

A.

- Project title: CAU-IKH
- List of team members:
20182705 고주형, 20185784 김호성, 20182610 손희승, 20162874 이준협, 20142611 이하람
- Presentation speaker name: 고주형
- Brief project description (summary)
Efficient kitchen helper that is used for managing a lot of recipes, and planning daily meals like breakfast, lunch or dinner.
By using our kitchen helper, you can SAVE/DELETE/SEARCH for recipes and manage your meal plans.
We also put ascii arts inside our program. Pretty UI makes program livelier and fresher. 😊

B.

- How to compile and execute
 - ① Change directory to Source Code Package Folder
 - ② Open the Solution File with Visual Studio 2019
 - ③ Compile (Ctrl-F5) and Execute
- And also, an executable file is in Release Folder so you can directly execute our project file. (How to use executable file is in README.txt)
- System requirement for compilation and execution
Target OS: Windows 7 / 10
System Recommendation: Same as Visual Studio 2019 system requirement

C. Description on functionality that was implemented in your SW system:

- Add recipe: Add a new recipe to our Database.
- Delete recipe: Delete existing recipe from our Database.
- Search recipe (Print all recipes):
Search existing recipe from our Database by recipe name. It is automatically printed in alphabetical order.
Also, The containing word(Keyword) Search is supported(If you search "pie", all kinds of recipes whose name includes substring "pie" will be searched. For Example, "apple pie", "raspberry pie" ...)
- Add meal plan: Add new meal plan to our Database.
- Print meal plan: Print all meal plans' title and breakfast, lunch, dinner's menu. It is automatically printed in alphabetical order.
- Delete meal plan: Delete existing meal plan from our Database.
- Pretty Ascii art: Maybe it will make user happy and make program fancier.

D. How you implemented (important implementation issues):

- At first, we tried to identify the project given to us before designing. What we were trying to create was IIKH, which needed to create and read a database of recipes and plans. Therefore, we focused on the database when designing the program. Recipes and plans have a database that contains recipe data or plan data. We needed to save the database as a file to load and save, and we chose the file format of the database to be csv. We chose csv format because the delimiter of csv was ','. We thought it would be easy to convert.

After that, the story came up to here, there were several things we had to do. One is to create the ability to read and write data using the csv file format, the other is to build the internal structure of the data and the database, and the other is to use the data to retrieve, display, or add data within the program. Based on this, we completed the division of roles, and after some communication, integrated the functions and finally completed the program.

➤ Important Issues

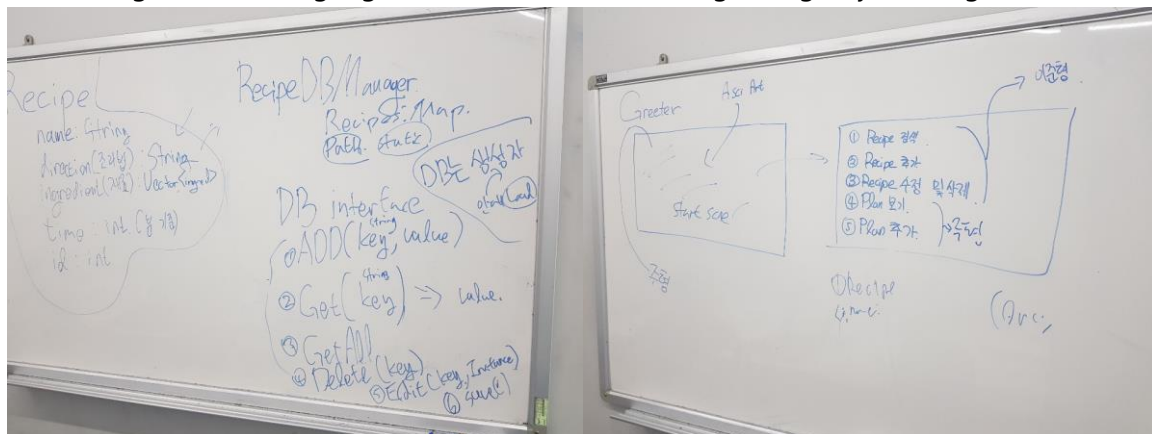
● Issue01: Problem was ambiguous.

Due to ambiguous specification, we had to talk a lot about what this program, what is our target, what should we implement and how we'll implement this system. Though we had clarified our objective (what we'll implement). The problem was that the overall program design which is about how we'll implement this program, was chosen by just talking. There was no document or logs about what we talked. Also, after some coding we found out everyone was thinking slightly different design.

