VECTOR LTD DATABASE DESIGN REPORT

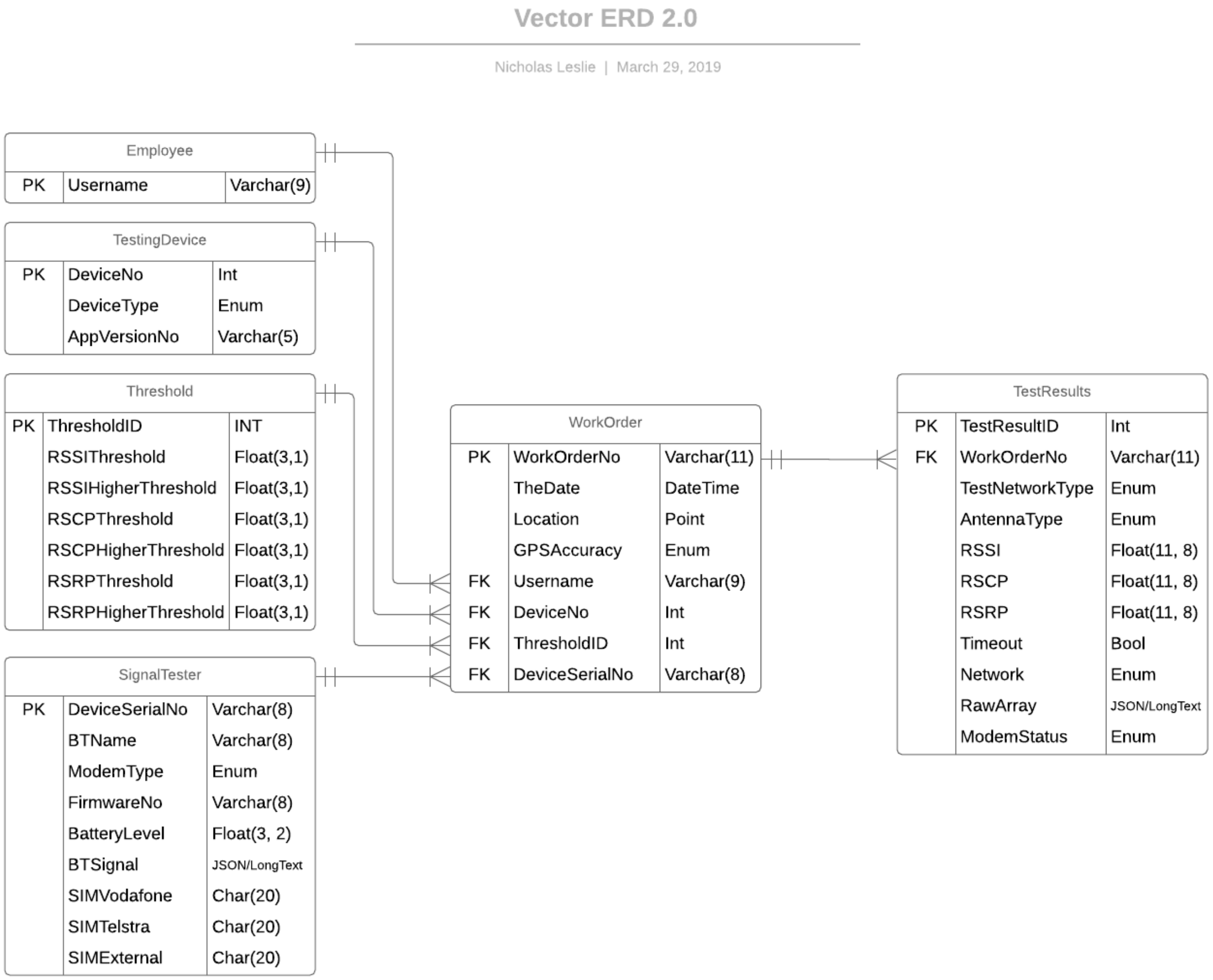
Nicholas Leslie

Due: 29/03/2019

Class: BCPR203 - Database Management Systems

Contents

Final ERD



Vector ERD 2.0

Nicholas Leslie | March 29, 2019

Employee

PK	Username	Varchar(9)

TestingDevice

PK	DeviceNo	Int
	DeviceType	Enum
	AppVersionNo	Varchar(5)

Threshold

PK	ThresholdID	INT
	RSSIThreshold	Float(3,1)
	RSSIHigherThreshold	Float(3,1)
	RSCPThreshold	Float(3,1)
	RSCPHigherThreshold	Float(3,1)
	RSRPThreshold	Float(3,1)
	RSRPHigherThreshold	Float(3,1)

SignalTester

PK	DeviceSerialNo	Varchar(8)
	BTName	Varchar(8)
	ModemType	Enum
	FirmwareNo	Varchar(8)
	BatteryLevel	Float(3, 2)
	BTSignal	JSON/LongText
	SIMVodafone	Char(20)
	SIMTelstra	Char(20)
	SIMExternal	Char(20)

WorkOrder

PK	WorkOrderNo	Varchar(11)
	TheDate	DateTime
	Location	Point
	GPSAccuracy	Enum
FK	Username	Varchar(9)
FK	DeviceNo	Int
FK	ThresholdID	Int
FK	DeviceSerialNo	Varchar(8)

TestResults

PK	TestResultID	Int
FK	WorkOrderNo	Varchar(11)
	TestNetworkType	Enum
	AntennaType	Enum
	RSSI	Float(11, 8)
	RSCP	Float(11, 8)
	RSRP	Float(11, 8)
	Timeout	Bool
	Network	Enum
	RawArray	JSON/LongText
	ModemStatus	Enum

Version Control

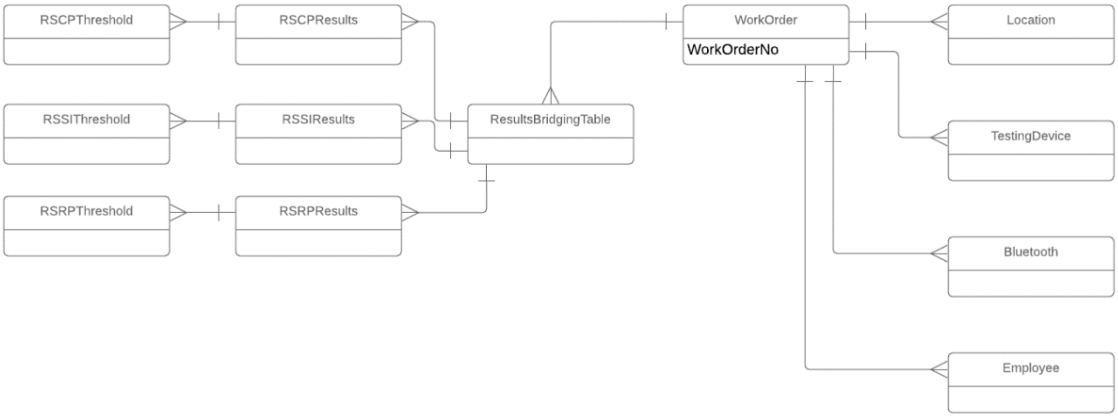
GitHub: https://github.com/ncpleslie/VectorLTD_SmartMeter_Database

Design Process:

The design process started with analysing the log files and attempting to find correlations between all the data within.

I parsed the data through Excel (Using Data from Text/CSV function), and having the data laid out I could see if I could infer any patterns. I created a rudimentary ERD.

This was a quick mock up of how I thought things would connect and relate to each other. This allowed me to make steps towards understanding the data.



My next step was Normalisation.

