

SPECTRUM HEALTH



Log Monitoring Compliance

Project Documentation – Fall 2011

Spectrum Health Representatives

Tammy Rhoades

Brett Hoffman

Michigan State University Capstone Students

Kathryn Bonnen

Collin Lotus

Will Seeger

Wayne Stiles

TABLE OF CONTENTS

1. Executive Summary	4
2. Core Functionalities	5
Primary usage and abilities of log manager	
2.1 Collect Items for Review	
2.1.1 Automated Parser	
2.1.2 Manual Entry	
2.2 Review Processing	
2.3 Dashboard	
2.4 Email Notifications	
2.5 Audit Export	
2.6 System Administrator Role	
3. Design and Implementation.....	6
Details the design and system experience.	
3.1 System Architecture	
3.1.1 Server Specifications	
3.1.2 Browsers Supported	
3.2 Database (Central Storage)	
3.3 User Interface	
3.3.1 Login	
3.3.2 Navigation Sidebar	
3.3.3 Quick Status	
3.3.4 All Reviews	
3.3.5 Pending/Past Due Reviews	
3.3.6 Create Audit	
3.3.7 Audit Export	
3.3.8 Dashboard	
3.3.9 System Administrator Role	
3.3 Parser	
3.5 Email Notifications	
4. Future Functionalities.....	18
Vision for future features and functionality.	
4.1 Future Vision	
4.2 Scorecard Features	
4.2.1 Compliance Drill-Down Functionality	

4.2.2 Trending Analysis across Systems

4.3 Systems and Audit Types

5. Development Standards..... 19

Vision for future features and functionality.

5.1 Code Conventions and Style

5.1.1 FxCop

5.1.2 StyleCop

5.2 User Interface

5.3 HTML

5.4 JavaScript

5.5 CSS Framework

5.6 Accessibility

6. Timetable 21

7. Point of Contact..... 23

8. Acknowledgements..... 24

1. EXECUTIVE SUMMARY

Spectrum Health Systems is a not-for-profit health system located in Western Michigan. It provides complete medical care through its nine hospitals, the Spectrum Health Medical Group, West Michigan Heart, physicians with over 600 providers, and their health plan, Priority Health.

Healthcare systems manage large quantities of extremely sensitive data. Monitoring access to this information is extremely important since it keeps medical information safe and helps Spectrum Health Systems stay in compliance with federal regulations.

The existing system for log monitoring compliance at Spectrum Health consists of a number of separate audit log control lists. Users gather information needed and perform review analyses manually. This system is not sufficient to effectively handle the volume of log data managed by Spectrum Health.

The Log Monitoring Compliance System will provide a unified approach to log management at Spectrum Health, facilitating efficient audit review processes and automated scorecard reporting for senior level leadership.

2. CORE FUNCTIONALITIES

2.1 Collect Items for Review

The database is a central storage area for reviews gathered from an automated parser, as well as records entered into the system manually via the user interface. This dual functionality will assist in future expansion of log monitoring compliance.

2.1.1 Automated Parser

For the systems and audit types with fully specified log policies, an automated data parser consumes the log files, determines records requiring review and adds the records and data about the log file to the central database.

2.1.2 Manual Entry

For easier integration as the policies surrounding the remaining systems and audit types are determined, manual entry is supported as an in-between measure. Audit conclusions, which may consist of multiple records, can be added to the database.

2.2 Review Processing

Designated reviewers can keep up-to-date information regarding a specific review within the web application. Once a review is complete, the reviewer marks it as closed. Reviewers also have the ability to select a proxy to complete their reviews for them.

2.3 Dashboard

The Dashboard contains scorecards which provide information about compliance and discrepancies via high-level visual summaries for each of the individual system-audit type combinations.

2.4 Email Notifications

The system will remind reviewers via email if they have pending reviews due within 2 days, so they can manage their time effectively.

2.5 Audit Export

All the records within the database can be exported into a single CSV file when desired, to show system compliance to auditors. The data is filtered by a start and end date, and by which system(s) and audit type(s) are to be exported.

2.6 System Administrator Role

Although reviewers and directors have the same visibility and permissions in the system, any user designated as a system administrator will have permissions in the database exceeding those of other employees. A system administrator can add, modify, and delete from all tables in the database.