By that problem when we are talking about our program, we had many issues such as, Team Member_A thought it is better to make Database Manager for all objects (recipeDBManager, planDBManager, mealDBManager). But other Team Member_B thought it is better to make integrated Database Manager that can handle all of the objects. By this mismatched when someone coded a Database header, few team members could understand it.

Also, the Plan class was confusing. It was not described correctly and specification was made by just talking. For instance, our one team member thought date should be addressed in plan but other team member thought date will be addressed in meal class. Such kinds of situation led our design more complicated and finally became not understandable. So, we had an emergency meeting and unified all of our design. After that we could start programming our given part for each peacefully and confidently.

Below images are meeting logs taken when we were doing Emergency Meeting.

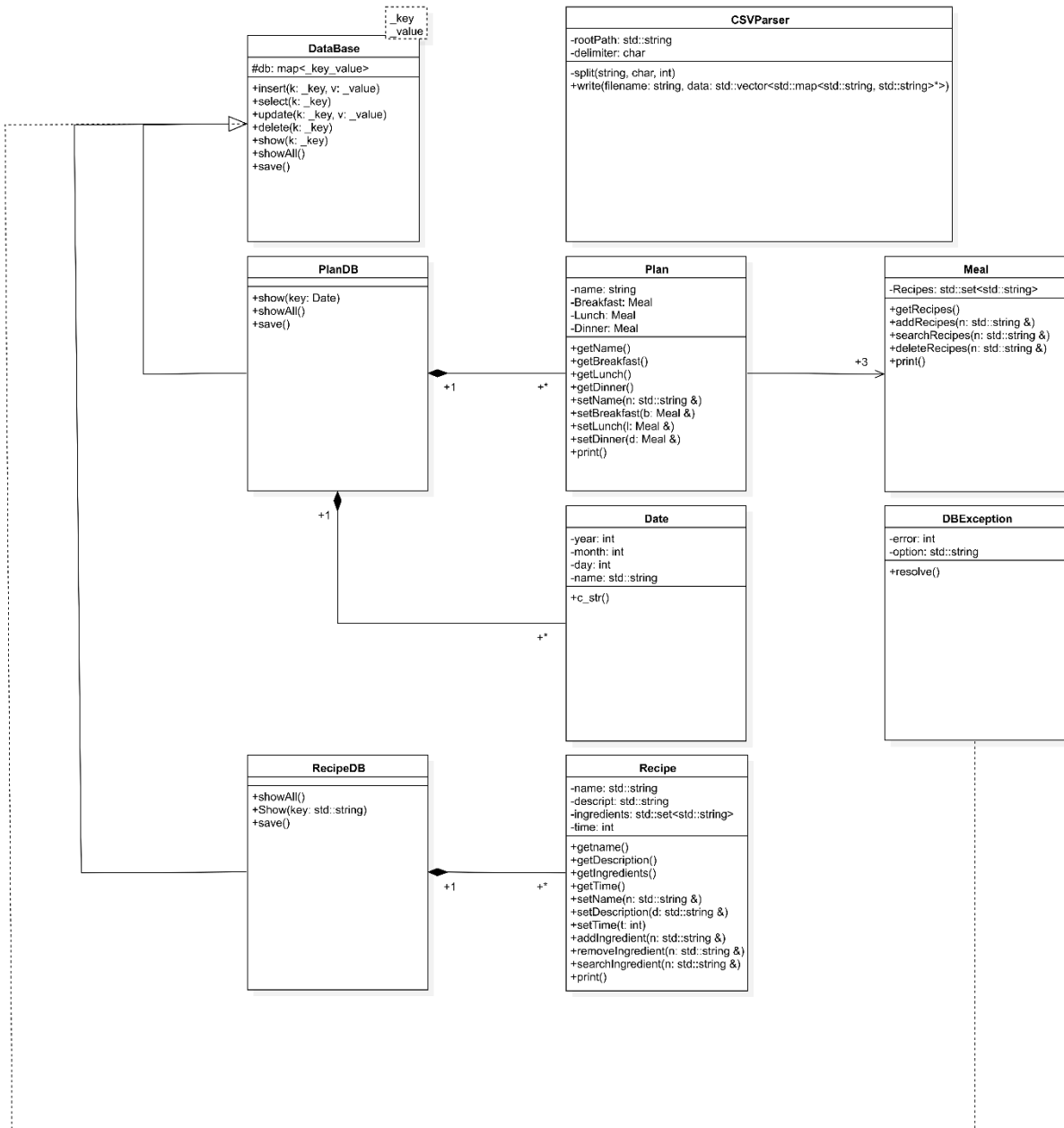


● Issue02: Designing component and giving responsibilities to specific class.

