Requirements Specification

Smart Scheduling

Requested by:

Dr. Robert Yoder
Associate Professor of Computer Science
Computer Science Department Head
Siena College

Tom Mottola
Jason Czajkowski
Brian Maxwell
Meghan Servello
Collin Lefeber
Jonathan Smith

October 29, 2010

Version 1.0

Smart Scheduling

Requirements Specification

Table of Contents

1.1 Product Overview and Summary

Convenience is of the utmost importance in today's world. Dr. Yoder is looking for a product that can deliver this same experience in regards to his management of the Siena College Computer Science Department class schedule. Scheduling is currently done by hand and is extremely cumbersome, so our product (Smart Scheduling) seeks to apply the speed and automation of today's technology with the flexibility of an internet application. The product will aim to provide a convenient and familiar calendar-like interface that Dr. Yoder can use easily and efficiently to plan the use of the department's resources. Other users like students and faculty would also be able to take advantage of the Smart Scheduling software by being able to view current semester schedules to facilitate their own plan making should they need a room.

1.2 Development, Operating, and Maintenance Environments

Smart Scheduling will be developed using our resources in the Software Engineering Lab. Here is a list of our available resources. We will use the items in the list below to develop and maintain our team website and our future product.

Server

Web Server: Apache version 2.2.9
Programming Language: PHP version 5.2.6
Database: MySQL 5.0.45

Windows Machine:

- Operating System: Microsoft Windows Vista Enterprise (32 bit)
  - Service Pack 2
- Hardware
  - Processor: Intel Core 2 Duo, 2.93 GHz
  - Memory: 4.00 GB RAM
- Software Installed:
  - Microsoft Office 2007
  - Macromedia Dreamweaver, Fireworks
  - Internet Explorer, Mozilla Firefox, Google Chrome

Macintosh Machine:

- Operating System: Apple Mac OS X
  - Version 10.6.4
- Model: iMac5
- Processor: Intel Core2 Duo
  - Speed: 2 GHz
- Memory (RAM): 1.00 GB

Smart scheduling will be functional across several versions of Internet Explorer, Firefox, Chrome, and Safari to make it as versatile as possible.

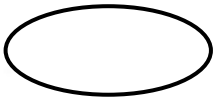
1.3 UML Use Case Diagram

UML Use Case Diagrams are created to show the different users and major functions of a system. The users are known as Actors. The functions these actions perform are known as Uses. Lines are drawn between the Actors and Uses to show the relationship. A legend is shown below, followed by the Smart Scheduling UML diagram.

1.3.1 UML Use Case Legend

System Boundary defines interactions between uses inside a system and the actors outside

Actors are the users, either human or nonhuman, which interact with uses inside the system.

