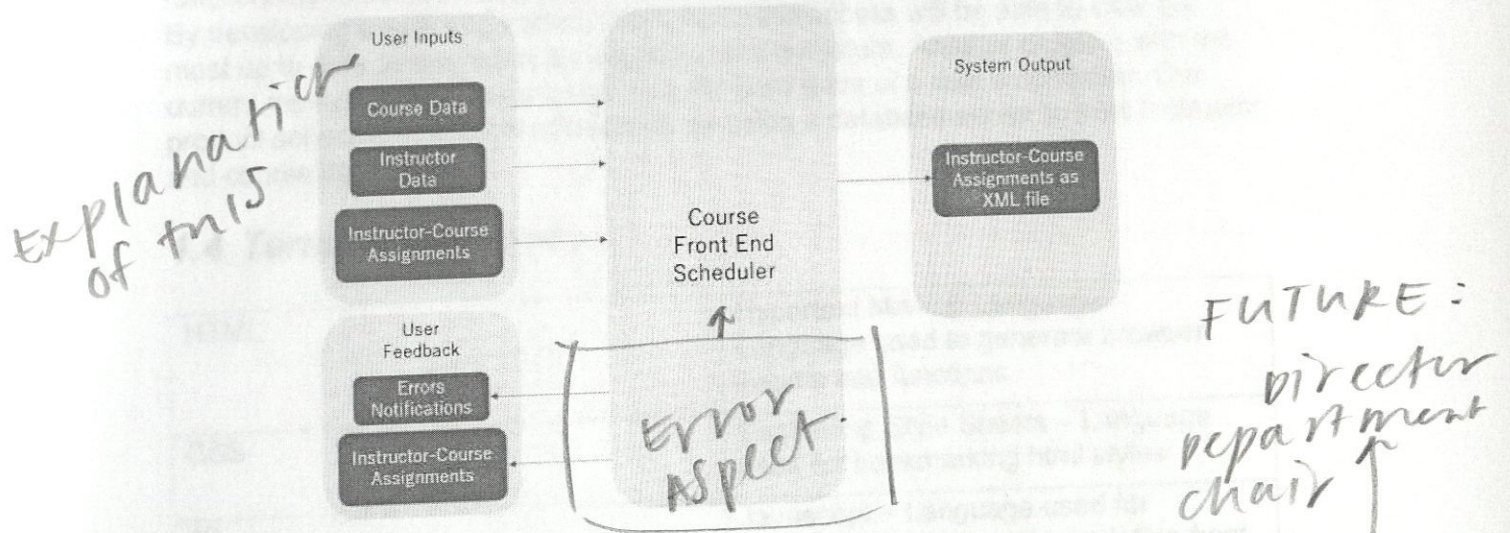


1 Introduction

This document contains the functional specification for the Course Front End Scheduler. The Course Front End Scheduler is a web-based application that allows University Program Coordinators to assign Instructors to the courses that they are teaching for a certain school-year semester. It also allows the user to export a complete document of all Instructor-course assignments as an XML file. Shown below is a block diagram of our system.



1.1 Summary – Kiana & David

The Course Front End Scheduler is a web-based application that encompasses the software components needed to allow users to remotely access a database and manage Instructor-course assignments. Interaction with the system occurs through a web-page based front end that allows the user to view and edit Instructor and course information. The primary goal of this project is to create an application that is convenient and ergonomic for the user, while remaining simple to implement and maintain by administrators. The project is being designed with the intention of being implemented on a secure server on the Texas State University network system. Users of this application include Program Coordinators and Administrative Assistants of the Ingram School of Engineering, and System Administrators.

1.2 Sponsor Requirements – Kiana

A key requirement outlined by the sponsor is to produce a system centred and designed around usability for Program Coordinators. The proposed system will have a user-friendly, easy-to-navigate interface that presents an intuitive process of Instructor-course assignment. A second requirement is to integrate the XML file export feature into the application. The exported XML file is to contain information regarding Instructor-course assignment to be used by an outside Class Scheduling Software, UniTime ref[1].

with UniTime, the course scheduling software will include the functionality of outputting a formatted XML file which contains all the course and Instructor information to be used by UniTime.

The project will be dependent on and will include the following software:

- NodeJS
- AngularCLI
- Python
- Flask
- SQLAlchemy
- PostgreSQL

2.6.3 Hardware – Phillip

The project is purely software, however, regarding hardware, the application will require a server to run on.

Recommended System Requirements:

Processor: Quad Cortex A53 @ 1.2 Ghz (or equivalent)

Memory: 1GB SDRam

Storage: 1GB

Network: An ethernet or wi fi connection

2.6.4 Mechanical N/A

2.7 Boundary Conditions and Constraints – David

The user interface must be able to be embedded into a webpage or otherwise remotely accessed, this necessarily limits the language and software that can be used to implement the user interface.

The platform that the system is deployed on has to support the software the system is employing, namely AngularJS, Flask, and MySQL. In other words it must satisfy the hardware conditions as listed in 2.6.3

The system itself needs to be deployed on a platform that is stable enough in nature that throughout its normal usage will not corrupt or interrupt the system. This means that the platform needs to be able to reset to a functioning state if it is shut off or loses power and begin running the course scheduling system automatically.

Per Dr. Compeau's request, a functional beta version of the program must be operational before Winter break

Texas State University has fairly strict policies on what programs are allowed to be hosted on their servers, it is unclear whether their policies would allow a program like ours to be run on their servers. If Texas State University will not allow our program to run on their servers, an alternative will have to be determined.

3 Project Alignment Matrix – ALL

Outside Advisors (if any) and affiliations:

TABLE 1: Knowledge Alignment Matrix

Course No.	Core knowledge	Specific knowledge incorporated by team
EE 3350 (Electronics I)	Design and analysis of active devices and equivalent circuits	NA
EE 3370 (Signals and Systems)	Frequency domain representation of signals and frequency response, transfer functions	
EE 3420 (Microprocessors)	Principles of operation and applications of microprocessors	
EE 4352 (Introduction to VLSI Design)	Analysis and design of CMOS integrated circuits	
EE 4370 (Communications Systems)	Transmission of signals through linear systems, analog and digital modulation, and noise	

new table w/ cs courses
TABLE 2: Constraint Alignment Matrix (and applicable standards)

ABET Criterion 3 (c): "an ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability."

Constraint Type	Specific Project Constraint
Economic	
Environmental	
Health and safety	
Social/Ethical	
Applicable Standards	