TE WHARE WĀNANGA O TE ŪPOKO O TE IKA A MĀUI

VICTORIA
UNIVERSITY OF WELLINGTON

SCHOOL OF ENGINEERING AND COMPUTER SCIENCE

Assignment Cover Sheet

Full Name: SONGBO WU

Student ID: 300422252 Course: SWEN432

Tutorial Day: _____ Tutorial Time: _____

Tutor's Name (if applicable): _____

Assignment No.: 1
Due Date: Wednesday 05 April 2017

CERTIFICATION OF AUTHENTICITY

I certify that this paper submitted for assessment is the result of my own work, except where otherwise acknowledged.

Signed: Songbo Wu *Date:05 April 2017*

Question 1. [10 marks] List the database write and update requests the application requires using plain English.

--1. Write records to drivers table.

--2. write records to timeTableForPassengers table.

--3. write records to dispatchAllocation table.

--4. write records to vehicles table.

--5. write records to dataPoint table.

--6. Write records to timeTable table.

--7.Update the password for driver 'pondy'.

--8. Driver 'pondy' comes to Wellington station thus update his current_position to 'Wellington'.

--9.Driver 'pondy' is assigned to a service thus to update his current_position to ‘FA4567’.

--10.Driver 'pondy' deregisters from his work thus to update current_position to ‘not_available’.

--11. Update the number of days per month for registered driver 'pondy'.

--12.Update the skill of driver 'pondy' from {'Matangi', 'Kiwi Rail'} to {'Matangi', 'Kiwi Rail','Guliver'}

--13. Update the status of a vehicle.

--14. Update the current_position to destination 'Upper Hutt' of driver 'pondy'.

--15. Update the status to destination 'Upper Hutt' of vehicle 'FA4567'.

--16.Update the total distance travelled for vehicle ‘FA4567’ .

--17.Update the total distance travelled for vehicle 'FA1122' .

Question 2. [12 marks] List the read requests the application requires using plain English.

--1. Get the current password of driver 'pondy'. ;

--2. Get the number of days per month for registered driver 'pondy'.

--3. Publish a time table for passengers.

--4. Extract a list of departure stations with time in descending order

--5.Find the vehicle in 'Wellington'.

--6.Find the driver in 'Upper Hutt' with the skill 'Matangi'.

--7.Select the last data point for a service on a day.

--8.Select all Data points for a service on a day in a time interval 10 seconds for vehicle 'FA4567' with driver 'pondy'

--9.Get the north direction neighbours for a given data point "-41.2262 IN services_no 1 and lineName 'Hutt Vale Line' "

--10.Get the south direction neighbours for a given data point "-41.2262 IN services_no 1 and lineName 'Hutt Vale Line' "

Question 3. [9 marks] Consider Cassandra data model design guidelines we discussed in lectures and list names of database tables the application requires using plain English. Recall, Cassandra tables strongly depend on requested queries. If there is no queries needing a table, the table should not exist. (Don't invent queries to justify the existence of any tables.) After each table name, list those queries you identified in your answer to question 2 that use the table.

--1. drivers table.

Get the current password of driver 'pondy'. .

Find the driver in 'Wellington' with the skill 'Guliver'.

--2. timeTableForPassengers table.

Publish a time table for passengers.

--3. dispatchAllocation table.

Extract a list of departure stations with time in descending order.

--4. vehicles table.

Find the vehicle in 'Wellington'.

--5. dataPoint table.

Select the last data point for a service on a day.

Select all Data points for a service on a day in a time interval 10 seconds for vehicle 'FA4567' with driver 'pondy'

--6. timeTable table.

Get the north direction neighbours for a given data point "-41.2262 IN services_no 1 and lineName 'Hutt Vale Line' ".

Get the south direction neighbours for a given data point "-41.2262 IN services_no 1 and lineName 'Hutt Vale Line' ".

--7. days_registered_counts table.

Get the number of days per month for registered driver 'pondy'.

