Safety Incident Reporting and Root Cause Analysis Application

by

Mikki Tran, Saori Hiratsuka Godfrey, Serhiy Shakhurdin

Submitted to the Faculty of the School of Information Technology
In Partial Fulfillment of the Requirements for
The Degree of Bachelor of Science
In Information Technology

Mikki Tran — Date: 4/18/16

Saori Hiratsuka Godfrey — Date: 4/19/2016

Serhiy Shakhurdin — Date: 4/18/2016

James Scott, Faculty Advisor — Date: 4/18/2016

Robin Carew, Faculty Advisor — Date: 4/18/16

University of Cincinnati

College of Education, Criminal Justice, and Human Services

April 2016

Table of Contents

List of Figures

Abstract

Safety incidents can happen to anyone and often have devastating consequences to both employee and employer. They need to be recorded and investigated to determine root causes, then acted upon in a timely and organized manner. Unaddressed incidents can recur, becoming very costly.

To that end, our team — consisting of Saori Hiratsuka Godfrey, Mikki Tran, and Serhiy Shakhurdin — envisioned a web application: Safety Incident Reporting (SIR). Accounts allow authorized users to view and manage safety incident data. Additionally, the application generates reports and action plans to help guide users in improving and maintaining safety. An email alert system reminds users to complete tasks as well as facilitate knowledge-sharing. The trio developed this application using Microsoft technologies (ASP.NET MVC, SQL, etc.).

Introduction

Workplace accidents are common. According to the US Bureau of Labor Statistics, there were 3 million non-fatal occupational injuries and illnesses reported by private industry companies in the United States, occurring at a rate of 3.2 cases per 100 full-time workers. Workplace injuries accounted for more than 95 percent of the 3 million cases mentioned. That number only grows when also considering unreported incidents.

These workplace safety incidents "can range from minor injuries to [...] death. Occupational accidents cause frequent loss of life, pain and suffering, lost wages for the injured workers and damage to production facilities and equipment".[1] Besides the harm to workers, accidents are also costly to the employer. The Occupational Safety and Health Administration (OSHA) "estimate[s] that employers pay almost $1 billion per week for direct workers' compensation costs alone", adding that "[t]he costs of workplace injuries and illnesses include direct and indirect costs. Direct costs include workers' compensation payments, medical expenses, and costs for legal services. Examples of indirect costs include training replacement employees, accident investigation and implementation of corrective measures, lost productivity, repairs of damaged equipment and property, and costs associated with lower employee morale and absenteeism".[3]

Accidents strike without warning, wreaking havoc on employees and employers alike. It was important to investigate them and keep track of their occurrences, because "probing into past or historical accidents is a good way to prevent future accidents."[1] The best way to deal with accidents, and the various harmful effects which they cause, is to learn from them and prevent them from reoccurring in the future. In order to accomplish this, the root causes of prior safety incidents needed to be determined and addressed.

"[T]he interaction of work environment, training, and employee factors" result in workplace accidents.[1] Major causes of accidents include stress, fatigue, unsafe act, machineries/tools, design of work place, training procedures, lack of awareness, and moral weakness.[1] These causes needed to be tracked, and this data put in an actionable report, to prevent accident reoccurrence.

Problem Statement

XYZ Company is a manufacturing company with multiple facilities across the US. It has thousands of employees working in various locations. The amount of employees in locations vary from 50-500. XYZ Company is facing multiple challenges when it comes to managing their safety incidents:

- Safety incidents were not being tracked at a corporate level. They were only tracked at individual facility level utilizing Microsoft Office environment (Word, Excel).
- Safety incident information was not being shared across the facilities of the company. This led to similar incidents happening at different facilities, placing a higher financial burden on the company.
- CAPA (Corrective and Preventive Action) action plans were not formally laid out and tracked. Action plan owners sometimes forgot their responsibility and were not being held accountable.

Solution

We delivered an application that will track safety incident reports. The minimum deliverable was an application with an interface which allows reading and writing to a database of safety incident reports. Additionally, this application would lead the user through root cause analysis process, generate action plans and reports, and send alerts as needed.

Project Scope

- Database of employee information, safety incidents, and root cause of incidents
- User interface allowing interaction with database
    - Inserting, updating, and reading data
- Generate reports based on entered data
    - Safety incident frequency by employee, department, or site
    - Safety incident frequency by equipment, injury type
    - Safety incident frequency by cause --- “Areas of Improvement”
- Different logins have different permissions to manage data
- Alert system

User profile

The users of the application would be limited to employees involved with safety reporting and assessment within the company, and not all employees of the company. User experience requirements are listed in Figure 1, as is the frequency of usage which would be expected.

Potential Users: Only certain employees of the company (Site managers, supervisors, and administrative staff). These users have different roles/permission levels: Manager and Administrator.
Software and Interface Experience: Users were expected to have only basic internet skills—checking email, browsing to a website, logging in, navigating a simple menu, filling a form, etc.
Experience with Similar Applications: There are similar applications. However, no experience with similar application was required.
Task Experience: The task experience for the user included entering information into a web form and reading information from the application sent via email. Knowledge of "5 Whys" or "Why-Why" analysis preferred for the root cause analysis feature of the application.
Frequency of Use: SIR was intended to be involved in multiple parts of a safety routine. Whenever a safety incident occurs, information should be recorded into the database. Ideally, this would occur with less frequency as time goes on and the reports generated by SIR help to improve safety conditions. Also, the frequency of usage of the safety incident recording feature would vary depending on how safe the worksite is. Reports generated and viewed on demand. Ideally, report review would occur frequently, to improve and maintain safe conditions.
Key Interface Design Requirements that the Profile Suggests: Browse to website using any browser. Log in. Menu options customized depending on permission levels. Also, SIR sends emails, which serve as reminders and alerts. • Login • Input information into forms via keyboard • View data as a report • Alerts sent to email addresses

Figure 1: User profile

Use Case Diagram

There are 5 types of user who interact with this application: company senior managers, site managers, safety manager or site supervisor, human resource administrators, and IT administrators. Each user type interacts with the application differently. Their roles in the usage of the application are listed below in Figure 2.

