**Project Proposal: Software Engineering in safety-critical systems**

**Motivation and rationale:**

In Software Engineering, there are many different approaches that can be taken to tackle a specific problem. Development processes such as agile are often used, along with programming languages which help to achieve fast development of software but at a cost to software quality [4].

A Safety-Critical System has the properties that where a failure could lead to human death or injury, loss or severe damage to equipment or environmental harm.

There is also a thought that developing software in a safety-critical way can be very time consuming due to the software processes and the way in which safety-critical languages can be hard to program in but, the main problem is that the need for rigorous analysis and proof of absence of run-time exceptions that’s required for safety-critical development, that makes specific demands on the programming language, that makes development more time consuming.

Within safety-critical there are few languages capable of saying it is fully safety critical to name a few Spark Ada [7], Spec # [5], safety-critical Java [6] and Checked C [8]. Within these, there are some very different programming styles as with spark Ada requiring a lot of heavy lifting work to program in and Spec # being an extension of the C# language has many features that can help developers prove that their code is correct before release, which could also be used within a normal development.

safety-critical Systems help aid this process, although safety Critical languages are usually used for safety-critical programs, some of the tools within the languages can aid people in the development of regular software. The Project will compare the use of these languages within a correct safety driven process to be able to aid programmers to choose which safety-critical system would best suit their needs.

This project would help aid the use of selecting which one would be best for certain areas with a carefully thought out evaluation criteria to compare two safety critical languages and comparing them together. I will also involve the use of Java as I have had 2 years’ experience using java and think that it is important to involve a regular OOP language that it very popular to use within a software process such as agile. This project would also aid my personal development as a programmer forcing myself to learn two languages, which will be spark Ada and Spec #. Would also help me in learning how to write correct code and in this sharing my ideas on how to do this using these safety critical languages.

**Aims and Objectives**  **Aim:** Investigation of approaches to safety-critical systems development.

The Aim of the project is an Investigation of approaches in which I look at a range of software development methods, draw comparisons to them in which I will choose the one, which is most beneficial to safety. I will then run an investigation by developing a small safety-critical system and comparing two safety-critical languages.

**Objectives:**

1. **Research into software process methods for safety-critical Systems.**

Researching into the correct software process method to use will be critical to the project for making sure areas are covered, at least four should be researched before starting anything else to make a comparison between which is best for a safety critical System.

1. **Research into safety-critical languages and see what features they offer.**

Researching current safety critical languages will involve looking at what kinds of safety languages are out there to date. I want to focus on the languages that companies are more likely to use to develop the systems I will do this by trying to find sources, which point to some popular safety critical languages, and trying to find a correct balance between the two. Eg if I choose Spec # and checked C then it would not make much sense as they are similar.

1. **Develop a full evaluation Criteria which evaluates both programming languages for a comparison**

This will be based on the research I have gathered together to find a full evaluation Criteria which should be based on things such as features, code coverage, usability and correctness.

1. **Learn two safety-critical Languages.**

I want to learn two safety critical Languages this is an important part of the project and it is also a personal aim which aids me in my development as a programmer.

1. **Develop a Rail Crossing simulation using two safety-critical languages.**

This development will be done with 2 safety critical languages, they will follow the same rules and implementations will not be different to ensure fairness for the evaluation.