--8. Optimize database solution just for iPhone application queries

Question 4. [20 marks] Create data model using CQL 3 statements that support the requirements. To answer questions, use Cassandra CCM. In your answers, copy your CCM and CQL commands.

a. [5 marks] Create a cluster and a keyspace that will satisfy infrastructure and availability requirements above.

-- need ccm ;

--ccm create -n 6 WellingtonTranzMetro;

--ccm start ;

--ccm node1 cqlsh , --assume set node1 as the coordinator node .

CREATE KEYSPACE IF NOT EXISTS WellingtonTranzMetro WITH replication = {'class':'SimpleStrategy', 'replication_factor' : 3};

USE WellingtonTranzMetro;

b. [15 marks] Define tables listed in your answer to question 3 above. For the table definitions include any non default property settings. Optimize your database solution just for iPhone application queries you identified in question 2 above.

```
CREATE TABLE IF NOT EXISTS drivers(
   drv_name text PRIMARY KEY,
   cur_pos text,
   mobile  int,
   pwd text,
   skill set<text>
 );



CREATE TABLE IF NOT EXISTS timeTableForPassengers(
   station_name  text,
   time SET<int>,
   lineName text,
  PRIMARY KEY (station_name)
  );



CREATE TABLE IF NOT EXISTS  dispatchAllocation(
   station  text,
   vehicle  text,
   driver  text,
   services_no int ,
   time int,
   lineName text,
  PRIMARY KEY ((station, lineName), time)
  )WITH CLUSTERING ORDER BY (time desc)  ;
ALTER TABLE dispatchAllocation WITH compaction = { 'class' :  'LeveledCompactionStrategy'  };
```

```
CREATE TABLE IF NOT EXISTS vehicles(
  vehicle_id text,
  status text,
  vehicle_type  text,
  distance_travelled  double ,
   PRIMARY KEY(vehicle_id)
  );


CREATE TABLE IF NOT EXISTS dataPoint(
  drv_name text,
  vehicle_id text,
  services_no int ,
  lineName text ,
  day int ,
  sequence timestamp ,
  latitude double,
  longitude double ,
  speed double ,
   PRIMARY KEY (vehicle_id , sequence )
    )WITH CLUSTERING ORDER BY (sequence DESC);

CREATE TABLE IF NOT EXISTS timeTable(
  station_name   text ,
  longitude double,
  latitude double,
  services_no int ,
  distance double ,
  time int,
  lineName text,
```

```
  PRIMARY KEY (( lineName , services_no ) , time)
  )WITH CLUSTERING ORDER BY (time DESC);

CREATE TABLE IF NOT EXISTS days_registered_counts
(drv_name text, date int,
total_count counter static,
daily_count counter,
  PRIMARY KEY (drv_name,date)
);

CREATE INDEX latitudeIndex ON dataPoint (latitude);
```

Question 5. [20 marks] Provide CQL3 statements to support each of the application write and update requests you specified in Question 1 above. Show the consistency level before each write and update statement.

--1. Write records to drivers table.

```
consistency  quorum ;

INSERT INTO drivers (drv_name, cur_pos, mobile, pwd, skill) VALUES('milan','Upper Hutt', 211111, 'mm77', {'Guliver'});

consistency  quorum ;

INSERT INTO drivers (drv_name, cur_pos, mobile, pwd, skill) VALUES('pavle','Upper Hutt', 213344, 'pm33', {'Ganz Mavag','Guliver'});

consistency  quorum ;

INSERT INTO drivers (drv_name, cur_pos, mobile, pwd, skill) VALUES('pondy','Wellington', 216677, 'pondy', {'Matangi', 'Kiwi Rail'});
```

--2. write records to timeTableForPassengers table.

```
consistency  quorum ;

INSERT INTO timeTableForPassengers (station_name, time ,lineName) VALUES('Wellington',  {0617, 0629} , 'Hutt Vale Line');

consistency  quorum ;

INSERT INTO timeTableForPassengers (station_name, time ,lineName) VALUES('Petone', {0620,0630}, 'Hutt Vale Line');

consistency  quorum ;

INSERT INTO timeTableForPassengers (station_name, time ,lineName) VALUES('Woburn', {0700,0720}, 'Hutt Vale Line');

consistency  quorum ;

INSERT INTO timeTableForPassengers (station_name, time ,lineName) VALUES('Upper Hutt', {0805,0837}, 'Hutt Vale Line');
```

