

2012

Bridgestone

Vishal Naidu

[BTECH MID SEMESTER REPORT]

The main aim of this project is designing a mobile application that helps Bridgestone's fleet technicians in completing Vehicle Inspection Reports efficiently.

Acknowledgements

This report is based on inputs from software developers, business analysts, and fleet technicians which include work conducted by Vishal Naidu, under the supervision of Dr. S. Manoharan. Valuable input, guidance, and support at various stages of the project were received from Philip Low (Business Analyst), Candy He (Business Analyst), and Robert A. Lee (Senior Business Analyst). I would like to thank everyone for their backing and direction in progressing my project further thus far.

Contents

Project Description.....	3
Company Description.....	3
The Problem.....	3
Objective	4
Dealing with Bridgestone Systems.....	4
Data Warehouse	5
The Vehicle repository	5
Choosing the Operating System.....	6
Google Android	6
Apple OS X.....	6
Windows 8	7
Web Services.....	8
My Web Service Assignment	9
SOAP or REST	10
The Bridgestone Field Trip	11
Tyre Replacement	11
Tyre Puncture.....	12
The Tyre Monitoring System Application	13
Workflow.....	13
Offline mode	13
Online Mode	14
Key Specifications	15
Interactive Vehicle Configuration Diagram.....	15
Synchronization.....	15
Validation	16
Vehicle Lookup Function	16
The Next Step.....	16

Chapter 1

Project Description

This report explains the work completed up till the end of semester 1 2012 for the final year BTech project at Bridgestone. The BTech degree is a 4 year honours course which requires students to complete a project in the final year for a company. This project will be carried out by me, Vishal Naidu. This project requires me to work at the Bridgestone headquarters for 8-10 hours a week.

Company Description

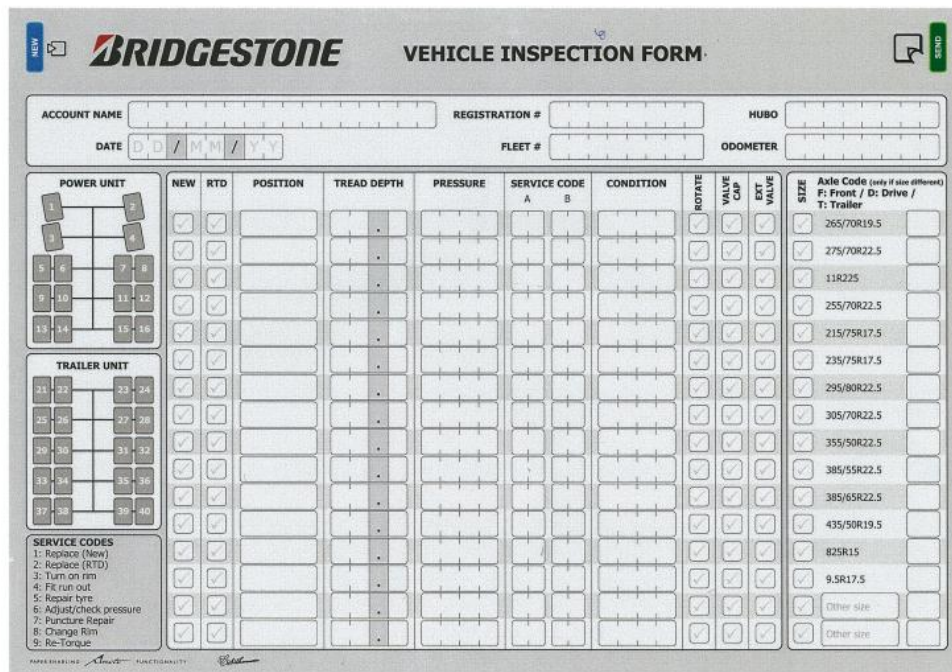
In 1931, Shojiro Ishibashi created Bridgestone Tire Company Ltd in Kyushu Japan. Today the Bridgestone Corporation is the world's leading tyre manufacturer, supplying quality tyres for cars, trucks, aircrafts and others. Bridgestone NZ Ltd was formed in September 1998, followed a merger between two world premier tyre companies: Bridgestone and Firestone. In New Zealand, Bridgestone, Firestone and Supercat branded tyres are available throughout the country at a number of retailers including; Firestone, Tony's Tyre Service and other Bridgestone stockists.



Figure 1: Bridgestone Logo

The Problem

Currently Bridgestone fleet technicians fill out Vehicle Inspection Reports for every Vehicle which takes a lot of time. Redundancy is also an issue as because information has to be noted multiple times before being keyed into the system. Also, there is no structure in the way they fill this form, and key information to the Company are being missed out.



