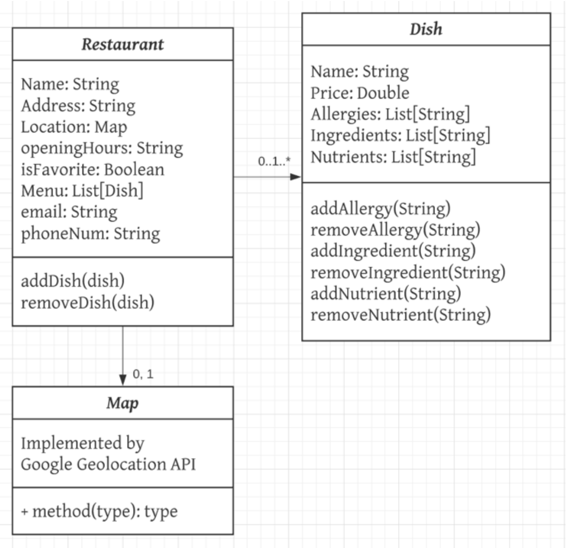
Programming 2: Studio Project’s General Plan

1. Personal Information

- Jeheon Kim (716954)
- Data Science, 2nd Year
- 12nd of February 2020
- Project: Lunch list Application
- Title: Lunch list Application with JSON and XML

2. Class Structure



- Mainly there will be 2 classes: Restaurant and Dish
- The class, Restaurant, can have none (when lunch not served) or multiple of dishes.
- Another class, Map, is still under contemplation for the Graphical User Interface.
- The method for the Map isn’t yet defined: perhaps None
- Overall relationship between classes are as depicted in the UML above

3. Use Case Description

: A user runs the program and the lunch menus from two different restaurants, Fazer and Sodexo, of the current date first appear on the screen. If the user chooses one of two available restaurants as their favorite, then the program will show the menu of chosen restaurant first when the program is initiated. From the menu, the user can check the detailed information of each dish by clicking it. From there, the user can add a specific ingredient into his/her list of favorite ingredients. And the program will notify the user if the dish with any of user’s favorite ingredients is available on the current day. Also, the user can filter the list of dishes based on their choice of specific diet or allergy.

Available allergies are: Gluten Free, Lactose-Free, Milk-free, Low-lactose and Contains Allergens
Available diets are: Wellbeing, Vegans, Including pork, The country of meat origin and etc.

4. Algorithms

The main algorithm (method) of this application is the JSON and XML parsing from the Fazer and Sodexo API. As long as the parsing is done smoothly without errors (Try... Catch exception), rest of the program is as described in the use case description above. (Allows user customization and etc.) JSON and XML parsing will be the watershed of this project.

5. Data Structures

Most of data required for this application will be achieved straight from the API in real-time. However, for the simplicity of the demonstration, it might try using the csv file as well.

6. Schedule

Week	1	2	3	4	5	6	7	8	9	10
Starting Date	Feb 21	Feb 28	Mar 7	Mar 14	Mar 21	Mar 28	Apr 4	Apr 11	Apr 18	Apr 25
System	🟦	🟦	🟦	🟦	🟦					
GUI				🟨	🟨	🟨	🟨	🟨	🟨	
Testing								🟩	🟩	🟩

7. Unit Testing Plan

Testing for this project will be focused on handling exception and errors from the parsing data. And for the possible network problem, program with csv file will be tested as well. For the GUI, several tests can be performed with fellow students. (Heuristic Testing)

8. References and Links

https://kanttiinit.fi/
https://www.fazerfoodco.fi/en/restaurants/Ravintolat-kaupungeittain/espoo/dipoli/
https://www.sodexo.fi/aalto-yliopiston-opiskelijaravintolat https://github.com/Kanttiinit
https://api.ruoka.xyz/
http://otfried.org/scala/gui.html
https://manuel.bernhardt.io/2015/11/06/a-quick-tour-of-json-libraries-in-scala/

9. Appendixes