

RoboChef Pepper Mills

Presented by:

Anthony El Chemaly
Catherina El Khoury
Elie El Rayess
Rodaina Fayad
Garo Margossian
Fawzi El Khoury

Presented to:

Dr. Elie Saad



Discussion Points

- Introduction
- Use Case and Personas
- Code
- Challenges
- Solution to challenges
- Future work

Pepper Mills





Introduction

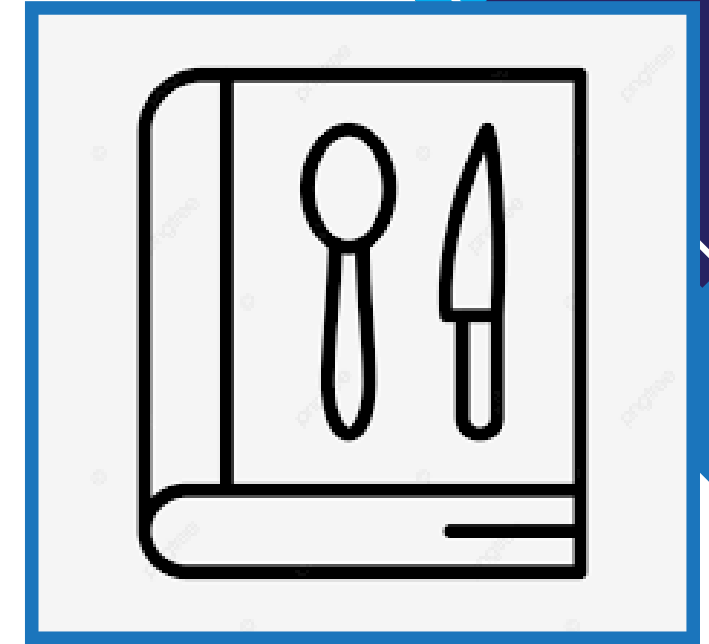
Imagine a kitchen where RoboChef assists you in preparing a delectable meal, from selecting recipes and gathering ingredients and help you be entertained while cooking.

Our project aims to achieve these goal by improving the cooking experience through technology.



Use Case

- Request a Recipe
- Recommend a recipe
- Create Account
- Provide Nutritional Information for a Specific Recipe
- Play music or tell a joke
- Monitor kitchen appliance

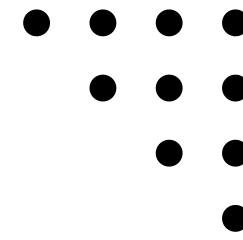


Covered Use Cases

- Request a Recipe
- Recommend a recipe
- Play music or tell a joke



Personas



Three different personas:

- Samira who wants to feed her family
- Sally who wants to be a Professional Chef
- Anastasia wants to learn how to cook

