SOFTWARE REQUIREMENTS SPECIFICATION

For

Docket Course Scheduling

Version 1.0
February 5, 2013

Developed by
Robert Brown
Nathal Fonseka
Max Haley
Michael Kovalichik
Joseph Nelson
Basil Nyachogo
Raymond Selfridge

Prepared for
Computer Science Department
Drexel University
Dario Salvucci, Advisor
## Version History

<table>
<thead>
<tr>
<th>Version</th>
<th>Author</th>
<th>Date</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>Joseph Nelson</td>
<td>4/20/13</td>
<td>Revisions</td>
</tr>
<tr>
<td>1.5</td>
<td>Joseph Nelson</td>
<td>4/28/13</td>
<td>Final Submission</td>
</tr>
</tbody>
</table>
# Table of Contents

1.1 Purpose.........................................................................................................................3  
1.2 Scope.............................................................................................................................3  
1.3 Definitions, Acronyms, Abbreviations.................................................................3  
1.4 Outline..........................................................................................................................3  
2. Overall Description...................................................................................................................... 3  
2.1 Product Perspective............................................................................................................. 3  
  2.1.1 User Interface.......................................................................................................... 3  
  2.1.2 Hardware Interfaces............................................................................................... 4  
  2.1.3 Software Interfaces............................................................................................... 4  
  2.1.4 Communications Interfaces.................................................................................... 4  
  2.1.5 Memory Constraints............................................................................................... 4  
2.2 User Characteristics.............................................................................................................. 4  
2.3 System Evolution................................................................................................................... 4  
3. Docket UX Specification........................................................................................................... 4  
3.2 Layout Specification.............................................................................................................. 4  
  3.2.1 General Layout......................................................................................................... 4  
  3.2.2 Navigation Bar......................................................................................................... 4  
  3.2.3 Interface Modularity............................................................................................... 4  
3.3 Account Management........................................................................................................... 4  
  3.3.1 Registering an Account.......................................................................................... 4  
  3.3.2 Logging In / Logging Out...................................................................................... 5  
  3.3.3 Department Administration.................................................................................. 5  
3.4 Views................................................................................................................................. 5  
  3.4.1 Schedule View......................................................................................................... 5  
  3.4.1.2 Calendar View Filtering..................................................................................... 5  
  3.4.2 Instructors View...................................................................................................... 5  
  3.4.2.1 Details View......................................................................................................... 5  
  3.4.3 Course View Page.................................................................................................... 5
When a user selects “Add” from the “Action” menu on the “Rooms View” page, a popup appears like the one in Figure 14 allowing the user to schedule a new section for the course. Users can enter all details for the section, including Course, Instructor, Location, Term, Tim and Date.  

3.4.4.3 Delete Room Dialog  
3.4.5 Terms View Page  
3.4.5.1 Term Edit Dialog  
3.4.5.2 Delete Term Dialog  

4. Functional Requirements  
4.1 Overview  
4.2 Application Setup  
4.2.1 Launching Docket  
4.2.2 Configuring Docket  
4.3 User Accounts  
4.4 Courses  
4.5 Instructors  
4.6 Locations  
4.7 Sections  
4.8 Data Views  
4.9 User Subscriptions  

5 Nonfunctional Requirements  
5.1 Portability  
5.2 Maintainability
1.1 Purpose

This document specifies the entire software architecture and design for the Docket Course Scheduling system. These design decisions directly relate to the functional and non-functional requirements as specified in the Docket Software Requirements Specification [reference?]

Docket is a decision support system that allows University administrators to effectively manage departmental resources, resolve scheduling conflicts, and reduce cost associated with scheduling academic courses. Docket’s primary goal is to identify scheduling conflicts and notify users in order to facilitate conflict resolution.

1.2 Scope

This document presents the specifications of all of the requirements of the Docket course scheduling application. The requirements relate to the functionality, restrictions, capabilities, and user interfaces of the software. This version of the document encompasses the requirements for the initial release of the system (Version 1.0).

The intended audience for this document includes designers, developers, and testers of the Docket software system.