BRIDGESTONE VEHICLE INSPECTION FORM

ACCOUNT NAME: _____ REGISTRATION #: _____ HUBO: _____
 DATE: DD / MM / YY FLEET #: _____ ODOMETER: _____

POWER UNIT

1	2
3	4
5	6
7	8
9	10
11	12
13	14
15	16

TRAILER UNIT

21	22	23	24
25	26	27	28
29	30	31	32
33	34	35	36
37	38	39	40

SERVICE CODES

- 1: Replace (New)
- 2: Replace (RTD)
- 3: Turn on rim
- 4: Fit run out
- 5: Repair tyre
- 6: Adjust/check pressure
- 7: Puncture Repair
- 8: Change Rim
- 9: Re-Torque

NEW	RTD	POSITION	TREAD DEPTH	PRESSURE	SERVICE CODE A B	CONDITION	ROTATE	VALVE CAP	EXT VALVE	SIZE	AXLE CODE (only if size different) F: Front / D: Drive / T: Trailer
☒	☒						☒	☒	☒	265/70R19.5	
☒	☒						☒	☒	☒	275/70R22.5	
☒	☒						☒	☒	☒	11R225	
☒	☒						☒	☒	☒	255/70R22.5	
☒	☒						☒	☒	☒	215/75R17.5	
☒	☒						☒	☒	☒	235/75R17.5	
☒	☒						☒	☒	☒	295/80R22.5	
☒	☒						☒	☒	☒	305/70R22.5	
☒	☒						☒	☒	☒	355/50R22.5	
☒	☒						☒	☒	☒	385/55R22.5	
☒	☒						☒	☒	☒	385/65R22.5	
☒	☒						☒	☒	☒	435/50R19.5	
☒	☒						☒	☒	☒	825R15	
☒	☒						☒	☒	☒	9.5R17.5	
☒	☒						☒	☒	☒	Other size	
☒	☒						☒	☒	☒	Other size	

Figure 2: Current Vehicle Inspection Report

Objective

The idea is to build a digital alternative of a Vehicle Inspection Report. A mobile application on a tablet PC would help quicken up the current workflow process of the filling out of VIR reports. It is designed also to reduce the redundancy of having the employees manually filling out each form and entering into the database.

Chapter 2

Dealing with Bridgestone Systems

The Tyre Monitoring System application is designed to help the fleet technicians at Bridgestone in completing Vehicle Inspection Reports quickly and efficiently. The application helps with redundancy issues, as you only have to enter the information once, and that information is stored in the database. That way if the fleet technician wants to access a particular piece of information, the application synchronizes with the database and all the information is populated on the application.

