Schedule Generator/Planner (Team H)

This project is intended to work similarly to the semester schedule planning feature available on ISIS, but as a standalone component. The ISIS planner allows for courses to be added to a student’s list of intended classes for a semester (out of all of the available courses for that semester) and it generates a list of possible schedules based on known class times and dates. The student can then choose which of these schedules best fits him or her, and that choice becomes their new schedule for the semester. Currently, this feature requires a student to log-in to ISIS, navigate to the enrollment tab, and launch the schedule planner—which has a completely different UI from ISIS—in another window.

The goal for our project is to improve upon this planner by making it the main feature on a webpage, available to desktop and mobile users alike. Requirements include designing the site so that it is flexible for a wide range in screen sizes, including a registration/log-in account system for users to keep track of their degree status and previously-completed courses, a database of all available courses obtained from the University website, the ability for users to include breaks and create their own schedules, pre-requisite and elective versus required handling, and status tracking for users based on their class level, completed courses, and required courses, along with suggesting courses to be taken. Most of these require only the understanding from Programming Paradigms to implement.

The majority of these requirements can be split into a few key components:

1. The website
2. The user account (along with the registration and log-in) system
3. The course database
4. The scheduling algorithms

A typical user would follow these steps to user our implementation:

Schedule Creation

Course Selection

Website

Registration and Profile Creation

Schedule Creation

Updated Course Selection

Profile/Class Standing Update

The sub-components for this design relate back to the original requirements. For example, the website would include:

* A server (could be done using the uark.edu domain)
* A mobile-friendly UI (supporting multiple screen sizes, possibly a mobile-only version)
* The actual schedule planner, available upon registration, which connects the account system, the course information, and the schedule algorithms

The account system requires:

* Registration of a new profile
* Log in to an existing profile
* Update or creation a profile’s information
* Storage of account information (possibly in plaintext, although that would not be up to standards if security is a requirement)
* Access to account information by users on the website (of which only users can see their own profile)

The database includes:

* A list of all available University courses
* Pre-requisites or co-requisites
* Class times and dates
* Requirements for degree completion

And the schedule algorithm incorporates:

* Sorting based on available classes (this is taken from the account information; users can only select classes they meet the requirements for)
* Sorting based on class times and dates (which are taken from the database of courses; only classes which don’t overlap can be taken)
* Inclusion/exclusion of breaks (which are essentially blank classes with a user-specified date and time)
* User scheduling (requiring the website’s user-friendly interface for a user to make his or her own schedule)

Essentially, these four major components are needed to implement a proper schedule planner. Once a user accesses the website, he or she creates an account and updates the information for their profile, which includes class standing, previously-completed courses, and majors/minors. This information is used to filter which courses are available to the user (for example, if he or she does not meet the pre-requisites for a course, it will be greyed out and the user cannot choose it for their schedule), and which courses are recommended for the user (the ones which are best fit for completing their degree). After courses are chosen, the user can either have a schedule generated for them, choose from several different generated schedules, or create a schedule on their own. Note that some of these features are subject to change, and additional features may be implemented during the project’s development.