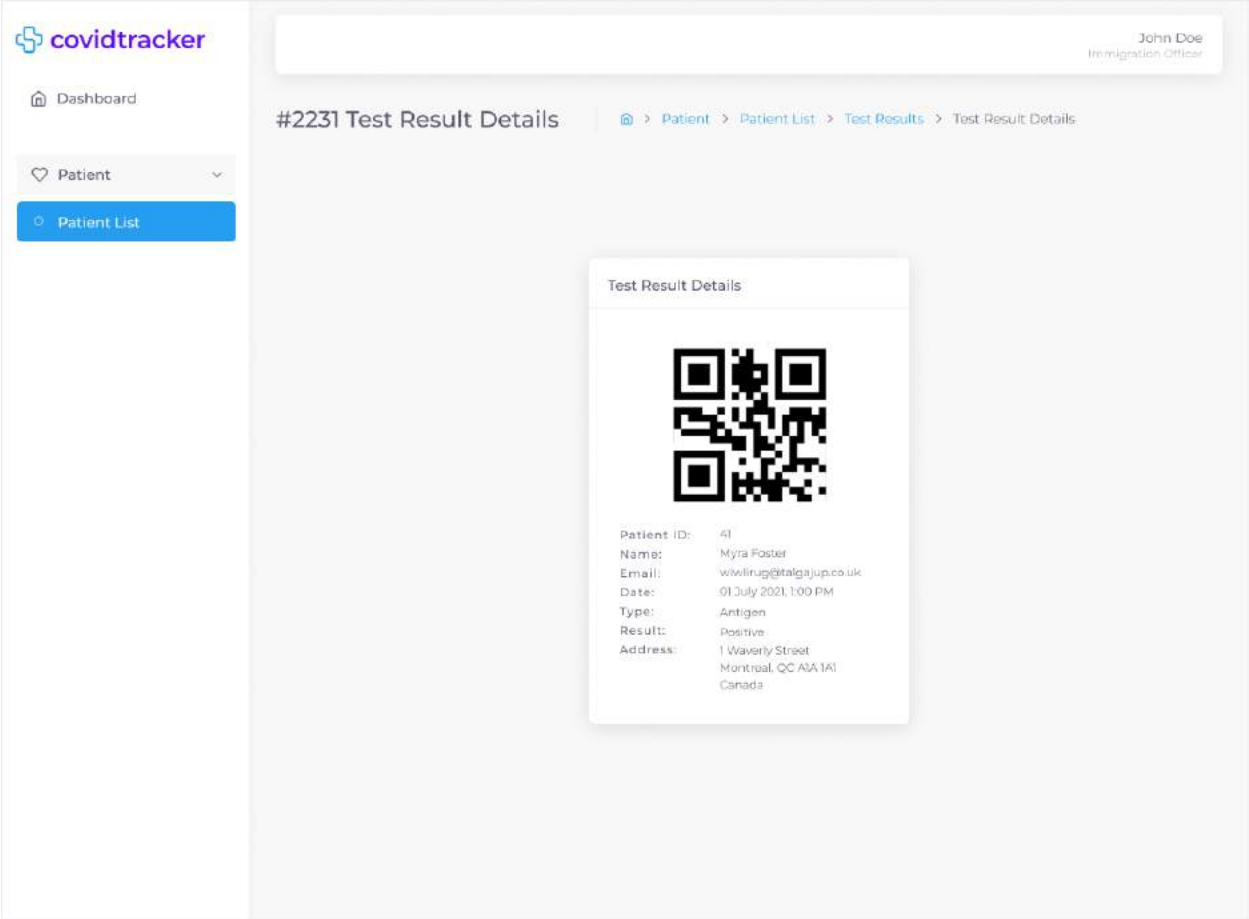


Figure 82: Test Result Details (Immigration Officer) Desktop & Tablet UI Mockup

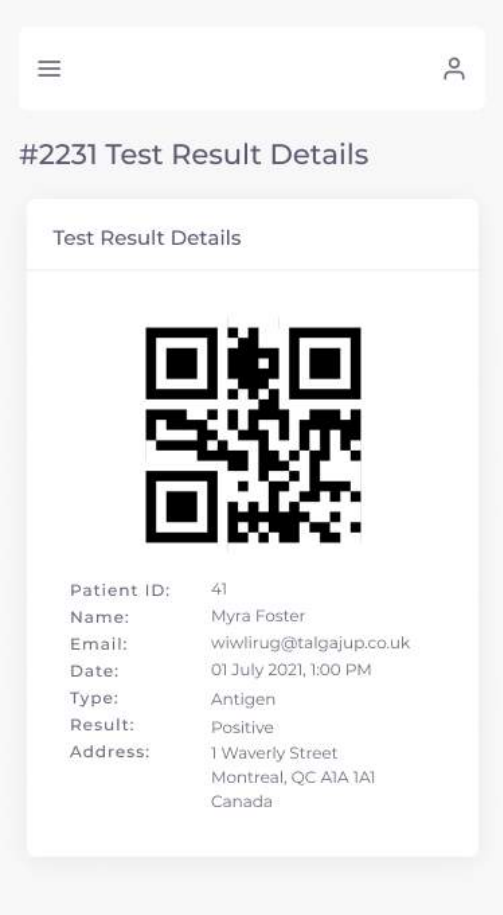


Figure 83: Test Result Details (Immigration Officer) Mobile UI Mockup

7.3.16 Chat

COV-119 - As a Patient, I want to direct message my Doctor, so that I can ask them questions

COV-120 - As a Patient, I want to mark my message with a priority level, so that my Doctor will view it quicker

A doctor and patient are able to communicate with each other through instant messaging. The UI is split in two sections: contacts (left) and chat (right). The contacts section is a list view of all the contacts assigned to either a doctor or patient. A doctor can communicate with any of their patients while a patient can only communicate with their assigned doctor. The number of unread messages from each contact is displayed as either a red (urgent message) or blue (regular message) bubble under the last message timestamp. A search bar is provided to easily and quickly find a contact. The chat section contains all exchanged messages between the two parties. A patient is able to flag a message as urgent by clicking the flag icon located in the message textbox. Urgent messages are then represented as a red background within the chat window and act as signals to doctors for immediate action. A doctor cannot flag a message as urgent. The UI is only accessible by the Patient and Doctor personas. The only UI element that adjusts based on the persona is the presence of the urgent message flag icon within the message textbox for the patient. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 84 and 85 for the Doctor person and Figures 86 and 87 for the Patient personas. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Chat (Doctor) / Desktop & Tablet
- UI and User Flow Mockup - Chat (Doctor) / Mobile
- ▸ Prototype - Chat (Doctor) / Desktop & Tablet
- ▸ Prototype - Chat (Doctor) / Mobile

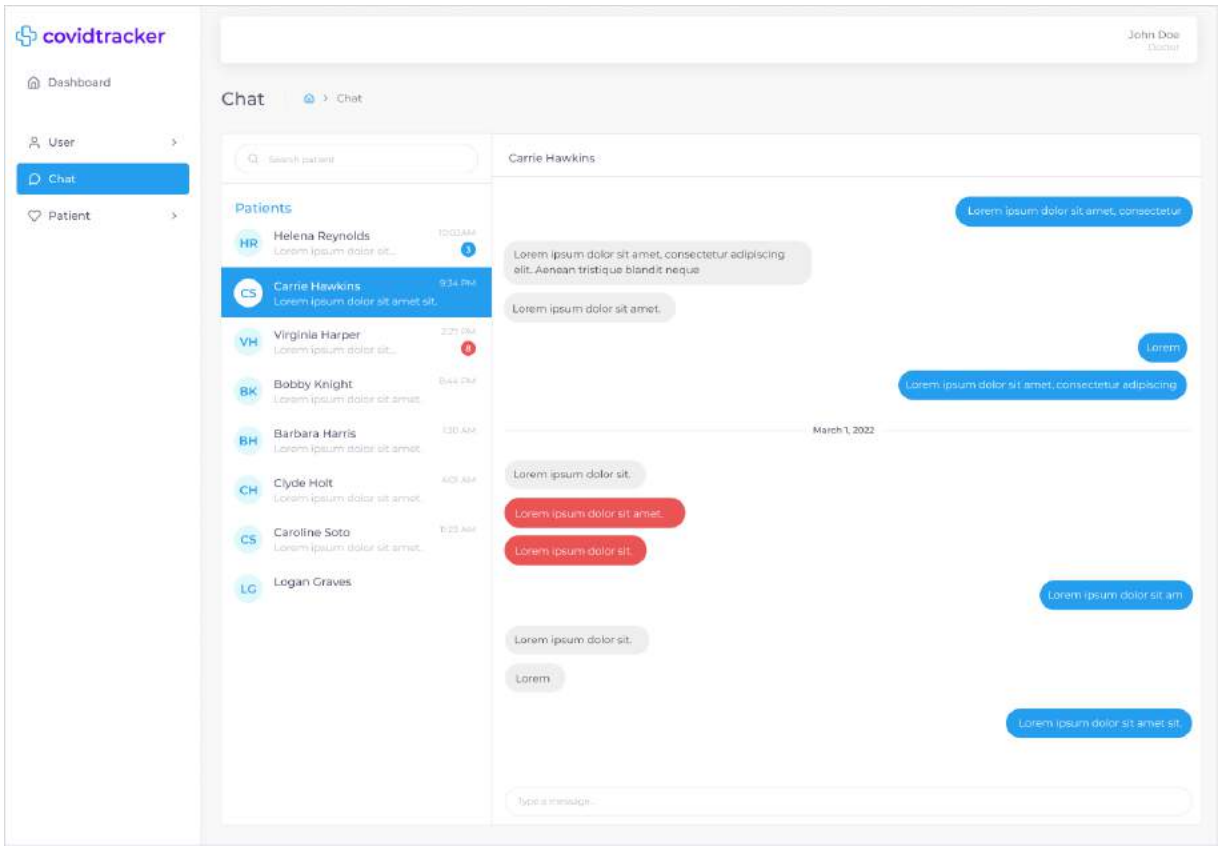


Figure 84: Chat (Doctor) Desktop & Tablet UI Mockup

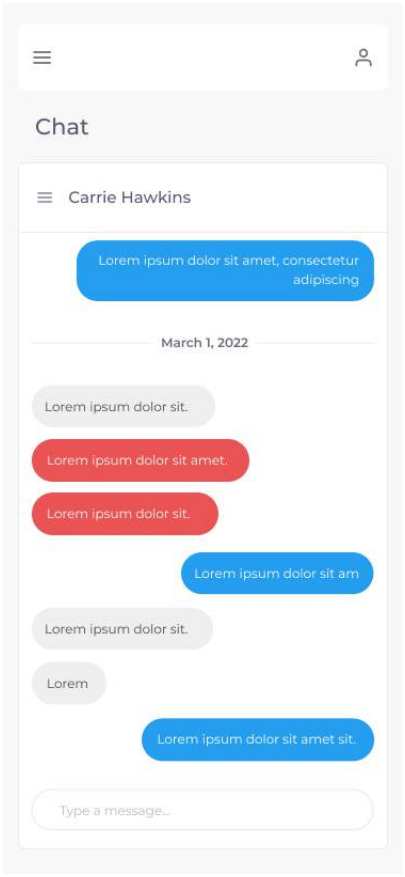


Figure 85: Chat (Doctor) Mobile UI Mockup

- UI and User Flow Mockup - Chat (Patient) / Desktop & Tablet
- UI and User Flow Mockup - Chat (Patient) / Mobile
- ▸ Prototype - Chat (Patient) / Desktop & Tablet
- ▸ Prototype - Chat (Patient) / Mobile

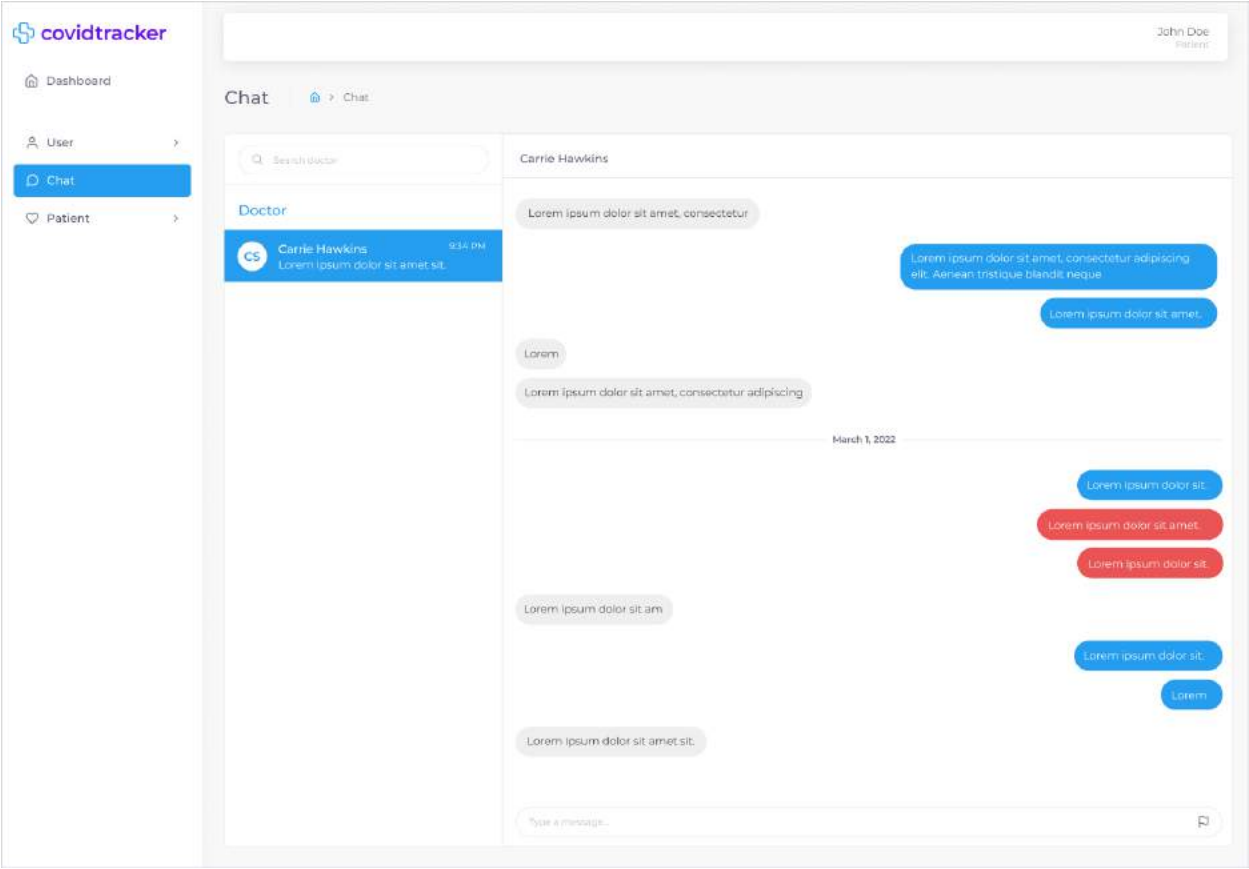


Figure 86: Chat (Patient) Desktop & Tablet UI Mockup

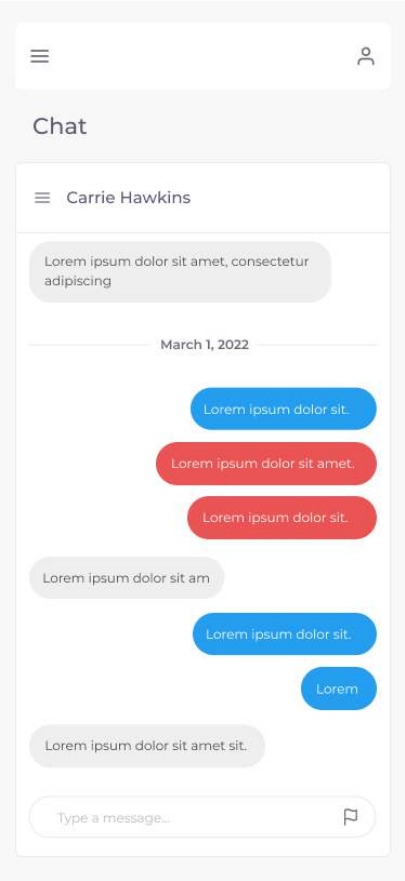


Figure 87: Chat (Patient) Mobile UI Mockup

7.3.17 Book Appointment

COV-116 - As a Doctor, I want to book an appointment with a Patient, so that we can discuss their symptoms

