**University of Louisiana at Lafayette**

**Requirements**

**UL Housing Assignment – Cmps 453**

Brandin Jefferson

Issa Samake

Zach Danjeau

Brian Okoye

Yee Wong

Jaquincy Nelson

Dr. Ashok Kumar

**Table of Contents**

Abstract ………………………………………………………………………………….3

Introduction ……………………………………………………………………………...3 List of Figures…………………………………………………………………………….3

Rationale.…………………………………………………………………………………4

Non Functional Requirements..…………………………………………………………..4

**Abstract**

The document below will address the following functional and non-functional requirements presented in the project. The use case diagram will exhibit the basic use cases the customer will apply within the system. Besides the functional requirements given by the customer, there are a few non-functional requirements the team has decided upon such as user accessibility, reliability, and application efficiency.

**Introduction**

**The UL Housing department is in need of a Room request web application that will interact with databases within the department. The application will consist of users (students) sending a request through a website, which will then be placed in database list . The administrative users will be allows to access the database at any time and add or remove request from the current database .**

**USE CASE DIAGRAM**

**Use Cases (Scenarios)**

Student

Administrator

Database

Database

Database

Database

Figure 1: Room Storage Use Case

|  |  |
| --- | --- |
| Use Case Name | Requesting a new room |
| Participating Actors | Students, Administrators |
| Flow of Events | 1. Student logs into Housing website. 2. Student navigates to request form 3. Request form is filled out. 4. Form is sent to the database. 5. Administrators are alerted to new request by e-mail 6. Admins open program 7. System displays list of unanswered room requests 8. Admin selects a room request 9. System displays information pertaining to that request 10. Admin accepts or rejects request 11. Letter sent to student alerting them of response. 12. (If accepted) System replaces student’s current file with new one 13. Student’s file removed from list of unanswered room requests 14. List of unanswered room requests displayed |
| Alternate Flows | 3a. User cancels request.  3b. User is returned to the Housing homepage.  11a. If rejected, rejection e-mail sent to student |
| Entry Condition(s) | Student has a valid CLID and password. |
| Exit Condition(s) | Form is correctly filled out and entered; alternately, user cancels request. |
| Special Requirements | N/A |

**RATIONALE**

It was determined that, in order to efficiently perform any of the client’s tasks, a database would be necessary. Since all of the tasks needed one, it was decided that all of the projects would be completely based around the database. The only extra things required would be interfaces unique to the individual projects.

**NON-FUNCTIONAL REQUIREMENTS**

**Product requirement**

The requirements include execution speed, reliability and accessibility of the system .The room-change system should be available to all current residents on campus during weekdays and weekends. Downtime for site maintenance shall not exceed 24 hours. The system should be easy to use by residents and should be organized in such a way that user errors are minimized.

**Organizational requirement**

Users of the room-change system shall authenticate themselves using their CLID and user-defined password.Priority of the room-change service should follow the first-come, first-served practice.

**External requirement**

The system shall implement resident privacy provisions.Maintenance should be done on the system regularly.

**EVIDENCE OF CONFIGURATION MANAGEMENT**

Git Repository: https://github.com/brandinjefferson/CMPS453-Docs



**REFERENCES**

Badgerati. “Software Engineering – Use Case Diagrams / Descriptions.” *Computer Science Source*. Computer Science Source, 22 Nov 2009. Web. 29 Sep 2014.