Rithik Bansal
CSS 498 – SPRING 2019
Professor Arkady Retik
June 9, 2019

Independent Study
Web Application for a Room Condition Report System

Done by:
Student name: **Rithik Bansal**
Faculty sponsor: **Arkady Retik**

Rithik Bansal
CSS 498 – SPRING 2019
Professor Arkady Retik
June 9, 2019

Table of Contents

Rithik Bansal
CSS 498 – SPRING 2019
Professor Arkady Retik
June 9, 2019

Purpose

The purpose of this project is to design an online system which will allow any residential life related organization to be able to conduct the room condition reports electronically. The motivation came from the intent of making the system paperless.

Learning goals

The goals of the project:

- Create a web interface using php.
- Create a database for the system to store data in an organized fashion
- Connect the web interface with the database

Implementation

The general steps taken to complete the project (bullet points in italics were not completed in this project sprint):

1. Created an Object Role Model diagram for the database using VisioModeler
2. Created a corresponding Logical Data model for the database using VisioModeler
3. Normalize the logical data model
4. Created the database using MS Access, and created another database using MySQL
5. Setup server to host the database using the UW provided vergill resource
6. Setup the database on the server and add dummy values to it
7. Create the .php files based on the number of queries and pages needed on the web interface
8. *Test the developed product through means of beta testing mechanism.*
9. *Launch product*

The database maintains a condition report for each part of the most common unit, namely:

- living room: walls, couch, windows, blinds
- bedroom: walls, bed, closet
- bathroom: sink, shower, toilet
- kitchen: dishwasher, refrigerator, stove

Besides the sections of a unit, it also maintains the information related to tenants:

- residents: Full name, studentId, contact number
- emergency contact: full name, email, contact number

We also maintain the sectors related to the unit itself:

- building: apartment floor, apartment number, building name

Rithik Bansal
CSS 498 – SPRING 2019
Professor Arkady Retik
June 9, 2019
The condition of each individual component is graded on an alphabetic scale which is as follows:

Condition Grade	**Meaning**
A	Perfect condition
B	Minor scratched (wear and tear)
D	Major damages
F	Major damages with unhygienic maintenance of the place

Detailed implementation of all the completed steps mentioned above, with screenshots and results:

- **Object Role Model diagram**

  Object role model is a way to demonstrate the domain concepts. It reflects the entities, values and the relationship between those objects. It also describes the constraints assigned to each relationship. Figure 1 shows the ORM model created for the project. I reviewed room condition report forms from different organizations that were available online to determine the objects I would need for mine.

