

MICHIGAN STATE UNIVERSITY  
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING  
ECE 480 SENIOR DESIGN

---

**Project Proposal**  
**Arcelormittal USA**  
**Safety Equipment Bar Code Scanner**

---

*Team Members:*

Kyle INCH  
Alexandria MARONE  
Seth MCKISSON  
Trevor SABO  
Ian GROSH

**Design Team 3**

*Facilitator:*

Dr. Bingsen WANG

October 13, 2016



# Contents

<b>I</b>	<b>Technical</b>	<b>2</b>
<b>1</b>	<b>Project Definition</b>	<b>2</b>
1.1	Scope . . . . .	2
1.2	Function Definition . . . . .	3
1.3	Use example . . . . .	4
<b>2</b>	<b>Technical Design</b>	<b>4</b>
2.1	Mobil Application . . . . .	4
2.1.1	Mobil User Interface & Experience . . . . .	4
2.1.2	Bar code Scanning . . . . .	5
2.1.3	Off-line Mode . . . . .	5
2.2	Web Application . . . . .	5
2.3	Database . . . . .	7
<b>II</b>	<b>Cost</b>	<b>7</b>
<b>III</b>	<b>Project Management</b>	<b>8</b>
<b>3</b>	<b>Scheduling</b>	<b>8</b>
3.1	Preliminary Set up . . . . .	8
3.2	Interconnectivity . . . . .	9
3.3	Preliminary Application Development . . . . .	10
3.4	Secondary Application Development . . . . .	10

## List of Figures

1	System Components . . . . .	2
2	FAST Digram . . . . .	3
3	Mocup of Start Screen . . . . .	5
4	Mocup of splash screen . . . . .	5
5	Database entity relations Diagram . . . . .	7
6	Gant Chart . . . . .	11

## List of Tables

1	Web Framework Solution Selection Matrix . . . . .	7
2	Database Solution Selection Matrix . . . . .	8

## Abstract

Design Team 3 has been asked to create a system to keep track of safety equipment on ArcelorMittal's buildings. To do this the team needs to build some systems. These systems will enable administrators to both monitor compliance standards on areas they are in charge of, and make sure that safety equipment is being properly checked and documented. Reports will be sent out periodically on the above to said administrators. On the user end, an Android application that uses a scanner will be created that will enable users to quickly answer questions on safety equipment standards. This project proposal is broken into the following parts:

1. The teams current understanding of the project.
2. Define the project in such a way that it will be easy to follow for developers that are maintaining the project for years to come.
3. Demonstrate that the team has internalized the design challenge faced.

## Part I Technical

### 1 Project Definition

This team has been assigned an industry sponsored project from ArcelorMittal USA, who need a way to track their industrial safety equipment within their buildings. In order to build the specified system Team 6 will need to build three primary systems. The first being an Android application for operators to scan bar codes on safety locations and equipment. A web application to allow administrators to specify questions for specific pieces of safety equipment, and furthermore assign responsibility to workers. Finally a server infrastructure will be created to ensure data is held properly and securely, host the web application, connect to the Android app, and generate reports.

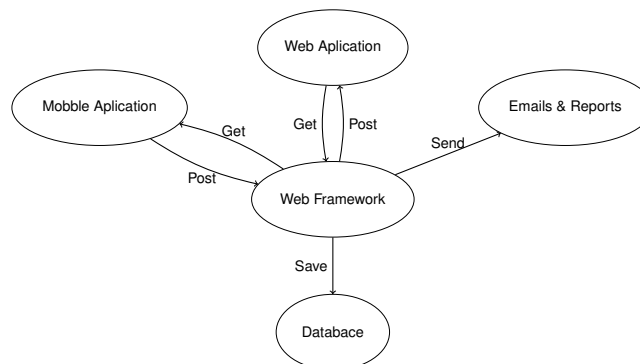


Figure 1: System Components

#### 1.1 Scope

In order to guide the teams design project and ensure the project has limits of what the team will build, a project scope has been defined:

- **Mobile Application**

- Off-line Mode
- Bar code Scanning
- Bar code based questions

- **Sever side Database and middle-ware**

- Host:
  - Web application
  - Database API for mobile app

- **Emails & Reports**

- Inform Administrators of:
  - Failing devices
  - Delinquent Inspectors
- Inform Inspectors of:
  - Upcoming Inspections
  - Missed Inspections

