Project: Connect

Folio

Daniel Ferguson, Ballarat Grammar

Software Development SAT, 2016

Table of Contents

[Design Brief 3](#_Toc452931667)

[Audience 3](#_Toc452931668)

[User Characteristics 3](#_Toc452931669)

[Secondary Students & Teachers 3](#_Toc452931670)

[Independent Teams 3](#_Toc452931671)

[Scope 4](#_Toc452931672)

[Items within Scope 4](#_Toc452931673)

[Operating Environment 5](#_Toc452931674)

[Operating System & Version 5](#_Toc452931675)

[Hardware Platform 5](#_Toc452931676)

[UI / UX 5](#_Toc452931677)

[Network & Security 5](#_Toc452931678)

[Data Collection 5](#_Toc452931679)

[Interviews 5](#_Toc452931680)

[Daniel F. – Captain of Energy Breakthrough, Grammar Racing Team 5](#_Toc452931681)

[Finn Clark. – Vice Captain of Grammar Racing Team 6](#_Toc452931682)

[Brandon M. – First Mate of Grammar Racing Team 6](#_Toc452931683)

[Functional Requirements 7](#_Toc452931684)

[Non-Functional Requirements 8](#_Toc452931685)

[Constraints 8](#_Toc452931686)

[Economic 8](#_Toc452931687)

[Technical 8](#_Toc452931688)

[Legal Requirements 8](#_Toc452931689)

[Usefulness 8](#_Toc452931690)

[Ease of Use 8](#_Toc452931691)

[Use Case Diagram (UCD) 9](#_Toc452931692)

[Context Diagram (CD) 10](#_Toc452931693)

[Data Flow Diagram (DFD) 10](#_Toc452931694)

# Design Brief

Energy Breakthrough is an amazing program, run throughout many schools across Australia, through independent groups and keen teams. The main goal of this program is to try and find viable alternatives to cars for transport, as they make up a large proportion of the CO2 we produce. One of the main events of the year is the Energy Breakthrough 24-hour race that is held each year in Maryborough, composing of hundreds of teams, all attempting to achieve the best lap times, number of laps and efficiency through their trikes (recumbent bikes).

To be more effective, teams need to be able to monitor the bikes, battery stations and riders effectively. This will allow teams to make smarter decisions about how they deploy and fit out their bikes, when and how hard specific riders should ride and when to bring them back into pits and change.

A solution, an application that helps teams to read these variables and make smarter decisions would help to raise the level of potential in the ordinary and extraordinary teams – hopefully taking us one step closer towards finding our next viable transportation method.

## Audience

This document is intended for developers, testers and further documentation (API, Backend Server) writers.

## User Characteristics

### Secondary Students & Teachers

#### Frequency of Use

Once or twice a year, but frequently/constantly while at races (12-24 hours)

#### Technical Expertise

Above ‘average’ user, understands how to use simple/basic programs. No knowledge of how GPS, Servers, APIs etc. work.

#### Experience (with Monitoring Applications)

Little to none prior.

### Independent Teams

#### Frequency of Use

Ranging from multiple times a month to few times during the year.

#### Technical Expertise

Above ‘average’ user, understands how to use simple/basic programs. No knowledge of how GPS, Servers, APIs etc. work. May understand how GPS basics work.

#### Experience (with Monitoring Applications)

Moderate to large amount of experience.

# Scope

## Items within Scope

#### Output

Outputs of this application will be viewable data from the bike that will help the users be able to make smarter decisions. Some examples of such data will be GPS signals, speed, laps, average lap times, average speed and more. The application will also be able to show the battery levels of the racing bike, but also the status of the battery in the charging tents (which is beneficial for teams that are EEV – energy efficient vehicles, which use batteries and human powered methods (such as pedalling).

