CAEN Customer Service Request (CSR) Database

Final Project Report

SI654: Database Application Design

April 2004

Dino Anastasia, Marla Gómez

Project Overview

The CAEN Service Center (CSC) within the College of Engineering (CoE) is an on-campus computer repair facility that offers convenient computer and laser printer repair for all University departments, students, and employees. They are a fully authorized warranty repair facility for Macintosh, Dell, and Gateway computers. They also perform non-warranty repairs and upgrades for these and other brands at competitive rates. Located in 1404 Media Union, with an entry through the loading dock, a reserved parking spot provides customers ease in dropping off or picking up equipment. They also offer a pickup and delivery service on campus for an additional charge.

When a customer drops off a computer to be repaired, the CSC staff use a Windows application, Omniform (a simple flat file database application), to fill out an electronic form - the CAEN Service Repair Request (CSRR). A new record is created in Omniform to track the repair. The form is then printed and a copy is given to the customer as a receipt and another copy is attached to the computer to be repaired. As the diagnosis and repair proceed, technicians write notes regarding the repair, parts ordered, parts used, and status on the paper copy with the computer. When the repair is completed, relevant portions of the notes along with parts used and associated costs are transferred from the paper copy back into the Omniform record.

The customer is given an updated copy of the completed CSRR as a receipt for the work performed. Another paper copy of the CSRR is routed to the CAEN financial staff to bill out the repair as necessary. On its way to the financial staff, other administrative staff summarize and record general activity information from the form in order to produce activity data such as how many of a specific type of computer were repaired during a month, how much warranty work was performed, etc.

Our goal was to transform the CSRR from a stand-alone application to a web-enabled database with a richer feature set that could be supported by the CoE, and not have to be maintained by CAEN staff. The application will also provide multiple views of the repair activity to better support the technician's workflow and need to annotate the repair. In addition, we hope to streamline the CSRR routing and provide additional functionality for management to generate reports on service activity.

Database Structure

The database consists of twelve tables. The main tables are the repair, contact and equipment tables. The remaining tables all point to the repair or repair_activity tables through foreign keys. Appendix B shows the complete entity-relationship diagram (in Visio).

DATABASE TABLE	DESCRIPTION
contact	Contains all of the customer's information.
equipment	Contains information about the piece of equipment that has been submitted for repair.
repair	Contains detailed information about the repair problem on a particular piece of equipment.
repair_activity	Serves as the main linking table for other tables by containing the foreign keys of the other tables.
repair_equipment	Contains information about the type of equipment received for repair (i.e. desktop, printer, laptop).
repair_notes	Contains the shop notes for the repair.
repair_other	Contains additional information related to the repair that may or may not exist for each repair.
repair_staff	Contains password/uniqname information about the CAEN Service Center staff.
repair_status	Contains information about the current status of the repair.
repair_techs	Contains information about which technician is associated with a repair.
repair_type	Contains information about the type of repair, used for tracking purposes (i.e. personal, CAEN computer, student, department).
repair_vendor	Contains information about vendors (i.e. Dell, HP, Apple).

Functionality

The database and web application allow technicians to complete the following functions:

- Enter a new customer service request form
- Modify an existing customer service request form

The database is password-protected and requires a login. There is also session control that automatically logs out the staff person if the session is idle for too long.

The application and database also allow the manager of the service center to generate counts in reports by vendor name, repair type and device type, as well as manage the table data.

Customers/Audience

The main customers for the application are the CAEN Service Center staff and manager (6 permanent staff and approximately 12 student staff). The staff receives the equipment, completes the repair, and documents the repair process. The manager tracks the repair activity completed by the service center.

Database Design Process

Our basic approach for the database project was to take an existing basic prototype (developed in PHP/mySQL) and optimize the database design. Our goal was to develop it with Oracle and JSP. We began the project by doing some planning, design, and review of the existing prototype. Our next step was to meet with one of the Oracle DBAs for CAEN to discuss the current Oracle structure we would have to design around. After several meetings, it became apparent that it would be a pretty complicated process to get the project done this way by the end of the semester and required a lot more Oracle skills than we currently had. We determined that the best course of action was to optimize the database design and web application as a stand-alone application so that it was at least ready for use.

We developed nine new tables and restructured one of the existing tables. Our table design provides future flexibility for growth and changes to the database tables. For example, the repair_activity table was created to allow for an easy way to add a new table representing activity to be tracked that is related to the repair_activity table through the addition of a foreign key from the newly created table.

We reworked the code to integrate it with the new tables, improved the basic layout and functionality, and allowed for a more object-oriented approach where pieces of code can be re-used. We accomplished this in PHP with the use of ‘include’ statements.

We learned several important lessons through the design process. These include:

1) Designing your tables is the most important step and must be done early in the project.
2) Building a database and web application from scratch is often easier than revising an existing database and application – which is why initial design is so important and was stressed throughout the course!
3) Being able to design a database well for a client requires a lot of understanding about the business process and needs behind the applications.

Future of the Database

The database is currently functional as a stand-alone web application with a mySQL back-end and can begin to be used. However, the ideal situation would be for the database to be completely wrapped into the CoE/CAEN's existing structure so that it could be a college-supported production application and not have to be maintained by CAEN.

Dino Anastasia has decided to do a DFE this summer to further develop the database using JSP and Oracle, as well as CoE's Zope content management system. It is anticipated that the following tasks will need to be accomplished in order to achieve the goals stated above:

1) Gather feedback (both direct and observational) from CAEN Service Center staff on feature set and usability requirements for the repair form using prototype as a model.
2) Develop information architecture for needed forms, web pages, and database tables to support the form.
3) Create the database tables in Oracle.
4) Code web pages using JSP to interact with database tables.
5) Conduct usability testing of the completed pages with CAEN Service Center staff and develop punch list of improvements and fixes needed.
6) Update code to incorporate feedback noted in #5 above.
7) Document and hand off code base for production implementation by CoE web team.

