



Adding the Safety Infraction page to MercuryWeb

Sophie Phillips, Berkeley High School
Summer 2016

What is the Safety
Infraction page?





Machine Shop Member Purchasing Resources Staff

Logout

Filter Options

Clear Filter

Date	Member	Reporter	Advisor	Description
08/04/2016	olek	sophi4	flounders	Click for Details
08/03/2016	olek	sophi4	flounders	Click for Details
08/02/2016	olek	sophi4	flounders	Click for Details
07/29/2016	micro	yjackson	olek	Click for Details
07/27/2016	olek	sophi4	flounders	Click for Details

Report New

Back

List of reported infractions



Safety Infraction Listing Page



Report Form

(Must complete all fields)

Reporter:

Reported Member:

Date of Infraction:



Description:

Save

Cancel





The Process

Bean - Database - Java - XHTML

What is a bean?



- Private properties
- Constructor has no arguments
- Implements serializable
 - (classes can be saved and restored through other services)

● Safety Infraction Bean

- Reporter ID
- Reporter name
- Reported member ID
- Reported member name
- Advisor name
- Infraction description
- Date of infraction
- Infraction ID

```
public SafetyInfractionBean() {  
    id = null;  
    reporterId = null;  
    reportedMemberId = null;  
    description = "";  
    dateReported = null;  
    reporterName = "";  
    reportedMemberName = "";  
    advisorName = "";  
}
```

Safety Infraction Bean Constructor

Adding to Database



- Stores data as tables
- Uses SQL
 - Structured Query Language

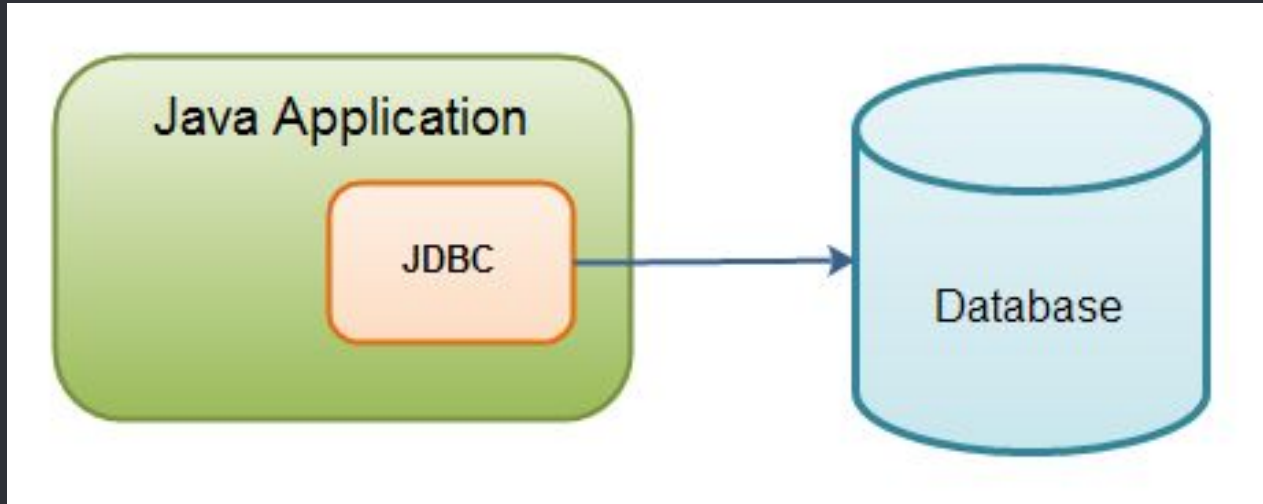
```
create table safety_infraction
(
  id int not null primary key default next value for global_sequence,
  reporter int references member(id),
  reported_member int references member(id);
  date_reported ingresdate default 'now',
  description varchar(4096) not null
);
```

Creating the Safety Infraction table

id	reporter	reported_member	date_reported	description
25816	6733606	1134	2016-07-27 10:15:49	TT
25860	6351268	1129	2016-07-29 15:24:49	test
25904	6733606	1134	2016-08-02 00:00:00	Got in trouble Aug 2nd
25905	6733606	1134	2016-08-03 00:00:00	Got in trouble on the 3rd this time.
25906	6733606	1134	2016-08-04 00:00:00	What a trouble maker !! 3 days in a row !!

