Requirement Analysis PolSim

Amin Haidar, Noel Monnerat,
Samuel Pearce, Alban Selimi

Table of Contents:

1. Project Description

Our project is a politics simulator using a database, that is flexible enough to simulate many different systems using a simple table structure. We named it PolSim as it's a nice contraction of Politics and Simulator. This simulator lets its participants create groups with each other and share their political power. It is also possible to take over and become a tyrannical dictator. Finally, if everybody has a "government" they're happy with, we can poll the whole system to see how the decision-making process plays out on a global scale. The users get to feel what it's like to live under various ruling systems. From pure, direct democracy all the way to a glorious, absolute monarchy.

The way it works is: To start with, every user is simply a member of "the world". Everybody in the world has the same amount of political power and can create votes. If some people decide to create a "country" they can form a group. Let's say — for example — The UK. These people will start off with a direct democracy. Everybody in the group has equal decision power and if a vote is started, everybody's vote counts the same. Except the group leader who has slightly more power, in order to avoid ties. However, the people of this United Kingdom might decide to create a parliament, and so create a group called "Parliament". Then they make an agreement with the people. If the people all give their political power to the parliament for a fixed term, then the people get to chose who is elected. If the majority agrees, then they all surrender their power to the parliament and any decisions made for all of the country is handled by the parliament alone.

These groups can be nested as deep as the user wishes. You could also have political parties within the groups as their own smaller groups. Inside those parties you could set up factions or conspiracies. This also applies in the opposite direction. If you already have some pseudo-countries set up you could join together and form grand alliances like the united nations.

2. Requirements

2.1. Functional Requirements

- Register/Login
- Create groups
- Join/leave groups

- Change group leader
- Remove someone from group
- Transfer political power
- Organise revolution

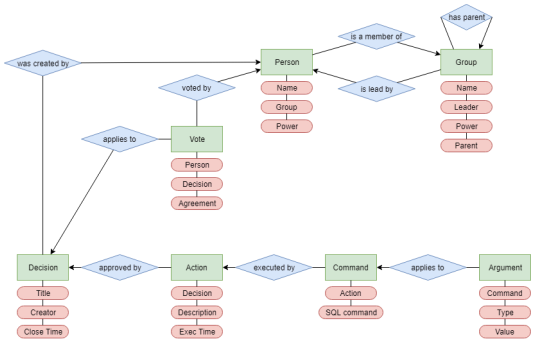
2.2. Non-Functional Requirements

- Database is in the 5th normal form
- All errors are handled correctly

3. Interview

> Insert notes from our interview here

4. Entity Relationship Model



5. Entity Descriptions

5.1. Person

Naturally, we will need an entity to store information about each user. This entity will have the ability to join groups and vote on decisions in that group. Each person also requires a username that is unique.

Attributes:

- Name: unique title for referencing users
- Group: this is the group the person is in
- Can Vote: a boolean value whether this person has the right to vote
- Power: the amount of weight this one person's vote has in a decision

5.2. Group

This entity stores all the information of the hierarchical structure of the different governments. This entity is also self-referential, so that the groups can be nested inside themselves.

Attributes:

- Name: unique title for referencing groups
- Leader: person who created this group and also who breaks ties in voting
- Parent Group: the group this group is in
- Power: the amount of weight this one group's vote has in a decision

5.3. Vote

This records what every person votes on the decisions. It references the decision and the person who cast the vote, as well as whether the person agrees or disagrees to approving the decision.

Attributes:

- Person: the person casting this particular vote
- Decision: the decision this person is deciding on
- Agreement: whether the person agrees to approve this decision.

5.4. Decision

Every decision that needs to be made, and has already been made, will be stored in this table. The outcome of this vote is gotten, by counting the number of unique and valid votes cast in favour of the decision.

Attributes:

- Title: a short description of the vote topic
- Creator: the person who initiated the decision
- Close Time: the time when voting is closed and the decision is either accepted or rejected. -

5.5. Action:

When a decision is approved it can perform any number of actions. An action is a group of SQL statements that get performed on the database. For example, if you want to change the leader of a group, that would be put in the database.

Attributes:

- Decision: the decision that will approve or deny that this action can pass
- Description: a short explaination of what this action will perform upon being approved.
- Exec Time: the time when this action is performed if it has been approved

5.6. Command:

Actions constitute groups of commands to be executed. You can't just create a group, you must also add all the potential members to the group as well. All those commands are performed in a single action, simply called "Create group".

Attributes:

- Action: the action that would perform this command
- SQL Command: the SQL code to be executed with this command with its arguments substituted

5.7. Argument:

Commands can have varying numbers and types of arguments, so this table contains the type and value of all the arguments to be passed to an SQL command.

Attributes:

- Command: the command that accepts this argument

- Type: the datatype of this argument
- Value: the value to be passed as this argument

6. DBMS Decision

Insert DBMS Decision matrix here

7. Specifications

7.1. Decision method

Decisions are approved or denied by a majority vote. Votes are either on the local, global or universal level. Local means all people and groups within the current group. Global means all the people and groups within the group one layer above the current level. Universal means all the people and groups in "the world". "The World" is the highest level group that contains all people and groups in the current simulation. Each group and person has a certain amount of political power on their level. Decisions are scaled based on the voters political power. Take for example this system:

```
The World (100%)

    Switzerland (50%)
        Bill  (25%)
        David (25%)
        James (25%)
        Bob   (25%)

    United Kingdom (50%)
        Parliament   (100%)
            Boris      (33%)
            Nigel      (33%)
            Jeremy     (33%)
        Jim      (0%)
        Jeremiah (0%)
```

The political power of each person and group is written in parentheses "()". Jim and Jeremiah’s votes won’t be counted at all, as they aren’t in the parliament.

Boris, Nigel and Jeremy’s votes each count for a third of the result. Bill, David, James and bob live in a direct democracy and each have one quarter of the decision power of switzerland. Both countries hold half of all power in the world.