A doctor is able to book an appointment with a given patient. The following information must be provided: date of appointment, time of appointment (start time and end time), and location of appointment. This page can be accessed from the Patient List by selecting the “Book Appointment” option found in the more options dropdown for a given patient as described in section 7.3.9 Patient List. The UI is only accessible by the Doctor persona and does not adjust based on persona. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 87 and 88. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Book Appointment / Desktop & Tablet
- UI and User Flow Mockup - Book Appointment / Mobile
- ▸ Prototype - Book Appointment / Desktop & Tablet
- ▸ Prototype - Book Appointment / Mobile

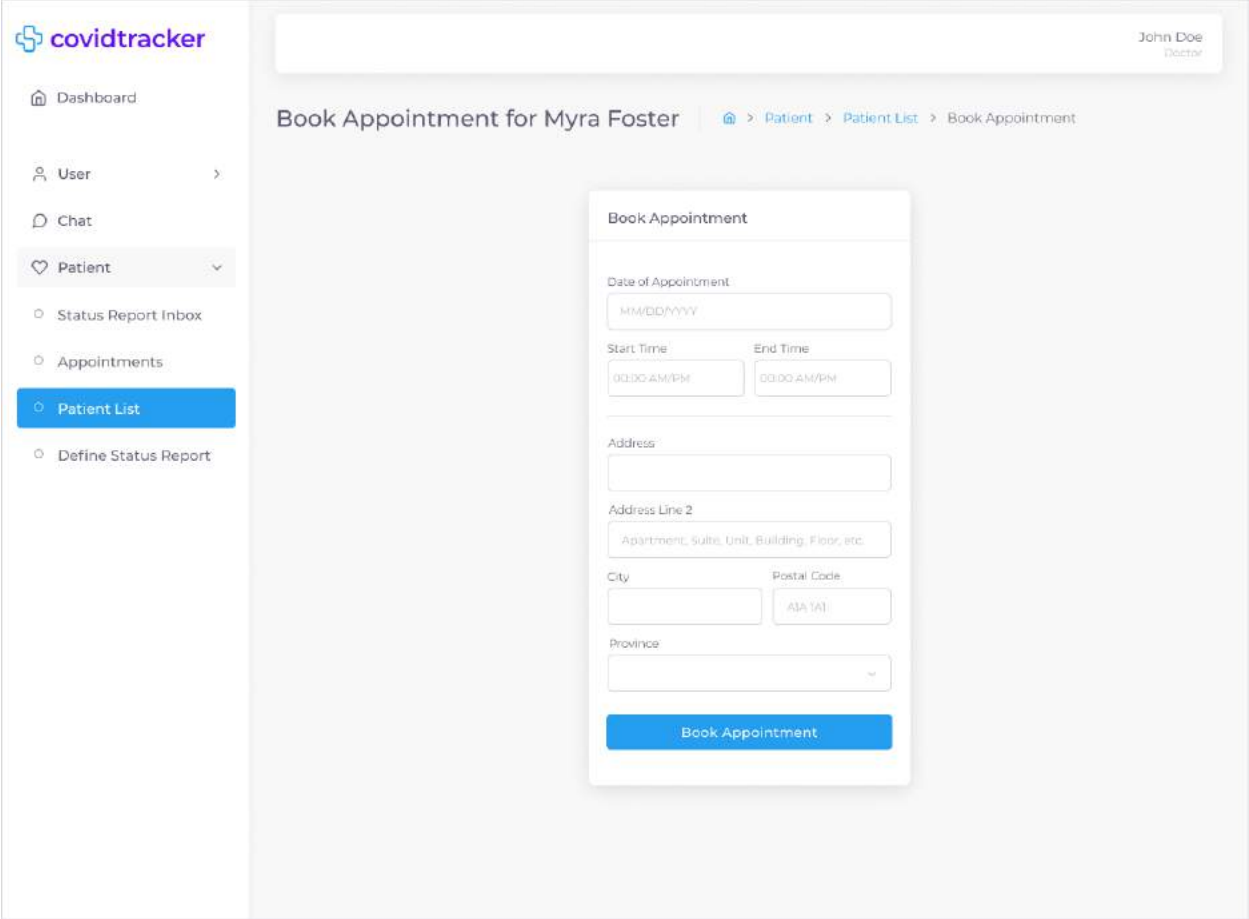


Figure 87: Book Appointment Desktop & Tablet UI Mockup

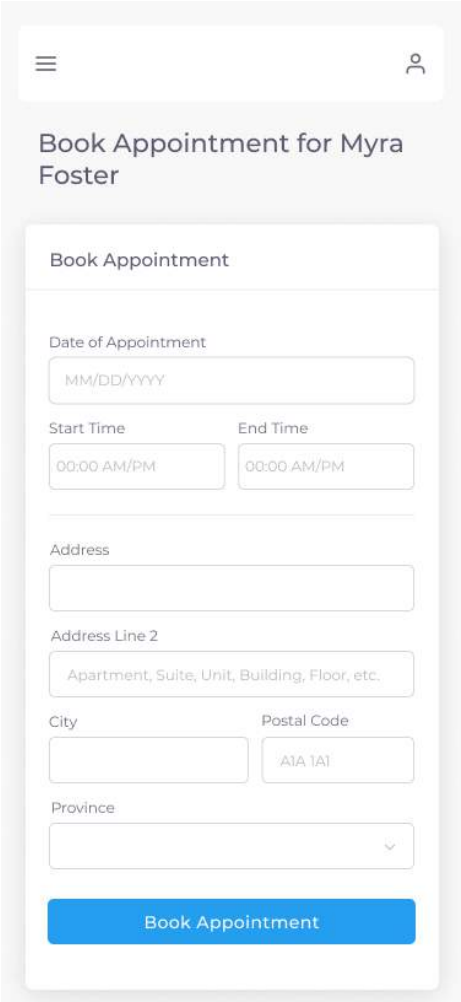


Figure 88: Book Appointment Desktop & Tablet UI Mockup

7.3.18 Appointments

COV-169 - As a Doctor, I want to view my appointments, so that I can schedule myself

A doctor is able to view a table containing all their patient appointments. There are also two information cards above the table describing the total number of appointments and number of appointments for the current day. This page is also accessible to a patient displaying a list of all appointments with their doctor. There are two UI adjustments present on the page between the doctor and patient. The first UI adjustment is the doctor will see information cards above the table while a patient will not. The second UI adjustment is that for a doctor each appointment will contain the patient name and email while for the patient the appointment table will contain the doctor name and email. Therefore, the UI is only accessible by the Doctor, and Patient personas. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 89 and 90 for the Doctor person and Figures 91 and 92 for the Patient personas. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Appointments (Doctor) / Desktop & Tablet
- UI and User Flow Mockup - Appointments (Doctor) / Mobile
- ▸ Prototype - Appointment (Doctor) / Desktop & Tablet
- ▸ Prototype - Appointments (Doctor) / Desktop & Tablet

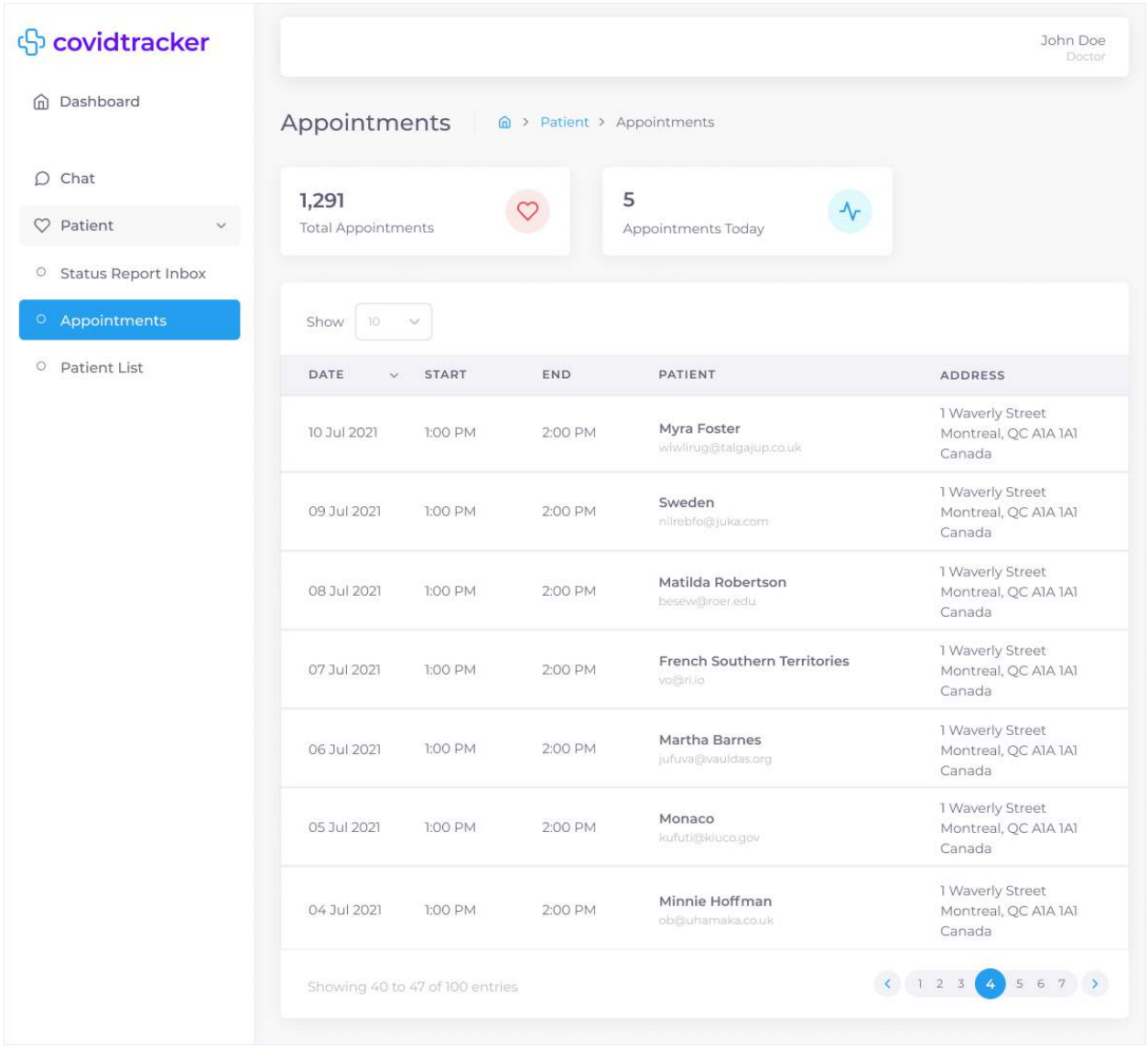


Figure 89: Appointments (Doctor) Desktop & Tablet UI Mockup

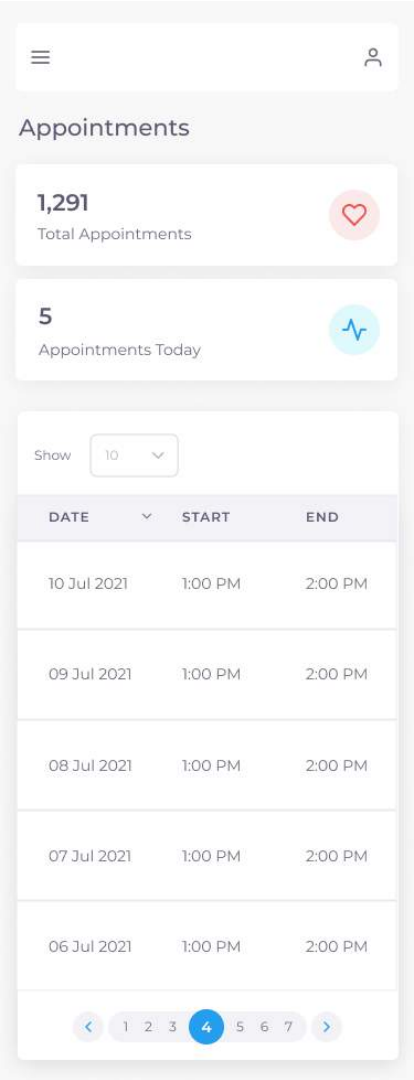


Figure 90: Appointments (Doctor) Mobile UI Mockup

- UI and User Flow Mockup - Appointments (Patient) / Desktop & Tablet
- UI and User Flow Mockup - Appointments (Patient) / Mobile
- ▸ Prototype - Appointment (Patient) / Desktop & Tablet
- ▸ Prototype - Appointments (Patient) / Desktop & Tablet

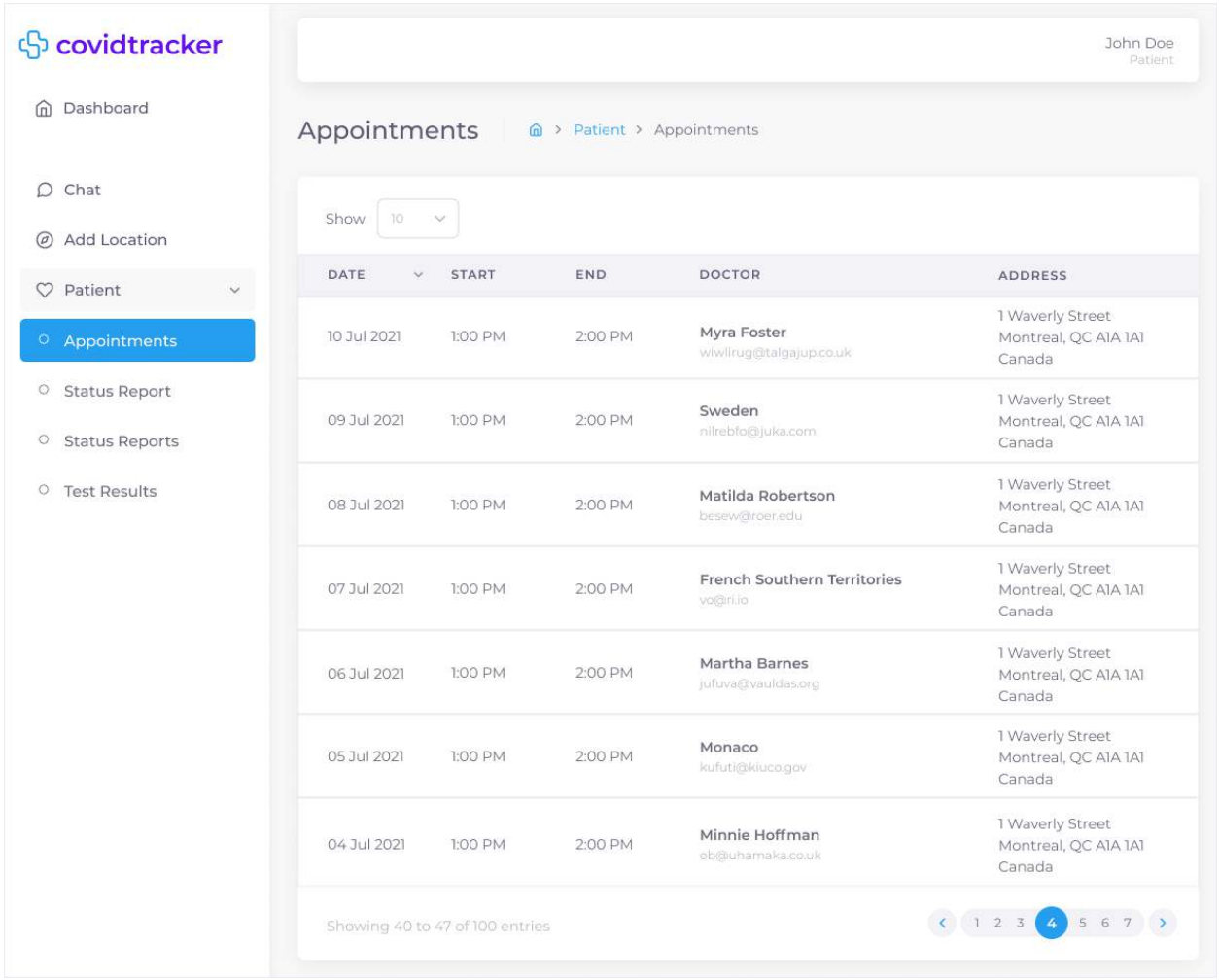


Figure 91: Appointments (Patient) Desktop & Tablet UI Mockup

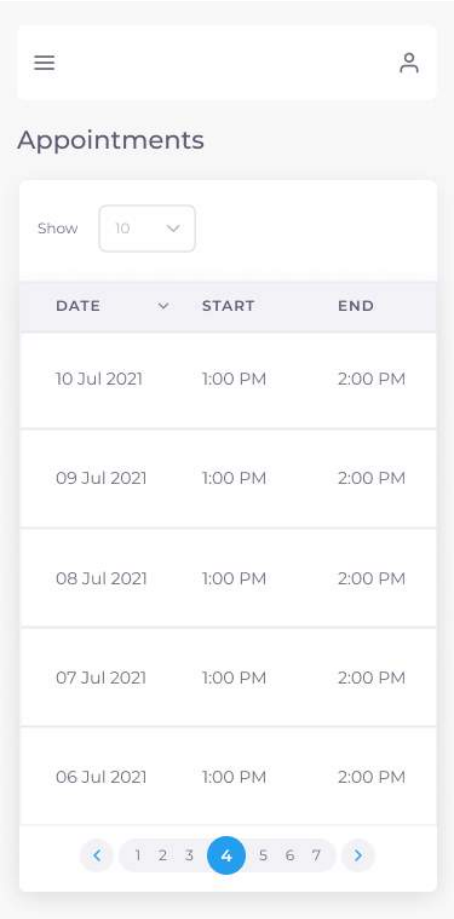


Figure 92: Appointments (Patient) Mobile UI Mockup

7.3.19 Add Location

COV-172 - As a Patient, I want to add the locations of where I have been during the day, so that I can be contact traced if I come in contact with someone that has tested positive with COVID-19