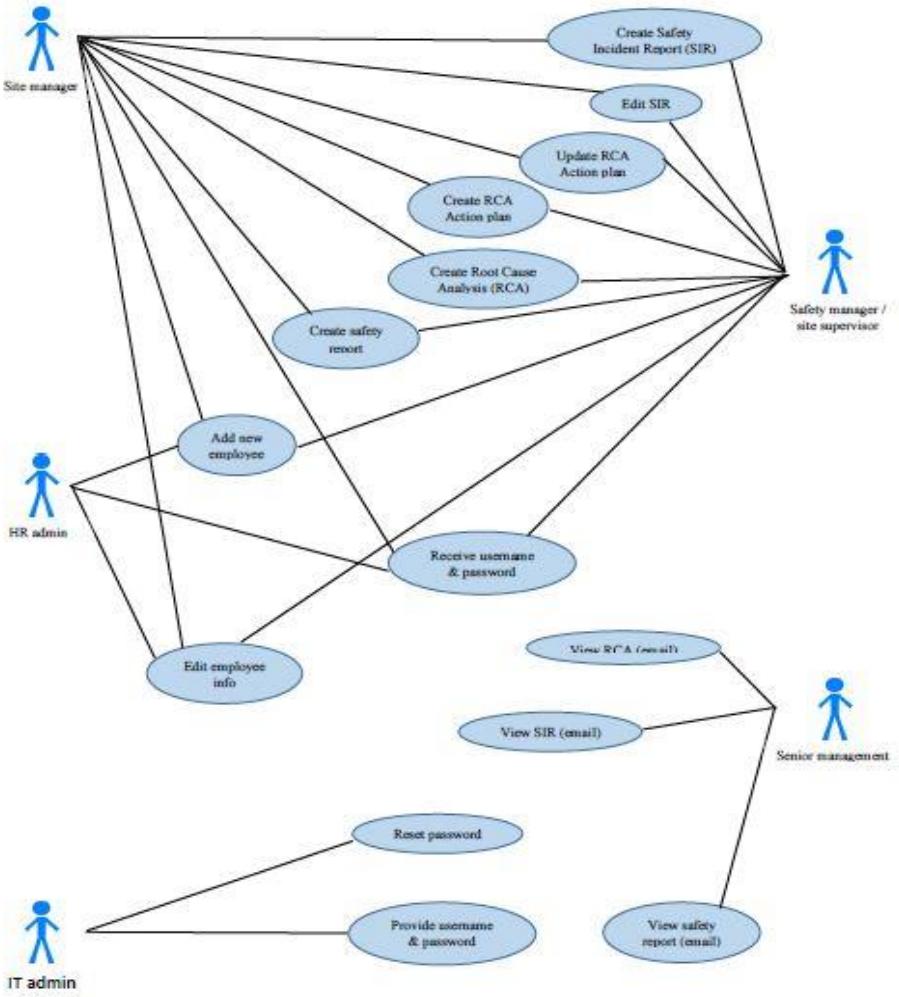


Figure 2: Use case diagram

Database Diagram

A SQL database was utilized for the storage of the application's data. The available data tables and their relationships are shown in Figure 3.

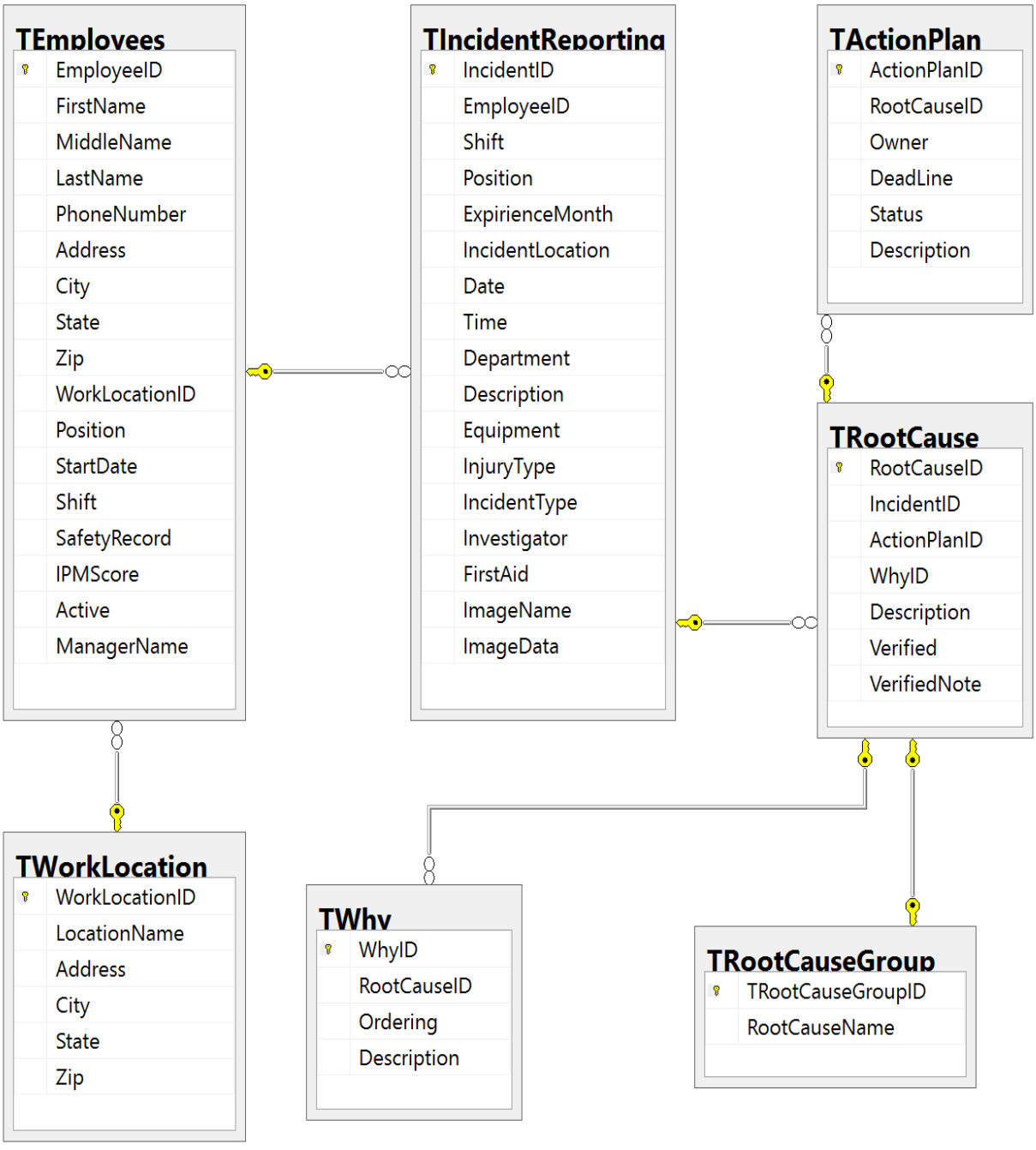


Figure 3: Database diagram

Timeline

The project was broken into multiple deliverables, with a different owner for each of them. Each of these deliverables, as well as the details of the deliverables, are listed in Figure 4.