We argued a lot about responsibilities. If a specific class controls every class, the dependency becomes high. This was bad. Bad to debug or understand or extend as we learned.

For example, meal information was needed in everywhere. A Planner had to know meals for daily meal plan, Database Manager should contain every meal for saving, and meal should contain date for searching meal plan by date. After some coding meal class became too huge, and dependent. So, we lowered the dependency of class by giving responsibility to each class and requesting to other class when other class' responsibility is need.

E. The result of SW system design [UML]:



(use screen capture)

Show that each function of the SW system is working correctly.

1. Start Scene

[illegible]

2. Adding Thigh Burger Recipe (Menu 2)

```

You've selected [2] to add recipe
Enter recipe information in [[ NAME -> DESCRIPTION -> TIME -> INGREDIENTS ]] order
Name                                     : 싸이버거
Direction                             : 1. 계란 2개를 쪄 후 허벅지 살을 당군다. 2. 허벅지 살에 튀김을 입힌다. 3. 튀긴다. 4. 빵 사이에 마요네즈와 함께 넣는다.
Cooking Time(min)                     : 20
Ingredients (If you want to stop enter "stop") : 뼈 없는 닭 허벅지 살
양상추
마요네즈
빵
튀김
소스
마늘
양파
파
네즈
양파
가루
stop

```

3. Searching Thigh Burger Recipe that I just added. (Menu 1)

```
You've selected [1] to search recipe
Search : 싸이
There is 1 matching result for "싸이"
=====
>> Recipe Name : 싸이버거
>> Ingredients :
- 달걀
- 마요네즈
- 밀가루
- 뼈 없는 닭 허벅지 살
- 소시지
- 양파
- 양파가루
- 전분가루
- 햄버거 빵
- 후추
>> Cooking Time(min) : 20
>> Direction(How To Cook)
- 1. 계란 2개를 푼 후 허벅지 살을 담군다. 2. 허벅지 살에 튀김을 입힌다. 3. 튀긴다. 4. 빵 사이에 마요네즈와 함께 넣는다.
=====
계속하려면 아무 키나 누르십시오 . . .
```

- You can see all Recipes containing searched word were being printed.

```
You've selected [1] to search recipe
Search : Chicken
There is 5 matching result for "Chicken"
=====
>> Recipe Name : Aussie Chicken
>> Ingredients :
- 1 cup sliced fresh mushrooms
- 1 tablespoon dried onion flakes
- 1 tablespoon vegetable oil
- 2 cups shredded Colby-Monterey Jack cheese
- 2 tablespoons chopped fresh parsley
- 2 teaspoons seasoning salt
- 4 skinless boneless chicken breast halves - pounded to 1/2 inch thickness
- 6 slices bacon cut in half
- 1/2 cup honey
- 1/2 cup prepared yellow mustard
- 1/4 cup light corn syrup
- 1/4 cup mayonnaise
>> Cooking Time(min) : 25
>> Direction(How To Cook)
- Step 1: Rub the chicken breasts with the seasoning salt cover and refrigerate for 30 minutes. Step 2: Preheat oven to 350 degrees F (175 degrees C). Place bacon in a large deep skillet. Cook over medium high heat until crisp. Set aside. Step 3: In a medium bowl combine the mustard honey corn syrup mayonnaise and dried onion flakes. Remove half of sauce cover and refrigerate to serve later. Step 4: Heat oil in a large skillet over medium heat. Place the breasts in the skillet and saute for 3 to 5 minutes per side or until browned. Remove from skillet and place the breasts into a 9x13 inch baking dish. Apply the honey mustard sauce to each breast then layer each breast with mushrooms and bacon. Sprinkle top with shredded cheese. Step 5: Bake in preheated oven for 15 minutes or until cheese is melted and chicken juices run clear. Garnish with parsley and serve with the reserved honey mustard sauce.
=====
>> Recipe Name : Chicken Enchiladas II
>> Ingredients :
- 1 (10.75 ounce) can condensed cream of mushroom soup
- 1 (4 ounce) can diced green chiles
- 1 cup shredded Cheddar cheese
- 1 tablespoon butter
- 1/2 cup chopped green onions
- 1/2 cup sour cream
- 1/2 cups cubed cooked chicken breast meat
- 1/2 teaspoon garlic powder
- 1/4 cup milk
- 6 (12 inch)s flour tortillas
>> Cooking Time(min) : 30
>> Direction(How To Cook)
- Step 1: Preheat oven to 350 degrees F (175 degrees C). Lightly grease a large baking dish. Step 2: In a medium saucepan over medium heat melt the butter and saute the green onion until tender (about 3 to 4 minutes). Add the garlic powder then stir in the green chiles cream of mushroom soup and sour cream. Mix well. Reserve 3/4 of this sauce and set aside. To the remaining 1/4 of the sauce in the saucepan add the chicken and 1/2 cup of shredded Cheddar cheese. Stir together. Step 3: Fill each flour tortilla with the chicken mixture and roll up. Place seam side down in the prepared baking dish. Step 4: In a small bowl combine the reserved 3/4 of the sauce with the milk. Spoon this mixture over the rolled tortillas and top with the remaining 1/2 cup of shredded Cheddar cheese. Bake in the preheated oven for 30 to 35 minutes or until cheese is bubbly.
=====
>> Recipe Name : Curry Stand Chicken Tikka Masala Sauce
>> Ingredients :
- 1 (14 ounce) can tomato sauce
- 1 cup heavy whipping cream
- 1 onion
- 1 tablespoon ground cumin
- 1 tablespoon vegetable oil
- 1 tablespoon white sugar
- 1 teaspoon cayenne pepper
- 1 teaspoon ground ginger
- 1 teaspoon salt
- 1 teaspoon white sugar
```