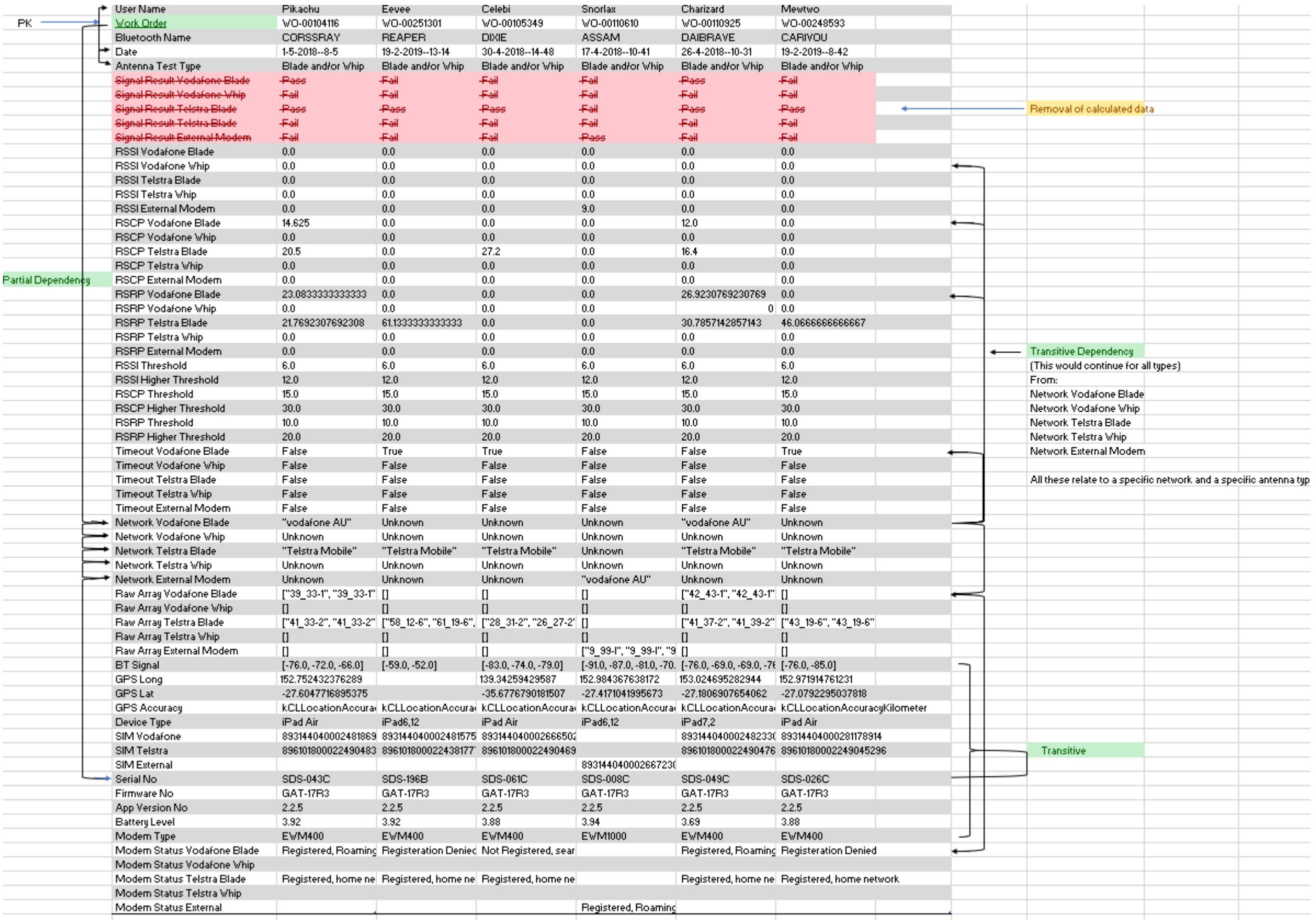
Normalisation

1NF

See Document (In Normalisation folder): VectorDataInExcel-1NF.xlsx

I removed calculated data from what we were given. I deduced that Signal Result Vodafone Blade, etc, where all calculated data. They were simply comparing the signal test numbers against the signal thresholds that related to those tests. Therefore, this data was removed.

Next step was to find relationships between the data. I documented this in the Excel document titled "VectorDataInExcel-1NF". This was a growing document that I changed and adapted as I went through the process. E.g. When I found out new information such as the Signal Testing and Testing Device (iPad) were two different things.



User Name	Pikachu	Eevee	Celebi	Snorlax	Charizard	Mewtwo
Work Order	WO-00104116	WO-00251301	WO-00105349	WO-00110610	WO-00110925	WO-00248593
Bluetooth Name	CORSSRAY	REAPER	DIXIE	ASSAM	DAIBRAVE	CARIVOU
Date	1-5-2018--8-5	19-2-2019--13-14	30-4-2018--14-48	17-4-2018--10-41	26-4-2018--10-31	19-2-2019--8-42
Antenna Test Type	Blade and/or Whip	Blade and/or Whip	Blade and/or Whip	Blade and/or Whip	Blade and/or Whip	Blade and/or Whip
~~Signal Result Vodafone Blade~~	~~Pass~~	~~Fail~~	~~Fail~~	~~Fail~~	~~Pass~~	~~Fail~~
~~Signal Result Vodafone Whip~~	~~Fail~~	~~Fail~~	~~Fail~~	~~Fail~~	~~Fail~~	~~Fail~~
~~Signal Result Telstra Blade~~	~~Pass~~	~~Pass~~	~~Pass~~	~~Fail~~	~~Pass~~	~~Pass~~
~~Signal Result Telstra Blade~~	~~Fail~~	~~Fail~~	~~Fail~~	~~Fail~~	~~Fail~~	~~Fail~~
~~Signal Result External Modem~~	~~Fail~~	~~Fail~~	~~Fail~~	~~Pass~~	~~Fail~~	~~Fail~~
RSSI Vodafone Blade	0.0	0.0	0.0	0.0	0.0	0.0
RSSI Vodafone Whip	0.0	0.0	0.0	0.0	0.0	0.0
RSSI Telstra Blade	0.0	0.0	0.0	0.0	0.0	0.0
RSSI Telstra Whip	0.0	0.0	0.0	0.0	0.0	0.0
RSSI External Modem	0.0	0.0	0.0	9.0	0.0	0.0
RSCP Vodafone Blade	14.625	0.0	0.0	0.0	12.0	0.0
RSCP Vodafone Whip	0.0	0.0	0.0	0.0	0.0	0.0
RSCP Telstra Blade	20.5	0.0	27.2	0.0	16.4	0.0
RSCP Telstra Whip	0.0	0.0	0.0	0.0	0.0	0.0
RSCP External Modem	0.0	0.0	0.0	0.0	0.0	0.0
RSRP Vodafone Blade	23.0833333333333	0.0	0.0	0.0	26.9230769230769	0.0
RSRP Vodafone Whip	0.0	0.0	0.0	0.0	0	0.0
RSRP Telstra Blade	21.7692307692308	61.1333333333333	0.0	0.0	30.7857142857143	46.0666666666667
RSRP Telstra Whip	0.0	0.0	0.0	0.0	0.0	0.0
RSRP External Modem	0.0	0.0	0.0	0.0	0.0	0.0
RSSI Threshold	6.0	6.0	6.0	6.0	6.0	6.0
RSSI Higher Threshold	12.0	12.0	12.0	12.0	12.0	12.0
RSCP Threshold	15.0	15.0	15.0	15.0	15.0	15.0
RSCP Higher Threshold	30.0	30.0	30.0	30.0	30.0	30.0
RSRP Threshold	10.0	10.0	10.0	10.0	10.0	10.0
RSRP Higher Threshold	20.0	20.0	20.0	20.0	20.0	20.0
Timeout Vodafone Blade	False	True	True	False	False	True
Timeout Vodafone Whip	False	False	False	False	False	False
Timeout Telstra Blade	False	False	False	False	False	False
Timeout Telstra Whip	False	False	False	False	False	False
Timeout External Modem	False	False	False	False	False	False
Network Vodafone Blade	"vodafone AU"	Unknown	Unknown	Unknown	"vodafone AU"	Unknown
Network Vodafone Whip	Unknown	Unknown	Unknown	Unknown	Unknown	Unknown
Network Telstra Blade	"Telstra Mobile"	"Telstra Mobile"	"Telstra Mobile"	Unknown	"Telstra Mobile"	"Telstra Mobile"
Network Telstra Whip	Unknown	Unknown	Unknown	Unknown	Unknown	Unknown
Network External Modem	Unknown	Unknown	Unknown	"vodafone AU"	Unknown	Unknown
Raw Array Vodafone Blade	["39_33-1", "39_33-1"	[]	[]	[]	["42_43-1", "42_43-1"	[]
Raw Array Vodafone Whip	[]	[]	[]	[]	[]	[]
Raw Array Telstra Blade	["41_33-2", "41_33-2"	["58_12-6", "61_19-6",	["28_31-2", "26_27-2'	[]	["41_37-2", "41_39-2"	["43_19-6", "43_19-6"
Raw Array Telstra Whip	[]	[]	[]	[]	[]	[]
Raw Array External Modem	[]	[]	[]	["9_99-I", "9_99-I", "9	[]	[]
BT Signal	[-76.0, -72.0, -66.0]	[-59.0, -52.0]	[-83.0, -74.0, -79.0]	[-91.0, -87.0, -81.0, -70.	[-76.0, -69.0, -69.0, -7(	[-76.0, -85.0]
GPS Long	152.752432376289		139.34259429587	152.984367638172	153.024695282944	152.971914761231
GPS Lat	-27.6047716895375		-35.6776790181507	-27.4171041995673	-27.1806907654062	-27.0792295037818
GPS Accuracy	kCLLocationAccura(	kCLLocationAccura(	kCLLocationAccura(	kCLLocationAccura(	kCLLocationAccura(	kCLLocationAccuracyKilometer
Device Type	iPad Air	iPad6,12	iPad Air	iPad6,12	iPad7,2	iPad Air
SIM Vodafone	893144040002481869	893144040002481575	893144040002666502		89314404000248233(	89314404000281178914
SIM Telstra	896101800022490483	896101800022438177'	896101800022490469		896101800022490476	89610180002249045296
SIM External				89314404000266723(		
Serial No	SDS-043C	SDS-196B	SDS-061C	SDS-008C	SDS-049C	SDS-026C
Firmware No	GAT-17R3	GAT-17R3	GAT-17R3	GAT-17R3	GAT-17R3	GAT-17R3
App Version No	2.2.5	2.2.5	2.2.5	2.2.5	2.2.5	2.2.5
Battery Level	3.92	3.92	3.88	3.94	3.69	3.88
Modem Type	EWM400	EWM400	EWM400	EWM1000	EWM400	EWM400
Modem Status Vodafone Blade	Registered, Roaming	Registeration Deniec	Not Registered, sear		Registered, Roaming	Registeration Denied
Modem Status Vodafone Whip						
Modem Status Telstra Blade	Registered, home ne	Registered, home ne	Registered, home ne		Registered, home ne	Registered, home network
Modem Status Telstra Whip						
Modem Status External				Registered, Roaming		