Safety Infraction table

Java Database Connectivity (JDBC)



- SELECT - FROM - WHERE

- SELECT [property]

FROM [table]

WHERE [qualifier]

```
SELECT safety_infraction.id,  
       safety_infraction.reporter,  
       safety_infraction.reported_member,  
       safety_infraction.date_reported,  
       safety_infraction.description,  
       members_1.name AS reporterName,  
       members_2.name AS reportedMemberName,  
       advisors.name AS advisorName  
FROM   safety_infraction,  
       members AS members_1,  
       members AS members_2,  
       advisors,  
       pr_members  
WHERE  safety_infraction.reported_member = members_1.id  
AND    safety_infraction.reporter = members_2.id  
AND    safety_infraction.reported_member = pr_members.member  
AND    pr_members.deleted = 'n'  
AND    pr_members.advisor = advisors.id
```

Command

Result 1

Messages

id	reporter	reported_member	date_reported	description	reportername	reportedmembername	advisorname
25860	6351268	1129	2016-07-29 15:24:49	test	micro	yjackson	olek
25906	6733606	1134	2016-08-04 00:00:00	What a trouble maker !! 3 days in a row !!	olek	sophi4	flounders
25816	6733606	1134	2016-07-27 10:15:49	TT	olek	sophi4	flounders
25905	6733606	1134	2016-08-03 00:00:00	Got in trouble on the 3rd this time.	olek	sophi4	flounders
25904	6733606	1134	2016-08-02 00:00:00	Got in trouble Aug 2nd	olek	sophi4	flounders

Output

Safety Infraction View

Components:

- Filter
- Table
- New / Back buttons

The screenshot displays a web application interface for reporting safety infractions. At the top, there is a navigation bar with the Marvell Nanofabrication Laboratory logo and name, and a version number '4.0 build(20160809)'. Below the navigation bar, there are tabs for 'Machine Shop', 'Member', 'Purchasing', 'Resources', and 'Staff', along with a 'Logout' button. The main content area is titled 'Filter Options' and contains a table of reported infractions. The table has columns for 'Date', 'Member', 'Reporter', 'Advisor', and 'Description'. Each row in the table includes a 'Click for Details' link. Below the table, there are 'Report New' and 'Back' buttons. At the bottom of the page, there is a footer with the text 'List of reported infractions' and a small circular icon.

Marvell Nanofabrication Laboratory
University of California, Berkeley

4.0 build(20160809)

Machine Shop Member Purchasing Resources Staff Logout

Filter Options [Clear Filter](#)

Date	Member	Reporter	Advisor	Description
08/04/2016	olek	sophi4	flounders	Click for Details
08/03/2016	olek	sophi4	flounders	Click for Details
08/02/2016	olek	sophi4	flounders	Click for Details
07/29/2016	micro	yjackson	olek	Click for Details
07/27/2016	olek	sophi4	flounders	Click for Details

Report New Back

List of reported infractions

● Adding a filter

```
SELECT safety_infraction.id,  
       safety_infraction.reporter,  
       safety_infraction.reported_member,  
       safety_infraction.date_reported,  
       safety_infraction.description,  
       members_1.name AS reporterName,  
       members_2.name AS reportedMemberName,  
       advisors.name AS advisorName  
FROM   safety_infraction,  
       members AS members_1,  
       members AS members_2,  
       advisors,  
       pr_members  
WHERE  safety_infraction.reported_member = members_1.id  
AND    safety_infraction.reporter = members_2.id  
AND    safety_infraction.reported_member = pr_members.member  
AND    pr_members.deleted = 'n'  
AND    pr_members.advisor = advisors.id
```