(This Screen capture is about keyword search)

As you see, by searching "Chicken", all recipes about chicken were printed out.

4. See all Recipes including Thigh Burger Recipe that I just added. (Menu 4)

```
- 1 clove garlic minced
- 1 cup sliced fresh mushrooms
- 1 pinch red pepper flakes
- 1/2 cup chopped onion
- 2 cups spinach leaves packed
- 2 tablespoons olive oil
- 3 cups chopped tomatoes
- 8 ounces feta cheese crumbled
- salt and pepper to taste
>> Cooking Time(min) : 15
>> Direction(How To Cook)
- 1. Bring a large pot of lightly salted water to a boil. Cook pasta in boiling water until al dente; drain. 2. Meanwhile heat olive oil in a large skillet over medium-high heat; add onion and garlic and cook until golden brown. Mix in tomatoes mushrooms and spinach. Season with salt pepper and red pepper flakes. Cook 2 minutes more until tomatoes are heated through and spinach is wilted. Reduce heat to medium stir in pasta and feta cheese and cook until heated through.
-----
>> Recipe Name : Turkey Pot Pie I
>> Ingredients :
- 1 1/2 cups cubed cooked turkey
- 1 recipe pastry for a (10 inch) double crust pie
- 1 small onion minced
- 1 teaspoon dried oregano
- 1/2 cup milk
- 2 carrots diced
- 2 cubes chicken bouillon
- 2 cups water
- 2 stalks celery chopped
- 3 potatoes peeled and cubed
- 3 tablespoons all-purpose flour
- 3 tablespoons dried parsley
- 4 tablespoons butter divided
- salt and pepper to taste
>> Cooking Time(min) : 60
>> Direction(How To Cook)
- 1. Preheat oven to 425 degrees F (220 degrees C). Roll out bottom pie crust press into a 10 inch pie pan and set aside. 2. Melt 2 tablespoons butter in a large skillet over medium heat; add the onion celery carrots parsley oregano and salt and pepper. Cook and stir until the vegetables are soft. Stir in the bouillon and water. Bring mixture to a boil. Stir in the potatoes and cook until tender but still firm. 3. In a medium saucepan melt the remaining 2 tablespoons butter. Stir in the turkey and flour. Add the milk and heat through. Stir the turkey mixture into the vegetable mixture and cook until thickened. Cool slightly then pour mixture into the unbaked pie shell. Roll out the top crust and place on top of filling. Flute edges and make 4 slits in the top crust to let out steam. 4. Bake in the preheated oven for 15 minutes. Reduce oven temperature to 350 degrees F (175 degrees C) and continue baking for 20 minutes or until crust is golden brown.
-----
>> Recipe Name : 싸이버거
>> Ingredients :
- 달걀
- 마요네즈
- 밀가루
- 햄 없는 닭 허벅지 살
- 소스
- 식용유
- 양파
- 양파가루
- 케첩
- 피클
- 햄버거 빵
- 후추
>> Cooking Time(min) : 20
>> Direction(How To Cook)
- 1. 계란 2개를 푼 후 허벅지 살을 담군다. 2. 허벅지 살에 튀김을 입힌다. 3. 튀긴다. 4. 빵 사이에 마요네즈와 함께 넣는다.
-----
계속하려면 아무 키나 누르십시오 . . .
```