3. TECHNICAL AND DESIGN SPECIFICATIONS

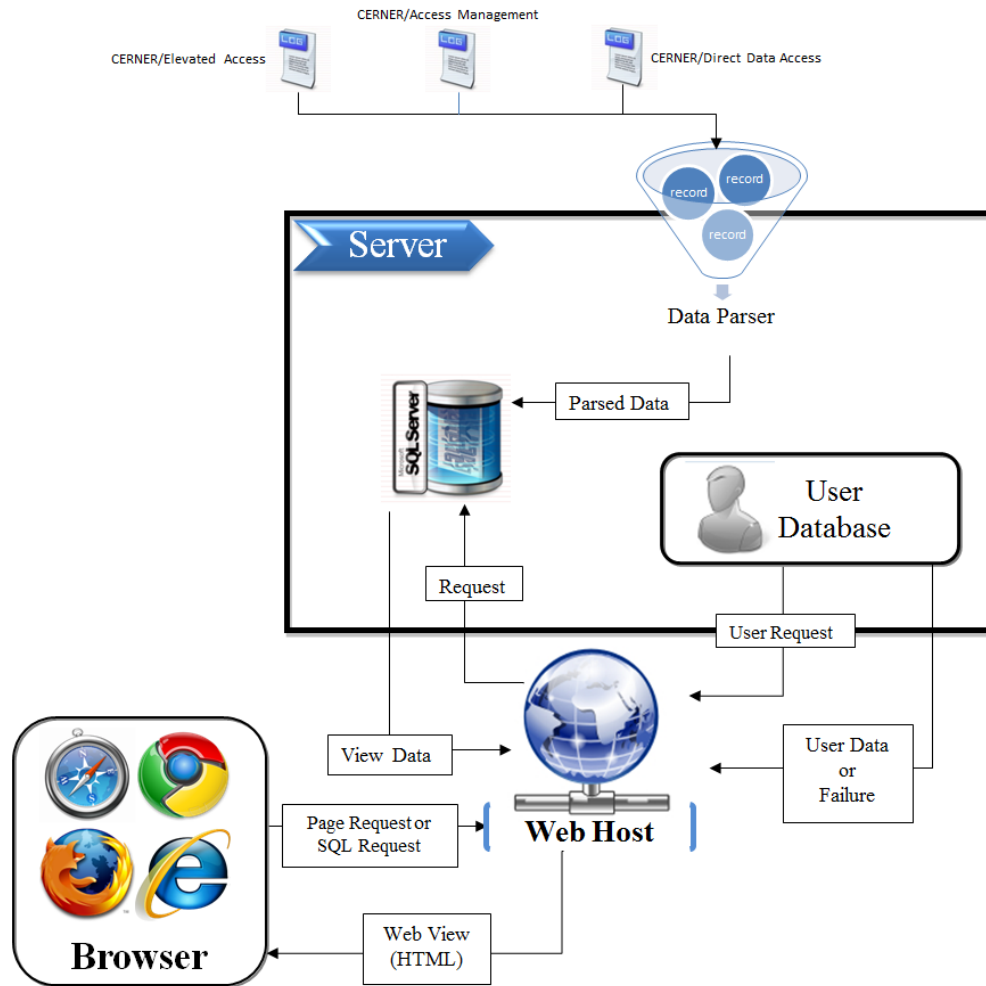


Fig 3.1 Diagram of System Architecture

3.1 System Architecture Overview

The Log Monitoring Compliance system is comprised of two major components: a web application/user interface and a backend database. Users log in using their preferred browser. The web host then pulls data from the database to display to the user. On the server side, the data parser adds records to the database and an email notification service runs, notifying clients when they have reviews due.

3.1.1 Server Specifications

The system runs on Windows Server 2008 using Microsoft SQL Server 2008, and requires Active Directory for verification.

3.1.2 Browsers Supported

- Internet Explorer 7, 8; This browser is the default in Windows operating system. Windows operating systems and Microsoft technology is the standard technology employed by Spectrum Health.

- Mozilla Firefox 3,4,5+; This is a relatively popular alternative to Internet Explorer.
- Google Chrome; Chrome is quickly becoming a heavily used browser for its speed and performance.
- Safari; Safari is the preloaded browser on Mac OS systems.

3.2 Database (Central Storage)

The database is a central storage with the main purpose of storing the reviews (automatically entered) and audit conclusions (manually entered). Below is an illustration of how the data is stored.

- The Reviews inherit from LogFiles, LogTypes, AuditTypes and Systems.
- Both AuditConclusions and Reviews have Histories.
- Both AuditConclusions and Reviews can have Tickets, Attachments.

