Time Line:

[date]: at this initial meeting I needed to desided on 3 potential projects. I knew I wanted to d o something to do with technology.

[date]: We visited the university of Bristol libery, in this we learned how to find things in the libery. However after looking around I found that the internet will contain more useful information due to the nature of the topic.

23-7-14: Because I was flying to America I had 11h so I wrote: What am I using, in which I decide what hardware I will use and how I will go about the project. And decide on the protocol and how I am going to transfer the data between the client and server. However both require more resurch whitch I require internet for and I do not have internet.

25-7-14: While at my hotel I did research in to how to use sockets/the internet in java and java for android. This includes sevral sample projects such as a server that capitalises every thing sent to it and a simple chat  example allowing multiple people to chat.

26-7-14: While driving I start to work on the basic code for the server. See progression of code, RC sever.

27-7-14: I started planning the android app(see planning/android app). i did the initial UI.

28-7-14: I finished the UI and added a few things as documented in planning/android app. I also progressed on the protocol and how it will be transmited.

[STUFF]

11-8-14:I added a movment.class to the server. This will proses the input from the server. I also made it so that the server will loop constantly taking feedback currently all it dose is prints it to console. Also see progretion of code: android app for today. Also see progrestion of code for The server.  
I have also met the first mile stone for the server and App. The idea of this is that I will save the code at its current state and it gives me something to work to.

16-8-14: added to both the code for the server and app so that the disconnect button works. Edited the Protocol.

DO BUILDY STUFF

25-8-14: I rewrote the server using UDP as explained in the RCserver proration of code doc.

26-8-14: I rewrote the client to use UDP, and made edits to the server. I also added to the protocol doc and made the appropriate changes.

27-8-14: I added the steering motor as show in the RC car doc. Updated to a 32 Tick rate as show in the protocol doc.

29-8-14: I changed the starting text and changed the pins for the motor’s, as explained in the RC server document. I also changed the App so the labels for the left and right buttons are correct and updated the code as show in the android app doc. I then prepared to put it inside the car and almost finished it when the raspberry pi died as explained in the raspberry pi document.

4-9-14: I set up the new raspberry pi as show in the raspberry pi document. I also completed the car it is now working as desicribed in the RC car doc. This is mile stone 2: the point when the car is fully working and the app and server are feature complete. I have attached a video of the car working.

10-9-14: I spent lots of time “cleaning” the code. This involved re writing the move class for the server. I also started commenting the code.

12-9-14: I spent more time predominantly commenting the code. I wrote a small bit in the app progrention of code doc as to why this is important.

14-9-14: I have re done the “expected time of things to be done” spread sheet and have redone the expected time line as I have had lots of delays

16-9-14: Because I want to be able to edit all of the files and code for anywhere, I have used GIT and Google drive. I have uploaded the folder to google drive to save re organising it and dealing with lots of single files, I have also zipped the code so that it is smaller and less files to upload. For my code I have used GIT hub this is a very large website that anybody can upload code to using GIT. GIT is a version control system that is mainly used in coding, this allows you to sync multiple clients to one “code base” it also allows you to roll back to a previous versions. I have also complied the app and in the “APK” folder. I followed the instruction on GIT Hub’s website to uploading the code using unix-like terminal.

17-9-14: today I had an EPQ tutorial session. From this I gathered that I should be using a proper referencing system, so I will use the Harvard system of referencing, I will use this for all the places I have gotten code from. So I will go thru all the documents and change the way I have referenced things as it is important to us an academic method. This is an important skill to use as it allows me to produce more professional looking documents.

24-9-14: I am thinking of ways that i can evaluate my project. i Think i can do it in x ways

1. What other think of it: I could do a survey or just interview people.
2. Look at the features it has and the ones I aimed to have.
3. Analyse how easy the app is to use and good it looks

Number 3 can be partially achieved by doing number one.

So today I started to make a question are to give people how have tested the app. On making the questionnaire I have thru trial and error worked out what looks best, I have used indents and consistent formatting to make it look good. It is very shit but this saves me paper.   
I also uploaded all of the files for my EQP (in this folder) to github as Google drive allow useful is far too slow and I find it takes to Meany clicks to do something simple EG edit a document.  
I have also included links to all the git repos in the root of this folder.

2-10-14: I have modified the questionnaire to make the questions more direct. I ask my teachers at school for help with this.

Email somnbody add more points, resurch

TODO: Add skills, refrence add more sorcers compare and analyse