5-1. Trying to Delete Thigh Burger Recipes that I just added. (Menu 3)

```
You've selected [3] to delete recipe
Enter recipe name which you want to delete : 싸이
>> Deletion Failed.
>> There is no matching recipe for "싸이"
>> Did you meant "싸이버거"? try it again
계속하려면 아무 키나 누르십시오 . . .
```

- Deletion Failed because I didn't write full name. It gives recommendations.

5-2. Deleting Thigh Burger Recipe that I just added. (Menu 3)

```
You've selected [3] to delete recipe
Enter recipe name which you want to delete : 싸이버거
>> Deletion Success.
계속하려면 아무 키나 누르십시오 . . .
```

- Deleted Thigh Burger because I wrote full name exactly.

- For convenience, we used CSV format so you can see and edit Recipes in the Excel.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Actually Delicious Turkey Burgers	1. In a large bowl mix ground	15	1	clove garlic peeled and minced,	1	teaspoon salt,	1/4	cup chopped fresh parsley,	1/4	cup finely diced onion,	1/4	cup seasoned bread crumbs,	1/4	teaspoon ground		
2	Aussie Chicken	Step 1: Rub the chicken breasts	25	1	cup sliced fresh mushrooms,	1	tablespoon dried onion flakes,	1	tablespoon vegetable oil,	2	cup shredded Colby-Monterey Jack cheese,	2	tablespoons chopped fresh				
3	Baked Dijon Salmon	1. Preheat oven to 400 degrees	15	1	1/2 tablespoons honey,	1	lemon for garnish,	1/4	cup butter melted,	1/4	cup dry bread crumbs,	1/4	cup finely chopped pecans,	3	tablespoons Dijon mustard,	4	
4	Baked Spaghetti	1. Preheat oven to 350 degrees	60	1	(16 ounce) package spaghetti,	1	(32 ounce) jar meatless spaghetti sauce,	1	onion chopped,	1	pound ground beef,	1/2	teaspoon seasoned salt,	1/3	cup grated Parmesan		
5	Chicken Enchiladas II	Step 1: Preheat oven to 350 degrees	30	1	(10.75 ounce) can condensed cream of mushroom soup,	1	(4 ounce) can diced green chilies,	1	cup shredded Cheddar cheese,	1	tablespoon butter,	1/2	cup chopped				
6	Corned Beef and Cabbage I	1. Place corned beef in large pot	145	1	large head cabbage cut into small wedges,	10	small red potatoes,	3	pounds corned beef brisket with spice packet,	5	carrots peeled and cut into 3-inch pieces						
7	Curry Stand Chicken Tikka Masala Sauce	Step 1: Heat ghee in a large saucepan	65	1	(14 ounce) can tomato sauce,	1	cup heavy whipping cream,	1	onion,	1	tablespoon ground cumin,	1	tablespoon vegetable oil,	1	tablespoon white sugar,	1	teaspoon
8	Easy Tuna Patties	Beat eggs and lemon juice in a bowl	10	1	pinch ground black pepper,	10	tablespoons Italian-seasoned bread crumbs,	2	eggs,	2	teaspoons lemon juice,	3	(5 ounce) cans tuna,	3	tablespoons grated Parmesan		
9	Grilled Salmon I	1. Season salmon fillets with lemon juice	16	1	1/2 pounds salmon fillets,	1/3	cup brown sugar,	1/3	cup soy sauce,	1/3	cup water,	1/4	cup vegetable oil,	garlic powder to taste,	lemon pepper to taste,	salt to taste	
10	Marinated Tuna Steak	1. In a large non-reactive dish	11		minced,	1	clove garlic,	1	tablespoon lemon juice,	1/2	teaspoon chopped fresh oregano,	1/2	teaspoon ground black pepper,	1/4	cup orange juice,	1/4	cup soy sauce
11	Meatball Sandwich	1. Preheat the oven to 350 degrees	20	1	(14 ounce) jar spaghetti sauce,	1	French baguette,	1	egg beaten,	1	pinch salt or to taste,	1	pound ground beef,	1	tablespoon extra-virgin olive oil,	1/2	teaspoon garlic
12	Philly Cheesesteak Sandwich with Garlic Mayo	1. In a small bowl combine meat sauce and cheese	20	1	(8 ounce) package shredded mozzarella cheese,	1	cup mayonnaise,	1	pound beef round steak cut into thin strips,	1	tablespoon olive oil,	1	teaspoon dried oregano				
13	Roasted Rack of Lamb	1. Preheat oven to 450 degrees	20	1	(7 bone) rack of lamb trimmed and frenched,	1	tablespoon Dijon mustard,	1	teaspoon black pepper,	1	teaspoon salt,	1/2	cup fresh bread crumbs,	1/4	teaspoon black		
14	Salsa Chicken	Step 1: Preheat oven to 375 degrees	40	1	cup salsa,	1	cup shredded Cheddar cheese,	2	tablespoons sour cream (optional),	4	skinless boneless chicken breast halves,	4	teaspoons taco seasoning mix				
15	Sausage Pasta	1. Bring a large pot of lightly salted water to a boil	30	1	(10 ounce) package frozen chopped spinach,	1	(14.5 ounce) can chicken broth,	1	(14.5 ounce) can diced tomatoes,	1	onion chopped,	1	pound spicy Italian sausage				
16	Spicy Garlic Lime Chicken	Step 1: In a small bowl mix together	20	1	skinless chicken breast halves,	1	tablespoon olive oil,	1/4	teaspoon black pepper,	1/4	teaspoon cayenne pepper,	1/4	teaspoon dried parsley,	1/4	teaspoon dried thyme		
17	Suki's Spinach and Feta Pasta	1. Bring a large pot of lightly salted water to a boil	15	1	(8 ounce) package penne pasta,	1	clove garlic minced,	1	cup sliced fresh mushrooms,	1	pinch red pepper flakes,	1/2	cup chopped onion,	2	cup spinach leaves		
18	Turkey Pot Pie I	1. Preheat oven to 425 degrees	60	1	1/2 cups cubed cooked turkey,	1	recipe pastry for a (10 inch) double crust pie,	1	small onion minced,	1	teaspoon dried oregano,	1/2	cup milk,	2	carrots diced,	2	cup