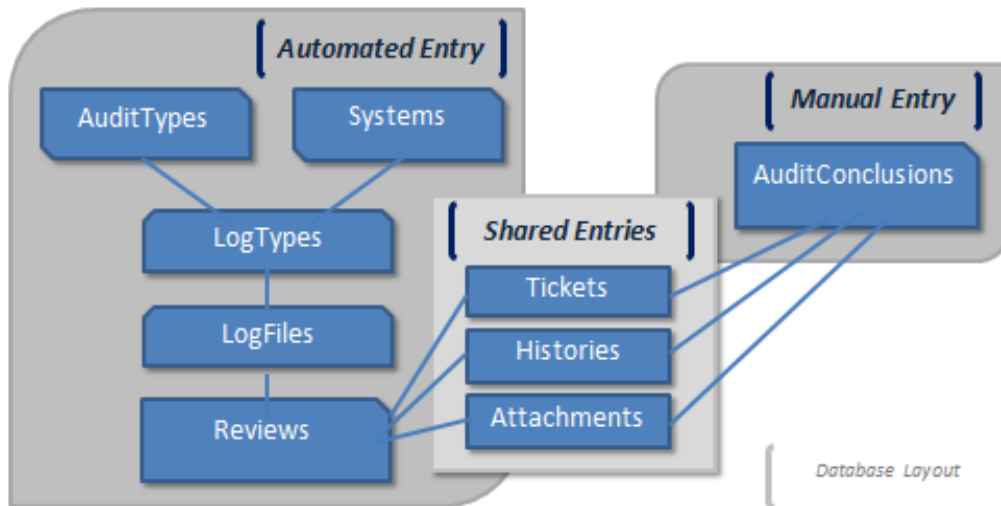


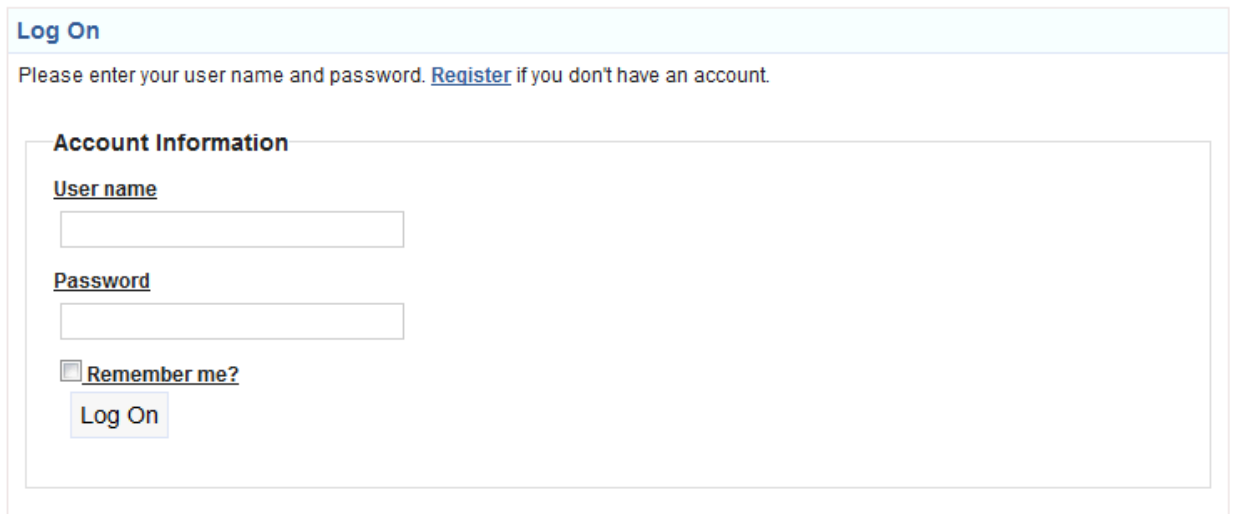
Fig 3.2 Diagram of Central Data Store

3.3 User Interface

The user interface is a crucial part of the system, as it provides the reviewers convenient access to the system. It provides ways to easily filter reviews to find a user's pending reviews, a process previously done manually. The following details the different views, which are all accessible from the sidebar navigation panel.

3.3.1 Login

- The login screen allows users to specify username and password, to be authenticated and gain access to the application.
- New users are added through an alternate user database; the application itself will not allow registration of new users.
- If a user submits invalid credentials, the application will display, "Incorrect username or password". The user will be allowed to resubmit credentials.
- If a user submits valid credentials, the application displays their default view.
- The application will mask the password field. It will validate username and password by sending an encrypted request to the server. After a failed login attempt, the password field will be cleared.



The screenshot shows a web form titled "Log On". Below the title is a light blue header bar. Underneath the header, there is a text prompt: "Please enter your user name and password. [Register](#) if you don't have an account." The main form area is enclosed in a light blue border and contains the following elements: a section header "Account Information", a label "User name" above a text input field, a label "Password" above a text input field, a checkbox labeled "Remember me?", and a "Log On" button.

Fig 3.3 Login View

3.3.2 Navigation Sidebar

The navigation bar allows users to navigate to the various views and functions in the application. Each of the entries ('Pending/Past Due', 'Dashboard', 'Create Audit', 'All Reviews', and 'Audit Export') corresponds to a view in the application. The 'About' section provides a brief description of the application.

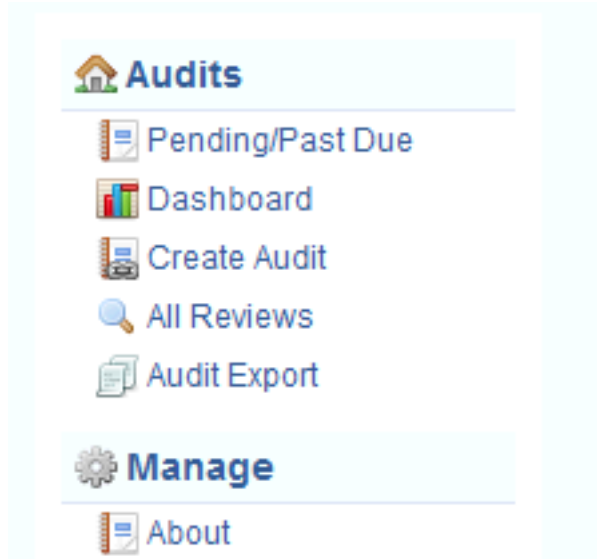


Fig 3.4 Navigation Sidebar

3.3.3 Quick Status

Quick Status contains links that will quickly redirect the user to frequently used functionality. The main content of the Quick status is a link that redirects users to their list of Active/Open reviews.

