Iram Jehan, Tara Le, Sophia Sklar, Ian Tompkins, Tien VoNguyen

COMP 330

Dr. Honig

4 October 2020

Use Case Model

1. Use Case Narrative

|  |  |
| --- | --- |
| Use case: | **Read in initial information from text file (Cycle 1)** |
| Primary Actor: | User |
| Purpose: | To build a genealogy tree with family members and relationships |
| Overview: | The user will input a text file into the system to be read. The system will read in this information to build a genealogy tree with all the listed members, their information, and connections explaining their relationship. The information is stored in a hashmap. |
| Type: | Essential |
| Preconditions: | File must contain member names and their relationships to each other |
| Postconditions: | Data structure created with all member information |
| Special requirements: | Text file must always be in the same format/order of information |

**Flow of Events**

|  |  |
| --- | --- |
| **Actor Action** | **System Response** |
| 1. This use case begins when the user desires to build a genealogy tree. |  |
| 1. The user inputs a text file with the first and last name, birth date and place, and possible death date and place, and relationships. | 1. A data structure is created with the members, their information, and relationships. |

**Alternative Flow of Events**

Line 3: User may want to add members not included in the file

Line 3: May want to search for specific member

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
| Use case: | **Add members to database (Cycle 2)** |
| Actor: | User |
| Purpose: | To add new members to the tree when new relationships are made or children are born |
| Overview: | The user will be able to add new members to the tree. This will include adding their relationship to other member(s) to connect them in the tree. |
| Type: | Essential |
| Preconditions: | In order to add a member, the user must have already created an initial tree from importing a file |
| Postconditions: | New member will be in the tree connected by relationship to current member |
| Special requirements: | User must add relationship(s) in addition to adding member’s name and information |

**Flow of Events**

|  |  |
| --- | --- |
| **Actor Action** | **System Response** |
| 1. This use case begins when the user wants to add members to the tree. |  |
| 1. The user will input new member name and information including relationship(s) to current member(s). | 1. Data structure is updated with the new member’s data and a new leaf is created that connects them. |

**Alternative Flow of Events**

Line 3: User may want to search for members

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
| Use case: | **Search for members (Cycle 2)** |
| Actor: | User |
| Purpose: | To find a member, their information, and their relationship(s) to other member(s) |
| Overview: | The user will type a member name into the search to find their location in the tree. The search will then return the member, their birth and death information (if applicable), and their relationship(s) to other member(s). |
| Type: | Essential |
| Preconditions: | Genealogy tree must already have members and their relationship |
| Postconditions: | The user will see the member that they searched for and the member(s) they are related to |
| Special requirements: | There must be a data structure already developed. Data structure must contain member that the user searches for. |

**Flow of Events**

|  |  |
| --- | --- |
| **Actor Action** | **System Response** |
| 1. This use case begins when the user wants to search for members within the tree. |  |
| 1. The user can search for a member by inputting their name. | 1. The search function will return the searched family member name, information, and those closely related to them in the tree. |

**Alternative Flow of Events**

Line 3: Searched member does not exist in tree. Please check spelling or search for another name.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Use Case Diagram



User