Task Name	Duration	Start	Finish	Resource Names
Generate project idea	8 days	Mon 8/31/15	Wed 9/9/15	Mikki,Serhiy,Saori
Getting project approved	6 days	Thu 9/10/15	Thu 9/17/15	Mikki,Saori
Determine business requirements	12 days	Fri 9/18/15	Mon 10/5/15	Mikki
Set up development environment	18 days	Thu 9/10/15	Mon 10/5/15	Serhiy
Set up work flow	12 days	Fri 9/18/15	Mon 10/5/15	Mikki
Progress report 1	2 days	Fri 9/18/15	Mon 9/21/15	Mikki,Saori,Serhiy
Project Gantt chart	2 days	Fri 9/25/15	Mon 9/28/15	Mikki
Elevator speech	2 days	Fri 9/25/15	Mon 9/28/15	Saori
Complete team contract	2 days	Fri 10/2/15	Mon 10/5/15	Mikki,Saori,Serhiy
Develop UI/UX	**48 days**	**Tue 10/6/15**	**Thu 12/10/15**	**Saori,Serhiy**
UX	7 days	Tue 10/6/15	Wed 10/14/15	Saori
User login UI	7 days	Wed 10/14/15	Thu 10/22/15	Saori,Serhiy
Employee Portal UI	7 days	Fri 10/23/15	Mon 11/2/15	Saori,Serhiy
Incident Reporting UI	7 days	Tue 11/3/15	Wed 11/11/15	Saori,Serhiy
Root Cause Analysis UI	7 days	Thu 11/12/15	Fri 11/20/15	Saori,Serhiy
Action Plan UI	7 days	Mon 11/23/15	Tue 12/1/15	Saori,Serhiy
Reporting Output UI	7 days	Wed 12/2/15	Thu 12/10/15	Saori,Serhiy
Misc. Programming	**111 days**	**Tue 10/6/15**	**Tue 3/8/16**	**Serhiy**
Develop database	21 days	Tue 10/6/15	Tue 11/3/15	Serhiy
Look into mobile option	90 days	Wed 11/4/15	Tue 3/8/16	Serhiy,Mikki,Saori
Develop business logic	**42 days**	**Tue 10/6/15**	**Wed 12/2/15**	**Mikki**
User login	7 days	Tue 10/6/15	Wed 10/14/15	Mikki
Employee Portal	7 days	Thu 10/15/15	Fri 10/23/15	Mikki
Incident Reporting	7 days	Mon 10/26/15	Tue 11/3/15	Mikki
Root Cause Analysis	7 days	Wed 11/4/15	Thu 11/12/15	Mikki
Action Plan	7 days	Fri 11/13/15	Mon 11/23/15	Mikki
Reporting Output	7 days	Tue 11/24/15	Wed 12/2/15	Mikki
Project abstract	2 days	Fri 10/9/15	Mon 10/12/15	Saori
Create user profile	2 days	Fri 10/9/15	Mon 10/12/15	Mikki,Saori,Serhiy
Final problem statement	2 days	Fri 10/16/15	Mon 10/19/15	Mikki,Saori
Progress report 2	2 days	Fri 10/23/15	Mon 10/26/15	Mikki,Saori,Serhiy
Use case diagram	2 days	Fri 10/23/15	Mon 10/26/15	Saori
Draft report	2 days	Fri 10/30/15	Mon 11/2/15	Mikki,Saori
Final draft report	7 days	Fri 11/20/15	Mon 11/30/15	Mikki,Saori,Serhiy
Prepare for oral presentation	6 days	Mon 11/9/15	Mon 11/16/15	Mikki,Saori,Serhiy
Application testing	14 days	Wed 3/9/16	Mon 3/28/16	Saori
Fixing application bug	5 days	Tue 3/29/16	Mon 4/4/16	Mikki,Saori,Serhiy
Application final testing	5 days	Tue 4/5/16	Mon 4/11/16	Mikki,Saori,Serhiy
Prepare for Tech expo	2 days	Tue 4/12/16	Wed 4/13/16	Mikki,Saori,Serhiy

Figure 4: Project timeline

Proposed Budget

The application was developed by students as a school project, utilizing various free license software provided by University of Cincinnati. As such, no cost was incurred during the development of this project. However, estimation of cost for development by a software company would have been significantly higher as shown in Figure 5. The software used for this project has a potential commercial cost of $2196.97. Developing hours cost estimation took into consideration that there was a total of 504 hours spent at a rate of $31.25/hr. This resulted in a potential labor costs of $15,750. The estimated total cost to develop this application would be $17,946.97.

Item	Quantity	Price per unit	Project cost	Potential cost
MS Visual Studio 2015	3	499.00	0	1497.00
MS SQL Server	1	699.97	0	699.97
Developing hours	504	31.25	0	15,750.00
Total			0	17,946.97

Figure 5: Budget comparison (actual vs.potential)

Testing plan

Overview

This section will explain the testing methodology for the SIR application (to run on Windows or Mac laptop environment). The following individuals will use this section:

- Developers
- Project Managers
- Outside Testers
- Q&A Personnel

Scope

The scope of testing was focused on SIR application's operation on Windows or Mac laptop environment. The test was organized based on the requirements of the application

Objective

The objective of this testing was to ensure that all modules of the source code were functioning as designed. Each developer was responsible for testing the functionality of their own source code. When the application was completed, the testers listed in Figure 6 agreed to test the application utilizing the test cases described in Figure 7. These technical experts, listed in Figure 6, provided both programming feedback and business expertise as appropriate.

