Data Model Outlines:

User

* userId - type: primitive
* userName – type: string – Name of user for display / login
* foodAwards – type: object – Includes the name and count of food awarded to user
  + {foodName}: foodCount
* dailySelection – type: object – Includes:
  + eventId – type: primitive - for the user selection for the particular day
  + ruleId – type: primitive – identifies the id of the rule for the user selection for the particular day
* setDailySelection – type: function – Sets the Daily Selection for the User

Team

* teamId – type: primitive – Correlates to team\_id from xmlstats Teams API
* teamName – type: string - Name of Team associated with Object
* teamLogo – type: string – Address of Logo for Team (for disp. in View)
* foodRule({statsObject}) – type: function – Determines whether food is awarded given box score stats
* ruleDescriptions – type: array – Array of Rule Descriptions indexed by ruleId's
* foodName – type: string – Name of food that team awards (e.g. 'McMuffin')
* foodLogo – type: string – Address of image associated with food

Event

* eventId – type: primitive – event\_id from xmlstats Events API
* homeTeamId – type: primitive - team from xmlstats Events API
* awayTeamId – type: primitive - opponent from xmlstats Events API
* eventStatus – type: primitive - Correlates to
  + (-1 before game complete)
* getStatsObject() - type: function – Executes when eventStatus != -1, populates statsObject w/ data from xmlstats Box Score API
* statsObject – type: object - includes stats from xmlstats Box Score API

Schedule is a collection of Events

Views:

Login View (only displayed if not logged in)

Game Selection View

User Record View

Functions to Implement:

homeTeamByEventLocation(location)

* Parameters: location
* Returns: Home