Rithik Bansal
CSS 498 – SPRING 2019
Professor Arkady Retik
June 9, 2019

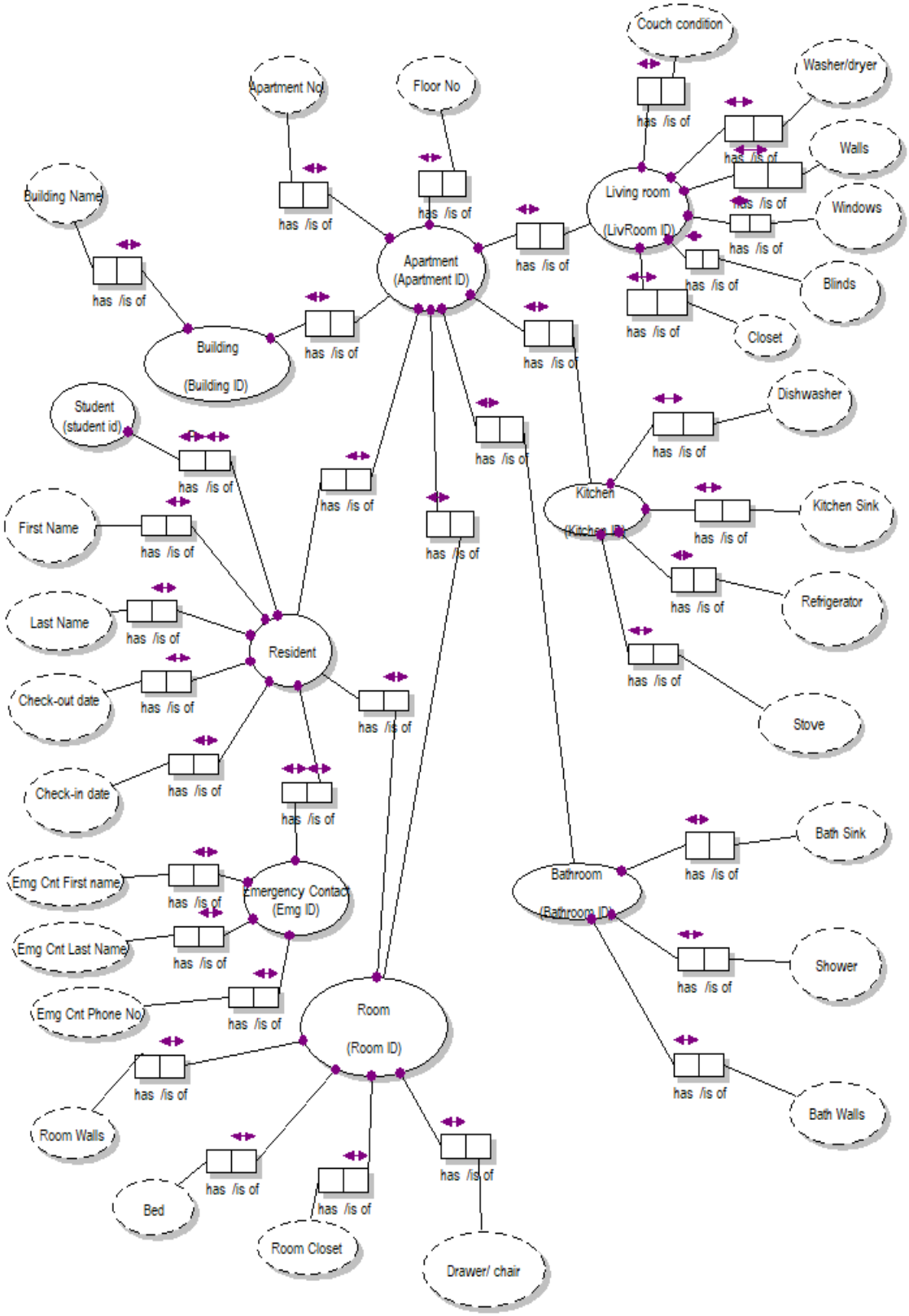

Figure 1 Object Role Model for the RCR System

Rithik Bansal
CSS 498 – SPRING 2019
Professor Arkady Retik
June 9, 2019

- **Logical Data model**

  Logical Data model is a data model drawn for a specific data domain which is showcased independent of the main database. It extends from the object role model and shows the primary keys, table names, table values, and relationships between individual tables. The tables are drawn based on the relationship between the entities and values in the object role model. The relationship between the tables in a logical data model represent the relationship between different entity objects in an object role model. Figure 2 shows the LDM created for the project. It was created using the VisioModeler software.

