

# Software Engineering

## GROUP 20

Name	ID	Email
Tudor-Nicolae Cotoc	2059730	txc933@student.bham.ac.uk
Patrisia Maria Pascan	2064061	pxp962@student.bham.ac.uk
Medeea-Ramona Zamciuc	2058546	mxz947@student.bham.ac.uk
Vijay Padmakumar	2096908	vxp908@student.bham.ac.uk
Siddharth Thakur	2085585	sxt986@student.bham.ac.uk
Shlok Verma	2055638	sxv938@student.bham.ac.uk
Sumeet Singh	2008315	sxs1687@student.bham.ac.uk
Bharath Routhu	2069145	brr945@student.bham.ac.uk
Samuel Irsai	2021783	sxi983@student.bham.ac.uk
Rahul Gheewala	2066546	rxg949@student.bham.ac.uk

**COVIDAid**  
**2020/21**



**University of Birmingham**

School of Computer Science



# TABLE OF CONTENTS

A. Requirements Engineering	4
A1. Proposal of System	4
Introduction	4
Scope	5
Assumptions	6
Summary	7
A2. Functional and Non-Functional Requirements	7
Functional Requirements	7
Non-Functional Requirements	10
B. Software Design with UML	13
B1. Use Case Diagram	13
B2. Two Non-Trivial Use Cases	13
Use Case 1: Getting guidance when suspecting you might be infected with COVID	13
Use Case 2: Using the COVID Map in order to see the effect of the pandemic	15
B3. Two Non-Trivial Scenarios	16
Scenario 1: COVIDIndex	16
Scenario 2: COVIDIndex	17
Scenario 3: COVID Map	18
Scenario 4: COVID Map	18
B4. Activity Diagram	20
B5. Class Analysis	21
Noun Analysis	21
Verb Analysis	23
Responsibility-Driven Analysis	25
First Cut Class Diagram	28
Full Class Diagrams	29
B6. Object Diagram	30
B7. Two Sequence Diagrams	31
Sequence Diagram for the COVIDIndex feature	31
Sequence Diagram for the Positive with COVID Survey	32
B8. Two State Diagrams State Diagram for the Form Class	33

State Diagram for the COVIDMap Class _____	34
<b>C. Software Architecture Style, Modelling and Evaluation _____</b>	<b>35</b>
C1. Two Possible Architecture Styles _____	35
First Candidate – Component Diagram _____	35
Second Candidate – Component Diagram: _____	36
C2. Two Deployment Diagrams First Candidate – Deployment Diagram: _____	37
Second Candidate – Deployment Diagram: _____	38
C3. Comparison of Architectures Trade-off Analysis _____	39
First Candidate: _____	39
Second Candidate: _____	40
Conclusion: _____	40
<b>D. Software Testing _____</b>	<b>41</b>
Introduction _____	41
Testing Objectives _____	41
Test Items _____	42
Frontend facing items _____	42
Backend facing items _____	42
Features To Be Tested _____	42
Features Not To Be Tested _____	43
Approach _____	43
Test Deliverables _____	44
Testing tools _____	44
Defect/Bug Triage _____	45
Pass Fail Criteria _____	46
Exit Criteria _____	62
Assumption _____	62
<b>E. Usability and Prototyping _____</b>	<b>63</b>
<b>F. Ethics and professional practice _____</b>	<b>69</b>
Code of Conduct for COVIDAid: _____	69
Respecting Privacy. _____	69
Hold safety paramount. _____	69
Treat all persons fairly and equally. _____	69
Act as a faithful agent. _____	69
Professional review. _____	71

# A. REQUIREMENTS ENGINEERING

## A1. Proposal of System

### Introduction

The aim of this project is to create an accessible, easy-to-use and secure application called COVIDAid that provides the users with the most important COVID-related information such as personalized medical advice, the newest guidelines from the government and the evolution of the pandemic. The application is mainly designed for smartphones, but it should have a web view that allows access on computers as well.

Besides centralizing all information related to COVID into one single app, two unique features of this app are the Positive with COVID Survey and COVIDIndex.

The Positive with COVID Survey is a feature that allows users that have been confirmed positive to describe their symptoms and get the corresponding medical advice 24/7, regardless of whether they have access to a GP or a hospital at the moment. By completing the survey, they also contribute to updating the Symptoms Database which is used as part of our second major feature, COVIDIndex. Furthermore, the data collected from patients can be extremely helpful in further research regarding COVID-19, effectively giving every person the opportunity to make a difference in the fight against this virus.

COVIDIndex is a tool that gives the user the probability of them having COVID based on their symptoms. This is made possible by an algorithm based on a Machine Learning Model trained on the Symptoms Database built with the data collected from the Positive with COVID Survey. Because the database is linked with the Positive with COVID Survey, each time a new user completes the survey the Machine Learning Model improves and returns more accurate results. With this, they can get a realistic feel of their condition, accurate medical advice and advice on what they should do next.

The app is not designed to replace medical care, but rather serve as a guideline for what patients need to do to manage their condition effectively if their symptoms are not too severe and when they need to seek in-person medical help. Its main purpose is to help the medical system from getting overwhelmed in these challenging times and provide a basis for future studies into this novel virus.



## Perks

- The app provides advice 24/7 so the users can have access to medical feedback regardless of whether they can contact their GP or go to the hospital.
- The app prevents the overcrowding of hospitals by bringing the most useful information right on the users' phones.
- The app is useful for people who are not aware of the symptoms that COVID-19 can trigger and that are not necessarily respiratory problems.
- The app collects the most recent information about COVID-19 into a single app, allowing users to access anything they need with just a few taps.
- The app can potentially make a major difference in areas where access to COVID-19 tests is limited.

## Scope

When using the app for the first time, the user is prompted to choose a language for the app from the available list: English, Hindi, Polish, German, Chinese, French.

The application's graphical user interface displays multiple tabs dedicated to the main features of the system: COVIDIndex, Positive with COVID Survey, COVID Map, Helpful Guides.

The application allows the user to log in if they have already created an account and presents them the option to create one if not. Account creation requires minimum information from the user, for privacy reasons. They need to provide a username, a password and, optionally, an email address or a phone number for password recovery reasons. The app must check all the login credentials are valid and not already in use. If they are all in order, a new entry is created for the current user in the account database.

All users are allowed access to the COVID Map and Helpful Guides tabs; however, they must be logged in if they want to use the COVIDIndex or want to take Positive with COVID Survey. If the user accesses the COVIDIndex, they are presented with a form they need to complete in order to receive a prediction of their likelihood of having COVID. The form contains three sections, namely Mandatory Information, Symptoms and Personal Information. Before submitting, the form first needs to pass a correctness check to verify all the required fields are completed. After submitting, the form goes through a security check to verify there is no contradictory information. The form is added to the forms database and linked with the user's account. The information is then sent to the algorithm to be processed. The result is displayed on the screen as a percentage, as well as further guidance based on it:

- <30%: tells the user to avoid large gatherings, to monitor their symptoms and come back to the app if they get worse or new ones develop. It will send a notification to the user after a few days to ask how their symptoms have developed.
- 30 - 50%: tells the user to avoid large gatherings and to isolate from their household if there are any people at risk. It also tells them to call their GP if they



can. It will send a notification to the user after a few days to ask how their symptoms have developed.

- 50 - 70%: tells the user to self-isolate and if they can to take a COVID test.
- >70%: tells the user that there is an extremely high chance of them being infected and that they should self-isolate, get a COVID test as soon as possible and announce the local health authorities.

The algorithm used to compute the result utilizes Machine Learning to analyse the user's symptoms based on the Symptoms Database: this one contains entries from people who have been confirmed positive and is continuously updated with entries from the Positive with COVID Survey.

The Positive with COVID Survey serves two purposes: it gives people that have been confirmed positive guidance on how to deal with the condition and updates the symptoms database with their information in order to provide more accurate COVIDIndex results. The process is largely the same: they complete the same form, an entry is created in the symptoms database and linked to their account and, finally, they are asked if they need further medical advice. If they say yes, a list of advice personalized to their symptoms and personal information is displayed on the screen. It tells them if they should stay at home or go to the hospital, when to call their GP, what medication they could take to alleviate symptoms.

The COVID Map tab displays a map of the UK and has multiple filters that can be applied to it. The COVID Spread view shows how the number of cases is distributed geographically and what the concentration is in each region. The Possible Cases view shows regions where there are a lot of incoming COVIDIndex submissions, which could mean there is a new focus of infection. Lastly, the Evolution view displays how the number of cases has evolved over time, both in each region as well as on a diagram. Each view of the map is updated daily, based on the official numbers released by the NHS (the COVID Spread view) and on the input from the users (the Possible Cases view).

The Helpful Guides tab contains links to relevant information from the NHS, articles on the latest government regulations as well as updates from the WHO.

## Assumptions

- We assume that the application is being developed in close connection with the NHS and the government. All the medical information given by the app as well as the accuracy of the algorithm must be verified by medical staff.
- We assume that the initial symptoms database will be contributed to by the NHS and will be updated afterwards according to user input.
- We assume that a new Positive with COVID Survey coming from the same user after more than 30 days from the last one means that they contracted COVID again, so a new COVID case. Thus, a new symptoms database entry will be created.



## Summary

In summary, this application is meant to grant users easy access to medical advice regarding COVID anywhere, at all times, which should be especially helpful whenever they are feeling sick, but their GP is unavailable at that time and they are unsure of whether their condition is bad enough to warrant going to the hospital. Thus, the app can help prevent the medical system from being overwhelmed by giving essential information to patients seeking medical help right on their phone or computer, allowing medical staff to focus on the more severe cases. Furthermore, given that the app receives information about the users' experiences with COVID, its databases could help serve as a base for further studies that could research in more detail how COVID affects people of different ages and ethnicity, with different medical conditions and habits.

## A2. Functional and Non-Functional Requirements

### Functional Requirements

1. The app **must** allow users to create new accounts. **(Account Creation)**
  - 1.1. The app **must** prompt the user to provide a username, a password and, optionally, an email address or a phone number for password recovery.
  - 1.2. The app **must** verify that all the provided credentials are available and have the correct format.
  - 1.3. The app **must** be capable of providing an indication of the strength of the password.
  - 1.4. The app **should** allow the user to verify their email through an external link.
  - 1.5. The app **should** allow the user to verify their phone number by providing a PIN sent via text.
2. The app **must** allow users to authenticate using previously created accounts. **(User Authentication)**
  - 2.1. The app **must** prompt the user to enter a username.
  - 2.2. The app **must** require the user to introduce the matching password.
  - 2.3. The app **could** block the user's access to the account after 5 failed attempts to log in.
  - 2.4. The app **must** allow users to change passwords.
  - 2.5. The app **must** allow users to delete their accounts.
  - 2.6. If the user provided an email address or a phone number, the app **must** allow the user to recover their password.
  - 2.7. The app **could** prompt the user to enable location services or provide a postcode.



3. If the user is logged in, the app **must** allow access to the “Positive with COVID survey” feature. **(Positive with COVID survey)**

3.1. The app **must** allow users to fill a form providing details about their symptoms and their medical condition.

3.1.1. If the user has completed the survey before in the last 30 days, the app **must** update the corresponding database entry only with the new data introduced.

3.1.2. If the user has completed the survey before but longer than 30 days ago, the app **must** add another database entry for the current user.

3.2. The app **must** update the diagnosing algorithm based on the answers submitted by the user.

3.3. The app **must** ask the user if they need further guidance.

**If they say yes:**

3.3.1. The app **must** give the user further guidance after submitting the survey, according to their symptoms.

3.3.2. The user **should** recommend the user which medications they can use to help with their symptoms.

3.3.3. The app **must** tell the user if they need to call their GP.

3.3.4. The app **must** tell the user if they need to go to the hospital.

3.3.5. If the user has provided their location, the app **could** provide a list of the nearest hospitals.

3.3.5. The app **should** ask the user to complete the survey again if their symptoms change.

3.3.6. The app **could** send the user a notification after 3 days, asking them to complete the survey again if their symptoms have changed.

4. If the user is logged in, the app **must** allow access to the “COVIDIndex” feature. **(COVIDIndex survey)**

4.1. The app **must** allow users to fill a form providing details about their symptoms and their medical condition.

4.2. The app **must** give the user a probability of having COVID based on their symptoms.

4.3. The app **must** give the user further guidance after submitting the survey, based on the result provided by the algorithm.

4.3.1. The app **must** tell the user if they need to self-isolate.

4.3.2. The app **must** tell the user if they must take a COVID test.



4.3.3. The app **could** tell the user how to contact their local health authorities.

4.3.4. The app **could** recommend the user what medication to take according to their symptoms.

4.4. The app **must** ask the user to complete the Positive with COVID survey if they test positive.

4.5. In order to get a new and more accurate prediction, the app **could** send the user a notification after 3 days, asking them to complete the survey again if their symptoms have changed.

5. The app **must** provide a form to be used by the “Positive with COVID Survey” and “COVIDIndex” features. **(Form)**

5.1. The app **must** check if the user completed the survey before.

5.1.1. If the form is used to update an existing entry, the app **should** autofill the survey with what the user submitted previously.