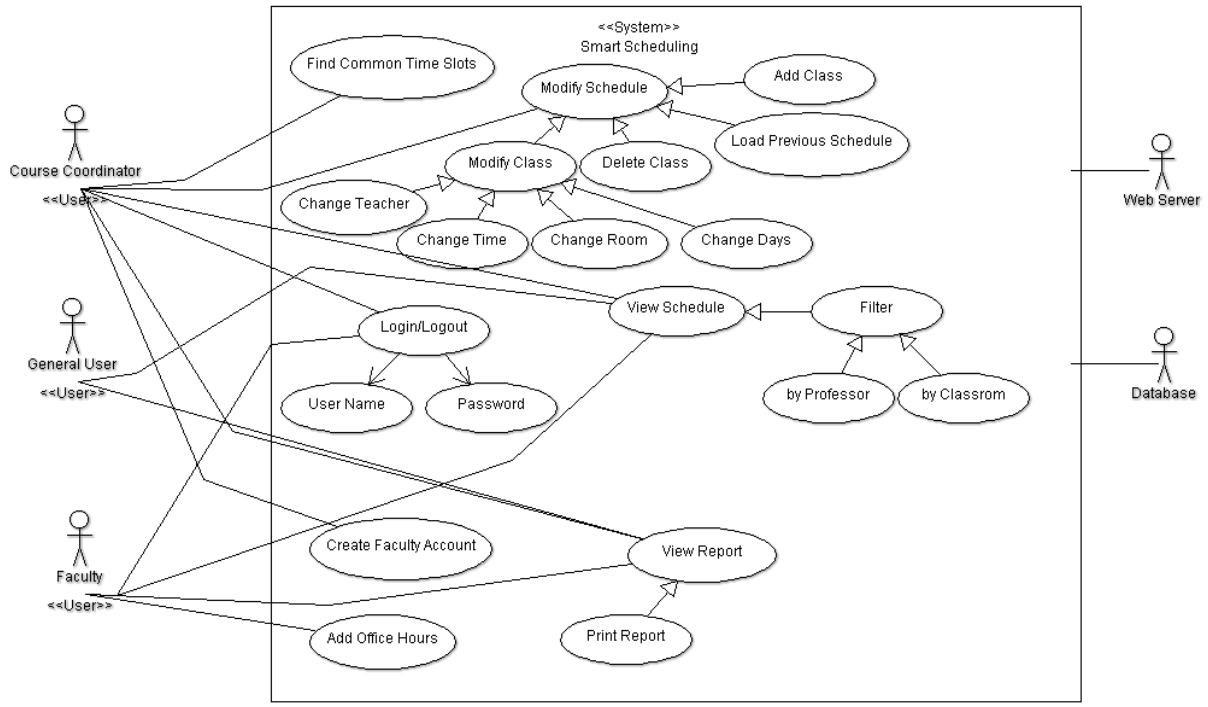


Uses are the activities that interact with actors outside the system.

Participations are the lines that connect actors with uses and uses with other sub-uses showing their participation between them.

The above image displays a legend for our UML Use Case Diagram which is shown on the next page. This legend explains the different objects used in the diagram and what each represents.

1.3.2 UML Use Case Diagram



This UML Use Case Diagram shows all the major functions of each user for the system.

1.4 Usage Case Narratives

Course Coordinator/Administrator
Course Coordinators will have the ability to log into the system via a unique username and password. System is defined as our (Empire Unlimited) application called Smart Scheduling. Once logged in, the Course Coordinator will have the ability to create new or edit existing schedules. The Course Coordinator will be able to assign classrooms, courses, and Faculty in the schedule. The Course Coordinator will have access to a database of past schedules which may be viewed or used as a template for a new schedule. The Course Coordinator will have the ability to view reports on classroom use and Faculty schedules.

The Course Coordinator is responsible for creating and managing schedules, and will submit them to the School Dean and all Faculty members for approval and viewing, respectively.

Faculty
Faculty will have access to the system via a unique username and password. Faculty can view all schedules submitted by the Course Coordinator, as well as be able to view classroom

and Faculty reports. Faculty will have the ability to add office hours and other meetings to their own schedule.

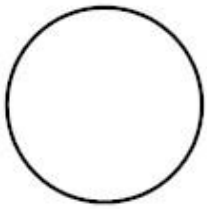
General User
General Users will have access to the system without any authorization. General users will be able to view faculty and classroom schedule reports. A general user will not have any rights to modify or add to a schedule.
In addition to being able to view faculty and classroom schedule reports, schedules will be printed out, and posted outside of classrooms and faculty offices. This will enable everyone, mostly general users, to see at a quick glance which classrooms are being used by which teacher.

1.5 Data Flow Diagrams

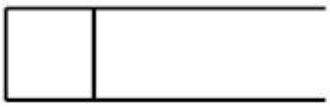
The purpose of Data Flow Diagrams is to show the "flow" and transformation of data through the system. These diagrams are used as a visualization tool to help the audience get a better idea of what exactly is going on in the system. The first image shown below will be a legend describing what each of the different items in the diagrams are and what they represent. A Context Diagram is shown next, which is the general overview of each of the different agents interacting with the system. The Level 0 Diagram shows some more details about which processes each of the agents will be interacting with. Arrows are drawn to show the flow of data between the agents and processes. Following the Level 0 Diagram are two Level 1 Diagrams, each for one of the processes of the system. Each Level 1 Diagram shows that individual process in greater detail and contains labeled arrows to indicate what information is flowing to and from the processes and agents.

1.5.1 Data Flow Diagram Legend

Entities/Sources/Sinks of the system which represents the users. Data will flow between these entities.



Processes of the system which represents the functions. Data will flow from and to these processes.