- Audit Export Button: Located on the light blue bar, the audit export button redirects users to the Audit Export View.
- Create Audit Button: Located on the light blue bar, the create audit button redirects users to a screen where they can create an audit conclusion.

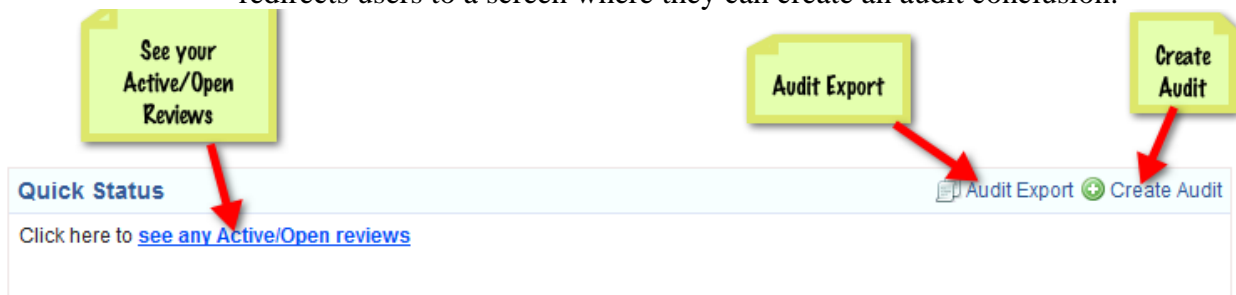


Fig 3.5 Quick Status

3.3.4 All Reviews

The user has the option to view all reviews in the system. The list of reviews can be sorted by column header and displayed on multiple pages if needed. This list of reviews can also be searched. From this page a Reviewer can access the details view and edit view for a particular review.

The attributes listed in this view of reviews are:

- *Due Date* - Date the review is due.
- *Initiator* - The employee that initiated the exception.
- *Description* – The description of the review.
- *System*
- *Audit Type*
- *Reviewer* – The employee that is reviewing the record.
- *Status* - Whether the review of the record is active or closed.

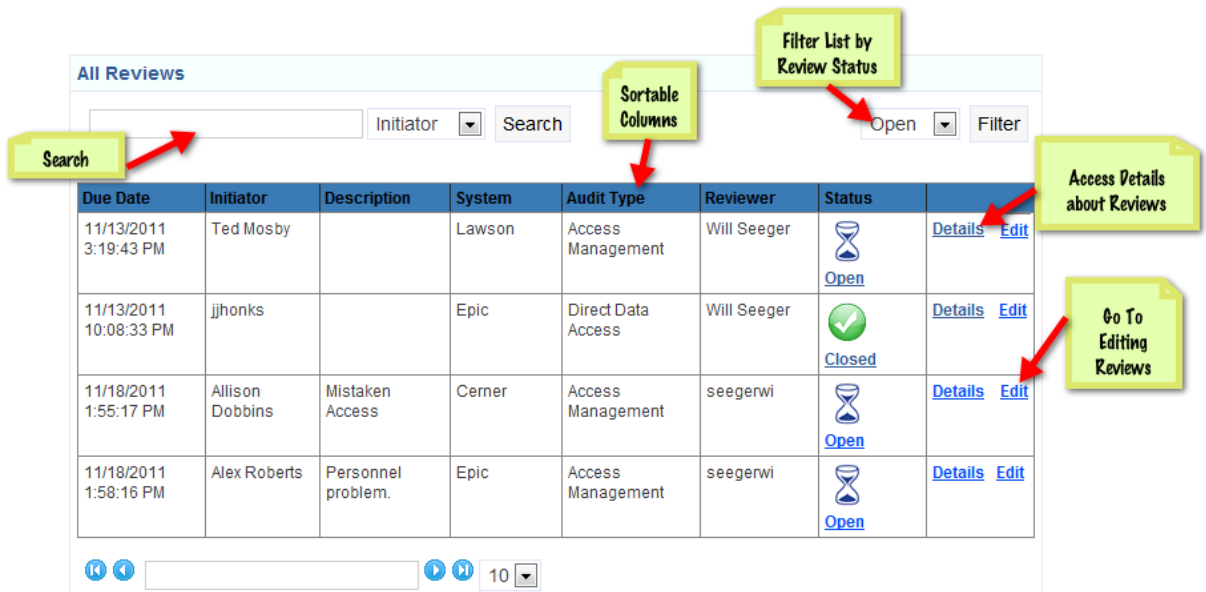


Fig 3.6 All Reviews

3.3.5 Pending/Past Due Reviews

- The pending and past due reviews page shows the user their own reviews that are still open or were not completed on time, so they can view all their obligations on one page.
- This list can be sorted by column and filtered by system. It also contains direct links to the Details view and the Edit view for each review.