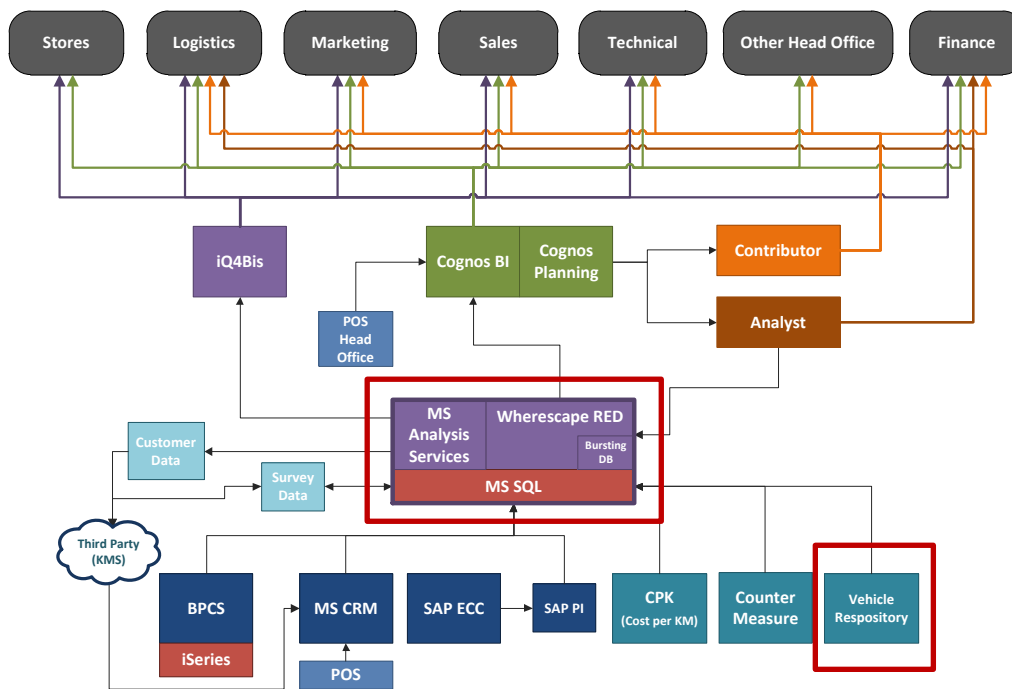


Figure 3: Bridgestone Systems

Data Warehouse

The systems that the Tyre Monitoring System Application primarily deal with, is the data warehouse itself. This is where majority of the data is held extending from several other systems like BPCS and CRM. The Wherescape RED tool helps build, manage and update the data warehouse. Other systems that are loosely going to be related to this application are the BPCS and the CRM systems. BPCS mainly deals with the transactional information of Bridgestone while CRM deals with the customer information. Both these systems information are stored in the data warehouse.

The Vehicle repository

Vehicle information comes from a vehicle repository database and if in fact the vehicle is registered, information such as vehicle configuration data and characteristics are returned. We access this information by using a web service which uses a registration number as a key, to look up and return information about the vehicle. Business rules are needed to be defined for refreshing the vehicle repository for those vehicles we haven't serviced to ensure WOF's (Warrant of Fitness) and COF's (Certificates of Fitness) are up to date.

Chapter 3

Choosing the Operating System

The most important feature for the tablet 'experience' is the Operating System. The Operating System is the heart of the tablet PC as it has all the features the device can support including the interface, application support and external device compatibilities. There are three main operating systems that the Tyre Monitoring System Application that can be developed on, Google Android, Apple IOS and Windows 8 OS.

Google Android

Google has made the developer tools and environment robust such that it is not a problem for developers to write apps for the Android tablet OS. The language is familiar to me, and I can easily develop on this tablet PC. The Android tablets have no problems in connecting to the New Zealand networks like Telecom and Vodafone, so 3G implementation on the application would not pose a problem. The Google Android isn't the easiest system to learn on if users haven't had any experience with it. It creates a sense of confusion and frustration in non-savvy users, in this case the fleet technicians.

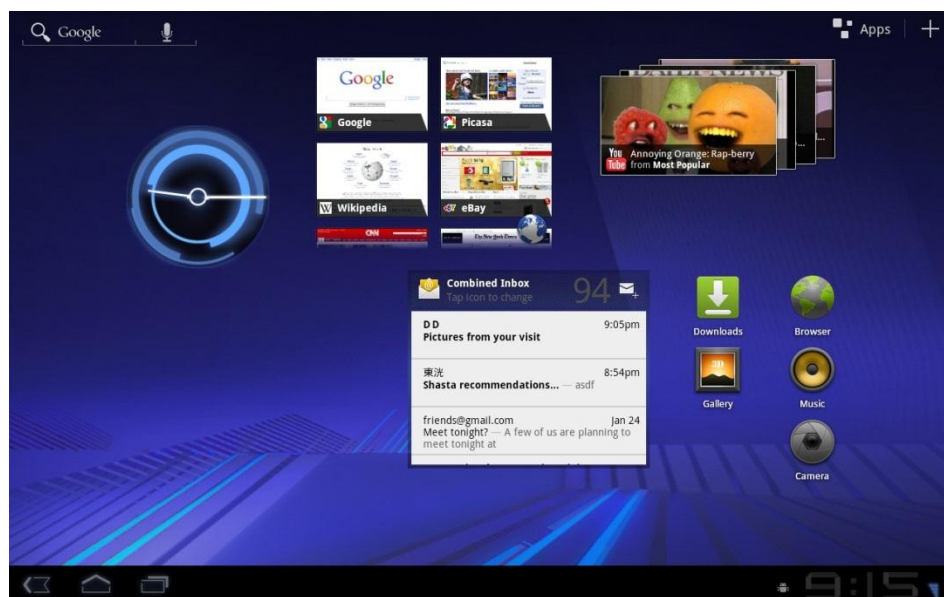


Figure 4: Google Android OS Screenshot

Apple OS X

The advantage that the Apple tablet, otherwise known as the iPad, has over the operating systems is that it's one of the easiest tablets to pick and use. Its interface isn't cloggy and since it's been in the market for the longest, it has the largest number of applications available to the market. If Bridgestone, were to expand on the iPad, pre-existing robust applications would be one of the positive things to have. One of the major disadvantages is that iPad cannot multitask. One of the main points in this project is to be able to sync two applications together, and if is the case one application has to run in the background while the other one is on. The iPad has the Vodafone

network as New Zealand's only authorized network carrier, but connections onto other networks aren't much of a problem. Additional technology would be needed to be added to the device if wanting to join other network providers. iPad does not have the multitasking unction which is a big downside when choosing the operating system best suited for my application. Another drawback that I would face when developing on the iPad is programming language itself. I have limited knowledge about the programming aspect of Objective C. Although it's not a big drawback, but I would like to finish the application of at the end of this year and learning the language would be a major constraint in achieving the project goal.



