# Project Summary

A movie rental place has one copy of each recently released movie available for customers to rent. Given the customer’s preferences, this model aims to assess whether or not it is possible for every customer to rent a movie they will enjoy.

# Propositions

-Customers (Ci,j)  
How many customers? All other propositions are created based off how many customers exist, as well as what those customers want. Only one person can rent a movie at a time (Customer I rents movie j)

The following propositions represent aspects about a movie. They are considered true if the customer has indicated a preference for them and false if they have not.  
  
-Genre (G):  
Choose genre Action, Adventure, Animation, Biography, Comedy, Crime, Drama, Fantasy, History, Horror, Mystery, Romance, Sci-Fi, Thriller, Western  
Set as choose one genre for now.  
  
-Quality (I):  
Choose a quality 1-3 or no preference, what number you choose is representative of a group of IMDB scores, with 3 beings the best. 1 represents IMDB score 0 – 3.9, 2 represents 4 – 6.9, 3 represents 7 – 10.  
  
-Runtime (R):  
Choose a length for your movie choice, with the two choices being over or under 2 hours  
  
-Popularity (P):

Choose how popular you want your movie to be, based off box office results. The options are niche, average popularity, popular, or no preference.  
  
-Age:   
Choose what decade you want your movie to be from

# Constraints

*-each customer can only choose 1 movie genre (for now at least)  
-each customer can only select one option for all (other than genre) options  
-no two customers can have the same movie recommended (even if all their preferences are the same)   
-For a movie selection for each person to be true, the customer preferences for genre, quality, runtime, popularity and age to be true has to align with the movie’s genre, quality, runtime, popularity and age  
-For the model to be complete every person needs to have a movie that satisfied their constraints.  
-*

# Model Exploration

*Exploring the model, we had to go in the following order:*

1. *Model implementation :*
2. *Ensure that all the propositions and constraints perform as expected.*

* *At present, the goal is to get the model to recommend movies to one user. This is not fully functional at the moment, but we plan to improve it in the future and add the option for multiple customers to rent movies.*

1. *Figure out how to connect the implemented proposition with the csv file.*

* *To be able to access the data set of movies, we used the python implementation that reads through the file, and converted the data set into rows and columns.*
* *This method would make it more efficient to look for looking up a value in a column, and then find all its info in the row.*

1. *Get inputs from the user, so it can create a proposition on which is the perfect fit to their preferences.*

* *The user is prompted by getting questions on their preferences.*
* *Their preferences are implemented into the program and is looked through the csv file.*
* *The propositions will set movies to either true or fake depending on their preferences.*

1. *Implement constraint that will ensure the customer would not get a movie that they are not looking for.*

* *By implementing the constraints in the constraint section.*

1. *Run the program through docker.*

* *By following through the steps provided on onq.*

1. *The failures the team encountered:*
2. *Running through infinite loops.*
3. *Finding constraints.*
4. *Making functions instead of class in the code.*
5. *Using the wrong loops to loop through the data set.*

# Jape Proof Ideas Graphical user interface, application Description automatically generated

*List4 the ideas you must build sequence & proofs that relate to your project.*

# Requested Feedback

1. *How can we enhance our code to perform as well as expected, in terms of proposition and constraints implementation ?*
2. *What constraints and propositions could be added to the model?*
3. *If you were the developer of the model, what different approaches would you take, in terms of proposition and constraints and their implementation.*

# First-Order Extension

*There are X customers, each customer needs a movie, therefore there will be X movies given out.  
for each customer,  
customer needs to select a genre, The options for genre are* Action, Adventure, Animation, Biography, Comedy, Crime, Drama, Fantasy, History, Horror, Mystery, Romance, Sci-Fi, Thriller, Western, therefore the customer will select one of the given genres  
  
The customer needs to select a quality, the options for quality are 1,2,3,4 or no pref, the customer will select a quality   
  
The customer needs to select a runtime, the runtime options are 1 (under 120m) , 2 (over 120) and no pref, the customer will select a runtime option  
  
The customer needs to select a popularity rating, the popularity ratings are 1-3 (3 being the most) and no pref, the customer will select what popularity rating (1-3) or No Pref  
  
The customer needs to select a age for the movie, the age options are in decades, the customer will select what decade they want the movie to be from *the customer(s) has selected a given option for every category and needs a movie back that has the same as what they picked, there exists a movie for every case scenario, therefore the customer will receive back a movie that meets their given criteria.*