**Background:**

|  |  |  |
| --- | --- | --- |
| Citation | Summary | Relevance |
| “Software Engineering” Ian Sommerville [4] | This book gives an overview of software engineering but mainly gives a lot of detail on software processes, modules and a good introduction into understanding principles for software engineering. | This book scoped the aim of my project. It contains the relevant information for choosing software process methods with discussing processes such as Agile, Waterfall, Spiral and variations of these models. |
| “The choice of computer languages for use in safety-critical systems” W.J. Cullyer, S.J. Goodenough and B.A. Wichmann [3] | This paper gives an in depth look into some safety- critical languages. Concludes that a well-defined sub-language is essential for use in safety-critical projects. | Through reading this it suggests some languages where some features could be stripped away and have a strict subset of these. These helps aid the choice of my languages I have only chose one from these as more of an in depth look due to the fact of some languages being old and some languages have strict sub sets of themselves such as Ada. Ada has spark which takes a subset of Ada as a safety critical language. |
| “Limitations of Agile Software Processes” Dan Turk, Robert France, Bernhard Rumpe [2] | This Paper sums up the limitations of Agile through Pointing out the limitations in certain areas and provides variations of agile which are not good in certain aspects. | This Paper provides a good range of reasons why agile has some limitations however the aspect of safety critical software is relevant, choosing the appropriate software method is vital within a safety critical development. This helps in proving that agile is wrong type of development for safety-critical systems. |
| “Railway Level Crossing Safety Bulletin” Australian Transport Safety Bureau [1] | This paper gives an overview on how accident happen in railway crossings it also gives a section dedicated to railway crossings. Gives several different reports on accidents that have happened, and gives a conclusion as to why it has happened. | This is a good paper for when I am developing my software. To know certain risks that play a factor with railway crossing is vital to make sure I can get a full scope of what could potentially happen in real life. As the point of safety critical systems is to try and change |
| “A (Very) Short Introduction to SPARK: Language, Toolset, Projects, Formal Methods & Certification” Eduardo Brito. [7] | Gives an overview of the spark language including sparks toolset previous projects and formal Methods. | This paper gives an overview highlighting some key areas of spark which I am interested in such as the spark toolset and the program verification. It helps me learn how to program in spark Ada and make use of some features. |
| “The Spec# Programming System: An Overview” Mike Barnett, K. Rustan M. Leino, and Wolfram Schulte [5] | This paper describes what the goals are of Spec #. It gives examples on where and how to use C# and gives the reader some intuition into what spec # is about | This paper is a very nice paper which gives some examples of spec # and nicely tells the reader on why these methods are used. This helped me a lot in choosing Spec# and helped me learn a lot about the actual architecture of the Spec # extension |

**Diagrammatic Work Plan:**

For a detailed overview of the project I created a Gantt chart to be able to keep track of where I am within the project and what else needs doing. I can keep track of tasks progression too.

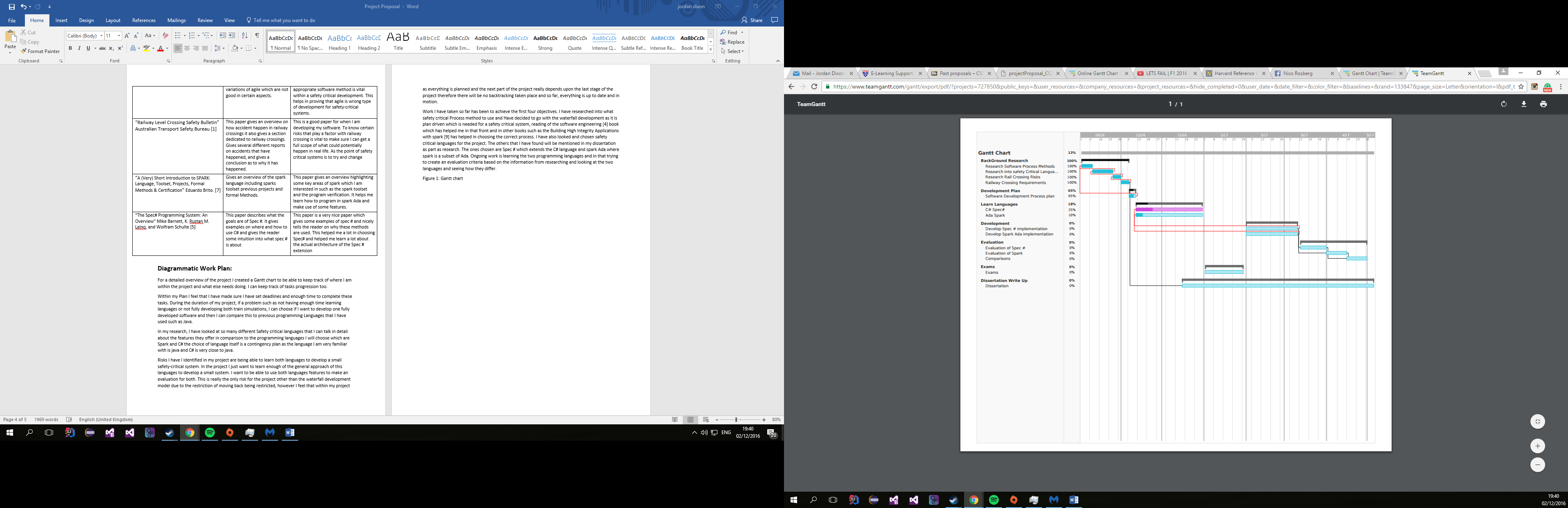
Within my Plan I feel that I have made sure I have set deadlines and enough time to complete these tasks. During the duration of my project, if a problem such as not having enough time learning languages or not fully developing both train simulations, I can choose if I want to develop one fully developed software and then I can compare this to previous programming Languages that I have used such as Java.