--3. write records to dispatchAllocation table.

```
consistency  quorum ;
```

```
INSERT INTO dispatchAllocation (station, vehicle, driver, services_no ,time ,lineName )
VALUES('Wellington', 'FA1122' ,  'milan' , 1, 0605 , 'Hutt Vale Line');

consistency  quorum ;

INSERT INTO dispatchAllocation (station, vehicle, driver, services_no ,time ,lineName)
VALUES('Wellington', 'FA4864' , 'pondy' ,2, 0617 , 'Hutt Vale Line');

consistency  quorum ;

INSERT INTO dispatchAllocation (station, vehicle, driver, services_no ,time ,lineName)
VALUES('Upper Hutt',  'FA4567' , 'pavle'  ,4,0700, 'Hutt Vale Line');

consistency  quorum ;

INSERT INTO dispatchAllocation (station, vehicle, driver, services_no ,time ,lineName)
VALUES('Upper Hutt',  'FP8899' , 'milan'  ,5,0705, 'Hutt Vale Line');


--4. write records to vehicles table.

consistency  quorum ;

INSERT INTO vehicles (vehicle_id, status, vehicle_type ) VALUES('FA1122', 'Upper Hutt', 'Matangi');

consistency  quorum ;

INSERT INTO vehicles (vehicle_id, status, vehicle_type) VALUES('FP8899', 'maintenance', 'Ganz
Mavag');

consistency  quorum ;

INSERT INTO vehicles (vehicle_id, status, vehicle_type) VALUES('FA4864', 'Wellington', 'Matangi');

consistency  quorum ;

INSERT INTO vehicles (vehicle_id, status, vehicle_type) VALUES('FA4567', 'Wellington', 'Matangi');


--5. write records to dataPoint table.

consistency  quorum ;

INSERT INTO dataPoint (drv_name,vehicle_id, services_no, lineName, day,
sequence,latitude,longitude,speed) VALUES('pondy','FA4567', 1,   'Hutt Vale
Line',20170325,1490475600000,-41.2262,174.77  ,29.1);

consistency  quorum ;

INSERT INTO dataPoint (drv_name,vehicle_id,
services_no,lineName,day,sequence,latitude,longitude,speed) VALUES('pondy','FA4567',1, 'Hutt Vale
Line',  20170325,  1490475610000,-41.2263 ,175,  70.1);

consistency  quorum ;
```

```
INSERT INTO dataPoint (drv_name,vehicle_id,
services_no,lineName,day,sequence,latitude,longitude,speed) VALUES('pondy','FA4567', 1, 'Hutt Vale Line',  20170325, 1490475620000,-41.2264, 175.07,  80.5);


--6. Write records to timeTable table.

consistency  quorum ;

INSERT INTO timeTable (station_name, longitude, latitude,services_no, distance, time ,lineName) VALUES('Wellington',174.7762, -41.2865, 1 ,0, 0605  , 'Hutt Vale Line');

consistency  quorum ;

INSERT INTO timeTable (station_name, longitude, latitude,services_no, distance,time ,lineName) VALUES('Petone',    174.8851, -41.227 , 1 , 8.3 ,0617, 'Hutt Vale Line');

consistency  quorum ;

INSERT INTO timeTable (station_name, longitude, latitude,services_no, distance, time ,lineName) VALUES('Woburn',  174.9081, -41.211, 1, 11.0,0620,     'Hutt Vale Line' );

consistency  quorum ;

INSERT INTO timeTable (station_name, longitude, latitude,services_no, distance, time ,lineName) VALUES('Waterloo',174.7762, -41.2092, 1 ,13.3, 0625  , 'Hutt Vale Line');

consistency  quorum ;

INSERT INTO timeTable (station_name, longitude, latitude,services_no, distance, time ,lineName) VALUES('Waikanae',174.7762, -40.8755, 5 ,62.8, 1025  , 'Waikanae Line');


--7.Update the password for driver 'pondy'.

consistency  quorum ;

UPDATE drivers SET pwd = 'newPwd' WHERE drv_name='pondy';


--8. Driver 'pondy' comes to Wellington station thus update his current_position to 'Wellington'.

consistency  quorum ;

UPDATE drivers SET cur_pos = 'Wellington' WHERE drv_name='pondy'   ;


--9. Driver 'pondy' is assigned to a service thus to update his current_position to ‘FA4567’.

consistency  quorum ;

UPDATE drivers SET cur_pos = 'FA4567' WHERE drv_name='pondy'  ;
```