A patient and user (not an administrator, doctor, health official or immigration officer) are able to add the locations of where they can be during any given day in order to be contacted by a health official if they have come in contact with someone that has tested positive to COVID-19. The following information must be provided: date, and location. A form was decided as the best course of action for handling contact tracing within the system as compared to GPS or bluetooth for a variety of reasons. The first reason is that, since CovidTracker is a web app, it would be infeasible to ask a user to constantly keep their mobile phone open and on the website whenever in public. Secondly, with the rise of online tracking, people are more aware and concerned about being tracked by websites and apps they use than ever before. This is most evident with the rise in popularity of disabling tracking capabilities in mobile phones. In fact, according to statista, as of September 2021, the opt-in rate of iOS users worldwide choosing to allow app tracking after iOS 14.5 update is 21%. Meaning, 79% of iOS users worldwide are choosing to not be tracked by the apps they use. Lastly, implementing GPS or bluetooth functionality is extremely difficult and as such would take months of planning, development and testing to get it right from a functionality and privacy perspective. Therefore, given these reasons it was decided that having users fill up a form each time they leave their homes would be the best course of action for implementing contact tracing. While there is no way of ensuring all users will fill up the form each time they are in public, an assumption is being made that they will. The UI is only accessible by the Patient persona and users and does not adjust based on persona. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 93 and 94 for the Patient person and users. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Add Location / Desktop & Tablet
- UI and User Flow Mockup - Add Location / Mobile
- ▸ Prototype - Add Location / Desktop & Tablet
- ▸ Prototype - Add Location / Mobile

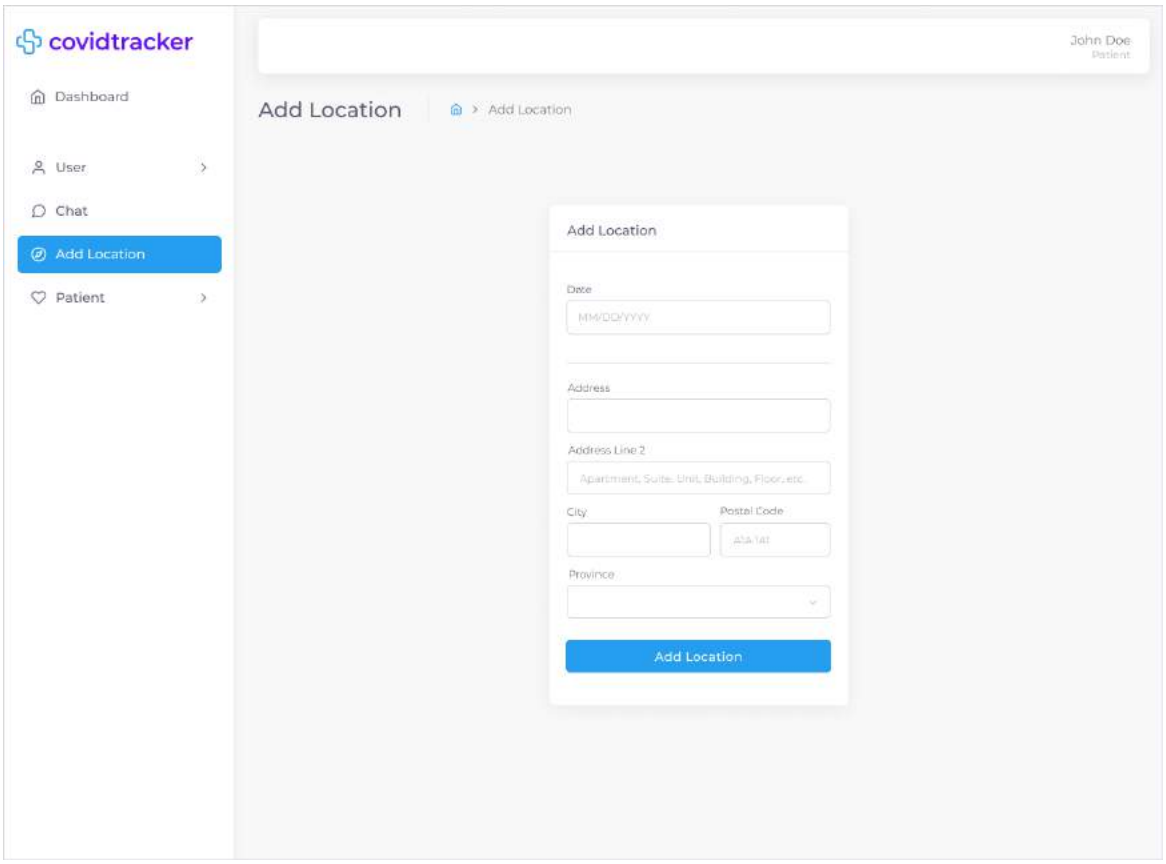


Figure 93: Add Location Desktop & Tablet UI Mockup

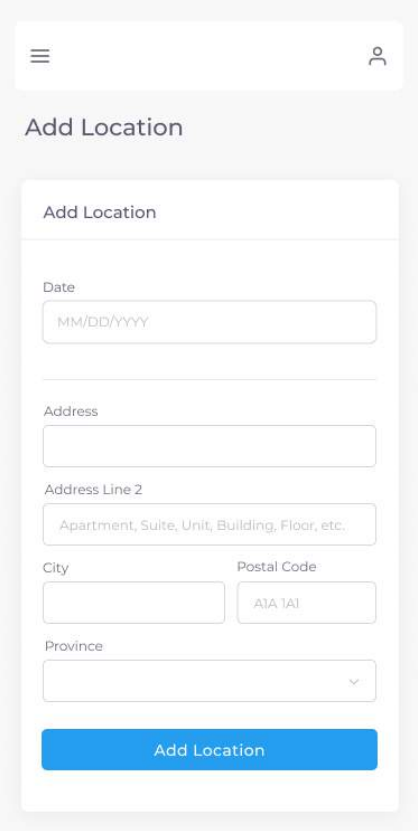


Figure 94: Add Location Desktop & Tablet UI Mockup

7.3.20 Contact Tracing

COV-171 - As a Health Official, I want to view a list of all patients who have tested positive in the last [x] days, so that I can contract trace them

A health official is able to view a table containing all patients that have tested positive for COVID-19. A health official is able to filter the result date (date patient tested positive) either by a range or single dates. A health official can also see all the individuals the patient has been in contact with over a period of time in order to begin the contact tracing process by clicking the “contacts” (multiple users) icon under the “Contacts” column. The UI is only accessible by the Health Official persona and does not adjust based on persona. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 95 and 96 for the Patient person and users. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Contact Tracing / Desktop & Tablet
- UI and User Flow Mockup - Contact Tracing / Mobile
- ▸ Prototype - Contact Tracing / Desktop & Tablet
- ▸ Prototype - Contact Tracing / Mobile

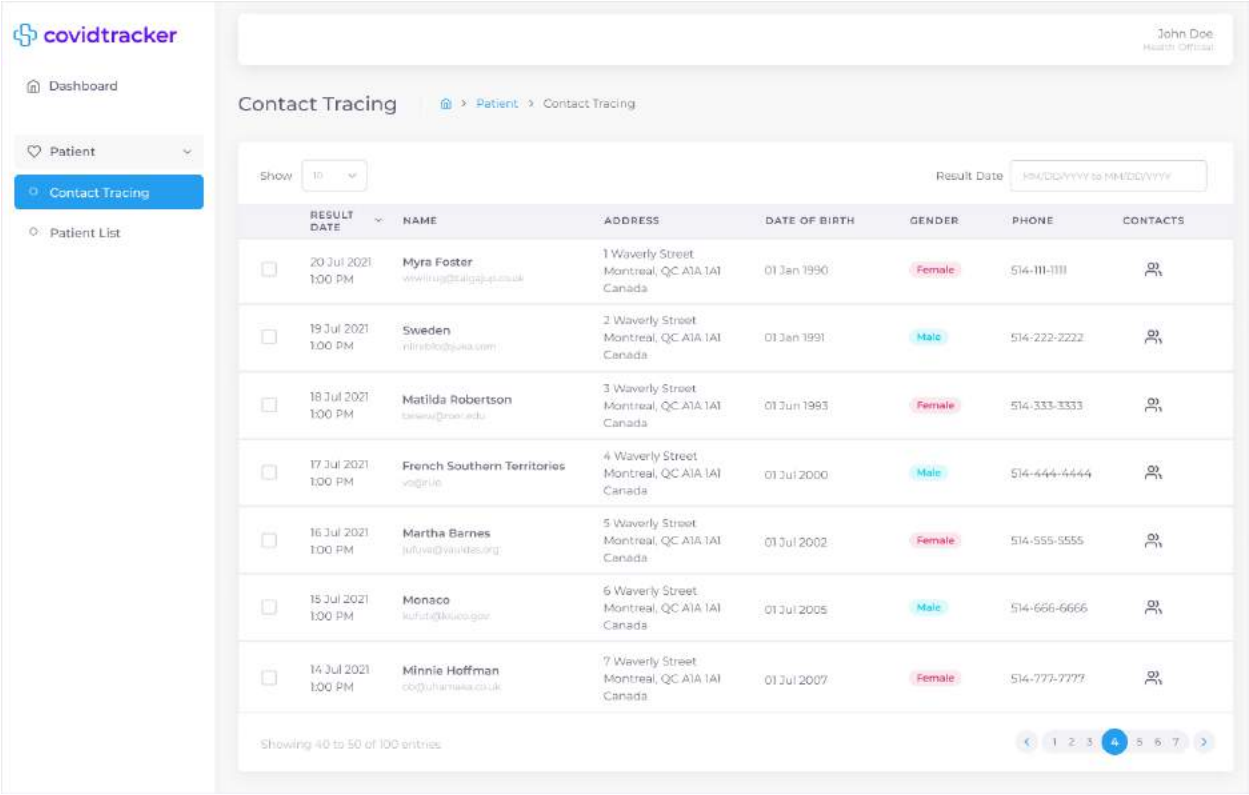


Figure 95: Contact Tracing Desktop & Tablet UI Mockup

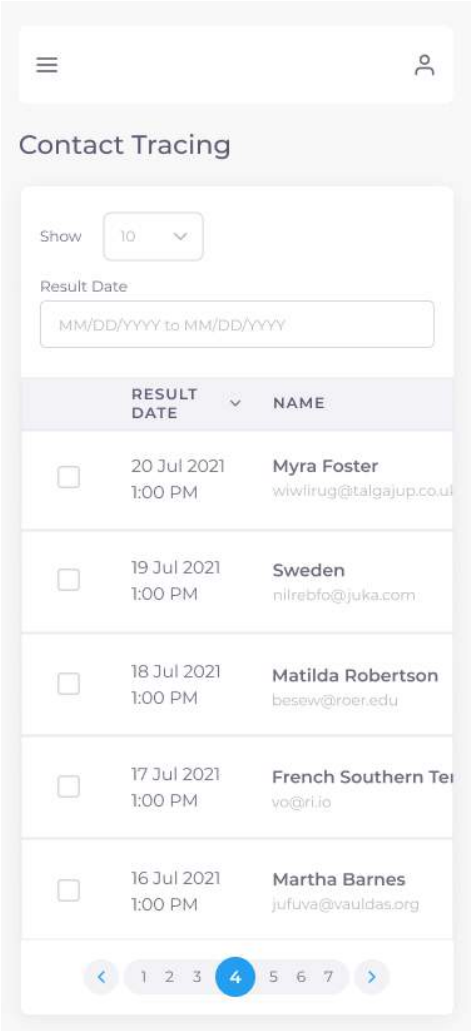


Figure 96: Contact Tracing Mobile UI Mockup

7.3.21 Contact Tracing Contacts

COV-126 - As a Health Official, I want to contact trace who a Patient has been in contact with in the last [x] days, so that I can manage who is at risk

A health official is able to view a table containing all individuals that have been in contact with a specific person that has tested positive for COVID-19. This page can be accessed from the Contact Tracing page by selecting the “contacts” (multiple users) icon found under the “Contacts” as described in section 7.3.20 Contact Tracing. Similar to the Contact Tracing page, a health official is able to filter the contact date either by a range of single dates. A health official is subsequently able to notify those individuals that have been in contact with the positive patient by clicking the “send” icon under the “Notify” column. Once the notification is successfully sent, a toast confirmation is displayed on the screen and the checkbox associated with the notified individual is checked. The UI is only accessible by the Health Official persona and does not adjust based on persona. A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 97 and 98 for the Patient person and users. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Contact Tracing Contacts / Desktop & Tablet
- UI and User Flow Mockup - Contact Tracing Contacts / Mobile
- ▸ Prototype - Contact Tracing Contacts / Desktop & Tablet
- ▸ Prototype - Contact Tracing Contacts / Mobile

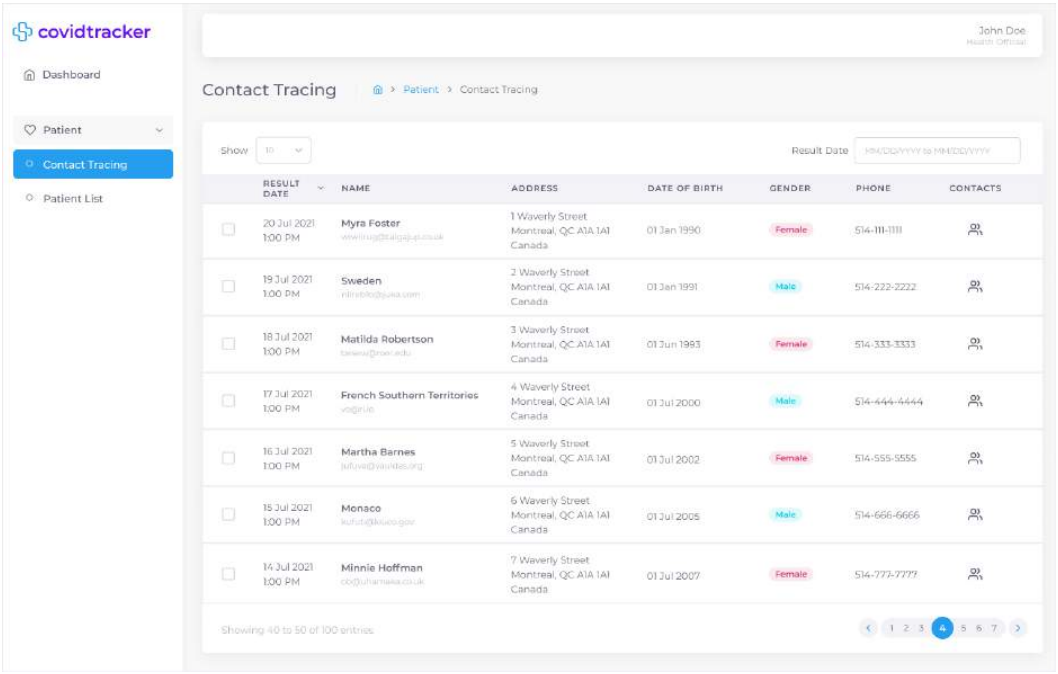


Figure 97: Contact Tracing Desktop & Tablet UI Mockup

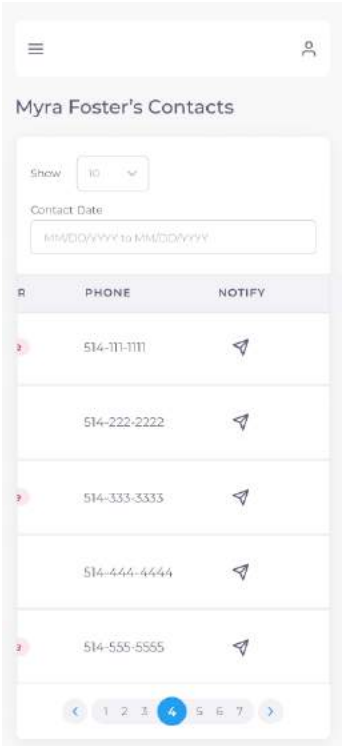


Figure 98: Contact Tracing Mobile UI Mockup

7.3.22 Dashboard

COV-125 - As a User, I want to a dashboard personalized for my role, so I can have an overview of the system

