

Mines Automatic Class Schedule Planner

Client: Neil Dantam (ndantam@mines.edu)

Project: Identifying and following a suitable plan-of-study is critical to the timely and successful attainment of a college degree. However, many issues—complex prerequisite chains, scheduling conflicts, selectively-offered courses, etc.—limit students’ abilities to define and follow a plan-of-study. Nontraditional, underrepresented, and first-generation students may face disproportionate challenges due to additional constraints, reduced family support, and other factors. This project will produce a tool to automatically generate a plan of study—i.e., the classes to take each semester—based on a student’s major, minor, and elective choices. The goal of this project is to produce a tool that will assist future Mines’ students in completing their degrees.

A back-end class schedule planner is being developed as part of Dr. Dantam’s CSCI-597 class (Robot Planning & Manipulation) as an extension to the research planner TMKit (<http://tmkit.dyalab.org/>). This back-end will have a text interface and be accessible as a web service, e.g. via XML-RPC or a REST-like interface.

The purpose of the field session project is to develop a web interface to make the class schedule planner usable to students. Project goals include:

- *Tier I:* Required
 1. Develop an interface for students to specify major, minor, elective choices, and other class constraints (“student facts”).
 2. Call the back-end planner via a suitable protocol (e.g., XML-RPC).
 3. Develop an interface to display the generated class schedules.
- *Tier II:* Desirable
 1. Develop an interface to iteratively update the student facts, call the back-end, and display alternate schedules.
 2. Develop a database to save alternate sets of “student facts” and generated schedules.
 3. Gather feedback from students (users) on the tool’s usability and desired changes or features.
- *Tier III:* Would-be-nice:
 1. Identify limitations in or desired improvements to the back-end for future development.
 2. Integrate with campus data sources to automatically obtain:
 - Class lists and prerequisites
 - Class times
 - Major, minor, and track requirements
 3. Integrate with shibboleth authentication (MultiPass)

Team Size: 2-4

Skills: Web programming (Python, PHP, and/or Javascript), Version Control (git), Linux

Location: Mines Campus