Figure 5: iPad Screenshot

Windows 8

When we are talking about Microsoft's Windows 8 tablet PC, one of the main benefits Microsoft is putting out there is that the OS will run on your desktop, laptop and tablet. This obvious advantage would mean that all your applications can work across all your hardware and you would not have to re-learn everything when you buy a new piece of equipment. This also fits into Bridgestone's current and future plans as the company's primary and only OS is Windows. Having the tablet being Windows too, would make things a whole lot easier, as when they need future programmers for expansion, whether it be on a desktop, laptop or the Windows 8 tablet I am creating, all Bridgestone require are Microsoft experienced programmers. Even though the Windows 8 tablets haven't been released yet, Microsoft NZ has confirmed that all Windows tablets would be able to connect to the giants of the network carrier industry, Telecom, Vodafone and 2 Degrees.



Figure 6: Windows 8 OS Screenshot

There are drawbacks; one being that there is also a learning curve for the Windows 8 OS. The Windows7 desktop still remains, but you still have to work with 'Metro' to some extent. The Metro environment will feature a new tile-based Start screen. Metro-style applications run in full-screen, and are able to share information between each other using "contracts". Metro-style apps are developed with the new Windows Runtime platform using various programming language's which includes the decision of designing it on: C++, Visual Basic, C#, or HTML/JavaScript.

Chapter 4

Web Services

A web service describes the method of integrating web-based applications using XML, SOAP, WDSL and UDDI open standards over an Internet protocol Backbone.

- XML is used to tag the data,
- SOAP is used to transfer the data,
- WSDL is used for describing the services available
- UDDI is used for listing what services are available.

It is clear to me that I would require to create and develop a web service in order to transfer data through and from the data warehouse. Before designing the web services for the mobile application itself, Bridgestone decided that I design a web service that interacts with the vehicle repository system and a simple interface.

My Web Service Assignment

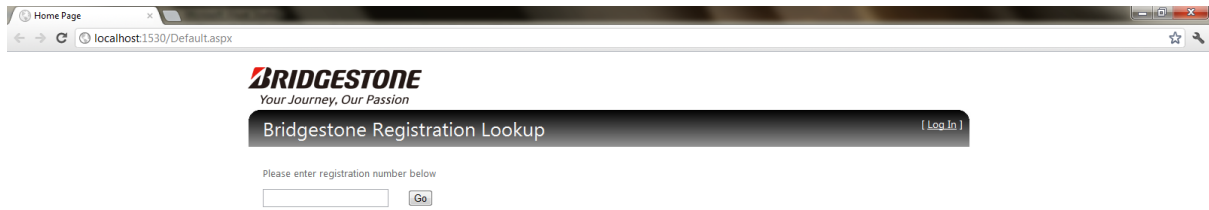


Figure 7: Test Interface

The front end part of this application was not really important, but was needed in order to display the returning values obtained from our web service. It was a simple plain interface which had just the one screen. It contained a Registration number text field and button. When the button was clicked, information such as Model Name, Make and latest odometer reading all appeared. When testing, if the correct information corresponding to that registration number appeared on the screen, the web service we have created was believed to be correct.

```
ServiceReferenceRego.RegoLookupClient c = new ServiceReferenceRego.RegoLookupClient();  
Label1.Text = Label1.Text + c.RegoSearch(regono);  
  
SqlConnection db = new SqlConnection("Data Source=baksql2 ;Initial Catalog=datawarehouse_UAT;Integrated Security=SSPI; persist security info=False;");  
try  
{  
    db.Open();  
}  
catch (Exception exc)  
{  
    Console.WriteLine("Problem opening connection..");  
}  
  
SqlCommand command1 = new SqlCommand("select plate, make, model, year_of_manufacture, expiry_date_of_last_successful_wof, latest_odometer_reading from dim_vehicle whe  
SqlDataReader myReader = command1.ExecuteReader();
```

Figure 8: Web Service Code Segment

The backend part of the application was the most important part of the assignment. Basically I had to create a web service that obtained certain information from the vehicle repository system. This was done through simple queries at first, just to test to see if the right information was being obtained from the corresponding registration number. After that was done we had started designing the web service. SOAP was being used by default as our communication protocol.