PK

Partial Dependency

Removal of calculated data

Transitive Dependency
(This would continue for all types)
From:
Network Vodafone Blade
Network Vodafone Whip
Network Telstra Blade
Network Telstra Whip
Network External Modem

All these relate to a specific network and a specific antenna typ

Transitive

I first noticed a unique identifier (WorkOrder) that could be used to determine everything and was unique to every log file. This was to become the primary key for my 1NF.

I could determine that WorkOrder was unique. It could be grouped with Username, date, bluetoothname, AntennaType.

The next grouping I found was each individual networks test. I originally clumped them all together so RSSI results, RSRP results and RSCP results would all contain the respective results from all the Network Providers. From the three results tables would be a threshold table the results tables would connect to. Each results table would connect up to a bridge table which in turn would connect up to the WorkOrder table. As I went through the process, I found that this was a redundant design and would be terrible. But one thing was consistent throughout the entire process. The WorkOrder table. This remained, but its attributes changed MANY times.

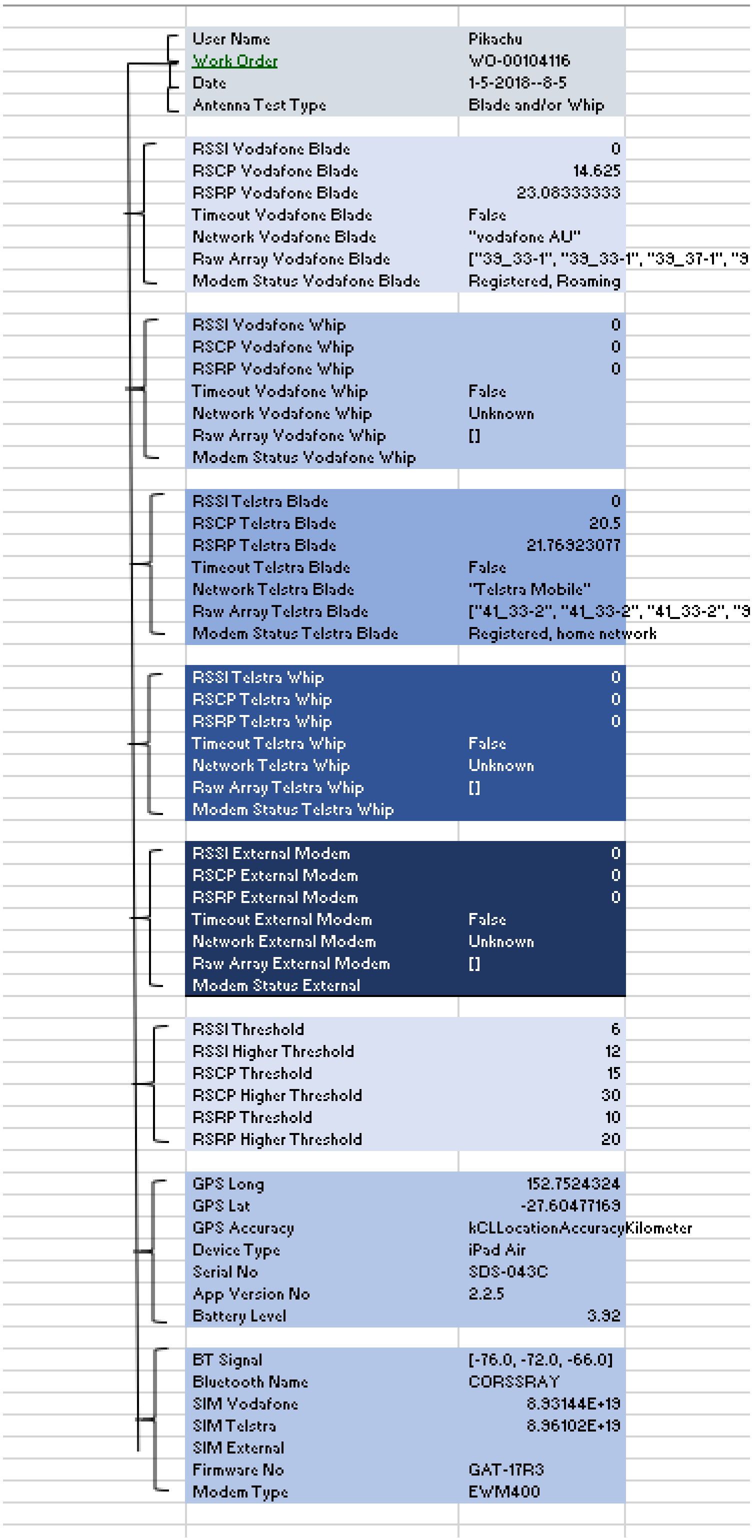
I had found how things could connect to the WorkOrder PK, and then I found how things that connected to the WorkOrder PK connected to other parts of the data (Transitive Dependencies). Please see the image for clarification.

2NF

See Document (In Normalisation folder): VectorDataInExcel-2NF.xlsx

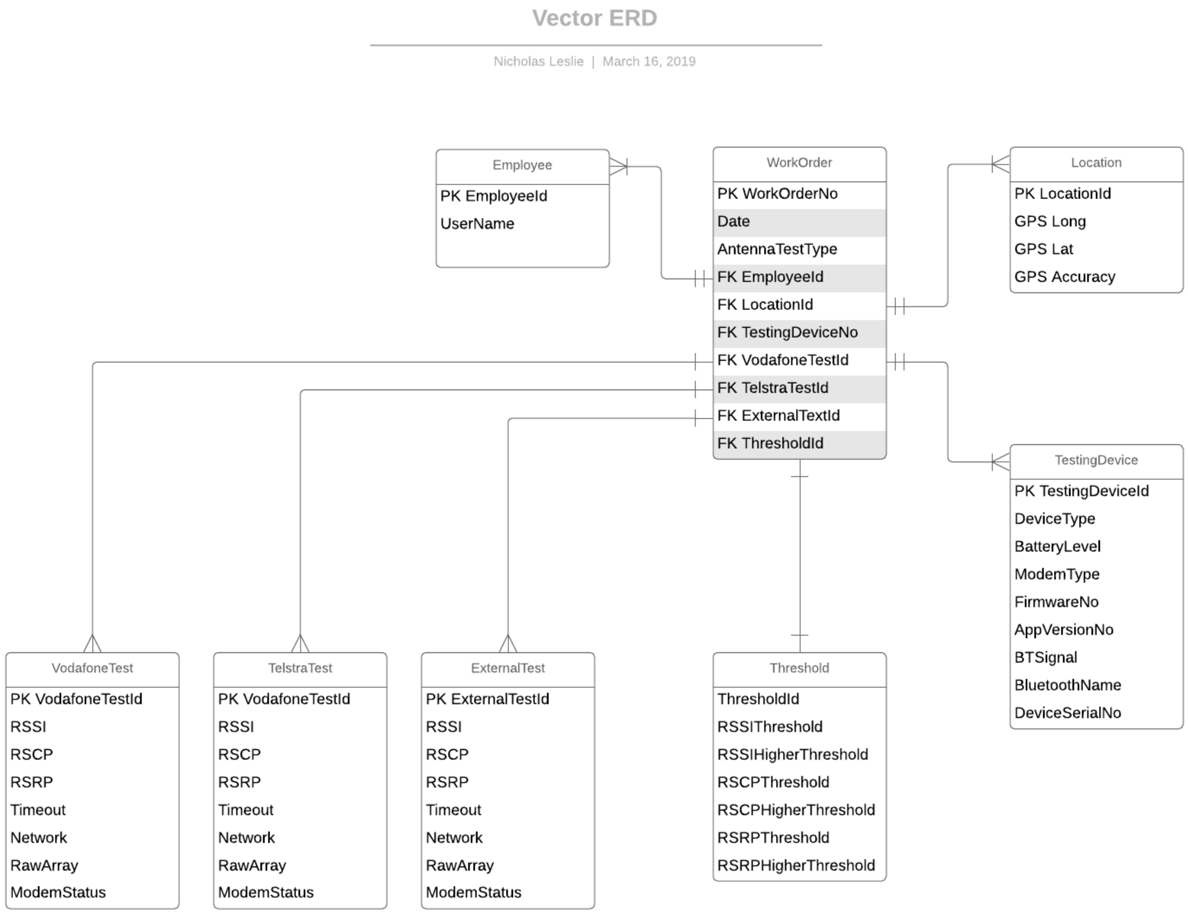
The 2NF is really where the "growing documentation" happens. Many changes have been made over the life of this project.

