### Introduction

### In this project, we extended the router to support the ability to save and retrieve a list of favorite dishes by each of the registered users. All registered users in the system should have the ability to save any dish as their favorite dish, retrieve all their favorite dishes and remove one or all their favorite dishes.

### *Salient Features:*

### will be supporting a new route <https://localhost:3443/favorites,> where the users can do a GET to retrieve all their favorite dishes, a POST to add a list of dishes to their favorites, and a DELETE to delete the list of their favorites. In addition, will support the new route <https://localhost:3443/favorites/:dishId> where the users can issue a POST request to add the dish to their list of favorite dishes, and a DELETE request to delete the specific dish from the list of their favorite dishes.

### Design and Implementation

### implementing a new Mongoose schema named favoriteSchema, and a model named Favorites in the file named favorite.js in the models folder. This schema should take advantage of the mongoose population support to populate the information about the user and the list of dishes when the user does a GET operation.

* When the user does a GET operation on '/favorites', you will populate the user information and the dishes information before returning the favorites to the user.
* When the user does a POST operation on '/favorites' by including [{"\_id":"dish ObjectId"}, . . ., {"\_id":"dish ObjectId"}] in the body of the message, you will (a) create a favorite document if such a document corresponding to this user does not already exist in the system, (b) add the dishes specified in the body of the message to the list of favorite dishes for the user, if the dishes do not already exists in the list of favorites.
* When the user performs a DELETE operation on '/favorites', you will delete the list of favorites corresponding to the user, by deleting the favorite document corresponding to this user from the collection.
* When the user performs a POST operation on '/favorites/:dishId', then you will add the specified dish to the list of the user's list of favorite dishes, if the dish is not already in the list of favorite dishes.
* When the user performs a DELETE operation on '/favorites/:dishId', then you will remove the specified dish from the list of the user's list of favorite dishes.

**MODULES:**

* *express*
* *path*
* *serve-favicon*
* *morgan*
* *cookie-parser*
* *body-parser*
* *express-session*
* *session-file-store;*
* *passport*

### Conclusions

### Allowed users to select a dish as their favorite, and add it to the list of favorites that are saved on the server.

* Allowed users to retrieve the list of their favorite dishes from the server
* Delete one or all of their favorite dishes from their favorites list on the server.