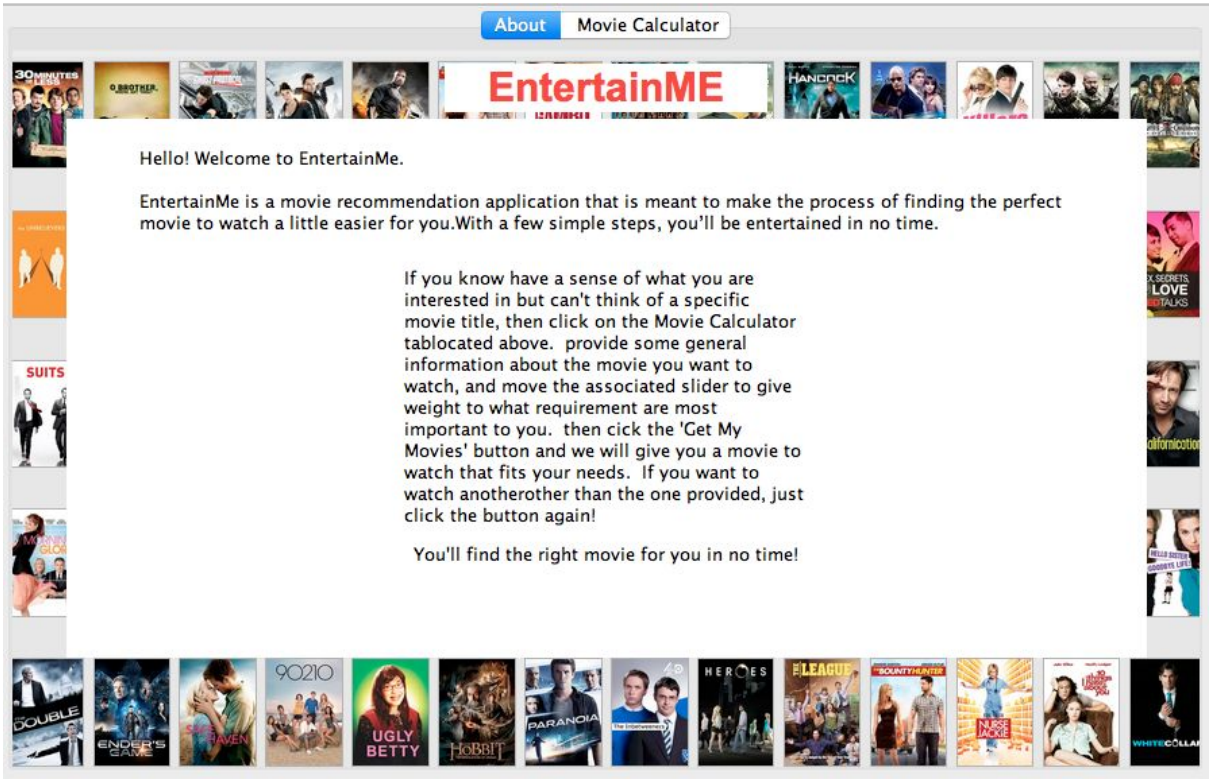
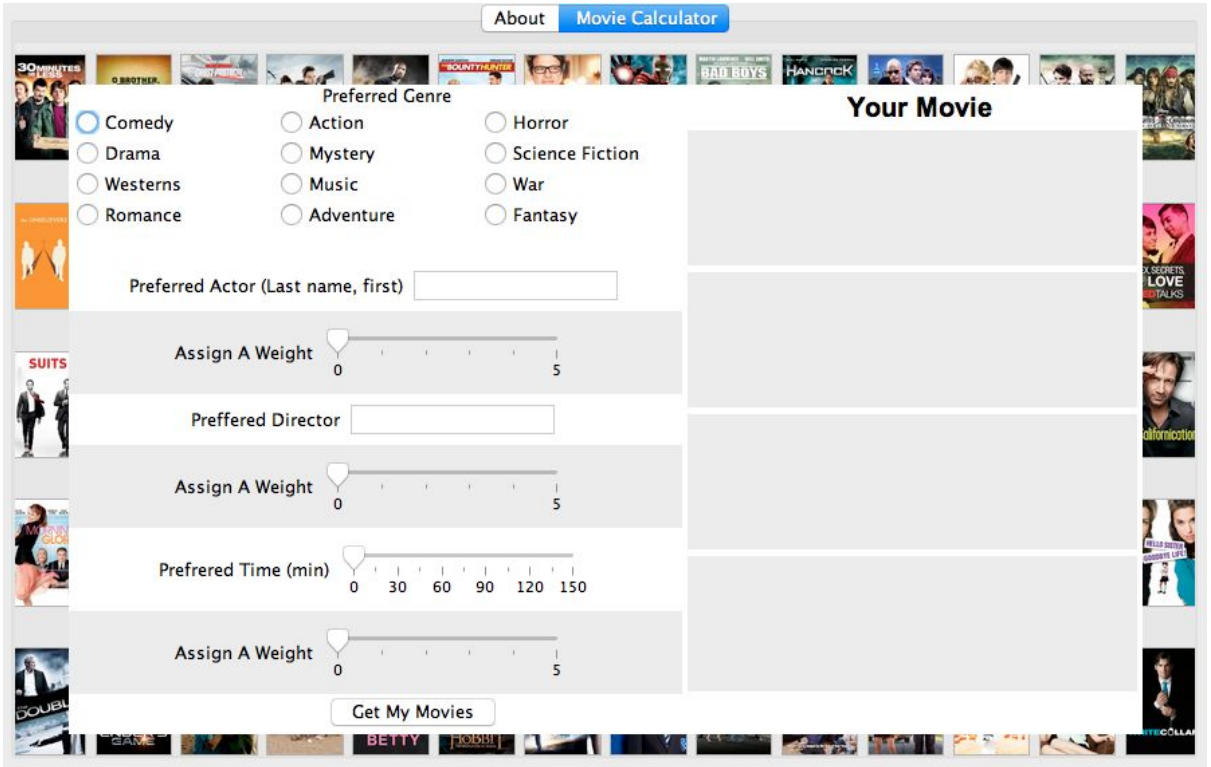
EntertainME