All the data was grouped together into how it would appear in the database.



User Name	Pikachu
Work Order	WO-00104116
Date	1-5-2018--8-5
Antenna Test Type	Blade and/or Whip
RSSI Vodafone Blade	0
RSCP Vodafone Blade	14.625
RSRP Vodafone Blade	23.08333333
Timeout Vodafone Blade	False
Network Vodafone Blade	"vodafone AU"
Raw Array Vodafone Blade	["39_33-1", "39_33-1", "39_37-1", "9
Modem Status Vodafone Blade	Registered, Roaming
RSSI Vodafone Whip	0
RSCP Vodafone Whip	0
RSRP Vodafone Whip	0
Timeout Vodafone Whip	False
Network Vodafone Whip	Unknown
Raw Array Vodafone Whip	[]
Modem Status Vodafone Whip	
RSSI Telstra Blade	0
RSCP Telstra Blade	20.5
RSRP Telstra Blade	21.76923077
Timeout Telstra Blade	False
Network Telstra Blade	"Telstra Mobile"
Raw Array Telstra Blade	["41_33-2", "41_33-2", "41_33-2", "9
Modem Status Telstra Blade	Registered, home network
RSSI Telstra Whip	0
RSCP Telstra Whip	0
RSRP Telstra Whip	0
Timeout Telstra Whip	False
Network Telstra Whip	Unknown
Raw Array Telstra Whip	[]
Modem Status Telstra Whip	
RSSI External Modem	0
RSCP External Modem	0
RSRP External Modem	0
Timeout External Modem	False
Network External Modem	Unknown
Raw Array External Modem	[]
Modem Status External	
RSSI Threshold	6
RSSI Higher Threshold	12
RSCP Threshold	15
RSCP Higher Threshold	30
RSRP Threshold	10
RSRP Higher Threshold	20
GPS Long	152.7524324
GPS Lat	-27.60477169
GPS Accuracy	kCLLocationAccuracyKilometer
Device Type	iPad Air
Serial No	SDS-043C
App Version No	2.2.5
Battery Level	3.92
BT Signal	[-76.0, -72.0, -66.0]
Bluetooth Name	CORSSRAY
SIM Vodafone	8.93144E+19
SIM Telstra	8.96102E+19
SIM External	
Firmware No	GAT-17R3
Modem Type	EWM400

This design is a far cry from what I initially thought the system would turn out to be like. This reinforces the importance of proper planning and design. I mocked up a quick ERD to see how this would play out.



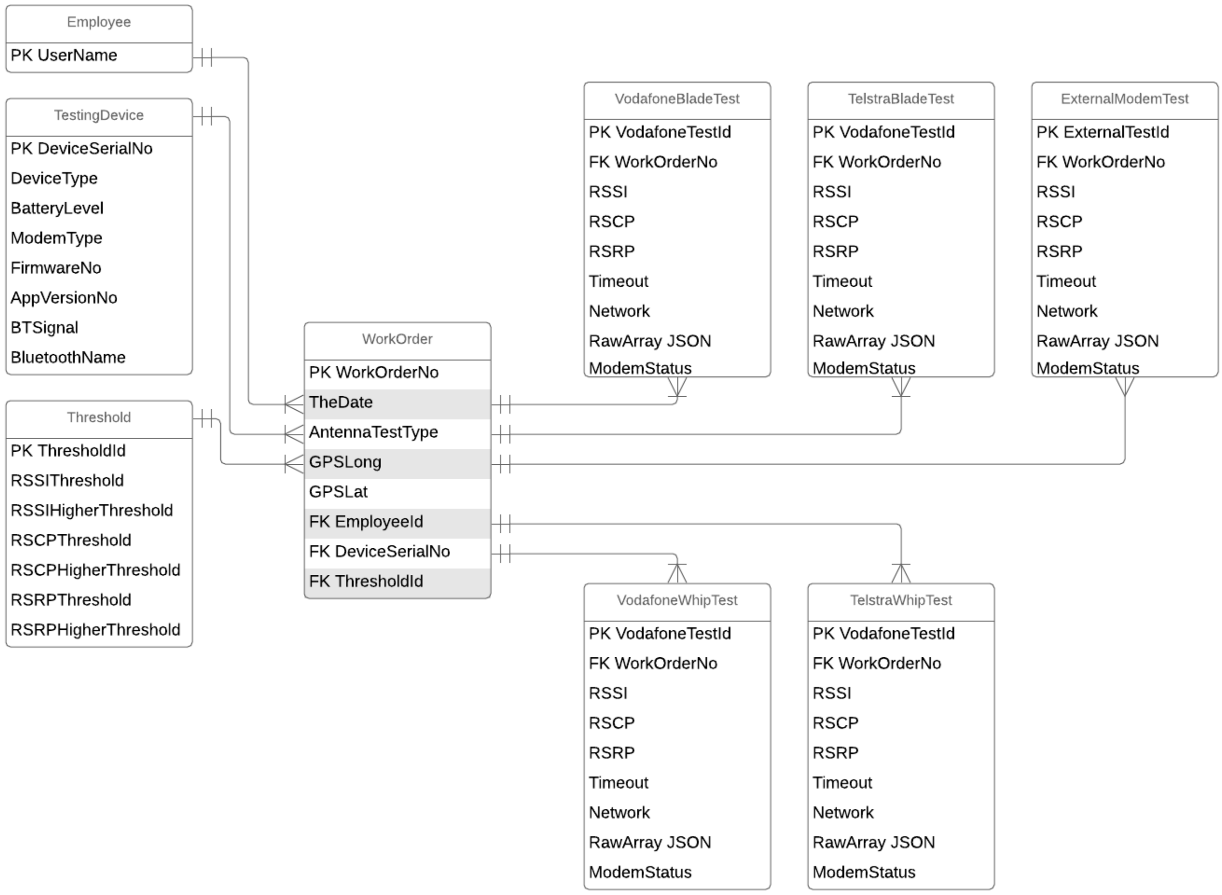
This was starting to come closer to how the final system would finally turn out. But it was still far from the end. I took the time, after creating this ERD, to plug in into MySQL. This is where I hit a roadblock. How would I connect the testing data up with the WorkOrder table? I couldn’t link the testing data up to the WorkOrder with Primary Keys. Keeping track of all the PKs would be a nightmare and add more confusion to the already complicated design. This led into 3NF.