6. Setting Meal Plan for 2019/10/11 (Menu 6)

```

You've selected [6] to add plan
Enter plan information in [[ YEAR -> MONTH -> DAY -> DAYNAME -> BREAKFAST -> LUNCH -> DINNER ]] order
Year      : 2019
Month     : 10
Day       : 11
Plan Name : CAU 101 Anniversary
Breakfast (If you want to stop enter "stop" ) : CAU Burger
Cola
stop
Lunch     (If you want to stop enter "stop" ) : CAU Tous les Jours
Milk
stop
Dinner    (If you want to stop enter "stop" ) : CAU Burger
stop
  
```


7. You can see all Meal Plans, including Plan you just added (Menu 8)

```
You've selected [8] to view plan
Plan Name      : Presentation Day
>> Breakfast
- Baked beans
- Glass of reduced milk
- Tomato
- Wholegrain toast with polyunsaturated margarine
```

```
>> Lunch
- Coffee with milk
- Wholegrain sandwich with roast beef
```

```
>> Dinner
- Boiled rice
- Broccoli
- Carrots
- Fish prepared with olive oil
- Potato
```

```
Plan Name      : My Birthday
```

```
>> Breakfast
- Reduced fat yoghurt
- Wholegrain breakfast cereal
```

```
>> Lunch
- Apple
- Sandwich with salad and chicken
```

```
>> Dinner
- Cake
- Pasta with beef mince and red kidney beans
```

```
Plan Name      : CAU 101 Anniversary
```

```
>> Breakfast
- CAU Burger
- Cola
```

```
>> Lunch
- CAU Tous les Jours
- Milk
```

```
>> Dinner
- CAU Burger
```

```
Plan Name      : New Year
```

```
>> Breakfast
- Rice Cake
```

```
>> Lunch
- Rice Cake
```

```
>> Dinner
- Rice Cake
```

계속하려면 아무 키나 누르십시오 . . .

8. You can delete Meal Plan with Date. I deleted CAU 101 Anniversary Plan that I just added. (Menu 7)

```
You've selected [7] to delete plan
Enter date [[ YEAR -> MONTH -> DAY]] order to delete plan
Year      : 2019
Month     : 10
Day       : 11
>> Deletion Success.
계속하려면 아무 키나 누르십시오 . . .
```

- For convenience, we used CSV format so you can see and edit Plans in the Excel.