1.3 Definitions, Acronyms, Abbreviations

**Authentication** The process of establishing and verifying the identity of a security principal

**Authorization** The process of verifying a security principal has the right to perform an action

**Access Control List** An Authorization mechanism that controls what security principals have access to what resources be specifying a list of authorized principals and associated permissions *also ACL*

**Controller** See **Resource Request Handler**

**CAS** See **Central Authentication System**

**Cascading Style Sheet also CSS** an markup language used to define the style (formatting) of an HTML document.

**Central Authentication System** A single-sign on authentication provider maintained by the Drexel University Information Resources and Technology department. Also called DrexelConnect, the Central
Authentication System centralizes authentication and authorization for Drexel’s various Information Systems

CSS  See Cascading Style Sheet

Data Access Object A software component responsible for the retrieval, update, and deletion of domain objects

Data Binding a mechanism used to insure synchronization between a view and model. A change in either a view is automatically propagated to a model and vice versa

Data Confidentiality A security principle that ensures only authorized security principals can access security data

Data Integrity A security principle that ensures secured data is modified only by authorized security principals. Additionally, Data Integrity ensures that unauthorized data modification is easily detectable

Decision Support System A computer –based information system that supports business or organizational decision-making processes also DSS

Dependency Injection A style of object configuration in which an object’s fields and collaborators are set by an external entity.

Docket UX A decoupled web-based GUI for the Docket system also UX

Domain Object See Resource

HibernateORM a framework that provides mapping of Object Oriented data structures used in OOP programming to relational tables used in a RDMS

Interactive Clients Any client which requires a human operator

Inversion of Control A design style that emphasizes decoupling object creation from usage simplifying lifecycle management by delegating instantiation to an external entity

Java Servlet Container An implementation package that describe and define the contracts between a servlet class and the runtime environment

JSON JavaScript Object Notation, a lightweight text-based open standard for human-readable data exchange.

Jersey a Java-RS compliant framework that provides Domain Object data marshaling, URI processing, and mapping of URI to Controllers

Non-Interactive Clients Any client whose interaction with this system is automated or semi-automated and may not require a human operator

Object-Relational Mapping Framework A software component used to map an OOP object model to a relational model for storage in a relational database

Persisting The act of transferring an object from in-memory storage to permanent or persistent storage (i.e. database, file, etc.)

Post-Data When an end -user enters data into a web form, the HTTP client submits the data to the HTTP server via an HTTP POST method

Principal See Security Principal

Resource Any named information or collection thereof. Docket resources include Courses, Sections, Instructors, and Rooms also Domain Object
Resource Request Handler A software component that coordinates the retrieval and update of system resources also Controller

Security Posture Describes the planning and implementation approach to security. The Security Posture may describe hardware and software protections deployed to secure an asset

Security Principal A user, device, or system that can perform an action within the application

SpringMVC a Web Services framework that coordinates dependency injections, Email, and Security services

Spring Security 3 a framework that provides Authentication and Authorization services for URL and Domain Object security

URL Uniform Resource Locator, the address of a World Wide Web page

URI Uniform Resource Identifier, the part of URL that uniquely identifies a resource

Web Application Server An application server that handles HTTP request and coordinates communication with application programs through any number of protocols

Web Services Framework A software component that facilitates development and integration for web services including facilities for IoC, Dependency Inject, Flexible Marshalling, and security

XML Extensible Markup Language, a meta-language designed to enable easy data transformation, transportation, and storage.

1.4 Outline

The remaining portion of this document outlines how to use Docket as well as the functional and nonfunctional requirements of Docket. Section 2 outlines the general details of Docket. Section 3 outlines the Functional Requirements for the Docket UX. Section 4 outlines the Functional Requirements and Business Logic for the Docket Back End. Section 4 outlines the Nonfunctional Requirements for the Docket application.

2. Overall Description

2.1 Product Perspective

2.1.1 User Interface

The Docket UX is a web-based application that provides a centralized location for accessing and modifying information provided by multiple departments. Aggregating departmental data, Docket improves the course scheduling workflow by notifying users of conflicting course time and date entries. Additionally, Docket UX provides powerful filters to allow users to
select and focus on key data entered by all users. Finally, the Docket’s Subscription feature automatically notifies users of changes (e.g. time, date, location, etc.) for courses they are subscribed to.