The screenshot shows a web interface for 'Pending/Past Due Reviews : seegerwi'. At the top left, there is a link 'Create New Audit Conclusion' with a red arrow pointing to it from a yellow callout box labeled 'Link to Create a New Audit Conclusion'. On the right side, there is a dropdown menu for filtering by system, currently showing 'Lawson', with a red arrow pointing to it from a yellow callout box labeled 'Filter Personal Reviews By System'. The dropdown menu is open, showing options for 'Lawson', 'Epic', and 'Cerner'. Below the dropdown is a 'Filter' button. The main content is a table with columns: Due Date, Initiator, Description, System, Audit Type, Reviewer, Status, and Actions. The table contains two rows of review data.

Due Date	Initiator	Description	System	Audit Type	Reviewer	Status	Actions
11/18/2011 1:55:17 PM	Allison Dobbins	Mistaken Access	Cerner	Access Management	seegerwi	Open	Details Delete
11/18/2011 1:58:16 PM	Alex Roberts	Personnel problem.	Epic	Access Management	seegerwi	Open	Edit Details

Fig 3.7 Past/Pending Item View

3.3.6 Create Audit

The user can enter a record or group of records into the system manually, as an audit conclusion. This is treated comparably to the automated reviews that are generated using the data parser.

Create Review

Audit Conclusion

System

Audit Type

Reviewer's Name

Initiator's Name

Activity Date

Corrective Action

Exception Response

Description

Can enter information into each of the Fields to create an Audit Conclusion


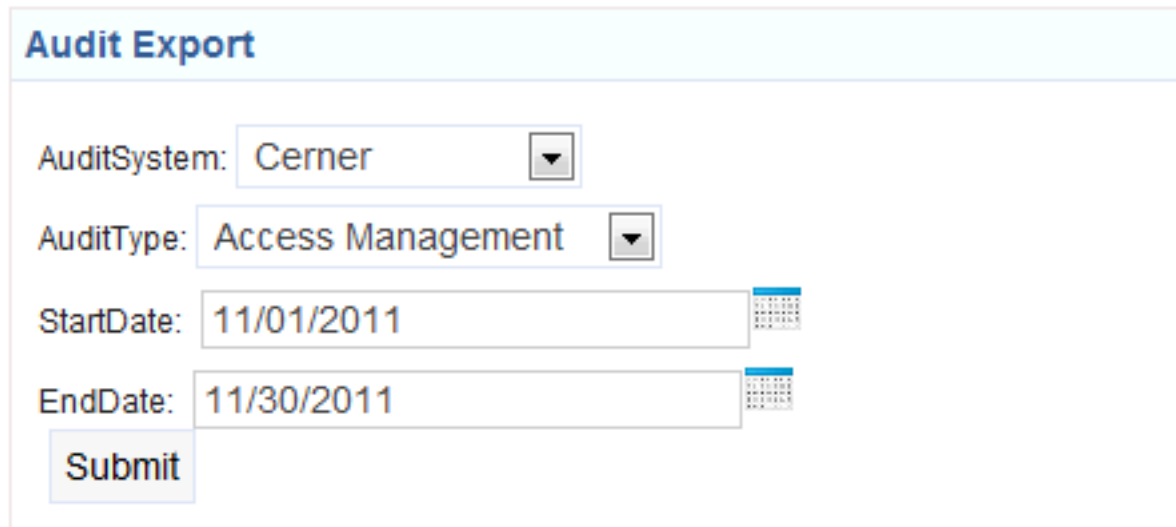


Fig 3.8 Create Audit View

3.3.7 Audit Export

This view will allow the user to export data into a csv file, specifying the System, Audit Type, and a time period (via begin and end dates).



The screenshot shows a web form titled "Audit Export". It contains four input fields and a submit button. The "AuditSystem" field is a dropdown menu with "Cerner" selected. The "AuditType" field is a dropdown menu with "Access Management" selected. The "StartDate" field is a text input with "11/01/2011" and a calendar icon to its right. The "EndDate" field is a text input with "11/30/2011" and a calendar icon to its right. Below the date fields is a "Submit" button.

Fig 3.9 Audit Export

3.3.8 Dashboard

The Dashboard provides a visual summary of the health of the system as a whole. It includes three types of scorecards: Compliance, Discrepancy and Discrepancy by System Comparison.

Definitions

- *Compliance*: A review that is executed within the due date.
- *Discrepancy*: An event requiring review in which there is an actual problem (e.g. a process was not followed)

Compliance Scorecard

This scorecard summarizes the compliance for a specific System over 13 months. The end month can be changed and the compliance chart updated.

