Wiki

This Wiki explains the uses of the functions that can be used for the GUI, allowing for the GUI programmer to quickly find what function they need and how it can be used in order to speed up development

CircularImageFrame

OVERVIEW

This file contains code related to changing the shape of a given image to circular

---

loadImage

OVERVIEW:

This function is used for loading in a user selected **image** and storing it in the given **path**

TAKES :

1. (**String** userId, **BufferedImage** image)

   This is used when using JFrame to get the image via directory

2. (**String** imagePath)

   This is used when the image is already in the users profile directory

RETURNS :

1. NULL
2. NULL

Constants

OVERVIEW:

This file contains variables which are used in the project to enable values to be changed quickly

---

Variables

1. **Int** numberOfRecipesReturned

   this is used in the diet and image search function to define how many recipes the program should return

LogIn

OVERVIEW:

This class contains code relating to the login flow

getUserProfiles

OVERVIEW:

Used for getting the users names, images and ids to populate the login screen

TAKES :

RETURNS :

1. List<DataStructures.StringImageIdPair>

   This is a custom data class which contains **String** username, **String** userId, **ImageIcon** image . Variables can be retrieved via **getText(), getId(), getImage()**

getNumberOfUserProfiles

OVERVIEW:

Used for getting the number of users in the system

TAKES :

RETURNS :

1. Int numberOfProfiles

   This is the number of profiles

deleteUserFolder

OVERVIEW:

Used for deleting a specific user folder

TAKES :

1. **String** userId

RETURNS :

Tools

OVERVIEW:

Contains general functions that can be used throughout the program

getRecipeImage

OVERVIEW:

Used for getting the image for a recipe

TAKES:

1. (**String** imageName)

    This image name can be gotten by using recipe.generateImageName();

RETURNS:

1. **ImageIcon**

getDietImage

OVERVIEW:

Used for getting the image for a diet

TAKES:

1. (**String** dietName)

    This name can be gotten by using diet.getName();

RETURNS:

1. **ImageIcon**

linkTagAndDietIcons

OVERVIEW;

Used for when displaying a recipe (**DO NOT USE when displaying the partial information in the recipe search function only when showing the whole recipe page**): pairs the diets and tags with their related icons

TAKES:

1. (**Recipe** recipe)

    This is the recipe that you want to display

RETURNS:

1.**List<DataStructures.StringImagePair>**

    This is a custom data class which contains **String** text (the diet/tag), **ImageIcon** image . Variables can be retrieved via **getText(), getImage()**