The dashboard allows users (i.e., without an assigned role), administrators, doctors, health officials, patients, and immigration officers to view an overview of various COVID-19 information, daily tasks, and personalized information. As such the UI adjusts based on each persona. All personas are able to view at a glance COVID-19 related information such as total cases, active cases, new cases for the day, total and new cases over time and cases by age. Information specific to the administrator persona is a patient summary (total patients in the system, and patients per doctor). Information specific to the doctor persona is a patient summary (total patients assigned, and newly assigned patients for the day), daily tasks (scheduled appointments, and unread status reports), and a primary & secondary symptoms chart of all symptoms submitted by patients through their daily status reports. Likewise, the health official persona has similar information such as a patient summary and a primary & secondary symptoms chart of all patients in the system. The immigration officer persona has a patient summary of all patients in the system. Information specific to the patient persona is a daily task summary (scheduled appointments, and number of status reports to submit). A selection of UI mockups for desktop, tablet, and mobile can be seen in Figures 99 and 100 for a User (i.e., without an assigned role), Figures 101 and 102 for an Administrator persona, Figures 103 and 104 for a Doctor persona, Figures 105 and 106 for a Health Official persona, Figures 107 and 108 for a Patient persona, and Figures 109 and 110 for an Immigration Officer persona. All UI mockups, user flows and associated interactive prototypes for desktop, tablet and mobile platforms are accessible at the following links:

- UI and User Flow Mockup - Dashboard (User) / Desktop & Tablet
- UI and User Flow Mockup - Dashboard (User) / Mobile
- ▸ Prototype - Dashboard (User) / Desktop & Tablet
- ▸ Prototype - Dashboard (User) / Mobile

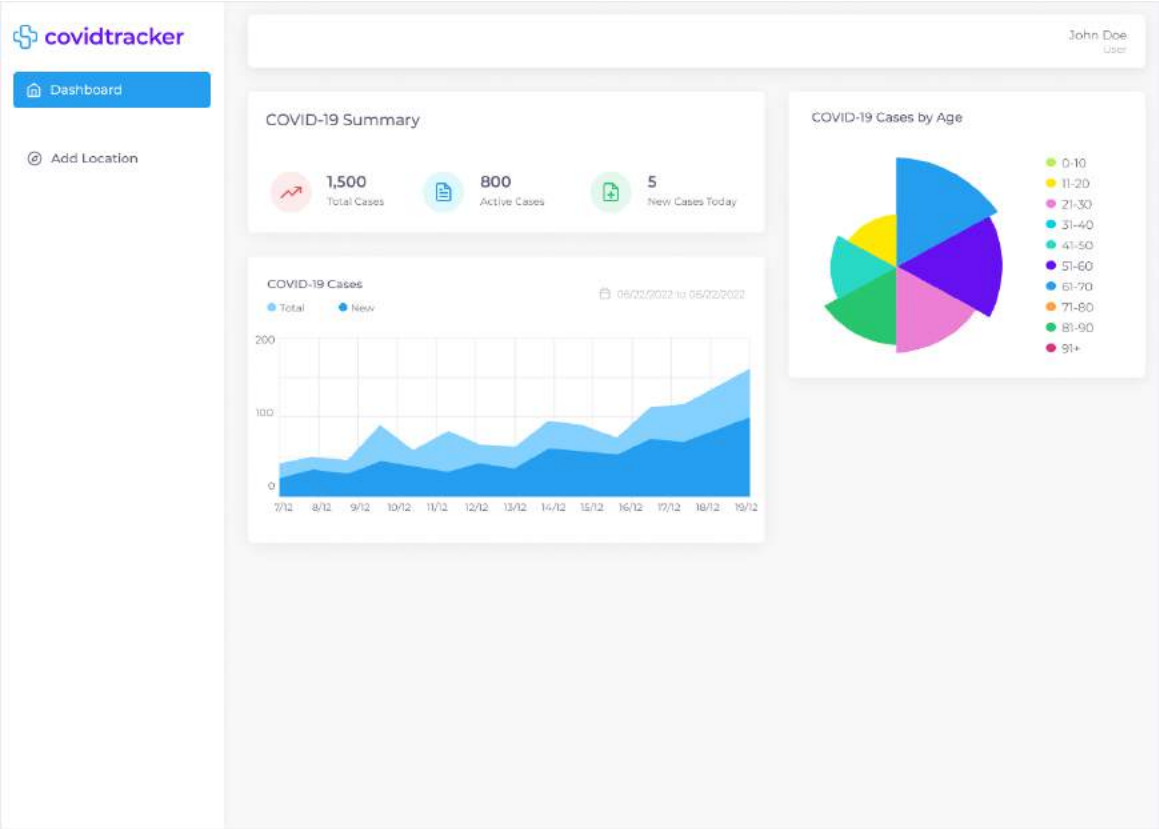


Figure 99: Dashboard (User) Desktop & Tablet UI Mockup

Figure 100: Dashboard (User) Mobile UI Mockup

- UI and User Flow Mockup - Dashboard (Administrator) / Desktop & Tablet
- UI and User Flow Mockup - Dashboard (Administrator) / Mobile
- ▸ Prototype - Dashboard (Administrator) / Desktop & Tablet
- ▸ Prototype - Dashboard (Administrator) / Mobile

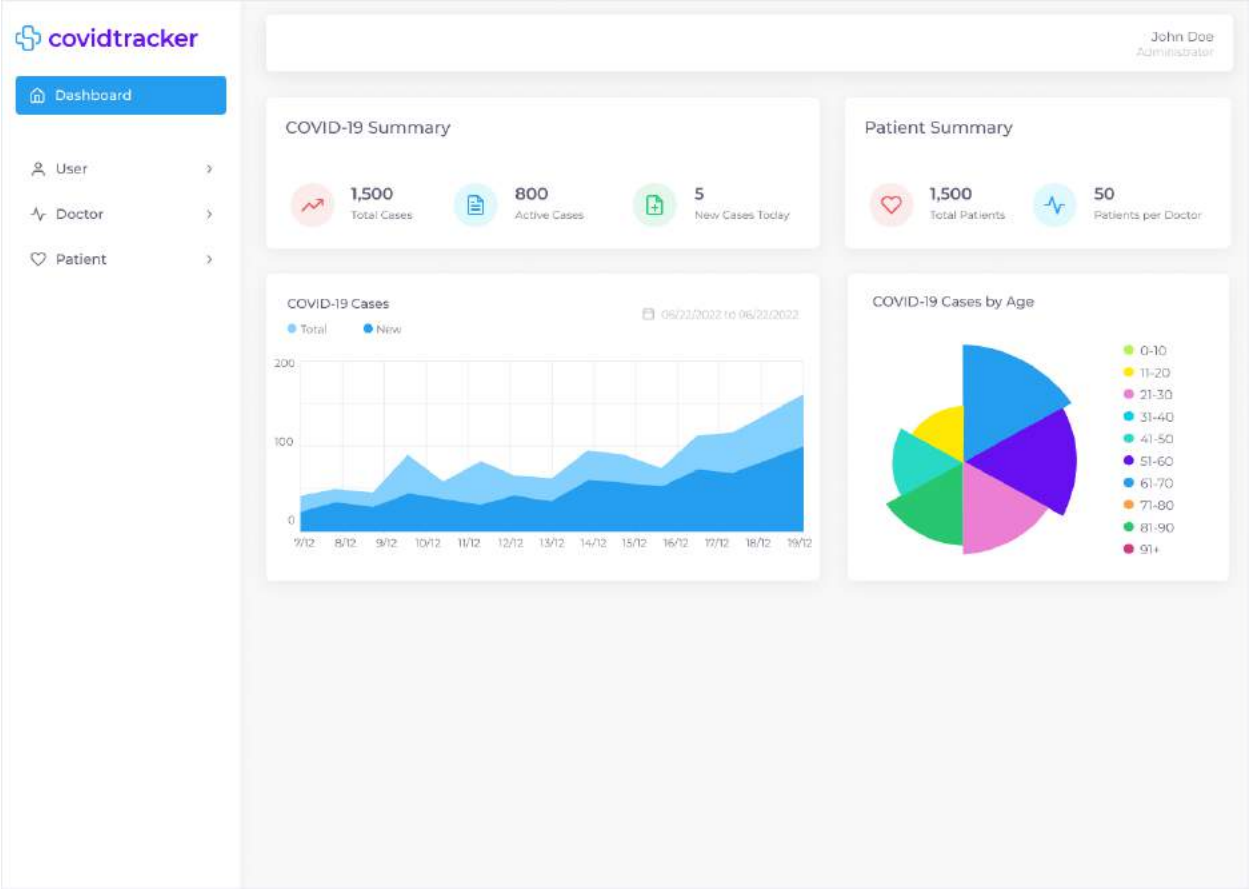


Figure 101: Dashboard (Administrator) Desktop & Tablet UI Mockup

Figure 102: Dashboard (Administrator) Mobile UI Mockup

- UI and User Flow Mockup - Dashboard (Doctor) / Desktop & Tablet
- UI and User Flow Mockup - Dashboard (Doctor) / Mobile
- ▸ Prototype - Dashboard (Doctor) / Desktop & Tablet
- ▸ Prototype - Dashboard (Doctor) / Mobile

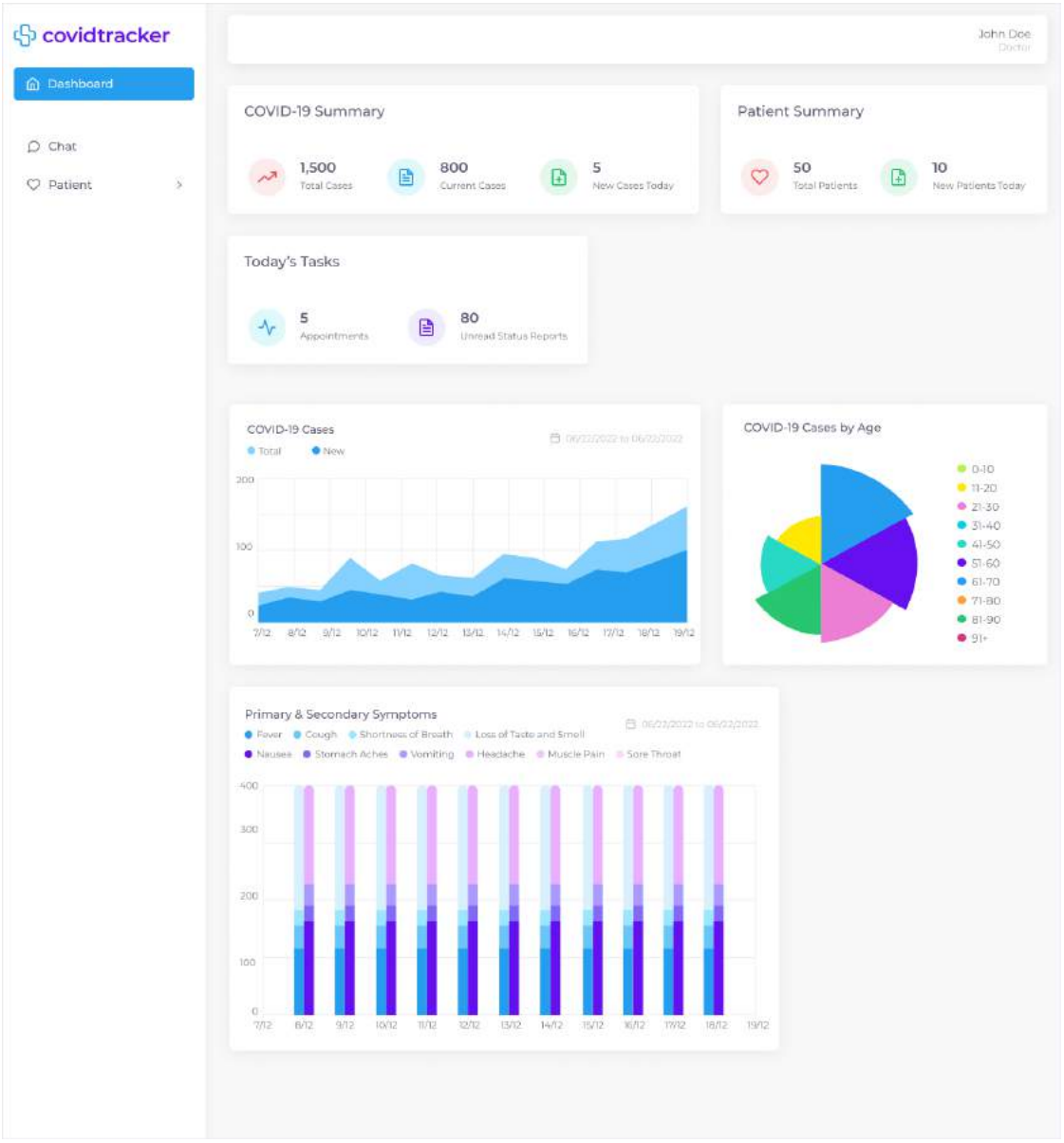


Figure 103: Dashboard (Doctor) Desktop & Tablet UI Mockup

Figure 104: Dashboard (Doctor) Mobile UI Mockup

- UI and User Flow Mockup - Dashboard (Health Official) / Desktop & Tablet
- UI and User Flow Mockup - Dashboard (Health Official) / Mobile
- ▸ Prototype - Dashboard (Health Official) / Desktop & Tablet
- ▸ Prototype - Dashboard (Health Official) / Mobile