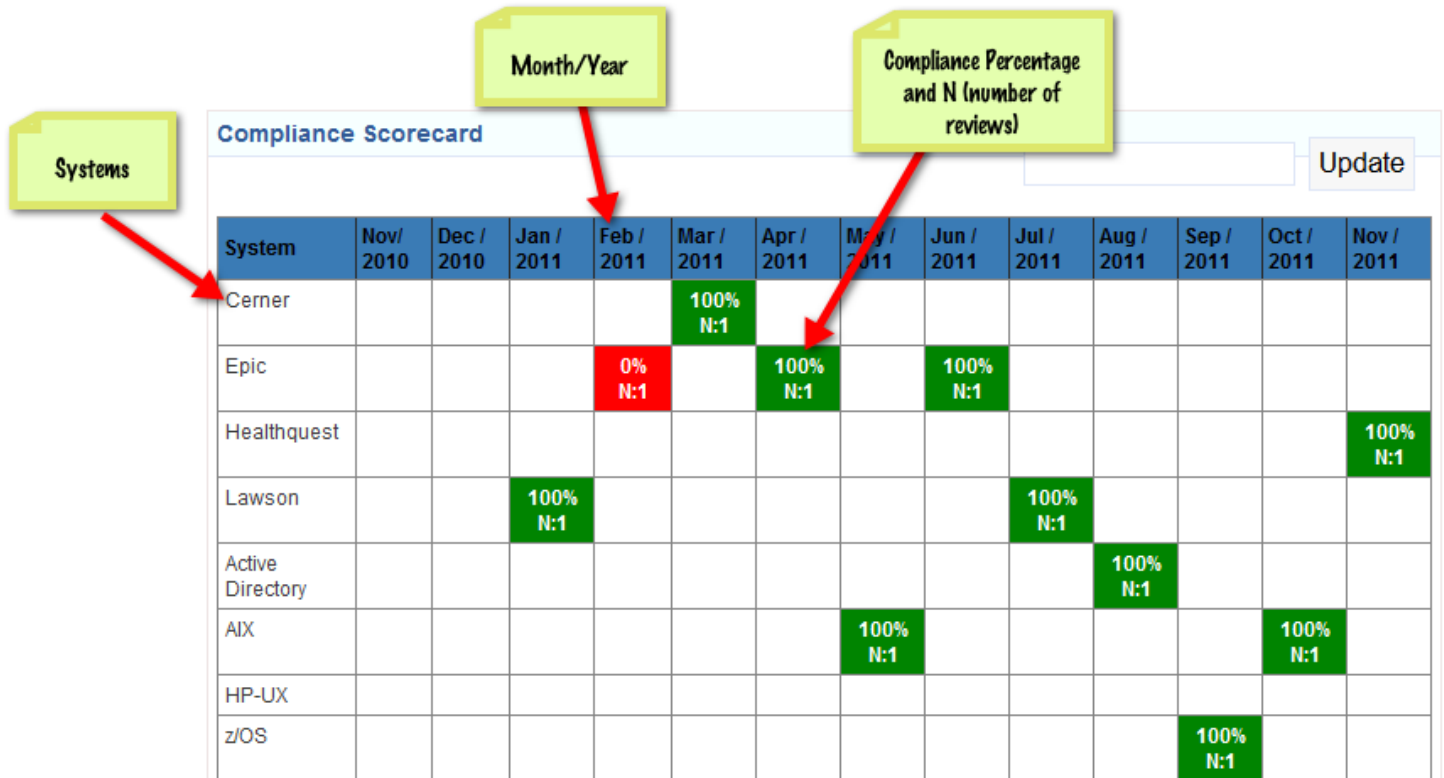


Fig 3.10 Compliance Scorecard

Discrepancies Scorecard

This scorecard displays the discrepancy count for a given System and Audit Type in the form of a bar graph over a 13 month period. The user can indicate the end month, the System and the Audit Type.



Fig 3.11 Discrepancies Scorecard

Discrepancy System Comparison Scorecard

This Scorecard displays the discrepancies for each system in a pie chart form to allow comparison between systems.