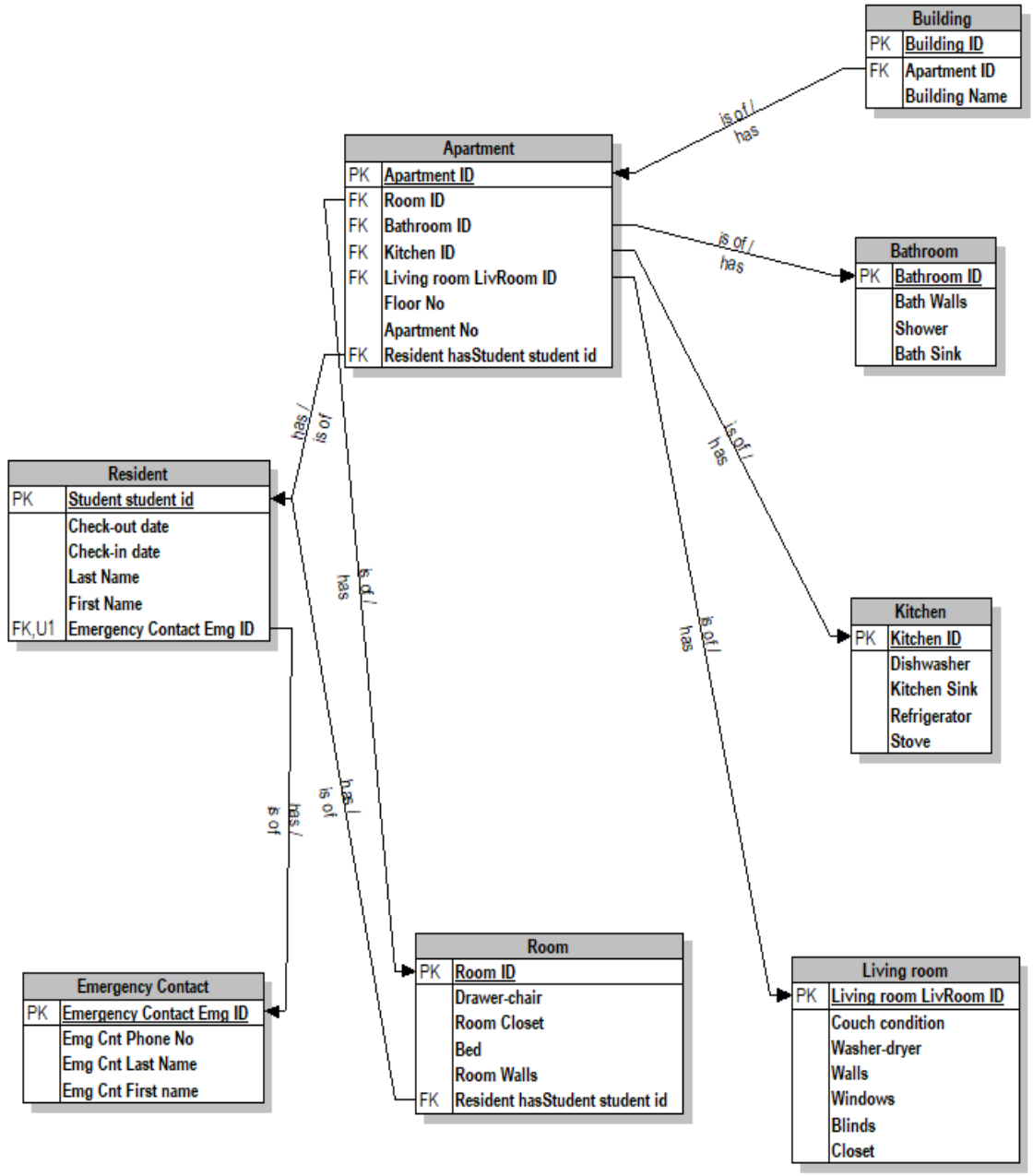

Figure 2 Logical data model for the RCR system

Rithik Bansal
CSS 498 – SPRING 2019
Professor Arkady Retik
June 9, 2019

- **Database in MS Access**

  After creating the ORM and LDM in VisioModeler, I created a local database using MS Access. I added the tables as shown in the LDM in figure 2. Relationships were added based on the key constraints and type. Also created some SQL queries in MS Access to verify that the database tables, and relationships were created correctly. Figure 3 shows a screenshot of the relationship between the database tables created in MS Access. The database file itself has been attached to the project submission, named "RCRMain.accdb". Figure 4 shows the test queries used to test the correctness of the developed database in MS Access.