The Details view displays information related to courses, focusing on one course from the department’s course list. This view allows users to modify information about courses and assign sections, professors, and rooms to the course. Details about courses can be accessed from the Calendar and Professor views and from the Courses menu in the navigation bar.

The Calendar view displays what days and times the courses meet and how the courses line up with each other. The Calendar view also allows users to filter the displayed courses based on selected heuristics, limiting the scope of which courses users focus on at a given moment.

The Professor view organizes and displays courses assigned to professors in a department and which terms the professor teaches classes. Unassigned courses are also represented on this page.

2.1.2 Hardware Interfaces

The Docket application can be run on any computer hardware that meets the following minimum requirements:

- Capable of connecting to an Ethernet-based TCP/IP network
- Capable interfacing with an instance of PostgreSQL Database system (the instance may be local to the same server or hosted on a remote server)
- Capable of running a Java Servlet 2.4 Container or self-hosting Java Servlet

Docket can be accessed from any client that meets the following minimum requirements:

- Capable of connecting to an Ethernet-based TCP/IP network on which the Docket application is running
- Cable of running an HTTP 1.1 compliant browser
• Includes a keyboard and pointing device

2.1.3 Software Interfaces

The Docket application integrates external software to provide some of its functionality.

**Authentication:** The Docket software authenticates users using an internal security component for access to the scheduling services API and can be extended to use a CAS

**Client:** The Docket UX is run within the user's web browser and expects that it is compliant with the HTTP 1.1 and HTML 4.0 standards

**Database:** The Docket software uses an external database to store state information and can be configured to use one of several popular relational database systems

**Server:** The Docket Decision Support System is deployed as a Java Servlet which can either be hosted in a standalone servlet container or can be run as a self-hosting using an embedded Jetty servlet container. Additionally, the default Docket UX is deployed as an AngularJS application

2.1.4 Communications Interfaces

Docket user common network protocols for communication between the client and Docket DSS. All client interaction with the Docket DSS API occurs via HTTP 1.1 GET and POST operations. Additionally, SMTP is used by the Subscription service to notify users via e-mail of changes to subscribed courses

2.1.5 Memory Constraints

The Docket Decision Support System requires a minimum 1 GB of memory. A client requires a minimum 512 MB of memory.

2.2 User Characteristics

Users of Docket should have a basic understanding of how to use a computer to access an Internet site. Proper use of Docket also requires the user to understand the University course scheduling workflow including intra- and inter-departmental curriculum as it relates to course scheduling.
2.3 System Evolution

The current version of Docket is designed for use by Drexel University. The Docket software is designed with the following possible evolutions to expand for additional users:

- Development of additional user interfaces to supplement or replace the default user interface
- Additional course and scheduling views
- Extensible scheduling rules for adoption by external colleges and universities

3. Docket UX Specification

This section describes the details related to implementing the Docket UX. The Docket UX is provided both as an example client-side application built on top the DSS and the primary interface for interactive (browser-based) clients. The following sections describe the Docket UX in detail.

3.2 Layout Specification

3.2.1 General Layout

Each page of the Docket UX contains the navigation bar and the content below it. The navigation bar always appears at the top of the screen, above the content of the page, including scrolling pages.

Each page uses a backing grid system to structure and group the page’s content. Actions that can be performed on pages are always grouped in a navigation bar on the right side of the page. Figure 1 shows the Docket homepage and grid structure containing hyperlinks to actions users can perform in addition to help text.
3.2.2 Navigation Bar

The navigation bar contains access to the different views provided by the Docket UX, along with quick access to pages related to courses, instructors, and rooms related to a user’s department. Clicking an item on the Navigation Bar allows users to further filter results.

3.2.3 Interface Modularity

The Docket UX uses a responsive interface that adjusts to the user’s screen size, accommodating large computer monitors and smaller mobile devices.
screens. For smaller browser windows and mobile browsers, the navigation bar adjusts to display on the page. Clicking on the navigation bar toggles the hidden options of the menu, and the contents of the page reorganize to better fit the browser window.

3.3 Account Management

3.3.1 Registering an Account

Users can register for an account from the main page; users cannot be logged in to register an account. The email address provided by the user is used as the identifier for logging into Docket.
3.3.2 Logging In / Logging Out

Users log into their account from the main page. In order to log in, the user cannot be currently signed in and must have an existing account. Users can click on the Sign In button at the right of the navigation bar to log in. Mobile browsers are provided with a login page instead of a pop-up window and access the page from the same location.