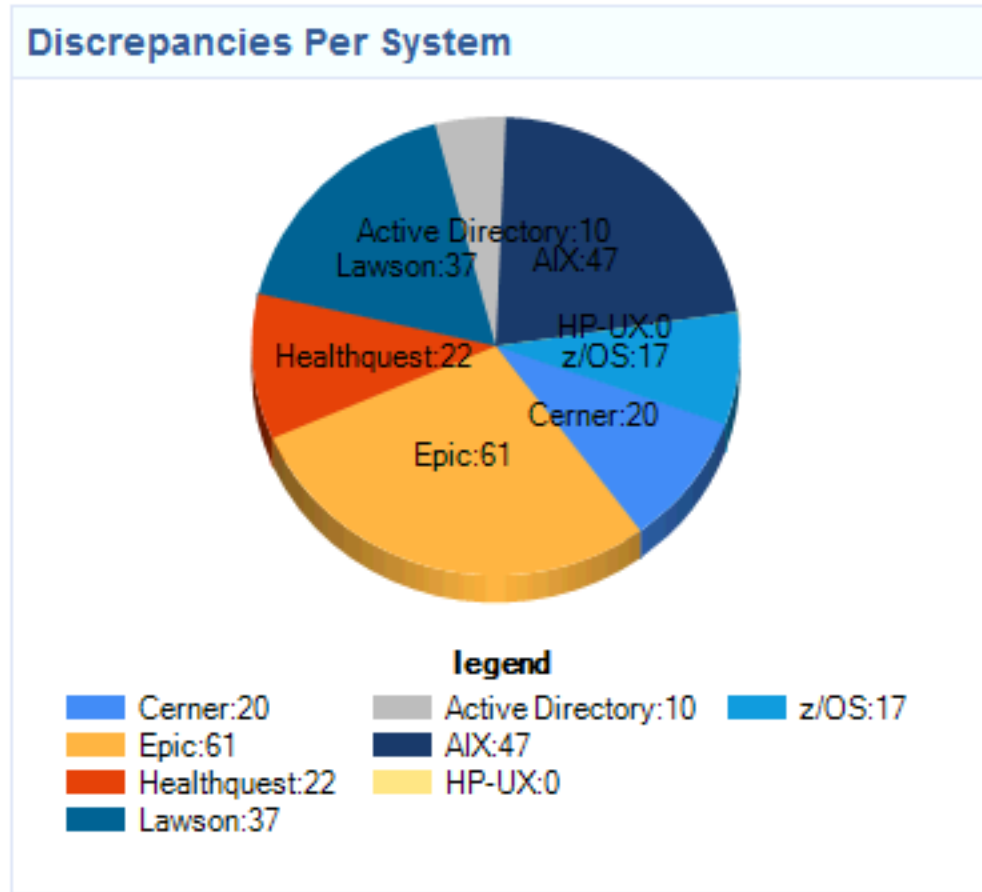


Fig 3.12 Discrepancy System Comparison Scorecard

3.3.9 System Administrator Role

The system administrator can view, add, edit and delete all records in the database. The system administrator role also provides an interface to add and delete from variable tables critical to the data parser.

3.4 Parser

- The automated log file parser reads files created by P2Sentinal in CSV file format from Access Management, Elevated Access activity, and Direct Data Access audit types for the Cerner system.
- It reads the data contained in the CSV file and adds it into the database.
- The parser is designed to be easily extensible and is based on the factory method with LogFile as the abstract creator and the AccessManagement, ElevatedAccess and DirectDataAccess LogFiles that create the product: the Review which can then be inserted into the database.

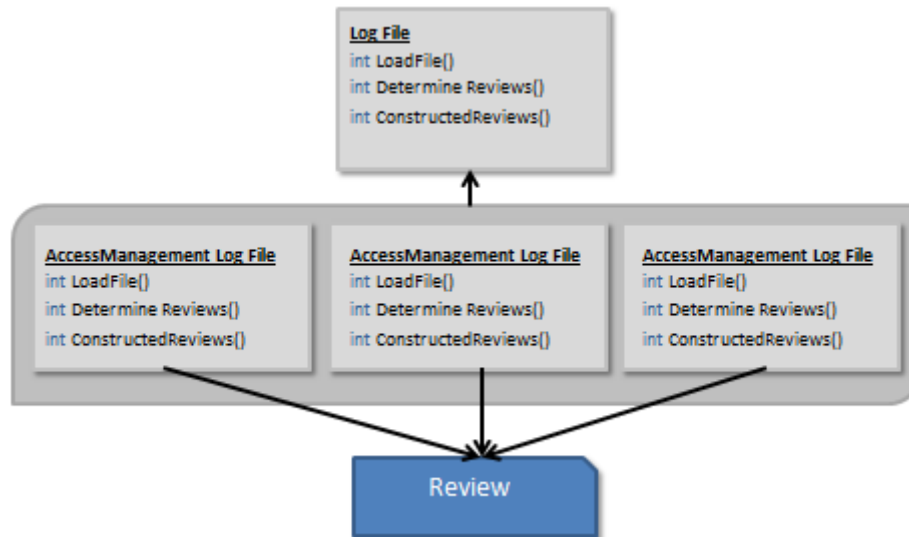


Fig 3.13 Parser Design

3.5 Email Notifications

Email notifications will run as a Windows service on the server, and will check every 24 hours for reviewers with pending reviews due in 2 days. Emails sent to reviewers will include a link back to the system.

4. FUTURE FUNCTIONALITIES

4.1 Future Vision

The future functionalities (the vision for this application) are important to note because it helps to define the rest of the project. Certain design decisions reflect the intention of the client to add the following functionality.

4.2 Scorecard Features

The scorecard is designed to be a higher-level view of the overall health of the Log Monitoring Compliance system.

4.2.1 Compliance Drill-Down Functionality

Clicking on any system or audit type from the scorecard view will show discrepancies for records within that category. Clicking on one of these records takes the user to its review view. In all views, including the record's review view, all areas of discrepancy are highlighted red.

4.2.2 Trending Analysis across Systems

One or more line graphs would show the overall health of a system or audit type over a long-term period. If there are minimums or maximums the data must stay between, these will be marked on the graph.

4.3 Systems and Audit Types

Remaining systems and audit types will be added to the automated data parser system as the business process for these systems is solidified. The application is currently designed to facilitate extendibility of the application to other systems and audit types.