	A	B	C	D	E	F	G
1	Presentation Day						
2	My Birthday	20191002	Baked beans	Glass of reduced coffee with milk	Wholegrain boiled rice	Broccoli	Carrots
3	CAU 101 Anniversary	20191011	CAU Burger	Cola	CAU Tous les Jours	Milk	CAU Burger
4	New Year	20200101	Rice Cake	Rice Cake	Rice Cake		
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							

9. You can search Plan by Date.

```
You've selected [5] to search plan
Enter date [[ YEAR -> MONTH -> DAY]] order
Year      : 2019
Month     : 10
Day       : 3
Plan Name  : My Birthday
>> Breakfast
- Reduced fat yoghurt
- Wholegrain breakfast cereal

>> Lunch
- Apple
- Sandwich with salad and chicken

>> Dinner
- Cake
- Pasta with beef mince and red kidney beans
```

계속하려면 아무 키나 누르십시오 . . .

G. Explain how you applied object-oriented concepts to the development for your project.

1. [Encapsulation] In our OOP class we learned how to hide unnecessary details.

We always started from private and when it has to be exposed, we made it public.

Plus, we designed the class with their appropriate responsibility. So, our class has only required variables or functions that are related with its responsibility.



```
Recipe.h  Plan.h  Greeter.h  main.cpp
Team1-ILKH  Recipe
10 // stores recipe's name, description, ingredients, time
11 // and performs action related to recipe and it's member.
12 class Recipe {
13 private:
14     // Recipe's Name
15     std::string name;
16     // Recipe's Description
17     std::string description;
18     // Recipe's Ingredients: it is set of ingredient(we use "string" for ingredient)
19     std::set<std::string> ingredients;
20     // Recipe's Expected Time it takes to cook.
21     int time;
22     // ingredientString for saving(Buffer)
23     std::string ingredientString;
24 public:
25     // CTOR: name, description, time, ingredients will be initialized
26     Recipe(std::string && n, std::string && d, int t, std::vector<std::string> i) :
27         name(n), description(d), time(t), ingredients(i) {
28         // insert all ingredient to ingredients set
29         for (std::string s : i)
30             ingredients.insert(s);
31     };
32
33     /* GET */
34     // get Recipe name
35     const std::string & getName() const { return name; }
36     // get Recipe description
37     const std::string & getDescription() const { return description; }
38     // get Recipe ingredients
39     const std::set<std::string> & getIngredients() const { return ingredients; }
40     // get Recipe time
41     int getTime() const { return time; }
42
43     /* SET */
44     // sets Recipe name
45     void setName(std::string & n) { name = n; }
46     // sets Recipe description
47     void setDescription(std::string & d) { description = d; }
48     // sets Recipe time
49     void setTime(int t) { time = t; }
50
51     /* Functions */
52     // insert Recipe ingredient to ingredients set. and returns second stored value
53     bool addIngredient(std::string & n) { return ingredients.insert(n).second; }
54     // remove Recipe ingredient to ingredients set. and returns second stored value
55     bool removeIngredient(std::string & n) { return ingredients.erase(n) == 1; }
56     // find if there is ingredient
57     bool searchIngredient(std::string & n) { return ingredients.find(n) != ingredients.end(); }
58     // make ingredients into single String
59     std::string getIngredientsIntoString() {
60         for (auto ingredient : ingredients) {
61             ingredientString.append(ingredient);
62             ingredientString.append("$");
63         }
64
65         return ingredientString;
66     }
67     // print Recipe data
68     void print() {
69         std::cout << "Recipe Name : " << name << std::endl;
70         std::cout << "Ingredients : " << endl;
71         for (std::string s : ingredients) {
72             std::cout << " - " << s << std::endl;
```

2. [Inheritance] We looked at the objects relation carefully and made programming more efficient.

For Instance, in case of database manager, many kinds of database managers were needed. We needed Plan Database manager, Meal Database Manager, Recipe Database Manager. Each Database managers was similar to each other and also the code was almost the same but saving format or few functions details were little bit different. So, we made DataBase [parent class] which had all overlapping details of all kinds of Database Managers. Inheriting Database [parent class] traits, making other Database Managers were easier (High Reusability). Also, we could eliminate the same codes.

In addition, with virtual keyword we implemented different details in the Database Managers' functions. That keyword acted like interface and all Database Managers [child class] had implemented mandatory details.

