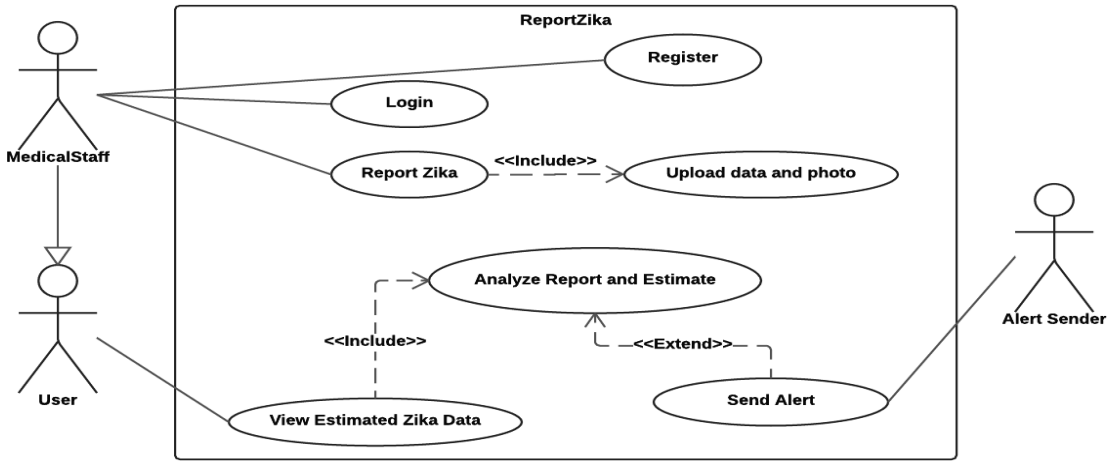
Solver: Yew Shao Jie

Email Address: YEWS0012@e.ntu.edu.sg

1)

a) The project is expected to require long term efforts. With Agile method and interactive and incremental development model, it will help to provide continuous attention to technical excellence and good design and even late changes in the requirements are not a problem.

b)



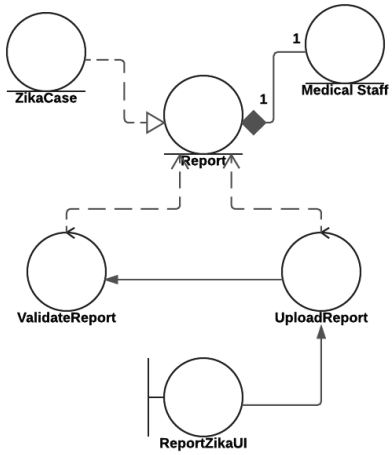
c)

Use case title:	Report Zika in
Description:	The use case allow actor to submit zika case information such as its date time, location and photograph to the cloud database of ReportZika.
Actors:	Medical Staff
Precondition:	The actor must be logged
Post condition:	Zika case is successfully uploaded to the cloud database
Flow of Event:	1) The user select report zika case button 2) The application prompt user to input report information such as time, date and location 3) The user input the zika case information 4) The user select a photo to upload if there is one 5) The user submit the report 6) The application validate the information submitted 7) The application upload the information to the cloud database 8) The application display successfully reported 9) Use case ends
Alternate Flow:	AF-6 - The information submitted is invalid 1) The application display invalid information 2) The application resume to step 2

