**CO600 Terms and Conditions Analyzer**

Adam Preston (ap720), Euel Domingo (ejd29), Kester Gideon (kg327), Marcus Robinson-Bailey(mer25)

*School of Computing*

*University of Kent*

*Canterbury, United Kingdom*

**Supervised by**: Alex Freitas ([A.A.Freitas@kent.ac.uk](mailto:A.A.Freitas@kent.ac.uk))

**Abstract**

[TALK ABOUT MACHINE LEARNING] - KESTER

**1. Introduction**

[TALK TERMS AND CONDITIONS ARE A LONG PROCESS AND WHY OUR PROGRAM WILL SIMPLIFY IT. INCLUDE FEATURES OF SOFTWARE] - KESTER

**2. Background**

2.1 Market Research

After conducting market research on our specific topic, we came across applications that held the same attributes as the ones we wanted to include in our software (EULAlyzer, Terms of Service Didn’t Read, Polisis). Terms of Service Didn’t Read was the one main application that mirrored features that we wanted to implement. The purpose of performing market research for the project was to gather and calculate ideas for our university project. It also allowed us to brainstorm ideas but more importantly inspired us to make an application that first met user requirements and secondly push the boundaries of our data mining knowledge.

2.1.1 Terms of Service Didn’t Read

Terms of Service Didn’t Read [1] is our biggest competitor within the market we have chosen. This is so, because it both analyses the terms and conditions of a website but also provides browser add on compatibility. This feature would naturally within your browser rate and label website policies with class ratings ranging from Class A (good) – Class E (very bad). This would then inform the user about the rights they have on the specific website they’re signing up to. From Terms of Service Didn’t Read we acknowledged that there were aspects of the application that we didn’t want to implement for the shear reason that the feature would be very time consuming, expensive to implement and would slow down production of our main features.

2.1.2 EULAlyzer

EULAlyzer[2] is another software that held the same concepts as the ones we wanted to implement. Created by Brightfort the software prides itself on identifying important elements of terms and conditions. Even though EULAlyzer wasn’t our main competition like Terms of Service Didn’t Read it again helped with research and understanding what is essential for our project idea.

2.1.2 Polisis

A smaller advertised application was Polisis[3], created by an independent developer. An application that visualised privacy policies using artificial intelligence. It highlights information that a website is collecting from you and possibly sharing to external agencies. Researching both levels of application production shows the difference in quality and outlined how we had to be patient with implementation. Researching a spectrum of applications, it reassured that there would be limitations in our project, and we shouldn’t label the lack of features as a limitation.

**3. Aims**

[AIMS OF THE PROJECT I.E. WHAT WE WANTED TO ACHIEVE, WHAT THE PROGRAM NEEDED TO ACHIEVE TO BE A SUCCESS AND IF IT CATERED TO OUR AUDIENCE] - EJ

**4. Requirements**

[TALK ABOUT SYSTEM REQUIREMENTS, DATA MINING AND AUTOMATED MACHINE LEARNING] - EJ

**5. Development**

[TALK ABOUT DATA COLLECTION & PRE-PROCESSING SCRIPT, CLASSIFICATION ALGORITHM (WEKA), WEBSITE DESIGN, PYTHON WHY CHOSE IT ] - ADAM

**6. Quality Assurance**

[TALK ABOUT TESTING] – ADAM

**7. Challenges**

[TALK CHALLENGE FACED] - MARCUS

**8. Conclusion**

**9. References**

**10. Acknowledgements**

Throughout the CO600 project we were guided and supervised by Alex Freitas. We’d just like to say thank you for all the help and guidance you gave.