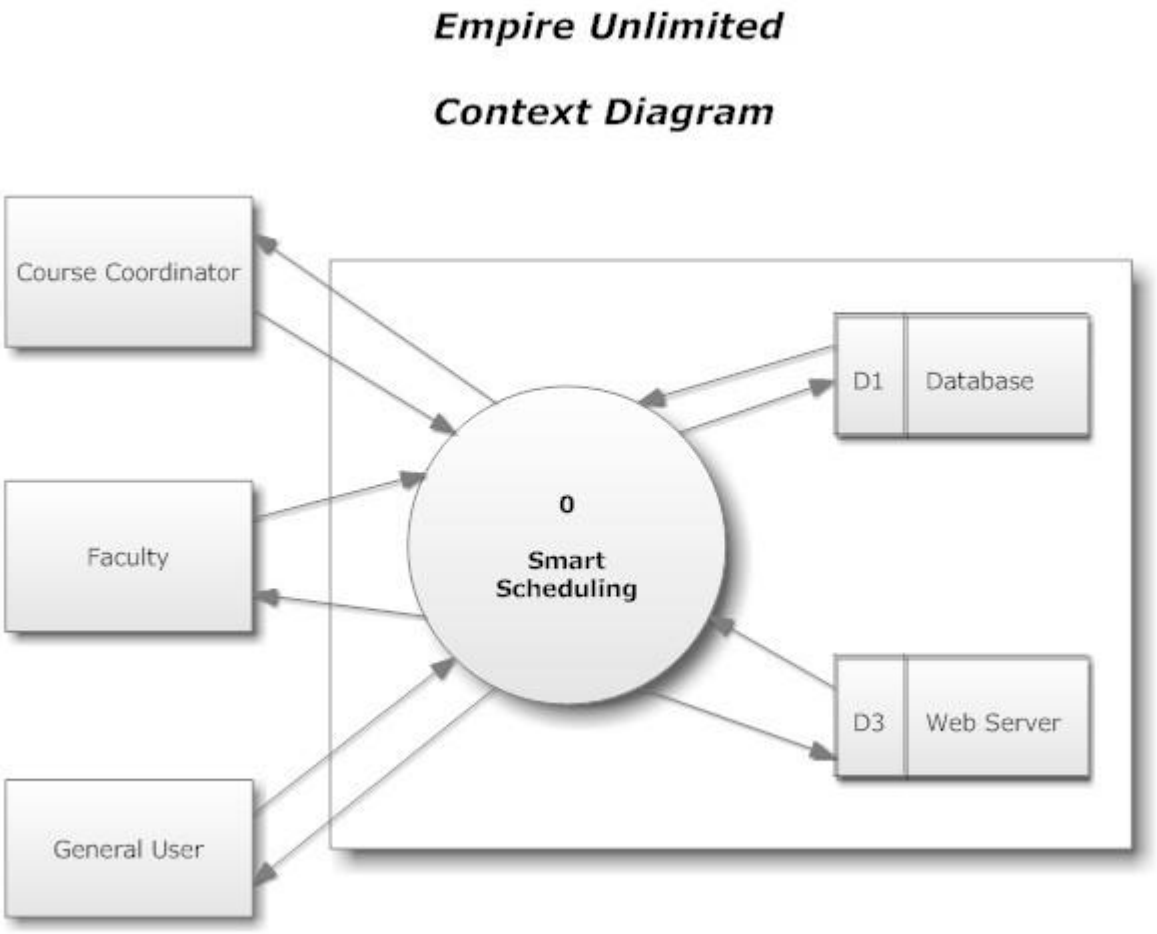


Stores of the system which represent what stores the data. Data will flow into, be saved and be pulled from these data stores.