Summary

The project was a learning experience for both of us and allowed us to improve upon our SQL, PHP, and HTML skills. We developed a usable application for the CAEN Service Center to track its computer repairs and are proud of the results of our efforts. We look forward to seeing the database being put into actual use later this summer.

Appendices

Appendix A: Current Oracle Structure of CAEN Tables (includes only tables relevant to this project)

Appendix B: CAEN Customer Service Request (CSR) Database Table Structure

Appendix C: Descriptive List of Source Code

Appendix D: Welcome Page Screenshot and Code

Appendix E: Customer Receipt Screenshot and Code

Appendix F: Repair Log Screenshot and Code

Appendix G: Repair Activity Reports Screenshot and Code (3 reports: by vendor name, type of device, and type of repair)

Appendix H: Create File (do at end)

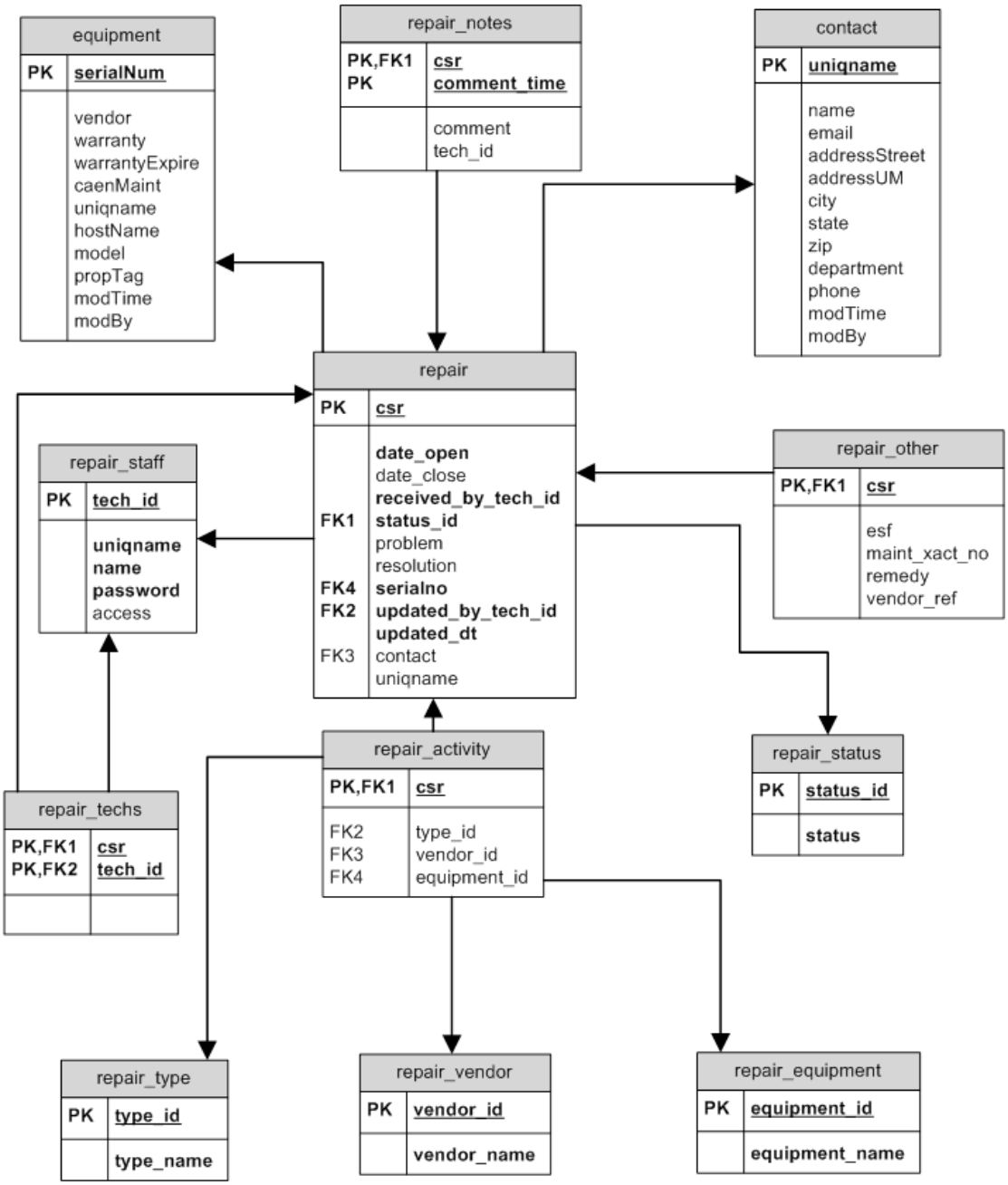
Appendix I: SQL Table Descriptions

Appendix J: Other Source Code

Appendix A: Current Oracle Structure of CAEN Tables (includes only tables relevant to this project)

Please see attached E-R diagram.