- DataBase [Parent]

```
23 // stores db map
24 // and controls entire DB system
25 class DataBase {
26 protected:
27     std::map<_key, _value> db;
28 public:
29     // static std::string recipeKeys[] = { "index", "name", "description", "time", "ingredient0", "ingredient1", "ingredient2", "ingredient3", "ingredient4" };
30
31     // insert pair of key and value
32     void _insert(_key k, _value v) { db.insert(std::pair<_key, _value>(k, v)); }
33
34     _value & _select(_key k) {
35         // if you do not find key (throws exception)
36         if (db.find(k) == db.end()) {
37             if (isEqual<_key, std::string>::value) throw DBException(0, k.c_str());
38             else if (isEqual<_key, Date>::value) throw DBException(1, k.c_str());
39             else throw DBException(2, k.c_str());
40         }
41
42         // if you find key (return value)
43         return (*db.find(k)).second;
44     }
45
46     // delete pair of key and value, insert new pair of key and value
47     void _update(_key k, _value v) {
48         db.erase(k);
49         db.insert(std::pair<_key, _value>(k, v));
50     }
51
52     // delete pair of key and value
53     void _delete(_key k) { db.erase(k); }
54
55     // show items with given key
56     virtual void _show(_key k) = 0;
57
58     // show all items in DB
59     virtual void _showAll() = 0;
60
61     // save DB(items) to local
62     virtual void _save() = 0;
63
64 };
65
```

● RecipeDB [Child]

```
95 // database manager for Recipe
96 // performs file I/O operations, db items printing
97 class RecipeDB : public DataBase<std::string, Recipe> {
98 private:
99
100 public:
101     // Ctor
102     RecipeDB() {
103         // msg
104         // std::cout << "[디버그] recipe DB 생성" << std::endl;
105         // Load File
106         std::vector<std::map<std::string, std::string>>> parsedData = CSVParser::instance().read("11KHRecipeDB.csv");
107
108         for (auto myMap : parsedData) {
109             // Construct Recipe Name from csv
110             std::string recipeName = myMap->find("name")->second;
111
112             // Construct Plan from csv
113             Recipe* recipePtr = new Recipe(
```

● PlanDB [Child]

```
152 // database manager for Plan(name + 3 meal)
153 // performs file I/O operations, db items printing
154 class PlanDB : public DataBase<Date, Plan> {
155 private:
156
157 public:
158     // Ctor
159     PlanDB() {
160         // Load File
161         std::vector<std::map<std::string, std::string>>> parsedData = CSVParser::instance().read("11KHPlanDB.csv");
162
163         for (auto myMap : parsedData) {
164             // Construct Date from csv
165             Date* datePtr = new Date(strdup(( myMap->find("date")->second.c_str())));
166
167             // Construct Plan from csv
168             Plan* planPtr = new Plan(
169                 // plan name
170                 std::string(myMap->find("name")->second),
171                 *datePtr,
172                 {
173                     //아침
174                     Meal( CSVParser::instance().split( myMap->find("breakfast")->second , '$', -1 ) ),
175                     //점심
176                     Meal( CSVParser::instance().split( myMap->find("lunch")->second , '$', -1 ) ),
```

- Explain what you felt and learned from the project.

[고주형] It was nice to experience object-oriented programming as a team project. It was good chance to think deeply about object-oriented programming and design.

I was team leader so I kept track of project progress. I had to give everyone's work equally to until the program is fully implemented. Also, I had to think in Bottom Up manner. This gave me enlightenment about what is good design and why we use object-oriented programming. I had to rapidly gathered all of our team member's code and I integrated it into our master branch (we used git). When convention was different, I had hard time understanding it. But, unified convention and nicely encapsulated code was easy to use and I didn't confuse much what to use because unneeded functions or variables were forbidden.

[김호성] I've never had a chance to do a project on a team-by-team basis before, and it's been great to be able to do it through this team project, and the process of working together was fun.

[손희승] Coding with others was hard at first but after few days later, it became comfortable. I learned few tips. I know what Github is but didn't use it much. In this project I got chance to know about Git like, how to cooperate using Git, how to invite collaborator. With kind team members, I think I learnt a lot about cooperating.

[이준협] It was interesting to think in various ways because no implementation or details were restricted. And I felt again that teamwork is important.

[이하람] Through this project, I learned and knew about what an Object-oriented programming in the abstract. Also, it was very interesting to discuss together for making the programming because it was the first time for me to learn about c++.

H. Conclusion

Our team project was successful. We argued a lot about what is more good design or what data structure we should use for many models. We learned from each other by talking, arguing and cooperating. We learned how to cooperate in bottom up style code. This kind of experience will be helpful in the future.

Thank You😊