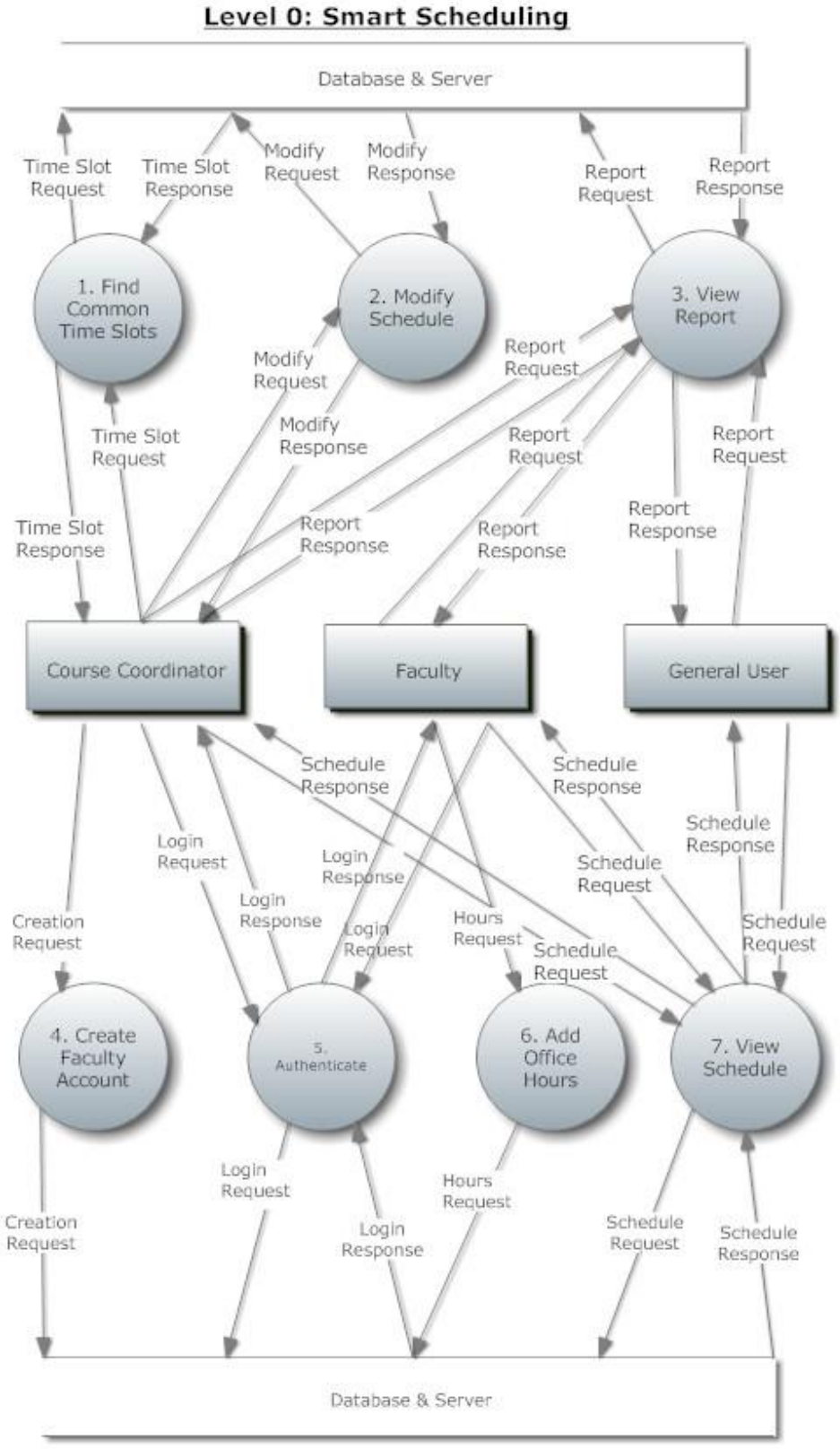
The Flows of the system which represent the chuck or packets of information going from one component to another.

Above is a legend for the Data Flow Diagrams. This legend explains all objects used in the diagram and what they represent.