+= AND safety_infraction.reported_member = this.memberID

● Retrieving data for table

```
stmt = getSessionBean().getDbConn().prepareStatement(sql);
rs = stmt.executeQuery();
this.incidentList.clear();

while (rs.next()) {
    SafetyInfractionBean si = new SafetyInfractionBean();

    si.setId(rs.getInt(1));
    si.setReporterId(rs.getInt(2));
    si.setReportedMemberId(rs.getInt(3));
    si.setDateReported(rs.getTimestamp(4));
    si.setDescription(rs.getString(5));
    si.setReporterName(rs.getString(6));
    si.setReportedMemberName(rs.getString(7));
    si.setAdvisorName(rs.getString(8));

    this.incidentList.add(si);
}
```

Buttons



Report New

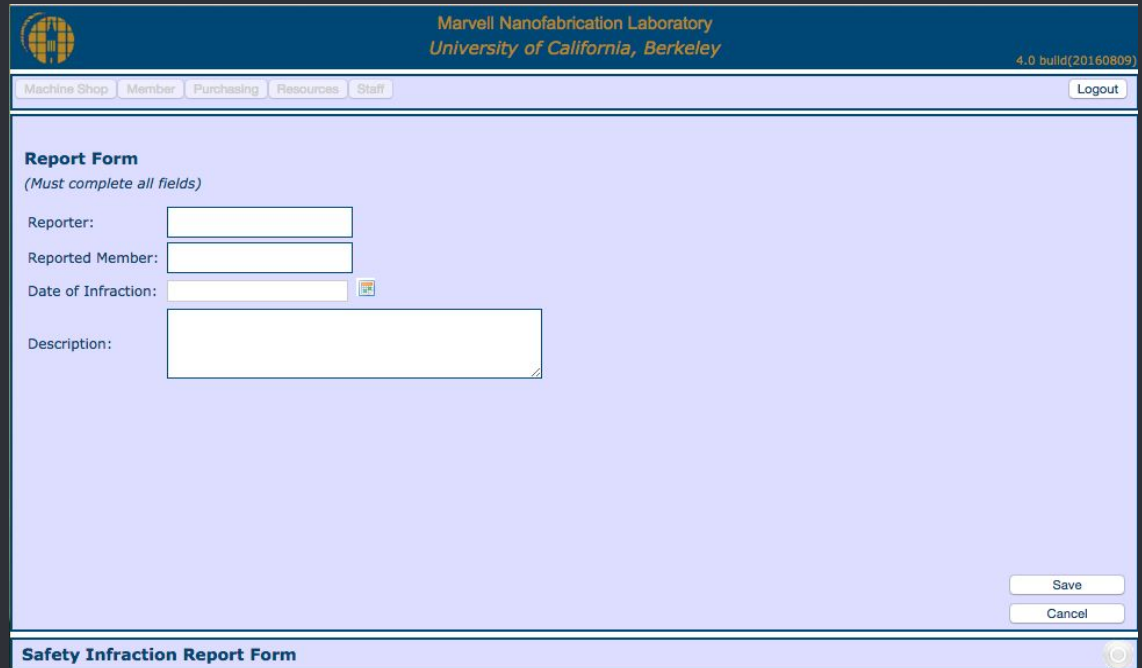
```
return "SafetyInfractionEdit";
```

```
<from-outcome>SafetyInfractionEdit</from-outcome>  
<to-view-id>/pages/staff/SafetyInfractionEdit.xhtml</to-view-id>
```

Safety Infraction Edit

Components:

- Form
- Submit button



The screenshot displays a web application interface for reporting safety infractions. At the top, there is a blue header bar with the Marvell Nanofabrication Laboratory logo and name, and the version number '4.0 build(20160809)'. Below the header is a navigation menu with links for 'Machine Shop', 'Member', 'Purchasing', 'Resources', and 'Staff', and a 'Logout' button. The main content area is titled 'Report Form' and includes a note '(Must complete all fields)'. The form contains four input fields: 'Reporter:', 'Reported Member:', 'Date of Infraction:', and 'Description:'. The 'Date of Infraction:' field has a calendar icon. At the bottom right of the form area, there are 'Save' and 'Cancel' buttons. A footer bar at the bottom of the page contains the text 'Safety Infraction Report Form' and a circular icon.

