Cemetery Management System

Samuel Kearns - Project Manager (skearns4)

Brittany Regrut - Quality Assurance (bnr12)

Louis Seefeld - Software Developer (lfs13)

Emily Crabb - Software Developer (ejc44)

Evan Alexander - Software Developer (ea914)

Due date: 27 OCTOBER 2015

CS 1530 - SPRINT 3 DELIVERABLE

Sprint 3 Accomplishments

During this sprint, we added one new user story relating to searching by last or full name and revisited our previously completed user stories. In the previous sprints, we did not implement any formal unit tests using JUnit assertions. Therefore, in this sprint, we changed our development process to focus more on testing. As such, we added a mainTest.java file containing assertions relating to both our new user story and the user stories from the previous sprint. Using these unit tests, we discovered and corrected defects. The details on the user stories and justification for choosing them are included in the next two sections, while the defects are discussed in the final section.

We met with the customer once during this sprint. We specifically met with Lonnie, the cemetery manager, rather than Rob, the director of the synagogue who initiated the project. During this meeting, we showed Lonnie our existing program, which at the time only searched by plot number and first name. His reactions are described in the section on customer feedback. He also provided us with a CSV file containing some of the real data from their existing database. This will shape our future development of the program, as this is the first time we have learned exactly what fields need to be stored for each entry in the database. We also anticipate implementing the user story relating to accessing the existing data during our next sprint by creating some sort of method to convert their Excel-generated CSV files into an H2 database compatible with our program.

During this sprint, we did not greatly alter our methods of communication. We continued to use Slack as our primary method of communication. However, as one of the problems we identified in our previous sprint was the small number of in-person meetings, we endeavored to meet more often. We had more short meetings after class to discuss each group member’s status, as well as some longer meetings on the weekends where we actually worked on coding, doing code reviews, and writing the report. However, everyone already had the IntelliJ IDE and H2 database working on their computers, so remote communication was easier and in-person communication was less critical. Overall, we felt our communication improved this sprint.

One problem we had was figuring out how to do code reviews. None of us have extensive experience with GitHub, so we had some difficulty in determining a good way to do code reviews directly on the website using pull requests. As such, we primarily did code reviews in person when we met on the weekends.

User Stories Completed and Updated Backlog

Link to the repository:<https://github.com/skearns4/Synagogue-Cemetery>

In this sprint, we completed one new user story. As listed in the backlog included in our previous deliverable, this completed user story is:

|  |  |  |
| --- | --- | --- |
| Rank | User Story | Difficulty Estimate |
| 2 | As a user, I want to be able to search the burial plot entries by last name or full name (rather than just first name) so that I can look up information on a specific client. | 2 |

However, we also returned to the four user stories we had believed were finished in the previous sprint. As listed in our previous deliverable, these user stories were:

|  |  |  |
| --- | --- | --- |
| Rank | User Story | Difficulty Estimate |
| 1 | As a user, I want to view information for existing burial plots so that I can see if plots are available and look up information for existing clients. | 4 |
| 2 | As an administrator, I want to create a new entry so that I can keep track of newly sold plots. | 8 |
| 4 | As a user, I want to be able to search the burial plot entries by name so that I can look up information on a specific client. | 2 |
| 5 | As a user, I want to be able to search the burial plot entries by interment number so that I can look up information on a specific client. | 2 |

The justification for implementing the one user story and revisiting previous user stories is given in the next section of this report, while some defects and limits on the functionality are given in the final section. We have not added any additional user stories, so the updated backlog of user stories is:

|  |  |  |
| --- | --- | --- |
| Rank | User Story | Difficulty Estimate |
| 1 | As an administrator, I want to be able to access our existing burial plot information in this new database so that no information is lost in the transfer. | 4 |
| 2 | As a user, I want to be able to print search results and individual user entries so that I can easily share the information with others. | 4 |
| 3 | As a user, I want to be able to view a map of the cemetery, including number of open graves per plot so that I can more easily show clients the layout of the cemetery. | 8 |
| 4 | As a user, I want to be able to able to see the status of plots that are in close proximity to a current plot so that I can easily determine whether clients can purchase adjacent plots. | 8 |
| 5 | As a user, I want to be able to click on the map of the cemetery so that I can easily look up information for a specific plot. | 8 |
| 6 | As an administrator, I want to be able to manage payments for the plots so that it is easier to coordinate plot and payment information. | 16 |

Justification for Choosing These User Stories

We chose to implement the user story relating to searching by last or full name because this functionality is critical to making the program useful. This user story was ranked second in priority in the backlog in our previous deliverable, and the customer values this functionality highly. This functionality is also already included in the user’s current cemetery management program, so it was necessary to implement this user story in order to reach our goal of making our program at least as useful as the current program. In the previous sprint, we only implemented searching by first name, which would not be very useful for a database with the real cemetery data which contains thousands of entries.