Name	Role	Title	Company
Victoria Brabbs	Tester	Analyst for the Global Consumer Technology – Testing	Citi
Nancy Ober	Tester	Site Manager – Sharon Woods Innovation Center	P&G
Tim Rymer	Technical Consultant	Senior Software Developer	Red Hawk Technologies
Alex Chastukhin	Technical Consultant	Software Developer	Hobart Service

Figure 6: Experts and testers

Logging test and reporting

Any bug(s) found during the testing by the developers was communicated with the other developers to be fixed immediately. Any bug(s) found during the testing by outside testers were documented by the testers and communicated to the developers. The developers met after the test to decide whether the issue was a bug, or a feature that need further development. All the bugs were fixed with higher priority given to the more severe.

Pass/Fail Conditions

SIR application was required to pass all testing in order to be considered successful and fully operational. Any test with outcome other than listed in Figure 7 was considered as a failure and documented in order for the developer team to fix.

Test Cases

The following test cases were performed by developers and outside testers. The steps taken to perform the test and their expected outcome are also listed below in Figure 7. Any result different from the expected outcome was documented and communicated to the developer team.

Category	Test case	Steps to test	Expected outcome
1. Admin setup	Initial admin user set up	1. Open Package Manager console in Visual Studio. 2. Run the "Update-database" command. 3. Log in using the provided credentials.	Initial user with admin rights created
2. User sign up	Sign up for access	1. With application opened, click "Sign up). 2. Fill out sign up form. 3. Click "Submit".	Receive email with confirmation link to application
3. User login	Login as user	1. Open confirmation email. 2. Click on confirmation link. 3. Fill out login form with credentials. 4. Click "Log in".	Login successful
4. User log out	Log out as user	1. After successfully logging in, click "Log out". 2. Click on any links that are part of "Employee Portal" or "Incident Portal".	"Log in" screen is displayed, no access to the "Employee Portal" or "Incident Portal"
5. Security	User lock out	Attempting to log in 5 times with incorrect password.	User locked out, password will need to be reset by admin
6. Usage	Search for Employee	1. Click on the "Employee Portal" option in the menu. 2. Enter a value in the search box and click "Search".	All employees found should have the search value appearing at least once in either their first or last name.
7. Usage	Create Employee	1. Click on the "Employee Portal" option in the menu. 2. Click "Create New" link. 3. Fill information and click "Create".	Employee should be added to the database.
8. Usage	Edit Employee info	1. Click on the "Employee Portal" option in the menu. 2. Click "Edit" link in any of the Employee rows. 3. Update any information and click "Save".	Employee info should be updated in the database.
9. Usage	Search for Safety Incident Report	1. Click on the "Incident Reporting" option in the menu. 2. Enter a value for any of the search criteria and click "Search".	All incidents found should match all of the non-blank search criteria.

Figure 7: Testing plan

10. Usage	Create Safety Incident Report	1. Click on the Incident Reporting option in the menu. 2. Click “Create New” link. 3. Fill in all information and click “Create”.	Incident info should be added to the database.
11. Usage	Edit Safety Incident Report	1. Click on the “Incident Reporting” option in the menu. 2. Click “Edit” link in any of the Incident rows. 3. Update any information and click “Save”.	Incident info should be updated in the database.
12. Usage	Create Root Cause Analysis	1. Click on the “Incident Reporting” option in the menu. 2. Click “Details” link in any of the Incident rows. 3. Click “RCA”. 4. Follow instruction to fill out RCA and click “Save” or “Save/Next” when prompted. 5. Click “Verified” option at the end.	Root cause analysis info should be added to database
13. Usage	Edit Root Cause Analysis	1. Click on the “Incident Reporting” option in the menu. 2. Click “Details” link in any of the Incident rows. 3. Click “RCA”. 4. Follow instruction to edit RCA and click “Save” or “Save/Next” when prompted. 5. Click “Verified” option at the end.	Root cause analysis info should be updated to database
14. Usage	Create Action Plan	1. Click on the “RCA” option in the menu. 2. Click “Details” link in any of the Incident rows. 3. Click “Action plan”. 4. Follow instruction to create action plan and click “Save”.	All owners of Action plan tasks receive email with their task’s details
15. Usage	Email notification	1. Click “Details” link in any of the Incident or RCA rows 2. Click “Send email”.	Email with Incident or RCA details sent to predetermined list of recipients
16. Usage	Chart/graph	1. Click on the Incident Reporting option in the menu. 2. Click “Visual Report” link. 3. Select the filtered categories and click “Create report”.	Graph showing the data pulled from database

Figure 7: Testing plan (cont.)