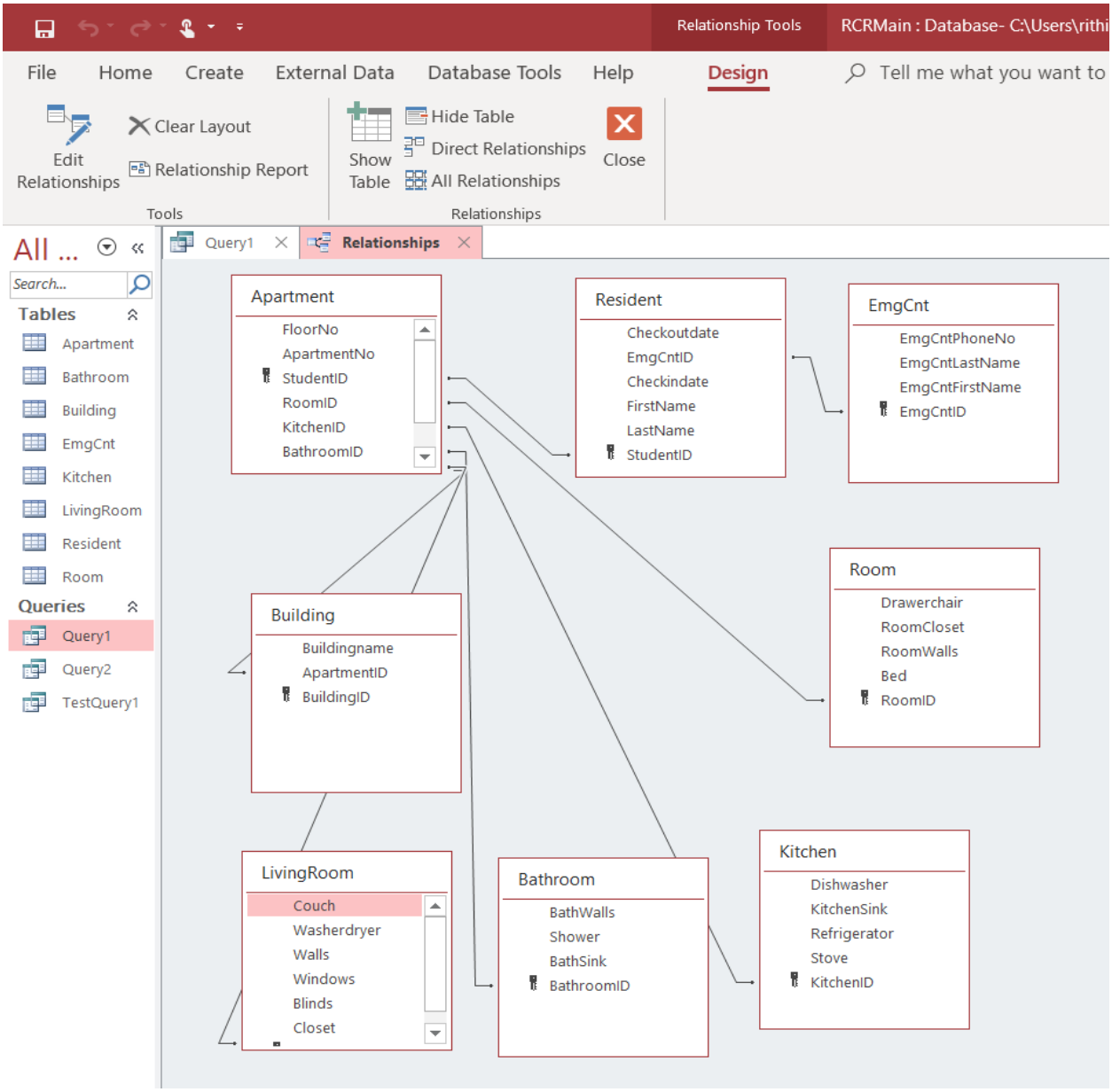

Figure 3 MS Access database relationship screenshot

Rithik Bansal
CSS 498 – SPRING 2019
Professor Arkady Retik
June 9, 2019

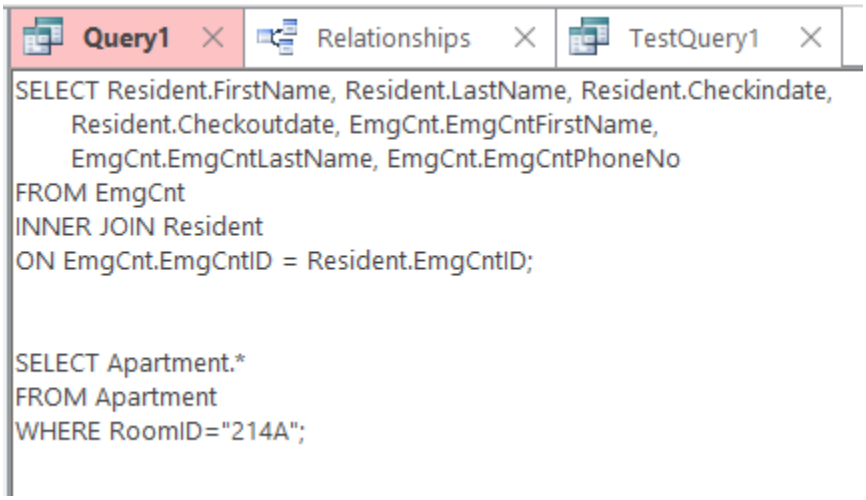

Figure 4 Screenshot of SQL queries used for testing

- **Database using MySQL**

  After creating the ORM and LDM in VisioModeler, I created a .SQL file as well. Using MySQL, I created the database, tables, added dummy values for each database, assigned primary and foreign keys to the values and added the relationships between each table. Figure 5 shows the screenshot of MySQL code written to create the table and add dummy values to it. Figure 6 shows the screenshot of MySQL code written to assign primary key constraints and introducing relationships between the created tables. The whole database code is in the room_condition_report.sql.

```
-- Table structure for `building`
--

DROP TABLE IF EXISTS
    `building`;
CREATE TABLE IF NOT EXISTS `building`(
    `BuildingID` VARCHAR(10) NOT NULL DEFAULT '',
    `BuildingName` VARCHAR(10) DEFAULT NULL
) ENGINE = INNODB DEFAULT CHARSET = latin1;
--

-- putting data for table `building`
--

INSERT INTO `building`(`BuildingID`, `BuildingName`)
VALUES('1000', 'Pine'),('1001', 'Spruce'),('1002', 'Aspen');
--
```

