Detailed Project Proposal

First name	Anthony
Last name	Sébert
Student number	1705851
Supervisor	Andrei Petrovski

Defining your Project

1.1 Project title

Real-Time Operating System in Rust

1.2 Background

Help: Provide the background to your project. This section should highlight the main topics in the area you are going to research. Essentially what is the project about, what has been done before and why is this project important ? ~500 words

---

This project aims to create a real-time operating system using the Rust language. It addresses a large area of design techniques and algorithms so as to reach three defined goals :

- **deterministic** execution time of the programs (or subparts of them), allowing the processes to be managed accurately
- **correctness** of the time when a result is expected, to meet or miss the deadlines, where treatment marks the difference between soft and hard RTOS
- **predictability** of the deadline associated with a set of constraints to define an expected state of the system

These three concepts are the core characteristics of a RTOS, and will frame the tools used to evaluate the efficiency of such an OS. But a particular effort will be made to discuss the performance of the system, because elegant operations are marks of quality; and in the ways it is possible for the developer/user to interact with it, typically through a shell, for graphic UIs will not be covered in this project. Efficient multitasking is, as well, a core principle of the system, and hence a particular attention to the scheduling policy is required, as well as fast context switches to reduce the scheduler's overcost. Since almost each OS has its own solution to this problem, plenty of examples from different families of policies will be discussed.

Note that some of the popular classic OS, like Windows or GNU/Linux distributions, can be turned into a RTOS either by an extension (RTX and RTX64 for Microsoft Windows) or a microkernel (RTLinux for Linux). Apparently this is not a common solution, for they lack certain specificities when compared to their counterparts.

Such systems are very useful in several sectors, including astronautics, mainframes, aeronautics, robotics, or embedded systems in general (and IoT in particular), for the simple fact that they are highly reliable and almost never crash if they are well-designed. But yet they do not really suit a common daily usage for a regular user because they are task-oriented and not meant to be focused on user experience.

Concerning the Rust language, it is often cited as a potential successor of C++. Indeed, if C++ and Rust are quite close, especially regarding the syntax, the low-level orientation and the memory management, Rust already includes in-built concurrency management and safe operations on memory (that can be bypassed if needed). The community seems quite active, with many contributors and online places (forums, IRC channels, subreddit) where people exchange ideas. A YouTube channel also includes several conferences on specific topics. The most important operating system project written in Rust, Redox, is a microkernel Unix-like OS. There are also a few more projects, some of them for educational purposes, plus several kernels/microkernels, but no real-time OS. A review of these projects during the system's design process will be informative.

> *"The tools we use have a profound and devious influence on our thinking habits, and therefore on our thinking abilities"* E. W. Dijkstra

It will also constitute a precedent in terms of combination of RTOS and Rust, useful in terms of computer science research. The programming language, a tool like any other, will have a deep influence on the final result. Rather than deny this relation, we choose to advantage profit of it, by selecting upstream the technological solution that will help to meet the goals.

1.3 Motivation

Help: To whom is this project important ? A project must address a question/problem that generates a small piece of new knowledge/solution. This new knowledge/solution must be important to a named group or to a specific client (such as a company, an academic audience, policy makers, people with disabilities) to make it worthwhile carrying out. This is the **motivation** for your project. In this section you should address who will benefit from your findings and how they will benefit. ~300 words

Example 1: If you intend to demonstrate that a mobile application that automates class registers at RGU will be more efficient than paper-based registers - the group who would be interested in knowing/applying these findings would be both academic and administrative staff at RGU and they would benefit by time saved and a reduction in their administrative workload.

Example 2: You are demonstrating that a particular 3D model design increases realism in 3D environments. The group that would be interested would be games designers or developers of 3D virtual environment applications. The would benefit from producing more realistic environments that could increase sales of their products.

Example 3: You have designed a new network topology for IrishOil plc's new Aberdeen headquarters. The interested group would clearly be IrishOil. They would benefit from easier maintenance and improved security of their computer network.

It seems that no real-time operating system have been written using this promising technology, probably because an important part of the companies that produce this kind of software started their business in the 80s and chose to use the relevant languages at this time, such as C/C++ with assembly. The advantages of Rust over C/C++ could significantly reduce the risks related to memory (segmentation fault, buffer overflow), keeping the flexibility of declaring variables on the heap or stack.

Embedded systems are sometime not easily remplaceable or maintainable, for example a space probe sent to an asteroid, or an aircraft computer, and then are required to have the lowest dysfunction rate possible; for any events (in the processor instructions stream) leading the RTOS to become inoperative or have undefined behavior, even for a short time, can have disastrous consequences and represent a consequent loss of money or precious data. But such systems, because of their importance, are extremely sensitive to perturbations, and therefore have to be both well-designed and well-implemented, thus the choice of Rust in that case.

This RTOS will have as aim to offer, for companies or organisations working in the fields mentioned above, a safe and reliable framework to develop & deploy applications that fits their needs in terms of real-time operations. That's also why the lisense of this project, a public copyright Creative Commons license, allow to remix, transform, and build upon the material.

And finally, this project is really important for the student itself, long been interested in operating systems, and willing to create one. Even better, in a research context. Not to mention the potential applications in astronautics, one of his centers of interests, that makes the idea thrilling.

1.4 Aim & Objectives

Help: Outline what are the main things your project is going to do and what steps or milestones will be used to achieve this aim. The Aim is unlikely to change throughout your project; however, the objectives are likely to adapt to your ongoing research and development. In particular it is highly likely that you may wish to split objectives into sub-objectives as work progresses. A good clear set of objectives give you something to evaluate your final project against.