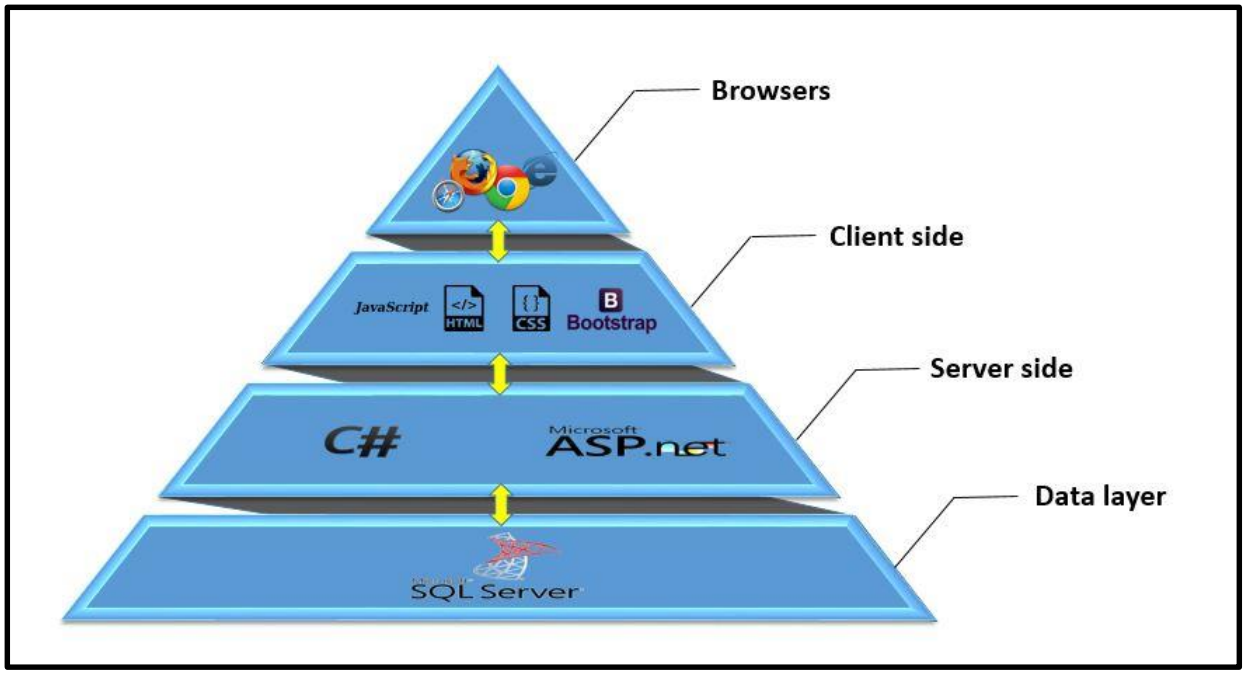
Conclusion

Workplace accidents are a serious matter, and it is common wisdom that those who don't learn from history are doomed to repeat it. This means safety incidents need to be investigated thoroughly, to discover the underlying root causes. This data can serve as a learning source, be analyzed and acted upon to prevent similar incidents in the future. SIR is intended to be a tool that helps companies learn from previous experiences and act upon the data now, to prevent future mishaps. It is not a band aid solution, but meant to be a meaningful part of a safety routine, being a log for incidents, an aid for the critical analysis of an adverse safety situation, and a source of action and understandable data.

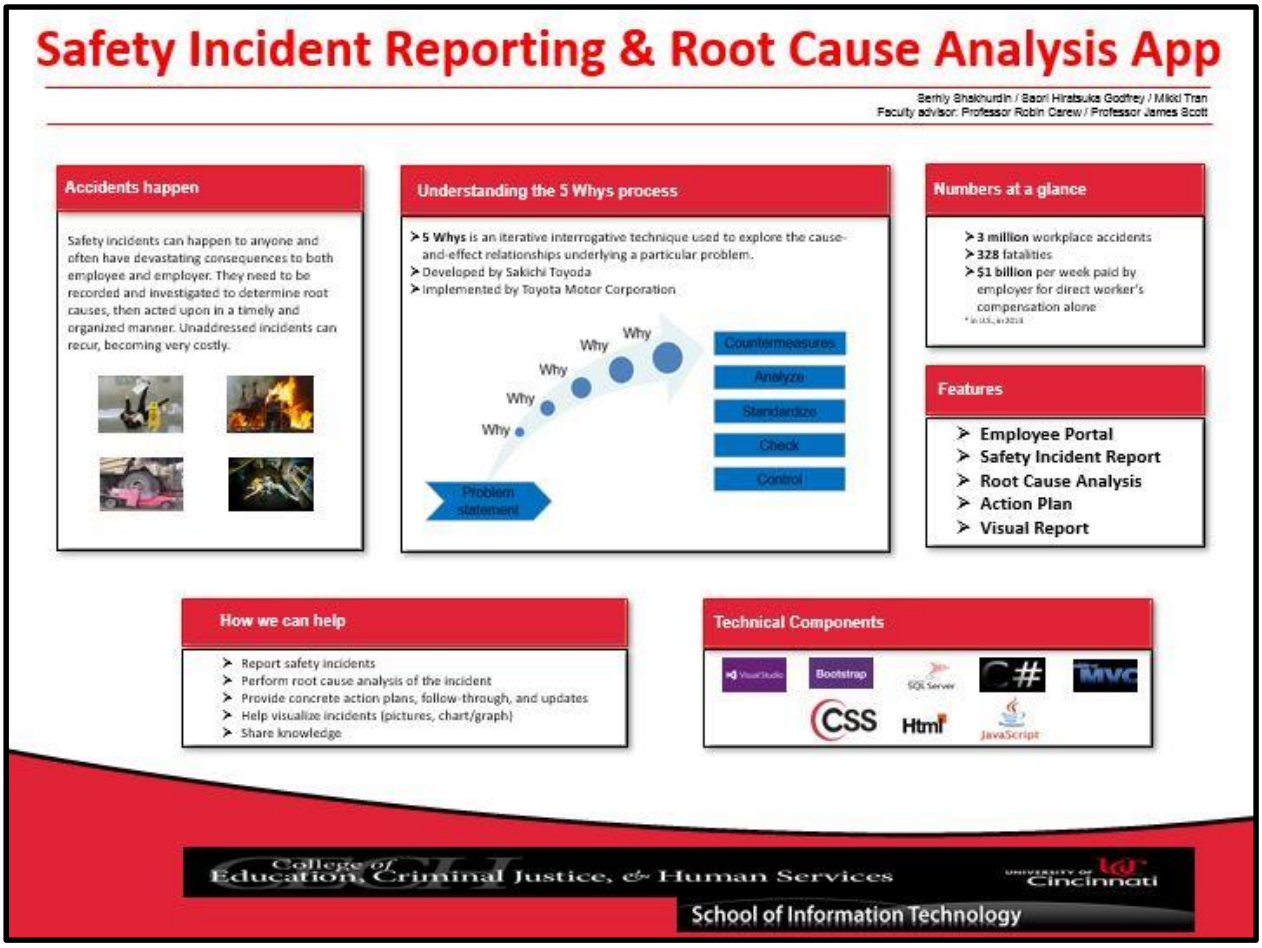
References

1. Bureau of Labor Statistics. Employer Reported Workplace Injuries and Illnesses - 2014. http://www.bls.gov/news.release/archives/osh_10292015.pdf.
2. Adhikari, P. “Errors and Accidents in the Workplaces”. Sigurnost 57, no. 2 (2015): 127-137. http://web.b.ebscohost.com.proxy.libraries.uc.edu/ehost/pdfviewer/pdfviewer?sid=d02c6bfc-8b91-409c-a2d0-674b208e6e0a%40sessionmgr120&vid=3&hid=101.
3. OSHA. "Safety and Health Topics | Business Case for Safety and Health - Costs." Occupational Safety & Health Administration. Accessed November 1, 2015. https://www.osha.gov/dcsp/products/topics/businesscase/costs.html.