5.2. The app **must** display the first part of the form - Mandatory Information.

5.2.1. The app **must** check that the user selected their range of age.

5.2.2. The app **must** check that the user gave their sex.

5.2.3. The app **must** check that the user answered if they got in contact with a person positive with COVID.

5.3. The app **must** display the second part of the form - Symptoms.

5.3.1. The app **must** give the users a list of symptoms to choose from.

5.3.2. The form provided by the app **must** provide means of describing each symptom related to COVID.

5.3.3. The app **must** allow users to provide other symptoms that they think might be COVID-related.

5.4. The app **must** display the final part of the form - Personal Information.

5.4.1. The app **must** inform the user that information in this section of the form is optional.

5.4.2. The app **must** allow users to select their ethnicity.

5.4.3. The app **must** allow users to detail any other medical conditions that they might have.

5.4.3. The app **must** allow users to detail those of their habits which could have an impact on their health.

5.4.4. The app **should** allow the user to give their location only if they want to.

5.5. The app **must** save the information introduced by the user until they submit.



- 5.6. The app **must** allow the user to submit the form for it to be processed.
- 5.7. The app **must** ensure all the required fields are completed.
  - 5.7.1. The app **could** redirect the user to the required fields which were left empty.
- 6. The app **must** allow all users of the app to access the “COVID Map” feature. (**Infection Map**)
  - 6.1. The app **must** provide the user a map which shows the COVID infection rate in each region.
  - 6.2. The app **must** give the user an alternative view of the map which shows what areas other “COVIDIndex” forms are being submitted from.
  - 6.3. The app **should** let the user put in their postcode.
    - 6.2.1. The app **should** zoom in the map on the area where the user lives.
  - 6.4. The app **must** update the database for the map everyday based on the official records.
  - 6.5. The app **should** provide the user with a view of how the pandemic is evolving over time.
- 7. The app **must** allow all users of the app to access the “Helpful Guides” feature. (**Helpful Guides**)
  - 7.1. The app **must** display relevant links, guides and further information from the NHS and the government.
  - 7.2. The app **should** update the guides based on the new information released by the government.
    - 7.2.1. The app **should** provide the latest lockdown measures.
    - 7.2.2. The app **should** specify the laws that are enforced.
    - 7.2.3. The app **should** provide the latest government updates.
  - 7.3. The app **should** update the guides based on the new recommendations given by WHO.
    - 7.3.1. The app **should** provide the latest medication advice from WHO.
    - 7.3.2. The app **should** provide the latest protection advice from WHO.

## Non-Functional Requirements

### 8. Usability

- 8.1. The app **must** enable the user to select the language.
- 8.2. The languages of the app **should** be chosen such that any person living in the UK could use it.



8.3. The app **must** make use of an intuitive layout such that it takes no longer than 3 minutes for people of all ages to learn how to interact with it.

## 9. Efficiency -> Performance

9.1. The app **must** be able to confirm success or failure on a form submission in less than one second.

9.2. The app **must** provide COVIDIndex results in no more than 5 seconds.

9.3. The app **must** be able to display the COVIDMap in no more than 5 seconds, regardless of the view selected by the user.

9.4. The app **must** be able to display the Helpful Guides in no more than 5 seconds.

## 10. Efficiency -> Space

10.1. The app **should** take no more than 200MB of storage.

## 11. Reliability

11.1. The app **must** be available 99% of the time.

## 12. Portability

12.1. The app **must** be available for both Android and iOS devices.

12.2. The app **could** also have a web view.

## 13. Implementation

13.1. The app **should** be designed using the Unified Modelling Language (UML).

## 14. Security

14.1. The app **must** check the validity of the user's details upon account creation.

14.1.1. The app **must** check both the username and the password use alphanumerical characters.

14.1.2. The app **must** check the username is no longer than 20 characters.

14.1.3. The app **must** check the password has at least one uppercase character, one number and one special character.

14.1.4. The app **must** hide/show the user password while being entered.

14.1.5. If the user has provided an email, the app **must** check it is of the following format: [address@domain.com](#)

14.1.6. If the user has provided a phone number, the app **must** check it is of the following format: (+44 / 07..)

14.1.7. The app **must** check none of the user's new account details are currently in use by another account.

14.2. The app **must** check the login credentials of every user match each time they enter the app.



14.3. The app **could** provide additional means to prevent non-human use of the app (e.g. CAPTCHA).

14.4. The form **must** adhere to the following formatting rules:

- Age must be presented as a list of possible ranges: <12, 12-17, 18-24, 25-34, 35-44, 45-54, 55-64, 65-74, >75.
- Temperature (in Celsius) must also be presented as a list of possible ranges: 36-37, 37-37.5, 37.5-38, 38-38.4, >38.4
- Any discrete values (such as sex, if the user has been in contact with an infected person, pain on a scale from 1 to 10 etc.) should be presented as a list of values, allowing the user to select one or multiple accordingly.
- Any numerical values should only allow floating point numbers.
- Any other values (such as other symptoms apart from the ones presented) should be introduced in a text box by the user.

14.5. The app **must** have a system in place which checks the form does not contain any contradictory information.

## 15. Legislative -> Privacy

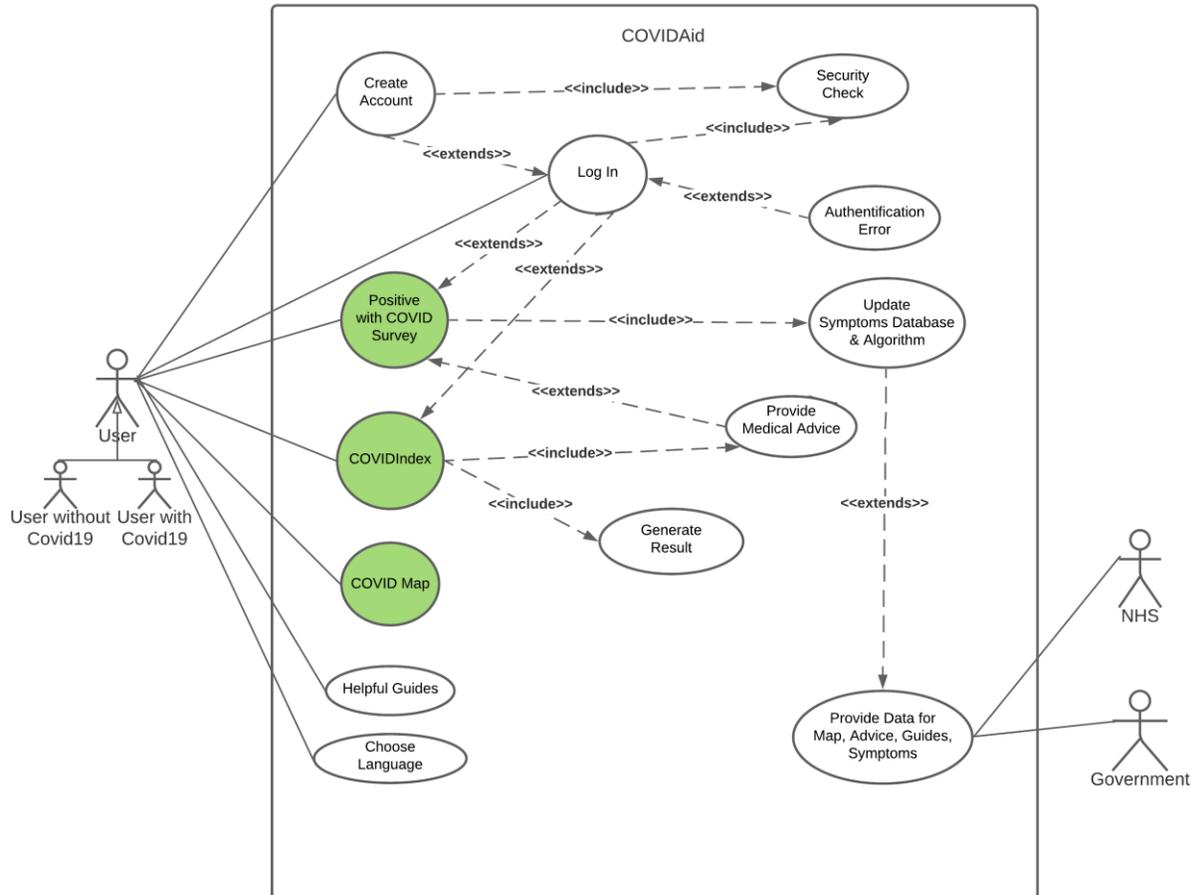
15.1. The app **must** minimize the risk of accidental disclosure of private information to third parties.

15.2. The app **must** comply with current privacy standards and laws.



# B. SOFTWARE DESIGN WITH UML

## B1. Use Case Diagram



## B2. Two Non-Trivial Use Cases

### Use Case 1: Getting guidance when suspecting you might be infected with COVID

**Description:** This use case describes how a user that suspects they might have COVID can use the COVIDIndex survey which tells them the likelihood of having COVID in order to get guidance on what they should do next.

#### Actors:

The user

#### Preconditions:

The app is connected to the internet.

The servers must be up and running.



**Basic flow of events:**

1. The use case begins when the user selects the COVIDIndex survey.
2. The app checks that the user is logged in.
3. The app displays the first part of the form: Mandatory information
4. The user selects their age, their sex, if they got in contact with a person positive with COVID, and then selects next.
5. The app displays the second part of the form: Symptoms.
6. The user selects their symptoms from the given list of symptoms and then selects next.
7. The app displays the third part of the form: Personal information (optional information)
8. The user selects what other medical conditions they have, give their location if they want to, answer questions regarding personal habits, and then select submit.
9. The app checks that the user answered all the required fields.
10. The app collects the answers given by the user and saves them in the backend.
11. The backend computes, using the diagnosing algorithm, the approximate probability of the user having COVID.
12. The app displays the approximate probability of the user having COVID and further steps that they should take based on the symptoms they have provided.
13. The use case ends successfully.

**Alternative flows:****User not logged in:**

If in step 2 of the basic flow the user is not logged in:

1. The app shall display the message: "Please log in to use this feature."
2. The user is redirected to the Log in Use Case.
3. After the user authenticates the use case returns to step 3 of the basic flow.

**REQUIRED fields not filled**

If in step 9 of the basic flow the required fields are not answered:

1. The app shall display the message: "Please fill all of the required fields."
2. The use case returns to step 3 of the basic flow.

**No response from the backend**

If in step 10 of the basic flow the backend doesn't answer:

1. The app shall display the message: "There was an error when submitting the form. Please try again later."
2. The use case returns at step 8 of the basic flow.

**Post-conditions:****Successful completion**

1. The user received the right percentage and guidance based on their symptoms.
2. The backend and the logs have been updated accordingly.



### Failure condition

1. The logs have been updated accordingly.

### Special requirements:

[SpReq:1] If the user completed the survey before, in the past month, the app should have an option to autofill the form with what they submitted last time.

[SpReq:2] The app should autosave the information introduced by the user until they submit the form.

## Use Case 2: Using the COVID Map in order to see the effect of the pandemic

**Description:** This use case describes how a user can use the COVID Map in order to see which regions are more affected by the pandemic and where new focuses of infections seem to appear.

### Actors:

The user

### Preconditions:

The app is connected to the internet.

The map is up to date.

### Basic flow of events:

1. The use case begins when the user selects the COVID Map.
2. The app displays a map of the UK, in the default view, which is **COVID Spread** view, a field where the user can put their postcode or the name of the region they want to see on the map, and a menu where they can choose the view of the map (the **COVID Spread** view, the **Possible Cases** view, the **Evolution** view).
3. The user selects the desired view of the map.
4. The map updates to the chosen view.
5. The map shows relevant information for each main region of the UK based on the chosen view.
6. The user enters their postcode or an area name.
7. The map automatically zooms in to the area the user selected.
8. The map shows detailed information for that area based on the chosen view.
9. The user leaves the map.
10. The use case ends successfully.

### Alternative flows:

#### No other view selected:

1. In step 3 the user does not choose another view of the map:
2. The use case continues from step 5 in the default view (Case Spread).

#### No postcode or area introduced:



In step 6 the user does not introduce a postcode or an area:

1. The user can manually navigate the map until they reach the desired area.
2. The use case continues from step 8 of the basic flow.

**Invalid postcode or area name:**

In step 6 the user enters an invalid postcode or area name:

1. The app displays the message: “Invalid input. Please write a correct postcode or area name.”
2. The use case returns to step 6 of the basic flow.

**Map doesn't load:**

1. The app displays the message: “An error has occurred. Please try again later.”
2. The use case ends with an error.

**Post-conditions:**

**Successful completion:**

The map worked as needed and the internal logs have been updated.

**Failure Condition:**

The user is returned to the main app and the logs have been updated accordingly.

## B3. Two Non-Trivial Scenarios

### Scenario 1: COVIDIndex

For better accuracy, the medication in this scenario was recommended by an internal medicine doctor treating acute COVID patients according to RKI guidelines.

For the past three days Miranda only had an unusual headache and diarrhea, but now she woke up with a fever of 38°C. Because one of her coworkers showed COVID symptoms two days ago, she decides to open the app, signs into her account, and proceeds to take the COVIDIndex survey.

Firstly, she fills in the questions from the Mandatory information part: “Age: [35-44]”, “Sex: Female”, “Did you get into contact with anyone positive with COVID in the past 14 days? Don't know”.

She then moves onto the Symptoms part and for each symptom that she experiences she chooses the intensity of it (for example, for Headache she can choose from “Occasional/Mild/Persistent”): “Fever: [38°C-38.4°C]”, “Headache: Persistent”, “Diarrhea”.



Afterwards, she completes from the third part of the survey, Personal information, only the fields she wants to: “Asthma”, “Smoking: Never”, “Drinking: once a month”, “Wash your hands: 2-3 times a day”, “Location: Birmingham”. She then submits the form.

The app displays the message:

“Based on your symptoms you have a chance of 65% of having COVID.

You should start self-isolating as soon as possible, call your GP to make sure that they know about your symptoms, and if you can, take a COVID test. For fever, most people can safely alleviate it with ibuprofen (unless they have kidney problems) or paracetamol (unless they have liver problems), and if these don't help, metamizole (if they do not suffer from hypotension), however **you should always contact your GP before taking any medication**. If your fever gets worse (higher than 39°C) despite taking medication, or if you develop dyspnea (shortness of breath), you should call your GP as soon as possible and go to the hospital. If your symptoms change, please complete this form again. If you take a COVID test and the result is positive, please return to the app and complete the “Positive with COVID Survey”! Thank you!”