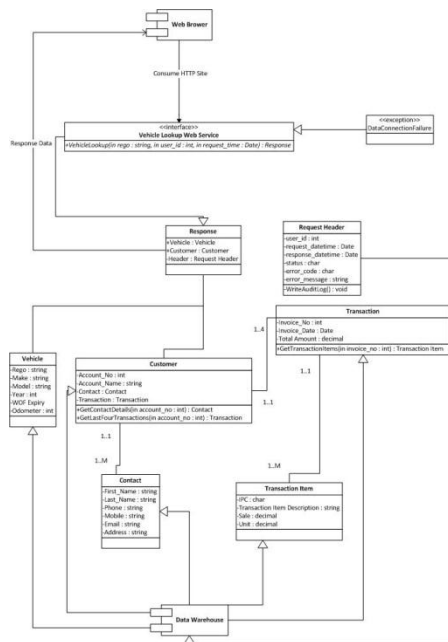


Figure 9: Class Methods for Web Service

SOAP or REST

SOAP (Simple Object Access Protocol) is a protocol that is used to communicate between applications. SOAP uses XML as its message format and HTTP for the message transfer. This is a very well designed mature technology that has widely been adopted in the industry. It is versatile because it allows the use of different transport protocols. Since the SOAP model tunnels fine in the HTTP get/response model, it allows for easier communication through firewalls. As stated early, SOAP was the default protocol when I created my web service. The reason this was defaulted was because generally SOAP objects are very easy to consume. A disadvantage is that SOAP is very slow compared to other technologies because of its heavy overhead (XML) when sending larger messages.

REST (Representation State Transfer) is an architectural style, not a protocol. REST philosophy adopts that the existing principles and protocols of the Web are enough to create robust Web services. This means that developers who understand HTTP and XML can start building Web services right away, without needing any toolkits beyond what they normally use for Internet application development. Another benefit of the REST is that requests and responses can be short. SOAP requires an XML wrapper around every request and response. Once namespaces and typing are declared, a four- or five-digit stock quote in a SOAP response could require more than 10 times as many bytes as would the same response in REST. REST uses the standard HTTP request and response mechanism, simplifying implementation and providing for a looser coupling of the client and server. Note that REST also supports the transfer of non-XML messages such as JSON (JavaScript Object Notation). One of the main disadvantages is that data needs to be secure should never be sent as parameters in URIs, which is the case using REST. And large amounts of data, like that in detailed purchase orders (POs), can quickly become cumbersome or even out of bounds within a URI.

I haven't decided on which protocol I would be using in the mobile application as yet, because both

methods pose strong cases for them to be used. They are too close to call and will make the decision when creating the web service itself.

