Andrew Hopkins

IST 311

8/29/2018

Register Customer Use Case Specifications

**Functional Requirements**

This use case will need to provide management with control in the application process so that management can ensure those applying for pool membership are part of the corresponding community (i.e. an apartment complex, a gated neighborhood, etc.) and, when applicable, ensure that applicants are not behind on rent and have paid the necessary fees for membership. The use case will need to collect all pertinent information regarding pool members: full name, age, address, allergies, emergency contact information, and swimming ability. Lastly the use case will need to store pool member information and their photo in a digital profile so that they can be easily identified for check-in at the pool.

**Nonfunctional Requirements**

Only management will have access to the register customer use case, this provides delegation of a sensitive task to an appropriate level of supervision and thereby ensures the security of pool members. The data will likely need to be stored in a database server. Therefore, the application will need to be able to extract information from database code such as PHP or MySQL Lite. The input form will need need to be clear and user-friendly while also being comprehensive. Solid graphic design and formatting should make input fields easy and obvious. Error checking will need to occur before submitting data to the database. Management will need to upload pictures from digital camera or scanner, the app will not include picture-taking software itself.

**Task Analysis**

1. A manager selects Register Customer from the navigational menu
2. The application displays a form with several input fields for general applicant information as well as radio button options for swimming ability.
3. The manager fills these out according to information supplied by the applicant
4. The manager selects the submit button
5. The application either displays a success message in the case of successful submission, or an error message addressing unfilled fields, faulty server connection, etc.

**Design Challenges:**

* The input form needs to be designed in a way that is simple, efficient, and easy for the user; the process should enhance the registration experience, not make it more difficult.
* The photo that is used for the profile will likely need to fit a pre-established dimension or the application will need to support photo cropping and positioning.
* The input form should provide sufficient error-checking before submission to minimize incorrect data (ages and zip codes need to be integers, swimming ability must be selected, etc.)

**Pre-Conditions**

* Applicant is verified by management as a member of the community
* Applicant is in good-standing and has paid any necessary fees
* Applicant consents to pool rules and agrees to uphold them
* Applicant is physically at the management office
* Applicant’s picture has been taken or is on file
* Manager successfully logs into the app
* Connection to the database server is present

**Post-Conditions**

* Success message is displayed
* Database stores and maintains membership data
* Member is able to check-in at the pool immediately following registration