SOFT7011

Analysis And Design of The Crownpass Owner Application.

Student Number: 19182799

Author: Sam Trowbridge

Word Count: 2221

[**Introduction**](#_3fd3poip8q4e)3

[**Analysis & specification of quality requirements**](#_6fwv4mj92z96)3

[1.1 Scalability](#_s91p9fvp38e4) 3

[1.2 Performance](#_xw6bep4gsh2o) 3

[1.3 Reliability](#_y5e0a9wus0d5) 4

[1.4 Security](#_3v5nucmzefv) 4

[1.5 Privacy](#_vsvonm5bretk) 4

[1.6 Usability](#_2kz6fb8oyoam) 4

[**Analysis and discussion of the PLES aspects associated to the system**](#_jae87h5mplip)5

[2.1 Professional Issues](#_82i86minxq84) 5

[2.2 Legal Issues](#_458z4qj1fv8r) 5

[2.3 Ethical Issues](#_bd8iu4vpixda) 5

[2.4 Social Issues](#_yxust1gav72x) 6

[**Architectural Design**](#_7g0fg720hwzn)7

[3.1 Overall Architecture](#_o363ppoimkj9) 7

[3.2 Area Application Architecture](#_ci9nx9758i1o) 8

[3.3 Area Application Use Case Diagram](#_vastjvpcdd0m) 9

[3.4 MicroServices](#_eed2ybgw79u8) 10

[3.5 Databases](#_nir4p9rvttr) 15

## 

## Introduction

The Crownpass Owner Application is a subsystem of a master application, it aims to provide area owners the services to manage and control the customers (Crownpass Holders) entering a facility during a pandemic scenario. The intention of the subsystem is to provide the functionality that allows an area owner to set entry standards desired of potential customers. To confirm these requirements the customers require a Crownpass Holder account in which they can safely store their infection state for example negative, vaccinated, double vaccinated etc.

The Crownpass Holder subsystem provides a unique QR code for each user, the Owner Application aims to implement a QR code reader during a check in function. Which can ensure each Crownpass Holder meets or exceeds the owner's desired state of infection. This report focuses on the design of the Owner application, starting with non-functional requirements, PLES issues the application may face and the architectural design including microservices and databases necessary.

## Analysis & specification of quality requirements

### **1.1 Scalability**

**QR-SC-01: Crownpass Owner.** The system must provide services to around 400-500,000 public areas across the UK, of whom it is expected that 90% will register within the first month. These services must also provide the ability to scan and receive infection status information of any of the potential 45 million Crownpass holders.

**QR-SC-02: Staff Accounts.** Each area should be able to create multiple staff accounts the number of which can vary greatly, However the number of staff in large areas such as hospitals and public arenas can be extensive. As many as 10-15000 staff accounts may be created for one area.

**QR-SC-03: Check-in.** Numerous areas should be able to “Check-in” Crownpass holders simultaneously, during common business hours and holidays it is likely that 10s of 000s of areas could be performing a Crownpass Holder “Check In” at any given time.

### **1.2 Performance**

**QR-PF-01: Registration.** The system must allow an area owner to register their establishment, a successful registration attempt should receive a response within 30 seconds.

**QR-PF-02: Check-in.** A successful Crownpass Holder check-in response time should be no longer than 5 seconds.

**QR-PF-03: Notification.** The system should inform all Crownpass Holders that were within the same area at the same time as a positive tested Crownpass Holder within 30 seconds.

**QR-PF-04: Emergency Evacuation.** The emergency evacuation service must be completed successfully within one minute.

### **1.3 Reliability**

**QR-RL-01: Notification.** The system should be capable of always receiving a positive notification, an area should also only receive an evacuation recommendation if the Crownpass Holder within question is in that area.

**QR-RL-02: Check-in.** A correct check-in attempt should have a 99% success rate and all QR codes must be unique to each assigned Crownpass Holder.

**QR-RL-03: Crownpass Owner.** The cloud servers must be capable of handling a massive increase of check-in activity at any given time of the day.

### 1.4 Security

**QR-SE-01: Passwords.** The system must hash and salt all area and staff account passwords to reduce risk of area or staff accounts being hacked for menacing behaviour or data leaks.

**QR-SE-02: Area Registration.** All area’s must be assigned a unique ID and details must be checked for duplication within the Area Account Database.

**QR-SE-03: Server & database security.** Measures must be taken to ensure the security of the implemented servers and databases to avoid data leaks or system failure.

### 1.5 Privacy

**QR-PV-01: Entry Management.** The database will not store any private Crownpass Holder information besides ID and infection status to avoid the risk of descrimination.

**QR-PV-02: Notification.** The service must not display any personal information of the Crownpass Holder that has received a positive test result.

