Python and Python Web Applications

ICT Enterprise Project

Supervisor: Catherine Woods

Submitted By: David Murphy (TL\_KCOMP\_GY6)

<<Submitted on: 28th April 2014>>

Table of Contents

[1 Introduction 1](#_Toc379479797)

[2 Requirements Specification 2](#_Toc379479798)

[2.1 Project goals 2](#_Toc379479799)

[3 Project Plan 3](#_Toc379479800)

[4 Python Research 5](#_Toc379479801)

[5 Comparison of Python with Java 6](#_Toc379479802)

[6 Python Web Application Research 7](#_Toc379479803)

[7 Domain Application of Python and Python Frameworks 8](#_Toc379479804)

[8 Python/Flask Web Application 9](#_Toc379479805)

[8.1 Introduction 9](#_Toc379479806)

[8.2 Analysis 9](#_Toc379479807)

[8.3 Design 9](#_Toc379479808)

[8.4 Implementation Increments 9](#_Toc379479809)

[8.5 Testing Strategy 9](#_Toc379479810)

[8.6 Conclusion 9](#_Toc379479811)

[9 Project Conclusions 10](#_Toc379479812)

REFERENCING – for citing

BIBLIOGRAPHY – other resources not cited

# Introduction

Introduction goes here.

# Requirements Specification

## Project goals

The goals of this project are as follows:

* Acquire programming skills in the Python programming language
* Acquire programming skills in a Python web framework (Flask)
* Compare and contrast Python with another high-level programming language (Java) in terms of programming paradigms, syntax, standard libraries, domain applications and web development
* Perform analysis, design, implementation and testing of a web application programmed in a Python/Flask environment

# Project Plan

The following is a detailed breakdown of the tasks required to achieve the project goals outlined above:

|  |  |
| --- | --- |
| **PROJECT GOALS** | **TIMELINE - Wk1 – Wk12 (college term)** |
| **Acquire Python programming skills**  Tasks to complete:   * study official Python.org documentation * complete official Python.org tutorial * complete ‘Dive Into Python’ online course   All research conducted in this phase will be documented in this section accordingly. | This research will be carried out continuously over Wks1 – Wks6 |
| **Acquire skills in a Python web framework (Flask)**  Tasks to complete:   * read Flask official documentation * complete Flask online tutorial * complete other online tutorials   All research conducted in this phase will be documented in this section accordingly. | This research will be carried out continuously over Wks3 – Wks7 |
| **Comparison of Python with another high-level programming language (Java)**  A critical comparison of Python with Java in terms of the following headings will be presented:   * programming paradigms * features of the language * syntax and semantics * standard libraries * domain applications * web development | Work in this section will be carried out over Wk5 – Wk8 |
| **Analysis, design, implementation & testing of a web application programmed in a Python/Flask environment**  Summary of implementation  The application to be developed will allow a client-side user to log on to a web application, select from a list of currencies and input a monetary amount to be converted from one denomination to another.  When instructed, the web application will call into a remotely hosted web service with the required input data, perform the necessary business logic calculation and return data to the web application for presentation in the web application.  The web service may connect to a database to retrieve currency rate.  Technologies to be employed   * The web application and web service will be developed in Python using the Flask web micro-framework for web development. * The web service will be remotely hosted on a suitable hosting service, such as Heroku * Analysis and design of the application will include analysis of the problem and a description of the solution design, and will also include UML diagrams as required.   Product delivery  Due to the nature of the research into Python and Flask being carried out the application will be developed incrementally, with each increment adding further functionality and/or improvements to the application until a satisfactory solution is implemented.  Testing strategy  A testing strategy will be designed and the application tested accordingly in conjunction with each increment. | Work in this section will be carried out over Wk3 – Wk11  Analysis & design to be completed by Wk4  First increment delivered Wk4  Weekly increments thereafter until satisfactory implementation is achieved  Testing strategy to be completed by Wk4, and carried out in conjunction with the various increments  Final submission 28/04/2014 |

# Python Research

# Python Web Framework Research

## Introduction

## Python Web Frameworks

# Comparison of Python with Java

# Python/Flask Web Application

## Introduction

This section describes the analysis, design, implementation and testing of a web application developed in the Python programming language using the Flask “microframework” for web development.

The implementation was designed such that it would make use of the technologies studied as part of this project.

The web application to be developed is a currency conversion application. The application will allow a client-side user to securely log on through a browser to a web application, hosted locally. The web application will present a selection of currencies to the user, and allow user-input of a monetary amount, with the user selecting the base currency the converted currency. The user submits the information, and a currency conversion is performed. The converted amount is displayed to the user.

The application will call out with currency and amount data to a remotely hosted RESTful web service. The web service will carry out the necessary business logic and return the resulting data to the web application for presentation to the user.

The design for this project has been kept deliberately simple in order to create a suitable working product commensurate with the new technologies explored earlier. It is considered that the proposed implementation will demonstrate a significant knowledge of Python application programming, Python/Flask web application development and remote API/web service deployment to a suitable host in the cloud. The focus is not on what the system does, but rather *how* it does it.

## Technologies

The web application will employ the following technologies in development:

* client-side web application: to be developed in Python employing the Flask web framework
* server-side web service: to be developed in Python employing the Flask web framework. The web service is to comply with the requirements of REST (REpresentational State Transfer) architectural principles
* remote hosting: the web service will be hosted on Heroku Cloud Application Platform. Heroku provides support for Python applications developed with Flask web framework
* Git/GitHub: a GitHub remote repository will be used for the duration of the project. Source code and documentation will be pushed to the repository such that a full history of incremental development will be recorded. Heroku also requires the use of Git and GitHub for the deployment of applications
* Notepad++ has been chosen as a simple and lightweight IDE for the development of Python source code for this project

## Product Delivery

The web application will be developed incrementally. It is considered that this is an appropriate method of delivering the final product, given the concurrent research into the various technologies that is taking place.

The first implementation will be a very simple version of the application, locally hosted, performing minimal calculation. It is intended that increments will be delivered approximately weekly, with each increment adding further functionality/usability/style until a satisfactory solution is implemented which achieves all of the requirements detailed in the following section.

## Analysis

### Requirements Hierarchy Chart

### User Requirements

The requirements of the web application are identified as follows:

* the web app will allow a new user to create a user account for logging into the application
* the web app will allow a previously registered user to securely log in to the application
* the web app will carry out a currency conversion on a specified amount between currencies of the user’s choice

### System Requirements

## Design

## Testing Strategy

## Implementation Increments

## Final Implementation

## Conclusion

# Project Conclusions