--10.Driver 'pondy' deregisters from his work thus to update current_position to ‘not_available’.

```
consistency  quorum ;

UPDATE drivers SET cur_pos = 'not_available' WHERE drv_name='pondy'  ;
```

--11. Update the number of days per month for registered driver 'pondy'.

```
consistency  quorum ;

UPDATE days_registered_counts SET daily_count = daily_count + 1, total_count = total_count+ 1
WHERE drv_name = 'pondy' AND date = 20160321;
```

--12.Update the skill of driver 'pondy' from {'Matangi', 'Kiwi Rail'} to {'Matangi', 'Kiwi Rail','Guliver'}

```
consistency  quorum ;

UPDATE drivers SET skill = skill + {'Guliver'} WHERE drv_name='pondy'  ;
```

--13. Update the status of a vehicle.

```
consistency  quorum ;

update  vehicles  set  status =  'in_use'  where  vehicle_id  ='FA4567'  ;
```

--14. Update the current_position to destination 'Upper Hutt' of driver 'pondy'.

```
consistency  quorum ;

UPDATE drivers SET cur_pos = 'Upper Hutt' WHERE drv_name='pondy'  ;
```

--15. Update the status to destination 'Upper Hutt' of vehicle 'FA4567'.

```
consistency  quorum ;

UPDATE vehicles SET status = 'Upper Hutt' WHERE vehicle_id='FA4567'  ;
```

--16.Update the total distance travelled for vehicle ‘FA4567’ .

```
consistency  quorum ;

UPDATE  vehicles  set  distance_travelled =  10  where  vehicle_id  ='FA4567'  ;
```

--17.Update the total distance travelled for vehicle 'FA1122' .

```
consistency  quorum ;
```

```
UPDATE vehicles set distance_travelled = 20 where vehicle_id ='FA1122' ;
```

Question 6. [29 marks] Provide CQL3 statements to support each of the application read requests you specified in Question 2 above. Show the consistency level before each read statement. In your answer copy your CQL statement and the result produced by Cassandra from the screen.

USE WellingtonTranzMetro;

--1. Get the current password of driver 'pondy'. ;

consistency quorum ;

SELECT pwd, drv_name from drivers where drv_name = 'pondy' ;

```
Consistency level set to QUORUM.

 pwd    | drv_name
--------+----------
 newPwd |    pondy

(1 rows)
```

--2. Get the number of days per month for registered driver 'pondy'.

consistency one ;

select * from days_registered_counts where drv_name = 'pondy' ;

```
Consistency level set to ONE.

 drv_name | date     | total_count | daily_count
----------+----------+-------------+-------------
    pondy | 20160321 |           1 |           1

(1 rows)
```

--3. Publish a time table for passengers.

consistency one ;

SELECT * FROM timeTableForPassengers ;

```
Consistency level set to ONE.

 station_name | linename         | time
--------------+------------------+------------
   Wellington | Hutt Vale Line | {617, 629}
       Woburn | Hutt Vale Line | {700, 720}
       Petone | Hutt Vale Line | {620, 630}
   Upper Hutt | Hutt Vale Line | {805, 837}

(4 rows)
```

--4. Extract a list of departure stations with time in descending order

consistency one ;

