Web Based Game to Teach Functional Programming

CS310 - Project Specification

Joe Moore, u1917702
Department of Computer Science
University of Warwick

Contents

1 Introduction

In 2021, there currently exist a multitude of websites and tools with which to teach hopeful novices programming, the most popular being web-based sites like Codecademy. However, these web-based learning aids can suffer from setbacks. For example, they can often be not very engaging, and therefore inaccessible to younger users. Whereas learning new skills through games gathers growing enthusiasm among researchers and lecturers [2]. With child programmers who learnt the skill initially through games showing a stronger aptitude for the skill years later [4].

Secondly the feedback given by such websites is vague. Codecademy, for example, simply allows you to get the correct code after 3 incorrect attempts. This can stunt learning as often it allows progression with understanding. Code is not analyzed; it is

not explained to the user how their code could be improved specifically. Feedback is generalized and vague. Either your code works, or it does not. Revealing a solution outright like this is proven to have a negative effect on learning [3].

The aim of this project is to create a web-based teaching tool for a Functional Programming language which makes use of the interactive advantage of game learning. Accompanied by some specific feedback on code written, rather than a binary correct or incorrect, feedback system.

2 Objectives

This project has the main aim of providing a web based environment through which users can learn more about functional programming through the medium of games. This is achieved through use of a backend-frontend web design, where code written by the user is passed ot the back end and run through an interpreter and compiler, resulting in two outcomes. If the code is correct, i.e. can be correctly interpreted then it run and the user is able to use the commands they have written. Otherwise the code is passed through analysis functions to return more personalised feedback on how they can improve.

The website will have a coding area into which the user types. Their code upon submission is then converted into *JSON* and passed to the backend. Feedback is also returned via *JSON* and then displayed in the front end. The front end will be built through use of react and the back end with use of Python Flask. This means the interpreter will also be built in Python.

The success of this project will mainly be gauged via feedback from users. A number of users who have little to no understanding of functional programming languages will be asked to use the tool and then take a test to identify the improvement of their knowledge. The interpreter and code feedback can also have their success analysed by more objective automated testing.

3 Formal Project Requirements

The first consideration in the design of my project requirements was the research conducted into the *MUSIC* model of learning [1]. This has shown five key conditions for successful student learning.

1. Feel empowered by having the ability to make decisions about some aspects of their learning,

2. Understand why what they are learning is useful for their short- or long-term goals,

3. Believe that they can succeed if they put forth the effort required,

4. Are interested in the content and instructional activities, and

5. Believe that others in the learning environment, such as the instructor and other students, care about their learning and about them as a person

(Jones, 2018, p.9)

This is therefore played a significant role in the design of the formal requirements for this project. The multiple solutions to problems will allow the satisfaction of point one. Explanations of the goals of the project before commencing will enable the user to understand the primary goal - the improvement of their functional programming skills. As well as explanations of the real world applications of each piece of programming they learn. Increasing difficulty of levels as well as responsive code feedback should allow belief in themselves that they can succeed, if effort is put in. Objective 4 is out of the hands of this project. However, a clean functional web environment should allow for the fulfillment of point five.

Another key consideration in the compilation of this project's requirements was Thomson's research into the teaching of functional programming [5], which showed that it was more successful when through the medium of problem solving. This provided further support to the initial research which on game based learning, whilst grounding it more to this specific project by linking it to functional programming.

The final consideration is taken from Subramanian and Budhrani's research into how project based learning is the most beneficial to the *MUSIC* model. Thus the property of the project by which the user creates a game over multiple puzzles should enable maximum learning potential.

Thus with these considerations in mind the following requirements have been defined.

3.1 Functional Requirements

- The backend of the website should successfully interpret and run all of the code written by the user.
  - This includes the handling of errors. These should be properly handled and feedback to the user. No code should be able to break the website.
  - The game should work if the user inputs correct code and the use of arrow keys etc should execute the code they haven written in order to play the game.
- The type-level system designed will use an Embedded Domain-Specific Language in order to make it accessible.
  - It will be written according to lambda calculus grammar but with the inclusion of variables and types.
  - Use the type-level model to type-check inputted code and provide feedback specifically about type issues.

3.2 Nonfunctional Requirements

- The teaching tool should allow students to fulfil the *MUSIC* model in their learning.
  - Present the user with problems with multiple solutions so as to provide them with significant choices and place them in control of their learning.
  - The content should increase in difficulty with later tasks being possible but fairly tricky. This should provide a challenging learning environment and such improve knowledge whilst allowing the user to be certain they have built up the skills in previous tasks to attempt the current one.
  - Explanations as to the usefulness and applications of different aspects of functional programming should be explained with each task.
- The total learning project should result in the creation of an overarching project, so as to facilitate *project based learning*.
- Students should feel that their understanding of functional programming has improved by engaging with the teaching tool.
- The code feedback should be helpful and understandable to students who are unfamiliar with functional programming.