#### Input

Inputs for this application will be the ability to toggle the status of the bikes headlight and LED under lights. Both are crucial for the rider, as a 24-hour race has a number of hours in the dark, and headlights are required by event organisers on each bike. To avoid hazards, event organisers will usually make it a rule that lights have to be on by 8pm. Majority of the pits will become congested during this time because of the sudden flow of recumbent bike traffic, slowing down lap times significantly, and causing teams to loose precious laps. The ability to be able to turn headlights and LED under lights off and on from the application will greatly help to minimise this loss of laps and lap times.

To protect others from accessing tools such as LED lights, users will need to log into the application with pre-defined users. This will help the solutions security.

#### Processing

The user/s will be able to view all of the data and statues, which will be hosted on a cloud backend server, from the app. It will communicate via 3G/4G or over Wireless networks.

Changed variables (e.g. Headlights) will be sent to the server, updated, and pushed to the bikes for further actions required to fulfil such requests. Information from the bikes (e.g. battery levels and GPS locations) will be pushed to the backend server, then pushed to the application for the teams to view in real-time.

There will need to be a component/page that allows users to login. This will give logged in user’s permission to view and change variables from the bike.

## Operating Environment

### Operating System & Version

The application will be built of the Xamarin platform, so it will be cross-platform compatible. To allow suffice time for building though, I will limit the application to the Android OS, though it does have the ability to be ported to iOS.

### Hardware Platform

Because it will be run off the Android platform, the user will need a hardware platform that can run Android APK packages (e.g. an Android Smartphone, Android TV).

Any of these platforms will need to have:

* Access to the internet (Cellular, Ethernet or Wireless)
* 2GB of Free Space
* 1GB of RAM (Recommended 2+GB RAM)
* An ARM CPU (Or equivalent to such)

### UI / UX

The user interface will need to be clear and concise – helping to minimise clutter and only present necessary variables and information when it is most required. This will give the user the best possibly experience, almost making the application intuitive and seamless.

### Network & Security

As stated above, the platform will need to have access to the internet in some form. This will allow for the application to communicate the the Backend Server and access the User Database.

# Data Collection

Members from the Ballarat Grammar Energy Breakthrough Program & Grammar Racing Team, recently taking out 2nd and 3rd in their category for EEV racing.

## Interviews

### Daniel F. – Captain of Energy Breakthrough, Grammar Racing Team

1. **Why is it important to improve the efficiency of the EBT Bikes?**

Because we want to win! We have tweaked everything we can think of to make the bike better, our riders are training to reach peak fitness. We want to get maximum efficiency from the bike, the ride and the battery over a 24-hour period. We are capable of winning but to do this we need real-time information and control of key data to optimise performance.

1. **What information do you need to get the most out of your bike, riders and battery?**
2. We need to know where the bike is, how fast it’s travelling – real-time and average, and if it has stopped for any reason.
3. We need to know battery power levels in the bike and at the charging station so we can optimise power levels and change over times.
4. We need information about each rider and how they are performing overtime
5. We need to be able to remotely manage the lights on the bike
6. **How will information be made available most effectively?**

The team needs to be able to monitor this key information in real-time at the pits. Information needs to able to be access remotely from the pit and anywhere else at the venue. Therefore, it can’t be dependant on wired or wireless connections. We need information to be available on handheld devices

1. **Who need to be able to access this information?**

All member of the team need to be able to see information about the bike, riders and the batteries. Only authenticated users should be able to manipulate data, e.g. turning on the headlights remotely

1. **When do you need this application to be available for use?**

The highest priority is that it be ready and available for use at Maryborough in November, 2016. It would be a bonus to be able to test it at the Bendigo Grand Prix on the 22 August, 2016.

### Finn Clark. – Vice Captain of Grammar Racing Team

1. **Why is it important to improve the efficiency of the EBT Bikes?**

Because the bikes themselves are amazing devices and we have made the bikes the best they can possibly be, our riders are trained and made for these 24 hour rides no all we need is real time information sent to our pit crew members to make for the most setup race we can possibly have.

1. **What information do you need to get the most out of your bike, riders and battery?**

We need to know speed and consistency of the speed that is being run.