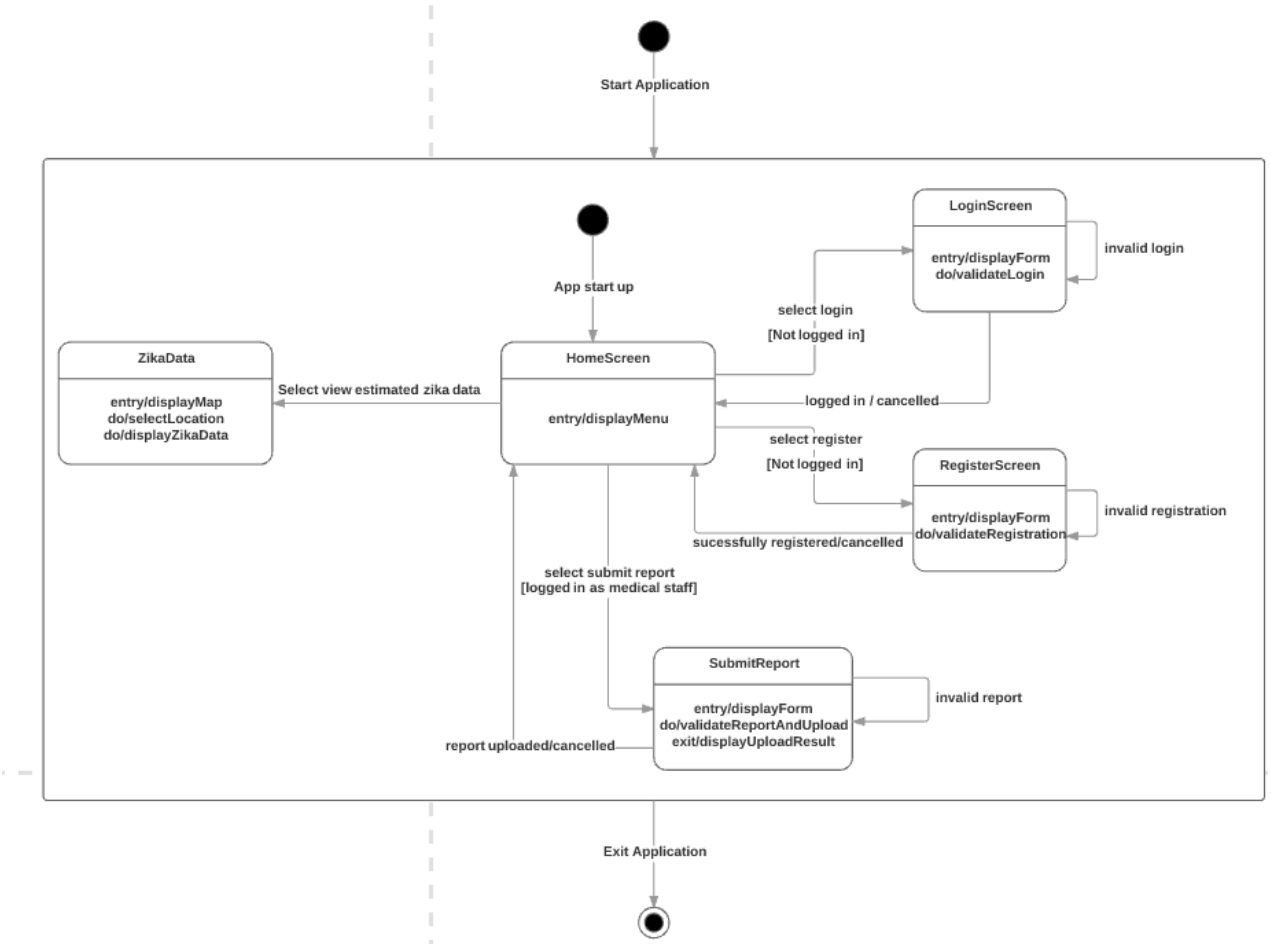
Exceptional Flow:	**EX1 –** User select cancel to exit the use case 1) Use case ends

2)

a)



b)



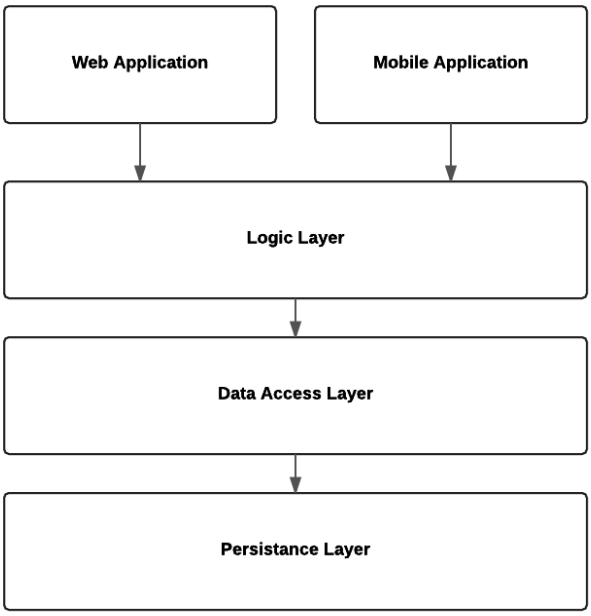
c) Two different methods to estimate the time to complete each task are Historical Data and Wideband Delphi. After estimating the duration for each task and using the Work Breakdown Structure, we look at the critical path, which is a sequence of activities that take the longest to complete which will help to determine the shortest time in which the project could be completed.