4 Methodology

4.1 Software Development Methodology

The employment of an incremental development approach is the most appropriate for this project. The increments being large components of the project such as, a functioning lambda calculus interpreter, the react front end of the website, the backend communication with the front, etc. Once individual components have been created and successfully tested they will be compiled together into the larger project. This therefore is an adaptation of spiral methodology. The waterfall development technique is what will be used in the development of each of the projects components.

4.2 Task Management

The classifications of tasks into 5 distinct categories will reflect the cycle of the development of a task in each development iteration. The classifications are as follows:

- **IDEA** When a task is in concept form, such that it could be developed into a full task. This happens when it is deemed to be beneficial to the project as a whole or any of its components.
- **PLANNED** The task has been fleshed out enough to be worked on. Some key test cases can now be devised
- **DOING** Task is currently being completed
- **REVIEW** The task has been completed but not reviewed yet. This means for example enough test cases have not yet been created and executed.
- **COMPLETED** A task that has been done and reviewed successfully. It requires no further work.

This system is used for all tasks and allows for mor efficient use of time by focusing on tasks that are deemed more useful for the overall project. It will also help to keep track of the whole development process and therefore lead to easier documentation writing later in the project.

4.3 Evaluation and Testing

The website must be function with a reasonable speed. Although, that seems vague placing a time limit on it this early in the project is unwise since the scale and design is still very ambiguous. Therefore I will ask if the user feels the website is responsive enough to enable learning. Errors must specify exact issues that have been encountered for example,

'Type error line 4'

is an unacceptable error whilst

'On line 4, the function 'x' is of 'y' type. We expected this type.'

is acceptable.

Automated testing can be used for the interpreter and the error handling to ensure it catches edge cases and interprets fast enough. Learning effectiveness will be evaluated and concluded based on the results of interviews with the users about their experience with the system, alongside the results of the test they participate in to test their knowledge of functional programming after using the system.

5 Timetable

Weeks 1-2

4 October

Draft and write project specification. Research further into learning techniques and start to put together an initial basic React website.

Weeks 3-4

18 October

Work further on the website creating the UI etc. such that the user can type code but without a backend.

Weeks 5-6

1 November

Flesh out the game by developing the custom functional programming language that will be used and design the levels on paper.

Weeks 7-8

15 November

Create the interpreter for the language designed. This will be done in Python

Weeks 9-10

29 November

Start work on code feedback algorithm.

Christmas Break

13 December

Catch up on any of the tasks that I fell behind on to ensure all the work that was meant to be completed by the end of week 10 was.

Weeks 1-2

10 January

Finish the code feedback code in Python.

Weeks 3-4

24 January

Connect frontend to backend, bug test and ensure the website is fully working.

Weeks 5-6

7 February

Start writing the initial report. This will focus on the introduction and the details of development stages.

Weeks 7-8

21 February

Finish writing the development sections of the report and start on the conclusion.

Weeks 9-10

7 March

Finish the report and analyse the success of the project from the tests and interviews.

Easter Break

21 March

Review report and make any necessary changes.

Weeks 1-2

25 April

Finish and submit the final report.

6 Resources and Ethics

6.1 Resources

VSCode is going to be used to create the website using Python Flask. This will be primarily done on my own laptop, a DELL Inspiron running Ubuntu Linux. The website will be hosted on a third party cloud service yet to be decided. The documentation is written in VSCode as a Latex file. GitHub is being used for version control.

6.2 Legal, social, ethical and professional issues

Any personal data under law will be handled in accordance with GDPR. With users having ability to see, delete and amend their personal data stored by the application. No other people besides me and my supervisor are involved in the creation of this project so there are no issues to consider then. Ethically it is important all data is stored securely in order to protect any children who use the tool. This is a major safeguarding issue and must be adhered to. From a professional and ethical standpoint it is important to ensure that the content taught is appropriate and more importantly correct for all ages and not of poor quality. Any evaluation performed may involve students so that should be done in an appropriate manner and in accordance with relevant safeguarding principles.

References

[1] B. D. Jones. Engaging second language learners using the music model of motivation. *Frontiers in Psychology*, 11:1204, 2018.

[2] M.-C. Li and C.-C. Tsai. Game-based learning in science education: A review of relevant research. *Journal of Science Education and Technology*, 22(6):877–898, 2013.

[3] H. F. O'Neil, R. Wainess, and E. L. Baker. Classification of learning outcomes: evidence from the computer games literature. *The Curriculum Journal*, 16(4):455–474, 2005.

[4] M. Pivec. Editorial: Play and learn: potentials of game-based learning. *British Journal of Educational Technology*, 38(3):387–393, 2007.

[5] S. Thompson. "where do i begin?" a problem solving approach in teaching functional programming. *International Conference on Principles and Practice of Declarative Programming*, pages 323–334, 1997.