In my research, I have looked at so many different Safety critical languages that I can talk in detail about the features they offer in comparison to the programming languages I will choose which are Spark and C# the choice of language itself is a contingency plan as the language I am very familiar with is java and C# is very close to java.

Risks I have I identified in my project are being able to learn both languages to develop a small safety-critical system. In the project I just want to learn enough of the general approach of this languages to develop a small system. I want to be able to use both languages features to make an evaluation for both. This is really the only risk for the project other than the waterfall development model due to the restriction of moving back being restricted, however I feel that within my project as everything is planned and the next part of the project really depends upon the last stage of the project therefore there will be no backtracking taken place and so far, everything is up to date and in motion.

Work I have taken so far has been to achieve the first four objectives. I have researched into what safety critical Process method to use and Have decided to go with the waterfall development as it is plan driven which is needed for a safety critical system, reading of the software engineering [4] book which has helped me in that front and in other books such as the Building High Integrity Applications with spark [9] has helped in choosing the correct process. I have also looked and chosen safety critical languages for the project. The others that I have found will be mentioned in my dissertation as part as research. The ones chosen are Spec # which extends the C# language and spark Ada where spark is a subset of Ada. Ongoing work is learning the two programming languages and in that trying to create an evaluation criteria based on the information from researching and looking at the two languages and seeing how they differ.

Figure 1: Gantt chart



**Bibliography**

[1]Australian Government. (2008). *Railway Level Crossing Safety Bulletin .* Available: <http://www.atsb.com.au/media/28330/rail_bulletin.pdf> Last accessed 30/10/2016.

[2]Dan Turk, Robert France, Bernhard Rumpe. (2002). *Limitations of Agile Software Processes.* Available: <https://arxiv.org/ftp/arxiv/papers/1409/1409.6600.pdf> Last accessed 15/10/2016.

[3]W.J. Cullyer, S.J. Goodenough and B.A. Wichmann . (1991). *The choice of computer languages for use in safety-critical systems .* Available: <https://pdfs.semanticscholar.org/827d/5c1c1b4ec83d92e957784baf314f2a2ddf0a.pdf> Last accessed 10/11/2016.

[4]Ian Sommerville (1982). *Software Engineering*. 9th ed. United States of America: Addison-Wesley. 4 - 77.

[5] Mike Barnett, K. Rustan M. Leino, and Wolfram Schulte. (2004). *The Spec# Programming System: An Overview.* Available: <https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/specsharp-krml136.pdf> Last accessed 15/11/2016.

[6] Daniel Tang, Ales Plsek, Jan Vitek. (2010). *Static Checking of Safety Critical Java Annotations.* Available: <http://d3s.mff.cuni.cz/conferences/jtres2010/slides/p148-tang-slides.pdf> Last accessed 30/10/2016.

[7] Eduardo Brito. (2010). *A (Very) Short Introduction to SPARK: Language, Toolset, Projects, Formal Methods & Certification.* Available: <http://inforum.org.pt/INForum2010/papers/especificacao-verificacao-e-teste-sistemas-criticos/Paper043.pdf> Last accessed 30/10/2016.

[8] Abel Avram. (2016 ). *Checked C - A Safer C/C++ from Microsoft.*Available: <https://www.infoq.com/news/2016/06/checked-c> Last accessed 25/10/2016.

[9] John W. McCormikc, Peter C. Chapin (2015). *Building High Integrity Applications with Spark*. Cambridge, United Kingdom: Cambridge University Press. 1 - 40.

[10] Gantt chart software <https://www.teamgantt.com> Last accessed 02/12/2016