We need to know the info of the rider and how they are going in the bike and if they will need a break or anything like that.

We need to know how the battery is coping with the ride and how everything is handling itself.

1. **How will information be made available most effectively?**

If it is a app in which the pit crew can see as well as the members at the camp so if there is an emergency it doesn’t become an issue and everything can be dealt with, in the most efficient way.

1. **Who need to be able to access this information?**

Everyone who is either riding or in the pit crew because it is widely needed information to make decisions on what will happen when

1. **When do you need this application to be available for use?**

Preferably before Maryborough Energy Breakthrough Race.

### Brandon M. – First Mate of Grammar Racing Team

1. **Why is it important to improve the efficiency of the EBT Bikes?**

Its a race, efficiency is obviously key

1. **What information do you need to get the most out of your bike, riders and battery?**

We need to know a battery and riders fatigue and the battery levels to plan out stops, less frequent pitting means more racing

1. **How will information be made available most effectively?**

Information to a screen of course from a remote spot for the team to monitor in real-time

1. **Who need to be able to access this information?**

This info will prove vital in the pits, the more we know about what’s going on when the bike is on the track, the less time needed to figure it out whilst the bike is stopped

1. **When do you need this application to be available for use?**

We need this effective by august 22 when we have a race to win!

## Functional Requirements

|  |  |  |
| --- | --- | --- |
| **Sequence** | **Description** | **Comments** |
| FR01 | Pull Data from Server | The application will need to able to Pull data from the Server, and display them appropriately and timely. For instance, the GPS location will need to update at least once per second, and be able to display as many bikes as teams are riding. It will also need to give accurate reading, because teams may need to locate their bike on the track in case of malfunction. |
| FR02 | Display Information | Once the application has Pulled the data from the Server, it will need to display such data appropriately as to create information from it.  This can be seen in that fact that GPS Longitude and Latitude will be used in a map, Headlight status (ON, or OFF) will need to be displayed in the way of toggle switches that display the current state of the variable, etc. |
| FR03 | Push Data to Server | Once users make changes to data (e.g. turning Headlights ON or OFF), the application must have the ability to PUSH such data to the Server, so that the Server can make appropriate actions to continue. |
| FR04 | User Authentication | Because there are certain functionalities, such as toggling Headlights, that shouldn’t be open to change from any user of the application, the application will need to have a user authentication process. This user authentication process will allow ordinary users to view information, and authenticated users to view and change variables – protecting the rider, software and authenticity/reliability of the application. |

Functional Requirements Table

## Non-Functional Requirements

|  |  |  |
| --- | --- | --- |
| **Sequence** | **Description** | **Comments** |
| NFR01 | Ease of Use | The application should be non-invasive, and easy to use. It shouldn’t be hard to use, nor hard to learn. It must also not get in the way of operation, being efficient will be key to achieving this. The application should also be intuitive in ways, to assist the user. |
| NFR02 | Reliability | As the program will be run over long periods of time, it should be incredibly stable, and error/warning free. This will assure users that they can reliably use this application as their main source of information. |
| NFR03 | Portability | This application needs to be small, and use as little resources – both memory and connectivity. This will keep the usage tax down, as if its connectivity resource heavy, it might cost users more than necessary. |

Non Functional Requirements

## Constraints

### Economic

This project has no bootstrap funding. This will mean that all resources will be generated in house, there is no money for external servers, frameworks, communication protocols, etc.

### Technical

The platform will not have a lot of resources – RAM, Disk space, etc.

### Legal Requirements

With user authentication, I have a legal obligation to respect the privacy of user credentials. I also have an obligation, legally, ethically or otherwise, to protect such information with some level of encryption.