Miranda closes the app. She is surprised by the result because she had no idea that, besides her fever, her other symptoms could also be related to COVID. She takes a paracetamol and after about an hour her temperature is lower, but her headache and digestive problems remain unchanged. Because of the result from the app, she decides to self-isolate and take a COVID test, which comes back positive.

## Scenario 2: COVIDIndex

Three days ago, Kurtis completed the COVIDIndex survey because he had a sore throat. The app gave him a low chance of having COVID (under 30%) and told him to avoid large gatherings and come back to the app if he develops new symptoms. Now, not only does his throat hurt, but he also partially lost his sense of taste and smell and one of his close friends is positive with COVID. Since he has just received a notification from the app asking him if his symptoms have changed, he decides to login to the app and complete the form again.

He chooses the option to autofill the form with his previous answers and then proceeds to change the answer to the question “Did you get into contact with anyone positive with COVID in the past 14 days?” from “No” to “Yes”.

He moves on to the Symptoms part and adds his new symptoms: “Loss of smell: Partial”, “Loss of taste: Partial”.

Afterwards, he submits the form and the app displays the message:

“Based on your symptoms you have a chance of 75% of having COVID.



Because there is an extremely high chance of having COVID, please self-isolate immediately, call your GP and take a COVID test as soon as possible. You should also tell anyone you have been in contact with in the past 48 hours that you have COVID symptoms. For more information on how to take a COVID test follow this link: <https://www.gov.uk/get-coronavirus-test>. For other information regarding self-isolating and what to do when you have symptoms follow this link: <https://www.nhs.uk/conditions/coronavirus-covid-19/self-isolation-and-treatment/when-to-self-isolate-and-what-to-do/>. If you develop dyspnea (shortness of breath), you should go to the hospital as soon as possible.

If your symptoms change, please complete this form again. If you take a COVID test and the result is positive, please return to the app and complete the “Positive with COVID Survey”! Thank you!”

Because of the result, Kurtis uses the link provided to find out how to take a COVID test and proceeds to take one.

### Scenario 3: COVID Map

Jenna lives in Birmingham and she just got invited to a birthday party. She doesn't really know what the situation has been like lately regarding COVID, so she decides to use the app to see.

She opens the app, logs in, and selects the COVID Map. At first, she can see a map of the UK (in **COVID Spread** view), which shows her the incidence of cases in each main region of the UK, as well as the **risk of infection** with COVID for each of these regions. Because she would like to see this information, but regarding Birmingham, she enters Birmingham in the “Postcode/Area” field.

The map now shows the same type of information as before, only now it is related to the situation in Birmingham. She can see that here the risk of infection is medium.

She then chooses the **Evolution** view to see if the situation is worse now. The map shows now how the number of COVID cases has **evolved over time** in Birmingham. She also switches to the graph which shows the evolution of COVID cases in that region over the course of the pandemic, as well.

Jenna leaves the app afterwards and decides to not attend the party because the risk of contracting COVID in Birmingham is high and the situation only seems to be getting worse.

### Scenario 4: COVID Map

Julien is curious to see which of the main regions of the UK are most affected by the pandemic now. He already has the app installed on his device, so he decides to use its map.



He opens the app, skips logging in, and selects the COVID Map. At first, he can see a map of the UK (in **COVID Spread** view), which shows him the incidence of cases in each main region of the UK, as well as the **risk of infection** with COVID for each of these regions.

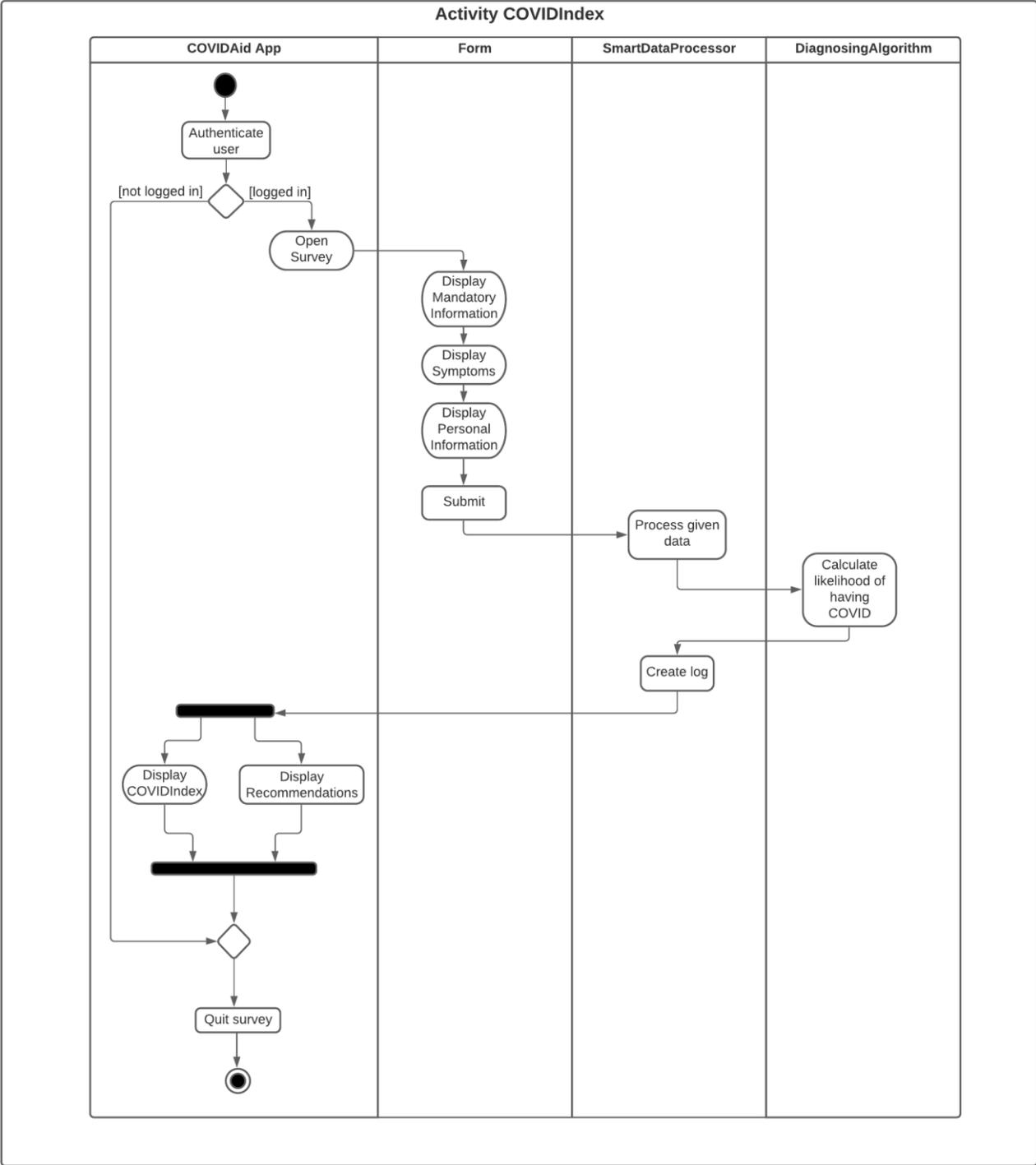
For example, he can see that the incidence of cases in the North West is one of the highest and the risk of infection is high. On the other hand, in the East of England both the risk of infection and the incidence of cases are relatively low.

Curious to see approximately what the situation will look like in the next few days he chooses the **Possible Cases** view. Now the map shows the **number of possible new cases** in each main region (based on the number of COVIDIndex users from each region that received a high probability of having COVID). He can see, now, that the **risk of a new focus of infection** appearing in Yorkshire and the Humber is high.

Julien can see that Northern England is the most affected by the pandemic now and he leaves the app.



# B4. Activity Diagram



## B5. Class Analysis

### Noun Analysis

In order to identify our candidate classes, we performed a noun analysis, using the specification outlined by our use-cases and requirements. The following table of nouns were identified as potential classes.

#### Identified Candidate Classes (Nouns)

Word/Phrase	Accepted	Reason
User	UserAccount	Refers to the logged-in individual that will be using the system.
User that is positive	no	This is just a kind of User, not important enough to be its own class.
App	COVIDAid App	This is the interface that the user will see and interact with.
Positive with COVID Survey	PositiveSurvey	This is the part of the App where a Positive Covid-19 User can fill out his data.
Form	Form	This is the component of the system which allows the user to input data. Therefore, it should be implemented as a class.
Mandatory information	no	This is an attribute that could form part of a class.
age	no	This is an attribute that could form part of a class.
sex	no	This is an attribute that could form part of a class.
Person positive with COVID	no	Same reason as for the "User that is positive".
Symptoms	Symptom	Due to the research into the novel virus, this could be turned into a class for more complex analysis.
Given list of symptoms	no	It is just a group of the previously defined Symptom.
Personal information	no	This is an attribute that could form part of class.
Other medical conditions	no	This is an attribute that could form part of a class.



Location	no	This is an attribute that could form part of a class.
Personal habits	no	This is an attribute that could form part of a class.
Required Fields	no	This is an attribute that could form part of a class.
Backend	no	This refers to the group of the system working together.
Smart database	SmartDataProcessor	This is what will process the data received from the app so there should be a class allocated to it.
Diagnosing algorithm	no	This can form part of the SmartDataProcessor class.
COVID Map	COVIDMap	This is what will be used to parse and illustrate the maps, so it should a separate class. This will be the basis for displaying different views of the map.
COVID Spread view	COVIDSpreadMap	This will show the UK map and detail how affected each area is.
Possible Cases View	PossibleCasesMap	This will show the UK map and where it is likely that another focus of infection will appear.
Evolution View	EvolutionMap	This will show the UK map and how the pandemic evolved over time in each area.
Regions	no	This is an attribute of a class.
Postcode	no	This is an attribute of multiple classes.
Area	no	This is an attribute of multiple classes.
NHS	Connector	This is a class that allows external systems (NHS, Government) to provide data for our databases.
Database entry	COVIDLog	This is the way the system stores submitted Surveys and COVIDIndex and hence
Helpful Guides	HelpfulGuide	This is the section of the system where a User can seek more guidance and hence should have a separate class.



## Verb Analysis

In order to identify our candidate operations, we performed a verb analysis, using the specification outlined by our use-cases and requirements. The following table of verbs were suggested as potential operations.

### Identified Candidate Operations (Verbs)

Word/Phrase	Accepted	Reason
Use the survey	openPositiveSurvey	This is a particular action of the COVIDAid App and hence becomes a method.
Get guidance	requestGuidance	This is a particular action of the Survey and COVIDIndex classes and hence becomes a method.
Checks that the user is logged in	authenticateUser	This is a particular action of the COVIDAid App and hence becomes a method.
Displays	displayForm	This is a particular action of the Survey and COVIDIndex classes and hence becomes a method.
Displays the fields of the form	displayFields	This is a particular action of the Survey and COVIDIndex and hence becomes a method.
Selects their age, their sex ...	no	Duplicate
Display the first part of the form	fillMandatoryInfo	Duplicate
Displays the second part of the form	fillSymptoms	Duplicate
Displays the third part of the form	fillPersonallInfo	Duplicate
Checks that the user answered all the required fields	verifyFields	This is a particular action of the Survey and COVIDIndex classes and hence becomes a method.



Collects the answers	saveFields	This is a particular action of the Survey and COVIDIndex classes and hence becomes a method.
Saves them in the backend	no	Too generic
Updates the diagnosing algorithm	updateAlgorithm	This is a particular action of the SmartDataProcessor and hence becomes a method.
Displays a thank you message	displayMessage	This is a particular action of the COVIDAid App and hence becomes a method.
User is redirected to the Authentication Use Case	redirectUser	This is a particular action of the COVIDAid App and hence becomes a method.
No response from the smart database	returnError	This is a particular action of the COVIDAid App and hence becomes a method.
User received the right guidance	displayAdvice	This is also a particular action of the COVIDIndex and hence becomes a method.
The app should autofill the form	suggestField	This is a particular action of the Survey and COVIDIndex classes and hence becomes a method.
The app should autosave the information	autoSave	This is a particular action of the Survey and COVIDIndex classes and hence becomes a method.
Submit the form	submit	This is an action of the Form class.
user selects the Covid Map	no	Duplicate
Displays a Map	displayMap	This is a particular action of the COVID Map and hence becomes a method.
Map updates to the chosen view	changeView	This is a particular action of the COVID Map and hence becomes a method.



