Design Decisions – Milestone 2

Written by: Monica Ruttle, Reid Cain-Mondoux and Justin Fleming

This document provides detail on design decisions made on the project SocialSim and reason each decision was made. The project is planning ahead by following java beans convention so we can use the XML encoder and decoder for saving. The project is currently separated into 4 main sections:

* Simulation
* Document
* Consumer/Producer (Users)
* Searching
* Main Window

# Consumer/Producer (User)

Since Milestone 1, we have made significant changes to the structure of Consumers and Producers. There is now an abstract User class which stores the followers and who they are following, the user payoff, the tag, the Id, their preferred search method and which documents they like. It also contains shared methods to follow users and like documents. The Consumer and the Producer class inherit from User, and implement how they take their turn and how they calculate their payoff. This allows for the possibility of the Consumer to have different behaviours from the Producer, as before the Producer simply did exactly what the Consumer did and more.

# Document

The Document class represent a document. It includes data to be used as metrics for the simulation. The Document is a representation of real documents but only include relevant simulation data. It stores a list of who likes that document, and a method to add to this list.

# Simulation

The simulation currently keeps track of all documents, consumers and producers. Since consumer and producers inherit from User, they are contained in a single User list which makes selecting whose turn it is independent of User type. This was modified from Milestone 1, as before it contained a list of Consumers. It is the central hub of the emulation software. It currently keeps track of all documents and the simulation state. The user interaction is handled by a Main Window, which gets updated by the Simulation. The simulation handles starting the game and controlling whose turn it is. It also updates the users each time a turn is taken and passing that information to the Main Window.

# Searching

Searching interface is using the strategy design paradigm to use different methods of selecting documents to discover what searching method is the most beneficial to the consumer and the producer. The consumer and the producer store which search method they use, as it may differ per user.

# Main Window