Figure 5 Screenshot of MySQL code to create table and add values to it

Rithik Bansal
CSS 498 – SPRING 2019
Professor Arkady Retik
June 9, 2019

```
    -- Indexes for table `resident`
    --

ALTER TABLE
    `resident` ADD PRIMARY KEY(`StudentID`),
    ADD KEY `RoomIdx`(`RoomID`),
    ADD KEY `EmgCntIdx`(`EmgCntID`),
    ADD KEY `ApartmentIdx`(`ApartmentID`);
    --

    -- Indexes for table `EmgCnt`
    --

ALTER TABLE
    `EmergencyContact` ADD PRIMARY KEY(`EmgCntID`);
ALTER TABLE
    `apartment` ADD CONSTRAINT `apartment_ibfk_1` FOREIGN KEY(`RoomID`) REFERENCES `room`(`RoomID`),
    ADD CONSTRAINT `apartment_ibfk_2` FOREIGN KEY(`KitchenID`) REFERENCES `kitchen`(`KitchenID`),
    ADD CONSTRAINT `apartment_ibfk_3` FOREIGN KEY(`LivingRoomID`) REFERENCES `livingRoom`(`LivingRoomID`),
    ADD CONSTRAINT `apartment_ibfk_4` FOREIGN KEY(`BuildingID`) REFERENCES `building`(`BuildingID`),
    ADD CONSTRAINT `apartment_ibfk_5` FOREIGN KEY(`BathroomID`) REFERENCES `bathroom`(`BathroomID`);
```

Figure 6 Screenshot of MySQL code to assign keys and add relationships between tables

- **Server setup and connecting with database**

  Students at UW are allowed to use vergill to host a server that allows web publishing. I followed the steps provided by the UW IT to setup my vergil server. Once the server was created, I added phpMyAdmin files to it. I picked a port number that was not being used by anyone to host my server. Once the server was live, I enabled MySQL on it by downloading and installing the relevant files. I used the .sql file discussed in the previous section and shown in figure 5 and 6 to create the database on my hosted server. I tested it rigorously to ensure correctness.

- **Creating php files for the web interface**

  I wrote php scripts as I worked on the front end of my database. Firstly, I planned out the layout of my web interface and then wrote SQL queries that will be needed to retrieve and display data. I have already given you access to the server, so you can view what the web interface to the database looks like as of now. The link to the hosted website is:

  https://students.washington.edu/sumbla/phpmyadmin

  Figure 7 shows a screenshot of the home page of the web interface to the database.

Rithik Bansal
CSS 498 – SPRING 2019
Professor Arkady Retik
June 9, 2019

Figure 7 Screenshot of the web interface

Problems Encountered

There were a number of problems encountered during the product development process. Here are a few major ones:

1. The vergil resource provided by UW for student to host on assigned server port, did not allow me to assign primary key to foreign key relationships. This was primarily the reason that I created my database in both MySQL (which was used on the hosted server) and the MS Access database where I could assign primary key to foreign key relationships (and be able to manipulate data in the database using queries).
2. The downloading of MySQL files on the server was erroneous for a reason I am unaware about. It was consistently failing to download a mysql.sock file for me. A loop around that I went for with this was to use my friends' credentials to host the server.

Rithik Bansal
CSS 498 – SPRING 2019
Professor Arkady Retik
June 9, 2019

Future work

The project is not complete and there are various other features to be developed. Some of the major features that are required and will be developed next are:

1. As an online system, I want to be able to make edit to the tenant records as and when required. I tried implementing this step, but I was running into errors in php.
2. I also want to introduce a login page where the visitors will be required to login using their account username and password. Alongside that, I want to set restrictions on the level of access each user needs to have.
3. I need to do more research and add a "Estimate fine" calculator that will show the tenant the approximate fine they might get charged based on the current condition of the unit they are staying in.

Conclusion

All in all, this project has seen fair amount of development. I hope to develop more features and update the web server's user interface for the database over the summer quarter. I also need to do more in-depth research on how the report is used to bill tenants with charges, and hopefully be able to add a payment feature.

Files submitted

All the files submitted along with this report are:

- Independent Study Report Rithik Bansal.pdf
- Php folder:
  - Base.css
  - Config.inc.php
  - Header.inc.php
  - Index.php
  - List_emergencyCnt.php
  - List_residents.php
  - ListBathroomCondition.php
  - ListKitchenCondition.php
  - ListLivingRoomCondition.php
  - ListRoomCondition.php
  - Update_resident.php
- LDMforRCR.png
- ORMforRCR.png
- RCRMain.accdb
- RCRORM.IMO
- Room_condition_report.sql