VICTORIA UNIVERSITY OF WELLINGTON
Te Whare Wānanga o te Ūpoko o te Ika a Māui

School of Engineering and Computer Science
Te Kura Mātai Pūkaha, Pūrorohiko

PO Box 600
Wellington
New Zealand

Tel: +64 4 463 5341
Fax: +64 4 463 5045
Internet: office@ecs.vuw.ac.nz

Preliminary Report

Zoltan Debre

Supervisor: Dr Stuart Marshall

Submitted in partial fulfilment of the requirements for
Master of Computer Science.

Abstract

My project is about building an online publishing tool prototype, using code editors and step by step instructions to present programming challenges or solution for a computer science problem.

The prototype has two parts, an administration area, where the content creator can build a tutorial, and a player tool, where the recorded steps would be presented. This player should be embeddable in any blog post or website.

Contents

Figures

Chapter 1

Introduction

My project is about building an online publishing tool prototype, using code editors and step by step instructions to present programming challenges or solution for a computer science problem.

The prototype has two parts, an administration area, where the content creator can build a tutorial, and a player tool, where the recorded steps would be presented. This player should be embeddable in any blog post or website.

My project primary target user is the creator, who composes a new tutorial. The creator can be a teacher, or an open source project owner, who would like to introduce his tool or code.

The secondary user is the consumer, who want to learn or know more about a problem or a coding solution.

1.1 Background

We all have the unstoppable desire to learn. We are keen to know more about the world around us, about our hobby and our profession. In software development, in computer science, the knowledge is essential, it is the key to succeeding. Reading, studying, sharing. An infinite loop of collecting and adapting new practices.

In information technology, especially in programming languages, writing blog posts, creating static, step by step tutorials are a popular way to share or learn something new. Producing and sharing the content is easier nowadays, but still required more effort from the creator, when they want to deliver easy to understand, high quality tutorials.

Creating interactive tutorials are appealing, but the production cost is much higher. Recording a video tutorial or especially updating it is time-consuming and involves more effort from the creator.

I think an ideal solution would be a healthy mix of static and dynamic contents. Where the learner can read instructions but meanwhile can watch the steps in a code editor, in a more realistic environment.

Chapter 2

Using this document and the `vuwproject` style

If you are writing an MSc or PhD thesis you should *not* be using this style. Instead use `vuwthesis`, which is based on the book style, and conforms to the VUW thesis rules. The thesis style is rather different from the project report style.

This document is formatted using a local (to ECS and MSOR at VUW) style file. When you write your project report you should be very careful when changing the beginning. The document class settings should read:

```
\documentclass[11pt
              , a4paper
              , twoside
              , openright
              ]{report}
```

The options to the document class specify that:

- 11pt font is to be used for the main body text,
- we will print on A4 paper,
- we will use duplex (two-sided) printing,
- we want chapters to start on a right-hand page.

The opitons you supply to the `vuwproject` style will depend upon what you are using the style for.

2.0.1 Specifying the details

The `vuwproject` style sets up the front page properly, and provides various commands allowing you to specify the author, title, supervisor or supervisors, the school from which the report is being submitted and the degree that the report is being submitted for. The style has deliberately been designed to do as little as possible. This means that your document can easily be re-formatted as a technical report, or for submission to a conference or journal by using the appropriate style.

It is also possible to use the style to easily produce documents on a stand-alone computer where your LaTeXinstalltion might not have all of the files and fonts available to machines within ECS or MSOR.

Most of the options to the vuwproject style are currently a simple choice and there's a default that will make it obvious if you do not make a choice.

Use one of the following options to use fonts available on ECS/MSOR machines or to use images that imitate them (assumes you have copies of the images)

- font
- image

Use one of the following options to set the school,

- ecs
- msor

Use one of the following options to choose a pre-defined degree,

- bschonscomp
- mcompsci

or use this command to use an explicit degree or diploma name

- \otherdegree{DEGREE OR DIPLOMA NAME}

So, for example, to submit a report for the Master of Comp Sci degree, which the style knows about, from within ECS, using the images, you'ld ensure the vuwproject line options looked like:

```
\usepackage[image,ecs,mcompsci]{vuwproject}
```

whereas for a degree from within MSOR, when creating the final version on an ECS or MSOR machine where you have access to the fonts, you would use these options

```
\usepackage[font,msor]{vuwproject}
```

and add the other degree's name using this command

```
\otherdegree{DEGREE OR DIPLOMA NAME}
```

To specify the supervisor or supervisors use either of the following commands in the preamble.

- \supervisor{The Supervisor}
- \supervisors{Super 1 and Super 2}

If you fail to set any degree or supervisor, or the school, then the front page will report this.

The vuwproject style also sets the default font to be Palatino, using the mathpazo package. Palatino is one of VUW's 'offical' fonts, and is the font used for the heading on the front page. The mathpazo package also typesets maths in a style which suits Palatino.