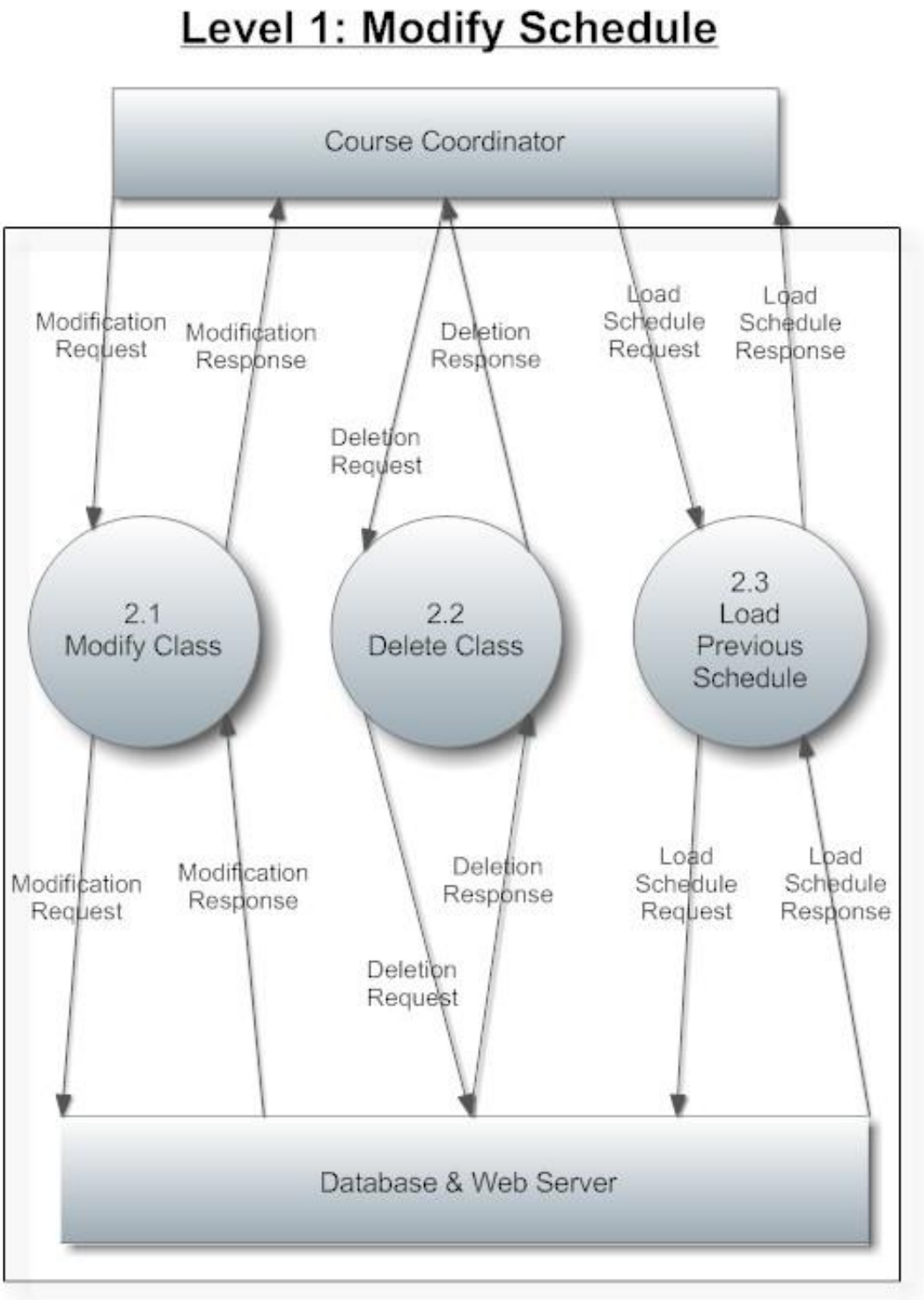
1.5.2 Context Diagram



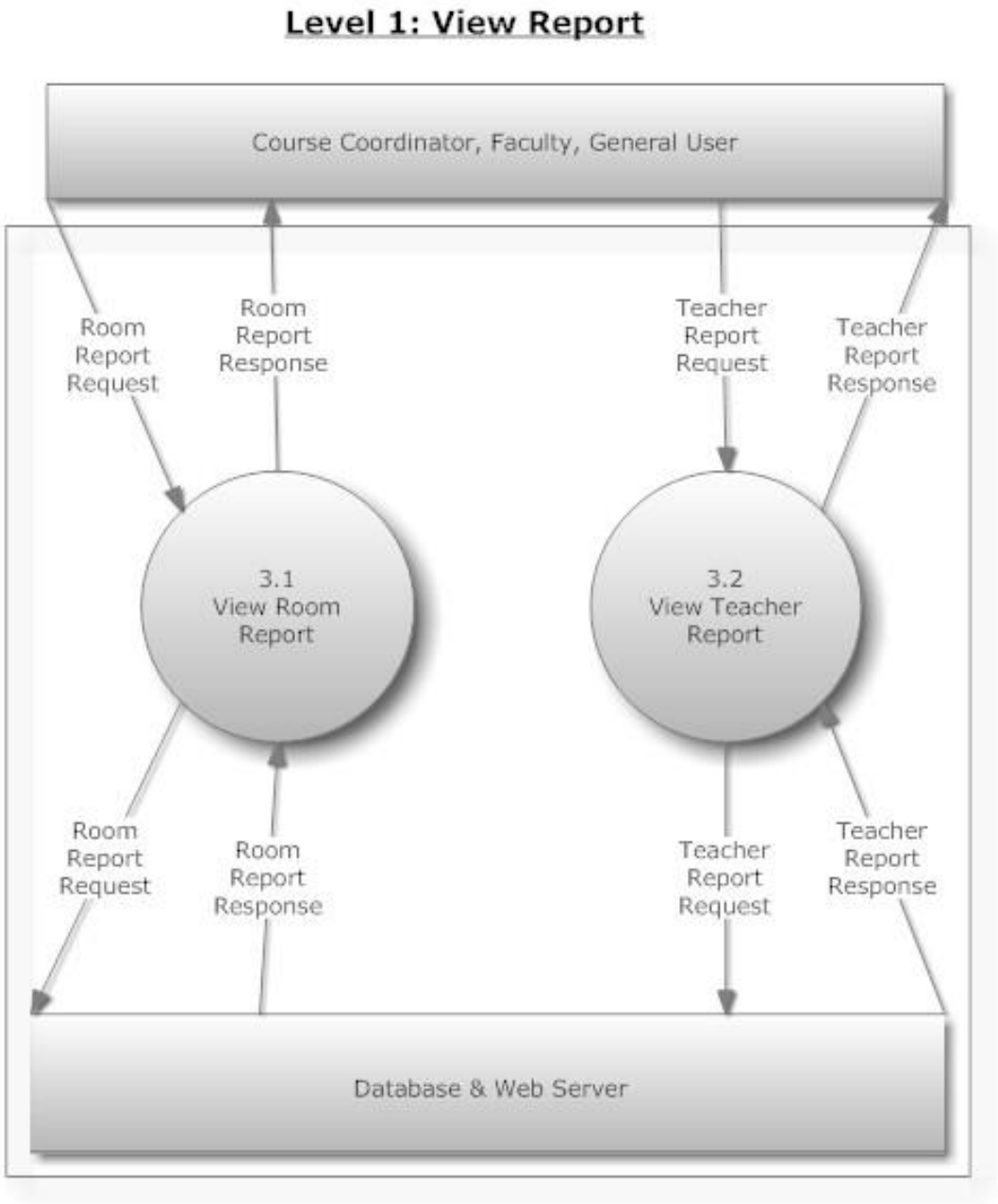
1.5.3 Level 0 Diagram



1.5.4 Level 1 Diagram



1.5.5 Level 1 Diagram: Register



1.6 Prototypes Used for Discovery

SMART SCHEDULING

day view - all professors, classrooms, times

8:00	**Class - Room - Professor - Time**
9:00	**Class - Room - Professor - Time**
10:00	**Class - Room - Professor - Time**
11:00	**Class - Room - Professor - Time**
12:00	**Class - Room - Professor - Time**
1:00	**Class - Room - Professor - Time**
2:00	**Class - Room - Professor - Time**
3:00	**Class - Room - Professor - Time**

The picture above is an example of viewing a single day's schedule.

SMART SCHEDULING

week view - all professors, classrooms, times

Day:	Sun	Mon	Tues	Wed	Thurs	Fri	Sat
8	5 events						
9		3 events			2 events		
10							
11							
12				5 events			
1							
2		5 events					

This diagram is an example of an overview of a week of the master schedule.

SMART SCHEDULING

week view - all professors, classrooms, times

Day:	Sun	Mon	Tues	Wed	Thurs	Fri	Sat
8	5 events						
9		Database - RB328 Discrete Math - RB300 Algorithms - RB328			2 events		
10							
11							
12				5 events			
1							
2		5 events					

When you click on a date and time that has events assigned to it the box expands to show the list of events.

1.7 Functional Requirements Inventory

Smart Scheduling will be able to run on all popular browsers including but not limited to: Internet Explorer, Mozilla Firefox, as well as Google Chrome.

The following is the functional requirements for each user. The requirements are grouped by the user; there are 3 users for Smart Scheduling.

Course Coordinator

- Will be able to find common time slots to schedule departmental meetings or new classes.
- Will be able to modify the schedule.
    - Will be able to delete previous schedules.
    - Will be able to delete classes
    - Will be able to modify classes.
- Will be able to add new classes.
- Will be able to securely log into the system using a registered username and password.
- Will be able to view the current schedule.
    - Will be able to filter the schedule by professor(s).
    - Will be able to filter the schedule by classroom(s).
- Will be able to create a faculty account, where faculty will be able to access the system.
- Will be able to view and print room reports.

