Spring #1 Backlog

**Project 2 builds on and adds more sophisticated features to project 1. All requirements for project 1 are included in and hold for project 2 unless they directly contradict with the features requested for project 2.

1. **Web application must be secure and protect confidentiality of a user's ImHungry data**
   a. Create a MySQL table with a following structure
      i. Users -> username, string, password , SHA256 hash data type, id, primary key
   b. Add a sign-in page to the Web Application, SignInView.jsp. SignInView will contain a single text input for the username and a single secure text input for the password. It will redirect to SignInController if both are not empty. SignInView contains a link to sign up if not a user.
   c. SignInContrtoller will validate the two to make sure username / password is valid combination in database (username exists, password matches) using DatabaseModel. If validated, redirect to SearchPageView. If not validated, redirect back to SignInView.
   d. On ResultsPageView.jsp, add a sign-out link that redirects to SignInView.jsp.
   e. DatabaseModel will directly access our mysql database and do the following
      i. Find user by username, validate password, return true or false depending on if valid sign in
      ii. Check if a user exists in DB by username only
      iii. Insert new user into database if one does not exist
   f. Add a sign-up page to the web application. SignUpView.jsp. SignUpView will contain 3 text boxes centered on the page. 1. Enter username. 2. Enter password, secure. 3. Enter password again.
   g. Create a SignUpController that will call DatabaseModel to insert the user after validating the parameters are not empty. SignUpView contains a link to SignInView if already a user.

2. **Allow for pagination of results returned by the search (Alex)**
   a. https://mdbootstrap.com/docs/jquery/tables/pagination/
      i. Above link shows demo of dynamic pagination (works without reloading the page b/c it uses javascript)
      ii. Note this component uses a framework called Material Bootstrap
   b. For results object taken in from the Servlet

3. **User interfaces must look modern and be attractive (Kayla)**
   a. Make the background a dark gray and the text a white so it looks like night/dark mode
   b. Comic sans font

  c. Add a table hover so the blocks representing each restaurant/recipe look more clickable and are more interactive

4. <mark>Set the radius of the restaurants search</mark> **(Cindy- backend, Kayla- frontend, Lazlo-testing)**

  a. Next to the search limit, put a text box for radius. Radius should be number > 0. Task A: Modify the JSP SearchPageView to have the text box specified above.

  b. Link into SearchPageController (Edit SearchPageController to Redirect with the radius from text box, and give default of 1 mile), ResultsPageController (Give ResponseModel checkParameters a radius input), ResponseModel, YelpRequestModel (Give YelpRequestModel a member var private double and checkParameters an additional input double for radius).