Marvell Nanofabrication Laboratory
University of California, Berkeley


4.0 build(20160809)

Machine Shop | Member | Purchasing | Resources | Staff | Logout

Report Form
(Must complete all fields)


Reporter:

Reported Member:

Date of Infraction: 

Description:

Save
Cancel

Safety Infraction Report Form 


● Input Form

- Reporter name
- Reported member name
- Description
- Date reported

Report Form
(Must complete all fields)

Reporter:

Reported Member:

Date of Infraction: 

Description:

Submit Button

```
SafetyInfractionBean sib = new SafetyInfractionBean();
sib.setReporterName(this.reporterName);
sib.setReporterId(this.staff_map.get(this.reporterName));
sib.setReportedMemberId(this.members_map.get(this.reportedMemberName));
sib.setReportedMemberName(this.reportedMemberName);
sib.setDateReported(this.dateReported);
sib.setDescription(this.description);
```

INSERT INTO [table]

VALUES [values]

- Submit Button Code

```
INSERT INTO safety_infraction(description, reporter,...)
VALUES( ?, ?, ?, ?);
```

```
stmt.setObject(1, this.getDescription());
```

```
stmt.setObject(2, this.getReporterId());
```

```
....
```

Email

- Retrieves template from database
- Replaces keywords for specific incident

```
private String replaceKeywords(String template) {  
    template = template.replace("$REPORTER$", this.getReporterName());  
    template = template.replace("$REPORTEDMEMBER$", this.getReportedMemberName());  
    template = template.replace("$BODY$", this.getDescription());  
    template = template.replace("$BODY2$", this.getDescription());  
    template = template.replace("$ADVISORS$", this.getAdvisorName());  
    return template;  
}
```

- Sends with Sendmail class

XHTML Files

- Like HTML but stricter
- Everything in <tags></tags>

```
<ace:column id="date" styleClass="table-font-bold"
            style="width: 15%;" headerText="Date" sortBy="#{list.dateReported}">
  <h:outputText id="date2" styleClass="table-font" value="#{list.dateReported}">
    <f:convertDateTime pattern="MM/dd/yyyy" />
  </h:outputText>
</ace:column>
```

Example Table Code

Date
08/04/2016
08/03/2016

```
<ace:dataTable
value="#{pages$staff$safetyInfractionView.incidentList}"
var="list" ...>
  <ace:column id="date" headerText="Date" ...>
    <h:outputText value="#{list.dateReported}" ...>
      <f:convertDateTime pattern="MM/dd/yyyy"/>
    </h:outputText>
  </ace:column>
</ace:dataTable>
```

● Example Form Code

Description:

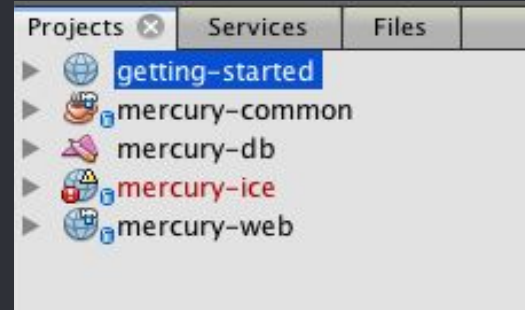
```
<h:outputLabel for="descr" value="Description: " />
```

```
<ace:textAreaEntry id="descr"  
value="#{pages$staff$safetyInfractionEdit.description}"... >
```

```
<ace:ajax event="blur" execute="@this" />
```

```
</ace:textAreaEntry >
```

Adding to MercuryWeb



- Commit to SVN
 - Subversion
 - Maintains versions of source code

● Upload to Server





<https://mercuryweb.berkeley.edu/MercuryWeb/Login.jsp>

Acknowledgements

Thanks to Olek and Yuliya for helping me with my project and to the Nanolab as a whole for this incredible opportunity!