Appendix B: CAEN Service Center Repair Table Structure



Last Updated; 2004-04-10 -- marlag

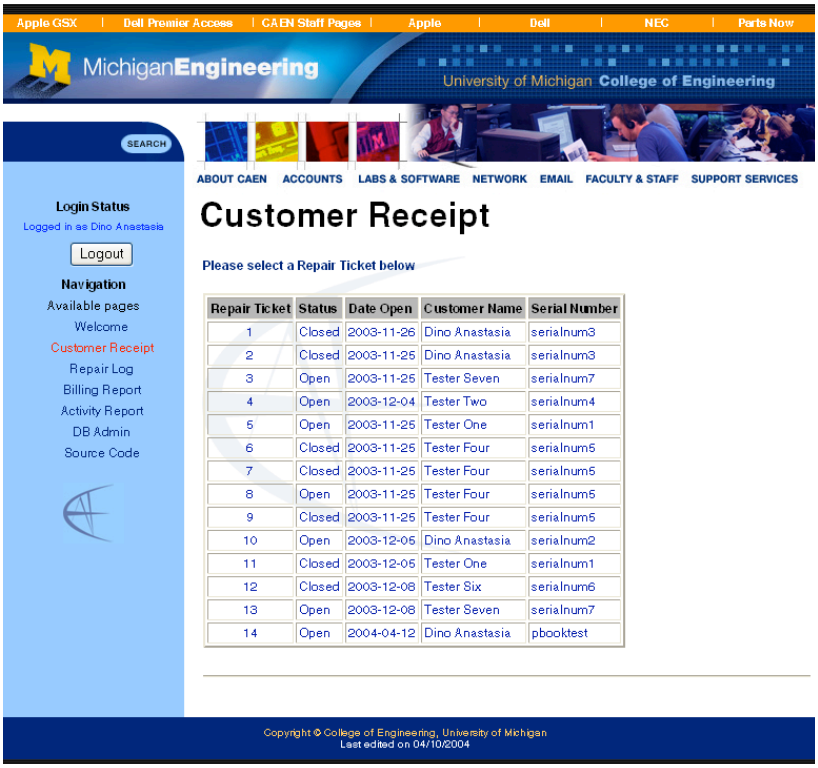
Appendix C: Descriptive List of Source Code

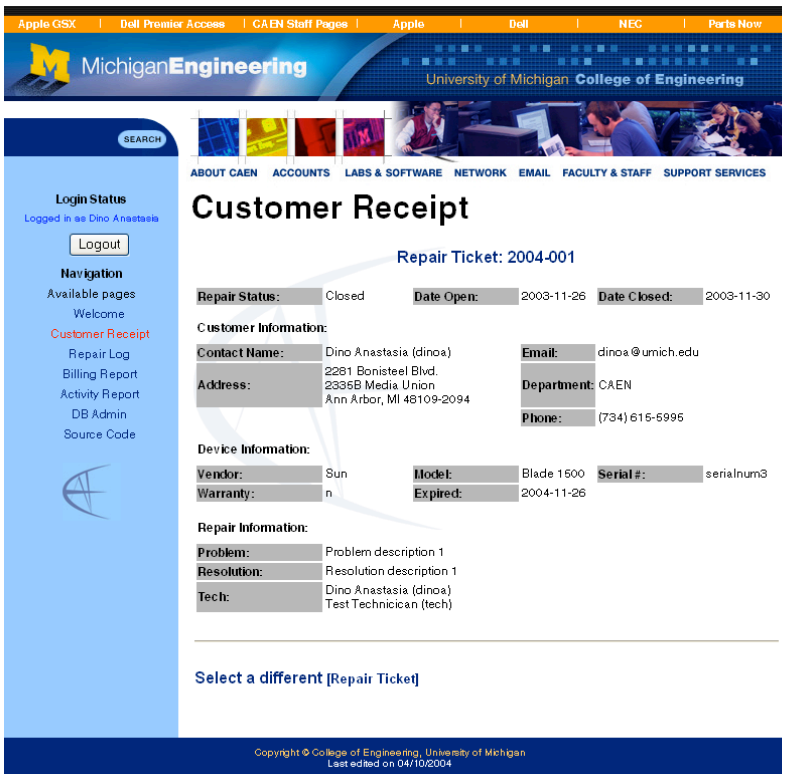
1. **auth.inc** Defines the authentication object class that is used to verify database users and their access rights.
2. **csc-activity.php** : CAEN Service Center Customer Service Request form
   This is the report page for the CSR. It will provide counts of the type of repair activity happening.
3. **csc-contact-add.php** : This code presents the ADD forms for adding records to the contact table.
4. **csc-contact-edit.php** : This code presents the EDIT forms for editing a records from the equipment table.
5. **csc-contact-verify.php** : This code verifies the data being edited from or added to the contact table.
6. **csc-custrcpt.php**: Customer Receipt
   This page generates a repair receipt that can be given to the customer. It is a limited view of the data shown on the repair ticket.
7. **csc-dbadmin.php** : This is the main page for the CSR – for dba?
8. **csc-dropdown-act-equipment.php** : This code queries the repair_equipment table to get the values needed to build the drop down menu.
9. **csc-dropdown-act-type.php** : This code queries the repair_type table to get the values needed to build the drop down menu.
10. **csc-dropdown-staff.php** through **csc-dropdown-staff2.php** : This code queries the repair_staff table to get the values needed to build the drop down menu.
11. **csc-dropdown-status.php** : This code queries the repair_status table to get the values needed to build the drop down menu.
12. **cs-dropdown-vendor.php** : This code queries the repair_vendor table to get the values needed to build the drop down menu.
13. **csc-equipment-add.php** : This code presents the ADD forms for adding a record from the equipment table.
14. **csc-equipment-edit.php** : This code presents the EDIT forms for editing a records from the equipment table.
15. **csc-equipment-verify.php** : This code verifies the data being edited from or added to the equipment table.
16. **csc-login.php** : HTML for building the user login form.
17. **csc-logout.php** : HTML for building the user logout button.
18. **csc-nav-bottom.php** : This code provides some navigation functionality at the bottom of various pages.
19. **csc-query-contact-exist.php** : This code queries the contact table to verify if a contact already exists in the table.
20. **csc-query-equipment-exist.php** : This code queries the equipment table to verify if a piece of equipment already exists in the table.
21. **csc-query-rcpt.php** : This code performs the queries needed to display information for the customer receipt.
22. **csc-query-repair.php** : This code performs the queries needed to display information for the complete repair log.