3NF

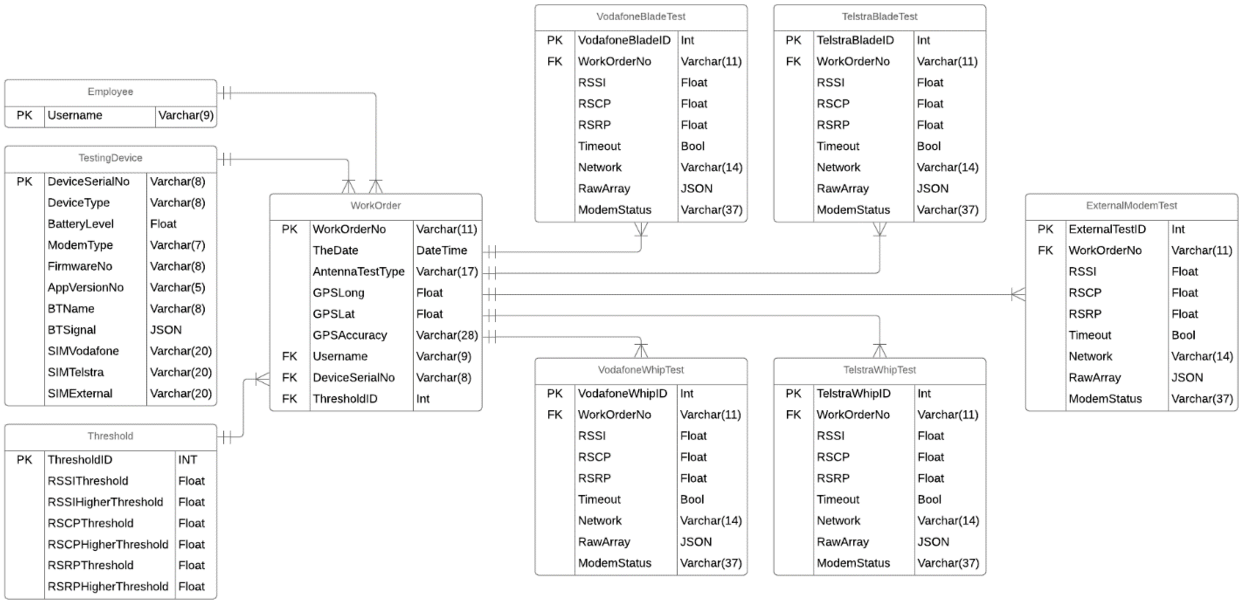
See Document (In same folder): VectorDataInExcel-3NF.xlsx

I then switched to the testing data referencing the WorkOrder (By the WorkOrderNo PK).

Breaking up the testing data based on each network, and each test antenna type (Blade or Whip), I was able to get a much clearer design. Also, merging the Location table into the WorkOrder table. My idea with the location table was that it would be able abstract data from the important tables that didn't need to be referenced. But, I later found that to be pointless.



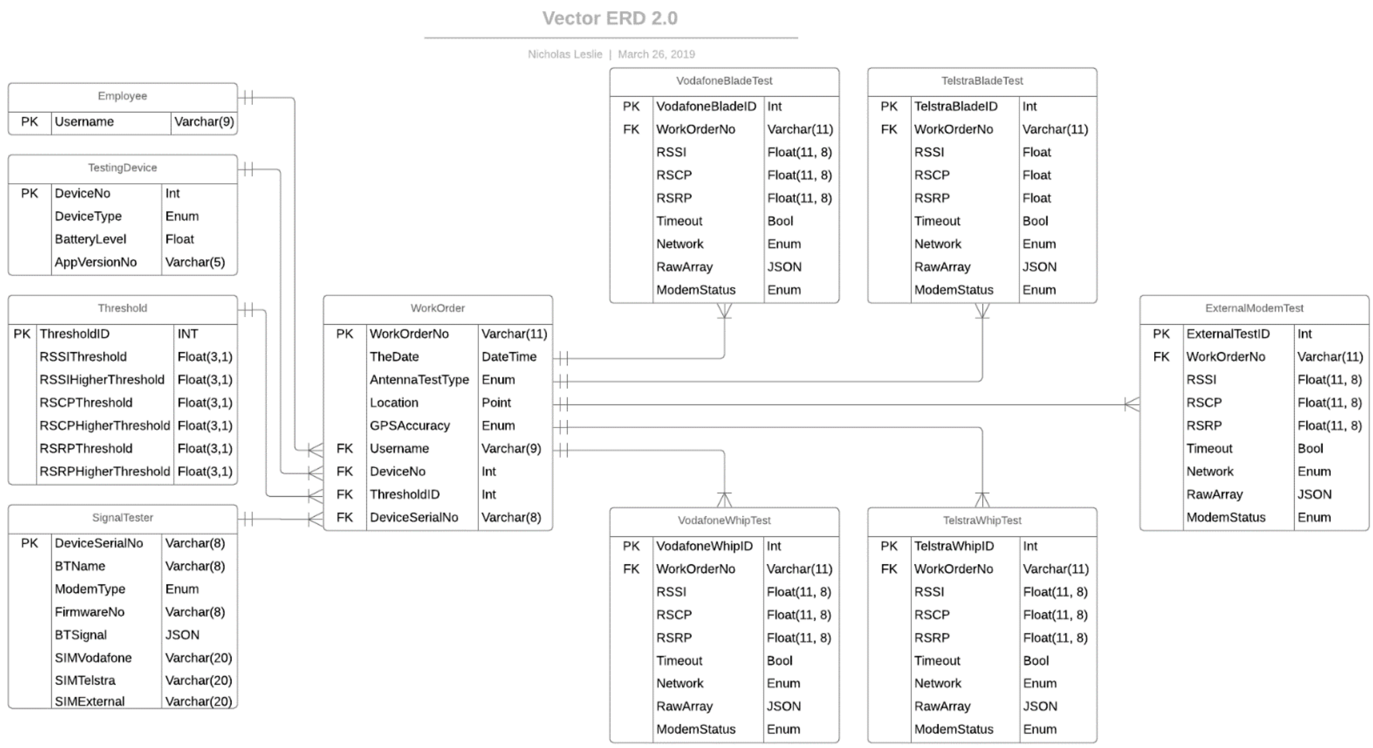
I also added cardinality and better relationships with proper Crows feet notation (Unlike other ERD that were mostly used just for reference). I then produced an ERD with datatypes to help with the construction of the database.



After Normalisation

Many changes were made after 3NF and the final form. Most were minor. Things like changing datatypes. Network names (and many others. See final ERD for reference) were changed from Varchar to Enum (Only two Networks), change GPS data (Originally Float) to Point, Raw Array data was changed from Varchar to Longtext (Was unable to get the JSON datatype to work. MySQL would not recognise it).

As I would find new information, such as finding out that the iPad and Signal Tester were two different devices (Originally thought it was all just part of the app), I could break things up further, and add new tables to reflect this change. At first I thought giving the Signal Tester the PK of BluetoothName would work (Its unique across all the data logs), but upon asking the employee from Vector, he states that the Serial Number relates to the Signal Tester. So, now I removed the Serial Number from the Testing Device (iPad), gave the Testing Device a unique PK, and then gave the Signal Tester the SerialNo as its PK.



Vector ERD 2.0

Nicholas Leslie | March 26, 2019

Employee		
PK	Username	Varchar(9)

TestingDevice		
PK	DeviceNo	Int
	DeviceType	Enum
	BatteryLevel	Float
	AppVersionNo	Varchar(5)

Threshold		
PK	ThresholdID	INT
	RSSIThreshold	Float(3,1)
	RSSIHigherThreshold	Float(3,1)
	RSCPThreshold	Float(3,1)
	RSCPHigherThreshold	Float(3,1)
	RSRPThreshold	Float(3,1)
	RSRPHigherThreshold	Float(3,1)

SignalTester		
PK	DeviceSerialNo	Varchar(8)
	BTName	Varchar(8)
	ModemType	Enum
	FirmwareNo	Varchar(8)
	BTSignal	JSON
	SIMVodafone	Varchar(20)
	SIMTelstra	Varchar(20)
	SIMExternal	Varchar(20)

WorkOrder		
PK	WorkOrderNo	Varchar(11)
	TheDate	DateTime
	AntennaTestType	Enum
	Location	Point
	GPSAccuracy	Enum
FK	Username	Varchar(9)
FK	DeviceNo	Int
FK	ThresholdID	Int
FK	DeviceSerialNo	Varchar(8)

VodafoneBladeTest		
PK	VodafoneBladeID	Int
FK	WorkOrderNo	Varchar(11)
	RSSI	Float(11, 8)
	RSCP	Float(11, 8)
	RSRP	Float(11, 8)
	Timeout	Bool
	Network	Enum
	RawArray	JSON
	ModemStatus	Enum

TelstraBladeTest		
PK	TelstraBladeID	Int
FK	WorkOrderNo	Varchar(11)
	RSSI	Float
	RSCP	Float
	RSRP	Float
	Timeout	Bool
	Network	Enum
	RawArray	JSON
	ModemStatus	Enum