3)

a)

i) Layered Architecture

- Presentation layer can be of multiple different application thus help to support the Web and Mobile application of ReportZika
- Logic Layer – provide service to presentation layer so that use can interact with the system
- Data access layer – provide data access implementation and database entities for manipulating information.



ii) Benefits:

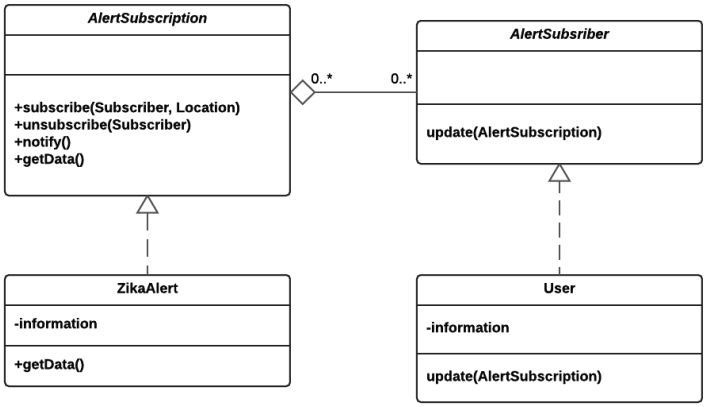
- Single responsibility
- Low coupling as does not need to know how other layers are implemented and only communicate through interfaces
- Only interact with layer just below
- Easy to maintain and make changes as changes to lower layer, all upper layers will get the update without need to update the upper layer. Thus web application and mobile application in presentation layer does not need to be updated when changes are made to the lower layers.

b)

i) Using the notification mechanism in observes pattern we could make use of its selective notification to notify only particular observers that are interested in the

selected interest. For example, when user location changed from a location to a new location. The user will only be able to receive the alert of the new location as alert of other location will be filtered out.

ii)



ZikaAlert is implements AlertSubscription
User implements AlertSubscriber
User register the interest in ZikaAlert by calling subscribe (Subscriber, Location)
ZikaAlert notify the registered AlertSubscriber by user.update(AlertSubscription)
Location parameter in subscribe (Subscriber, Location) of AlertSubscription is used for selective notification, so that in the notify() method, only alert will be sent to user in a particular location.

4)

a)

i)

Equivalence classes	Number of Zika case for alert
Valid	0<=number of zika case<=5
	6<=number of zika case<=20
	number of zika case>20
Invalid	number of zika case<0

For simplifying of test case later, let "**Every one other than those in Class B or Class A" be Class C** and for all Classes that are not Class A, B or C is Class F

Equivalence classes	Pregnancy status	
Valid	Class C: Every one other than those in Class B or Class A	"student"
	Class B	"couples planning pregnancy"
	Class A	"pregnant woman"
Invalid	Class F	"ABC"

ii)

Pregnancy status is a discrete value, thus no boundary

Equivalence classes	Number of Zika case for alert	Boundary
Valid	0<=number of zika case<=5	0,5
	6<=number of zika case<=20	6,20
	number of zika case>20	21
Invalid	number of zika case<0	-1

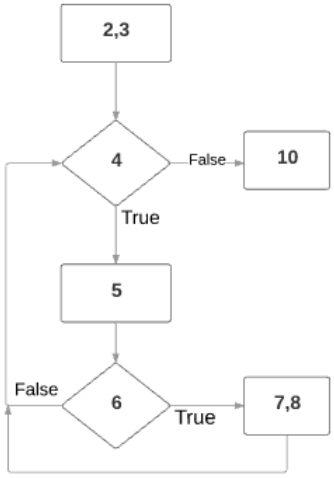
iii)

*__bolded__ are invalid equivalence class parameter for testing

Number of Zika case	Pregnancy status	Expected Alert Outcome	Expected Result
0	Class A	No alert	Accept
5	Class A	No alert	Accept
6	Class A	Alert	Accept
20	Class A	Alert	Accept
21	Class A	Alert	Accept
0	Class B	No alert	Accept
5	Class B	No alert	Accept
6	Class B	Alert	Accept
20	Class B	Alert	Accept
21	Class B	Alert	Accept
0	Class C	No alert	Accept
5	Class C	No alert	Accept
6	Class C	No alert	Accept
20	Class C	No alert	Accept
21	Class C	Alert	Accept
-1	Class A	No alert	Reject
-1	Class B	No alert	Reject
-1	Class C	No alert	Reject
0	***Class F***	No alert	Reject
5	***Class F***	No alert	Reject
6	***Class F***	No alert	Reject
20	***Class F***	No alert	Reject
21	***Class F***	No alert	Reject

b)

i)



ii)

	List<Town> towns	Position	Return value
2,3,4,10	Empy towns	(10,10)	Null
2,3,4,5,6,4,10	Yishun, (20,20)	(10,10)	Null
2,3,4,5,6,7,8,4,10	Sembawang, (10,10)	(10,10)	Sembawang, (10,10)