### 1.6 Usability

**QR-UB-01: Area Registration.** It should take no more than two minutes for an area owner to successfully complete the registration process.

**QR-UB-02: Emergency Evacuation.** The service must be available and always displayed as an option to ensure quick evacuation in the event of a positive test result, thereby decreasing the risk of an infection outbreak. A confirmation prompt must be provided before service enaction to avoid accidental evacuation.

**QR-UB-03: Emergency Evacuation.** A user of the application should be able to identify how to evacuate the area on their first attempt within 10 seconds.

## Analysis and discussion of the PLES aspects associated to the system

### 2.1 Professional Issues

* Privacy of users is vital, therefore any company or individual working on the development and maintenance of the system are obligated to adhere to user confidentiality. The companies and individuals also have a right to the privacy of their contribution to this application which should be kept unless given permission.
* Use of the application is voluntary; therefore, area owners must be made aware of the seriousness of the situation and the importance of tracking positive cases. In order for this application to be successful the majority of area owners’ participation is required as well as strict usage of the application.
* The Crowpass Holder application is a vital part of the system, without this application the entire system becomes redundant. Completion of the Crownpass Holder is required in order to efficiently test the area owner application functions.

### 2.2 Legal Issues

* In order to adhere to “Data Protection Act 2018” (ref) a Crownpass Holders personal information or identification such as name and address will never be acquired by the area owner application. This also includes in the event of a positive test result or infection status entry check failure.
* Liability, how is responsibility assigned in the event of Crownpass Holder data leakage or other forms of malicious attacks?
* Copyright, who owns this application? As this is a free application, focused on the protection of the public people, it may not require copyright.

### 2.3 Ethical Issues

* As it is expected that a large number of the public will use the application, the system ought to integrate different methods of check-in and be applicable to multiple types of users. By providing the ability to print a Crownpass QR code a user is not required to have a mobile device.
* Transparency regarding the data gathered, why and what it is required for, may assist in building public trust of the application. Also making the application code and algorithms public will provide the opportunity for scrutiny and show that no ill intent is intended.
* The application must be completely voluntary, users can download and use the application of their own free will. Users must also have the control to delete their account and all related data if desired.

### 2.4 Social Issues

* Discrimination issues apear to be fairly common in these scenarios, therefore measures must be put in place to protect users personal information, particularly in the case of positive test results.
* The system should not be discriminatory towards Anti-vax ideology in any way. Crownpass users should not feel pressured to be vaccinated or any distress not being so. Therefore, steps must be taken to ensure this information is secure, the area owner application should not display these details in any shape or form.

