CompSci 734 Assignment 2 Report

This report mainly focuses on two things, there are my experiment of developing the assignment, and my view on the article "software architectures based on plug-ins".

In this assignment, we were required to develop a plug-in system that helps a student demonstrate their understanding of any topic in one of the undergraduate courses. This assignment is developed by using the C# programming language. The assignment can generally separate into two parts, one is the delivery of the question; another one is answer marking.

The first obstacle of this assignment is choosing a topic from one of the undergraduate courses. The restriction is that the topic chosen cannot be similar to other classmates. Originally, I put my attention on binary tree searching and dynamic programming. Unfortunately, there was a topic that overlap with other mates, but eventually, I decided to do a color spacing problem from CompSci 373.

The Second thing that I need to do is to design questions that can adequately represent the topic and the knowledge that it is trying to delivery to the students. By reviewing the Color spacing problem, I found that RGB, CIE XYZ color spacing, and HSV (Hue, Saturation, Value) color spacing can be meaningful questions, then I evaluated some color spacing calculation questions. The aim is that by doing these questions, students will understand how to calculate the RGB, CIE XYZ, and HSV color spacing distance problem when given two different color coordinates in 3D space and know how to convert RGB coordinates into HSV values.

The late part of this assignment is programming. Develop a plug-in that would work with individualized assessment framework Dividni. This assignment has a specification that is an HTML file, and I believe the Plug-in system is the C# file that we need to develop. The HTML file is there for designed questions delivery. The C# plug-in file is designed for operating with the HTML file. The mechanism is that C# gets a student ID number from the server, for that particular ID number, the C# plug-in system will randomly generate some parameters for each question so that each student ID will have different parameters for the questions. This allows students to follow academic integrity; also, in the process of problem-solving, students tend to discuss the question together to find out the how to solve the question and to evaluate their understanding of the color spacing.

Parameter generation is not enough to complete the plug-in system. I also developed algorithm functions to compute the answer for each question when given random parameters. Then we pass these answers to the server. The server will then compare the computed answers with student's answers on the answer sheet that they handed in and generate a marking result text file, which shows how many questions they got it right — this how we achieve auto-marking facility.

The point of this assignment is not about how to do programming, it is about to put ourselves in the perspectives of teacher and students. As a teacher, we need to know what questions can strengthen the knowledge for the students. As a student, it is essential to utilize our knowledge in a practical situation.

View on "Software architecture based on plug-ins"

This article provides general abstraction of pure plug-in software. A software architecture where everything is a plug-in.

This concept is novel, but it also comes with advantages and disadvantages.

Pros:

- Scalability: the application can be dynamically extended to include new features. As I implement my questions as many as I can do into the HTML file.

- parallel development: since features can be implemented as separate components, they can be developed in parallel by different teams.

- simplicity: a plugin could typically have one function. The developers can put their focus more on one task. While doing the assignment, I could focus on designing my questions and getting the solution for that question.

Cons:

Security: the architecture can be vulnerable and easy to be hacked. Security cannot be assured all the time. If vicious user gets my source cs plug-in file, then he/she could change my question or perform vicious actions.

Complexity: each plugin can work separately when they are operating. However, if installing to many plugins, the interaction between plugins can cause conflict, slow down your machine running performance, and cause bugs.