Faculty

- Will be able to securely log into the system using a registered username and password.
- Will be able to view the current schedule.
    - Will be able to filter the schedule by professor(s).
    - Will be able to filter the schedule by classroom(s).
- Will be able to view and print room reports.
- Will be able to add office hours or other meeting, where they will be available in order to meet with students or fellow teachers.

General User

- Will be able to view and print room reports.
- Will be able to view the current schedule.
    - Will be able to filter the schedule by professor(s).
    - Will be able to filter the schedule by classroom(s).

This list is subject to change as more information is gathered about the project.

1.8 Non-Functional Requirements Inventory

The non-functional requirement inventory is a list of non-functional system requirements. This list is composed of requirements that specify how the system *should* be. This list is subject to change as more information is gathered about the project.

- The system will be easily maintained.
- The system will be stable.
- The system will be viewable on multiple browsers.
- The system will be efficient.
- The system will be user friendly and easy to use.

1.9 Exception Handling

In the current phase of our project, it is unclear as to what types of exceptions may need to be handled for Smart Scheduling. However, as we progress further through the phases of development and gather more information, exceptions will arise. An exception that is expected to arise will be the warning system failing to give an error for a conflict, such as when two teachers are scheduled to a classroom at the same time. Another exception could be if an outside source attempts to hack the database.

1.10 Early Subsets and Implementation Priorities

Important implementation priorities are as follows:

- The ability to run on major browsers.
- A friendly user interface.
- A secure login.
- The ability for the system to be updated and improved, if necessary, in the future.

1.11 Foreseeable Modifications and Enhancements

Empire Unlimited currently sees no need for future modifications and enhancements for Smart Scheduling. The reason for this is that once the system is completed, it will accomplish all necessary goals for scheduling. However, if any new constraints need to be added for the self-check system, depending on what our client Dr. Yoder wishes to have, then that could be an improvement.

1.12 Acceptance Criteria and Testing Requirements

The acceptance criteria for the system will be defined by the functional requirements inventory which was listed in section 1.7 of this document. Additionally, the non-functional requirements inventory listed in section 1.8 of this document will be used. Functional requirements, by their very definition, are what the system will be able to accomplish, and what can be tested to see if they exist within the system. Non-functional requirements cannot be tested for, but define how the system should behave. After the completion of Smart Scheduling, Empire Unlimited will test and determine which requirements were met, and which were not.

Smart Scheduling will be tested on all popular browsers including but not limited to: Internet Explorer, Mozilla Firefox, as well as Google Chrome. Testing requirements will be approved and developed by Empire Unlimited. Testing will occur by first testing each functional requirement separately. When and if each functional requirement works correctly, they will all be put together to represent the system, then the system as a whole will be tested. This testing plan will be explained in further detail in the Preliminary Design and Detailed Design documents which will be completed as the project progresses. The results of these tests will be described in the Acceptance Test document, where it will be determined whether or not all the requirements were met, and if they work correctly. The acceptance Test document will be created as the project progresses.

1.13 Design Hints and Guidelines

Due to Smart Scheduling being in the very early stages of the development process, there are few hints and guidelines that can be described at this time. As more information is gathered regarding the project, and more is learned as we progress through the development stages, guidelines will be discovered. These guidelines can also be laid out by the Client, Dr. Yoder as he becomes aware of any specific features that he wants.

Appendix A: Sources of Information

The information presented in this document was gathered through meetings held with the client, Dr. Eric Breimer. Information was also taken from class lectures held with Dr. Lederman and from documents written by past Software Engineering groups.

Appendix B: Glossary of Terms

Ajax (**Asynchronous JavaScript and XML)** - A group of web development techniques used on the client-side to create interactive web applications.

Apache HTTP Server (Web Server) - Referred to as Apache, it is web server software notable for playing a key role in the initial growth of the World Wide Web.

Cascading Style Sheets (**CSS**) - A style sheet language used to describe the presentation semantics (the look and formatting) of a document written in a markup language.

Chrome – Internet browser designed by Google.

Conflict and Constraint – When an activity can't be scheduled due to room use, weekend, and one resource being currently in use.

Database - An organized collection of data for one or more uses, typically in digital form.

Dreamweaver – A web development application.

Dropbox - A Web-based file hosting service operated by Dropbox, Inc. which uses cloud computing to enable users to store and share files and folders with others across the Internet using file synchronization.

Firefox – Internet browser designed by Mozilla.

Gantt Chart - A type of bar chart that illustrates a project schedule. Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project.