- **Web Application**

- Add locations to the database
- Add safety equipment types to the database
- Create questions
- Associate Bar codes with:
  - Locations
  - Safety equipment
- Associate Locations with:
  - Safety Equipment
  - Questions
- Associate safety equipment with
  - Locations
  - Questions
- Add Questions to reports
- Create timetable for reports
- Add recipient to reports

## 1.2 Function Definition

In order to justify the existence of items in Part 1.1 the team created a number of function definitions. These definitions were then consolidated into the FAST diagram in Figure 2 along with a more detailed description. The Primary function for the project is **Ensure Compliance** This is

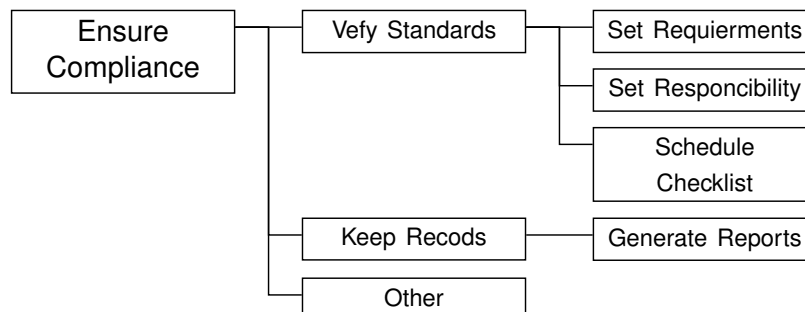


Figure 2: FAST Digram

the main goal of the system we have been commissioned to build. From our primary function are derived two secondary functions each having there own tertiary functions:

- **Verify Standards:**  
In order to Ensure Compliance with all pertinent safety regulations the system must be able to check the and verify that all the stranded are being upheld in the various locations and across the numerous safety devices.
- **Keep Records:**  
In order to ensure that all of the safety laws and regulations are followed we must keep

records of all the locations which must have safety equipment present in a building, what safety equipment must be present, and how to verify that it is in working condition.

- Other

From the secondary function Keep Records we have derived:

- Generate Reports:  
In order to keep records and ensure that every location and item is in compliance the system must be able to generate reports on any set of data on which record are kept.

From the secondary function Verify Standards we have derived:

- Set Requirements:  
In order to Verify Standards the system must be able to set compliance requirements for each location and item which is being tracked by the system.
- Set Responsibility:  
In order to Verify Standards the standards are being upheld a person must be assigned Responsibility for a number of locations and items within the system. Once this Responsibility is set the owner can be held accountable for their set of locations and items.
- Schedule Checklists:  
so that Standards are verified checklists should be generated to show those who are responsible for items which items need to be checked to ensure compliance.

### **1.3 Use example**

Here the team will use the data from the project description provided an example of how the team plans the how the system will be used.

## **2 Technical Design**

### **2.1 Mobil Application**

Design team 3 will build a Mobile application for a tablet device which will allow a user to: download a database of safety equipment and locations to the mobile device to allow the user to operate without access to the intranet, scan bar codes of locations and safety devices to allow the user to answer a series of pass fail inspection questions, push question responses back to the database over the sponsors intranet.

#### **2.1.1 Mobil User Interface & Experience**

Design Team three has commissioned the world renowned user interface and user experience designer, Trevor Sabo to design a front end for both our mobile and web applications.

Design Team three will create a user interface and experience (UI/UX) to guide the safety inspector through the process of an inspection. Figure 3 the user is prompted to enter their payroll number so that the data relevant to that user can be sync to the mobile device and so that the inspection results can be logged under the correct employees name. If the users device is connected to the

intranet will depress the button labeled sync in Figure 4 which will retrieve the inspection locations and safety device information pertinent to the user. Once the inspector reaches a location they will depress the Scan Barcode Location button shown in Figure 4



Figure 3: Mockup of Start Screen

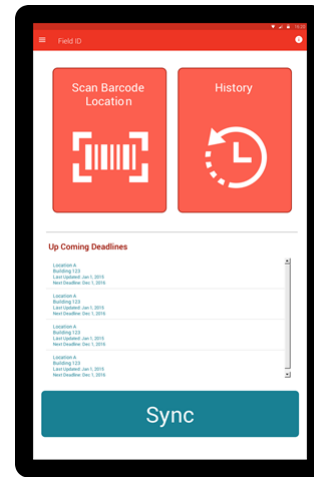


Figure 4: Mockup of splash screen

### 2.1.2 Bar code Scanning

### 2.1.3 Off-line Mode

## 2.2 Web Application

In order to allow the Administrators to add safety equipment and inspection questions to the database, assassin ownership, set email reminder schedules, and generate reports, Design Team 3 must build a simple web application. In order to do so the design team will use a web app framework to build the fetchers required quickly and robustly.