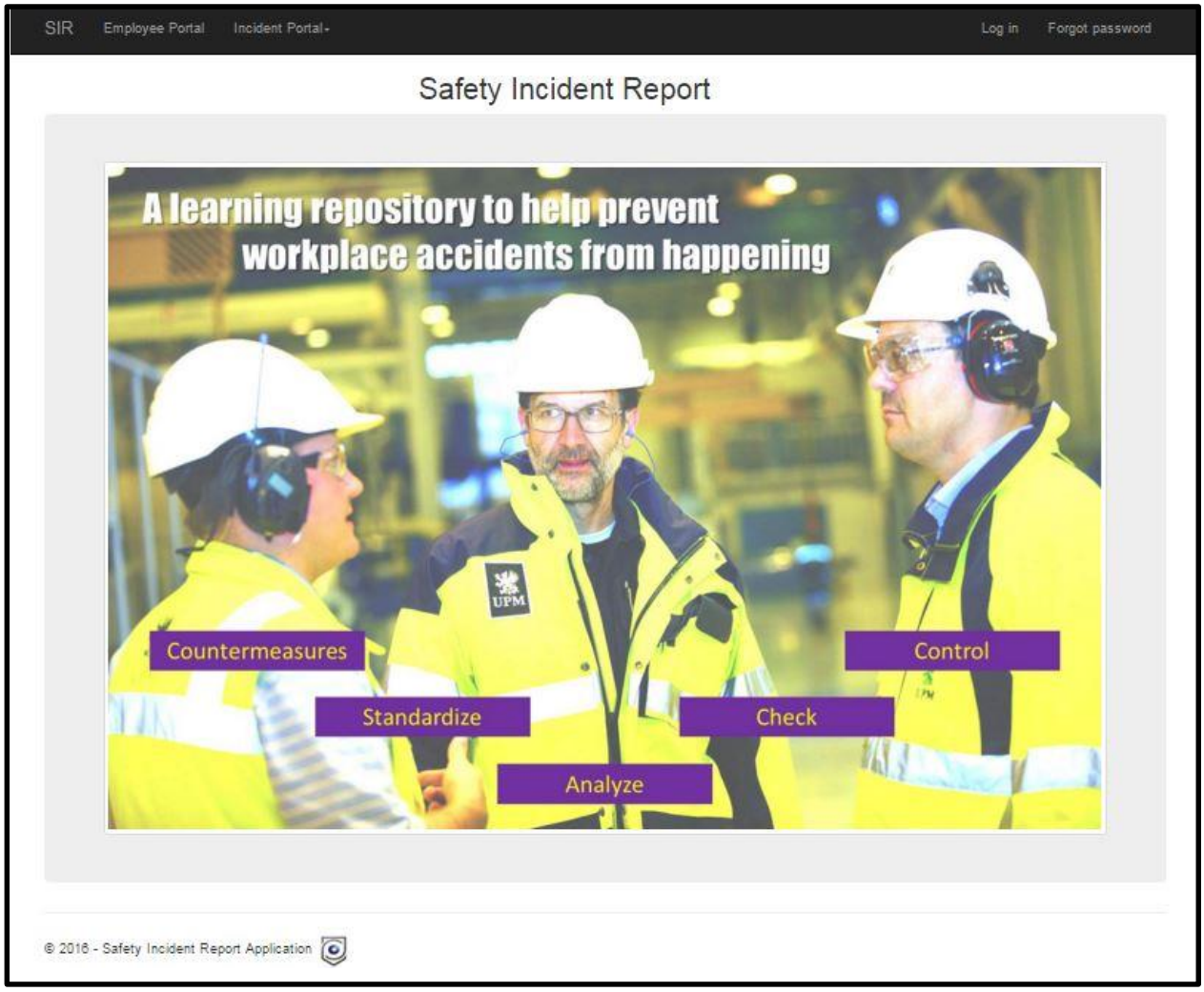
APPENDIX A: Architecture diagram



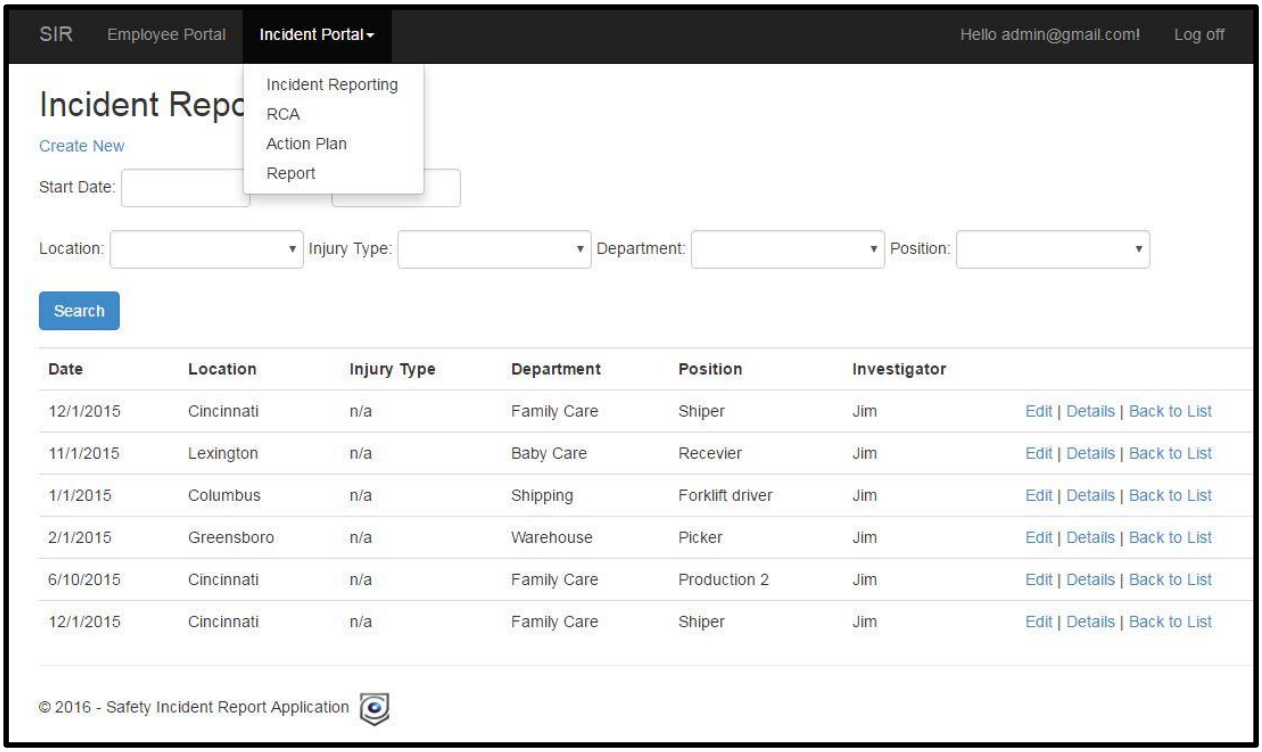
APPENDIX B: Tech Expo poster