23. **csc-query-vendor-name.php** : This code performs the queries needed to associate the vendor_name with the vendor_id.
24. **csc-repair-addedit.php** : This code presents the ADD forms for adding records to the repair table and also the EDIT forms for editing records.
25. **csc-repair-verify.php** : This code verifies the complete repair record being accessed by the technician.
26. **csc-repair.php** : This page provides the technicians with the repair receipt entry and edit form allowing them to update the shop notes, etc.
27. **csc-tbl-contact1.php** : This code displays the full content of the contact table.
28. **csc-tbl-contact2.php** : This code displays an abbreviated version of the contents for the contact table.
29. **csc-tbl-contact3.php** : This code displays an abbreviated version of the contents for the contact table used for DELETE verification.
30. **csc-tbl-equipment1.php :** This code displays the full contents of the equipment table.
31. **csc-tbl-equipment2.php** : This code displays an abbreviated version of the contents for the equipment table.
32. **csc-tbl-equipment3.php :** This code displays an abbreviated version of the contents for the equipment table used for DELETE verification.
33. **csc-query-rcpt.php** : This code generates a selection table for the customer receipt.
34. **csc-query-repair.php** : This code queries for a complete repair record.
35. **csc-tbl-repair1.php** through **csc-tbl-repair11.php** : This code displays various views of the repair log for the repair.
36. **csc-verify-activity.php** : This code verifies that activity information was provided for the repair record being entered.
37. **csc-verify-contact.php** : This code verifies the existence of contact information for the repair record being entered.
38. **csc-verify-equipment.php** : This code verifies the existence of equipment information for the repair record being entered.
39. **csc-verify-tech.php** : This code verifies the existence of technician information for the repair record being entered.
40. **csc-welcome.php** : This is the main welcome and login page for the application.
41. **dbconnect.php** : Connects to database server and selects the database to use. Provides a fail-over connection to the local host if primary database host is unavailable.
42. **redirects.php** : Defines assorted javascript functions that are used for the final project.

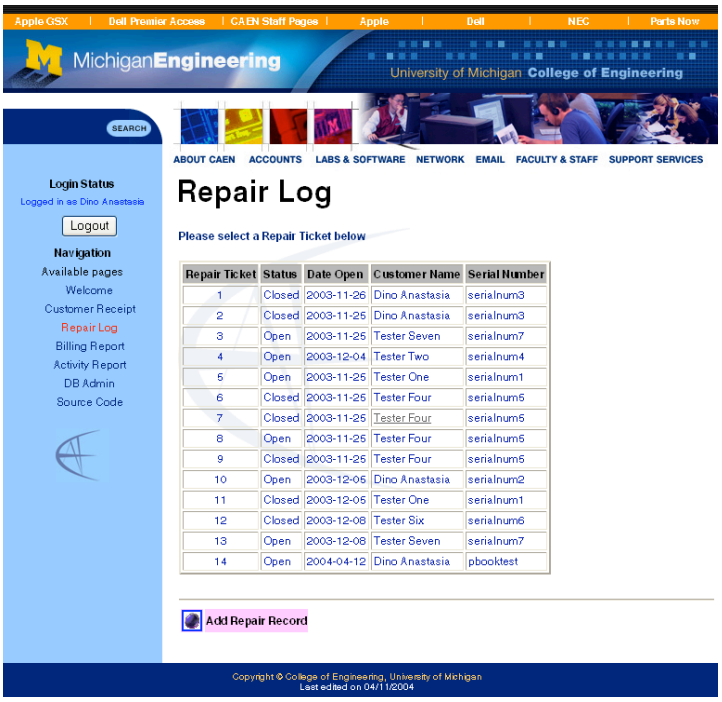
Appendix D: Welcome Page- Screenshot and Code

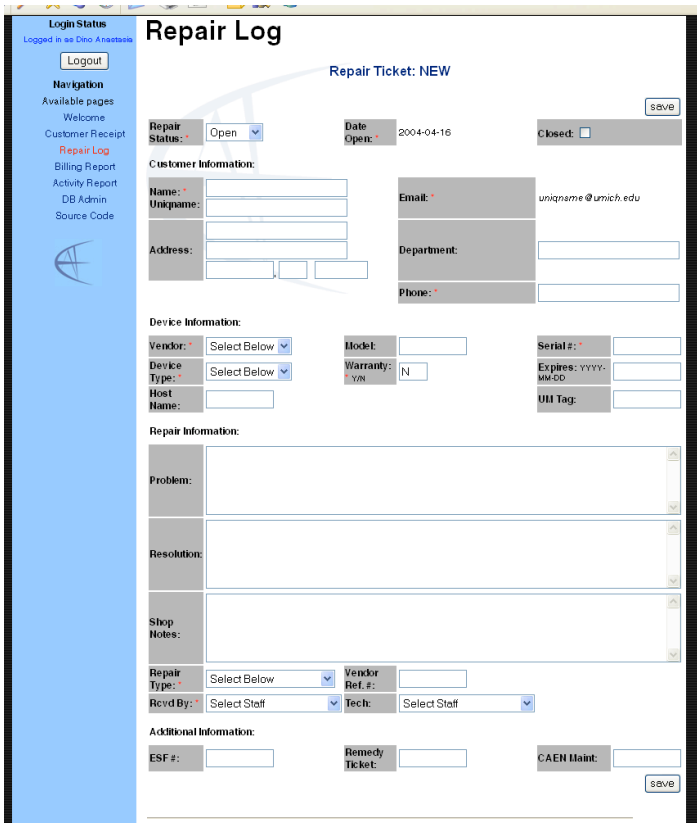
Appendix E: Customer Receipt - Screenshot and Code