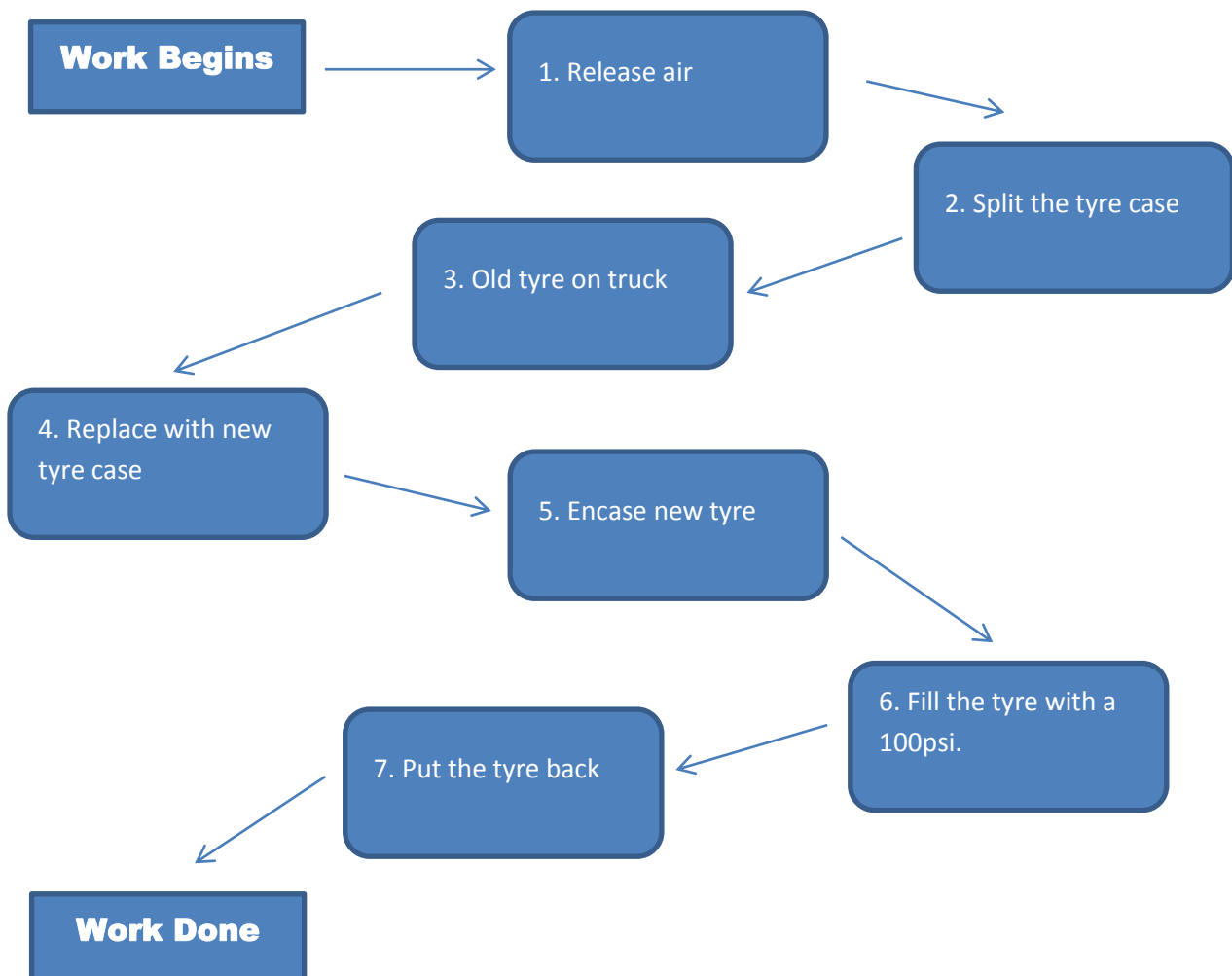
Chapter 5

The Bridgestone Field Trip

In search of understand and defining the scope even further in this project, Bridgestone decided that a field trip out into the actual workspace was imperative. This would help me by understanding the workflow process of each job being undertaken by a fleet technician and the completion of a Vehicle Inspection Report (VIR).

There were two different common scenarios which were being undertaken at jobs: Tyre Replacements and Tyre Punctures.

Tyre Replacement



Tyre Puncture



By learning the physical workflow process of what the fleet technicians do at the job, we were able to find out what the flaws are in this process. It is much a disorganized process, and by me going out onto the field, I was able to put forth an organized and flexible scheme which should be done at

every job. By following what the application does, the fleet technician can be doing the job quickly and efficiently solving the issue of the client efficiently and completing the VIR form simultaneously.

Chapter 6

The Tyre Monitoring System Application

Workflow

There were two ways to go about designing what the final workflow of the mobile application is going to be: online or offline. Both methods have their advantages and disadvantages, but both methods do solve the underlying goal of the project: to complete Vehicle Inspection Reports as quickly and as efficiently as possible.

Offline mode

The offline scenario requires the fleet technicians at the start of every work day to download an image of the data warehouse directly into the Tablet PC. This can be done via Wi-Fi at the Bridgestone warehouse itself. Fleet technicians then can head out to the field with their tablet offline out onto the field. Once at the field, the fleet technician enters the registration number of the Vehicle is inspecting upon. Most of the customer fields will be filled up with information linked to that registration number through querying the image. Scenarios where the fields haven't been filled are where the technician has to manually enter the information about the vehicle himself. There will be appropriate validation based on the selected fields.

Once the details have been filled, the technician has to complete the VIR form on the tablet. This would contain a vehicle configuration diagram and a notes panel for additional notes on the vehicle that is necessary in the report. Validations in the form of backend Service Level Agreements (SLA) for each client are included when the fleet technicians are dealing with rotations and tread depths of the tyres.

A damage report option would be available as well to the user if that possibility would occur. Once the VIR form is completed, it would be ready to be synchronized with the Job Card application. From there, this application can be closed, and the information that has been filled out would automatically be populated onto the Job Card Application. Once the fleet technicians has returned back to the warehouse, he would be able to synchronize the all the information back onto the data warehouse.

