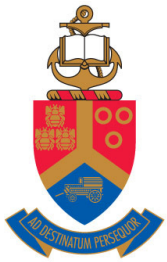

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Project Tender

Project:
Flowchart Planning and Simulation Tool

Client: Willem van Heerden

Team: <undecidables>

Elzahn Botha *13033922*
Jason Richard Evans *13032608*
Renette Ros *13007557*
Szymon Ziolkowski *12007367*
Tienie Pritchard *12056741*
Vivian Venter *13238435*

Department of Computer Science, University of Pretoria

Date: May 3, 2015

Contents

1 The Team

1.1 Elzahn Botha

1.1.1 Interests

- Video Games
- Game development
- Programming
- Anime

1.1.2 Technical Skills

- I have already been exposed to Java and been working with it for about a year now

1.1.3 Past Experience

- I have experience in writing programs and systems in Java

1.1.4 Non-Technical Strengths

- Hard and dedicated worker
- Always ready to learn new technologies and languages
- Functions best under pressure

1.1.5 Why I want to do this project

This project seemed like it would be a rather interesting project to do as well as the fact that it would be great exposure to a whole new side of IT that I have not yet had the privilege of delving into. I believe that by doing this project I will get valuable exposure that can later help in other projects that I might attempt. This project also seems like it would be a good challenge since I am more comfortable with smaller projects and systems and by doing a larger project such as this it will help me broaden my range of capabilities. Lastly due to the size of the project and the strict time line the pressure will be greatly increased helping me to work at my best without losing interest with the project.

1.2 Jason Richard Evans

1.2.1 Interests

- Music Enthusiast
- Software development
- Adventure Sports

1.2.2 Technical Skills

- C++ and Java
- Web Development

1.2.3 Past Experience

Assignments and projects completed for University purposes. A lot of exposure with writing Java applications and servers.

1.2.4 Non-Technical Strengths

- Positive outlook on life
- Hard worker
- Good team worker
- An eager learner

1.2.5 Why I want to do this project

The reason I started with software development is to make life a bit easier for everyone, and this project fits right into that goal. Creating a software product to teach others an important part of computer science logic more easily and austhetically pleasing will help thousands of students.

1.3 Renette Ros

1.3.1 Interests

- Reading
- Playing Games
- Painting
- Puzzles and problem solving
- Programming
- New interesting technologies

1.3.2 Technical Skills

- Java
- C++
- Web Development
- HCI - User Experience and User interface design
- XML, XML Schemas and related technologies.
- Good at identifying possible problems and debugging code

1.3.3 Non-Technical Strengths

- Hard Worker
- I don't like to do things halfway
- I learn new technologies easily
- I'm very enthusiastic about things I'm interested in.

1.3.4 Why I want to do this project

I want to do this project because creating an application like this will be very interesting because it is completely different from the things we have done up to now. I also believe it is important to have flowcharting software that forces students to create correctly structured flowcharts but is still easy to use since I got frustrated a lot with the flowcharting software we used in COS151.

1.4 Szymon Ziolkowski

1.4.1 Interests

- Networks
- Security
- Computer hardware and electronics
- Paintball
- Video games

1.4.2 Technical Skills

- C# and Java
- Web development
- SQLite, MySQL and SQL Server

1.4.3 Past Experience

I have no past experience that might be relevant to the project.

1.4.4 Non-Technical Strengths

- Don't give up easily
- Helpful

1.4.5 Why I want to do this project

1.5 Vivian Laura-Lee Venter

1.5.1 Interests

stuff

1.5.2 Technical Skills

stuff

1.5.3 Past Experience

stuff

1.5.4 Non-Technical Strengths

stuff

1.5.5 Why I want to do this project

stuff

2 Project Execution

2.1 Development Methodology

Our team will be using the **agile software development methodology**. The agile manifesto states that we find value in the following four points:

- **Individuals and interactions** over processes and tools.
- **Working software** over comprehensive documentation.
- **Customer collaboration** over contract negotiation.
- **Responding to change** over following a plan.

Although we find value in the items on the right, we value the items on the left, emphasised in bold, even more.
In agile development, the testing phase, which is usually separate from development in other methodologies such as Waterfall, runs concurrently with development. This increases the *quality* of the product and means that at any point in the development phase there will be a *working model* available with new features and functionality being added only after they pass vigorous testing.
The agile methodology has a proven low project schedule risk and the ability to respond to change, from the client or development team, quickly.

2.2 Informing the Client

Our team will have an active, regularly updated, wiki on GitHub and we will also use the GitHub issue tracker with Milestones to keep track of work and goals. We will meet with the client on a weekly or bi-weekly basis to get feedback and discuss progress and problems.

2.3 Initial Ideas on Solving Some Technical Challenges

- We will use the strategy design pattern to create different classes for the different types of blocks available.
- Each block-class will have an execute function that performs the required action.
- The flowchart will be implemented as a directed graph with each block holding a reference to its successor(s)
- The program will use the Model-View-Controller architectural pattern to separate the interface and the functionality and to simplify flow-chart error checking.
- If required, flowcharts will be as stored xml and the program will be able to load these xml files.

2.4 Technologies We will Use

We plan to use Java with JavaFx to build this application since all our team members know Java and some of us already have experience with JavaFx. It is also platform independent so it can be run on both Windows and Linux.

We will use XML to save flowcharts so a student can save or submit their work. The reason for this choice is that XML is platform independent and can easily be parsed.

2.5 What Will The Client Receive

On completion of the development cycle the client will receive the following:

- The functional application
- The Program source code
- Code Documentation
- Requirement and design documentation
- A user manual for the program.