Appendix F: Repair Log - Screenshot and Code





Appendix G: Repair Activity Reports – Screenshots and Code


Apple GSX | Dell Premier Access | CAEN Staff Pages | Apple | Dell | NEC | Parts Now
MichiganEngineering
University of Michigan College of Engineering
SEARCH
ABOUT CAEN ACCOUNTS LABS & SOFTWARE NETWORK EMAIL FACULTY & STAFF SUPPORT SERVICES
Login Status
Dino Anastasis
Logout
Reports
Select Report
Device Type
Repair Type
Vendor
Navigation
Available pages
Welcome
Customer Receipt
Repair Log
Billing Report
Activity Report
DB Admin
Source Code
Repair Activity Reports
Count by Vendor:
Vendor Total
Dell 4
Sun 2
Gateway 2
Apple 2
HP 2
Other 1
NEC 1
Copyright © College of Engineering, University of Michigan
Last edited on 04/10/2004

Appendix H: Create File

```
-- MySQL dump 9.08
--
-- Host: localhost    Database: caen
---------------------------------------------------------
-- Server version   4.0.14-standard

--
-- Current Database: caen
--

-- CREATE DATABASE /*!32312 IF NOT EXISTS*/ caen;

-- USE caen;
USE dinoa;


--
-- Table structure for table 'contact'
--

CREATE TABLE contact (
  uniqname varchar(10) NOT NULL default '',
  name varchar(30) NOT NULL default '',
  email varchar(30) NOT NULL default '',
  addressStreet text,
  addressUM varchar(30) default NULL,
  city varchar(20) default NULL,
  state char(2) default NULL,
  zip varchar(10) default NULL,
  department varchar(30) default NULL,
  phone varchar(25) NOT NULL default '',
  modTime timestamp(10) NOT NULL,
  modBy smallint(6) default NULL,
  PRIMARY KEY  (uniqname)
) TYPE=MyISAM;

--
-- Dumping data for table 'contact'
--
INSERT INTO contact VALUES ('tester1','Tester One','tester1@umich.edu','Street Address
1','2335B Media Union','Ann Arbor','AZ','48109-2094','CAEN - Labs - MediaU','(734)
615-1234','0311300921',1);
INSERT INTO contact VALUES ('dinoa','Dino Anastasia','dinoa@umich.edu','2281 Bonisteel
Blvd.','2335B Media Union','Ann Arbor','MI','48109-2094','CAEN','(734) 615-
5995','0311301044',1);
INSERT INTO contact VALUES ('tester2','Tester Two','tester2@bob.com','Street Address
2','1318 Markley','Ypsilanti','MI','48187','Physics','876-0987','0312061636',1);
INSERT INTO contact VALUES ('tester3','Tester Three','tester3@bob.com','Street Address
3','3505 South Quad','Ann Arbor','MI','48107','Personal','(847) 765-
0983','0402161055',1);
INSERT INTO contact VALUES ('tester4','Tester Four','tester4@bob.com','312 S. State
St.','422 Michigan Union','Ann Arbor','MI','48109','Astrology','5-
9876','0312061637',1);
INSERT INTO contact VALUES ('tester5','Tester Five','tester5@bob.com','','1404 Media
Union','','','2094','Engin-MSE','647-2112','0312061638',1);
INSERT INTO contact VALUES ('tester6','Tester Six','tester6@umich.edu','Street Address
6','tester6 Address 2','tester6 City','MI','48109','tester6 Department','(800) 55F-
Test','0312081035',1);
INSERT INTO contact VALUES ('tester7','Tester
Seven','tester7@umich.edu','','','','','','Sites','5-8173','0312081041',1);
```

```
--
-- Table structure for table 'equipment'
--

CREATE TABLE equipment (
  serialNum varchar(20) NOT NULL default '',
  vendor varchar(20) NOT NULL default '',
  warranty char(1) NOT NULL default '',
  warrantyExpire date default NULL,
  caenMaint varchar(5) default NULL,
  uniqname varchar(10) default NULL,
  hostName varchar(20) default NULL,
  model varchar(30) default NULL,
  propTag varchar(7) default NULL,
  modTime timestamp(10) NOT NULL,
  modBy smallint(6) default NULL,
  PRIMARY KEY  (serialNum)
) TYPE=MyISAM;

--
-- Dumping data for table 'equipment'
--

INSERT INTO equipment VALUES ('serialnum2','Apple','y','0000-00-
00','lvl1','dinoa','rivendell','PowerBook G4/500','a675987','0311301432',1);
INSERT INTO equipment VALUES ('serialnum1','Apple','y','2005-12-
25','','tester1','','G5/dual2G','','0312052051',3);
INSERT INTO equipment VALUES ('serialnum3','Sun','n','2004-11-
26','','dinoa','yule','Blade 1500','a657895','0311301608',1);
INSERT INTO equipment VALUES ('serialnum5','Dell','n','0000-00-
00','lvl2','tester4','mu3sp01','GX270/3.2G','','0311301517',1);
INSERT INTO equipment VALUES ('serialnum4','Dell','y','2004-12-