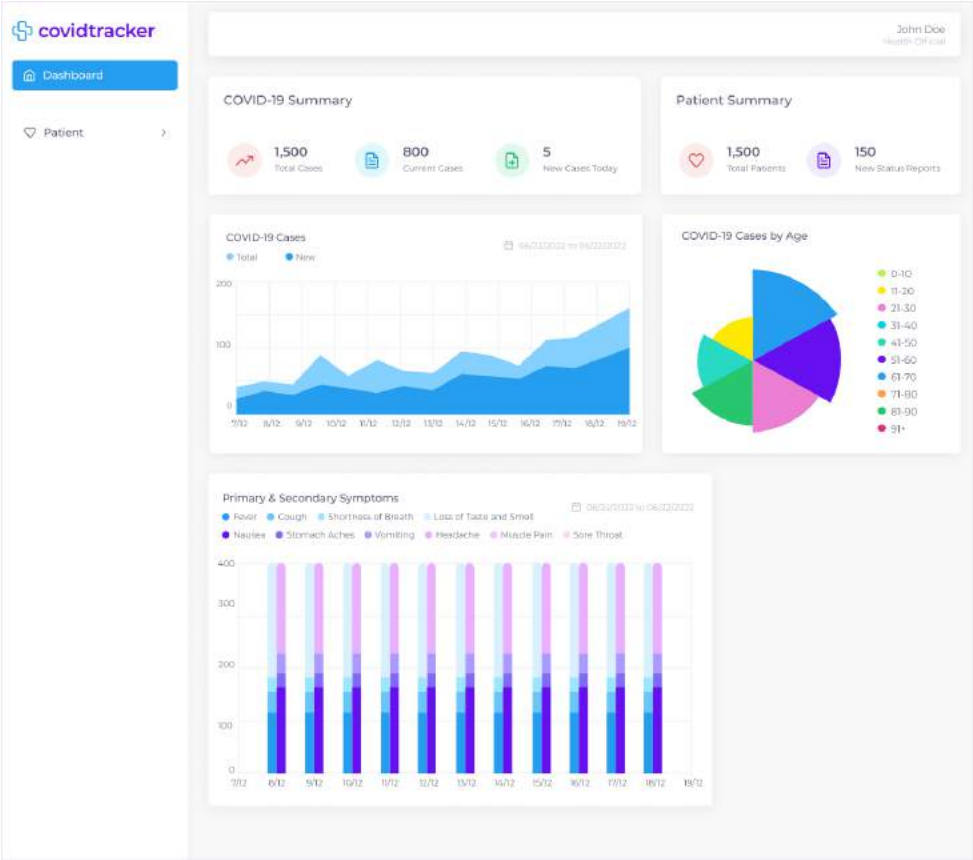


Figure 105: Dashboard (Health Official) Desktop & Tablet UI Mockup

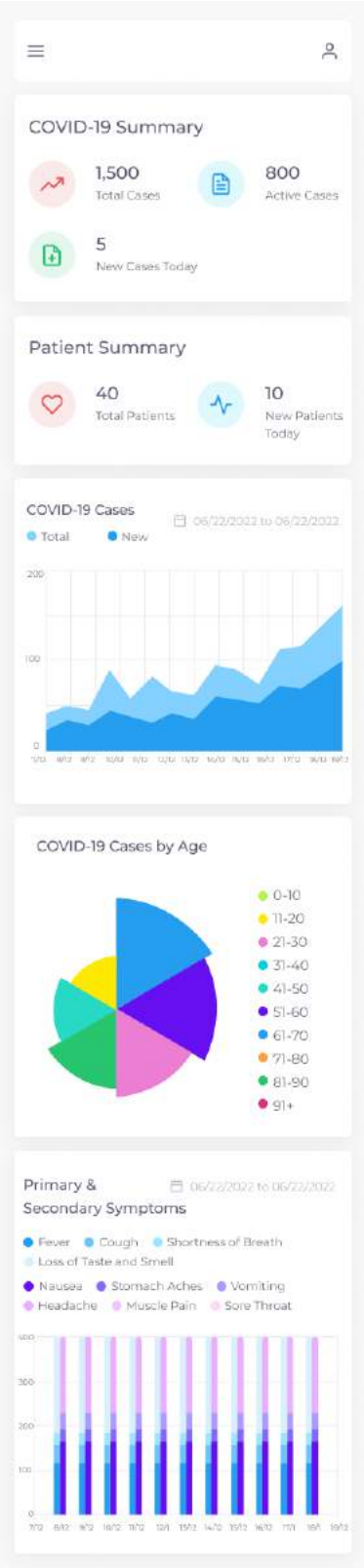


Figure 106: Dashboard (Health Official) Mobile UI Mockup

- UI and User Flow Mockup - Dashboard (Patient) / Desktop & Tablet
- UI and User Flow Mockup - Dashboard (Patient) / Mobile
- ▸ Prototype - Dashboard (Patient) / Desktop & Tablet
- ▸ Prototype - Dashboard (Patient) / Mobile

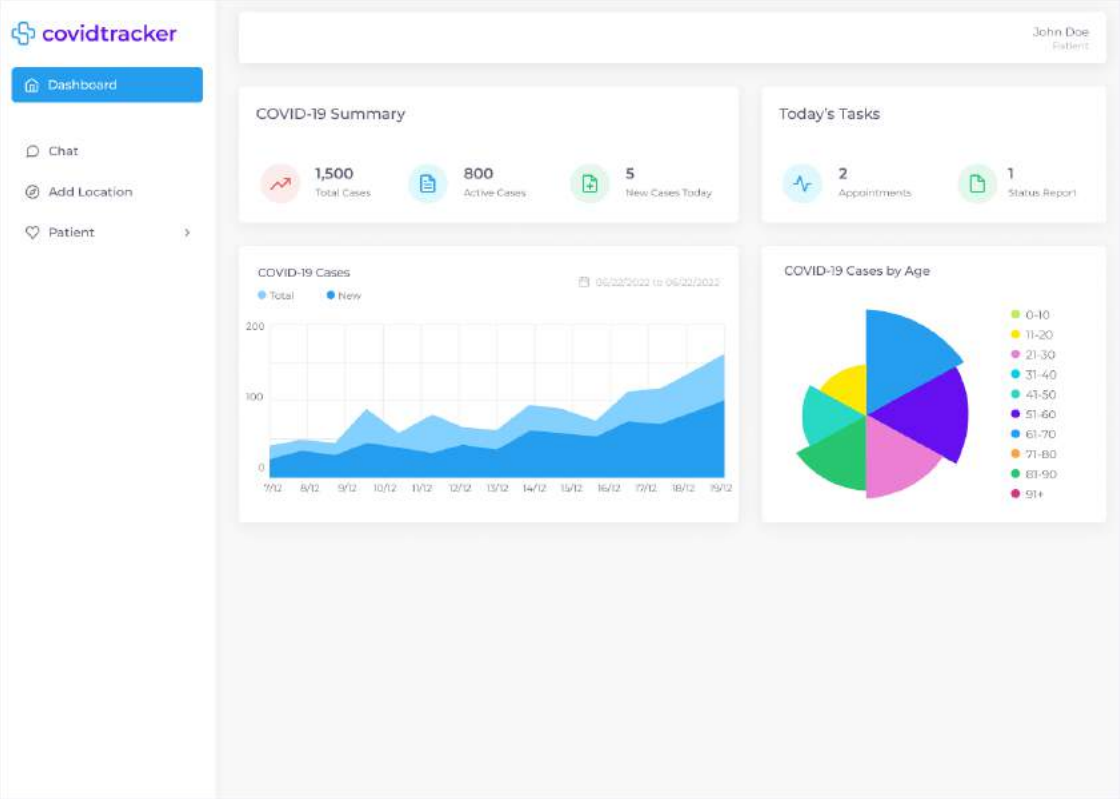


Figure 107: Dashboard (Patient) Desktop & Tablet UI Mockup

Figure 108: Dashboard (Patient) Mobile UI Mockup

- UI and User Flow Mockup - Dashboard (Immigration Officer) / Desktop & Tablet
- UI and User Flow Mockup - Dashboard (Immigration Officer) / Mobile
- ▸ Prototype - Dashboard (Immigration Officer) / Desktop & Tablet
- ▸ Prototype - Dashboard (Immigration Officer) / Mobile

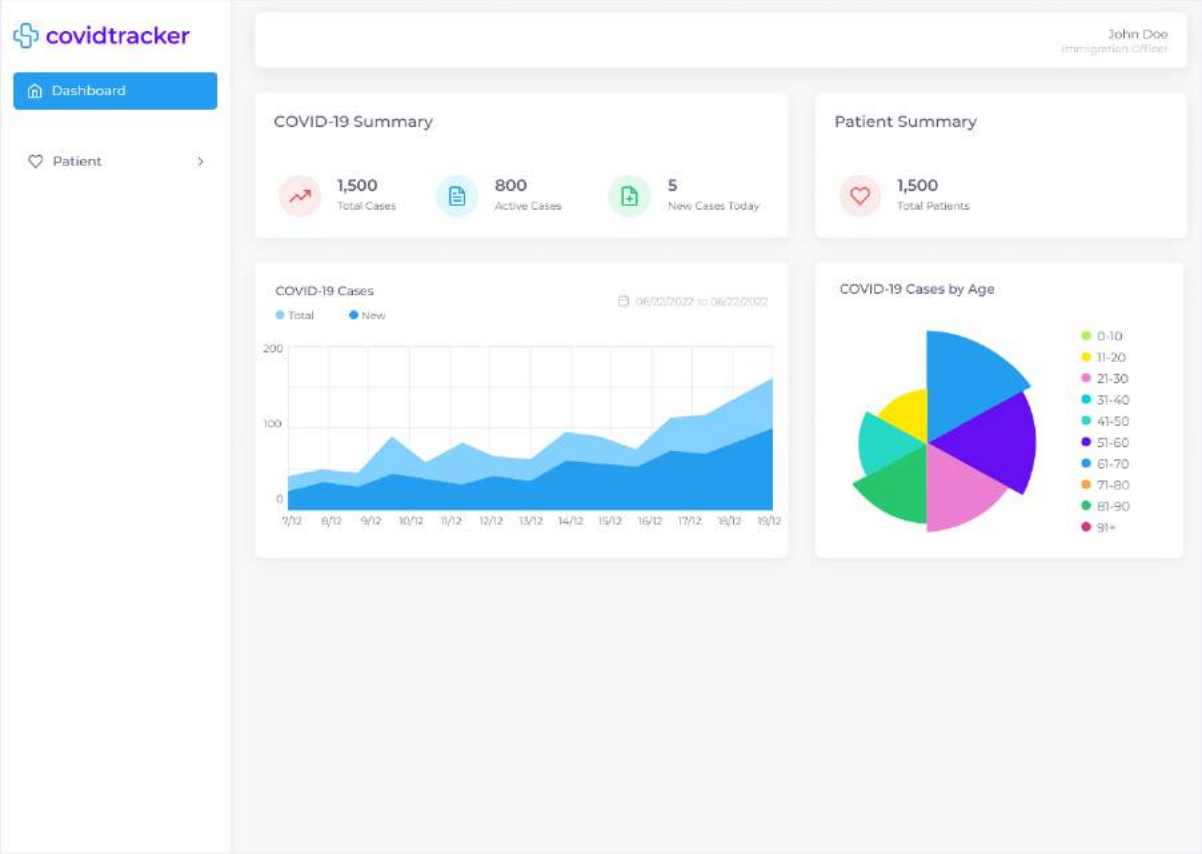


Figure 109: Dashboard (Immigration Officer) Desktop & Tablet UI Mockup

Figure 110: Dashboard (Immigration Officer) Mobile UI Mockup

8.0 TESTING PLAN AND REPORT

8.1 Unit Tests

8.1.1 Client

Unit tests for the client are automated tests that are run through the CI/CD pipeline on every pull request and commit on the main branch. These tests can be run using the command `npm run test`. We will also be using snapshot testing which will render our front end javascript into HTML and save it in a file. We can then compare it later to make sure no unintended changes were made to the rendered HTML.

All unit tests, including snapshot tests, for the front end will be using the Jest testing framework. We chose this framework because it has the best support for snapshot testing which is the primary way we will be unit testing front end components.

8.1.2 Server

Unit tests for the server are automated tests that are run through the CI/CD pipeline on every pull request and commit on the main branch. These tests can be run using the command `npm run test:unit`.

All unit tests for the server will be using the mocha testing framework and the sinon library to generate spies, mocks, fakes, and stubs. We chose mocha because it has the best support for TypeScript testing suites and integrates well with chai - our assertion library - and sinon which allows us to creates spies, mocks, fakes, and stubs extremely easy so there is little to no boilerplate required when writing unit tests.

A generated unit test report of the system is depicted in the following figure.

appointment_controller.ts

/tests/unit/controllers/appointment_controller.ts

AppointmentController::postAppointment

/tests/unit/controllers/appointment_controller.ts

5ms 3 3

✓ should call res.json with 201	3ms
✓ should call res.json with 500	0ms
✓ should call res.json with 400	2ms

AppointmentController::getAppointments

/tests/unit/controllers/appointment_controller.ts

1ms 2 2

✓ should call res.json with 200	1ms
✓ should call res.json with 500	0ms

authentication_controller.ts

/tests/unit/controllers/authentication_controller.ts

AuthenticationController::signUp

/tests/unit/controllers/authentication_controller.ts

8ms 7 7

✓ should return jwt token when sign up is successful	3ms
✓ should return status 500 if service throws an error	1ms
✓ should return status 400 when phone number is incorrect length	1ms
✓ should return status 400 when gender is not in enum	1ms
✓ should return status 400 when email is missing	1ms
✓ should return status 400 when password is missing	0ms
✓ should return status 400 when password does not follow correct format	1ms

AuthenticationController::signIn

/tests/unit/controllers/authentication_controller.ts

1ms 4 4

✓ should return jwt token when authentication is successful	0ms
✓ should return status 500 if service throws an error	1ms
✓ should return status 400 when email is missing	0ms
✓ should return status 400 when password is missing	0ms

base_controller.ts

/tests/unit/controllers/base_controller.ts

BaseController::index

/tests/unit/controllers/base_controller.ts

0ms 1 ✓1

✓	should call res.json with 200 status ok	0ms

dashboard_controller.ts

/tests/unit/controllers/dashboard_controller.ts

DashboardController::getDashboards

/tests/unit/controllers/dashboard_controller.ts

1ms 2 ✓2

✓	should call res.json with 200	1ms
✓	should call res.json with 500	0ms

doctor_controller.ts

/tests/unit/controllers/doctor_controller.ts

DoctorController::setStatusFields

/tests/unit/controllers/doctor_controller.ts

0ms 2 ✓2

✓	should return status 200 with correct patient counts	0ms
✓	should return status 500 if service throws an error	0ms

location_report_controller.ts

/tests/unit/controllers/location_report_controller.ts

LocationReportController::postLocationReport

/tests/unit/controllers/location_report_controller.ts

0ms 3 ✓3

✓	should return status 201 if request is valid	0ms

✓	should return status 400 if data is invalid	0ms
✓	should return status 500 if service throws an error	0ms

message_controller.ts

/tests/unit/controllers/message_controller.ts

MessageController::postMessage

/tests/unit/controllers/message_controller.ts

3ms 3 ✓ 3

✓	should return status 201 if request is valid	1ms
✓	should return status 400 if data is invalid	1ms
✓	should return status 500 if service throws an error	1ms

MessageController::getMessages

/tests/unit/controllers/message_controller.ts

2ms 3 ✓ 3

✓	should return status 200 if request is valid	1ms
✓	should return status 400 if data is invalid	1ms
✓	should return status 500 if service throws an error	0ms

MessageController::getChats

/tests/unit/controllers/message_controller.ts

2ms 2 ✓ 2

✓	should return status 200 if request is valid	1ms
✓	should return status 500 if service throws an error	1ms

notification_controller.ts

/tests/unit/controllers/notification_controller.ts

NotificationController::postSMS

/tests/unit/controllers/notification_controller.ts

1ms 3 ✓ 3

✓	should return status 201 if request is valid	1ms
✓	should return status 400 if data is invalid	0ms
✓	should return status 500 if service throws an error	0ms

NotificationController::postEmail

/tests/unit/controllers/notification_controller.ts

3ms 3 3

✓	should return status 201 if request is valid	1ms
✓	should return status 400 if data is invalid	1ms
✓	should return status 500 if service throws an error	1ms

patient_controller.ts

/tests/unit/controllers/patient_controller.ts

PatientController::assignDoctor

/tests/unit/controllers/patient_controller.ts

0ms 4 4

✓	should assign doctorId to the given patient	0ms
✓	should return status 400 if doctorId is not passed	0ms
✓	should return status 400 if patientId is not passed	0ms
✓	should return status 500 if service throws an error	0ms

PatientController::getPatients

/tests/unit/controllers/patient_controller.ts

1ms 3 3

✓	should return status 200 if data is valid	0ms
✓	should return status 500 if service throws error	0ms
✓	should return status 400 if data is invalid	1ms

PatientController::putPatientPrioritized

/tests/unit/controllers/patient_controller.ts

2ms 3 3

✓	should return status 200 if data is valid	1ms
✓	should return status 500 if service throws error	1ms
✓	should return status 400 if data is invalid	0ms

status_controller.ts

/tests/unit/controllers/status_controller.ts

StatusController::postStatusFields

/tests/unit/controllers/status_controller.ts

3ms 5 5

✓ should return status 201 when no errors	1ms
✓ should return status 400 when no patient id	0ms
✓ should return status 400 when no fields	1ms
✓ should return status 400 when a field has a non boolean type	1ms
✓ should return status 500 if service throws an error	0ms