We also revisited the previous user stories relating to searching by interment number and name and creating a new entry because we discovered new defects and code deficiencies relating to these user stories. For one, we did not implement any unit tests using JUnit assertions in the previous sprint. As a result, all of our user stories were only tested informally with the programmers entering an input and checking the result against the expected output. In this sprint, we added JUnit assertions in the mainTest.java file that tested the newly implemented user story about searching by last or full name as well as the previous user stories relating to searching by plot number and first name and creating a new entry. As a result, we discovered and then fixed a defect that each search only returned one result even if there were multiple matching entries in the database, as will be discussed in the final section on defects.

The reason we did not implement the user stories relating to accessing existing data and adding a map graphical user interface was because of problems with the customer. With respect to the user story about accessing existing data, the customer only provided us with some of the real cemetery data on the Friday before the sprint was due. This did not allow us enough time to add this data to our database, as we will need to write a method to read their data from the Excel file they provided and add it to our H2 database file. With respect to the user stories relating to the cemetery map, the customer has not provided us with a map with sufficient detail to create a map GUI. The map they have on their website is too low resolution to read the plot numbers, so we cannot link the plots on the map to entries in the database. Even if we manually link each plot on a map to an entry in the database, we will still at least need a legible map.

Customer Feedback

During this sprint, we were able to meet with Lonnie, the cemetery manager and one of the primary stakeholders. While he does not have a very strong technical background, he is the sole user of the software, so his input is very valuable to us and the project. We showed him a working version of our code that could search by interment number and first name. He was satisfied with the overall format of the GUI, including the search buttons and display window showing the results. However, he was concerned with the current limits on the functionality, and because Lonnie does not have a strong technical background, he had some inaccurate opinions on what would be easy or hard to implement.

One of Lonnie’s primary concerns was the limited information currently stored in the database. Our database entries presently only include the deceased’s first and last name, interment number, and date of death. The existing database has many additional fields, such as grave section and grave number within the plot. Lonnie does not need to be able to search these fields, but he does want them to be displayed on the screen. As such, we assured him that it will be easy to add and display additional fields when we begin converting their existing database in CSV format to our database in H2 format, as we anticipate doing during our next sprint.

We hope to show our program to Rob, the director of the synagogue, in the next sprint. He has more of a technical background, so we expect he may have more concrete opinions on what functionality he would like to be implemented. He is also the person who initiated this project, while Lonnie has no real desire for a new program. As such, it would be useful to get his feedback in addition to Lonnie’s.

Defects and Limits on Functionality

One defect we discovered was that each search only returned one entry even if there were multiple valid entries in the database. We initially caught this defect by just manually testing creating new entries and searching for these entries. The expected behavior when searching for a name was that each entry with the specified name would be outputted in the display box in the GUI. The observed behavior was that only one entry with the specified name would be outputted. For example, if we searched for the first name of “Mary”, only one entry would be returned even if there were several entries with the first name of “Mary” in the database. This defect was due to the fact that the SQL exectureQuery command in the ResultSet interface only returns one object, while we believed it returned all of the matching objects. We have since corrected our code to print all of the matching entries.

There is also an issue with running the program through a command line interface. The GUI runs but does not link to the H2 database unless the Java path files are properly configured.  Thus, our program cannot be downloaded from the repository and immediately executed without altering the Java path to include the necessary H2 driver.  This is not currently an issue because we are running the program through the IntelliJ IDE which supports the H2 database, but our ultimate goal for the entire project is to have the entire program encapsulated in an executable that will circumvent the driver problem and be easy to use.

Additionally, we can predict possible future defects based on the real data from the Beth Shalom Cemetery we were given near the end of this sprint. We have not yet started to use the real data, but we have discovered some of its fields have slightly different formats than we anticipated. For example, some of the first name fields include prefixes, as in “Dr. Smith”. Therefore, once we convert the existing database and begin to use it, we may encounter additional defects due to these formatting discrepancies.