Map shows detailed information	processMap	This is a particular action of the COVIDMap and hence becomes a method.
Give the user further guidance	getAdvice	This is a particular action of the Positive Survey and the COVIDIndex features.
Tell the user	displayAdvice	This is a particular action of the Positive Survey and the COVIDIndex features.
Send the user a notification	notifyUser	This is a particular action of the COVIDAid App and hence becomes a method.
App must display relevant links ... from NHS and the government	displayGuides	This is a particular action of the Helpful Guides class and hence becomes a method.`

## Responsibility-Driven Analysis

COVIDMap	
Responsibilities	Collaborators
Hold information regarded to the postcode of the user. Displays an accurate map based on the coronavirus infection on a specific area. Allows the users to change the view mode in the map. Processing the map.	COVIDAid App SmartDataProcessor

Form	
Responsibilities	Collaborators
Creates a Survey Form if requested. Creates a COVIDIndex Form if requested.	Symptom FormController



### Helpful Guide

Helpful Guide	
Responsibilities	Collaborators
Provides medical guidance and support for the users interested. Provides information regarding the restrictions. Provides information about financial support. Can request data and validate it.	COVIDAid App SmartDataProcessor

### Survey

Survey	
Responsibilities	Collaborators
It analyses the symptoms of the user. It saves the symptoms of the user. It displays a "Thank you" message at the end of the survey.	COVIDAid App FormController

### COVIDIndex

COVIDIndex	
Responsibilities	Collaborators
It analyses the conditions of the user. Based on the results it also provides helpful advices.	COVIDAid App FormController

### COVIDAid App

COVIDAid App	
Responsibilities	Collaborators
Maintain overall control of the app. Allows the user to use the survey mode. Allows the user to use the COVIDIndex mode. Allows the user to use the helpful guide. Allows the user to see the infection map. It provides User Authentication It notifies the user of possible new regulations.	Survey COVIDMap COVIDIndex Connector UserManager SmartDataProcessor Helpful Guides



Symptom	
Responsibilities	Collaborators
Holds information about a specific symptom.	COVIDLog SmartDataProcessor COVIDMap

UserManager	
Responsibilities	Collaborators
Handles the creation, update and deletion of the user account. Manages the user's requests.	SmartDataProcessor UserAccount COVIDAid App

COVIDLog	
Responsibilities	Collaborators
Based on the type of the subclasses COVIDIndexLog and SurveyLog, it handles specific log messages.	SmartDataProcessor UserAccount Symptom

SmartDataProcessor	
Responsibilities	Collaborators
The core component of the app responsible for the management of the data provided. Updates the algorithm based on the data provided. Creates log messages when a change is made. Updates the data for distribution.	UserManager COVIDLog COVIDMap Symptom FormController Helpful Guides COVIDAid App





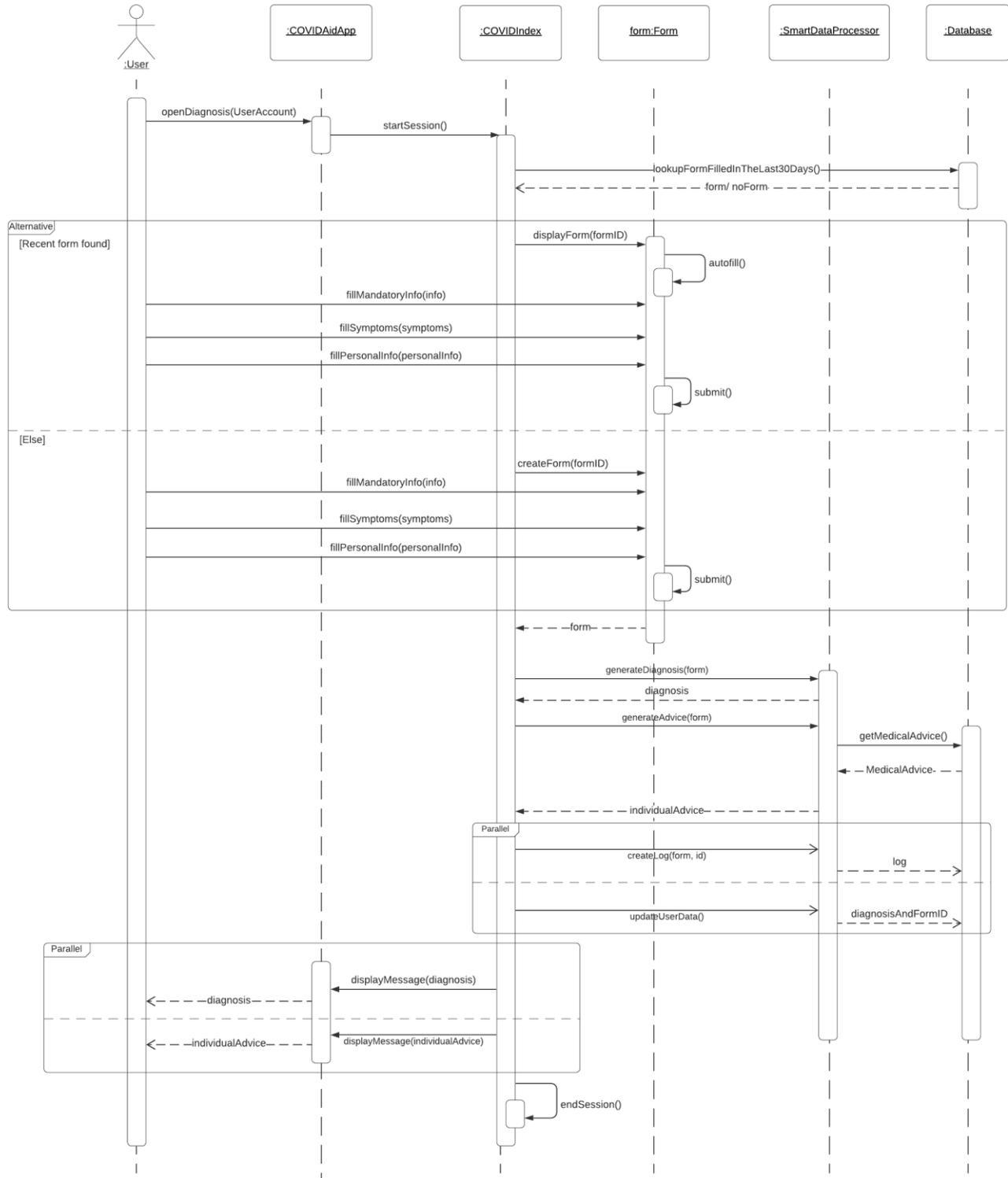




## B7. Two Sequence Diagrams

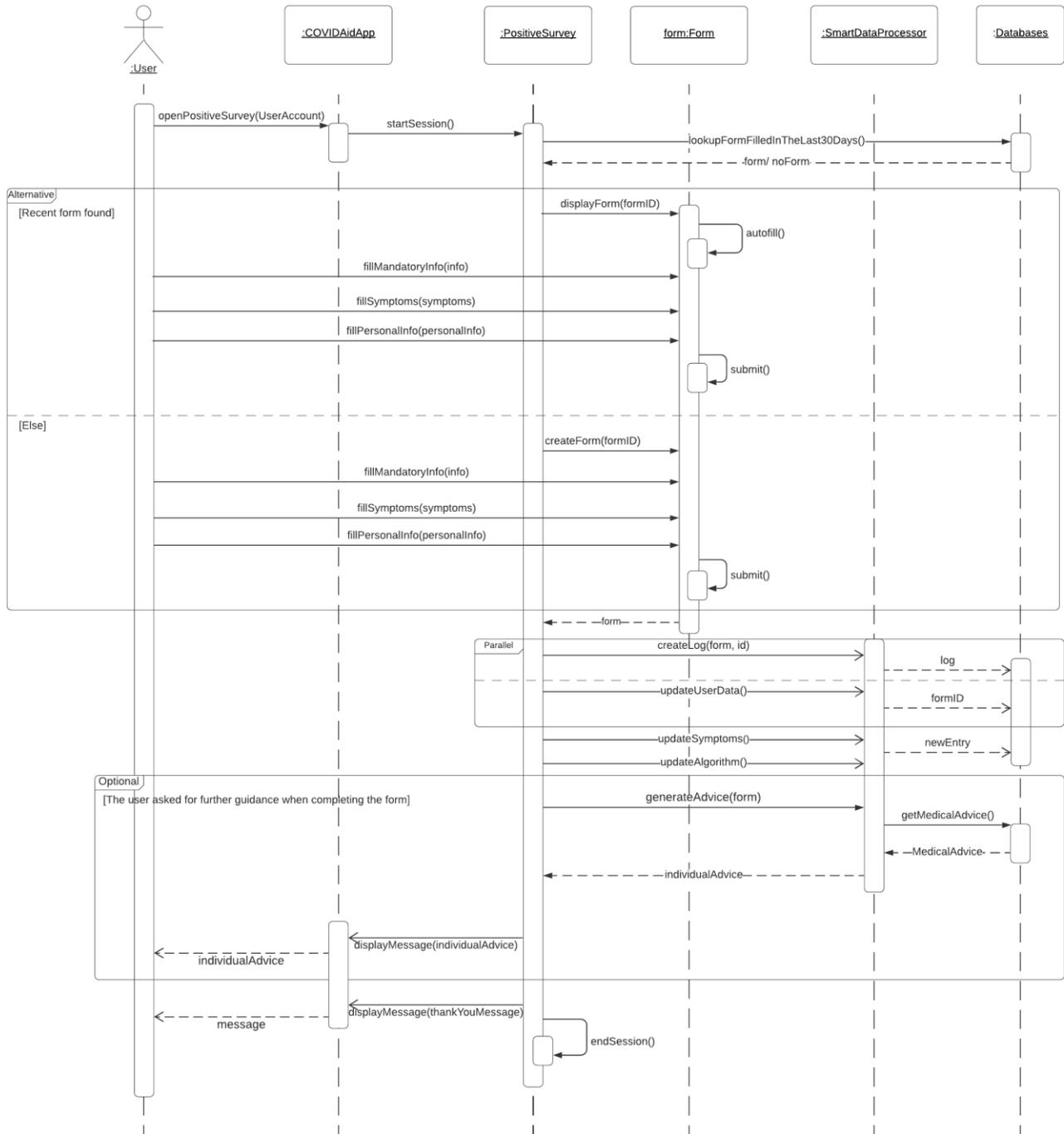
### Sequence Diagram for the COVIDIndex feature

Assume the user is already logged in.



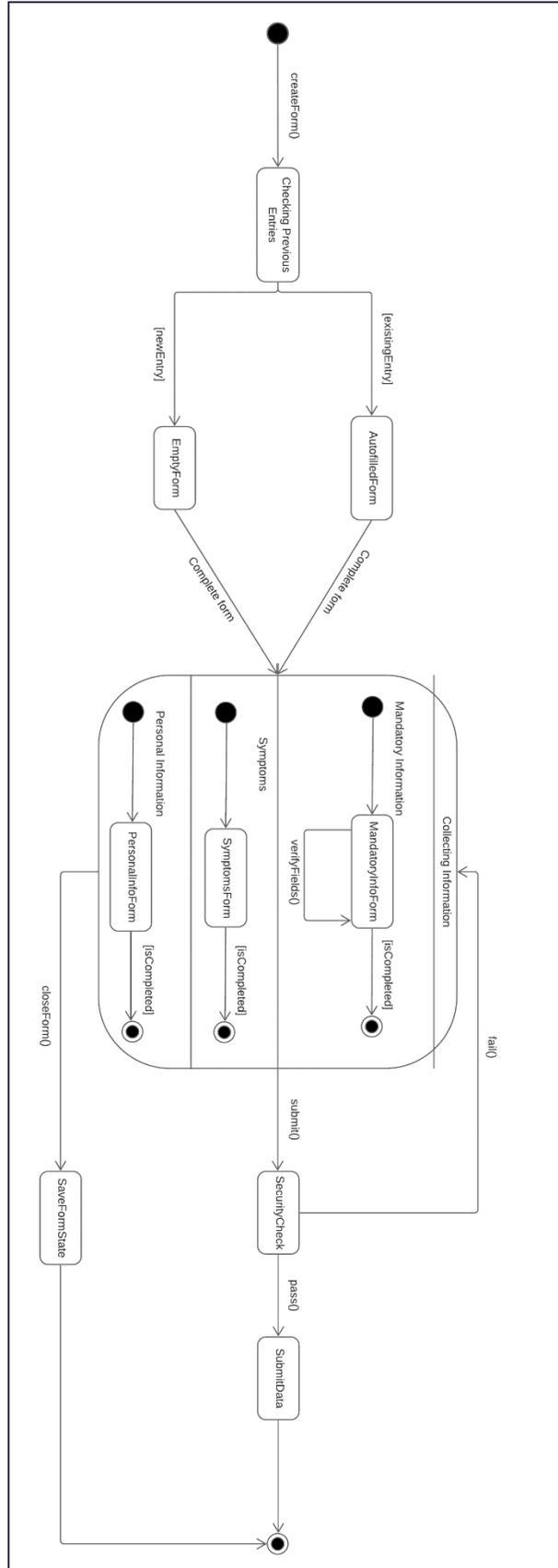
# Sequence Diagram for the Positive with COVID Survey

Assume the user is already logged in.