Based on the technical requirements and anticipated need Design Team 3 has devised the following criteria to judge possible choices of web frameworks:

- **Integrates With Sponsor IT**

Design Team 3s sponsor Arcelormittal USA uses a Microsoft windows based infrastructure for its internal IT. This is significant because most of the Internet and so most of the web frameworks are built around UNIX like operating systems such as Linux or free BSD, While many have been built to work in a windows environment we must determine if that exists and if so how well.

- **Documentation**

One of the most important parts of any software library is the documentation, If a developer is required to dig through source code to determine how a library works because there is no documentation which shows how to use it the amount of time which will be required will increase exponentially and will result in code which is more prone to Constantine defects.

- **Ease Of use**

The complexity of a framework can be quite large and part of that can be in the language the framework is written in. This requires Design Team 3 to determine how difficult stranded tasks such as serving a web page, and dynamically creating content. By looking at stranded tasks the overall difficulty of building the full web application will be.

Then we chose from the plethora of existing frameworks seven which are commonly used in web app development to compare against these requirements. The each criteria is marked on a log scale of either 1,3, or 9 and displayed in Table 1.

- **Ruby on Rails**

Ruby on Rails is a web framework named Rails built in the languid Ruby and is commonly referred to as Ruby on Rails.

- **Django**

Django is a python one the most popular web frameworks and is built in python. Django is well sported out of the box with numerous built in interfaces and a healthy ecosystem of free and open third-party add-ons to add interesting features.

- **Play**

Play is a Java Web framework. This framework has a large appeal for us as we will be writing our android application in java, and there is a huge amount of support for the language which would be helpful

- **Learning curve**

- **Stack fullness**

- **Third party Ecosystem**

- **Express**

Express is a Node.js Web application framework.

- **Laravel**

Laravel Is a popular PHP based web framework

- **Revel**

Revel is the go language web framework. As with menny things in go there is one right way to do something and this is it.

- **TurboGears** Next generation python web framework. Taking ques from frameworks like Dajango and Rails this framework uses advanced python and other langiges to build a web app.

	Django	Rails	Play	Laravel	Revel	Flask	TurboGears
Integrates With Sponsor IT	3	3	3	3	1	1	1
Documentation	9	3	3	3	9	3	3
Ease Of use	9	1	3	3	1	9	9
Learning curve	3	1	1	1	1	3	1
Stack fullness	9	9	9	9	9	1	9
Third party Ecosystem	9	9	9	9	3	1	3
<b>Total</b>	<b>42</b>	<b>26</b>	<b>28</b>	<b>28</b>	<b>24</b>	<b>18</b>	<b>26</b>

Table 1: Web Framework Solution Selection Matrix

## 2.3 Database

In order to see how items in the system are related to each other the team has created a entity relation diagram.

