Rushi Samirbhai Patel rs397441@dal.ca
B00886157

Final Project

Genealogy Media Integration System

Milestone 1

Overview:

The project is to develop a Genealogy Media Integration System (GMIS) that stores family information which includes details on an individual and their relations. These details would be linked to an archive of pictures which will assist genealogists to fetch pictures of ancestors or descendants of a particular individual. GMIS can be used to fetch a set of pictures of an individual based on biological relations.

High-Level Analysis:

In this project, the user can easily find relations between two individuals and can also determine ancestors or descendants of an individual. The system should also be flexible to retrieve the nearest common ancestor of two individuals (if any) along with their relation name, based on the degree of cousinship and removal. I believe this would require a good structural design, efficient use of abstract data types and data structures, optimal algorithm, set of tests, and database management clarity. The code that would be developed must be modular since there is mapping of various features and should be achieved in an efficient manner. It is also understood that I would require two tables in the database schema: Individual details table and media details table. These tables would be used with the help of queries to fetch and store data in the code. The data management of GMIS is extremely crucial so the structure developed should complement the flow of data in the entire project code.

I analysed the problem and segregated my analysis based on the below factors which could be useful going forward when I develop the code:

- ★ **Long-term data :**
  - ➔ Family tree database
    - Details of the individual (name, date and location of birth, date and location of death, gender, occupation, references to the source material, notes on the individual).

        - Relations (Parent/Child relations, partnering ceremony relations, partnering dissolutions).
    - Media Archive (filename, date, location, tags, and individuals seen in the media).

- **Abstract Data Type and Data Structures :**
    - Tree as a data structure where the nodes will represent individuals and edges will denote the relations between two individuals.
    - ArrayList stores multiple objects of individuals as a list which could be helpful to retrieve information.
    - Map to store multiple ascendants or descendants of an individual.

      .
- **Structures :**
    - Interfaces to define methods which could then be implemented in their respective implementation classes.
    - PersonIdentify class to store details of an individual in its object.
    - FileIdentifier class to store details of the media file in its object.
    - BiologicalRelation class to determine relations between two individuals.
    - Implementation class where the main system will be implemented.
    - To make the code more modular, I would create separate classes for different feature methods.

The above mentioned factors and details listed might change based on my re analysis in the future during the implementation phase.

<u>Constraints/Difficulties:</u>

While analyzing this project, I made notes on some queries or constraints related to this project. They are listed below:

- Since I have not yet connected with MySql workbench and we just had a lab on it yesterday, I will have to look into this to better use the database in this project.
- Whilst the use of files (to fetch pictures/videos filename) was mentioned in the project outline, I am not able to see any file operations required in the method interfaces mentioned in the project document. Are we expected to do some file operations to fetch filename or any other operations that I might be failing to understand?
- In our schema, we will have two tables one for individuals and the other for media archive data. It is still unclear to me how to map both these tables.

<u>Next Steps:</u>

To develop this system, I believe I would need to follow the below steps:

- Since I have already identified the purpose of this project and have also analyzed bits and pieces of the problem, I will have to plan for development once the constraints and difficulties are more clear to me.
- I will have to list down the abstract data types (probably List and Map) which are the most efficient for this project along with data structures that can be used in this scenario.
- Simultaneously, I will have to work on designing a code that can be correlated with loose coupling and high cohesion.
- Once I have a design ready, I will have to think of some algorithm for the actual implementation of reporting the relations.
- Further, I will have to develop Junit tests to better test my code.
- Finally, code refinements would be necessary to remove code smells.