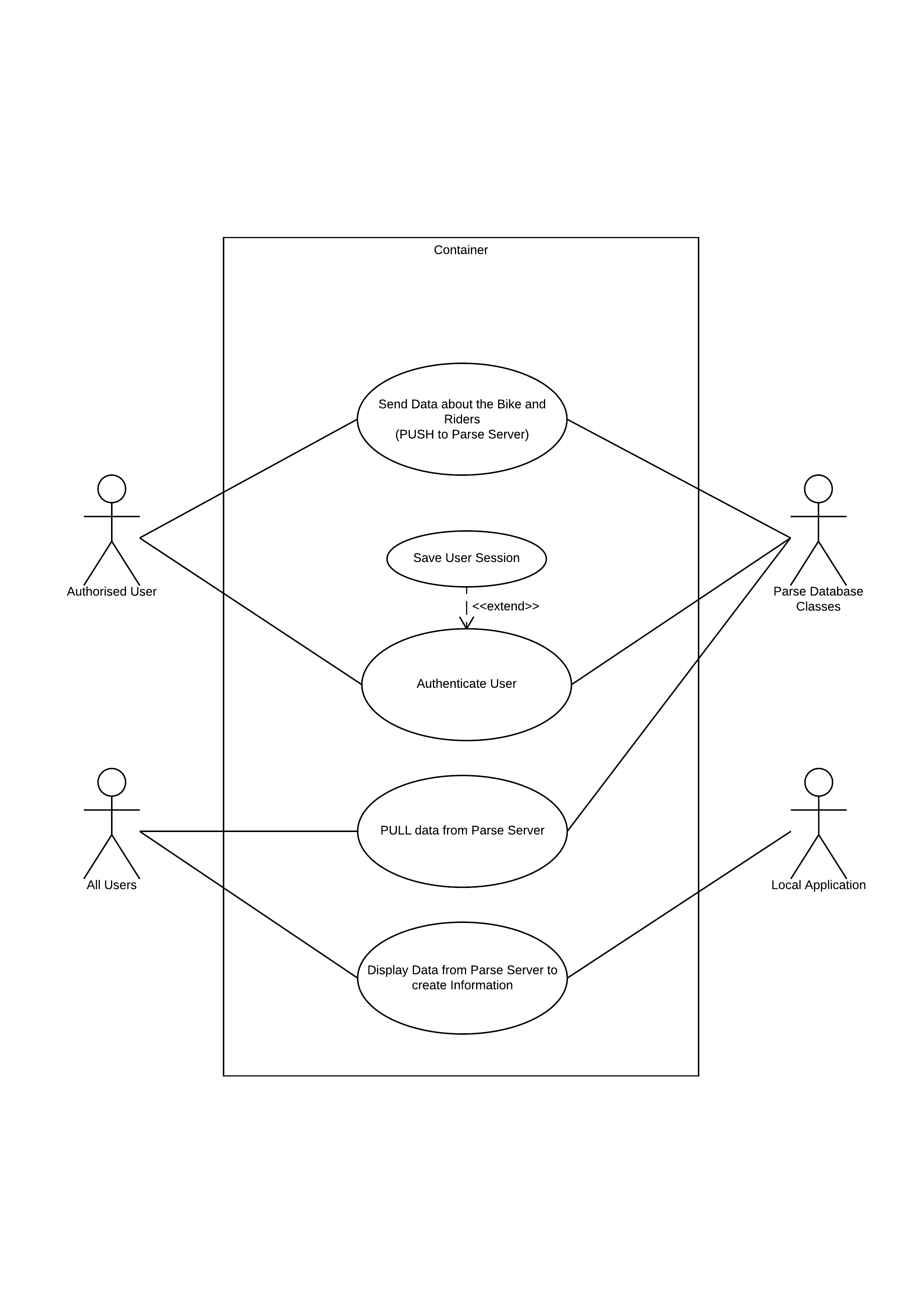
### Usefulness

The application will need to be lightweight, perform its job and run as fast and reliably as possible. The combination of these constraints will help achieve maximum usefulness for the users.