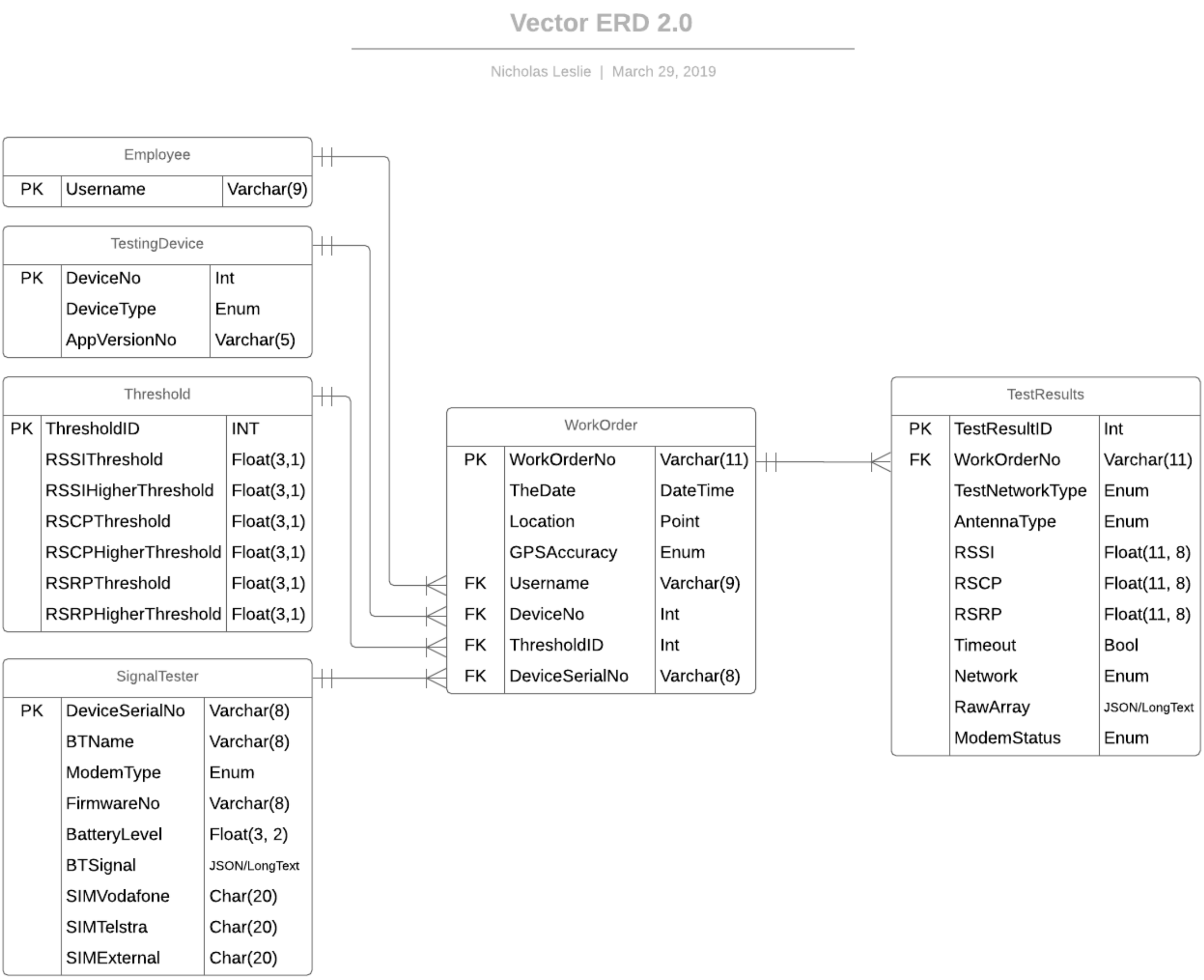
ExternalModemTest		
PK	ExternalTestID	Int
FK	WorkOrderNo	Varchar(11)
	RSSI	Float(11, 8)
	RSCP	Float(11, 8)
	RSRP	Float(11, 8)
	Timeout	Bool
	Network	Enum
	RawArray	JSON
	ModemStatus	Enum

VodafoneWhipTest		
PK	VodafoneWhipID	Int
FK	WorkOrderNo	Varchar(11)
	RSSI	Float(11, 8)
	RSCP	Float(11, 8)
	RSRP	Float(11, 8)
	Timeout	Bool
	Network	Enum
	RawArray	JSON
	ModemStatus	Enum

TelstraWhipTest		
PK	TelstraWhipID	Int
FK	WorkOrderNo	Varchar(11)
	RSSI	Float(11, 8)
	RSCP	Float(11, 8)
	RSRP	Float(11, 8)
	Timeout	Bool
	Network	Enum
	RawArray	JSON
	ModemStatus	Enum

As I started building the database, I noticed I was repeating tables with the same properties. One of the fundamentals of programming is not repeating yourself. I changed all the Test Results tables into one table. And I could differentiate between them with an Enum (Declaring the Network name and the Antenna type). Reducing the 5 test results tables into 1 table.

Final ERD



Vector ERD 2.0

Nicholas Leslie | March 29, 2019

Employee

PK	Username	Varchar(9)

TestingDevice

PK	DeviceNo	Int
	DeviceType	Enum
	AppVersionNo	Varchar(5)

Threshold

PK	ThresholdID	INT
	RSSIThreshold	Float(3,1)
	RSSIHigherThreshold	Float(3,1)
	RSCPThreshold	Float(3,1)
	RSCPHigherThreshold	Float(3,1)
	RSRPThreshold	Float(3,1)
	RSRPHigherThreshold	Float(3,1)

SignalTester

PK	DeviceSerialNo	Varchar(8)
	BTName	Varchar(8)
	ModemType	Enum
	FirmwareNo	Varchar(8)
	BatteryLevel	Float(3, 2)
	BTSignal	JSON/LongText
	SIMVodafone	Char(20)
	SIMTelstra	Char(20)
	SIMExternal	Char(20)

WorkOrder

PK	WorkOrderNo	Varchar(11)
	TheDate	DateTime
	Location	Point
	GPSAccuracy	Enum
FK	Username	Varchar(9)
FK	DeviceNo	Int
FK	ThresholdID	Int
FK	DeviceSerialNo	Varchar(8)

TestResults

PK	TestResultID	Int
FK	WorkOrderNo	Varchar(11)
	TestNetworkType	Enum
	AntennaType	Enum
	RSSI	Float(11, 8)
	RSCP	Float(11, 8)
	RSRP	Float(11, 8)
	Timeout	Bool
	Network	Enum
	RawArray	JSON/LongText
	ModemStatus	Enum

Entities

WorkOrder

This is a table that links all the testing data (Vodafone/Telstra signal results, etc) with the data that is potentially reused (Employees, testing devices like iPads, signal tester, and the minimum thresholds need to pass the test).

WorkOrderNo: Primary Key Varchar(11)

Each job consisted of a unique identifier referred to as Work Order (e.g.: “WO-00251301”). Due to it being unique for every job it became the primary key for the WorkOrder table. Referred to as “WorkOrderNo”.

TheDate: DateTime

The date was stored in the table under the name “theDate”. Date is a reserved name so could not be used. It should be entered with a date formatter to ensure it becomes a DateTime entity.

Example:

```
DATE_FORMAT(STR_TO_DATE("1-5-2018--8-5", "%d-%m-%Y--%H-%i"),'%Y-%m-%d %H:%i')
```

Location: Point

Combining GPS Lat and GPS Long into one data point. Datatype is Point. Point datatype is used in case the need to reference other geolocational data

GPSAccuracy: Enum

Denotes the accuracy. Enum is used due to references that GPS accuracy could go to “Metre” accuracy (In log files, accuracy is only to the Kilometre).

Username: Foreign Key Varchar(9)

Foreign Key reference to the Employee table

DeviceNo: Foreign Key Int

Foreign Key reference to the TestingDevice table

ThresholdID: Foreign Key Int

Foreign Key reference to the Threshold table

DeviceSerialNo: Foreign Key Varchar(8)

Foreign Key reference to the SignalTester table

TestingDevice

The table for the iPad that connects via Bluetooth to the SignalTester

DeviceNo: Primary Key Int

Used to link the testing device (iPad) up to the WorkOrder table.

DeviceType: Enum

This is a list of different iPads. I used Enum due to if you are wanting to search by iPad type, it would be quicker than search by varchar. Enum is used to ensure data validation.

