

FOODIEGRAM

CE/CZ 1003 iNTRODUCTION TO COMPUTATIONAL THINKING

GERALD TAN JING KAI ( U1622611H )

SUN QI NAN ( U1620772L )

ANG KOK JUN KEITH ( U1620572G )

SIOW YUAN KAE ( U1620559A )

TEO ZHE WEI ( U1620645F )

# **Problem:**

Have you ever noticed that the food courts are always congested during the lunch hours? This problem of crowds in food courts has always been existent throughout any food court or canteen during peak hours. The root of this problem could be due to the nature of food courts – there are too many food choices to choose from. Food choices vary from Japanese cuisine, local cuisine to western cuisine. There are simply too many choices to choose from! This increases our thinking time, which in turn, increases the time one have to spend in the food court, which caused the large crowd.

In developed societies a large amount of choice is commonly associated with welfare and freedom but too many choices can cause the feeling of less happiness, less satisfaction and can even lead to paralysis.[[1]](#footnote-1) This is the “ Paradox of Choice”, as defined by Barry Schwartz, an American psychologist at Swarthmore College.

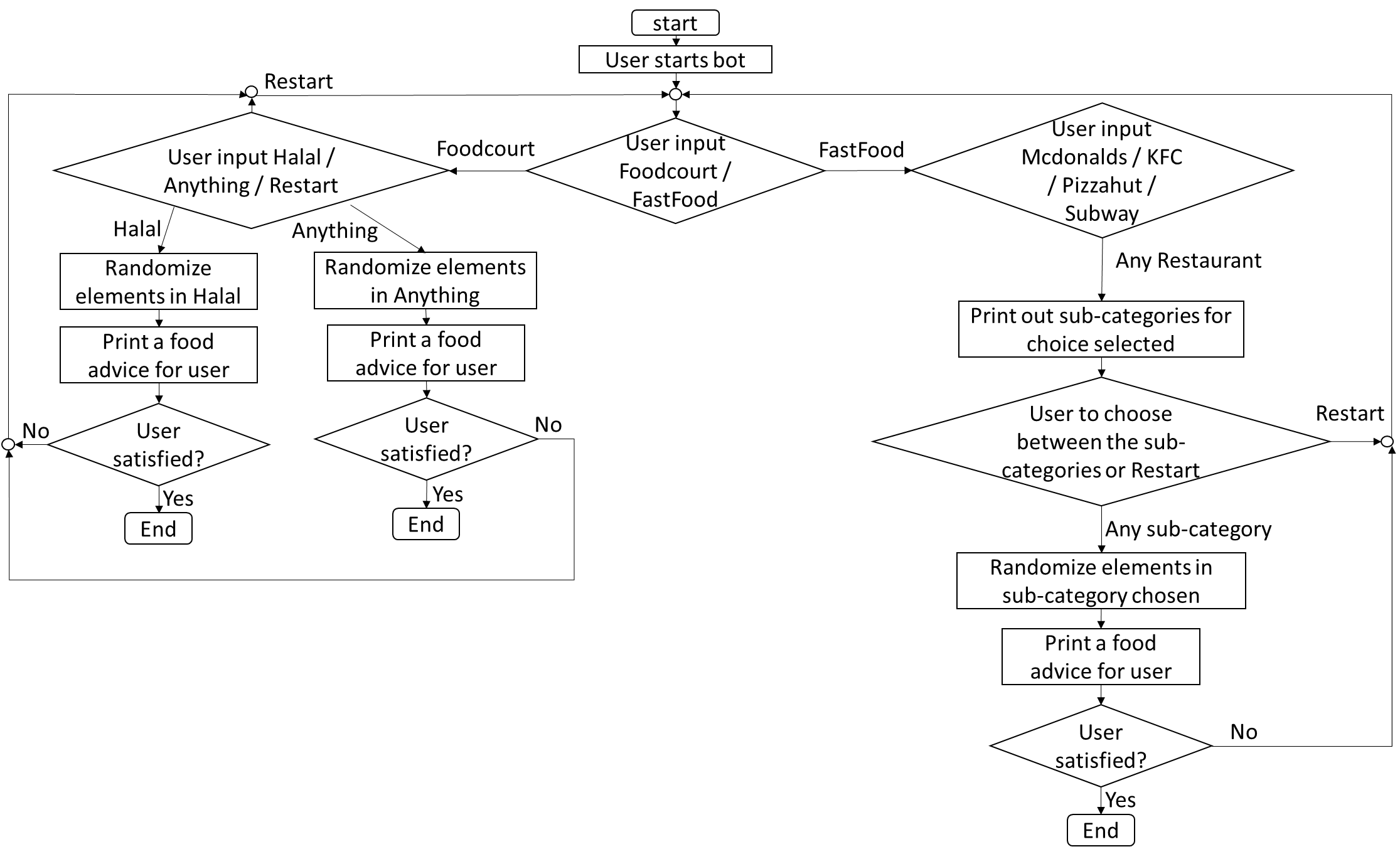
Too many choices often overwhelm us and lead us to absconding from making good decisions all together. With increasing amount of choices, the time and effort spent gathering information to make a good choice also increases.[[2]](#footnote-2)

This problem haunts us food courts goers on a daily basis. It not only affects us, but also other consumers! As the seats and standing spaces in each food courts are very limited, every time a student gets lost in his thoughts choosing which stall to queue for would cause a seat being hogged or a standing space being occupied for unnecessary reasons. This adds up to the space constrains that we are already facing and hence it worsens the congestions even more.