15','lvl3','tester2','biggy','GX270','','0312052044',3);
INSERT INTO equipment VALUES ('serialnum6','Dell','Y','2004-07-
01','lvl9','tester6','tester6host','tester6','Atest','0312081035',1);
INSERT INTO equipment VALUES ('serialnum7','Dell','N','0000-00-
00','','tester7','aha3','','','0312081041',1);

--
-- Table structure for table 'repair'
--
CREATE TABLE repair (
  csr int(11) NOT NULL auto_increment,
  date_open date NOT NULL default '0000-00-00',
  date_close date default NULL,
  received_by_tech_id smallint(6) default NULL,
  status_id tinyint(4) NOT NULL default '0',
  problem text,
  resolution text,
  serialno varchar(20) default NULL,
  contact varchar(10) NOT NULL default '',
  updated_by_tech_id smallint(6) NOT NULL default '0',
  updated_dt timestamp(10) NOT NULL,
  PRIMARY KEY  (csr)
) TYPE=MyISAM;
```

```
--
-- Dumping data for table 'repair'
--
INSERT INTO repair VALUES (1,'2003-11-26','2003-11-30',1,2,'Problem description
1','Resolution description 1','serialnum3','dinoa',2,'0312061230');
INSERT INTO repair VALUES (2,'2003-11-25','2003-12-04',2,2,'The CD drive does not
recognize CDs are in the drive when the computer is booted up. Customer unable to boot
from CD. Once the computer is started up there is an intermittent problem whether or
not the computer will actually read the CD.','CD-ROM
replaced.','serialnum3','dinoa',2,'0312052013');
INSERT INTO repair VALUES (3,'2003-11-25','0000-00-00',3,1,'problem
description','','serialnum7','tester5',1,'0312061239');
INSERT INTO repair VALUES (4,'2003-12-04','0000-00-00',2,1,'The display has white
blotches all over it.','','serialnum4','tester3',1,'0312061635');
INSERT INTO repair VALUES (5,'2003-11-25','0000-00-00',2,1,'C key on keyboard sticks
due to pop spill.','','serialnum1','tester4',1,'0312051957');
INSERT INTO repair VALUES (6,'2003-11-25','2003-12-05',3,2,'Mouse broken.','Mouse
replaced.','serialnum5','tester5',3,'0312061242');
INSERT INTO repair VALUES (7,'2003-11-25','2003-12-05',1,2,'Test on 12/5','Closed
problem','serialnum5','tester5',2,'0312052031');
INSERT INTO repair VALUES (8,'2003-11-25','0000-00-00',2,1,'Test with no techID
entered','','serialnum5','tester5',2,'0312062117');
INSERT INTO repair VALUES (9,'2003-11-25','2003-12-05',3,2,'Creating a new one with a
techID entered.','Just closing it','serialnum5','tester5',2,'0312051957');
INSERT INTO repair VALUES (10,'2003-12-05','0000-00-00',1,1,'This machine is
toast.','','serialnum2','tester3',1,'0312061240');
INSERT INTO repair VALUES (11,'2003-12-05','2003-12-05',2,2,'Machine freaked
out.','Called for an exorcism.','serialnum1','tester4',3,'0312052049');
INSERT INTO repair VALUES (12,'2003-12-08','2003-12-08',2,2,'This is a final test of
the full entry form.','This is a final resolution of the full entry
form.','serialnum6','dinoa',2,'0312081035');
INSERT INTO repair VALUES (13,'2003-12-08','0000-00-00',3,1,'Busted like a
bugger.','','serialnum7','tester5',2,'0312081041');

--
-- Table structure for table 'repair_notes'
--

CREATE TABLE repair_notes (
  csr int(11) NOT NULL,
  comment_time timestamp(10) NOT NULL,
  comment text,
  tech_id smallint(6) NOT NULL,
  PRIMARY KEY (csr, comment_time)
) TYPE=MyISAM;

--
-- Dumping data for table 'repair_notes'
--

INSERT INTO repair_notes VALUES (1,'0312061230', 'tech note 1',2);
INSERT INTO repair_notes VALUES (1,'0312071252', 'tech note 2',2);
INSERT INTO repair_notes VALUES (2,'0403071111', 'tech note 3',3);
INSERT INTO repair_notes VALUES (12,'0403091018', 'tech note 4',2);
```

```
--
-- Table structure for table 'repair_other'
--
CREATE TABLE repair_other (
  csr int(11) NOT NULL,
  esf varchar(10) default NULL,
  maint_xact_no varchar(15),
  remedy varchar(15) default NULL,
  vendor_ref varchar(25) default NULL,
  PRIMARY KEY (csr)
) TYPE=MyISAM;

--
-- Table structure for table 'repair_techs'
--
CREATE TABLE repair_techs (
  csr int(11) NOT NULL,
  tech_id smallint(6) NOT NULL,
  PRIMARY KEY (csr, tech_id)
) TYPE=MyISAM;

--
-- Dumping data for table 'repair_techs'
--

INSERT INTO repair_techs VALUES (1,2);
INSERT INTO repair_techs VALUES (1,1);
INSERT INTO repair_techs VALUES (2,2);
