**Team <Scentaur>**

**Ga Jun Young** (ga.young@ucdconnect.ie)

Royal Thomas (royal.thomas@ucdconnect.ie)

Zheng Ju (zheng.ju@ucdconnect.ie)

William Ikenna-Nwosu (william.ikenna-nwosu@ucdconnect.ie)

Use a splash page image here [optional]

Use *LaTeX* if you wish, but use the general spacing and font/style you find here (1.5 spacing, 12 point font for text, etc.).

Be sure to submit a PDF (not a .DOC file) as your report. Overall it should be 12 to 15 pages, including diagrams.

**1. Introduction**

Introduce your vision of the project and what you hope to achieve. What are the main goals of the project in your view? What should the typical user experience when using your system?

This project focuses on building a software which detects code smells. It gives a good opportunity for our team to combine different ideas together to work on a common goal. As the requirements of software engineering become stricter and the code environment is getting more complex, the size of the source code becomes more and more bigger and potential risks of projects are getting higher. So, finding these potential problems, solving them and simplifying the code are put on our calendar.

The software we build helps users to detect smells in their code. The process should be as simple and clear as possible and the feedback should be straightforward. The only thing the software requires from user is to upload their source files and choose what kind of code smells they want the software to detect. During the detection, the whole process is visible and real-time feedback should be shown to the user. Once the detection finishes, a report should be generated by the software and can be downloaded by users. In the report, what we want is all the code that is detected as potential code smells is highlighted and the type of that smell is also marked. What’s more, if possible, solutions are also provided to users.

Compare with other smell detection tools, we want ours to be simple to use, clear to see, quick to run and finally to be reliable. ------Ju Zheng

**2. Specification**

Provide a more formal specification of the planned features of your system here. What kind of user interface will it provide (web framework or application GUI)?

**3. Software Overview**

Provide a schematic view of your design here. A UML diagram might be useful. What interfaces are you putting in place to ensure different team members are working toward a coherent body of software elements are can be coherently integrated and tested.

**4. Major Responsibilities and Work Breakdown**

Provide a modular view of the work here, with an assignment of responsibilities to each team member. A Gantt diagram is useful here.

|  |  |
| --- | --- |
|  | Lead Designer - [Gajun][William] |
|  | Reflection API - [Jackie][Royal] |
|  | File Analysis - [Gajun][Royal] |
|  | Code Analysis - [Main lead: Gajun][Sub lead: Royal] |
|  | Smell Detection - [Everyone] |
|  | Bloater -> [Jackie,Will] |
|  | Abuser -> [Gajun, Royal] |
|  | Coupler -> [Will, Gajun] |
|  | Disp. -> [Royal, Jackie] |
|  |  |
|  | Interface Design - [Main lead: William][Sub lead: Jackie] |
|  | Smell Visualization - [Everyone] [Main lead: Royal][Sub lead: William] |
|  | GUI Interactions w/user - [Everyone] [Main lead: Jackie][Sub lead: Gajun] |

**5. Team Communication**

Provide an overview of the tools and methodology you will be using to ensure productive teamwork and a transparent flow of information. Will you be using any groupware applications to coordinate your efforts? How often will you be meeting as a team? How will you use resources like GitHub to maximum effect? How will you deal with disagreements or scheduling conflicts or misalignment of goals or plans?

For better communication between team members, several Apps such as Slack, Messenger and Discord, are used by team Scentaur. A Github Project Board is also used. Apart from these internet-based chat, face-to-face group meetings are arranged randomly each week. Usually, decisive agreements such as project interface design and user interface choosing, are reached during the face-to-face meeting.

In order to make the software development more specific, smooth and efficient, for general discuss, Messenger is used by us to share any ideas on the project. Questions are carefully discussed and addressed by group. Useful links are shared on Messenger. Discord is where we make voice meeting. Team Scentaur has 3 channels in total. General channel is used for general talk. Team members discuss advantages and disadvantages, agreements and disagreements of the project and real-time ideas, information or links found are shared immediately here. Also, group work and communication are pushed in progress simultaneously. Any suggestions on the development of the project are considered carefully. Another two channels are for sub-groups. Our 4-memeber team are divided into 2 2-member sub-groups during different period of development. These channels are used for sub-groups to work on specific modules of the project. Therefore, different modules can be developed at the same time.

Basically, real-time information is shared during the group discussion, but any of the useful or important information is also posted on Slack. Team Scentaur uses Slack to record the progress of the project. Any critical decisions or task assignment are pushed on Slack. Usually, team members will not talk on Slack. This makes Slack a clean and useful reference for team members to look into.

Team Scentaur uses Github to manage the project. Short-term and long-term goals of the project are set on Github Project Board. Useful information and progress are also posted on the board.

Team Scentaur holds team meetings frequently. Weekly meeting is fixed on Wednesday and other meetings are randomly held if necessary on campus. During break, daily talk on Messenger is compulsory. Team Scentaur needs to have the knowledge of everyone’s work progress. Voice meeting is random. Usually once every 3 or 4 days but team Scentaur will hold a voice meeting everyday when it is very necessary. ------Ju Zheng

**6. Concluding Remarks**

Provide a brief statement of your team’s philosophy here. Mention any special of noteworthy aspects of your approach to the problem. Highlight any risks you feel may impact the work, and offer mitigation strategies if necessary.

**Acknowledgements**

Every team member should contribute an equal effort to this report. Use this optional section to provide report credits, or to highlight a special contribution by a given team member.

**References**

List any bibliographical citations here [optional]