HTML (HyperText Markup Language) - The predominant markup language for web pages. It is written in the form of HTML elements consisting of "tags" surrounded by angle brackets within the web page content. It is the building blocks of all basic websites.

Internet - A global system of interconnected computer networks that use the standard Internet Protocol Suite (TCP/IP) to serve billions of users worldwide. It is a *network of networks* that consists of millions of private, public, academic, business, and government networks, of local to global scope, that are linked by a broad array of electronic and optical networking technologies.

Internet Explorer (IE) – Internet browser designed by Microsoft.

JavaScript - An implementation of the ECMAScript language standard and is typically used to enable programmatic access to computational objects within a host environment.

MySQL - A relational database management system that runs as a server providing multi-user access to a number of databases.

PHP (Hypertext Preprocessor) - A widely used, general-purpose scripting language that was originally designed for web development to produce dynamic web pages.

Ruby - A dynamic, reflective, general purpose object-oriented programming language that combines syntax inspired by Perl with Smalltalk-like features.

Room Report – Schedule of times when a room, lab, or the Computer Science Library is booked, by a class or group, on a given day of the week.

Spiral Model - A software development process combining which elements of both design and prototyping-in-stages, in an effort to combine advantages of top-down and bottom-up concepts.

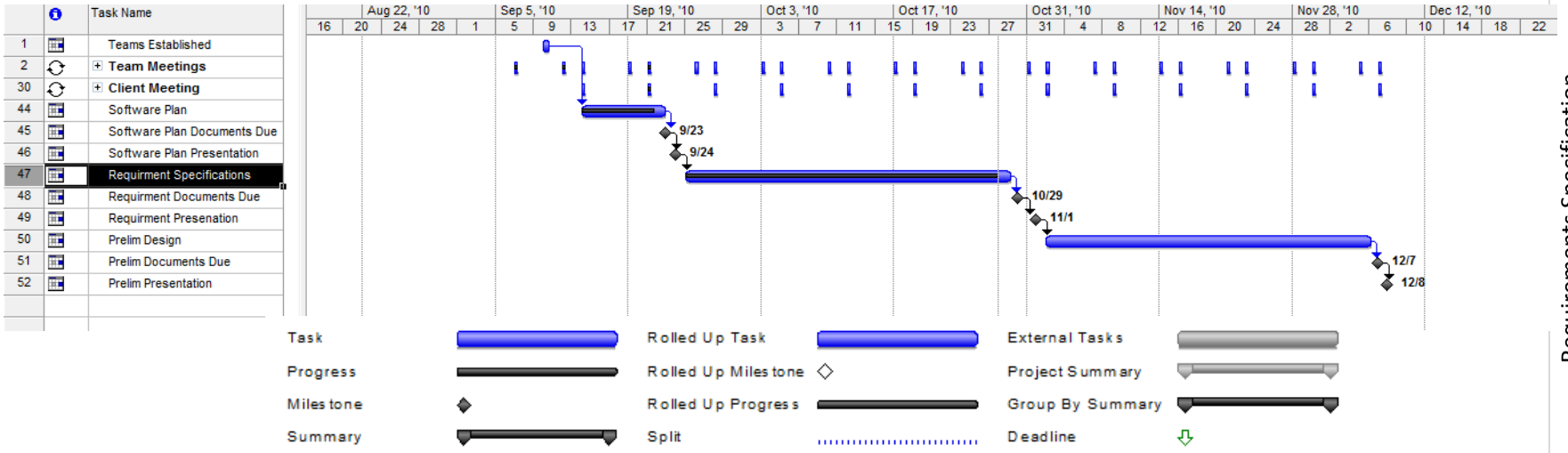
Waterfall Model (Classic) - The Classic Waterfall Model is a sequential software development model in which development is seen as flowing steadily downwards (similar to a waterfall) through the phases of requirements analysis, design, implementation, testing, integration, and maintenance.

WinZip - A proprietary file archiver and compressor for Microsoft Windows,

XHTML (eXtensible Hypertext Markup Language) - A family of XML markup languages that mirror or extend versions of the widely used Hypertext Markup Language (HTML), the language in which web pages are written.

XML (Extensible Markup Language) - A set of rules for encoding documents in machine-readable form.

Timeline - Gantt Chart



The Gantt Chart displays Empire Unlimited's timeline for the semester, showing team meetings, client meetings, and other important dates.