Professor Choi

COEN 241: Cloud Computing

PoS (Program of Studies) Validator

Project Midterm Review

Team members

Name	Email
Amanda Taylor	a1taylor@scu.edu
Gagan Gupta	ggupta2@scu.edu
Xin Guan	xguan2@scu.edu
Xiheng Yang	xyang10@scu.edu

Project Description

Introduction

This is a web application that targets to determine whether a student's tentative program of studies (PoS) is valid. The PoS includes the list of courses a student plans to take in order to successfully complete a certain program. There are usually many constraints on the courses (usually determined by the department), including but not limited to: course prerequisites should be ful-filled for the courses listed; total course credits should exceed the minimum required; certain courses are required to be in the list, while others are elective courses.

The task to validate a PoS is not only tedious but also time-consuming. It is very common for students to make small changes to their PoS, only to realize that such small changes have resulted in an invalid PoS later, which could pose a big problem especially if this is only discovered close to the graduation date. Moreover, it is also a lot of responsibility for the department assistants to validate each student's PoS, given its importance and difficulty.

Our application aims to alleviate such pain by providing a service to automatically validate the PoS. Here's a list of its functionalities as well as possible extensions if time permits:

- Take the list of courses as input from the student (user) from a web interface
- Validate the list of courses according to predefined rules and existing info of courses
- Let the user understand the reason if the PoS is invalid
- [Possible extension] Suggestions on courses to add / drop
- [Possible extension] Let admin to change rules / course info

Design

Tentatively, we plan to create virtual machines on public cloud providers (possibly AWS) to build the servers. It is also possible to use containers for some of the parts instead of VMs. We will also use virtualized storage to store course information and their relationships. If time permits, we could also use MapReduce / Hadoop to batch process a large amount of data. Using cloud technologies will allow for this project to be both flexible and lightweight for its deployment. Computational and storage resources can be dynamically allocated allowing for user growth to become a natural process.

The overall architecture of the the application is as follows:

- A server S1 is used to present a web page frontend to the users, and to get the user inputs. It will send the inputs to server S2 (possibly as JSON arrays) and display the returned validation results.
- Another server, S2, is the logic part that validates the PoS. It should take the input from S1, and query for necessary information from S3, in order to determine whether the given PoS satisfies the predefined requirements.
- The server S3 is used to interface with a cloud database, where information of the courses (credits, prerequisites, etc.) are stored. It should provide a set of APIs to query for certain information (e.g. the prerequisites of a course, courses that have a certain course as pre-requisites, etc.) and return the results (potentially with RESTful API and JSON objects). It will also be necessary for the students to initially input all the courses they've taken in the past for course dependency checks and all of this information will be stored on this server.

Tentative Task Assignments

Server S1 and frontend	Xin Guan, Amanda
Server S2 and logic part	Amanda, Xiheng Yang
Server S3 and database	Gagan