Individual Report
COMP204P - System Engineering

Marco Concetto Rudilosso

Personal Achievements

Team 9, the team I am part of, for the System Engineering module has been working on two Visual Studio Code plugins and one cloud system: Haskelly, providing IDE-like features for Haskell in VSCode and SEAT, a system which consists of a VSCode plugin and a cloud system, that will make scaffolding exercises easier for both lecturers and students.
As part of team 9, I contributed in various ways in the project:

- In the first 2 weeks of the project I have contributed to the general idea behind it and I have generated personas and use cases.
- SEAT idea was, in part, originated by a previous work I have done with an UCL lecturer, that inspired me in creating a tool to make the experience of scaffolding exercises smoother.
- I was the person in charge of doing research about the status quo of code completion in the Haskell world. This has led me to read about different tools, such us ghc-mod, ide-backend, stack-ide and intero. I have also read the source code of ghc-mod and intero, in order to understand how they worked and how they could be used in our project. Thanks to my work, we discovered Intero, which we now use to provide code completion successfully.
- I was one of the main contributor to the design of the grammar used in lab assignment files.
- I have created a parser in F# that will be used by SEAT in order to parse and extract information from a lab assignment file, which has a grammar similar to markdown, written by the lecturer. I decided to use F# for two main reasons: it was fully supported by Azure Functions that we plan to use extensively in our cloud system and it had a library called FParsec, that uses monadic parser combinators, that I knew from previous experiences in Haskell. The parser is already working and it can successfully extract a JSON file from a lab assignment file.
- I have helped debugging different problems with Haskelly, mainly on Windows.
- I have pushed the idea of integrating Stack with Haskelly and I have written code to make it happen.
- I have coded the first version of the website and I have continued to beone of the main contributor to it. I have written some of the content in it and participated to the design decisions about it.

Personal Assessment of Team 9 work

During the first part of the project, Team 9 was able to deliver a first version of Haskelly, which has been a success for our team so far. We have published the extension on Visual Studio Marketplace and we have got 650+ downloads. This has let us discover bugs that we have solved or we are going to discover, but also has made clear the need of an extension like Haskelly in the Visual Studio Code environment.
Intero as the backend used by Haskell has been a good choice, it provides an easy way to get code completions and it’s closely interlinked with Stack.
In order to get Intero working for a first alpha of our extension, we are spawning a process with Node, but due to the asynchronous nature of this process, doing so creates new challenges and problems. This decision was taken in order to have a first working product, but we are going to change this in future, with a system that makes asynchronous message passing easier, such as JSON RPC.

First class support for stack is a great choice, this tool makes it easy to manage and install libraries, compared to cabal that was considered one of the worst part of the Haskell experience (survey findings proving this here).
Using Azure built-in authentication in the SEAT Cloud System has enabled us to provide a fast and secure way to access our system for lecturers, that can now login using their UCL email with no need to create a new account on our platform. We took this decision because we wanted to have a smooth user experience for our users and we think that authentication is a key part of this.
Using Azure webapp and Azure functions has been part of our effort of creating a serverless architecture for our system: this is cost effective and decreases the time spent deploying our system and maintaining it.
The use of F# with FParsec for the creation of the parser for lab assignment files has been dictated by two factors: F# was the only language supported by Azure Functions to have a monadic combinator parsing library and the advantages of monadic combinator parsers, which benefits from type safety and development tools of the language in which they are implemented.
The next stages of our project will be:

- Finalising a new version of Haskelly that eliminates the problems that the alpha has in running on Windows and in giving code completions for modules imported with the “as” keyword.
- Creating SEAT webapp, with a main focus on how encryption of lab assignment files will work and how we will deliver analytics for lecturers.
- Creating test suites for both of our systems.
- Constantly iterate on our systems, in order to solve bugs and add functionalities.
- Make Haskelly more visible, so that a larger audience can use it and gives feedback on it. This could be accomplished by writing a launch post on different channels, such as the Haskell IRC and the Haskell subreddit.

Marco’s Evaluation:

General Overview

Marco has been involved in the creation of Team 9 systems since the beginning, coming up with main ideas about SEAT and pushing the use of stack in Haskelly.
He has been involved in most of the decisions about the development of the project, as it is expected by a Technical Lead.

Contributions:

- Researched on Haskell libraries which provide autocompletion
- Experimented with Intero and ghc-mod
- Created lab assignment files parser
- Contributed to Haskelly, debugging on Windows and promoting stack integration
- Contributed to the website
- Created Personas and use cases

Marks

- Quality of contributions: B
- Quantity of contributions: B

Skill analysis

Skill	Grade	Justification
Reliability	B	He accomplish the jobs given to him
Technical Expertise	A	He has a good knowledge of Haskell and he has picked up F# in a day to create the parser.
Communication	B	He is sometimes difficult to contact, but he is good at communicating the main goals of the projects.

Janos's Evaluation

General Overview

Janos has been critical as a team leader, thanks to his project management and communication skills. He has also done a very good job researching about SEAT cloud system, making an important impact on the direction of the technology that will power SEAT in the future.

Contributions

- Researched on Azure WebApp, Azure Functions and Azure Authentication
- Main promoter for a serverless infrastructure for SEAT system
- Wrote content for the website
- Set up the initial webapp for SEAT
- Set up authentication with UCL emails for SEAT
- Set up deployment for SEAT system

Marks

- Quality of contributions: B
- Quantity of contributions: B

Skills analysis

Reliability	B	He is really keen on his duties.
Technical Expertise	B	The work that he has done has proved a good level of expertise, but not an excellence
Communication	A	He has done a good job communicating with our clients and supervisors and he is a good speaker.

Marti's Evaluation

General Overview

Marti's contribution to the project has been critical due to his large contribution to Haskelly that has made possible to release a first alpha version of the extension. This result comes from different experiments that he has run due to its role of Chief Researcher. He has contributed to the style and the content of the website as Chief Editor.

Contributions

- Main contributor to Haskelly
- Added code highlighting to Haskelly
- Created Visual Studio Code interface to Haskelly
- Made Haskelly talk with Intero in order to get autocompletion
- Made Haskelly talk with stack in order to use: ghci, runhaskell, build and exec
- Made Haskelly test source code with an integration with QuickCheck
- Contributed to the website

Marks

- Quality of contributions: A
- Quantity of contributions: B

Skills analysis

Reliability	A	He has worked hard on the project and has often delivered important features to the team
Technical Expertise	A	His work on Haskelly has proved his expertise in the development of visual studio code extension.
Communication	B	He clearly states his ideas and he is easy to reach.