AppVersionNo: Varchar(5)

The version the app is on. Used varchar, even though its all number, because it has multiple decimals.

SignalTester

This relates to information about the Signal Testing box

DeviceSerialNo: Primary Key Varchar(8)

The serial number is unique too all signal testers, so is used as a primary key, and links up with the WorkOrder table

BTName: Varchar(8)

Name the TestingDevice (iPad) connects up to (In real life)

ModemType: Enum

Only two modems are used but more could be added. Enum is used to ensure data integrity

FirmwareNo: Varchar(8)

A string of letters and numbers. Consistent length across all log files.

BatteryLevel: Float(3,2)

Even though the data is with a redundant "3" in the front (Example: 3.98), I used float in case the "3" had an importance I couldn't infer. The battery percentage is probably what is after the decimal.

BTSignal: JSON/LONGTEXT

The signal received back from the SignalTester (to the iPad). Comes back in array format. Longtext is used, but JSON would be preferred. JSON would not work on my version of MySQL. Longtext has similar functionality to JSON

SIMVodafone + SIMTelstra + SIMExternal: Char(20)

This is a large number. I used char instead of BIGINT because, other than no math or calculations being performed on these numbers, is that the numbers are above the limit of 64bit unsigned numbers of "18,446,744,073,709,551,615".

Other number data types were considered but with the possibility of truncation/rounding, I didn't want to risk the data changing. The SIM numbers would be worthless without exact data integrity.

Threshold

All threshold data repeats throughout the log files. Data is only entered into this table once.

Float(3,1) is used for all attributes in this table (Apart from the Primary Key. It uses INT)

Links up to the workOrder table.

Employee

This table could be expanded if Vector need to add more data relating to employees (Name, branch, etc)

Username: Primary Key Varchar(9)

Used as the primary key due to username being unique throughout all log files.

TestResults

This table relates to all the test results: Vodafone Blade, Vodafone Whip, Telstra Blade, Telstra Whip, External Modem. Also, all the results they each receive: RSSI, RSCP, RSRP, Raw Array. Each test can be determined by the enum's stating the Antenna type and Network Name.

TestResultID: Primary Key Int

Primary key for each testResult table

WorkOrderNo: Foreign Key Varchar(11)

Foreign Key reference to WorkOrder table

TestNetworkType: Enum

An enum for each network, plus to determine if using an external modem. This is reference to the desired network to connect to and each individual network's test

Values will be Telstra, Vodafone, or External Modem.

AntennaType: Enum

Used to denote if its testing for with Whip or Blade. All log files show that they only test with Blade. Enum is used with the intention that Whip testing could happen.

RSSI + RSCP = RSRP: Float(11,8)

Float was used, over decimal, due to the double-precision decimal offers isn't important when we are comparing the results (Which could have up to 8 decimal places) to a number that only goes to 1 decimal place. So, exact values aren't important and the rounding that could happen shouldn't affect the results. I have set the precision to (11, 8), so the option is there if Vector determine if they need the accuracy. Otherwise, I would recommend altering this if accuracy isn't important.

Timeout: Bool

If test times out. Bool due to all the data being True or False

Network: Enum

The network name that the test receives back from the carrier. Only "vodafone AU", "Unknown", "Telstra Mobile".

RawArray: JSON/LONGTEXT

Originally meant to be JSON but was unable to get it to work on my version of MySQL. MySQL documentation states that it is similar to LONGTEXT, so I have set to LONGTEXT. JSON could be selected if needed.

ModemStatus: Enum

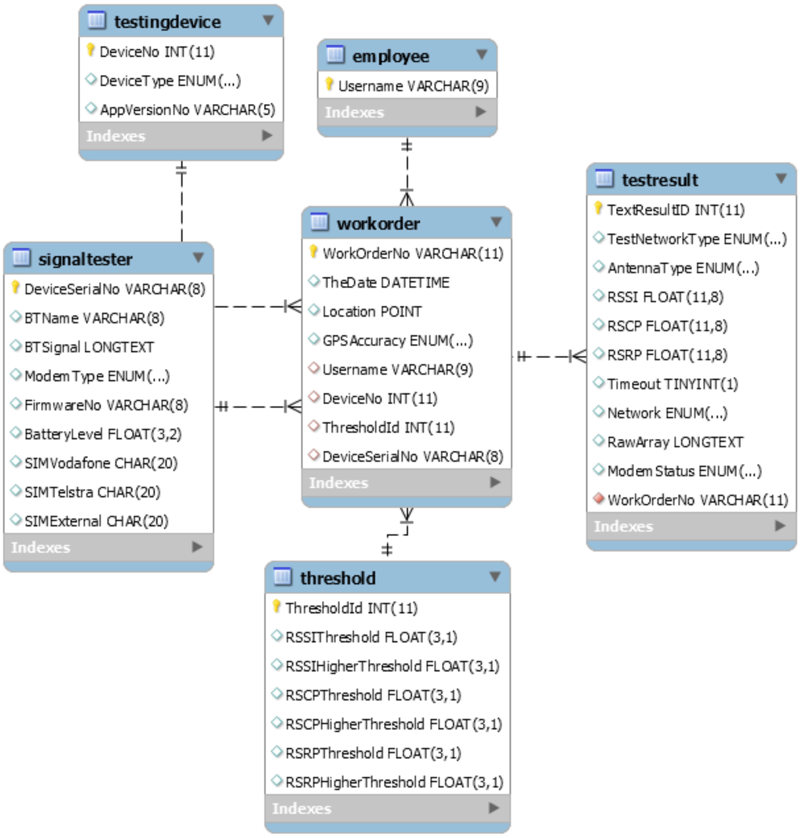
Status that is determined from the results of the test: "Registered, Roaming", "Registered, home network", "Not Registered, searching for network", "Registeration Denied". Data is copied exactly as seen in logs, including all spelling to ensure cohesiveness with all data.

Testing

Comparing ERD to reverse engineering

As you can see, the ERD diagram I created and the final ERD of the database that I created are the same. This means that the database was created correctly and to specifications.

Reverse Engineered



testingdevice
- DeviceNo INT(11)
- DeviceType ENUM(...)
- AppVersionNo VARCHAR(5)
- Indexes

employee
- Username VARCHAR(9)
- Indexes

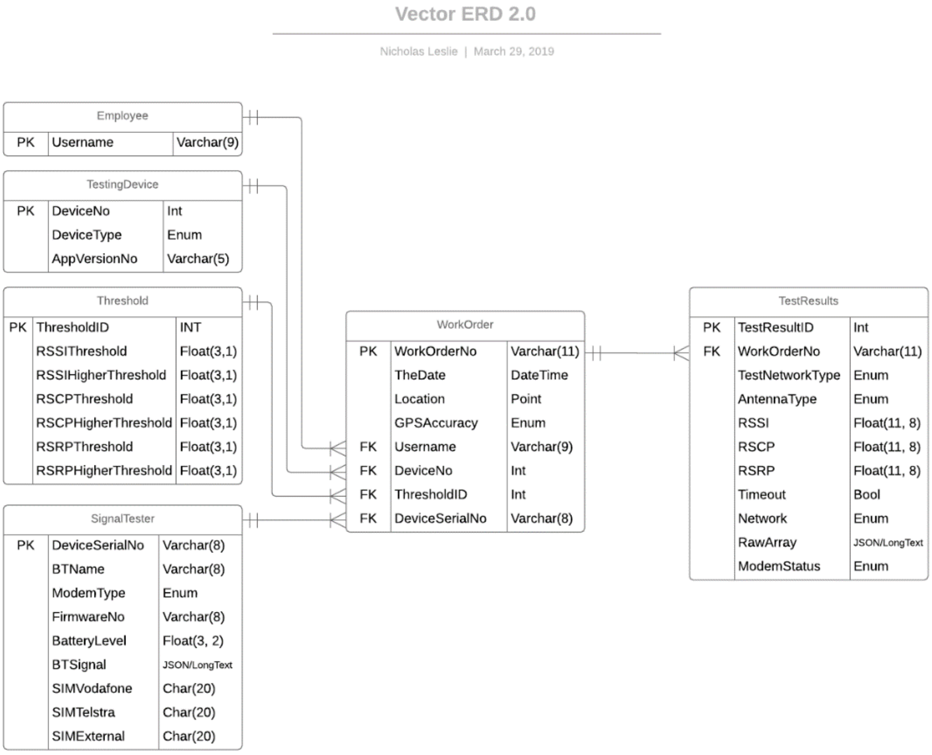
testresult
- TextResultID INT(11)
- TestNetworkType ENUM(...)
- AntennaType ENUM(...)
- RSSI FLOAT(11,8)
- RSCP FLOAT(11,8)
- RSRP FLOAT(11,8)
- Timeout TINYINT(1)
- Network ENUM(...)
- RawArray LONGTEXT
- Modem Status ENUM(...)
- WorkOrderNo VARCHAR(11)
- Indexes

