**Sprint 0 Documents**

**Things We Know / Are Needed: (For Sprint 1, Program):**

* Random Number Generator
  + With probabilities based on averages/distribution
  + <https://stackoverflow.com/questions/20327958/random-number-with-probabilities>
* Adjacency Matrix
  + Each array holds the connections for lift, x
  + Lifts use a global 2D matrix to calculate waiting time
    - matrix has time in the day and mountain level (higher = more difficult)
* Command Line Parser
  + Validator
* CSV Writer
  + Formatting
* Skier, Resort, Lift, Ride classes
  + To store relevant data
* **Testing:**
  + Unit testing
    - Ski Generator
    - Skier, LiftRide
    - SkiResort
    - PathGenerator
      * Verifying algorithm creates *accurate* paths for each skier
  + CSV data generator:
    - **Create a program** to read in output CSV and analyze
      * Input the csv file into and test a chunk of the skiers
      * Make sure skiers are following their attributes/habits
      * Analyzes overall output to verify distributions and intuitions
        + Lunchtime skiers decrease by 50%
        + Ski Lifts maxed out from 9:30am-10:30am
        + Starting distribution 30%/20%/50% from starting base
        + Number of runs falls within 3 to 25 range
* User Stories Info:
  + From perspective of client/customer: Ski resort CTO / Analyst
  + What they want:
    - CSV file where each line is a lift ride
    - Each lift ride has:
      * Time start
      * Time finish
      * Riders
      * Location start
      * Location end
    - Effective/Easy User Interface
* Program Inputs:
  + Day of the week:
  + Given: average number of skiers in given day
  + Time in the season:
    - Given: late/early season = 20% fewer skiers
    - Given: peak season = 10% more skiers
  + Weather of Day;
    - Given: “good” weather = 25% more skiers
    - Given: “stormy” day = 10% fewer skiers **and** upper lifts stations can close
* Skier “Habits”/Resort Info:
  + Skiers Path: “lap the same lift a few times before moving on” - AKA lazy rider
  + Base Area:
    - 20%, 50% and 30% load from Blackcomb, Whistler and Creekside respectively
  + Time Skiing:
    - resort opens = 9am, closes = 3pm
  + Base Traffic:
    - average/stormy days = normal distribution, peak load time = 9:30am
    - good day = full capacity for 60 min
    - Lunch (12-1pm) = 50% decrease in skiers
  + Number of runs by Skier in a Day = 12:
    - Distribution: 3 = min, 25 = max
  + Trails and Lift Status:
    - https://www.whistlerblackcomb.com/the-mountain/mountain-conditions/terrain-and-lift-status.aspx

**User Stories:**

1. As a ski resort, I want synthetic data of ski lift rides so that I can pass skier traffic predictions to my operations, financial, and marketing departments.
2. As a CTO, I want synthetic data of ski lift rides so give my team can build a skier prediction and tracking system.
3. As a resort operations analyst, I want synthetic data of ski lift rides to prepare my lift operators and mountain staff appropriately.
4. As a resort marketing analyst, I want synthetic data of ski lift rides as a way to find out which paths/areas are most appealing to customers.
5. As a resort marketing analyst, I want synthetic data of ski lift rides as a way to find out which times customers are in certain locations to effectively market to them.
6. As a resort financial analyst, I want synthetic data of ski lift rides as a way to measure customer purchases and predict future revenue.
7. As a member of the tech team I want organized synthetic data to test our future wait time apps against.

**Sprint 0 Trello Board:**

* URL: <https://trello.com/b/PCjV7j6T/msd-project>

**Future Trello Board Things-To-Do:**

* Implement adjacency matrix for mountain
* Implement command line parser/input validators
* Implement CSV data format mechanism (file writer class)
* Implement skier and skier path generator
* Create Java classes for skiers, ski lifts, and the ski resort

**Use Case and UML Diagrams:**