Users logout from the Docket Front End by clicking on the same location as the login page while logged in.

3.3.3 Department Administration

System administrators that login to the Docket UX have access to the administration page. Administrators can approve or reject user accounts, as well as remove accounts from the Docket system. Currently, when a user
account is listed as “Reject”, the account remains in the system, however, the user account cannot be used to login to the system.

![Administration Page](image)

**Figure Administration page**

### 3.4 Views

#### 3.4.1 Schedule View

The Calendar View is the primary means of viewing courses in the Docket UX. The calendar view provides a weekly itinerary of courses provided by the department and what days and times the courses are offered. The column on the right of the page contains a list of courses the department offers as well as the courses the department is subscribed to. Clicking on either the links in the list of courses or on the time blocks in the calendar will link to the course’s details page.

![Calendar View](image)

**Figure Calendar View**

#### 3.4.1.2 Calendar View Filtering

The Docket Front End allows users to filter the courses shown on their calendar. The filter options will display in a window at the top of the
page. After filtering, only courses that meet the requirements will show on the calendar.

3.4.2 Instructors View

The Instructor View displays a list of instructors associated with the department, along with which courses they are assigned to teach. Courses assigned to an instructor are grouped together by term. A list of courses that have not been assigned, grouped by term offered, is displayed below the instructors.
3.4.2.1 Details View

3.4.3 Course View Page

The Course View displays a list of courses associated with the department, along with which sections for each course. Users can schedule a section or edit an existing section.

3.4.3.1 Course Edit Dialog
When a user selects “Edit” from the “Action” menu on the “Course View” page, a popup appears like the one in Figure 13 allowing the user to make changes to the course. Users can change the Department code, Course Number, Description, number of credit hours and Type.
3.4.3.2 Assign Section Dialog

When a user selects “Add” from the “Action” menu on the “Course View” page, a popup appears like the one in Figure 14 allowing the user to schedule a new section for the course. Users can enter all details for the section, including Course, Instructor, Location, Term, Time and Date.

![Create New Section](image)

3.4.4 Rooms View Page

The Rooms View displays a list of Rooms associated with a department. From the Rooms view, users can add a new room, edit or delete an existing room, or assign a section to a room.
3.4.4.1 Room Edit Dialog

When a user selects “Edit” from the “Action” menu on the “Room View” page, a popup appears like the one in Figure 16 allowing the user to make changes to the Room. Users can change the Building code, Room Number, and room capacity.

3.4.4.2 Assign Section Dialog
When a user selects “Add” from the “Action” menu on the “Rooms View” page, a popup appears like the one in Figure 14 allowing the user to schedule a new section for the course. Users can enter all details for the section, including Course, Instructor, Location, Term, Tim and Date.

### 3.4.4.3 Delete Room Dialog

When a user selects “Delete” from the “Action” menu following selecting a room in the “Room View” page, a popup dialog like the one in Figure 17 appears asking for confirmation of the action. If the user clicks “Confirm” the room is deleted from the system. Any section is schedule still exist, but the room is set to “Unassigned.” If the user selects “Close” the system performs no action.

![Delete CURTIS 340](image)

### 3.4.5 Terms View Page

The Terms View displays a list of Terms associated with a institution. From the Terms view, users can add a new term, edit or delete an existing term, or see the whose start and end date falls within the start and end date configured for the Term.
3.4.5.1 Term Edit Dialog

When a user selects “Edit” from the “Action” menu on the “Term View” page, a popup appears like the one in Figure 19 allowing the user to make changes to the Term. Users can change the Term Name, Start Date and End Date.

3.4.5.2 Delete Term Dialog
When a user selects “Delete” from the “Action” menu following selecting a Term in the “Term View” page, a popup dialog like the one in Figure 20 appears asking for confirmation of the action. If the user clicks “Confirm” the Term is deleted from the system. Any section is scheduled still exist, but the Term is set to “Unassigned.” If the user selects “Close” the system performs no action.