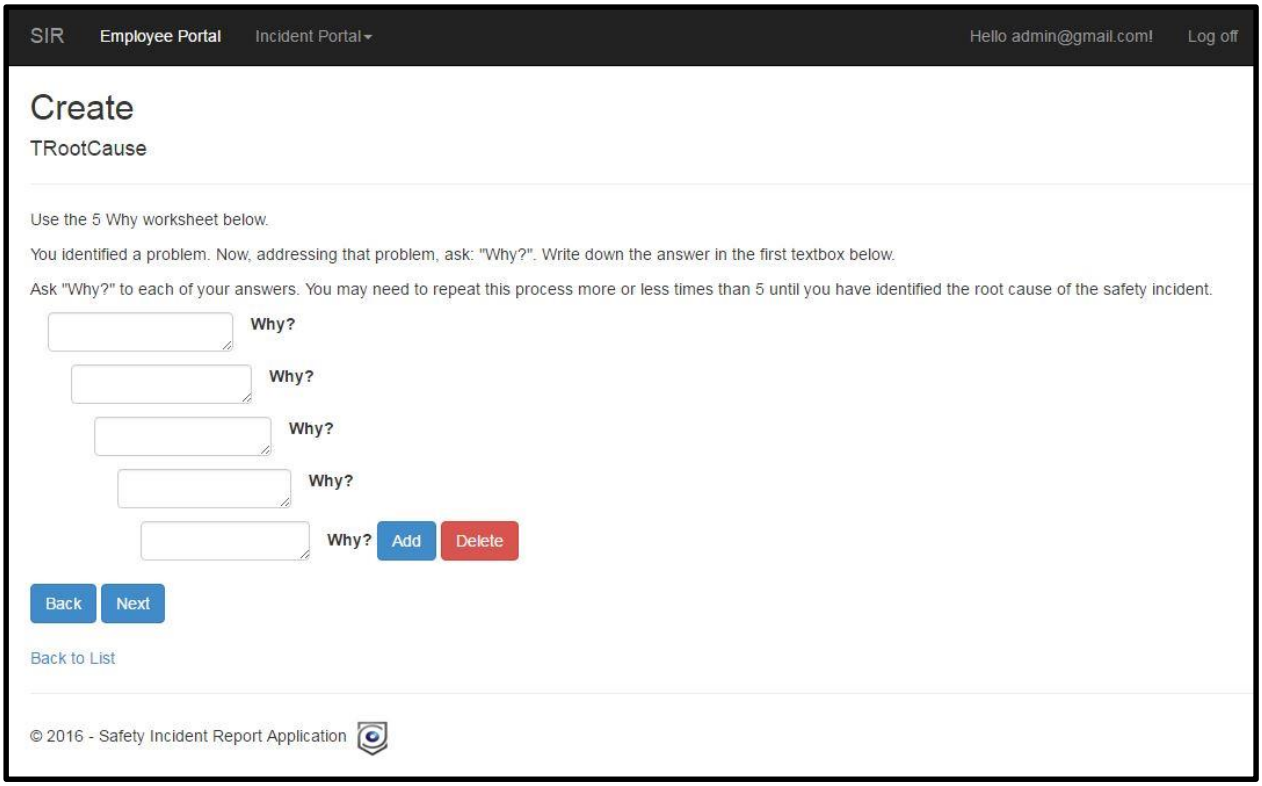
APPENDIX C: Application main screen



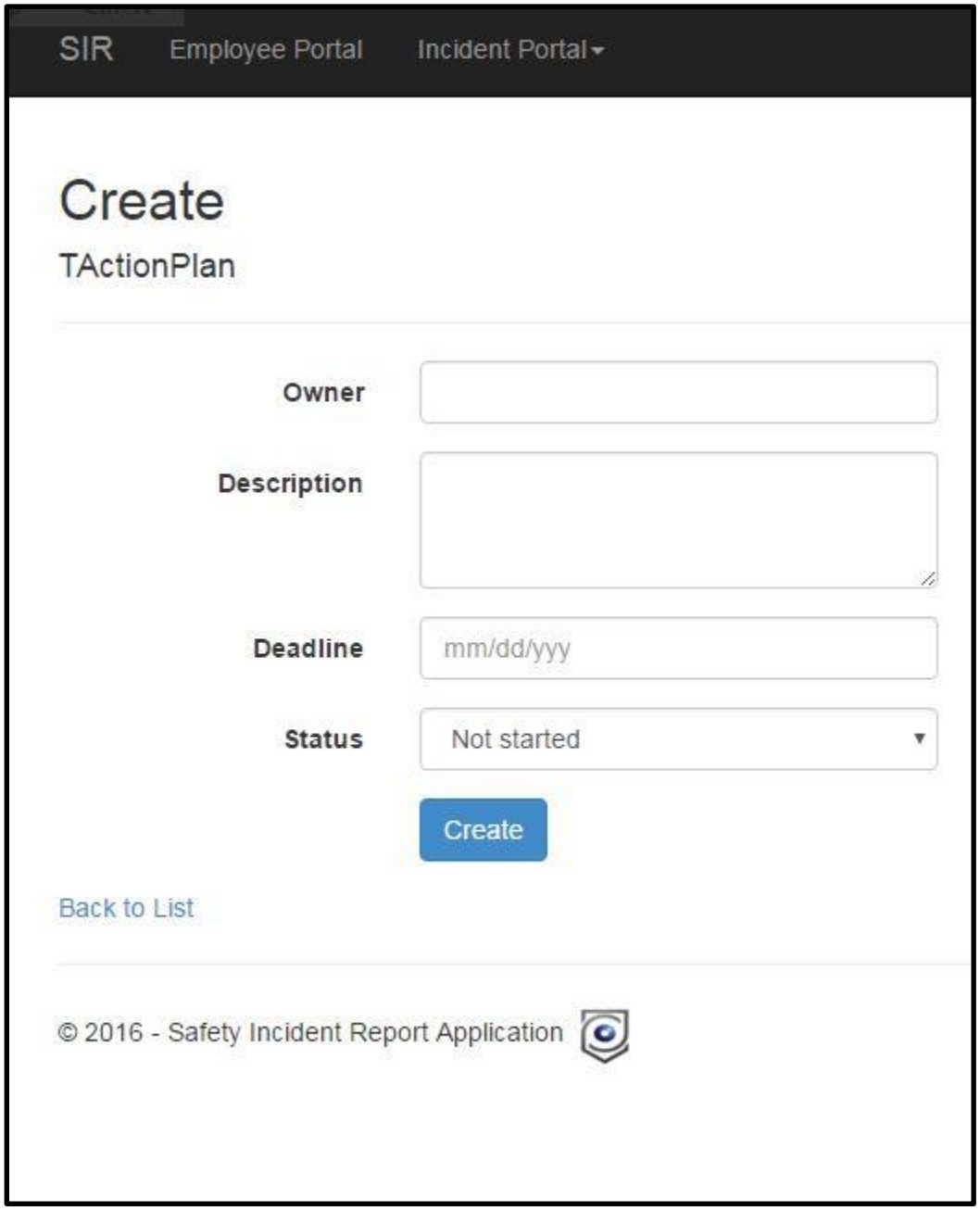
APPENDIX D: Incident report screen



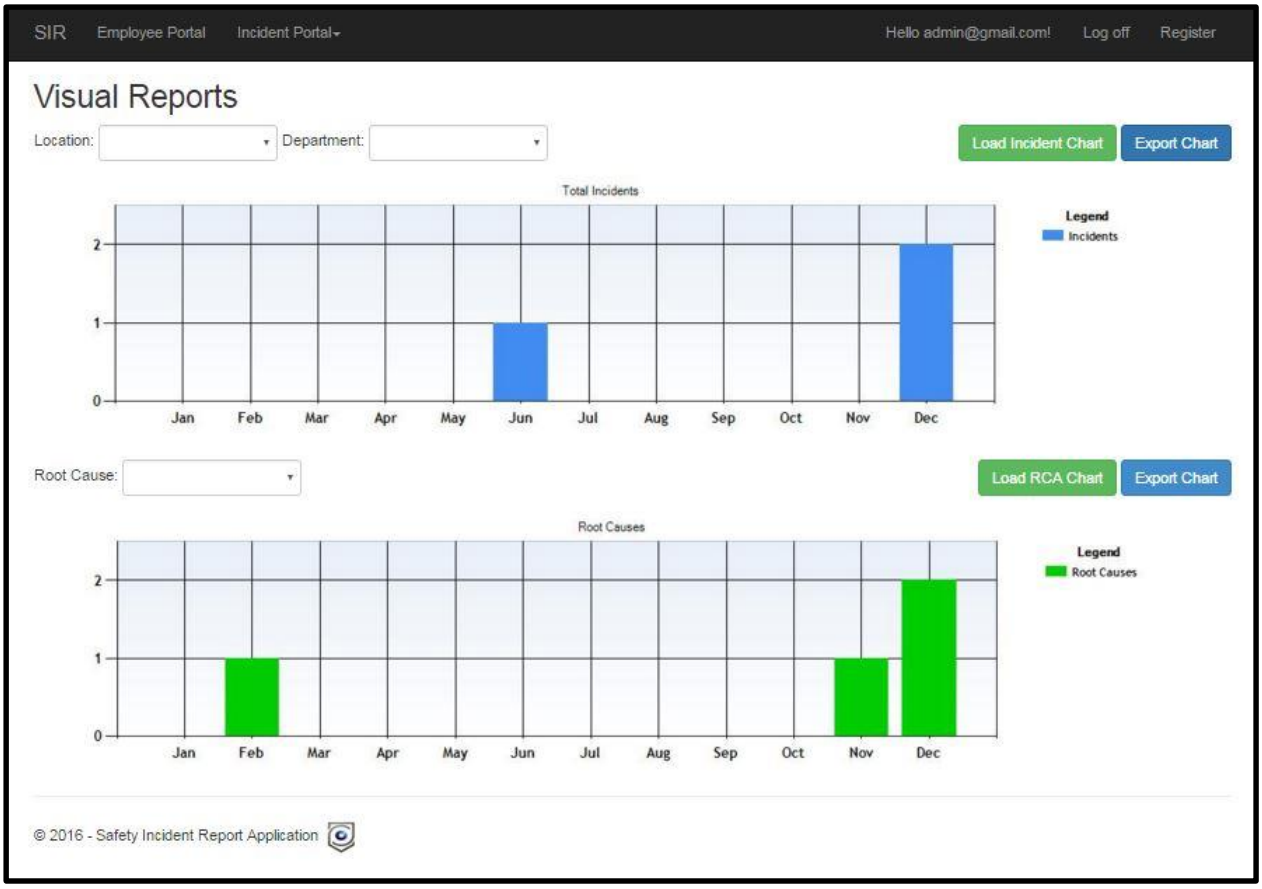
APPENDIX E: Root cause analysis screen



APPENDIX F: Action plan screen



APPENDIX G: Visual report screen



APPENDIX H: Employee portal screen