Subsystem D – Operations of interfaces

|  |
| --- |
| <<interface>>  Login |
| +operatorLogin(email: string, password: string)  +operatorRegister(email: string, password: string)  +viewDetails()  +updateDetails() |

|  |
| --- |
| <<interface>>  View Bookings |
| +viewBookings(vaccinationSite: string, date: date, timeslot : timestamp) |

|  |
| --- |
| <<interface>>  Update Booking State |
| +updateBookingState(bookingId: int) |

|  |
| --- |
| <<interface>>  Add Booking |
| +addBooking(vaccinationSite: String, holderId: int, date: date, timeslot : timestamp) |

|  |
| --- |
| <<interface>>  Record Vaccination |
| +recordVaccination(vaccination: Vaccination)  +updateVaccinationState(holderId: int, vaccinationState: string) |

|  |
| --- |
| <<interface>>  Remove Booking |
| +removeBooking(vaccinationSite: String, holderId: int, date: date, timeslot : timestamp) |

Subsystem D – Textual documentation of Components & Connectors

Vaccine operator = Operation staff of vaccination stations/centres

|  |  |
| --- | --- |
| Service Name | Service Provided |
| Login Manager | A microservice that allows users of the app to send requests relating to the vaccine operator accounts to the cloud |
| Authentication Manager | A microservice running on the cloud to process a vaccine operators requests related to managing (or creating) their user account by interacting with the User Database. |
| Booking Requests | A microservice that relates to queries a vaccine operator will have regarding bookings and allows an operator with some information regarding a booking to send a request to the cloud. |
| Booking Manager | A microservice running on the cloud that will query the Booking database depending on what request it has received. It is capable of verifying a booking can be created and also provides access to all bookings in the database. |
| Vaccination Manager | A microservice that allows a vaccine operator to send recorded vaccine information to the cloud which in turn allows for vaccination states to be updated. |

|  |  |
| --- | --- |
| Database Name | Database Description |
| User Database | A database that stores information on all registered users in the crownpass system, including emails, passwords, vaccination states, account roles and unique crownpass ID’s. For my subsystem this is concentrated on crownpass holders and vaccine operators. Through queries with the authentication manager it can verify the correct password has been entered when logging in, update account information upon request from the mobile app and verify a Crownpass HolderID exists and is correct for the booking manager. |
| Booking Database | The booking database contains all information related to bookings. This includes the vaccination site associated, the date and time a booking is scheduled to take place and whether they are complete or not. |
| Vaccination Database | The vaccination database contains all information recorded in each unique vaccination. This includes the holderID associated, the bookingID, the date and time of injection, the vaccine name and batch number, the vaccination site and the ID of the Vaccine Operator that logged it. The vaccination state is stored on the cloud and the state transition is triggered whenever a new entry is recorded in the database by looking at the HolderID associated with it. |

|  |  |
| --- | --- |
| Connector Name | Connector Description |
| Login | Provided by the Authentication Manager on the cloud and required by the Login Manager on the app. This connector associates account information entered with the login manager with a request to the cloud to login with a password, or view / update data associated with the account. |
| Check Password | Provided by User Database and required by Authentication Manager. Verifies the password entered during login matches the one stored in the database (after hashing). |
| Register New Operator | Provided by User Database and required by Authentication Manager. Adds a new vaccine operator the database with a email and password, and assigns them a ID. |
| Update Account Info | Provided by User Database and required by Authentication Manager. Allows a vaccine operator to update their account information stored on the cloud. |
| View Account Info | Provided by User Database and required by Authentication Manager. Allows a registered vaccine operator to view their information stored on the cloud. |
| Verify HolderID | Provided by User Database and required by Authentication Manager, Booking Manager & Vaccination Database. Verifies that a HolderID does exist in the system and gives that holders information. Useful for booking manager such that they can verify when receiving booking data for an add request that the holderID does exist, similarly for Vaccination DB ensuring the holderID exists but also ensuring the updated vaccination state is associated with the correct Crownpass Holder. |
| Crownpass Holder ID + Data | Required by User Database. From subsystem A crownpass holder information regarding new crownpass accounts & current test states. |
| Requested Booking Info | Required by Booking Requests. Contains all booking information associated with a unique request, which could be some of a holderID, a date time or a vaccination site. |
| Add Booking | Provided by Booking Manager and required by Booking Requests. When a vaccine operator is asked to add a booking to the system they send through the requested time slot and holderID associated with the booking. |
| Remove Booking | Provided by Booking Manager and required by Booking Requests. Allows vaccine operators to remove a booking from the system by searching via a HolderID or datetime. |
| View Bookings | Provided by Booking Manager and required by Booking Requests. Allows vaccine operators to view all bookings at a vaccination site or on a specific date / time. |
| Update Booking State | Provided by Booking Manager and required by Booking Requests. Allows vaccine operators to update a booking state to completed when the associated vaccination has been recorded. |
| Check time available | Provided by Booking Database and required by Booking manager. Verifies that a request timeslot at a given vaccination centre is indeed available for a requested booking. |
| Update Table | Provided by Booking Database and required by Booking manager. Allows for the updating of bookings via add, remove or update booking state requests. |
| View entry | Provided by Booking Database and required by Booking manager. Allows a booking manager to return all booking entries given some criteria received from the app. |
| New Vaccination Data | Provided by Vaccination Manager. Includes all information when each vaccination is performed. |
| Record Vaccination | Provided by Vaccination Database and required by Vaccination Manager. Allows the vaccination manager to take the performed vaccination information and record it and send it to the cloud. |
| Update Vaccination State | Provided by Vaccination Database and required by User Database. Triggers the state transition of vaccination state associated with a HolderID when a vaccination is recorded concerning them. |