INSERT INTO repair_techs VALUES (3,2);
INSERT INTO repair_techs VALUES (4,2);
INSERT INTO repair_techs VALUES (5,3);
INSERT INTO repair_techs VALUES (6,2);
INSERT INTO repair_techs VALUES (7,2);
INSERT INTO repair_techs VALUES (8,2);
INSERT INTO repair_techs VALUES (9,1);
INSERT INTO repair_techs VALUES (10,3);
INSERT INTO repair_techs VALUES (11,2);
INSERT INTO repair_techs VALUES (12,1);
INSERT INTO repair_techs VALUES (13,2);

--
-- Table structure for table 'repair_status'
--
CREATE TABLE repair_status (
  status_id tinyint(4) NOT NULL auto_increment,
  status varchar(20) NOT NULL,
  PRIMARY KEY (status_id)
) TYPE=MyISAM;

--
-- Dumping data for table 'repair_status'
--
INSERT INTO repair_status VALUES (1,'Open');
INSERT INTO repair_status VALUES (2,'Closed');

--
-- Table structure for table 'repair_activity'
--
CREATE TABLE repair_activity (
  csr int(11) NOT NULL,
  type_id tinyint(4) NOT NULL,
  vendor_id tinyint(4) NOT NULL,
  equipment_id tinyint(4) NOT NULL,
  PRIMARY KEY (csr)
) TYPE=MyISAM;
```

```
--
-- Dumping data for table 'repair_activity'
--

INSERT INTO repair_activity VALUES (1,5,6,1);
INSERT INTO repair_activity VALUES (2,3,3,5);
INSERT INTO repair_activity VALUES (3,3,6,5);
INSERT INTO repair_activity VALUES (4,5,7,4);
INSERT INTO repair_activity VALUES (5,6,2,3);
INSERT INTO repair_activity VALUES (6,5,2,2);
INSERT INTO repair_activity VALUES (7,5,2,1);
INSERT INTO repair_activity VALUES (8,8,1,5);
INSERT INTO repair_activity VALUES (9,2,3,5);
INSERT INTO repair_activity VALUES (10,2,4,3);
INSERT INTO repair_activity VALUES (11,2,4,3);
INSERT INTO repair_activity VALUES (12,1,5,2);
INSERT INTO repair_activity VALUES (13,4,2,1);

--
-- Table structure for table 'repair_type'
--

CREATE TABLE repair_type (
  type_id tinyint(4) NOT NULL auto_increment,
  type_name varchar(50) NOT NULL,
  PRIMARY KEY (type_id)
) TYPE=MyISAM;

--
-- Dumping data for table 'repair_type'
--

INSERT INTO repair_type VALUES (1,'CSC-dept');
INSERT INTO repair_type VALUES (2,'CSC-personal');
INSERT INTO repair_type VALUES (3,'CSC-personal-student');
INSERT INTO repair_type VALUES (4,'CAEN');
INSERT INTO repair_type VALUES (5,'CAEN-Labs');
INSERT INTO repair_type VALUES (6,'CAEN-Labs-DC');
INSERT INTO repair_type VALUES (7,'ESF');
INSERT INTO repair_type VALUES (8,'Unbillable');
INSERT INTO repair_type VALUES (9,'CSC-Internal');
INSERT INTO repair_type VALUES (10,'Other');

--
-- Table structure for table 'repair_vendor'
--

CREATE TABLE repair_vendor (
  vendor_id tinyint(4) NOT NULL auto_increment,
  vendor_name varchar(50) NOT NULL,
  PRIMARY KEY (vendor_id)
) TYPE=MyISAM;

--
-- Dumping data for table 'repair_vendor'
--

INSERT INTO repair_vendor VALUES (1,'Apple');
INSERT INTO repair_vendor VALUES (2,'Dell');
INSERT INTO repair_vendor VALUES (3,'Gateway');
INSERT INTO repair_vendor VALUES (4,'HP');
INSERT INTO repair_vendor VALUES (5,'NEC');
INSERT INTO repair_vendor VALUES (6,'Sun');
INSERT INTO repair_vendor VALUES (7,'Other');
```

```
--
-- Table structure for table 'repair_equipment'
--

CREATE TABLE repair_equipment (
  equipment_id tinyint(4) NOT NULL auto_increment,
  equipment_name varchar(50) NOT NULL,
  PRIMARY KEY (equipment_id)
) TYPE=MyISAM;

--
-- Dumping data for table 'repair_equipment'
--

INSERT INTO repair_equipment VALUES (1,'Desktop');
INSERT INTO repair_equipment VALUES (2,'Laptop');
INSERT INTO repair_equipment VALUES (3,'Printer');
INSERT INTO repair_equipment VALUES (4,'Monitor');
INSERT INTO repair_equipment VALUES (5,'Other');

--
-- Table structure for table 'repair_staff'
--

CREATE TABLE repair_staff (
  tech_id smallint(6) NOT NULL auto_increment,
  uniqname varchar(10) NOT NULL default '',
  name varchar(30) NOT NULL default '',
  password varchar(20) NOT NULL default '',
  access char(1) default NULL,
  PRIMARY KEY  (tech_id)
) TYPE=MyISAM;

--
-- Dumping data for table 'repair_staff'
--

INSERT INTO repair_staff VALUES (1,'dinoa','Dino Anastasia','pw','a');
INSERT INTO repair_staff VALUES (2,'tech','Test Technicican','pw','t');
INSERT INTO repair_staff VALUES (3,'marlag','Marla Gomez','pw','a');
```