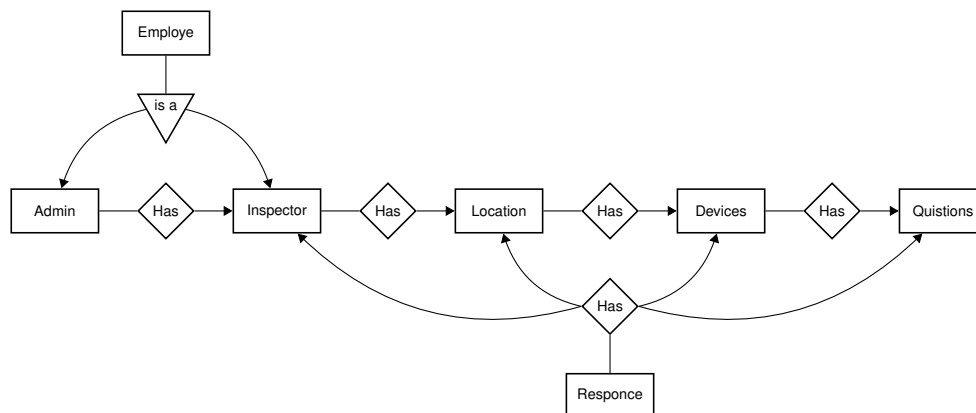


Figure 5: Database entity relations Diagram



	SQLite	Microsoft SQL	Oracle Database	IBM DB2	SAP SQL	MySQL	PostgreSQL	Firebird SQL	mongoDB	Hadoop
Integrates With Sponsor IT	3	9	9	9	3	9	3	3	1	1
Ease of Use	9	1	1	1	1	3	3	3	1	1
Documentation	9	3	3	1	1	3	1	1	3	1
Django Support	9	3	9	3	3	9	9	3	1	1
Fast insertion	1	3	3	3	3	3	3	3	9	9
Large Data sets	1	9	9	9	9	3	3	3	9	9
Enterprise Security	1	9	9	9	9	3	3	3	3	3
Team Experience	9	1	1	1	1	3	1	1	3	1
The correct Price	9	9	1	1	1	9	9	9	9	9
<b>Total</b>	<b>51</b>	<b>47</b>	<b>45</b>	<b>37</b>	<b>31</b>	<b>45</b>	<b>35</b>	<b>29</b>	<b>39</b>	<b>35</b>

Table 2: Database Solution Selection Matrix

## Part II

# Cost

The only cost incurred by the design team will be purchasing a nexus 7 tablet of the same model year or smiler the sponsors devices due to there security measures we are un able to use one of the sponsor owns devises. untill such time as one can be procured, a group member has volunteered there nexus7 tablet to be used.

## Part III

# Project Management

### 3 Scheduling

In order to better adapt to the complex requirements defined in Section 1 , we have broken the tasks into four iterative cycles. Each cycle is planed to address a number of the design requirements which must be develop in or near parallel do to the highly interconnected nature of the individual subsystems.

#### 3.1 Preliminary Set up

In the first cycle the team pursues actions which move to the understanding of the finer details of the system it has been tasked to build. This cycle includes several instances of contact with the sponsor in order to both better understand the customers needs and build a relationship for ongoing communication. At the end of this cycle the team will deliver a project proposal to the faculty advisor and the project sponsor along with having verified that the most basic functionality of the database, web application, Android application and design mock-ups.

- **Meet with Sponsor:** (The Team)

The team prepares questions and meets in person with the Jim Lang from ArcelorMittal. In order to better understand the needs layed out in the project description. In this meeting the project sponsor was asked to describe key features and what a successfully project looked like was discussed.

- **Solidify Understanding of Project:** (The Team)

In this task the Team meets to discuss what was learned in the meeting with the project sponsor previously. In order to layout the framework in which the team can build the requested system.

- **Additional Questions for Sponsor:** (The Team)

After exhaustive discussion on the both the high level work flows and technical feasibility of the project the team will compose a set of questions to be electronically mailed to the project sponsor in order to clear up lingering discontinuity in the teams understanding of the system.

- **Set up Database:** (Alexandria & Ian)

In this task two team members will decide on a framework for building the server side infrastructure for the system. This will include choosing and setting up the server operating system, choosing the main programing language to be used for building the sever infrastructure.

- **Hello World on Tablet:** (Kyle)

In this task a team member will build a simple hello world program for style of android tablet indicated by the project sponsor. In building a hello world program for the tablet the team member will also choose a library for decoding bar codes using the tablet camera.

- **Draw Mock ups:** (Trevor)

In this task a team member will draw the initial designs for the user experiences that will be had in the various user portals of the system so that the project sponsor can have an idea of what the user interface will be like and can give us insights that will help all aspects of the teams design.

- **Set up Web Page:** (Seth)

In this task a team member will setup an initial front end for the administrator web sight. The team member will decide on a set of predefined web objects that can be used to build web applications which will best allows the team to build an effective web application front end.

- **Ask Clarifying questions for sponsor:** (The Team)

After facing the initial feasibility challenges involved in the building the main subsystem of the project the team will compile the design report and a number of clarifying questions for the sponsor in order to further refine the teams initial design decisions.