StatusController::getStatusFields

/tests/unit/controllers/status_controller.ts

1ms 4 4

✓ return status 200 when no errors	0ms
✓ return status 400 when patientId is not passed	0ms
✓ return status 400 when invalid patientId is passed	0ms
✓ return status 500 when service throws an error	1ms

StatusController::postStatus

/tests/unit/controllers/status_controller.ts

1ms 4 4

✓ should return status 201 when no errors	0ms
✓ should return status 400 when no patient id	1ms
✓ should return status 400 when no status	0ms
✓ should return status 500 if service throws an error	0ms

StatusController::getStatus

/tests/unit/controllers/status_controller.ts

0ms 3 3

✓ should return status 200 when no errors	0ms
✓ should return status 400 when data is wrong	0ms
✓ should return status 500 if service throws an error	0ms

StatusController::getStatusForPatient

/tests/unit/controllers/status_controller.ts

0ms 3 3

✓ should return status 200 when no errors	0ms
✓ should return status 400 when data is wrong	0ms
✓ should return status 500 if service throws an error	0ms

StatusController::getStatuses

/tests/unit/controllers/status_controller.ts

0ms 2 2

✓	should return status 200 when no errors	0ms
✓	should return status 500 if service throws an error	0ms

StatusController::putStatusReviewed

/tests/unit/controllers/status_controller.ts

1ms 3 3

✓	should return status 204 when no errors	1ms
✓	should return status 400 when data is invalid	0ms
✓	should return status 500 if service throws an error	0ms

test_controller.ts

/tests/unit/controllers/test_controller.ts

test_controller::getTestResult

/tests/unit/controllers/test_controller.ts

0ms 3 3

✓	should return status 201 if no error	0ms
✓	should return status 500 if service throws error	0ms
✓	should return status 400 if body is not as expected	0ms

test_controller::getPatientTests

/tests/unit/controllers/test_controller.ts

1ms 3 3

✓	should return status 201 if no error	0ms
✓	should return status 500 if service throws error	0ms
✓	should return status 400 if body is not as expected	1ms

test_controller::postTestResult

/tests/unit/controllers/test_controller.ts

3ms 3 3

✓	should return status 201 if no error	1ms
✓	should return status 400 if body is not as expected	1ms
✓	should return status 500 if service throws error	1ms

user_controller.ts

/tests/unit/controllers/user_controller.ts

UserController::me

/tests/unit/controllers/user_controller.ts

0ms 2 2

✓	should return user when user is found by id	0ms
✓	should return status 500 if service throws an error	0ms

UserController::getUser

/tests/unit/controllers/user_controller.ts

0ms 3 3

✓	should return user when user is found by id	0ms
✓	should return status 400 if data invalid	0ms
✓	should return status 500 if service throws an error	0ms

UserController::assignRole

/tests/unit/controllers/user_controller.ts

2ms 5 5

✓	should return status 204 if no errors	0ms
✓	should return status 400 if userId is not a number	0ms
✓	should return status 400 if role is not valid	1ms
✓	should return status 400 if role is not passed	1ms
✓	should return status 500 if service throws an error	0ms

8.2 Integration Tests

8.2.1 Client

Integration tests for the client are automated tests that are run through the CI/CD pipeline on every pull request and commit on the main branch. These tests can be run using the command `npm run test`. All integration tests for the front end will be using the Jest testing framework. We chose Jest for the same reasons mentioned in the above section.

These tests use a mock API that returns mock server responses to test the integration between all the client side code and the server API.

8.2.2 Server

Integration tests for the server are automated tests that are run through the CI/CD pipeline on every pull request and commit on the main branch. These tests can be run using the command `npm run test:integration`.

All integration tests for the server will be using the mocha testing framework and supertest in order to create a callable instance of our web server. We chose to use supertest because it provides the easiest integration with our web framework library.

These tests use a database to test the integration between all the server side code and the database implementation.

8.3 Acceptance Tests

Acceptance tests will be documented and run manually to show an entire flow of the application. These tests will use the client to interface with the server which will persist the data in the database. All these tests will be based on the user stories to ensure that all user flows work as specified by the requirements.

These tests can be automated using a tool like Selenium in order to mock a real user interacting with the full system. The acceptance tests are also written in Gherkin Syntax which is a behavioral driven development syntax that allows us to define our tests in terms of user state and behavior.

Below are the current acceptance tests for the system.

AT-1	**COV-42 - As a User, I was to be able to sign up, so that I can access the apps features**	
Preconditions	New user without an account is accessing the website	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN I am on the sign up page	Sign up page should be displayed	
AND I input all required fields with valid data on the Personal page	No form errors should be displayed	- First name - Last name - Phone - Gender - Date of birth - Address - City - Postal Code - Province
AND I click the Next button	The Account page of the Sign Up form should be shown	
AND I input all required fields with valid data on the Account page	No form errors should be displayed	- Email - Password - Confirm Password
WHEN I click the Sign Up button		
THEN my account should be created		
AND I should be logged in		
AND I should be redirected to the main screen.		
Result	**PASS**	

Table 8: Acceptance Test for COV-42

AT-2	**COV-48 - As a User, I want to be able to sign in, so that I can access my account**	
Preconditions	New user with an account is accessing the website	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the sign in page	Sign in page should be displayed	
AND that I input my valid email	No form errors should be displayed	email: doctor@test.com
AND that I input my valid password		password: Test123!
WHEN I click the Sign In button		
THEN I should be logged into the site		
AND my session should persist		
AND I should be redirected to the main screen		
Result	**PASS**	

Table 9: Acceptance Test for COV-48

AT-3	**COV-52 -As a User, I want to be able to sign out so, that I can delete my session**	
Preconditions	User is logged in on the site	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on a page with a navbar	Navbar should be displayed	
AND that I am signed in	User info should be displayed on the top right of the webpage	
WHEN I click the Sign Out button		
THEN I should be logged out of the site		
AND my session should be deleted		
AND I should be redirected to the sign in page		
Result	**PASS**	

Table 10: Acceptance Test for COV-52

AT-4	**COV-85 - As an Administrator, I want to assign a role to a User, so that I can manage access rights**	
Preconditions	User is logged in on the site as an admin	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the assign role page	Assign role page should be displayed	

AND that I am signed in as an admin	User info should be displayed on the top right of the webpage	
AND I inputted a valid user id that has no current role	No form errors should be displayed	userId: 1
AND I selected a role from the dropdown		role: PATIENT
WHEN I click the “Add a Role” button		
THEN the role should be assigned to the user		
AND I should receive a confirmation of my action	Should see a green confirmation message at the top right of the screen	
Result	**PASS**	

Table 11: Acceptance Test for COV-85

AT-5	**COV-26 - As an Administrator, I want to assign a Patient to a Doctor, so that I can manage the Patients**	
Preconditions	User is logged in on the site as an admin	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the assign doctor page	Assign doctor page should be displayed	
AND that I am signed in as an admin	User info should be displayed on the top right of the webpage	
AND I input a valid patient id that has no doctor assigned	No form errors should be displayed	patientId: 2
AND I input a valid doctorId		doctorId: 5
WHEN I click the “Assign a Patient” button		

THEN the patient should be assigned to the doctor		
AND I should receive a confirmation of my action	Should see a green confirmation message at the top right of the screen	
Result	**PASS**	

Table 12: Acceptance Test for COV-26

AT-6	**COV-95 - As a Doctor, I want to define the status report fields for my Patients, so I can properly track them**	
Preconditions	User is logged in on the site as a doctor	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the define status report fields page	Assign doctor page should be displayed	
AND that I am signed in as a doctor	User info should be displayed on the top right of the webpage	
AND I inputted a valid patient id that is assigned to the doctor	No form errors should be displayed	patient: 3
AND I selected the fields to assign		- Fever - Cough - Nausea
WHEN I click the “Define Status Report” button		
THEN the patient should be assigned the status report fields		

AND I should receive a confirmation of my action.	Should see a green confirmation message at the top right of the screen	
Result	**PASS**	

Table 13: Acceptance Test for COV-95

AT-7	**COV-25 - As a Patient, I want to submit my status, so that I can keep my Doctor updated**	
Preconditions	User is logged in on the site as a patient	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the submit status page	Submit status page should be displayed	
AND that I am signed in as a patient	User info should be displayed on the top right of the webpage	
AND I inputted the required fields	No form errors should be displayed	- temperature - weight - other symptoms
WHEN I click the “Submit” button		
THEN the my status report should be submitted		
AND I should receive a confirmation of my action.		
Result	**PASS**	

Table 14: Acceptance Test for COV-25

AT-8	**COV-27 - As an Administrator, I want to view the number of Patients assigned to a Doctor, so that no Doctor has too many Patients**	
Preconditions	User is logged in on the site as an admin	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the view patients count page	Submit status page should be displayed	
AND that I am signed in as an admin	User info should be displayed on the top right of the webpage	
THEN I should be able to view the patient count per doctor		
AND the total number of assigned patients		
AND the average number of patients per doctor.		
Result	**PASS**	

Table 15: Acceptance Test for COV-27

AT-9	**COV-107 - As a Health Official, I want to input COVID test results, so that I can report if a Patient tested positive or negative**	
Preconditions	User is logged in on the site as a health official	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the submit test result page	Submit test result page should be displayed	
AND that I am signed in as a health official	User info should be displayed on the top right of the webpage	
WHEN I input the test result of a patient	No form errors should be displayed	Test result: POSITIVE Type of Test: PCR

AND the location of the test		- Address - City - Postal Code - Province
AND the date of the test		Date of Test: select a date
THEN the result should be persisted in the database for that patient		
AND I should get a confirmation of my action		
Result	**PASS**	

Table 16: Acceptance Test for COV-107

AT-10	**COV-112 - As a Patient, I want to view the details of a single status report of a Patient, so that I can view their progress at a point in time**	
Preconditions	User is logged in on the site as a patient	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the status report details page	Status report details page should be displayed	
AND that I am signed in as a patient	User info should be displayed on the top right of the webpage	
AND the status report is mine	Page should be accessible to user	
THEN I should be able to view the details of the status report.		
Result	**PASS**	

Table 17: Acceptance Test for COV-112

AT-11	**COV-124 - As a Patient, I want to view the details of a single COVID test result, so that I'm aware of my diagnosis**	
Preconditions	User is logged in on the site as a patient	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the test result details page	Test result details page should be displayed	
AND that I am signed in as a patient	User info should be displayed on the top right of the webpage	
AND the test result is mine	Page should be accessible to user	
THEN I should be able to view the details of the test result		
Result	**PASS**	

Table 18: Acceptance Test for COV-124

AT-12	**COV-123 - As a Patient, I want to view all my line item COVID test results, so that I'm aware of my diagnosis**	
Preconditions	User is logged in on the site as a patient	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the test results page	Test results page should be displayed	
AND that I am signed in as a patient	User info should be displayed on the top right of the webpage	
THEN I should be able to view the line item details of my latest results		

AND click into any of them to view more details about a given test results	User should be redirected to the details page for that test result	
Result	**PASS**	

Table 19: Acceptance Test for COV-123

AT-13	**COV-111 - As a Patient, I want to view all my line item statuses, so that I can monitor my progress over time**	
Preconditions	User is logged in on the site as a patient	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the status reports page	Status reports page should be displayed	
AND that I am signed in as a patient	User info should be displayed on the top right of the webpage	
THEN I should be able to view the line item details of my status reports		
AND click into any of them to view more details about a given status report	User should be redirected to the details page for that status report	
Result	**PASS**	

Table 20: Acceptance Test for COV-111

AT-14	**COV-157 - As a Doctor, I want to view a list of my Patients, so that I can easily navigate to their specific detailed views**	
Preconditions	User is logged in on the site as a doctor	
Acceptance Criteria	**Expected Result**	**Input Data**

GIVEN that I am on the patient list page	Patient list page should be displayed	
AND that I am signed in as a doctor	User info should be displayed on the top right of the webpage	
THEN I should be able to view a table of my patients		
AND actions associated to each patient	The actions button should be clickable displaying the possible actions - Add Test Result - Test Results - Status Reports	
Result	**PASS**	

Table 21: Acceptance Test for COV-157

AT-15	**COV-113 - As a Doctor, I want to view a line item list of my patients with their most recent line item status update, so that I can keep track of any updates**	
Preconditions	User is logged in on the site as a doctor	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the status report inbox page	Status report inbox page should be displayed	
AND that I am signed in as a doctor	User info should be displayed on the top right of the webpage	
THEN I should be able to view the line item status reports of all my patients sorted in descending order by date		

Result	**PASS**

Table 22: Acceptance Test for COV-113

AT-16	**COV-115 - As a Doctor, I want to mark a Patient's status update as "Reviewed", so that I can see which statuses I've already seen**	
Preconditions	User is logged in on the site as a doctor and there is at least status not marked as reviewed and there is at least 1 status marked as reviewed	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the status report inbox page	Status report inbox page should be displayed	
AND that I am signed in as a doctor	User info should be displayed on the top right of the webpage	
THEN I should be able to mark a status report as reviewed		Mark flag with boolean value isReviewed: false
AND I should be able to mark a reviewed status report as unreviewed		Mark flag with boolean value isReviewed: true
Result	**PASS**	

Table 23: Acceptance Test for COV-115

AT-17	**COV-114 - As a Doctor, I want to flag certain patients, so that their updates are prioritized over others**	
Preconditions	User is logged in on the site as a Doctor	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the patient list page	The patient list should be displayed	

AND that I am signed in as a doctor	User info should be displayed at the top of the page	
THEN I should be able to mark a patient as prioritized	flag should turn red on line item	Mark flag with boolean value isFlaged: true
AND I should be able to mark a prioritized patient as unprioritized	flag should go from red to white on line item	Mark flag with boolean value isFlaged: false
Result	**PASS**	

Table 24: Acceptance Test for COV-114

AT-18	**COV-121 - As a Patient, I want to be able to generate a QR code for a status report, so that I can share it with others**	
Preconditions	User is logged in on the site as a Patient	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the status report details page	The status report page should be displayed	
AND that I am signed in as a patient	User info should be displayed at the top of the page	
AND the test result is mine		
THEN I should be able to view and scan a qr code that can be shared with others which links to the status report page	should see a qr code on the screen that scans for my test result	
Result	**PASS**	

Table 25: Acceptance Test for COV-121

AT-19	**COV-122 - As a Patient, I want to be able to generate a QR code for a lab test result, so that I can share it with others**	
Preconditions	User is logged in on the site as a Patient	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the test result details page	The status report page should be displayed	
AND that I am signed in as a patient	User info should be displayed at the top of the page	
AND the test result is mine		
THEN I should be able to view and scan a qr code that can be shared with others which links to the test result page	should see a qr code on the screen that scans for my test result	
Result	**PASS**	

Table 26: Acceptance Test for COV-122

AT-20	**COV-108 - As a Patient, I want to update my status for the day after already submitting, so that my Doctor stays up to date**	
Preconditions	User is logged in on the site as a Patient	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the submit status report page	The status report page should be displayed	
AND that I am signed in as a patient	User info should be displayed at the top of the page	
AND I have already submitted a status report in the same calendar day		

WHEN I fill out the form with updated status info		- Fever - Cough - Nausea
AND click submit	no form error should be displayed	
THEN my new status should be persisted in the database		
AND the status report should be viewable by my doctor	Should see a green confirmation message at the top right of the screen	
Result	**PASS**	

Table 27: Acceptance Test for COV-108

AT-21	**COV-171 - As a Health Official, I want to view a list of all patients who have tested positive in the last [x] days, so that I can contract trace them**	
Preconditions	User is logged in on the site as a Health Official	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the contact trace page	The contact trace page should be displayed	
AND that I am signed in as a Health Official	User info should be displayed at the top of the page	
WHEN I select a date range from the calendar input	The date range should be displayed in the input	- From (February 1st) - To (March 1st)
THEN the patients who tested positive in that date range should appear	The positive users should be displayed	
Result	**PASS**	

Table 28: Acceptance Test for COV-171

AT-22	**COV-119 - As a Patient, I want to direct message my Doctor, so that I can ask them questions**	
Preconditions	User is logged in on the site as a Patient	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the chat page	The chat page should be displayed	
AND that I am signed in as a patient	User info should be displayed at the top of the page	
AND I have been assigned a doctor	The patients doctor should be displayed as a contact on the left hand panel	
WHEN I input a message		- “Test message text”
AND click the send button	no form error should be displayed	
THEN my message should be sent to my doctor		
AND the message should be visible in my chat page	Should see the message in the chat with the doctor	
Result	**PASS**	