## Architectural Design

### 3.1 Overall Architecture

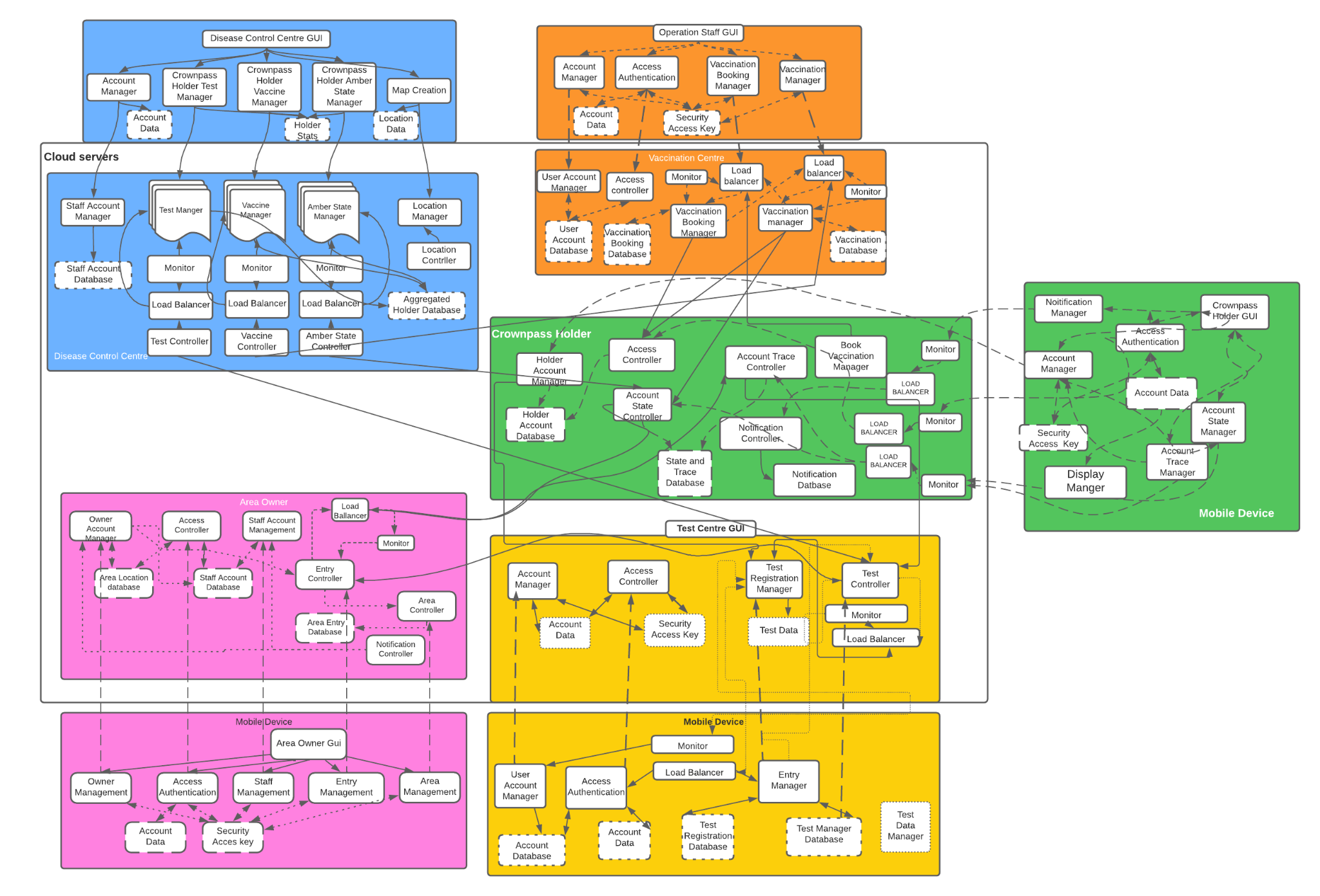


Figure 1

Figure 1 shown above displays the design architecture of the overall application which includes 5 subsystems; Crownpass Monitor Desktop App (blue), Crownpass Vaccinator App (orange), Crownpass Holder App (green), Crownpass Owner App (pink) and Crownpass Tester App (yellow). The Crownpass Holder subsystem is the most vital application with which all subsystems require communication and will not be able to function correctly without, therefore it has a central position. Various microservices will be implemented to allow these subsystems to communicate important information between them, displayed by the arrows between cloud servers.

### 

### 3.2 Area Application Architecture

### 

Figure 2

A closer look at the Crownpass Owner subsystem architecture can be seen in figure 2, it is split into two sections: Mobile Device (GUI) and Cloud Servers. Microservices allow the GUI to communicate with the cloud servers to import and export data, however since a single microservice is best suited to perform a single task a large number are often required. This subsystem also requires communications with other subsystems, the Crownpass Tester subsystem will provide any updates to the entry conditions available.

The Crownpass Holder subsystem communications are vital to the successful operation of this application. The check-in functionality first requires confirmation that the customer is a Crownpass Holder after which a confirmation of whether the entry requirements are met by that holder. Also, the Crowpass Holder subsystem must provide notifications of positive test results within an area. The entry controller server could face enormous increases and decreases of traffic from opening and closing times, due to this a load balancer and monitor will work in tangent to provide more or less instances depending on this traffic.

### 3.3 Area Application Use Case Diagram

### 

Figure 3

### 3.4 MicroServices

|  |  |
| --- | --- |
| Service Name | Owner Management |
| Description | The service runs on an owner's mobile device to enable users to set up area registration, login to area owner accounts, create staff accounts and receive notifications. |
| Provided Services 1 | Function: Register new area; |
| Parameters: password, AreaName, Postcode, County, City, Street, No.; |
| Requested Services 1 | Service: Owner Account Manager; |
| Operation: Input new area Information; |
| Parameters: AreaID, password, AreaName, Postcode, County, City, Street, No., SecurityKey; |
| Reply: Confirmation; |
| Provided Services 2 | Function: Update area details; |
| Parameters: Password AreaName, Postcode, County, City, Street, No.; |
| Requested Services 2 | Service: Owner Account Manager; |
| Operation: Update area; |
| Parameters: AreaID, password, AreaName, Postcode, County, City, Street, No., SecurityKey; |
| Reply: Confirmation; |
| Provided Services 3 | Function: Create Staff Account; |
| Parameters: Username, Password; |
| Requested Services 3 | Service: Owner Account Manager; |
| Operation: Input new staff account login details; |
| Parameters: AreaID, StaffID, Username, Password, SecurityKey; |
| Reply: Confirmation |
| Requested Services 4 | Service: Notification Controller; |
| Operation: Receive Notifications of positive results; |
| Parameters: AreaID, CrownpassID; |
| Reply: Alert to possible infected |

