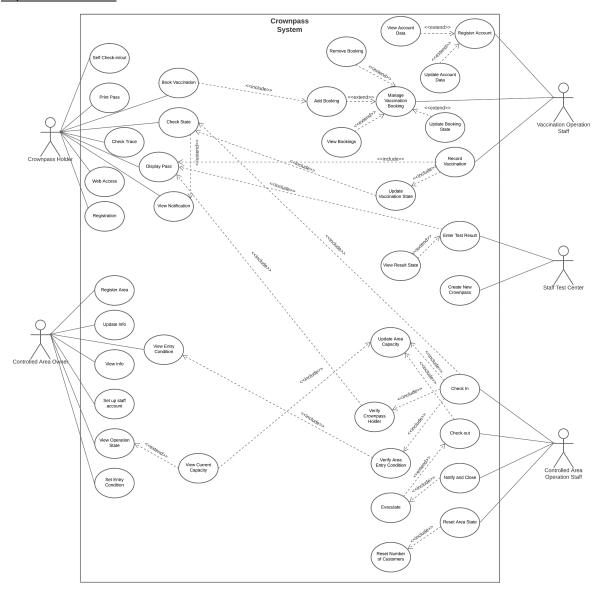
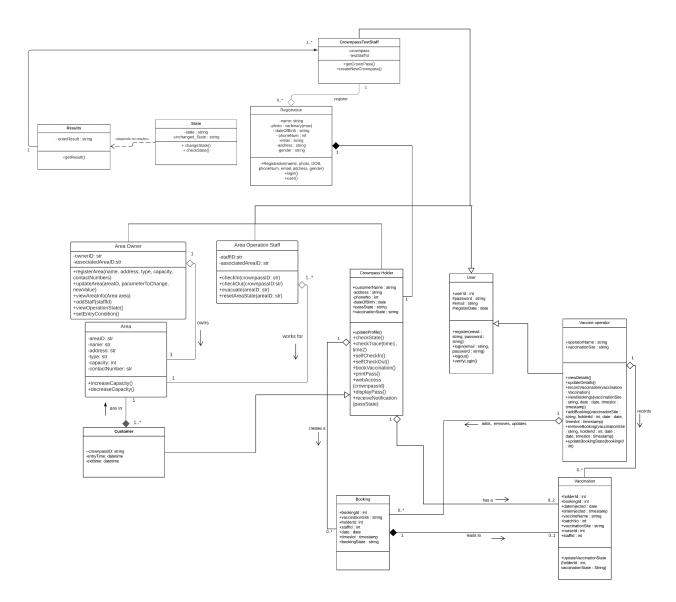
G17 - Overall system COMP6030

1a) Use case model

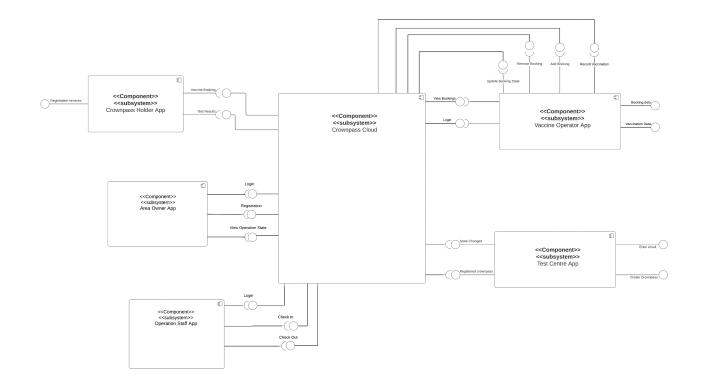


 $\underline{1c}$ NOTE: Spec does not require a overall class diagram, but in moodle and lectures we were advised to create one e.g. from week 7, moodle COMP6030:

• Class diagram: Each member of the group should identify the classes in your subsystem, and to work out what are the needed attributes and methods of the classes and what are the relationship between them. Some classes may be in more than one subsystems, thus can be identified by more than one group members, but such a class may well have different attributes and methods in different subsystems. The group should work together to merge the sets of classes together and draw a big class diagram for the whole system that includes all classes.

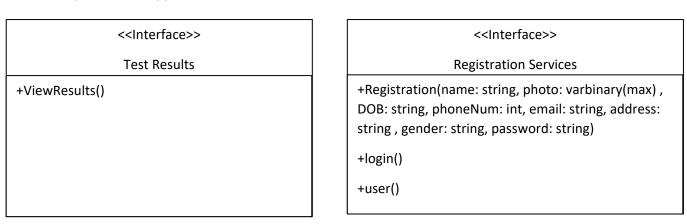


2a) Overall Architecture of the whole system



Interface specification:

Crownpass Holder App:



Vaccine Operator App:

<<interface>>
Login

+operatorLogin(email: string,

password: string)

+operatorRegister(email: string,

password: string)
+viewDetails()
+updateDetails()

<<interface>>
Update Booking State

+updateBookingState(bookingId: int)

<<interface>>
Record Vaccination

+recordVaccination(vaccination:

Vaccination)

+updateVaccinationState(holderId:

int, vaccinationState: string)

<<interface>> View Bookings

+viewBookings(vaccinationSite: string, date: date, timeslot: timestamp)

<<interface>>
Add Booking

+addBooking(vaccinationSite: String, holderId: int, date: date, timeslot:

timestamp)

<<interface>>
Remove Booking

+removeBooking(vaccinationSite: String, holderId: int, date: date,

timeslot : timestamp)

Test Centre App:

<<interface>> Enter Result

+Result:(Result:string)

<<interface>> Create Crownpass

- +AddCrwonpass: (crownpass: sting); +Registeration: (call
- +Registeration; (call registeration)

<<interface>> State

- -State:(State:string)
- +checkState()
- +Changed_State:string

<<interface>> Registeration

+ Registeration: (name: string, photo, DOB: string, gender: string, address: string,

Birthdate: int, phoneNum: int,

email: string)

-login -user

Area Owner App:

<<interface>>
Owner Login

+ownerLogin(email: str, password:str)
+ownerRegister(email:str,
Password:str)

<<interface>>
Registration

+registerArea(name: str, address: str, type: str, capacity: int, contact_no: str) +updateRegistrationData() +viewRegistrationData()

<<interface>>
View Operation State

+viewOperationState()

Operation Staff App:

<<interface>>
Staff Login

+staffLogin(email: str, password: str)

<<interface>>
Check In

+checkIn(crownpassId)

<<interface>>
Check Out

+checkOut(crownpassID)