4. Functional Requirements

4.1 Overview

Sections 4 – 5 specify the requirements for the Docket scheduling system. Each requirement is given a corresponding priority code to indicate the importance of the feature:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority 1 (P1)</td>
<td>Primary feature; P1 requirements must be fully satisfied for the current version.</td>
</tr>
<tr>
<td>Priority 2 (P2)</td>
<td>Supporting feature; P2 requirements should be satisfied for the current version, but are not mandatory.</td>
</tr>
</tbody>
</table>
4.2 Application Setup

4.2.1 Launching Docket

R4.2.1.1 Self-hosting Server: The Docket DSS can be executed as a standalone application server. Priority 1

R4.2.1.2 Within a Servlet Container: The Docket DSS service can be served within a Java Servlet Container. Priority 1

R4.2.1.3 Default GUI: The default Docket UX application can be served as an AngularJS application. Priority 1

4.2.2 Configuring Docket

R4.2.2.1 Changing the listening port: The port which the Docket Back End application listens on for incoming HTTP connections can be changed. Priority 1

R4.2.2.2 Enabling HTTPS: The Docket Back End has the ability to use TLS encrypted HTTPS connections in lieu of unencrypted HTTP connections. Priority 2

R4.2.2.3 Configuring SMTP: The Docket Back End's SMTP connection details can be specified at the time that the application is started. Priority 1

R4.2.2.4 Configuring API Location: The Docket Front End's expected API URI can be specified at the time that the application is started. Priority 1

R4.2.2.5 Specifying the Database Connection: The Docket Back End's connection details for interfacing with the relational
database instance can be specified at the time that the application is started. *Priority 1*

### 4.3 User Accounts

**R4.3.1 User Account Types and Privileges:** The Docket software system classifies user accounts by the following roles:

- **R4.3.1.1 System Administrator** *Priority 1*
- **R4.3.1.2 Scheduling User** *Priority 1*

**R4.3.2 Account Role:** The account role defines the actions that are available to the user. *Priority 1*

**R4.3.3 System Administrator Exists:** At all times at least one user account with the System Administrator Role exists within the system. *Priority 1*

**R4.3.4 System Administrator Privileges:** A System Administrator user account can perform any action within the system. *Priority 1*

**R4.3.5 Department Association:** User accounts with the Scheduling User role have a department associated with them. *Priority 1*

**R4.3.6 Registering a New Account:** Registering a user account using the Front End submits the account for verification by an administrator. *Priority 1*

**R4.3.7 Approving a New Account:** System Administrators can approve a new account submission for activation within the system. *Priority 1*

**R4.3.8 Rejecting a New Account:** System Administrators can reject a new account submission. *Priority 1*
4.4 Courses

R4.4.1 Course information: Each course offered by a department has the following information associated with it which can be modified by users:

- **R4.4.1.1 Department Code** *Priority 1*
- **R4.4.1.2 Course Number** *Priority 1*
- **R4.4.1.3 Course Name** *Priority 1*
- **R4.4.1.4 Course Prerequisites** *Priority 2*
- **R4.4.1.5 Course Description** *Priority 1*
- **R4.4.1.6 Course Corequisites** *Priority 1*

R4.4.2 Course Modification Privileges: Course information for courses offered by a department can only be modified by user accounts associated with that department. *Priority 1*

4.5 Instructors

R4.5.1 Instructor Information: Each instructor within a department has the following information associated with it which can be modified by users:

- **R4.5.1.1 Name** *Priority 1*
- **R4.5.1.2 Courses the instructor is preferred to teach** *Priority 1*
- **R4.5.1.3 Number of courses the instructor should teach each term** *Priority 2*

R4.5.2 Instructor Modification Privileges: Instructor information can only be modified by user accounts associated with that department. *Priority 1*
4.6 Locations

R4.6.1 Location Information: Each location which may be associated with a scheduled section has the following information associated with it which can be modified by users:

- **R4.6.1.1 Building Code** *Priority 1*
- **R4.6.1.2 Room Number** *Priority 1*
- **R4.6.1.3 Owning Department** *Priority 2*
- **R4.6.1.4 Capacity** *Priority 1*

R4.6.2 Non-exclusive Locations: To facilitate online, cross-listed, and bi-weekly sections a location can be marked as non-exclusive *Priority 1*