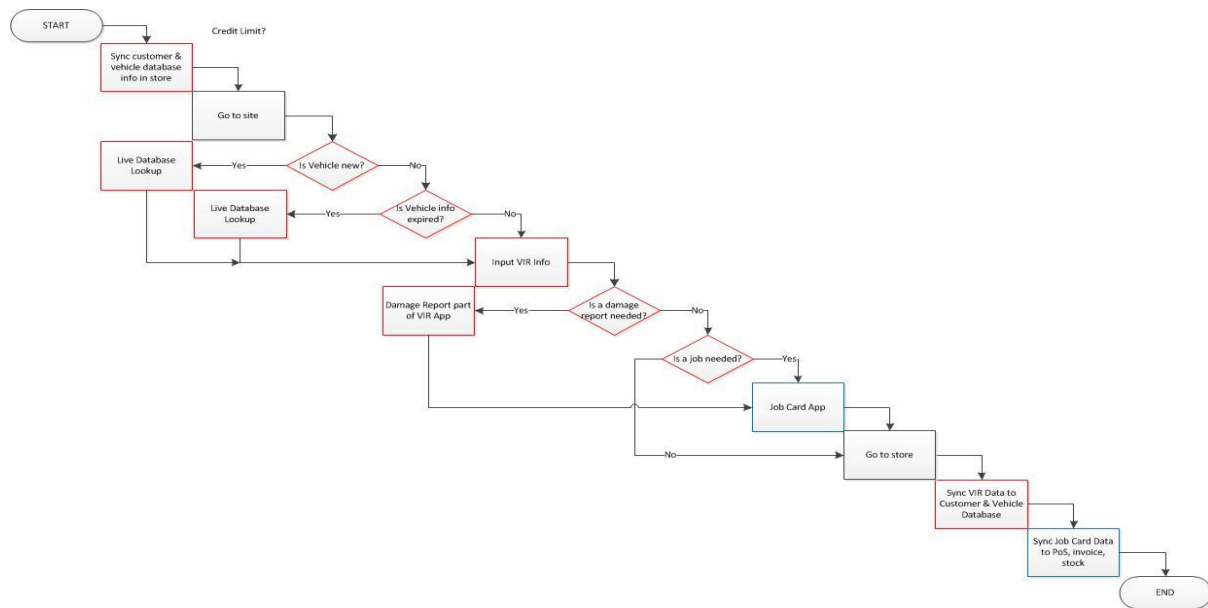


Figure 10: Workflow of offline method

Online Mode

The online scenario would have almost the same approach in terms of usability of the application, but everything would be done online via 3G. There would be no database image on the application. When the fleet technician leaves for the job, all he has to do is to enter the registration number of the vehicle he would be working on, and the web service would populate all the necessary information in order to complete the report. The same process would be undertaken from then on till the synchronization of the two mobile applications. The main downside to the method is when the fleet technicians are doing jobs in areas where there is no 3G connectivity; he would not be able to obtain customer information when entering the registration number. In this particular scenario, the fleet technician would manually have to enter the customer details onto the application and continue with the same process and record all Vehicle Inspection information into the application. Since there is no 3G connectivity, there would be an option given to the user apart from synchronizing with the Job Card Application, which would be to store the VIR into a temporary storage. This way no information is lost, and when in 3G space, you can send the VIR back to the Bridgestone Data warehouse. Offline mode was designed to save online costs, but when done further analytics on the costs of 3G plans and the number jobs done per day, it's well worth designing it using the online method. A decision hasn't been made Bridgestone whether to give the go ahead on the online method of the application yet.