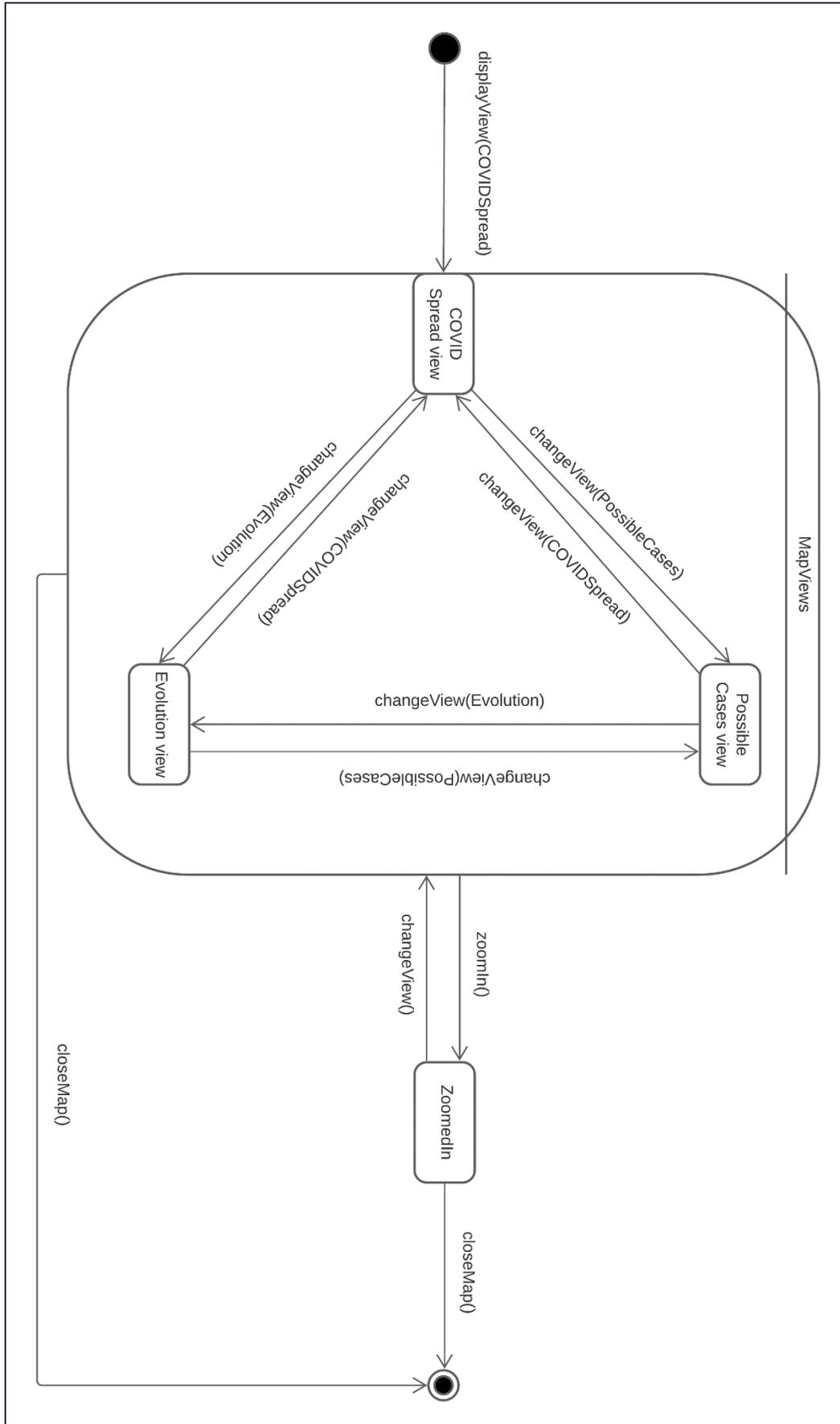


## B8. Two State Diagrams

### State Diagram for the Form Class



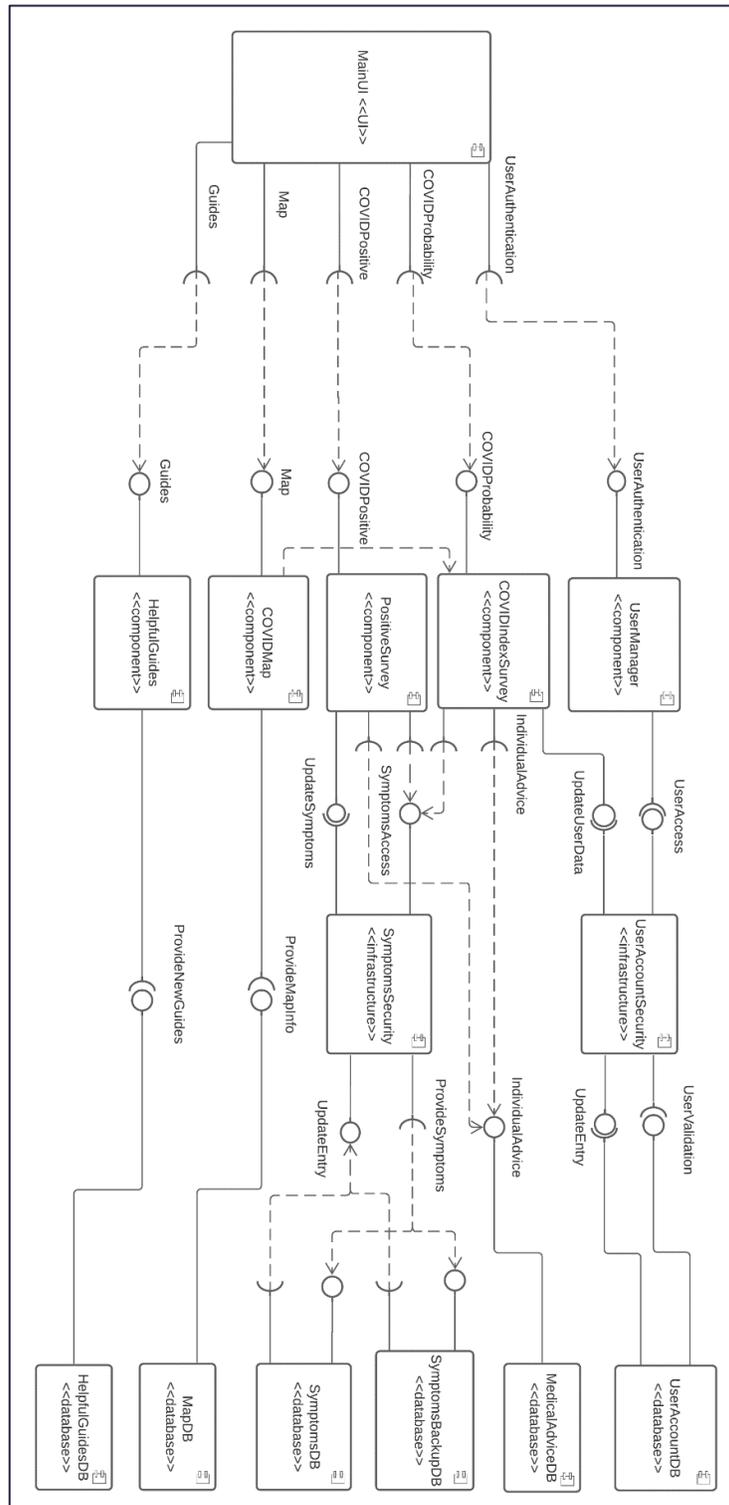
## State Diagram for the COVIDMap Class



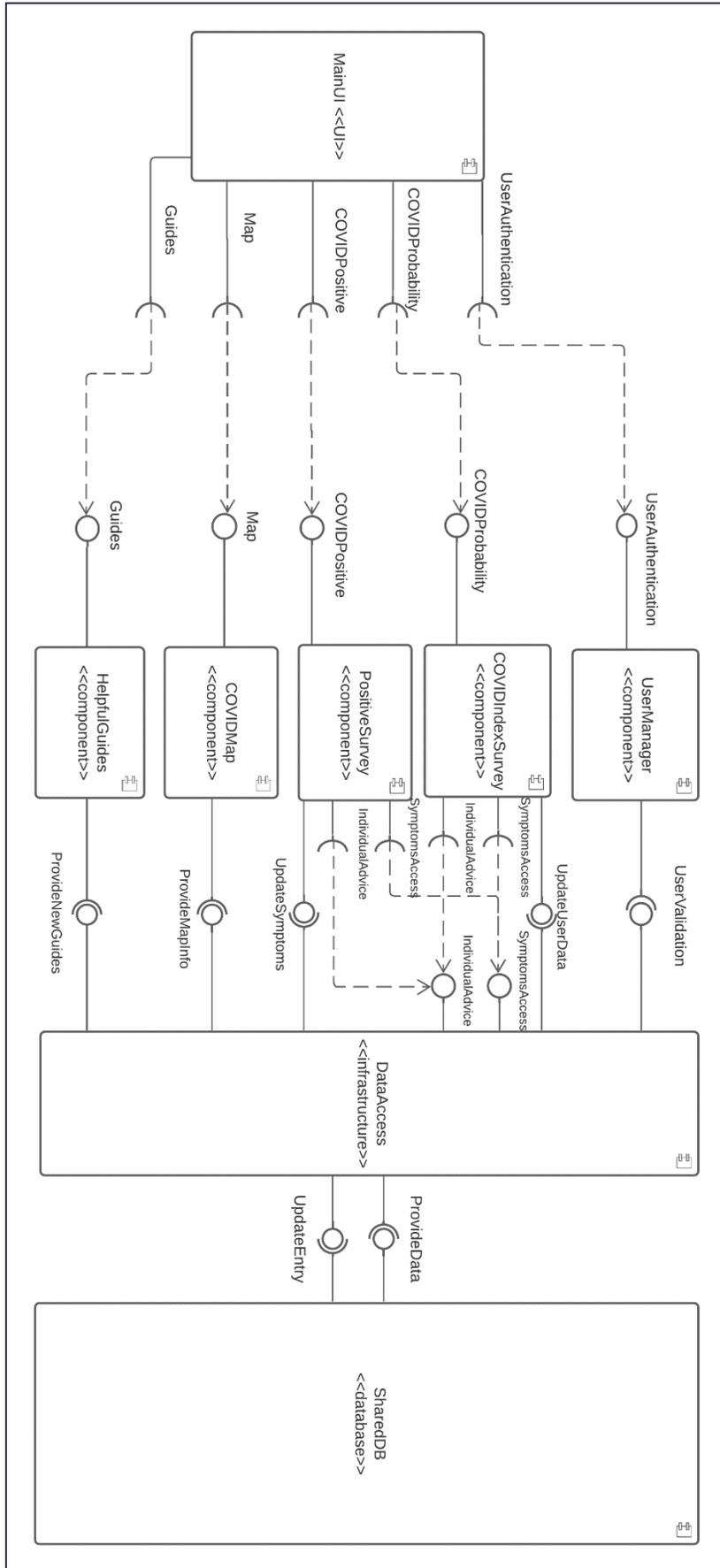
# C. SOFTWARE ARCHITECTURE STYLE, MODELLING AND EVALUATION

## C1. Two Possible Architecture Styles

### First Candidate – Component Diagram

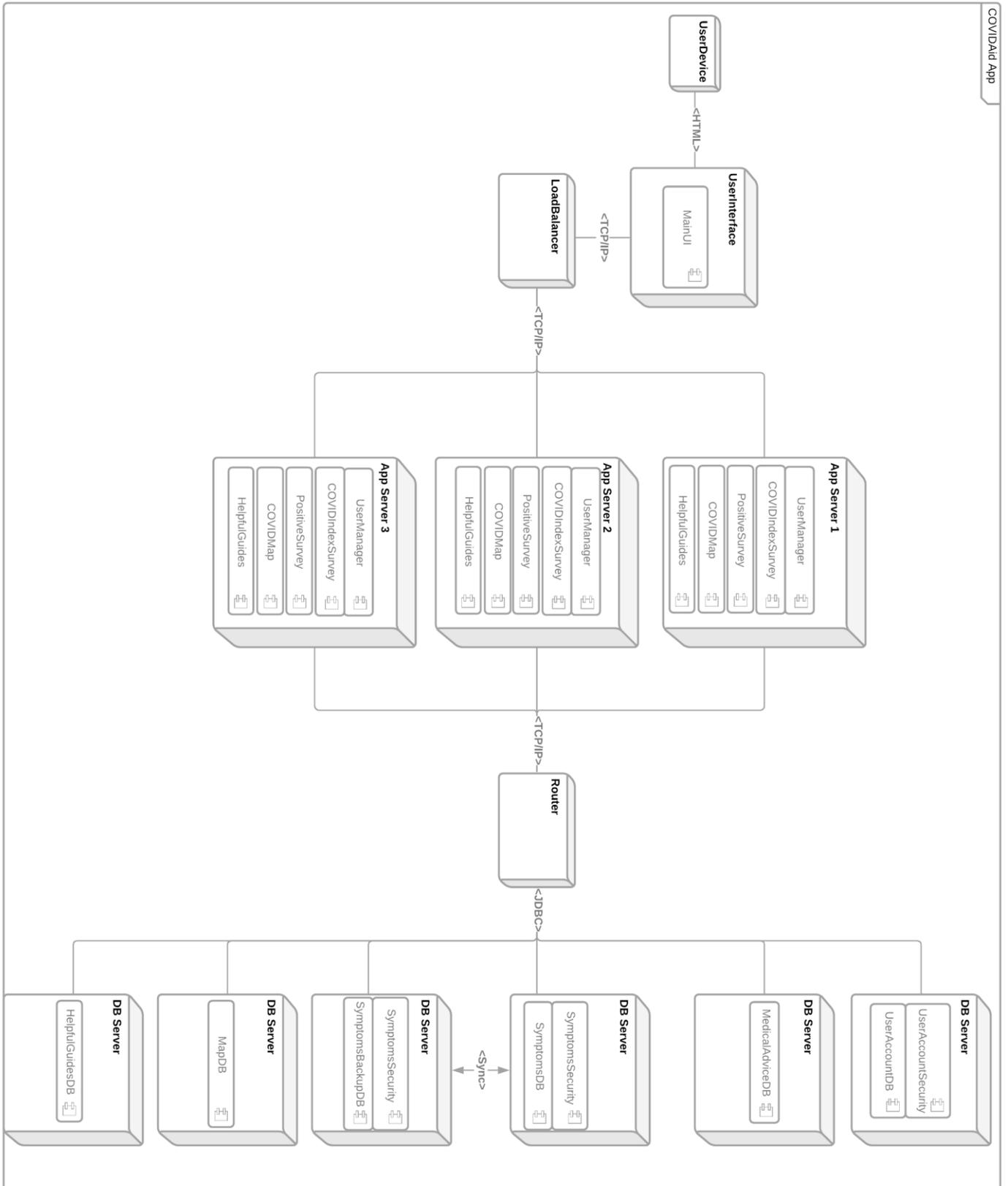


## Second Candidate – Component Diagram:

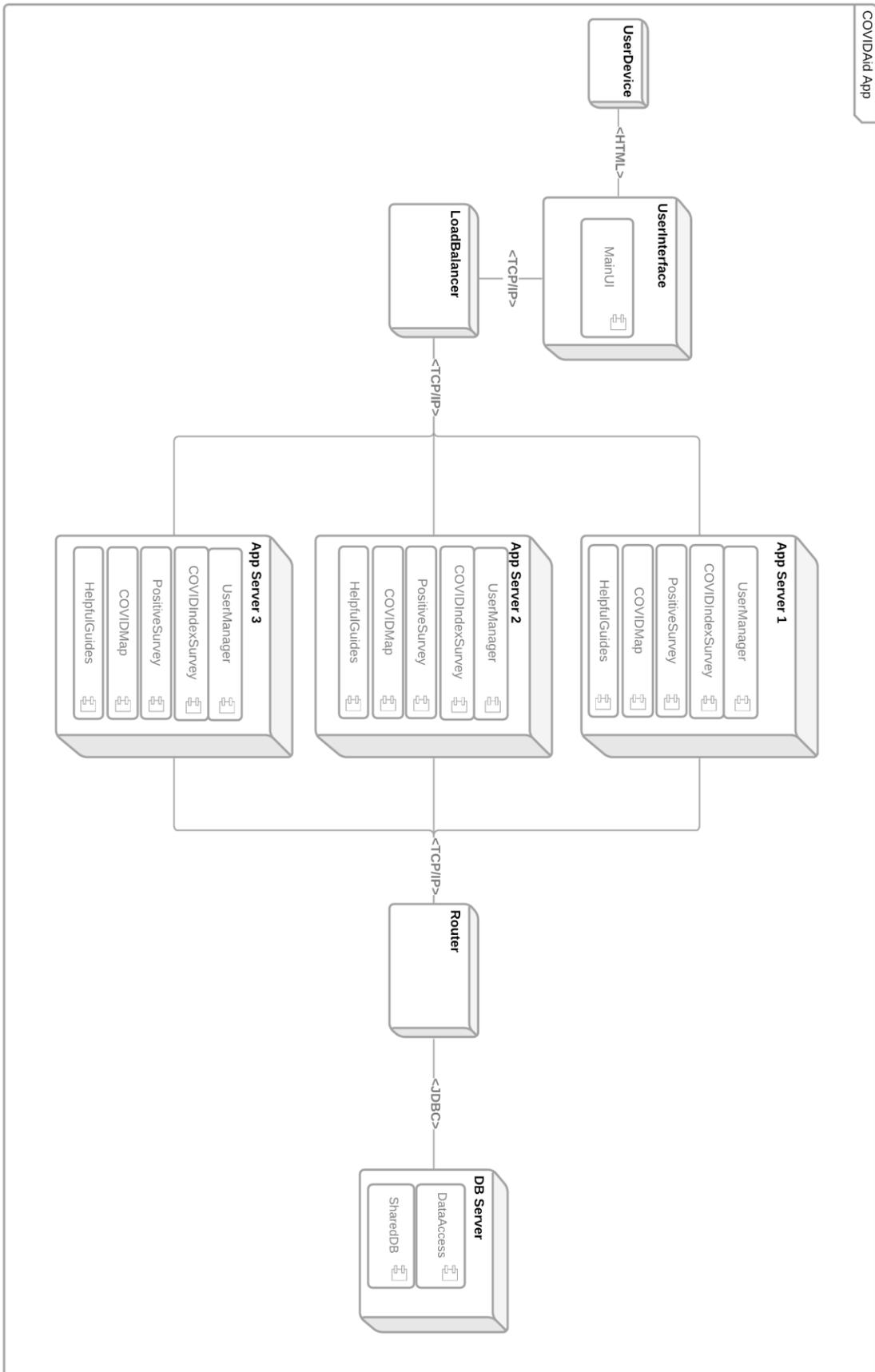


## C2. Two Deployment Diagrams

### First Candidate – Deployment Diagram:



## Second Candidate – Deployment Diagram:



## C3. Comparison of Architectures

### Trade-off Analysis

In the case of the COVIDAid application the most important quality attributes are performance, because the system must be able to return results as quickly as possible and security, because, even though providing personal details is optional, some users may still opt to provide some personal information and efforts must be made to guarantee the protection of this data. As such, when designing the candidate architectures for this system, performance and security have been the focus, but attributes such as maintainability, availability and error tolerance were important factors as well.

### First Candidate:

The first architecture design is inspired by the n-tier style of architecture. The first layer is focused on the client, so it contains the main user interface component which provides the client with access to the actual features of the system. The second layer represents the main functionalities of the app: user management, COVIDIndex, Positive with COVID Survey, COVIDMap and the Helpful Guides. The final layer contains the databases for the system. An argument can be made that there is one more layer between the features and the databases which is related to security.

The main advantage of this design is security. Firstly, because the user accounts database and the symptoms database are the most critical parts of the system, both from a functionality point of view as well as the privacy one, these databases have their own security components to provide encryption of data and protect against malicious insertions of data. Adding such security to each of the databases was considered, but because this would decrease performance the decision was made to ensure protection only for the most important databases. Apart from the security put in place, the symptoms database also has a synchronized backup such that the backup can take over primary function or be used to restore the main one if needed. Secondly, roughly each feature has its own database, so if one component is compromised then the others can still be secure. However, the measures taken to increase security lead to an increase in communication between components and therefore, a decrease in performance.

When it comes to deployment, there are three servers which provide the main features of the system. The system makes use of a load balancer to direct the client to the server with the least traffic to avoid any performance issues. The servers are connected to the databases through a



router to direct data to the corresponding database. Although the use of a load balancer aims to increase performance, using multiple servers as well as multiple databases will inevitably increase the cost of the project. Furthermore, having multiple servers increases the number of components which need maintaining.

### **Second Candidate:**

The second architecture design is inspired by the blackboard architecture style. Because a lot of data transfers take place inside the system, the use of one main, shared database should improve performance. As such, the client and business logic levels remain unchanged, but the way they access data is different. All the connections from the individual features to the shared database are managed by an infrastructure called DataAccess which replaces the previous security layer between the functionality layer and database layer.

This design's advantage is performance. Because there is only one database which contains all of the required data and only one point of access there is less communication and, therefore, this increases performance. Another advantage is that the other components do not need to concern themselves with security and data management - the database takes care of these aspects itself. However, one disadvantage is that there is no scope for specific management policies and therefore the partitioning of data becomes very difficult. Another disadvantage could be represented by security: given how all of the data is stored in a single shared database, if the database gets compromised then potentially all of the data could get leaked or become unreliable, since data integrity cannot be guaranteed anymore.

Deployment is similar to the previous design, with a load balancer and three servers that provide functionalities. The main difference is having only one shared database instead of multiple ones.

### **Conclusion:**

Considering the specification of the system, we consider that the first candidate structure suits the application's requirements better: it is more secure than the second one due to how data is stored and the actual security protocols that are put in place for the critical parts of the system, it is more maintainable, due to the use of multiple, smaller components rather than large ones and this also facilitates the addition of any future functionalities or any updates for the current ones. The difference in cost and performance should not be significant enough to warrant adopting the second design.



# D. SOFTWARE TESTING

## Introduction

The test plan is designed to prescribe the scope, approach, and list all the testing activities of the project COVIDAid.

The plan identifies the items and features to be tested, the types of testing to be performed, the schedule required to complete testing, and the risks associated with the plan.

Please note that the COVIDAid app will be referred to as the Application Under Test or AUT.

## Testing Objectives

Please refer to the requirements section while reading these objectives.

- Ensure the AUT conforms to functional and non-functional requirements.
- Any user of the AUT must be able to create an account with the AUT through a registration - REQ 1.
- Any registered member should be able to login with previously created accounts - REQ 2.
- Any invalid login attempts must be warned with an error message - REQ 2.
- The AUT must restrict access to users which are not registered for features like the “Positive with COVID survey” - REQ 3.
- The AUT must be able to compute and give guidance tailored for each specific user - REQ 3.3, 4.3.