R4.6.3 Location Modification Privileges: Location information can only be modified by user accounts associated with that department. *Priority 1*

4.7 Sections

R4.7.1 Section Information: Each section of a course scheduled by a department has the following information associated with it which can be modified by users:

- **R4.7.1.1 Scheduled Course** *Priority 1*
- **R4.7.1.2 Scheduled Time** *Priority 1*
- **R4.7.1.3 Term** *Priority 1*
- **R4.7.1.4 Section number** *Priority 1*
- **R4.7.1.5 Instructor** *Priority 1*
R4.7.1.6 Location Priority 1

R4.7.1.7 Corequisite Sections Priority 1

R4.7.2 Scheduling a Section: When a new section is added by a user if any of the following constraints are violated a notification will be displayed to the user indicating the error:

R4.7.2.1 The course scheduled defines corequisites but no corequisite sections are defined Priority 1

R4.7.2.2 The scheduled location is not marked non-exclusive and has another section scheduled during the same time this section has defined Priority 1

R4.7.2.3 The scheduled instructor is scheduled with another section at the same time Priority 1

R4.7.3 Scheduling Locations: Users associated with a department may only assign locations to sections which are owned by that department. Priority 2

4.8 Data Views

R4.8.1 Calendar View: The default Docket Front End includes a calendar view which displays a weekly agenda view of courses. Priority 1

R4.8.2 Calendar View Filters: The default Docket Front End's calendar view has the option to filter the courses displayed by the following heuristics:

R4.8.2.1 By Department Priority 1

R4.8.2.2 By Instructor Priority 1

R4.8.2.3 By Location Priority 1
**R4.8.3 Detailed Course View:** The Docket Front End includes a detailed view of selected courses which displays the following information:

- **R4.8.3.1 Course Name** Priority 1
- **R4.8.3.2 Department Code** Priority 1
- **R4.8.3.3 Course Number** Priority 1

**R4.8.3.4 Course Section Details:** The following information about every section currently scheduled for this course:

- **R4.8.3.4.1 Section Number** Priority 1
- **R4.8.3.4.2 Scheduled Time** Priority 1
- **R4.8.3.4.3 Scheduled Location** Priority 1
- **R4.8.3.4.4 Assigned Instructor** Priority 1

**R4.8.4 Detailed Instructor View:** The Docket Front End includes a detailed view of selected instructors which displays the following information. Priority 1

- **R4.8.4.1 Instructor Name** Priority 1
- **R4.8.4.2 Courses instructor is preferred to teach** Priority 1

**R4.8.4.3 Assigned Section Details:** The following information about every section the instructor is currently assigned to:

- **R4.8.4.3.1 Section Number** Priority 1
- **R4.8.4.3.2 Scheduled Time** Priority 1
- **R4.8.4.3.3 Scheduled Location** Priority 1
- **R4.8.4.3.4 Scheduled Course** Priority 1
4.9 User Subscriptions

**R4.9.1 Adding a Subscription:** The Docket application allows users to subscribe to courses, instructors, and scheduled sections to be notified by email when the subscribed object is modified. *Priority 1*

**R4.9.2 Forwarding Address:** The Docket application allows users to specify the contact information used when the application sends them notifications. *Priority 1*

**R4.9.3 Cancelling a Subscription:** The Docket application allows users to cancel an existing subscription associated with their account. *Priority 1*

5 Nonfunctional Requirements

5.1 Portability

**N5.1.1 Database Portability:** The Docket software system's database support is limited to the following relational database systems:

- **N5.1.1.1 PostgreSQL 9.1** *Priority 1*
- **N5.1.1.2 Microsoft SQL Server 2008** *Priority 2*
- **N5.1.1.3 Oracle 11g** *Priority 2*
- **N5.1.1.4 DB2 9.7** *Priority 2*

**N5.1.2 Communication Portability:** The Docket Back End data format is limited to the following standards:

- **N5.1.2.1 JSON** *Priority 1*
5.2 Maintainability

N5.2.1 Modularity: For long-term maintainability, the software is broken into a client user interface and a scheduling service API. **Priority 1**

N5.2.2 Test Coverage: The scheduling service API will have automated testing coverage. **Priority 1**