Table 29: Acceptance Test for COV-119

AT-23	**COV-116 - As a Doctor, I want to book an appointment with a Patient, so that we can discuss their symptoms**	
Preconditions	User is logged in on the site as a Doctor	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the create appointment page	The create appointment page should be displayed	

AND that I am signed in as a Doctor	User info should be displayed at the top of the page	
WHEN I fill out the form with the appointment info		- Start time (March 1st 1PM) - End time (March 1st 2PM) - Address - City - Postal Code - Province
AND click submit	no form error should be displayed	
THEN the appointment should be persisted in the database	Should see a green confirmation message at the top right of the screen	
AND the patient should receive a notification for the appointment	The notification will be by text and email to make sure the patient sees the information	
Result	**PASS**	

Table 30: Acceptance Test for COV-116

AT-24	**COV-172 - As a Patient, I want to add the locations of where I have been during the day, so that I can be contact traced if I come in contact with someone that has tested positive with COVID-19**	
Preconditions	User is logged in on the site as a Patient	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the location report report page	The location report page should be displayed	
AND that I am signed in as a patient	User info should be displayed at the top of the page	
WHEN I fill out the form with a location and select the date		- Date (March 1st) - Address

		- City - Postal Code - Province
AND click submit	no form error should be displayed	
THEN my location report should be persisted in the database		
AND I should see a confirmation of my action	Should see a green confirmation message at the top right of the screen	
Result	**PASS**	

Table 31: Acceptance Test for COV-172

AT-25	**COV-169 - As a Doctor, I want to view my appointments, so that I can schedule myself**	
Preconditions	User is logged in on the site as a Doctor	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the view appointments page	The view appointments page should be displayed	
AND that I am signed in as a Doctor	User info should be displayed at the top of the page	
AND I have at least 1 appointment booked with a patient		
THEN I should see a list of my appointments with the details	The list should be sorted by date in descending order so the newest appointments are at the top	
Result	**PASS**	

Table 32: Acceptance Test for COV-169

AT-26	**COV-126 - As a Health Official, I want to contact trace who a Patient has been in contact with in the last [x] days, so that I can manage who is at risk**	
Preconditions	User is logged in on the site as a Health Official	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the contact trace page	The contact trace page should be displayed	
AND that I am signed in as a Health Official	User info should be displayed at the top of the page	
AND I select a date range from the calendar input	The date range should be displayed in the input	- From (February 1st) - To (March 1st)
WHEN I click on a patients contact trace button		
THEN I should see a list of patients they have been in contact with		
AND I should be able to adjust the date range for the results	The list should auto update as the date range is updated so the list is dynamic with the selection	- From (February 1st) - To (March 1st)
Result	**PASS**	

Table 33: Acceptance Test for COV-126

AT-27	**COV-120 - As a Patient, I want to mark my message with a priority level, so that my Doctor will view it quicker**	
Preconditions	User is logged in on the site as a Patient	
Acceptance Criteria	**Expected Result**	**Input Data**

GIVEN that I am on the chat page	The chat page should be displayed	
AND that I am signed in as a patient	User info should be displayed at the top of the page	
AND I have been assigned a doctor	The patients doctor should be displayed as a contact on the left hand panel	
WHEN I input a message		- “Test message text”
AND mark the message as priority	Priority flag should turn red when checked	- Check priority flag
AND click the send button	no form error should be displayed	
THEN my message should be sent to my doctor	The message text should display as red	
AND the message should be visible in my chat page	Should see the message in the chat with the doctor	
Result	**PASS**	

Table 34: Acceptance Test for COV-120

AT-28	**COV-222 - As a Doctor, I want to define status report fields from the patient list page**	
Preconditions	User is logged in on the site as a Doctor	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the patient list page	The patient list page should be displayed	
AND that I am signed in as a Doctor	User info should be displayed at the top of the page	
WHEN I click on the action items		

AND select define status fields	The browser will redirect to the define status report fields page for that patients	
THEN I should be able to define the status report fields for that patient without inputting the patientId	Should be able to define the status report fields for that patient directly without inputting the patientId as normal	
Result	**PASS**	

Table 35: Acceptance Test for COV-222

AT-29	**COV-223 - As an Immigration Officer, I want to view the patients list, so I can be aware of test results and prioritize them**	
Preconditions	User is logged in on the site as an Immigration Officer	
Acceptance Criteria	**Expected Result**	**Input Data**
GIVEN that I am on the patient list page	The patient list page should be displayed	
AND that I am signed in as a Immigration Officer	User info should be displayed at the top of the page	
THEN I should see limited patient information		
AND limited action items for each patient	The Immigration Officer should only see a list of the patients test results	
Result	**PASS**	

Table 36: Acceptance Test for COV-223

AT-30	**COV-125 - As a User, I want to a dashboard personalized for my role, so I can have an overview of the system**
Preconditions	User is logged in on the site as a User

Acceptance Criteria	Expected Result	Input Data
GIVEN that I am on the dashboard page	The dashboard page should be displayed	
AND that I am signed in as a User	User info should be displayed at the top of the page	
THEN personalized dashboard should be displayed	Dashboard should only contain information the user is allowed to see	
AND the summary widgets should be displayed	Should see at least 2 personalized summary widgets	
AND the chart widgets should be displayed	Should see at least 2 chart widgets	
Result	**PASS**	

Table 37: Acceptance Test for COV-125

8.4 System Tests

System tests will be documented and run manually to show an entire flow of the application. These tests will use the client to interface with the server which will persist the data in the database. All these tests will be based on the user stories to ensure that all user flows work as specified by the requirements

These tests can be automated using a tool like Selenium in order to mock a real user interacting with the full system.

Below are the current system tests for the system.

ST-1	**COV-42 - As a User, I was to be able to sign up, so that I can access the apps features**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the sign up page (relative url “/sign_up”) 2. Fill all required fields with valid inputs 3. Click the Sign Up Button	1. You should see the sign up page 2. The form should not give any input errors 3. The form should not give any input errors, your account should be created, you should be signed in, and you should be redirected to the main screen	2. First name Last name Phone Gender Date of birth Address City Postal Code Province 3. Email Password Confirm Password
Result	**PASS**	

Table 38: System Test for COV-42

ST-2	**COV-48 - As a User I want to be able to sign in, so that I can access my account**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the sign in page (relative url “/sign_in”) 2. Input your email and password 3. Click the Sign In Button	1. You should see the sign in page 2. The form should not give any input errors 3. The form should not give any input errors, you should be signed in, and you should be redirected to the main screen	2. email: doctor@test.com password: Test123!
Result	**PASS**	

Table 39: System Test for COV-48

ST-3	**COV-52 - As a User I want to be able to sign out, so that I can delete my session**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to a page where the navbar can be seen 2. Click the Sign Out Button	1. You should see the sign out button in the navbar 2. You should be signed out, your session should be deleted, and you should be redirected to the Sign In page	
Result	**PASS**	

Table 40: System Test for COV-52

ST-4	**COV-85 - As an Administrator, I want to assign a role to a User, so that I can manage access rights**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the assign role page (relative url “/assign_role”) 2. Input the user id and select a role 3. Click the “Add a Role” Button	1. You should see the assign role form 2. The form should not give any input errors 3. The form should not give any input errors, the user should be assigned the role, and the page should display a confirmation message for your action	2. userId: 1 role: PATIENT
Result	**PASS**	

Table 41: System Test for COV-85

ST-5	**COV-26 - As an Administrator, I want to assign a Patient to a Doctor, so that I can manage the Patients**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the assign doctor page (relative url “/assign_doctor”) 2. Input the patient id and the doctor id 3. Click the “Assign a Patient” Button	1. You should see the assign doctor form 2. The form should not give any input errors 3. The form should not give any input errors, the patient should be assigned the doctor, and the page should display a confirmation message for your action	2. patientId: 2 doctorId: 5
Result	**PASS**	

Table 42: System Test for COV-26

ST-6	**COV-95 - As a Doctor, I want to define the status report fields for my Patients, so I can properly track them**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the define status report page (relative url “/define_status_report”) 2. Input the patient id 3. Select the checkboxes	1. You should see the define status report form 2. The form should not give any input errors 3. The form should not give any input errors 4. The patient should be assigned the status fields and the page should display a confirmation message for your action	2. patientId: 3 3. Fever: boolean checkbox Cough: boolean checkbox Nosea: boolean checkbox

for each status field you wish the patient to input 4. Click the “Define Status Report” Button		
Result	**PASS**	

Table 43: System Test for COV-95

ST-7	**COV-25 - As a Patient, I want to submit my status, so that I can keep my Doctor updated**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the status report page (relative url “/status_report”) 2. Input the required fields based on your symptoms 3. Click the “Submit” Button	1. You should see the status report form 2. The form should not give any input errors 3. The status report should be submitted and the page should display a confirmation message for your action	2. Temperature: number Weight: number Other Symptoms: text
Result	**PASS**	

Table 44: System Test for COV-25

ST-8	**COV-27 - As an Administrator, I want to view the number of Patients assigned to a Doctor, so that no Doctor has too many Patients**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the patients assigned page (relative url “/patients_assigned”)	1. You should see the total number of patients assigned, the average number of patients, and a table view of the number of patients assigned to each doctor	
Result	**PASS**	

Table 45: System Test for COV-27

ST-9	**COV-107 - As a Health Official, I want to input COVID test results, so that I can report if a Patient tested positive or negative**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the status report page (relative url "/add_test/patients/:patientId") 2. Input the required fields based on the test result 3. Click the “Submit” Button	1. You should see the add test result form 2. The form should not give any input errors 3. The test result should be submitted and the page should display a confirmation message for your action	2. Test result: POSITIVE Type of Test: PCR Address City Postal Code Province Date of Test: select a date
Result	**PASS**	

Table 46: System Test for COV-107

ST-10	**COV-112 - As a Patient, I want to view the details of a single status report of a Patient, so that I can view their progress at a point in time**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the page (relative url "/statuses/:statusId")	1. You should see the details of the given status report based on the id in the parameters	
Result	**PASS**	

Table 47: System Test for COV-112

ST-11	**COV-124 - As a Patient, I want to view the details of a single COVID test result, so that I'm aware of my diagnosis**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the page (relative url "/tests/:testId")	1. You should see the details of the given test result based on the id in the parameters	
Result	**PASS**	

Table 48: System Test for COV-124

ST-12	**COV-123 - As a Patient, I want to view all my line item COVID test results, so that I'm aware of my diagnosis**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the page (relative url	1. You should see the the line item test results of the patient whose id is in the URL	

"/tests/patients/:patientId")		
Result	**PASS**	

Table 49: System Test for COV-123

ST-13	**COV-111 - As a Patient, I want to view all my line item statuses, so that I can monitor my progress over time**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the page (relative url "/statuses/patients/:patientId")	1. You should see the the line item status reports of the patient whose id is in the URL	
Result	**PASS**	

Table 50: System Test for COV-111

ST-14	**COV-157 - As a Doctor, I want to view a list of my Patients, so that I can easily navigate to their specific detailed views**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the patient list page (relative url “/patients”) 2. Click on the actions button (three dots)	1. You should see a line item list of patient assigned to the doctor 2. You should see a list of available actions for that user including a. Add a test result b. Test results c. Status reports	
Result	**PASS**	

Table 51: System Test for COV-157

ST-15	**COV-113 - As a Doctor, I want to view a line item list of my patients with their most recent line item status update, so that I can keep track of any updates**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the status report inbox page (relative url "/statuses/inbox") 2. Click on the “eye” icon	1. You should see the line item status reports of all the patients assigned to the doctor sorted in descending order by date 2. The app will redirect you to the status report details page for that status	
Result	**PASS**	

Table 52: System Test for COV-113

ST-16	**COV-115 - As a Doctor, I want to mark a Patient's status update as "Reviewed", so that I can see which statuses I've already seen**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the status report inbox page (relative url "/statuses/inbox") 2. Click on the checkbox by a patient's status 3. Click a checked checkbox	1. You should see the line item status reports of all the patients assigned to the doctor sorted in descending order by date 2. The status will be marked as reviewed and the app will give confirmation of your action 3. The status will be marked as unreviewed and the app will give confirmation of your action	2. Mark flag with boolean value isReviewed: true 3. Mark flag with boolean value isReviewed: false

Result	**PASS**

Table 53: System Test for COV-115

ST-17	**COV-114 - As a Doctor, I want to flag certain patients, so that their updates are prioritized over others**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the patient list page (relative url “/patients”) 2. Click on the flag by a patient 3. Click a red flag	1. You should see a line item list of patient assigned to the doctor 2. The patient will be marked as prioritized and the app will give confirmation of your action 3. The patient will be marked as unprioritized and the app will give confirmation of your action	2. Mark flag with boolean value isFlaged: true 3. Mark flag with boolean value isFlaged: false
Result	**PASS**	

Table 54: System Test for COV-114

ST-18	**COV-121 - As a Patient, I want to be able to generate a QR code for a status report, so that I can share it with others**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the page (relative url "/statuses/:statusId") 2. Scan the qr code with a qr code reader	1. You should see the details of the given status report based on the id in the parameters and a qr code associated with that status 2. You should be redirected to the status report page though the qr code reader	

Result	**PASS**

Table 55: System Test for COV-121

ST-19	**COV-122 - As a Patient, I want to be able to generate a QR code for a lab test result, so that I can share it with others**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the page (relative url "/tests/:testId ") 2. Scan the qr code with a qr code reader	1. You should see the details of the given test result based on the id in the parameters and a qr code associated with that test result 2. You should be redirected to the test result page though the qr code reader	
Result	**PASS**	

Table 56: System Test for COV-122

ST-20	**COV-108 - As a Patient, I want to update my status for the day after already submitting, so that my Doctor stays up to date**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the status report page (relative url “/status_report”) 2. Input the required fields based on your symptoms	1. You should see the status report form 2. The form should not give any input errors 3. The status report should be submitted and the page should display a confirmation message for your action 4. The status report should be submitted properly and a confirmation message should be displayed	2. Fever Cough Nausea

3. Click the “Submit” Button 4. Repeat steps 1-3 on the same calendar day		
Result	**PASS**	

Table 57: System Test for COV-108

ST-21	**COV-171 - As a Health Official, I want to view a list of all patients who have tested positive in the last [x] days, so that I can contract trace them**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the contact tracing page (relative url “/contact_tracing”) 2. Select a date range	1. You should see a list of patients who are currently positive with covid 2. The list of patients should automatically update to the new patients who have tested positive in that date range	2. From (Feb 1st) To (March 1st)
Result	**PASS**	

Table 58: System Test for COV-171

ST-22	**COV-119 - As a Patient, I want to direct message my Doctor, so that I can ask them questions**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the chat page (relative url “/chat”)	1. You should see the chat app 2. The message should be displayed in the input box	2. “Test message text”

2. Input message 3. Click the “Send” Button	3. The message should be sent to the recipient and you should see the message in the chat for that user	
Result	**PASS**	

Table 59: System Test for COV-119

ST-23	**COV-116 - As a Doctor, I want to book an appointment with a Patient, so that we can discuss their symptoms**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the patients list page page (relative url “/patients”) 2. Click on the action items for a patient then book appointment 3. On the book appointment page input the data for the appointment 4. Click the submit button	1. You should see a list of your patients 2. You should see the book appointment page for that user 3. The form should not display any errors for the input 4. You should see a confirmation of your action and the patient should receive a notification for the appointment	3. Start date (Feb 1st 1PM) End date (Feb 1st 2PM) Address City Postal Code Province
Result	**PASS**	

Table 60: System Test for COV-116

ST-24	**COV-172 - As a Patient, I want to add the locations of where I have been during the day, so that I can be contact traced if I come in contact with someone that has tested positive with COVID-19**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the location report page (relative url "/location_report") 2. Input the address and date fields 3. Click the "Submit" Button	1. You should see the add location report form 2. The form should not give any input errors 3. The location report report should be submitted properly and a confirmation message should be displayed	2. Date ex: March 1st Address City Postal Code Province
Result	**PASS**	

Table 61: System Test for COV-172

ST-25	**COV-169 - As a Doctor, I want to view my appointments, so that I can schedule myself**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the appointments page (relative url "/appointments")	1. You should see a list of appointments for that user sorted in descending order by date	
Result	**PASS**	