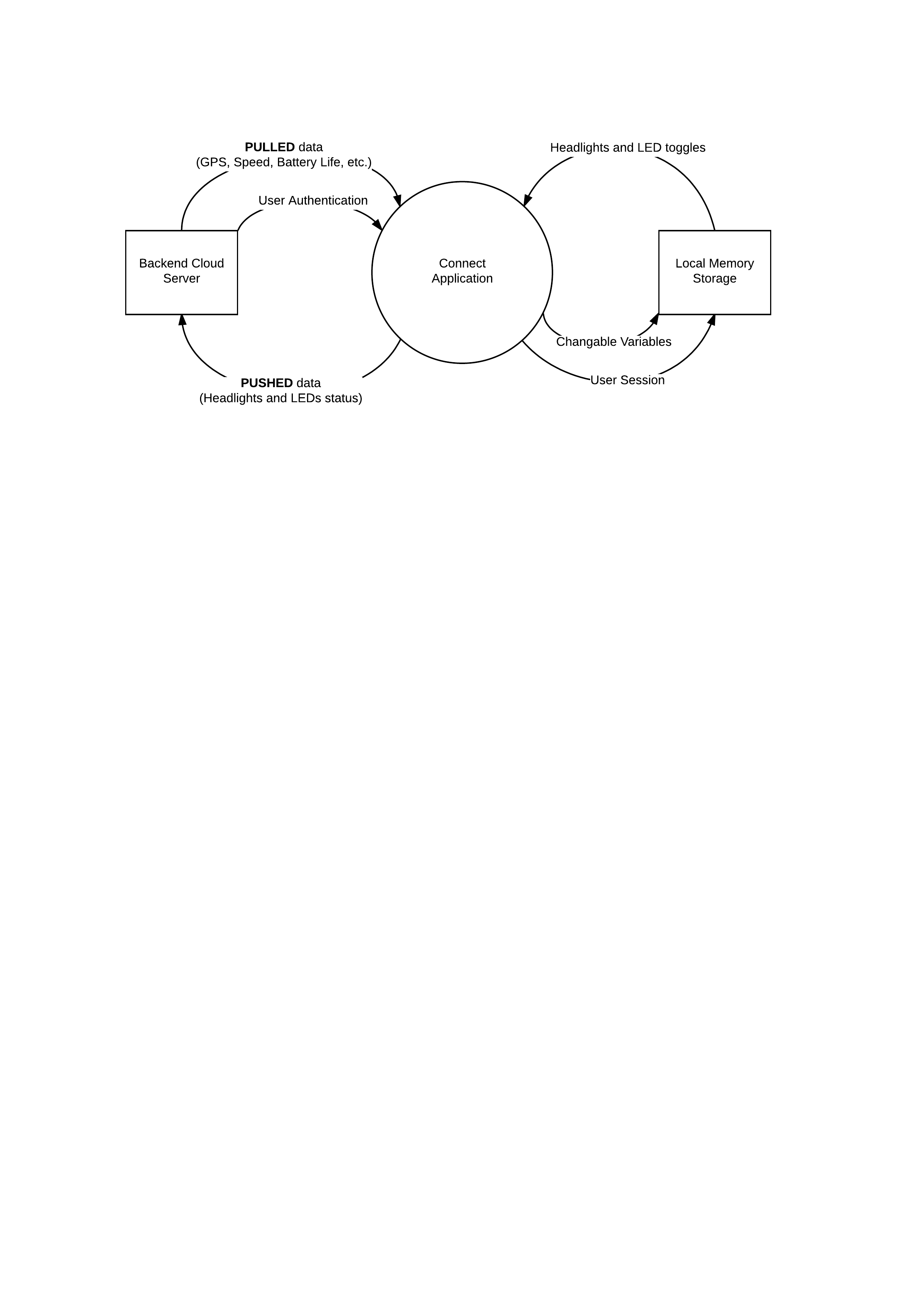
### Ease of Use

The application should be as simple and clutter free as possible. Achieving this will mean that users from all sorts of technological backgrounds will be able to use the application, and hopefully apply the information to their team’s circumstance.

# Use Case Diagram (UCD)



# Context Diagram (CD)



# Data Flow Diagram (DFD)