## 3.2 Interconnectivity

In the Interconnectivity step design team 3 will focus on building and testing connections between the baseline system, setup in 3.1

- **Web speaks to Database:**

- **Mobile speaks to Database:**
- **Demonstrate to Sponsor:**
- **Mock ups and wireframes:**

### **3.3 Preliminary Application Development**

- **Mobile App Development:**
- **Web App Development:**
- **Database Refinements:**
- **Feedback:**
- **New Mock ups:**

### **3.4 Secondary Application Development**

- **Mobile App Development:**
- **Web App Development:**
- **Database Refinements:**
- **Make project shippable to sponsor:**
- **New Mock ups:**

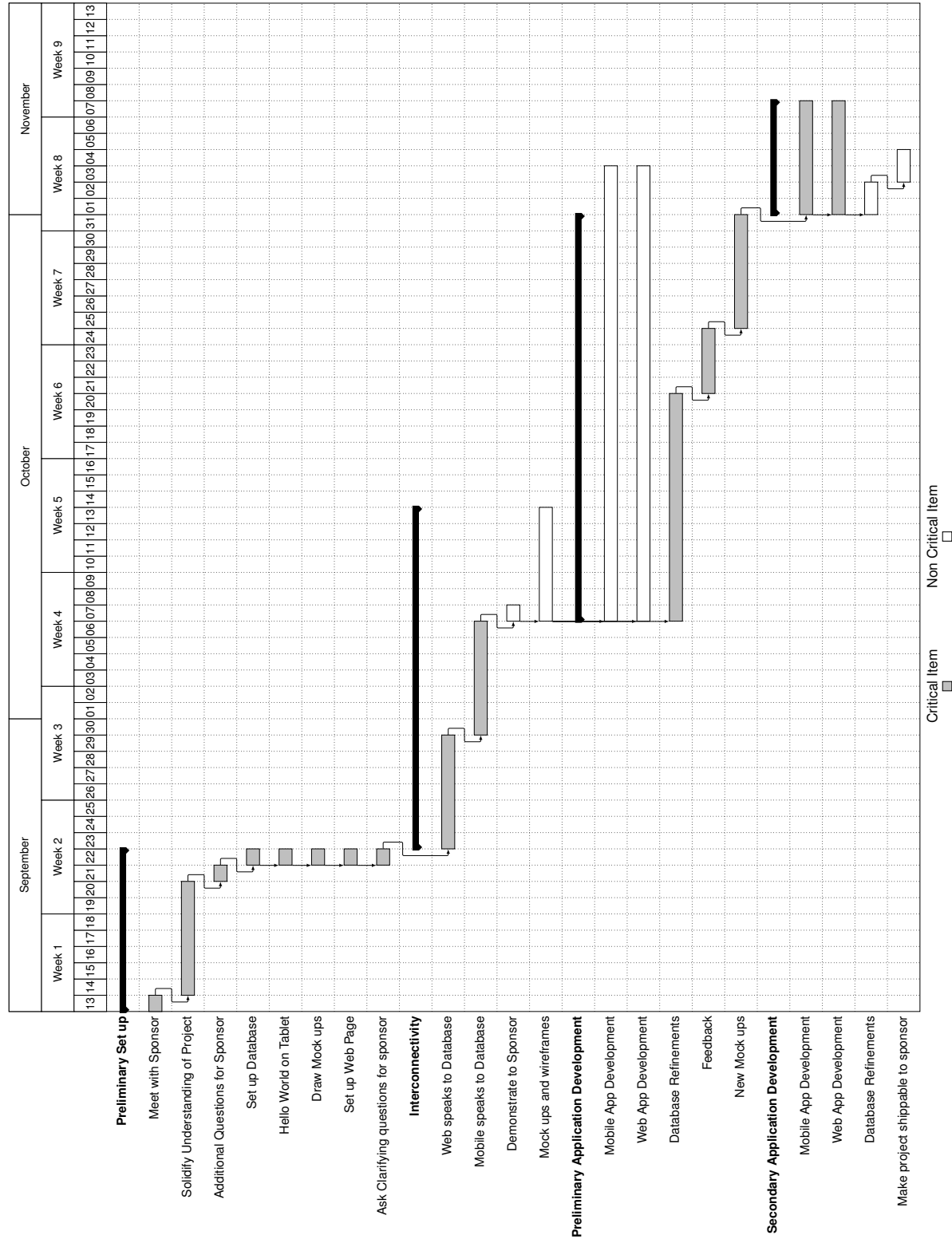


Figure 6: Gant Chart