|  |  |
| --- | --- |
| Service Name | Staff Management |
| Description | A service that allows staff to set their initial passwords and receive notifications. |
| Provided Services 1 | Function:Set initial password; |
| Parameters: AreaID, StaffID, Password, Securitykey; |
| Requested Services 1 | Service: Set Password; |
| Operation: Set staff password; |
| Parameters: AreaID, StaffID, Password, Securitykey; |
| Reply: Confirmation; |
| Requested Services 2 | Service: Notification Controller; |
| Operation: Receive Notifications of positive results; |
| Parameters: AreaID, CrownpassID; |
| Reply: Alert to possible infected |

|  |  |
| --- | --- |
| Service Name | Access Authentication |
| Description | A service that authenticates owner & staff login. |
| Provided Services 1 | Function: Identify owner requesting access to system |
| Parameters: AreaID, Password, SecurityKey; |
| Requested Services 1 | Service: Owner Account Manager; |
| Operation: Authenticate login details; |
| Parameters: AreaID, Password, SecurityKey; |
| Reply: Login Confirmation or Failure; |
| Provided Services 2 | Function: Identify staff requesting access to system; |
| Parameters: StaffID, Password, SecurityKey; |
| Requested Services 2 | Service: Staff Account Manager; |
| Operation: Authenticate login details; |
| Parameters: AreaID, Password, SecurityKey; |
| Reply: Login Confirmation or Failure |

|  |  |
| --- | --- |
| Service Name | Entry Management |
| Description | The service provides area owners the ability to set and change the requirements to be met for Crownpass Holders to be allowed entry. Also allows Staff to check-in Crownpass Holders. |
| Provided Services 1 | Function: Set infection state entry requirement; |
| Requested Services 1 | Parameters: AreaID, Password, Entrylvl; |
| Requested Services 1 | Service: Owner Account Manager; |
| Operation: Set entry requirement; |
| Parameters: AreaID, Password, Entrylvl, SecurityKey; |
| Reply: Confirmation |
| Provided Services 2 | Function: Check-in; |
| Parameters:CrownpassID, Entrylvl; |
| Requested Services 2 | Service: Entry Controller; |
| Operation: Entry Validation; |
| Parameters: CrownpassID, Entrylvl; |
| Reply: Success or Failure Confirmation; |
| Requested Services 3 | Service: Authorization; |
| Operation: Crownpass\_holder\_check; |
| Parameters: CrownpassID; |
| Reply: Success or Failure Confirmation; |
| Requested Services 4 | Service: Infection Status; |
| Operation: Entry Requirments Met; |
| Parameters: CrownpassID, Entrylvl; |
| Reply: Success or Failure Confirmation; |

|  |  |
| --- | --- |
| Service Name | Area Management |
| Description | A service that enables area owners and staff to track and manage the number of Crownpass holders within the area and initiate an emergency evacuation. |
| Provided Services 1 | Function: Check-out; |
| Parameters: AreaID, CrownpassID, Time; |
| Requested Services 1 | Service: Area Controller; |
| Operation: Check-out; |
| Parameters: AreaID, CrownpassID, Time; |
| Reply: Confirmation; |
| Provided Services 2 | Function: Reset State; |
| Parameters: AreaID, Time; |
| Requested Services 2 | Service: Area Controller |
| Operation: Area Reset |
| Parameters: AreaID, Time; |
| Reply: Confirmation; |
| Provided Services 3 | Function: Emergency Evacuation; |
| Parameters: AreaID, CrownpassID’s, Time; |
| Requested Services 3.1 | Service: Area Controller; |
| Operation: Alert Crownpass Holders; |
| Parameters: AreaID, CrownpassID’s; |
| Reply: Confirmation; |
| Requested Services 3.2 | Service: Area Controller; |
| Operation: Area Reset; |
| Parameters: AreaID, Time; |
| Reply: Confirmation; |

### 

### 3.5 Databases

|  |  |
| --- | --- |
| Database Name | Area Database |
| Description | The database stores area information given during the registration process. |
| Type | Relational Database |
| Design | Tables: Owner Account |
| Table: Owner Account | Fields: AreaID, Password, AreaName, Postcode, County, City, Street, No., EntryLvl, SecurityKey |

|  |  |
| --- | --- |
| Database Name | Staff Database |
| Description | The database stores staff account information and the area the staff are assigned to. |
| Type | Relational Database |
| Design | Tables:StaffAccounts |
| Table: StaffAccounts | Fields: AreaID, StaffID, Username, Password, SecurityKey |

|  |  |
| --- | --- |
| Database Name | Area Entry Database |
| Description | The database stores Crownpass Holders that have checked into areas over the past two weeks |
| Type | NoSQL Database |
| Design | Semi-structured data model in figure 4 |
| Table: CrownpassEntries | Fields: AreaID, CrownpassID, EntryDate, EntryTime, ExitDate, ExitTime |

## 

Figure 4