Example : For the timetable app outlined above

Aim: To create a functioning attendance application that efficiently automates the taking of class registers.

Objective 1: study existing register system in place at RGU and identify weaknesses

Objective 2: research existing automation technology’s and identify and evaluate those that may be appropriate to taking in class registers

Objective 3: Implement chosen technologies to create prototype application

Objective 4: Conduct user trials to evaluate capabilities of prototype application

Objective 5: Create a refined application incorporating feedback from user trials

---

Aim : to create a functional real-time operating system using the Rust language with the following characteristics : **deterministic**, **correctness**, **predictability**.

Objective 1 : review theory and implementations of real-time operating systems

Objective 2 : familiarization with Rust

Objective 3 : implementation of the operating systems core components

Objective 4 : implementation of a shell

Objective 5 : testing the operating system

1.5 Key Techniques

Help: Perform some initial research into the area and outline what techniques you might research in further detail here. The techniques you cover here should include references to the papers where you have sourced the information. The techniques mentioned here are very likely to become the section headers in your literature review.

Microkernel OS architecture

Type of OS architecture where the kernel is as light as possible and only provide the necessary functions. Typically, clock driver, display driver, IPC, physical memory and scheduler. The kernel, services and programs communicate through IPC. - Hansen, P. (1970). The nucleus of a multiprogramming system. Communications of the ACM, 13(4), pp.238-241. - Wulf, W., Cohen, E., Corwin, W., Jones, A., Levin, R., Pierson, C. and Pollack, F. (1974). HYDRA: the kernel of a multiprocessor operating system. Communications of the ACM, 17(6), pp.337-345.

Scheduling algorithms

The implementation of this algorithm will manage the resources among the programs (including the processor itself) in order to minimize resource starvation and ensure fairness. - Liu, C. and

Layland, J. (1973). Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment. Journal of the ACM, 20(1), pp.46-61. - Buttazzo, G. (2013). Hard real-time computing systems. 3rd ed. Johanneshov: MTM. - Lu, C., Stankovic, J., Son, S. and Tao, G. (2002). Real-Time Systems, 23(1/2), pp.85-126. - Meumeu Yomsi, P. and Sorel, Y. (2007). Extending Rate Monotonic Analysis with Exact Cost of Preemptions for Hard Real-Time Systems. 19th Euromicro Conference on Real-Time Systems (ECRTS'07).

JSON data format

Simple and widely used, this data format will be used by the processes to communicate with each other. These types of data are sufficiently generic and abstract to be represented in any programming language, on the one hand, and to represent any concrete data on the other. A library for Rust is available on GitHub under free software licenses (MIT or Apache 2.0 most of the time). - The JavaScript Object Notation (JSON) Data Interchange Format. (2017).

Asynchronous I/O operations

As the exchange of messages between processes will have a great importance, non-blocking I/O could significantly reduce the deadlocks. Further investigation on this topic is needed, but a polling system might be an interesting solution. - Colomiets, P. (2018). Asynchronous IO in Rust – Sudo vs Root. [online] Sudo vs Root. Available at: https://blog.skcript.com/asynchronous-io-in-rust-36b623e7b965 [Accessed 27 Sep. 2018].

1.6 Legal, Social, Ethical, Professional and Security issues

Help: Here you should discuss any legal, social, profession and security issues that you believe may occur during the course of your project. It is not acceptable to write none in this box, all projects, regardless of focus will have to address issues in one, or more, of these categories. This is an extremely important part of your honours project to which there is no correct answer, this section must be fully discussed with your Honours Supervisor.

Example 1 : In the class register example above – there would be a Legal and Security issue with the gathering and storage of student data. There may be a social constraint as you may be relying on a user to have access to a specific technology. There will need to be consideration of user accessibility.

Example 2 : A 3D model design may have ethical considerations in its evaluation. What if your model made users feel nauseous. Social constrains may again be access to technology or accessibility issues.

Example 3 : You network design need to adhere to specific company policies. You would need to consider the possibility that your design could be wrong, compromising the company’s security.

In case of a dysfunction of a software that runs on the RTOS and/or a product running the RTOS, causing damage to material and/or people, and/or leading to the destruction of the system that runs the RTOS, the responsability does not lie with the programmer of the RTOS but rather with the physical person or corporation that have designed and/or created the product running the RTOS and/or the software running on the RTOS.

1.7 Project Plan

Help: This is the project plan as to how you will go about achieving the objectives of the project.

Example: In the class register example above the research plan may involve:

Collecting and analysing paper-based registers in a given class on five occasions.

Identifying the error rate average on these occasions

Researching existing automation techniques

Designing and implementing a mobile application that automatically records attendance in class.

Deploying the application in the class on five occasions.

Identifying the error rate average of the mobile application on these occasions.

Comparison of data and summary of findings.

Collecting and analysing research material and articles (including from professional-oriented blogs focused on programming) about RTOS

Gathering RTOS online code repositories to have an overview of the goals to reach and comparison elements

Learn and practice on a daily basis the Rust language through coding challenges websites

Designing and implementing the clock driver, display driver, IPC, physical memory manager and scheduler plus a few services

Designing and implementing and interpreter (containing a parser) for a Unix-like shell

Write unit tests for all the features and benchmark tests to measure the performances

---

1.8 Ethics Form

**You must include in your signed ethics form in this submission or you will not be able to continue the project.