Appendix I: SQL Table Descriptions

```
mysql> describe contact;
+---------------+---------------+------+-----+---------+-------+
| Field         | Type          | Null | Key | Default | Extra |
+---------------+---------------+------+-----+---------+-------+
| uniqname      | varchar(10)   |      | PRI |         |       |
| name          | varchar(30)   |      |     |         |       |
| email         | varchar(30)   |      |     |         |       |
| addressStreet | text          | YES  |     | NULL    |       |
| addressUM     | varchar(30)   | YES  |     | NULL    |       |
| city          | varchar(20)   | YES  |     | NULL    |       |
| state         | char(2)       | YES  |     | NULL    |       |
| zip           | varchar(10)   | YES  |     | NULL    |       |
| department    | varchar(30)   | YES  |     | NULL    |       |
| phone         | varchar(25)   |      |     |         |       |
| modTime       | timestamp(10) | YES  |     | NULL    |       |
| modBy         | smallint(6)   | YES  |     | NULL    |       |
+---------------+---------------+------+-----+---------+-------+
12 rows in set (0.00 sec)

mysql> describe equipment;
+----------------+---------------+------+-----+---------+-------+
| Field          | Type          | Null | Key | Default | Extra |
+----------------+---------------+------+-----+---------+-------+
| serialNum      | varchar(20)   |      | PRI |         |       |
| vendor         | varchar(20)   |      |     |         |       |
| warranty       | char(1)       |      |     |         |       |
| warrantyExpire | date          | YES  |     | NULL    |       |
| caenMaint      | varchar(5)    | YES  |     | NULL    |       |
| uniqname       | varchar(10)   | YES  |     | NULL    |       |
| hostName       | varchar(20)   | YES  |     | NULL    |       |
| model          | varchar(30)   | YES  |     | NULL    |       |
| propTag        | varchar(7)    | YES  |     | NULL    |       |
| modTime        | timestamp(10) | YES  |     | NULL    |       |
| modBy          | smallint(6)   | YES  |     | NULL    |       |
+----------------+---------------+------+-----+---------+-------+
11 rows in set (0.00 sec)

mysql> describe repair;
+--------------------+---------------+------+-----+------------+----------------+
| Field              | Type          | Null | Key | Default    | Extra          |
+--------------------+---------------+------+-----+------------+----------------+
| csr                | int(11)       |      | PRI | NULL       | auto_increment |
| date_open          | date          |      |     | 0000-00-00 |                |
| date_close         | date          | YES  |     | NULL       |                |
| received_by_tech_id | smallint(6)  | YES  |     | NULL       |                |
| status_id          | tinyint(4)    |      |     | 0          |                |
| problem            | text          | YES  |     | NULL       |                |
| resolution         | text          | YES  |     | NULL       |                |
| serialno           | varchar(20)   | YES  |     | NULL       |                |
| contact            | varchar(10)   |      |     |            |                |
| updated_by_tech_id | smallint(6)   |      |     | 0          |                |
| updated_dt         | timestamp(10) | YES  |     | NULL       |                |
+--------------------+---------------+------+-----+------------+----------------+
11 rows in set (0.00 sec)
```

```
mysql> describe repair_activity;
+--------------+------------+------+-----+---------+-------+
| Field        | Type       | Null | Key | Default | Extra |
+--------------+------------+------+-----+---------+-------+
| csr          | int(11)    |      | PRI | 0       |       |
| type_id      | tinyint(4) |      |     | 0       |       |
| vendor_id    | tinyint(4) |      |     | 0       |       |
| equipment_id | tinyint(4) |      |     | 0       |       |
+--------------+------------+------+-----+---------+-------+
4 rows in set (0.00 sec)

mysql> describe repair_equipment;
+----------------+-------------+------+-----+---------+----------------+
| Field          | Type        | Null | Key | Default | Extra          |
+----------------+-------------+------+-----+---------+----------------+
| equipment_id   | tinyint(4)  |      | PRI | NULL    | auto_increment |
| equipment_name | varchar(50) |      |     |         |                |
+----------------+-------------+------+-----+---------+----------------+
2 rows in set (0.00 sec)

mysql> describe repair_notes;
+--------------+---------------+------+-----+---------+-------+
| Field        | Type          | Null | Key | Default | Extra |
+--------------+---------------+------+-----+---------+-------+
| csr          | int(11)       |      | PRI | 0       |       |
| comment_time | timestamp(10) | YES  | PRI | NULL    |       |
| comment      | text          | YES  |     | NULL    |       |
| tech_id      | smallint(6)   |      |     | 0       |       |
+--------------+---------------+------+-----+---------+-------+
4 rows in set (0.00 sec)

mysql> describe repair_other;
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| csr           | int(11)     |      | PRI | 0       |       |
| esf           | varchar(10) | YES  |     | NULL    |       |
| maint_xact_no | varchar(15) | YES  |     | NULL    |       |
| remedy        | varchar(15) | YES  |     | NULL    |       |
| vendor_ref    | varchar(25) | YES  |     | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)

mysql> describe repair_staff;
+----------+-------------+------+-----+---------+----------------+
| Field    | Type        | Null | Key | Default | Extra          |
+----------+-------------+------+-----+---------+----------------+
| tech_id  | smallint(6) |      | PRI | NULL    | auto_increment |
| uniqname | varchar(10) |      |     |         |                |
| name     | varchar(30) |      |     |         |                |
| password | varchar(20) |      |     |         |                |
| access   | char(1)     | YES  |     | NULL    |                |
+----------+-------------+------+-----+---------+----------------+
5 rows in set (0.00 sec)

mysql> describe repair_status;
+-----------+-------------+------+-----+---------+----------------+
| Field     | Type        | Null | Key | Default | Extra          |
+-----------+-------------+------+-----+---------+----------------+
| status_id | tinyint(4)  |      | PRI | NULL    | auto_increment |
| status    | varchar(20) |      |     |         |                |
+-----------+-------------+------+-----+---------+----------------+
2 rows in set (0.00 sec)
```

```
mysql> describe repair_techs;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| csr     | int(11)     |      | PRI | 0       |       |
| tech_id | smallint(6) |      | PRI | 0       |       |
+---------+-------------+------+-----+---------+-------+
2 rows in set (0.01 sec)

mysql> describe repair_type;
+-----------+-------------+------+-----+---------+----------------+
| Field     | Type        | Null | Key | Default | Extra          |
+-----------+-------------+------+-----+---------+----------------+
| type_id   | tinyint(4)  |      | PRI | NULL    | auto_increment |
| type_name | varchar(50) |      |     |         |                |
+-----------+-------------+------+-----+---------+----------------+
2 rows in set (0.00 sec)

mysql> describe repair_vendor;
+-------------+-------------+------+-----+---------+----------------+
| Field       | Type        | Null | Key | Default | Extra          |
+-------------+-------------+------+-----+---------+----------------+
| vendor_id   | tinyint(4)  |      | PRI | NULL    | auto_increment |
| vendor_name | varchar(50) |      |     |         |                |
+-------------+-------------+------+-----+---------+----------------+
2 rows in set (0.00 sec)
```

Appendix J: Other Source Code

Please see attached source code pages.