workorder
- WorkOrderNo VARCHAR(11)
- TheDate DATETIME
- Location POINT
- GPSAccuracy ENUM(...)
- Username VARCHAR(9)
- DeviceNo INT(11)
- ThresholdId INT(11)
- DeviceSerialNo VARCHAR(8)
- Indexes

signaltester
- DeviceSerialNo VARCHAR(8)
- BTName VARCHAR(8)
- BTSignal LONGTEXT
- Modem Type ENUM(...)
- FirmwareNo VARCHAR(8)
- BatteryLevel FLOAT(3,2)
- SIMVodafone CHAR(20)
- SIMTelstra CHAR(20)
- SIMExternal CHAR(20)
- Indexes

threshold
- ThresholdId INT(11)
- RSSIThreshold FLOAT(3,1)
- RSSIHigherThreshold FLOAT(3,1)
- RSCPThreshold FLOAT(3,1)
- RSCPHigherThreshold FLOAT(3,1)
- RSRPThreshold FLOAT(3,1)
- RSRPHigherThreshold FLOAT(3,1)
- Indexes

Final ERD



Vector ERD 2.0

Nicholas Leslie | March 29, 2019

Employee

PK	Username	Varchar(9)

TestingDevice

PK	DeviceNo	Int
	DeviceType	Enum
	AppVersionNo	Varchar(5)

Threshold

PK	ThresholdID	INT
	RSSIThreshold	Float(3,1)
	RSSIHigherThreshold	Float(3,1)
	RSCPThreshold	Float(3,1)
	RSCPHigherThreshold	Float(3,1)
	RSRPThreshold	Float(3,1)
	RSRPHigherThreshold	Float(3,1)

SignalTester

PK	DeviceSerialNo	Varchar(8)
	BTName	Varchar(8)
	ModemType	Enum
	FirmwareNo	Varchar(8)
	BatteryLevel	Float(3, 2)
	BTSignal	JSON/LongText
	SIMVodafone	Char(20)
	SIMTelstra	Char(20)
	SIMExternal	Char(20)

WorkOrder

PK	WorkOrderNo	Varchar(11)
	TheDate	DateTime
	Location	Point
	GPSAccuracy	Enum
FK	Username	Varchar(9)
FK	DeviceNo	Int
FK	ThresholdID	Int
FK	DeviceSerialNo	Varchar(8)

TestResults

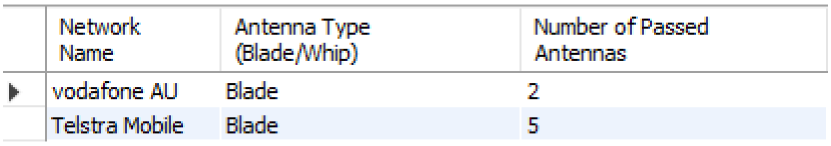
PK	TestResultID	Int
FK	WorkOrderNo	Varchar(11)
	TestNetworkType	Enum
	AntennaType	Enum
	RSSI	Float(11, 8)
	RSCP	Float(11, 8)
	RSRP	Float(11, 8)
	Timeout	Bool
	Network	Enum
	RawArray	JSON/LongText
	ModemStatus	Enum

Queries

1. How many blade and whip antennas have been passed in the signal test for each network provider? (If both blade and whip passed the test for the same Work Order Number only consider that as blade)

```
SET sql_mode=(SELECT REPLACE(@@sql_mode,'ONLY_FULL_GROUP_BY',''));

SELECT

  testResult.network as "Network Name",

  testResult.antennaType as "Antenna Type (Blade/Whip)",

   CASE

   WHEN testResult.antennaType = "Blade" AND testResult.antennaType = "Whip" THEN 0

   WHEN testResult.antennaType = 'Blade' THEN SUM(1)

   WHEN testResult.antennaType = 'Whip' THEN 0

   ELSE COUNT(testResult.antennaType) END as "Number of Passed Antennas"

FROM

   threshold,

   workorder,

   testResult

WHERE

          testResult.TestNetworkType NOT LIKE 'External Modem' -- Excluding External Modem since out of scope of
question(Neither Blade or whip)

      AND testResult.workOrderNo = workOrder.workOrderNo

      AND (testResult.RSSI >= threshold.RSSIThreshold

      OR testResult.RSCP >= threshold.RSCPThreshold

      OR testResult.RSRP >= threshold.RSRPThreshold)

GROUP BY testResult.network;
```

Result:



Network Name	Antenna Type (Blade/Whip)	Number of Passed Antennas
vodafone AU	Blade	2
Telstra Mobile	Blade	5

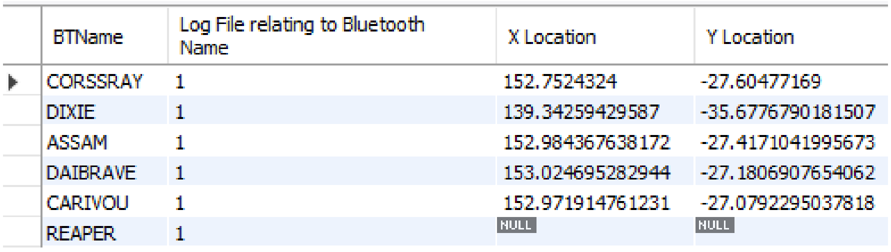
6. Count the number of signal log files per Bluetooth Names and print out each of the locations.

Assumptions:

Signal Log Files refers to the text documents received from Vector

```
SELECT
   signalTester.BTName,
   COUNT(workOrder.workOrderNo) AS 'Log File relating to Bluetooth Name',
   ST_X(workOrder.location) AS 'X Location',
   ST_Y(workOrder.location) AS 'Y Location'
FROM
   signalTester,
   workOrder
WHERE
   workOrder.DeviceSerialNo = SignalTester.DeviceSerialNo
GROUP BY workOrder.workOrderNo;
```

Result:

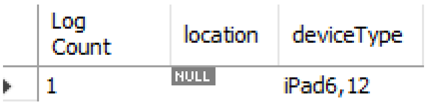


BTName	Log File relating to Bluetooth Name	X Location	Y Location
CORSSRAY	1	152.7524324	-27.60477169
DIXIE	1	139.34259429587	-35.6776790181507
ASSAM	1	152.984367638172	-27.4171041995673
DAIBRAVE	1	153.024695282944	-27.1806907654062
CARIVOU	1	152.971914761231	-27.0792295037818
REAPER	1	NULL	NULL

7. count the number of log files without GPS coordinates and sort them by iPad models

```
SELECT
   COUNT(workOrder.workOrderNo) as 'Log Count',
   workOrder.location,
   testingDevice.deviceType
FROM
   workOrder,
   testingDevice
WHERE
   testingDevice.DeviceNo = workOrder.DeviceNo
      AND ST_X(location) IS NULL
      AND ST_Y(location) IS NULL
GROUP BY workorder.workorderNo
ORDER BY testingDevice.deviceType;
```

Result:

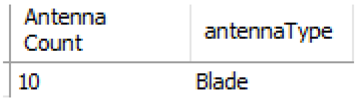


Log Count	location	deviceType
1	NULL	iPad6,12

8. count the number of whip antenna tested vs blade antenna tested

```
SELECT
   COUNT(testResult.antennaType) as 'Antenna Count',
   testResult.antennaType
FROM
   testResult,
   workOrder
WHERE
          testResult.antennaType IS NOT NULL
AND
          testResult.modemStatus IS NOT NULL
AND
   testResult.workOrderNo = workOrder.workOrderNo
GROUP BY testResult.antennaType;
```

Result:

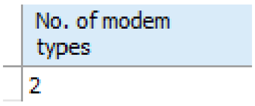


Antenna Count	antennaType
10	Blade

9. Count the number of different modem types used during the test

```
SELECT COUNT(DISTINCT modemType) as "No. of modem types" FROM SignalTester;
```

Result:

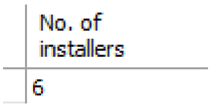


No. of modem types
2

10. Count the number of number of different installers

```
SELECT COUNT(DISTINCT username) as "No. of installers" FROM employee;
```

Result:



No. of installers
6