- The AUT must display the “Positive with COVID survey” in the format specified in the requirements section - REQ 5.
- A user should get suggestions from the AUT while filling a form in the app which is not the first time - REQ 5.1.
- The AUT must validate and verify all data entries in the Forms - REQ 5.7.
- The AUT must be able to validate and verify the entries in the Forms within 1 second - REQ 9.1.
- The AUT must filter out any contradictory data - REQ 14.5.
- Any user of the AUT must be able view and render maps in the COVID Map feature - REQ 6.
- The AUT must be able to display the rendered maps within 5 seconds - REQ 9.3.
- The AUT must display the Helpful Guides within 5 seconds - REQ 9.4.
- Any user of the AUT must be able to access helpful information and guidance from the Helpful Guide feature - REQ 7.
- The user must be able to select his/her preferred language - REQ 8.1.



## Test Items

The systems to be tested include the frontend facing website along with components of the application in the backend. These systems should be tested on mobile devices running on Android 4.0.3 and newer, and iOS 9 and newer.

### Frontend facing items

Item	Reference
Positive with COVID Survey	REQ 5 of Requirements in section A Class Analysis in section B Components Diagram in section C
COVIDIndex	REQ 4 of Requirements in section A Class Analysis in section B Components Diagram in section C
COVID Map	REQ 6 of Requirements in section A Class Analysis in section B Components Diagram in section C
Helpful Guides	REQ 7 of Requirements in section A Class Analysis in section B Components Diagram in section C
Login feature	REQ 1 and 14 in section A

### Backend facing items

Components defined in the class analysis (check section B) and in the components diagram include the SmartDataProcessor and various databases as defined in the deployment diagram and APIs (represented as the Connector in the class analysis of section B). Most of these items will only be able to be tested with the use of frontend facing items and a gradual collection of data(which can also be simulated using external software or with the help of volunteers). Items like the SmartDataProcessor will not be able to be tested using a standard test as most of its features are based on self-learning and data trending (this excludes any database logic).

More test items should be defined with each iteration.

## Features To Be Tested

Features to be tested include the following:

- As a user, logging into the app



- As a user, registering for the app
- As a user, deleting the account from the app
- As a user, navigating through the app
- As a user, accessing the Positive with Covid Survey
- As a user, filling and submitting the form from the Positive with Covid Survey
- As a user, accessing the COVIDIndex section
- As a user, filling and submitting the form from the COVIDIndex Section
- As a user, accessing the COVID Map
- As a user, selecting the various types of views from the COVID Map
- As a user, accessing the Helpful Guides
- Network, with all its layers

## Features Not To Be Tested

Features not to be tested include the following:

- Hardware Interfaces
- Database logic

## Approach

The quality team will create test sets for each tester using white-box and black-box testing methods. The tester will execute the tests in a Pass/Fail test log and mark each case as Pass/Fail. The tester should leave notes on actual results and any other relevant details when possible.

In the project COVIDAid, 5 types of testing should be conducted:

- **Unit** testing, where each component defined in the components diagram should be tested. This must be done by the developers with the purpose of validating each unit of the software performs as designed.
- **Integration** testing, where individual software modules are combined and tested as a group, please refer to the components diagrams.
- **System** testing, conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements.
- **API** testing, test all the APIs created for the software under test that will transfer data between the tested system and external system like the NHS or Government.
- **Acceptance** testing, this is the last test that should be performed in order to evaluate the system's compliance with the requirements defined in the requirements section and assess whether it is acceptable for delivery.

Participants involved with testing:

- Project Manager
- Tester
- Developer



## Test Deliverables

Test deliverables are provided as below

Before testing:

- Test plans
- Requirements and specification defined in section A
- Test cases (in the test plan)
- Design Specifications as defined in Section B and C of the document.

During testing:

- Test tool simulators
- Test data
- Test Logs(Pass/Fail criteria) and execution logs

After the testing cycle is over:

- Test Results/Reports
- Defect/Bug Triage Report
- Release notes

## Testing tools

These are based on the component and deployment diagrams:

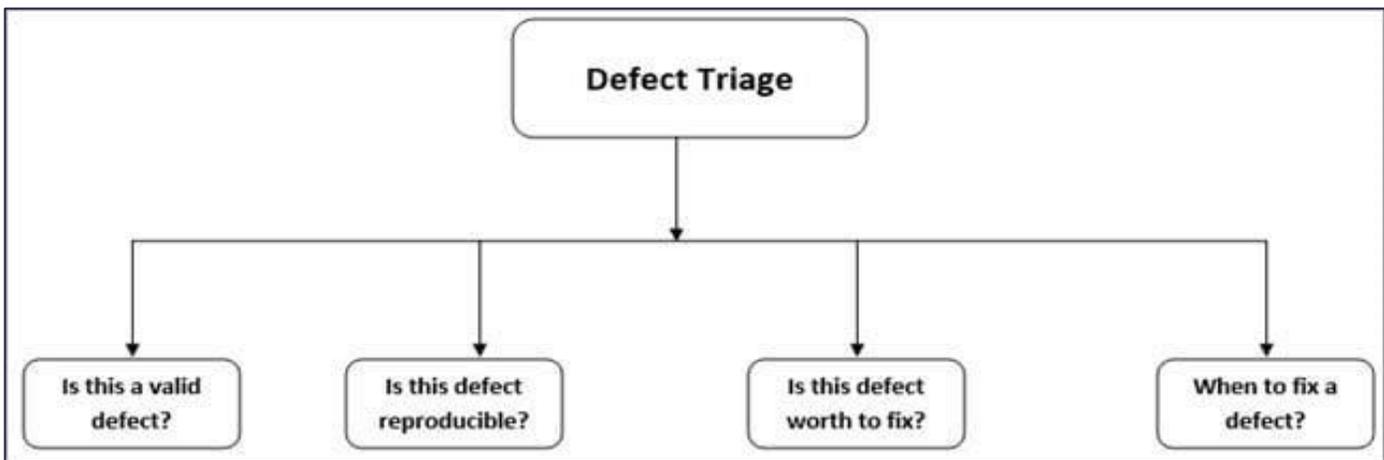
Tool	Description	Reference
App Servers	Used to handle requests for features of the AUT which interact with the user. At least 3 required	Part A of the document Part B of the document Components and Deployment Diagram of Part C
Database Servers	Required to handle the various database logic of the AUT's components. At least 6 required	Part B of the document Components and Deployment Diagram of Part C
Test tool	Develop a Test tool which can auto generate the test result to the predefined form and automated test execution	Refer to Defect/Bug Triage of this section
Network	Setup a LAN Gigabit and 1 internet line with the speed at least 5 Mb/s	Refer to the requirements in section A
Mobile Device	Needed for testing the load stress handling and performance of the AUT. At least 4 mobile	Requirements in section A



	devices running Android 4.0.3 and newer, and iOS 9 and newer.	Testing Objectives
External load and stress tools	Third party software used to test the performance of the UAT and various components of the architecture under soft, mild and extreme conditions.	As a requirement in the requirements in section A

## Defect/Bug Triage

When tests are marked as Fail, bug reports will automatically be created and assigned to a developer. The developer makes the change and returns it back to the responsible tester. The test manager reviews the test report in the test log for final approval. A defect triage report should be also included which reflects the points in the diagram below:



Each bug logged in the bug reports should be categorised based on if it's a valid defect, is it reproducible, is it worth to fix or when it should be fixed.