2.1 Copying the style

If you want to write your project report away from VUW you will need to make your own copy of the `vuwproject` style.

You can find out where the original lives by reading the messages that LaTeX prints when it is run.

Alternatively, you can down load a copy of the `vuwproject` style from the ECS webpages.

Any changes made to your own copy of the `vuwproject` style will not be reflected in the original, and *vice versa*. Hence it makes sense to leave this as it is, and use a local style file for your own definitions.

Chapter 3

Some LaTeX hints and tips

LaTeX is a very good tool for producing well-structured documents carefully. It is very bad tool for banging things together in a rush and panic.

3.1 Floats

One perennial problem with LaTeX is its treatment of *floats*. Suppose you have a figure or table which you want to include in your document. Where should it go? Traditional typesetting practice is to put these in some convenient place, such as the top or bottom of the current or next page, or at the end of the section or chapter. LaTeX adopts a similar strategy, and allows floats to “float” away from where they were defined. You can give a hint about where you want the figure, but LaTeX may move it. Sometimes this is fine but sometimes you may want to have more control and insist that a float goes *here*. Anselm Lingau’s float package gives you this flexibility. For example, the following figure is an example of a non-floating float:

δ	a	b	Λ
S_1	$\{\}$	$\{\}$	$\{S_2, S_5, S_{10}\}$
S_2	$\{S_3\}$	$\{\}$	$\{\}$
S_3	$\{S_4\}$	$\{\}$	$\{\}$
S_4	$\{S_3\}$	$\{\}$	$\{\}$
S_5	$\{\}$	$\{S_6\}$	$\{\}$
S_6	$\{\}$	$\{S_7\}$	$\{S_8\}$
S_7	$\{S_6\}$	$\{\}$	$\{\}$
S_8	$\{S_9\}$	$\{\}$	$\{\}$
S_9	$\{\}$	$\{S_8\}$	$\{\}$
S_{10}	$\{S_{11}\}$	$\{\}$	$\{\}$
S_{11}	$\{\}$	$\{S_{10}\}$	$\{\}$

Figure 3.1: The transition function of an NFA with Λ transitions

On the other hand, Figure **??** is a floating float.

You can define different types of new floats, and you can have tables of them in the contents pages.

δ''	a	b
T_1	T_2	T_3
T_2	T_4	T_5
T_3	T_6	T_7
T_4	T_8	
T_5	T_{10}	
T_6		T_{11}
T_7	T_3	
T_8	T_4	
T_{10}		T_5
T_{11}	T_6	

Figure 3.2: The transition function of an FA to accept the same language.

3.2 URL's

Use `\url` from the url package to typeset URL's. Just using `\texttt` or `\tt` does not work:

- `\texttt{http://www.mcs.vuw.ac.nz/˜neil/}`
- `\url{http://www.mcs.vuw.ac.nz/˜neil/}`

Give:

- `http://www.mcs.vuw.ac.nz/ neil/`
- `http://www.mcs.vuw.ac.nz/~neil/`

If you use the hyperref package then you can produce PDF files with clickable hyperlinks using `\url`.

3.3 Graphics and LaTeX

LaTeX offers rather poor support for the inclusion of graphics. There are lots of ways to include pictorial material in LaTeX, all of which are deficient in some way or other. Look at [?] for a description of them. If your document does need to have pictures in it it is worth thinking about what is needed *before* you generate the pictures.

3.4 The bibliography

You should build up your bibliography as you go along. Trying to get the details of the bibliography correct at the end of the project is hard work. Make sure that you record all the relevant details. Beware that material on the internet is likely to change very rapidly. If you are going to include material which is only available on the internet, then you should probably include in the reference the date on which you obtained the document.

3.5 Run LaTeX, run

LaTeX builds up information about your document for the table of contents, references and so on at each run. This means that, for example, the table of contents is really the table of contents of the previous compilation. You may need to run LaTeX two or three times to let it catch up with itself. If you have cross references within your bibliography (for example two papers from the same collection, such as [**?**, **?**]) you may need to run BibTeX more than once.

It is also possible that the table of contents file has garbage in it, and will prevent the document from being compiled. This may happen if you have had to abort compilation, due to a bug in the source file. If this is the case then removing the `.toc` file will usually solve the problem. You will have to fix the original bug, of course.

3.6 Find out more by...

You can find out more by:

- reading any one of a number of books, such as [**?**, **?**]. The VUW library has copies of these;

- visiting the Comprehensive TeX Archive Network (CTAN) at `www.ctan.org`;

- typing `latex` into Google.

It is *highly unlikely* that you are the first person who ever wanted to do what you want to do with LaTeX. Therefore it is likely that someone has already solved your problem: the real key to using LaTeX well is to make effective use of what other people have done.

3.7 Summary

In this chapter we explained some things about LaTeX.

Chapter 4

Conclusions

The conclusions are presented in this Chapter.