Besides, this thinking time will add on even more if the next group of eaters also decides to spend some time thinking of the same question. These time wasted is completely avoidable and could have been better spent doing everything else.

# **Solution:**

Our group propose to program a telegram bot, named “@Foodiegram” , to help food court goers choose his food. This is done by randomizing all the food choices available from all stalls in a food court and then printing out the chosen choice for the user.

The following diagram will show an overview on how @Foodiegram will work.

Steps on how a user uses our bot is as followed:

1) User starts the bot by sending " /start " to the bot.

2) User is given 2 choices to choose from: "Food court " or " FastFood ".

3) If user chose “ Food court”, he is given 3 more choices to choose from: " Halal" or " Anything " and “Restart”. In this case, the user will carry on from **step 4**.

If user chose “FastFood”, he will be given 4 choices to choose from: Mcdonalds, KFC, Subway & Pizzahut. In this case. the user will carry on from **step 7**.

4) Bot will now look into the narrowed down database based on the choice user has selected.

5) Bot will randomize and print one choice to the user.

6) User is given 2 choices, “YES” and “NO!” to show his satisfaction on the food advice given. If user is satisfied with the choice, he will choose the “YES” option, where he will end chat with the bot. Otherwise, the user will choose the “NO!” option where he will be redirected back to **step 2**.

7) Bot will now print out the sub-categories for each individual restaurant. Some examples of sub-categories are: “Alacarte” or “meals”. The “Restart” option is also available for the user to go back to **step 2.**

8) Bot will now look into the narrowed down database based on the choice user has selected.

9) Bot will randomize and print one choice to the user.

10) User is given 2 choices, “YES” and “NO!” to show his satisfaction on the food advice given. If user is satisfied with the choice, he will choose the “YES” option, where he will end chat with the bot. Otherwise, the user will choose the “NO!” option where he will be redirected back to **step 2**.

11) User ends the chat.

Screenshots illustrating how @Foodiegram works:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step 1 | C:\Users\sykice\AppData\Local\Microsoft\Windows\INetCacheContent.Word\WhatsApp Image 2016-09-30 at 12.51.50 AM.JPEG |  | Step 2 | C:\Users\sykice\AppData\Local\Microsoft\Windows\INetCacheContent.Word\WhatsApp Image 2016-09-30 at 12.51.51 AM.JPEG |
| Step 3 |  |  | Step 3\* | C:\Users\sykice\AppData\Local\Microsoft\Windows\INetCacheContent.Word\WhatsApp Image 2016-09-30 at 12.51.54 AM.JPEG |
| Step 4 +5 | C:\Users\sykice\AppData\Local\Microsoft\Windows\INetCacheContent.Word\WhatsApp Image 2016-09-30 at 12.51.53 AM (1).jpeg |  | Step 6\* | C:\Users\sykice\AppData\Local\Microsoft\Windows\INetCacheContent.Word\WhatsApp Image 2016-09-30 at 11.46.34 AM.JPEG |
| Step 4+5\* | C:\Users\sykice\AppData\Local\Microsoft\Windows\INetCacheContent.Word\WhatsApp Image 2016-09-30 at 12.51.52 AM (1).jpeg |  | Step 6\* |  |
| Step 7 |  |  | Step 8+9 | C:\Users\sykice\AppData\Local\Microsoft\Windows\INetCacheContent.Word\WhatsApp Image 2016-09-30 at 12.51.55 AM.JPEG |
| Step 10 | C:\Users\sykice\AppData\Local\Microsoft\Windows\INetCacheContent.Word\WhatsApp Image 2016-09-30 at 11.41.02 AM.JPEG |  |  |  |

\* Alternative choice

**Benefits:**

With the help of @FoodieFinder, food court goers can now have an advice of which meal to get. Even if the user does not like and rejects the advice given by our bot, our bot can repeat the process to advice the user, until the user is contented with what is recommended by our bot.

This can help the user decide on what to eat even before he goes to the food court. Therefore, he will save up the thinking time in the foodcourt and can go straight to queue for his desired meal instead of hogging the seats and causing unnecessary congestions while thinking of what to have for lunch. Assuming that this works on more people, the congestion problem in the food court can be reduced significantly.

# **Constrains and Limitations:**

Even though our bot is able to make sound advice with regards to the choice of food to users, it does not advice the user about the available additional toppings like eggs, sauces or other a la carte ingredients.

This bot also does not allow the user to indicate his dietary preferences, like whether the user prefers spicy food, or whether the user prefers fried food. It is not customized to suit the preference of each individual. In addition, information such as quantity and food availability is not shown. There are also no details about any new food products that the stalls are going to launch soon.

Currently, this bot only includes the database of the food menu in North Spine Canteen A. However, it can be further improved if it includes menus of other food courts like The Quad and Koufu, etc. This will ensure that our bot can serve a larger population.

We were also unable to scrap data from some of the websites, thus resulting in hard-coding. For example: https://www.kfc.com.sg/our-food/#burgers

# **Conclusion:**

This bot is effective in advising food court goers with a list of choices of food to choose from. Despite the various insufficiencies and the limited information the bot can provide, it can definitely serve its basic yet most crucial function.

This is only a trial bot for just the food court and fast food in North Spine. Given more time and resources, further improvements can be made for this bot to serve a larger population.

1. http://discovery.ucl.ac.uk/1336308/

   [↑](#footnote-ref-1)
2. https://www.fastcompany.com/3031364/the-future-of-work/why-having-too-many-choices-is-making-you-unhappy [↑](#footnote-ref-2)