SELECT * FROM dispatchAllocation ;

```
Consistency level set to ONE.

 station    | linename         | time | driver | services_no | vehicle
------------+------------------+------+--------+-------------+---------
 Upper Hutt | Hutt Vale Line |  705 |  milan |           5 |  FP8899
 Upper Hutt | Hutt Vale Line |  700 |  pavle |           4 |  FA4567
 Wellington | Hutt Vale Line |  617 |  pondy |           2 |  FA4864
 Wellington | Hutt Vale Line |  605 |  milan |           1 |  FA1122

(4 rows)
```

--5.Find the vehicle in 'Upper Hutt'.

consistency quorum ;

select * from vehicles where status='Upper Hutt' allow filtering ;

```
Consistency level set to QUORUM.

 vehicle_id | distance_travelled | status     | vehicle_type
------------+--------------------+------------+--------------
     FA4567 |                 10 | Upper Hutt |      Matangi
     FA1122 |                 20 | Upper Hutt |      Matangi
```

--6.Find the driver in 'Upper Hutt' with the skill 'Matangi'.

consistency quorum ;

select * from drivers where cur_pos='Upper Hutt' and skill CONTAINS 'Matangi' ALLOW FILTERING;

```
Consistency level set to QUORUM.

 drv_name | cur_pos     | mobile | pwd    | skill
----------+-------------+--------+--------+-------------------------------------
    pondy | Upper Hutt | 216677 | newPwd | {'Guliver', 'Kiwi Rail', 'Matangi'}

(1 rows)
```

--7.Select the last data point for a service on a day.

consistency one ;

select * from dataPoint limit 1;

```
Consistency level set to ONE.

 vehicle_id | sequence                        | day      | drv_name | latitude | linename       | longitude | services_no | speed
------------+---------------------------------+----------+----------+----------+----------------+-----------+-------------+-------
     FA4567 | 2017-03-25 21:00:20.000000+0000 | 20170325 |    pondy | -41.2264 | Hutt Vale Line |    175.07 |           1 |  80.5

(1 rows)
```

--8.Select all Data points for a service on a day in a time interval 10 seconds for vehicle 'FA4567' with driver 'pondy'

consistency one ;

select * from dataPoint;

```
Consistency level set to ONE.

 vehicle_id | sequence                        | day      | drv_name | latitude | linename       | longitude | services_no | speed
------------+---------------------------------+----------+----------+----------+----------------+-----------+-------------+-------
     FA4567 | 2017-03-25 21:00:20.000000+0000 | 20170325 |    pondy | -41.2264 | Hutt Vale Line |    175.07 |           1 |  80.5
     FA4567 | 2017-03-25 21:00:10.000000+0000 | 20170325 |    pondy | -41.2263 | Hutt Vale Line |       175 |           1 |  70.1
     FA4567 | 2017-03-25 21:00:00.000000+0000 | 20170325 |    pondy | -41.2262 | Hutt Vale Line |    174.77 |           1 |  29.1

(3 rows)
```

--9.Get the north direction neighbours for a given data point "-41.2262 IN services_no 1 and lineName 'Hutt Vale Line' "

consistency one ;

select * from timeTable where latitude > -41.2262 and services_no = 1 and lineName ='Hutt Vale Line'   order by time asc limit 1 ALLOW FILTERING ;

```
Consistency level set to ONE.

 linename       | services_no | time | distance | latitude | longitude | station_name
----------------+-------------+------+----------+----------+-----------+--------------
 Hutt Vale Line |           1 |  620 |       11 |  -41.211 |  174.9081 |       Woburn

(1 rows)
```

--10.Get the south direction neighbours for a given data point "-41.2262 IN services_no 1 and lineName 'Hutt Vale Line' "

consistency one ;

select * from timeTable where latitude < -41.2262 and services_no = 1 and lineName ='Hutt Vale Line' limit 1 allow filtering ;

```
Consistency level set to ONE.

 linename       | services_no | time | distance | latitude | longitude | station_name
----------------+-------------+------+----------+----------+-----------+--------------
 Hutt Vale Line |           1 |  617 |      8.3 |  -41.227 |  174.8851 |       Petone

(1 rows)
```