The bug triage report should be of the following format:

Defect ID	Description	Priority	Severity	Detected Date	Detected By	Status
...						



## Pass Fail Criteria

User logging into the application:

Test Case ID	Test description	Test steps	Test Data	Expected result	Actual result	Pass / Fail	Test comments
TD-1	Verify the login with valid userName and password	Go to COVIDAid Enter username Enter password Press submit	userName: hope password:lwantcure23 @	User should be able to login			
TD-2	Verify the login with valid userName and invalid password	Go to COVIDAid Enter username Enter password Press submit	userName: hope password:iwantcure	User should not be able to login into the application			
TD-3	Verify the login with valid userName and	Go to COVIDAid	userName: hope password:iwantcure	User should not be able to login into			



	invalid password	Enter username Enter password Press submit		the application			
TD-4	Verify the login with valid username and invalid password	Go to COVIDAid Enter username Enter password Press submit	userName: hope password:lwantcure	User should not be able to login into the application			
TD-5	Verify the login with valid username and invalid password	Go to COVIDAid Enter username Enter password Press submit	userName: hope password:lwantcure2	User should not be able to login into the application			
TD-6	Verify the login with valid username	Go to COVIDAid	userName: hope password:lwantcure@	User should not be able to			



	e and invalid password	Enter username Enter password Press submit		login into the application			
TD-7	Attempt to login 5 times with valid username and invalid password	Go to COVIDAid Enter username Enter password Press submit	userName: hope password:lwantcure@	The application should block access to the user account			

User using the Positive with Covid Survey feature:

Test Case ID	Test description	Test steps	Test Data	Expected result	Actual result	Pass / Fail	Test comments
TD-8	Access the Positive...Survey to fill the form without being logged in	Go to COVIDAid Open Positive...Survey	Screen Press	User should not be able to access this feature			



TD-9	Access the Positive...Survey to fill the form by being logged in	Go to COVIDAid Open Positive...Survey	Screen Press	User should be able to access the form found in the Positive...Survey			
TD-10	Validate the form with complete fields in the Mandatory section, complete in the Symptoms section and complete in the Personal Information section	Go to COVIDAid Open Positive...Survey Fill the form Press submit	Complete data for Mandatory section Complete data for Symptoms section Complete data for Personal Information section	The user should be able to submit the form and receive appropriate medical advice if they requested it			
TD-11	Validate the form with incomplete fields in the Mandatory section, complete in the Symptoms section and complete in the Personal	Go to COVIDAid Open Positive...Survey Fill the form Press submit	Incomplete data for Mandatory section Complete data for Symptoms section Complete data for Personal Information section	User should not be able to submit the form			



	Information section						
TD-12	Validate the form with complete fields in the Mandatory section, incomplete in the Symptoms section and complete in the Personal Information section	Go to COVIDAid Open Positive...Survey Fill the form Press submit	Complete data for Mandatory section  Incomplete data for Symptoms section  Complete data for Personal Information section	User should not be able to submit			
TD-13	Validate the form with complete fields in the Mandatory section, complete in the Symptoms section and incomplete in the Personal Information section	Go to COVIDAid Open Positive...Survey Fill the form Press submit	Complete data for Mandatory section  Incomplete data for Symptoms section  Incomplete data for Personal Information section	User should be able to submit			



TD-14	Check that the AUT suggests fields if the user already submitted a form before and the fields were also previously filled in	Go to COVIDAid Open Positive...Survey Fill the form	Data for fields which were previously filled in past forms	User should get a suggestion about past data entries for each specific field			
TD-15	The user selects the range of ages from the Mandatory section, selects his/her sex and selects if he/she got in contact with somebody COVID positive. While having completed the Symptoms sections.	Go to COVIDAid Open Positive...Survey Fill the Mandatory section of the form Press submit (full form)	age_range: Selection from drop-down menu with a list of the range of ages  sex: Selection from drop-down menu  check_if_in_contact: Selection from options yes or no	User should be able to submit the full form			



TD-16	The user does not select the range of age from the Mandatory section, selects his/her sex and selects if he/she got in contact with somebody COVID positive. While having completed the Symptoms sections.	Go to COVIDAid Open Positive...Survey Fill the Mandatory section of the form Press submit (full form)	age_range: No selection from drop-down menu with a list of the range of ages  sex: Selection from drop-down menu  check_if_in_contact: Selection from options yes or no	User should not be able to submit the full form			
TD-17	The user selects the range of age from the Mandatory section, does not select if his/her sex and selects if he/she got in contact	Go to COVIDAid Open Positive...Survey Fill the Mandatory section of the form Press submit (full form)	age_range: Selection from drop-down menu with a list of the range of ages  sex: No selection from drop-down menu	User should not be able to submit the full form			



	with somebody COVID positive. While having completed the Symptoms sections.		check_if_in_contact: Selection from options yes or no				
TD-18	The user selects the range of age from the Mandatory section, selects if his/her sex and does not select if he/she got in contact with somebody COVID positive. While having completed the Symptoms sections.	Go to COVIDAid Open Positive...Survey Fill the Mandatory section of the form Press submit (full form)	age_range: Selection from drop-down menu with a list of the range of ages  sex: Selection from drop-down menu  check_if_in_contact: No selection from options yes or no	User should not be able to submit the full form			



TD-19	The user selects symptoms from a list of symptoms shown by the application, describes each symptom selected from the list in the description section and enters any new symptoms which are not listed by the application. While having completed the Mandatory section.	Go to COVIDAid Open Positive...Survey Fill the Symptoms section of the form Press submit (full form)	symptoms: Selection from a range of symptoms listed by the application.  symptoms_descriptions: Description given to each selected symptom  other_symptoms: New symptoms entered by the user	User should be able to submit the full form			
TD-20	The user does not select symptoms from a list of	Go to COVIDAid Open Positive...Survey	symptoms: No selection from a range of symptoms listed by the application.	User should not be able to submit the full form			



	<p>symptoms shown by the application, describes each symptom selected from the list in the description section and enters any new symptoms which are not listed by the application. While having completed the Mandatory section.</p>	<p>Fill the Symptoms section of the form Press submit (full form)</p>	<p>symptoms_descriptions: Description given to each selected symptom</p> <p>other_symptoms: New symptoms entered by the user</p>				
TD-21	<p>The user selects symptoms from a list of symptoms shown by the application, describes</p>	<p>Go to COVIDAid Open Positive...Survey Fill the Symptoms section of the form</p>	<p>symptoms: Selection from a range of symptoms listed by the application.</p> <p>symptoms_descriptions: Description given</p>	<p>User should be able to submit the full form</p>			



each symptom selected from the list in the description section and does not enter any new symptoms which are not listed by the application. While having completed the Mandatory section.	Press submit (full form)	to each selected symptom  other_symptoms: No new symptoms entered by the user				
---	--------------------------	--	--	--	--	--

User using the COVIDIndex feature:

Test Case ID	Test description	Test steps	Test Data	Expected result	Actual result	Pas s / Fail	Test comment s
TD-22	Access the COVIDIndex feature to fill the form without	Go to COVIDAid Open COVIDIndex	Screen Press	User should not be able to access this feature			



	being logged in						
TD-23	Access the COVIDIndex to fill the form by being logged in	Go to COVIDAid Open COVIDIndex	Screen Press	User should be able to access the form found in the COVIDIndex			
TD-24	Validate the form submission with selected and valid symptoms and selected and valid medical conditions	Go to COVIDAid Open COVIDIndex Fill the form Press submit	<p>symptoms: Selection from a range of symptoms listed by the application or further description given for unknown symptoms by the application.</p> <p>medical_conditions: Selection from a range of medical conditions listed by the application or further description given for unknown medical conditions by the application.</p>	The user should be able to submit the form and the application should return a result showing the probability of the user having COVID			



TD-25	Validate the form submission with no selection of symptoms and selected and valid medical conditions	Go to COVIDAid Open COVIDIndex Fill the form Press submit	symptoms: No selection from a range of symptoms listed by the application.  medical_conditions: Selection from a range of medical conditions listed by the application or further description given for unknown medical conditions by the application.	The user should not be able to submit the form			
TD-26	Validate the form submission with selected and valid symptoms and no selection of valid medical conditions	Go to COVIDAid Open COVIDIndex Fill the form Press submit	symptoms: Selection from a range of symptoms listed by the application or further description given for unknown symptoms by the application.  medical_conditions: No selection from a range of medical	The user should not be able to submit the form			



			conditions listed by the application.				
TD-27	Submit the form of the COVIDInd ex after filling the form with selected and valid symptoms and no selection of valid medical conditions	Go to COVIDAid Open COVIDInd ex Fill the form Press submit	symptoms: Selection from a range of symptoms listed by the application or further description given for unknown symptoms by the application.  medical_conditions: No selection from a range of medical conditions listed by the application.	The application should return and display the computed result to the user within 5 seconds			

#### User using the Covid Map:

Test Case ID	Test description	Test steps	Test Data	Expected result	Actual result	Pass / Fail	Test comments
TD-28	Access the Covid Map	Go to COVIDAid Open Covid Map	Screen Press	User should be able to access this feature			
TD-29	Enter a valid postcode	Go to COVIDAid	postcode: DE23	The application should			



		Open Covid Map Enter postcode Press submit		show a map of the inputted postcode's area			
TD- 30	Enter an invalid postcode	Go to COVIDAid Open Covid Map Enter postcode Press submit	postcode: D&23	The application should return an error message to the user warning of the incorrect postcode			
TD- 31	Select the COVIDSpread Map view	Go to COVIDAid Open Covid Map Press COVIDSpread Map view	Screen Press	The application should return a map showing the mentioned view with infections rates and number of cases			
TD- 32	Select the Possible Cases Map view	Go to COVIDAid Open Covid Map	Screen Press	The application should return a map			



		Press Possible Cases Map view		showing the mentioned view with possible cases and probability of risk			
TD-33	Select the Evolution Map view	Go to COVIDAid Open Covid Map Press Evolution Map view	Screen Press	The application should return a map showing the mentioned view with evolution of daily cases			
TD-34	Select the desired view from the COVID Map	Go to COVIDAid Open Covid Map Press any Map view	Screen Press	The application should return a map showing the mentioned view within 5 seconds			

User using the Helpful Guides feature:

It is required that there is a constant connection between the external sites such as the NHS and the Government site for it to show useful guides.



The Helpful Guides should be loaded in less than 5 seconds alongside the various requested data to be shown and tailored on the user's needs.

## Exit Criteria

If the team members report that there are **20%** of test cases **failed**, **suspend** testing until the development team **fixes** all the failed cases.