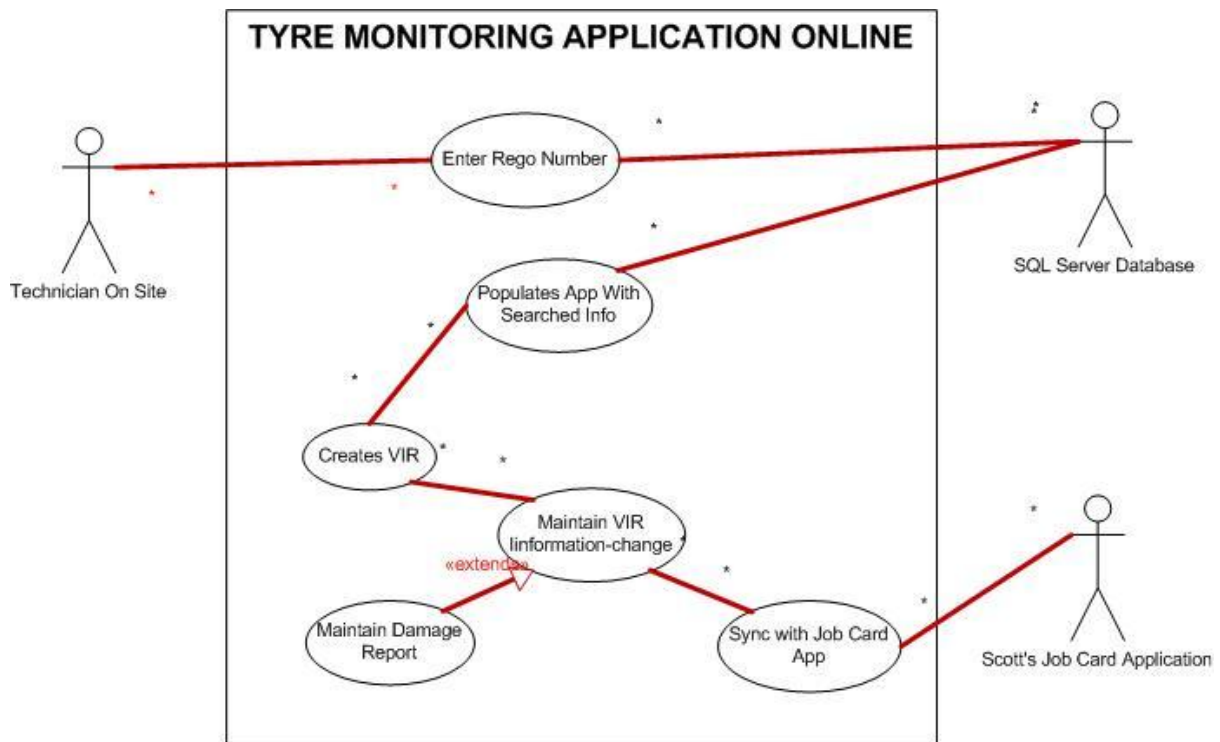
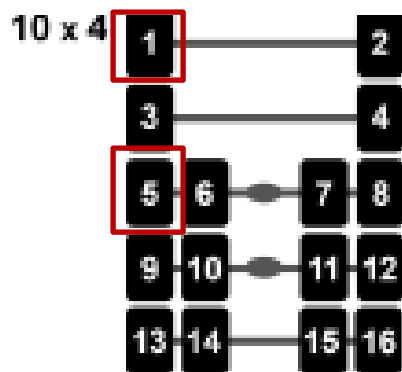


Figure 11: Use Case Diagram for online method

Key Specifications

Interactive Vehicle Configuration Diagram



There will be an interactive vehicle diagram in the Vehicle Inspection Report, as I feel that it makes the fleet technicians life a whole lot easier than dragging and replacing rather than typing out the tyres that needs jobs to done on them. All the fleet technician has to do is to highlight the wheels that are going to having jobs done on them and this diagram gets synced to the Job Card Application.

Synchronization

The mobile application on the tablet PC must have a 'SYNC' option at the end of the application, so that the fleet technician can decide whether a job is required to be done and needs to synced across the Job Card Application or otherwise sending the form directly to the warehouse. If in a remote location without 3G the fleet technician also has the option of storing it in the tablet PC storage itself, which can later be used when in 3G or Wi-Fi space.

Validation

In this mobile application, it is very important that the application validates everything, every time the fleet technician adds/modifies/ deletes. It has to validate based on what information is suited to that field and the tables corresponding with that field in the data warehouse itself. Also, SLA's are going to be implemented in the same manner as validation. For each client Bridgestone has, there are certain agreements made between the two parties. These rules are going to be applied as soon as the fleet technician deals with the client, and in the background the rules are set to validate anything which goes against those rules.



Vehicle Lookup Function

If faced with the scenario that the vehicle doesn't exist in the Bridgestone data warehouse, the fleet technician would require to search the vehicle repository system in order to get the information relating to the registration number and carry on completing the Vehicle Inspection Report. This task would be very similar to the web service we created before but I just need to expand on it a little bit further to integrate this specification onto the application.

Chapter 7

The Next Step

The next stage would now be designing and developing the application starting from the front end design interfaces to the back end web service creations. The big decision will be made whether Bridgestone wants the mobile application wants to be an offline or online tool, and based on that verdict I can go ahead and develop the application. REST versus SOAP is another choice I have to make when it comes to designing the web service. Both approaches have got the pros and cons, and I will have to weigh them out against each other and make a decision. I will be designing the application at Bridgestone headquarters subject to the machines having the Windows 8 Operating System already setup on them. If that's not the case, I will be able to use my own PC to develop the application till there is a machine setup to interact with the database. I hope to successfully finish off my interface designs within the end of July, leaving me for 3-4 months to develop the mobile application itself along with detailed research and documentation to go along with it.