5. DEVELOPMENT STANDARDS

5.1 Code Conventions and Style

- Variable names: camel case
- Classes, properties and methods: Pascal case
- Class data member names: prefixed with underscore
- Curly brackets should fall on the next line of the flow control operator

```
namespace Application
{
    /// <summary>
    /// This class does something.
    /// </summary>
    public class SomeClass
    {
        #region Private Members

        /// <summary>
        /// Holds an integer.
        /// TODO: Horrible documentation.
        /// </summary>
        private int _field;

        #endregion

        #region Methods

        /// <summary>
        /// Retrieves a list of ints.
        /// </summary>
        /// <param name="parameter">Something thats passed to the method.</param>
        /// <returns>Useful collection of ints.</returns>
        public static ICollection<int> GetSomething(int parameter)
        {
            yield return 1;
        }

        #endregion

    }
}
```

Fig 5.1 Example Source Code

5.1.1 FxCop

This is a tool used to maintain code quality. All violations existing should be addressed. See below for an example command (includes commands to suppress certain rules).

```
"C:\Program Files (x86)\Microsoft Fxcop 10.0\FxCopCmd.exe"
```

```
/rid:-Microsoft.Design#CA2210
/rid:-Microsoft.Design#CA1020
/rid:-Microsoft.Design#CA1031
/rid:-Microsoft.Globalizatiion#CA1305
/rid:-Microsoft.Globalizatiion#CA1307
/rid:-Microsoft.Usage#CA2201
/rid:-Microsoft.Performance#CA1822
/rid:-Microsoft.Design#CA1024
```

```
/directory:"c:\Program Files (x86)\Microsoft ASP.NET\ASP.NET MVC 1.0\Assemblies"  
/directory:"c:\Program Files (x86)\Microsoft ASP.NET\ASP.NET MVC 2\Assemblies" /directory:"c:\Program Files  
(x86)\Microsoft ASP.NET\ASP.NET MVC 3\Assemblies" /out:%1 %2 %3 %4 %5 %6 %7 %8 %9
```

5.1.2 StyleCop

- Set of rules for finer control over enabled and customized rules regarding source code standards.
- Resolving violations are ‘highly recommended’ but non-blocking.
- Rules provided by Spectrum Health.

5.2 User Interface

- Should use Spectrum Health master pages and framework.

5.3 HTML

- Developed to at least XHTML 1.0 Transitional standard
- Must be validated. (<http://validator.w3.org/>)

5.4 JavaScript

- JQuery should be primary JavaScript framework

5.5 CSS Framework

- Developed using Spectrum’s common CSS framework.
- Must be validated. (<http://jigsaw.w3.org/css-validator>)

5.6 Accessibility

- Verify problematic areas in accessibility (<http://www.w3.org/TR/WCAG10/full-checklist.html>)

6. TIMETABLE

Week 1 (9/18-9/24)

- Project Plan
 - Mockups
 - Database Schema (Iteration 1)
- Practice Project Plan presentation
- System Configuration Proof of Concept Website

Week 2 (9/25-10/1)

- Database Schema (Iteration 2)
- Log File Parser planned
- Email Notifications Proof of Concept
- Practice Project Plan Presentation
- “Create Review” view prototyped
- Project Plan Presentation

Week 3 (10/2-10/8)

- Database Schema Finalized
- Log File Parser prototyped
- “Edit Review” view, “Create Review”, “Detail”, “List” view completed for Audit Conclusions (manual entry side)

Week 4 (10/9-10/15)

- Log File Parser Debugging
- Email Notifications Executable Debugging
- “Edit Review”, “Detail”, “List” views incorporating Reviews (automated side)
- Alpha presentation drafted and practiced

Week 5 (10/16-10/22)

- “Manage” view completed
- “Scorecard” view completed
- 10/19 Alpha Presentation

Week 6 (10/23-10/29)

- Log File Parser Completed
- Polish all views
- System Admin Functionality completed
- Debugging

Week 7 (10/30-11/5)

- Complete and Practice Beta presentation
- Continue practicing Beta presentation
- All remaining features

Week 8 (11/6-11/12)

- Documentation
- Debugging

Week 9 (11/13-11/19)

- 11/14 Beta Presentation
- Design and storyboard making project video
- Debugging

Week 10 (11/20-11/26)

- Begin filming and recording project video
- Finish debugging

Week 11 (11/27-12/3)

- 11/29 Final Project Presentation at Spectrum Health
- Complete Project Video
- Make and practice Design Day presentation

Week 12 (12/4-12/10)

- Continue practicing Design Day presentation
- 12/9 Design Day Presentation

7. POINT OF CONTACT

For further information concerning this document and project, please contact **Professor Wayne Dyksen** at Michigan State University (dyksen@cse.msu.edu). All materials in this document have been sanitized for proprietary data.

8. ACKNOWLEDGMENTS

We would like to acknowledge Spectrum Health Systems for their contribution to the Senior Design program at Michigan State. Specifically, we would like to thank Tammy Rhoades, Brett Hoffman, and Sergey Stoma for dedicating much of their time to ensuring this project's success. Additional thanks goes to Mary Delrue, Gary Lacher, Jeremy Meller and Patrick O'Hare for their support.

Furthermore we would like to thank the Michigan State University Department of Computer Science, Dr. Wayne Dyksen and Meredith Schmidt for their resources and support.