Table 62: System Test for COV-169

ST-26	**COV-126 - As a Health Official, I want to contact trace who a Patient has been in contact with in the last [x] days, so that I can manage who is at risk**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the contact tracing page (relative url “/contact_tracing”) 2. Select a date range 3. Click on the contact trace icon for a given patient 4. Select a date range	1. You should see a list of patients who are currently positive with covid 2. The list of patients should automatically update to the new patients who have tested positive in that date range 3. You should see a list of patients that patient has been in contact with 4. The list should be dynamically updated to display only the contacts within that date range	2. From (Feb 1st) To (March 1st) 4. From (Feb 1st) To (March 1st)
Result	**PASS**	

Table 63: System Test for COV-126

ST-27	**COV-120 - As a Patient, I want to mark my message with a priority level, so that my Doctor will view it quicker**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the chat page (relative url “/chat”) 2. Input message 3. Mark the message as priority	1. You should see the chat app 2. The message should be displayed in the input box 3. The priority flag should turn red 4. The message should be sent to the recipient and you should see the message in the chat for that user	2. “Test message text” 4. Priority flag should be true

4. Click the “Send” Button		
Result	**PASS**	

Table 64: System Test for COV-120

ST-28	**COV-222 - As a Doctor, I want to define status report fields from the patient list page**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the patients list page page (relative url “/patients”) 2. Click on the action items for a patient then define status field	1. You should see a list of your patients 2. You should see the define status fields page for that user and should be able to define the fields and submit without directly inputting the patients id	
Result	**PASS**	

Table 65: System Test for COV-222

ST-29	**COV-223 - As an Immigration Officer, I want to view the patients list, so I can be aware of test results and prioritize them**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the patients list page page (relative url “/patients”)	1. You should see a list of your patients 2. You should see only the option to view a list of covid tests for that patient and no other actions	

2. Click on the action items for a patient		
Result	**PASS**	

Table 66: System Test for COV-223

ST-30	**COV-125 - As a User, I want to a dashboard personalized for my role, so I can have an overview of the system**	
Steps to reproduce	**Expected output for each step**	**Input Data**
1. Navigate to the dashboard page (relative url “/dashboard”)	1. You should see the dashboard for the user which contains the personalized summary widgets and general chart widgets which contain information about the general population. Some roles will see data specific to their patients that other roles will not see. Every user role has a unique dashboard.	
Result	**PASS**	

Table 67: System Test for COV-125

8.5 Test Code Coverage

8.5.1 Client

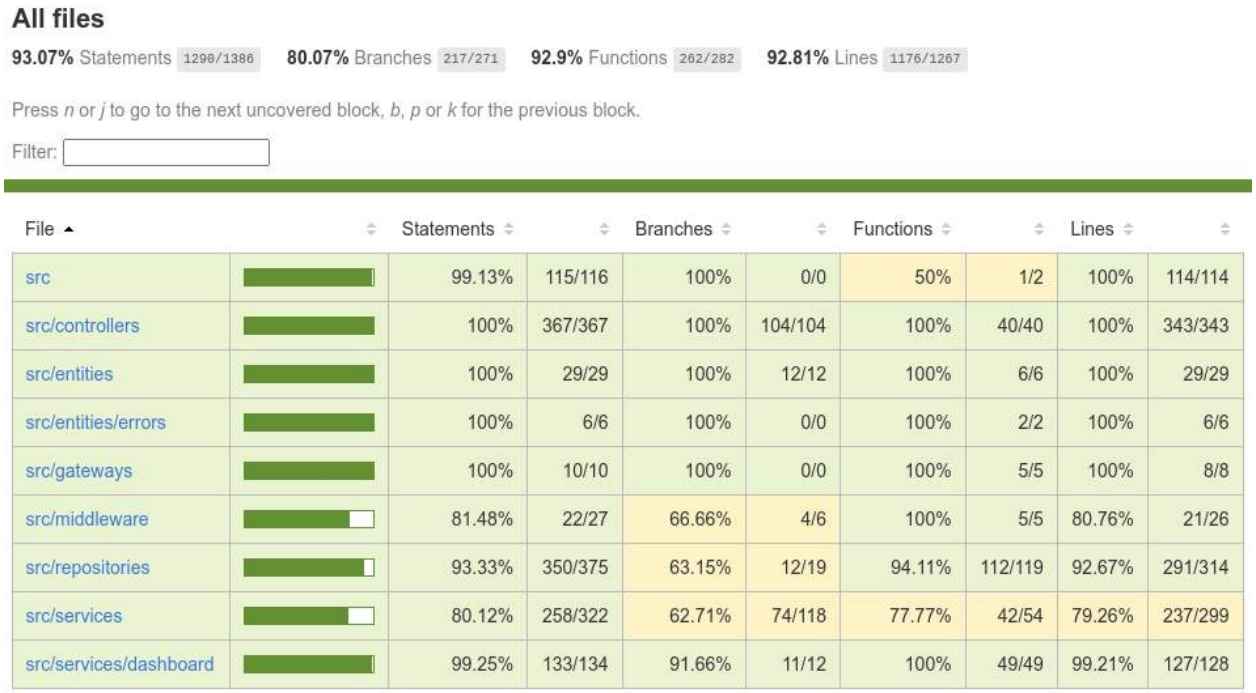
A report of the code coverage can be generated by running the command `npm run test --coverage` this will produce a coverage report of the client side code.

8.5.2 Server

A report of the code coverage can be generated by running the command `npm run test:coverage` this will produce a coverage report of the server side code.

A code coverage report of the server side code is depicted in the following figure. NYC/Istanbul was used to compute the code coverage. It reports coverage by folder,

you can then click into the folder and view the other folders coverage or individual file coverage. Then you can open a specific file to view line by line coverage reports.



All files

93.07% Statements 1290/1386 **80.07%** Branches 217/271 **92.9%** Functions 262/282 **92.81%** Lines 1176/1267

Press *n* or *j* to go to the next uncovered block, *b*, *p* or *k* for the previous block.

Filter:

File	Statements		Branches		Functions		Lines	
src	99.13%	115/116	100%	0/0	50%	1/2	100%	114/114
src/controllers	100%	367/367	100%	104/104	100%	40/40	100%	343/343
src/entities	100%	29/29	100%	12/12	100%	6/6	100%	29/29
src/entities/errors	100%	6/6	100%	0/0	100%	2/2	100%	6/6
src/gateways	100%	10/10	100%	0/0	100%	5/5	100%	8/8
src/middleware	81.48%	22/27	66.66%	4/6	100%	5/5	80.76%	21/26
src/repositories	93.33%	350/375	63.15%	12/19	94.11%	112/119	92.67%	291/314
src/services	80.12%	258/322	62.71%	74/118	77.77%	42/54	79.26%	237/299
src/services/dashboard	99.25%	133/134	91.66%	11/12	100%	49/49	99.21%	127/128

Figure 111: Code Coverage Report of Server Side Code

9.0 DEFECT TRACKING AND REPORT

Sprint 4 had no new bugs and didn’t include any bug fixes as there are no active defects in the system The team continued to focus on testing each user story thoroughly and as a result we were able to catch bugs before merging in and closing the user stories. The defect tracking report for the project can be viewed on the next page.

T	Key	Summary	Story point estimate	P	Risk	Parent	Start date	Due	Status	Assignee(s)	Description
	COV-194	Fix mobile navbar not showing up on screen sizes below 1200px	1		High	Authentication and Authorization	28/Feb/22	28/Feb/22	DONE	Jason Gerard	The mobile navbar should appear at 1200px and should match the same fields as is seen in the mockups and prototypes. The bug maybe as simple as the navbar button not appearing but may also be caused by a misconfiguration in the layouts.
	COV-145	The drop down inputs in the sign in form are pre-selected when they should not have a default value	1		Low	Authentication and Authorization	04/Feb/22	04/Feb/22	DONE	Jason Gerard	Currently the sign in form has some drop down input for selecting the gender and province. These drop downs have pre-selected values so that no validation needs to take place. This should be fixed so that there is no value as default and be required to have a user input the selection.
	COV-144	The navbar shows a placeholder name instead of the logged in users name	1		Low	Authentication and Authorization	06/Feb/22	06/Feb/22	DONE	Jason Gerard	Currently when a user logs in the navbar will show a placeholder first and last name. This should show the logged in users first and last name. After the sign in and sign up requests start returning the users name in the payload and saving it in the redux store we can update the navbar to fetch this data from the store and display it.

10.0 QUALITY MEASUREMENTS

This section depicts and describes the various metrics being used, the cause of the results and how the results can be improved.

10.1 Metrics Used

The following metrics are being used:

- **Statement coverage**: Checks to see if each statement in the program has been executed while running the test suite.
- **Branch coverage:** Checks to see if all conditional branches (if statement and terinaries) are covered while running the test suite.
- **Function coverage:** Checks to see if every function in the source code was called at least once while running the test suite.
- **Line coverage:** Checks if each physical line in the source code has been executed at least once while running the test suite. This is mostly covered by statement coverage, which is generally superior because it ignores coding styles better, but we are including it for completeness of all generated metrics.
- **Linting errors:** Errors we receive if we run our automated linting package, ESLint. The types of errors and severity are defined by our .eslintrc file. This includes checks against many things, primarily language standards.
- **Formatting errors:** Errors we receive if we run our automated formatting package, prettier. This package has defined an opinionated formatting standard that can automatically be applied to most code, but some must still be manually formatted to follow the standard.

10.2 Cause of Results

There aren't any formatting and linting errors because the CI/CD pipeline will fail if any are detected resulting in them immediately being fixed.

Test coverage has improved since sprint 3 and we continue to implement integration and unit tests for each user story which makes sure it is implemented properly and there are never any regressions on the applications functionality.

10.3 Improving the Results

In sprint 4 the team made a big push to not only match the current code coverage standards but bump many of the categories above 90%. The branch coverage is still lagging behind the rest and can be improved but is sitting a bit over 80% which is well within a good margin.

	Sprint 1	**Sprint 2**	**Sprint 3**	**Sprint 4**	**Sprint 5**
Statement Coverage	65.6%	87.33%	89.11%	93.07%	
Branch Coverage	55.55%	77.52%	71.5%	80.07%	
Function Coverage	41.66%	81.7%	88.63%	92.9%	
Line Coverage	64.34%	86.94%	88.73%	92.81%	
Linting Errors	0	0	0	0	
Formatting Errors	0	0	0	0	

Table 68: Test Coverage for Each Sprint

APPENDIX A: TEAM COLLABORATION AND COMMUNICATION

Stakeholders use a set of tools to collaborate and communicate throughout the project lifecycle.

A.1 Collaboration

- **Google Suite (Docs, Drive, Sheets)**:
  G Suite is a collection of business, productivity, collaboration, and education software developed and powered by Google. The primary G Suite tools include Gmail, Drive, Docs, Sheets, Slides, Forms, Calendar, Google+, Sites, Hangouts, and Keep. [2] Google Suite is used for documentation since it is widely accessible and available to all development team members.
- **GitHub:** GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere. We use github to be able to work on different sections of the code at the same time and have a version control. [3]

A.2. Communication

- **Discord:** Discord is a free voice, video, and text chat app that's used by tens of millions of people ages 13+ to talk and hang out with their communities and friends. [6] Discord is used for communication and meetings among development team members. Voice and text channels are named according to the different development team groups (i.e. back end, front end and UI design).
- **Slack:** Slack is a messaging app for business that connects people to the information they need. By bringing people together to work as one unified team, Slack transforms the way organizations communicate. [5] Slack is used to communicate with the product owners when clarification is needed or to schedule meetings.

- **Zoom:** Zoom is a cloud-based video conferencing platform that can be used for video conferencing meetings, audio conferencing, webinars, meeting recordings, and live chat. [1] Zoom is used for meetings with the product owners.

A.3 Tools

- **Issue and project tracking tool:** Jira
  https://www.atlassian.com/software/jira.
  Jira is a software application used for issue tracking and project management. The tool has become widely used by agile development teams to track bugs, stories, epics, and other tasks. [4]
- **Diagram modeling tool:** Draw.io
  https://app.diagrams.net/
  Draw.io is an online diagram editor that enables you to create flowcharts, UML, entity relation, network diagrams, mockups and more.
- **User interface design and prototyping tool:** Figma
  https://www.figma.com/
  Figma is a UI and UX design application, with excellent design, prototyping, and code-generation tools. It's arguably the industry's leading interface design tool, with robust features which support teams working on every phase of the design process.

APPENDIX B: GLOSSARY

- **Application Programming Interface (API):** An application programming interface (API) is a computing interface which defines interactions between multiple software intermediaries. It defines the kinds of calls or requests that can be made, how to make them, the data formats that should be used, the conventions to follow, etc. [7]
- **Logical Layered Architecture:** Layered architecture is an architecture pattern that promotes high cohesion and low coupling through separation of concerns by layers. Each layer depends on the layer below it.
- **UML Domain Model:** A conceptual view of the domain represented through UML classes and relationships. [8]
- **Risk Management:** Practice of identifying, evaluating, and preventing or mitigating risks to a project that have the potential to impact the desired outcomes.
- **Database:** Databases store aggregations of data records or files that contain information, such as sales transactions, customer data, financials and product information. [9]
- **UI prototype:** User interface prototyping is an iterative analysis technique in which users are actively involved in the mocking-up of the UI for a system. [10]
- **UI/UX mockup :** A mockup is a static wireframe that includes more stylistic and visual UI details to present a realistic model of what the final page or application will look like. [11]
- **CI/CD pipeline:** Series of steps that must be performed in order to deliver a new version of software. Continuous integration/continuous delivery (CI/CD) pipelines are a practice focused on improving software delivery using either a DevOps or site reliability engineering (SRE) approach. [12]

REFERENCES

1. Barron, Sophia. “Everything You Need to Know about Using Zoom.” *Owl Labs Blog*, https://resources.owllabs.com/blog/zoom#:~:text=Zoom%20is%20a%20cloud%2Dbased,meeting%20recordings%2C%20and%20live%20chat.
2. Decker, Allie. “The Ultimate Guide to G Suite.” *HubSpot Blog*, 31 Mar. 2019, https://blog.hubspot.com/marketing/google-suite#:~:text=G%20Suite%20is%20a%20collection,Google%20Apps%20for%20Your%20Domain%E2%80%9D.
3. “Hello World.” *GitHub Docs*, https://docs.github.com/en/get-started/quickstart/hello-world.
4. “Jira.” *ProductPlan*, 9 Feb. 2021, https://www.productplan.com/glossary/jira/.
5. Slack. “What Is Slack?” *Slack Help Center*, https://slack.com/help/articles/115004071768-What-is-Slack-#:~:text=Slack%20is%20a%20messaging%20app,transforms%20the%20way%20organizations%20communicate.
6. “What Is Discord: A Guide for Parents and Educators.” *Discord*, https://discord.com/safety/360044149331-What-is-Discord#:~:text=Discord%20is%20a%20free%20voice,with%20their%20communities%20and%20friends.&text=The%20vast%20majority%20of%20servers,touch%20and%20spend%20time%20together.
7. Business Standard. “What Is API, API Definition, API News.” *Business Standard*, https://www.business-standard.com/about/what-is-api.
8. *UML Class Diagrams as a Conceptual Models*, http://www.cs.sjsu.edu/~pearce/modules/lectures/ooa/domain/domainModels.htm
9. Lutkevich, Ben, and Adam Hughes. “What Is a Database? Definition from Searchdatamanagement.” *SearchDataManagement*, TechTarget, 27 Sept. 2021,

https://searchdatamanagement.techtarget.com/definition/database#:~:text=Computer%20databases%20typically%20store%20aggregations,data%2C%20financials%20and%20product%20information.&text=They%20collect%20information%20on%20people,can%20be%20observed%20and%20analyzed.

10. *User Interface (UI) Prototypes: An Agile Introduction*, http://agilemodeling.com/artifacts/uiPrototype.htm.
11. "Wireframes vs Mockups: Determining the Right Level of Fidelity for Your Project." *Wireframes vs Mockups Explained | Lucidchart Blog*, 27 Feb. 2020, https://www.lucidchart.com/blog/wireframes-vs-mockups.
12. "What Is a CI/CD Pipeline?" *Red Hat - We Make Open Source Technologies for the Enterprise*, https://www.redhat.com/en/topics/devops/what-cicd-pipeline.