Sharon Kim and Eva Stern-Rodriguez





// **Part I.** User Manual

Our program organizes movies into categories and suggests movies to you. Its function is to provide entertainment for those who are unsure of what movie to watch.

Our GUI is split into two tabs: 1) About Panel and 2) Movie Calculator. The window will open up to the About Panel, which details how exactly the user should proceed, including how to input information in the Movie Calculator in order to get the perfect recommendation.

Instructions: Movie Calculator is made with an easy and intuitive GUI so you can easily input information and get the movie that is right for you. To start, click on the Movie Calculator tab located above. The only information you are required to provide is the genre. Click on one of categories that fits you best. Then, you can provide some more general information about the movie you want by inputting the Name of an actor or a director you like into the text box. Move the slider in order give weight or importance to the category directly above. The Preferred Time slider allows your preferred maximum time you would like to spend watching a movie. When you have filled out enough information click the 'Get My Movies' button and we will give you a movie to watch that fits your needs. If you are unhappy with our selection or would prefer to watch another movie other than the one provided, just click the button again!

The Movie Calculator tab contains radio buttons for the desired genre, a textbox to input an actor’s name (in the format [last name, first name]) and a slider to assign weight to the importance of the actor, a text box and weight slider for a director, a slider for choosing a time limit and a slider to give weight to the time. Once all necessary information is inputted, press the “Get My Movies” button and your movie recommendation will appear on the right! Click the button again to generate a new movie.

// **Part II.** Technical Report

Part a:

> Hashtable<String, Vector<Movie>> allMovies: all movies and their corresponding instance data will be accessed via a Scanner parsing through a static .csv file. Movie objects are instantiated and initially stored in this Hashtable, organized

by genre, the key, associated with a Vector of Movie objects that are considered part of this genre.

Vector<Movie> moviesOfGenre: a Vector of Movie objects, all of which are part of one genre. At this point, the Movie objects are not yet ordered by rank.

ArrayStack<Movie> myRankedStack: an ArrayStack of Movie objects, ordered by rank that has been computed by the computeRank method, which depends on the user's preferences/weights. The weights are determined by the sliders that indicate how strongly they value their input for length, actor, and director. A new movie is popped off the stack every time the user requests a new movie so that the same movie will never be repeated. When the ArrayStack runs out of Movies, a pop-up box displays

Part b:

Movie.java: Contains instance variables associated with a Movie object, including title, genre, length, year, actors and actresses, director, popularity, and rank. Implements the Comparable interface for later sorting purposes.

MovieLibrary.java: Parses through the .csv file and creates all instances of Movie objects and adds them to the corresponding genre key and Vector value in a Hashtable.

MyMovieCollection.java: Given a genre, creates an ArrayStack of ordered Movie objects by rank, according to user weights/preferences.

AboutPanel.java: Tab panel that gives users instructions on how to use the program.

MovieCalcPanel.java: Tab panel where users directly choose their genre and time preference, director, etc with a slider and text inputs to weigh the importance of their choices. These are used to compute the rating for movies. This class pulls up the movie recommendation after the user submits the necessary information fields.

EntertainMEGUI.java: creates a tabbed pane and adds all aforementioned panels to the frame. This is the file to run.

Part c:

compareTo(Movie other): in the Movie.java class, implements the Comparable interface so that Movie objects can later be sorted by rank value.
computeRank(): in the Movie.java class, additively computes the ranking of each Movie object given user weights/preferences.

MovieLibrary(String inFile): in the MovieLibrary.java class, this constructor parses through the static .csv file using Scanner, instantiates Movie objects, and adds them to a Hashtable under the appropriate genre key.

prioritize(): in the MyMovieCollection.java class, calls the mergeSort algorithm implemented by ADTs that extend the Collection interface (in our case, Vector). After sorting this Vector of Movie objects, these Movies are passed into an ArrayStack so that Movie objects can easily be popped off as each new Movie is generated in the GUI (no newly generated movie is repeated).

Button listeners: Attach event listeners all genre radio button and to the “Get My Movies” button. The radio buttons create a new MyMovieCollection every time a new genre is chosen and the “Get My Movies” button organizes the movies into a stack and pops movies off to display them to the user.