**Pass** rate is **92%** and achieving the pass rate is **mandatory**.

**Run** rate is mandatory to be **98%** unless a clear reason is given.

## Assumption

N/A



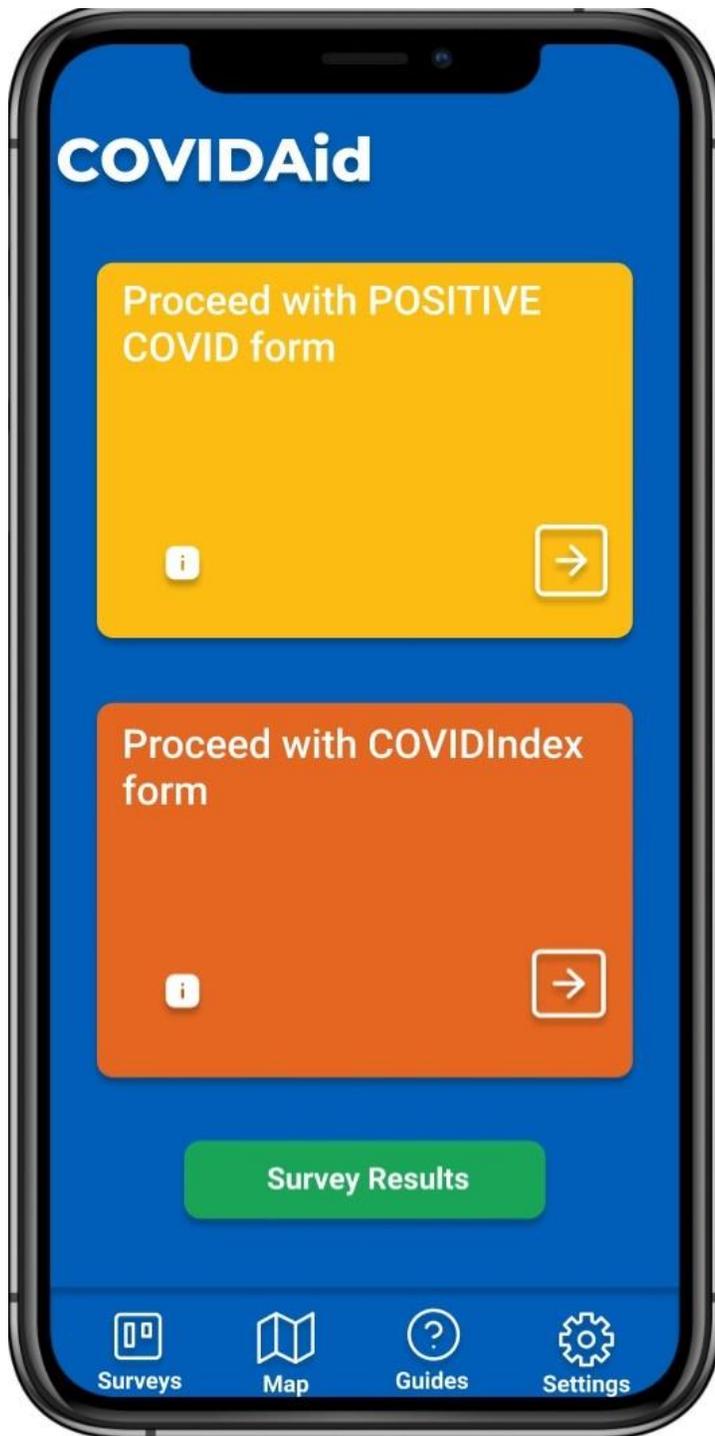
## E. USABILITY AND PROTOTYPING

For this section, please open the following link which showcases our application prototype:

<https://www.figma.com/file/VYrtFhiNBIBhUGbNdRCpgk/COVIDAidProtoype?node-id=0%3A1>

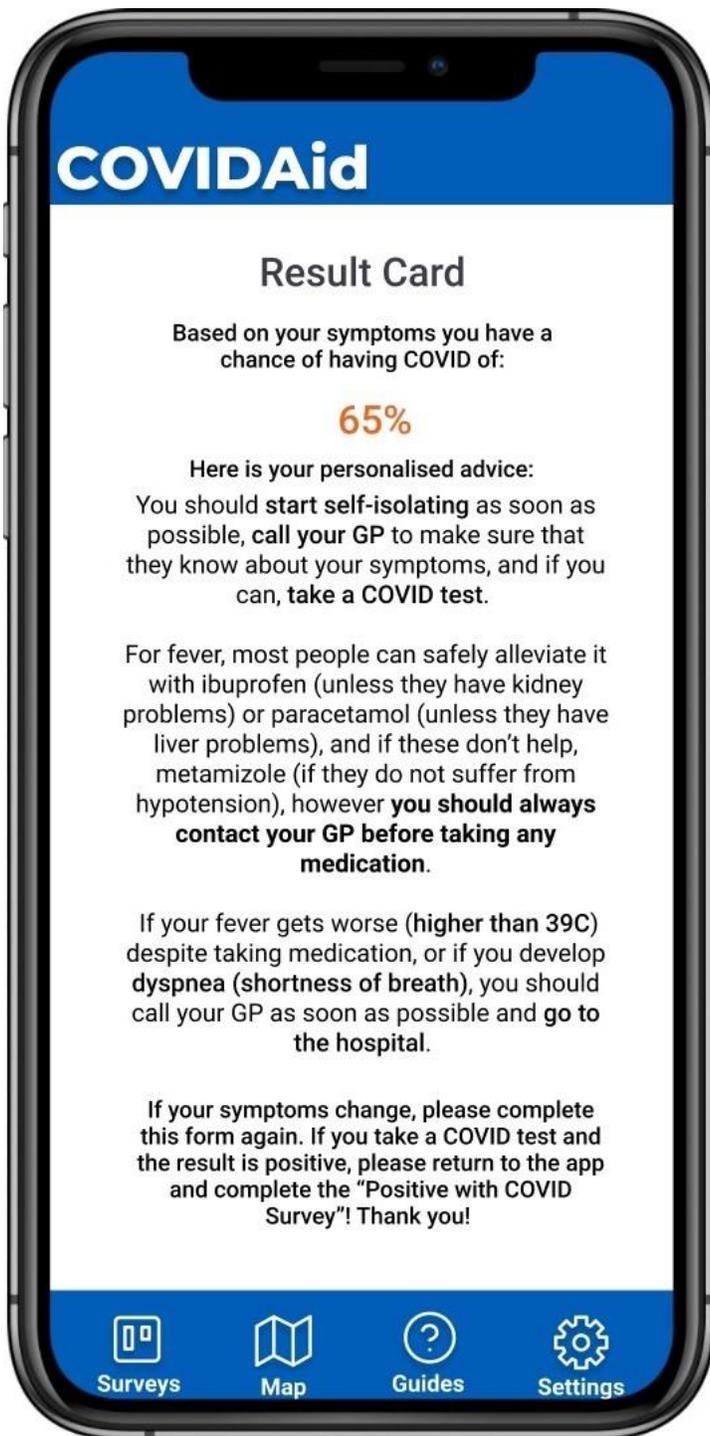
This was made using Figma which is a digital design and prototyping tool. Furthermore, please see below where we include a selection of screenshots showing our prototype, each accompanied with a brief description.





This is the screen the user is greeted with after logging into their account. From here, they get easy access to the most important features of COVIDAid - COVIDIndex and Positive with COVID Survey. If they have completed any of the surveys before, they can also see their previous results. The users can also check the COVIDMap, the Helpful guides or manage their account details.





This screen of the app follows a series of forms (namely 'Mandatory Information', 'Symptoms' and 'Personal Information') in which a user inputs information, which is then passed into a Machine Learning algorithm which calculates the probability of the user having COVID-19 (COVIDIndex). Depending on the 'COVIDIndex', in addition to the input provided prior, the user receives tailored advice which recommends the next best course of action.

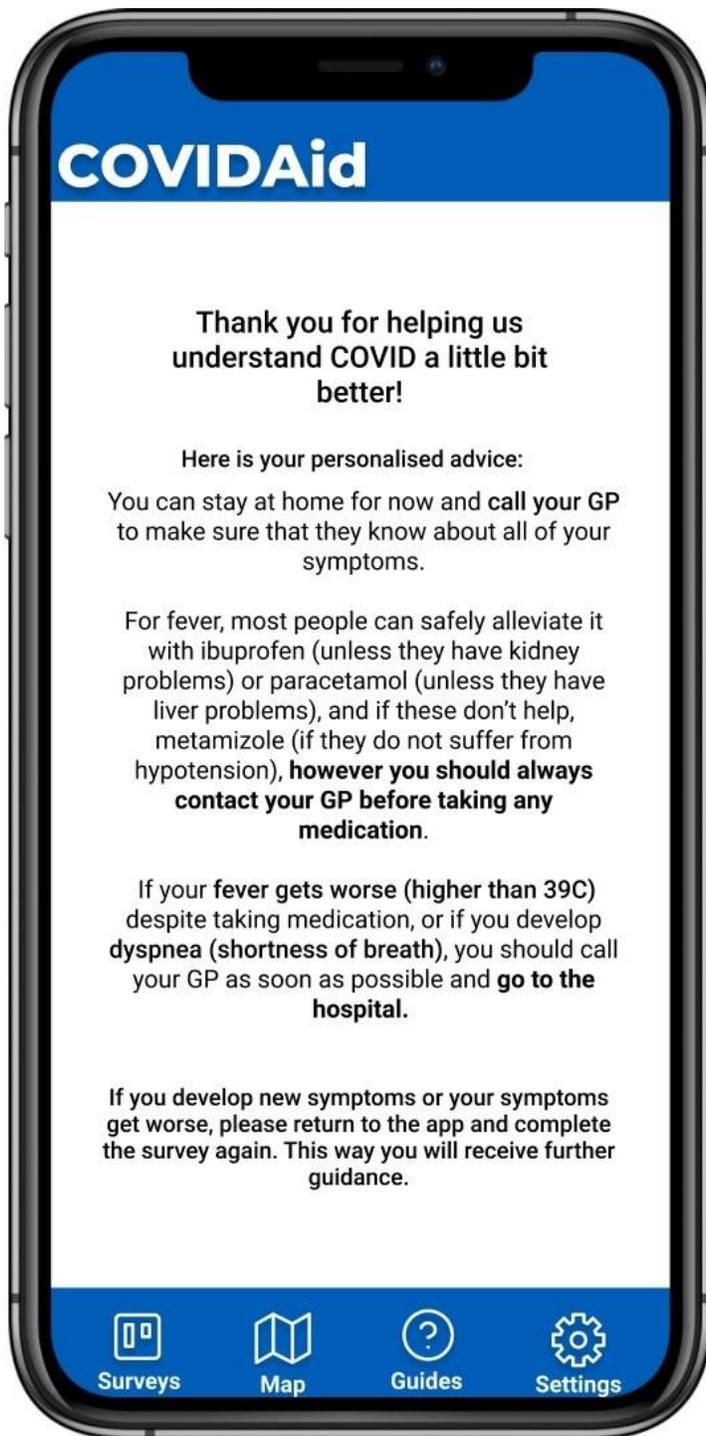
In this case, the user introduced the following information:

**Mandatory Information:** "Age: [35-44]", "Sex: Female", "Did you get into contact with anyone positive with COVID in the past 14 days? Don't know"

**Symptoms:** "Fever: [38°C-38.4°C]", "Headache: Persistent", "Diarrhoea"

**Personal Information:** "Asthma", "Smoking: Never", "Drinking: once a month", "Wash your hands: 2-3 times a day", "Location: Birmingham".





This is the screen presented to the user after completing the Positive with COVID Survey. After completing the three main sections of the survey, the information related to COVID symptoms that they have provided is added to the database of training data which is used to update the Machine Learning Model, so its results can get more accurate. If they request further advice, they will get personalized advice based on their symptoms.

In this case, the user introduced the following information:

**Mandatory Information:** "Age: [35-44]", "Sex: Female", "Did you get into contact with anyone positive with COVID in the past 14 days? Don't know"

**Symptoms:** "Fever: [38°C-38.4°C]", "Headache: Persistent", "Diarrhoea"

**Personal Information:** "Asthma", "Smoking: Never", "Drinking: once a month", "Wash your hands: 2-3 times a day", "Location: Birmingham".



This screen highlights only one of the multiple views of our COVIDMap, which is the Possible Cases view. The app can predict where it is more likely that a new focus of infection will appear, based on how many submissions of COVIDIndex forms with a high likelihood of being COVID positive have been received in previous days. The zones on the map where it is likely that a new focus might appear are color-coded to better represent the risk.



This screenshot shows the 'Helpful Guides' section of our application. This section includes links to relevant websites such as the 'NHS', 'WHO' and 'GOV.UK'. Additionally, the app includes the latest information regarding breakthroughs related to COVID-19 as well government rules which must be abided, such as lockdown measures. The user is also able to search for relevant advice.





This is the screen that appears when the users access COVIDAid for the first time. You can observe the app logo at the top of the screen, followed by different ways of opening the app: if they have an account, they can log in and get full access to the app; if they do not have an account, they can create one with minimum information and credentials required or they can choose to continue as a guest. However, if they continue as a guest, they can only access the COVIDMap and the Helpful Guides sections, as both COVIDIndex and Positive with COVID Survey require the user to be logged in.



# F. ETHICS AND PROFESSIONAL PRACTICE

## Code of Conduct for COVIDAid:

### Respecting Privacy.

- Our product and services are built based on an important factor: privacy. We understand the magnitude of sensitive data we handle and we have designed the systems so as your data is safeguarded.
- We have placed multiple security measures, and we are continuously developing even more ways to protect.
- We only store the bare minimum amount of data required to give you optimum results, and to train the complex Artificial Intelligence models we use non-localised data hence making it anonymous.

### Hold safety paramount.

- The core ideology of our product is to help everyone who is using it. In these uncertain times, where medical systems around the world are a bottleneck, we have designed our products to help you.
- The medical advice we provide through our product has been carefully designed to suggest certain medications that might help you recover from some of the symptoms, and the probability of you being infected is based on complex algorithms designed to give you the best possible result.
- If any issue is to rise, it will be communicated to the public appropriately, clearly and honestly.

### Treat all persons fairly and equally.

- All employees are treated with equality, tolerance and respect, irrespective of their age, colour, disability, ethnicity, family status, gender identification, union membership, nationality, military status, race, religion or belief, sexual orientation, and we are proud of providing a safe and natural environment for our employees to work in.
- We do not tolerate any hateful or toxic behaviour by any employee. The technology has been designed to be inclusive, accessible and we constantly try to improve the accessibility and inclusivity of our product.

### Act as a faithful agent.

- Our employees are held at a position of trust and they have a moral responsibility to provide honest, credible and valuable work to the employer, other employees, clients, users and the general public. The work done by the employees will strictly adhere to the local, regional, national, and international rules and regulations to which our products belong.
- An employee is required to work in an area they have expertise in. If they identify a lack of ability during an assignment, they will disclose this.



## Professional review.

- The work done by the employees will be subject to stringent tests before reaching the hands of users.
- Extreme care and attention are given to detect any potential risks in the core Artificial Intelligence models that are in place to determine the probability of the user infected.
- The tests will be thorough, perceptive, and objective.