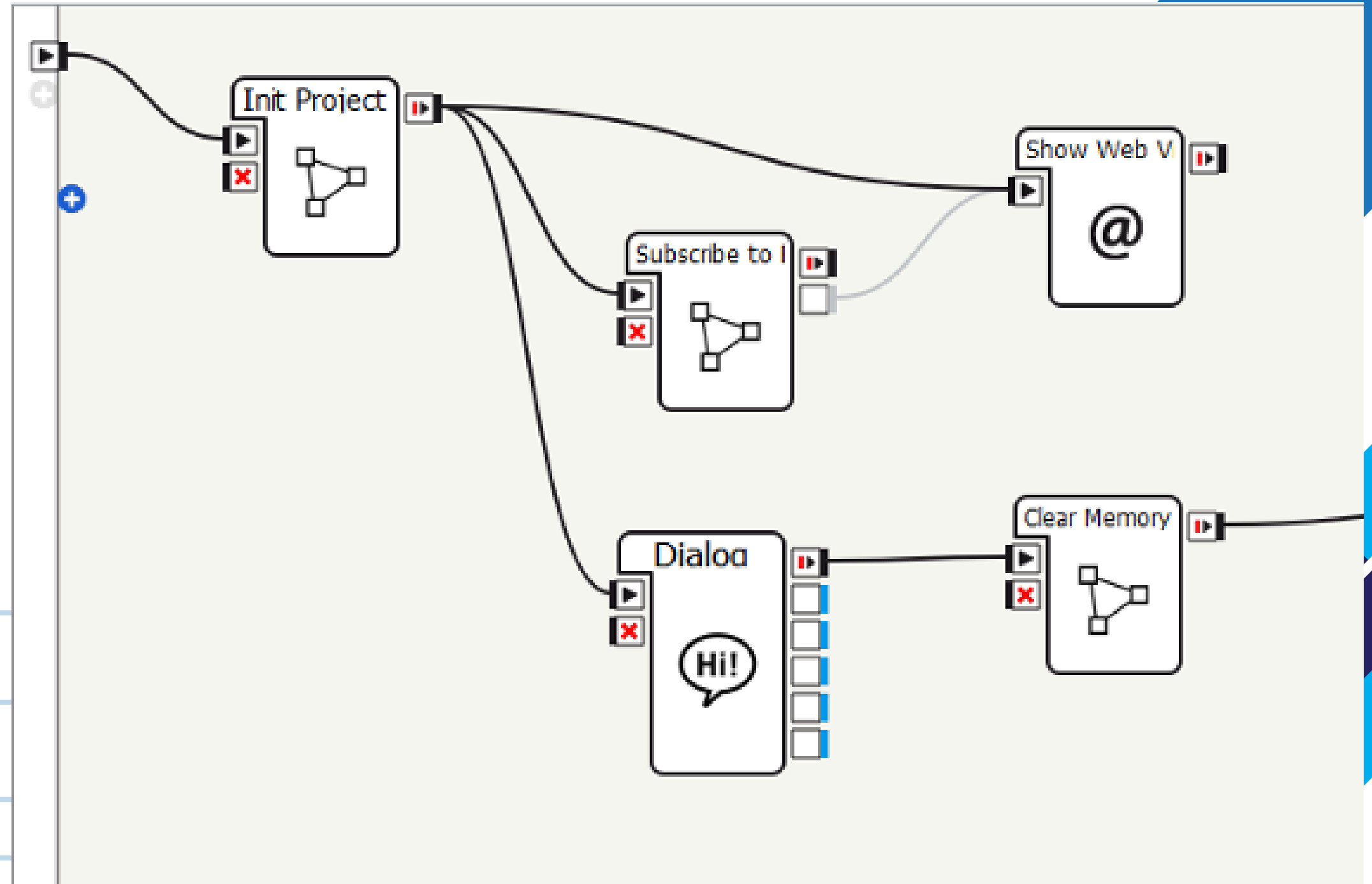


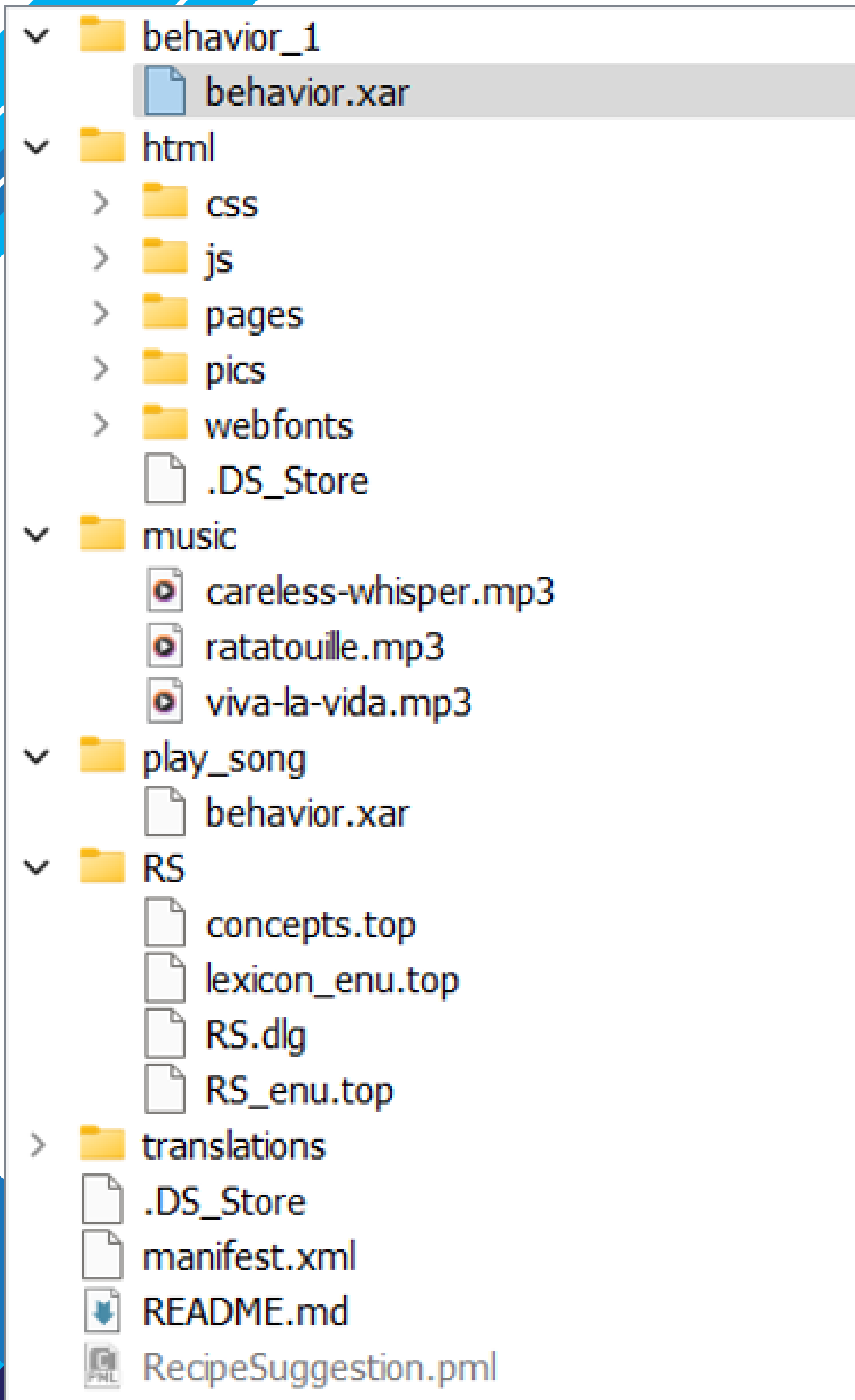
Our Work

In our project we made use of different blocks to reach the required functionality.

The functionality we aimed to achieve was that of the three use cases :

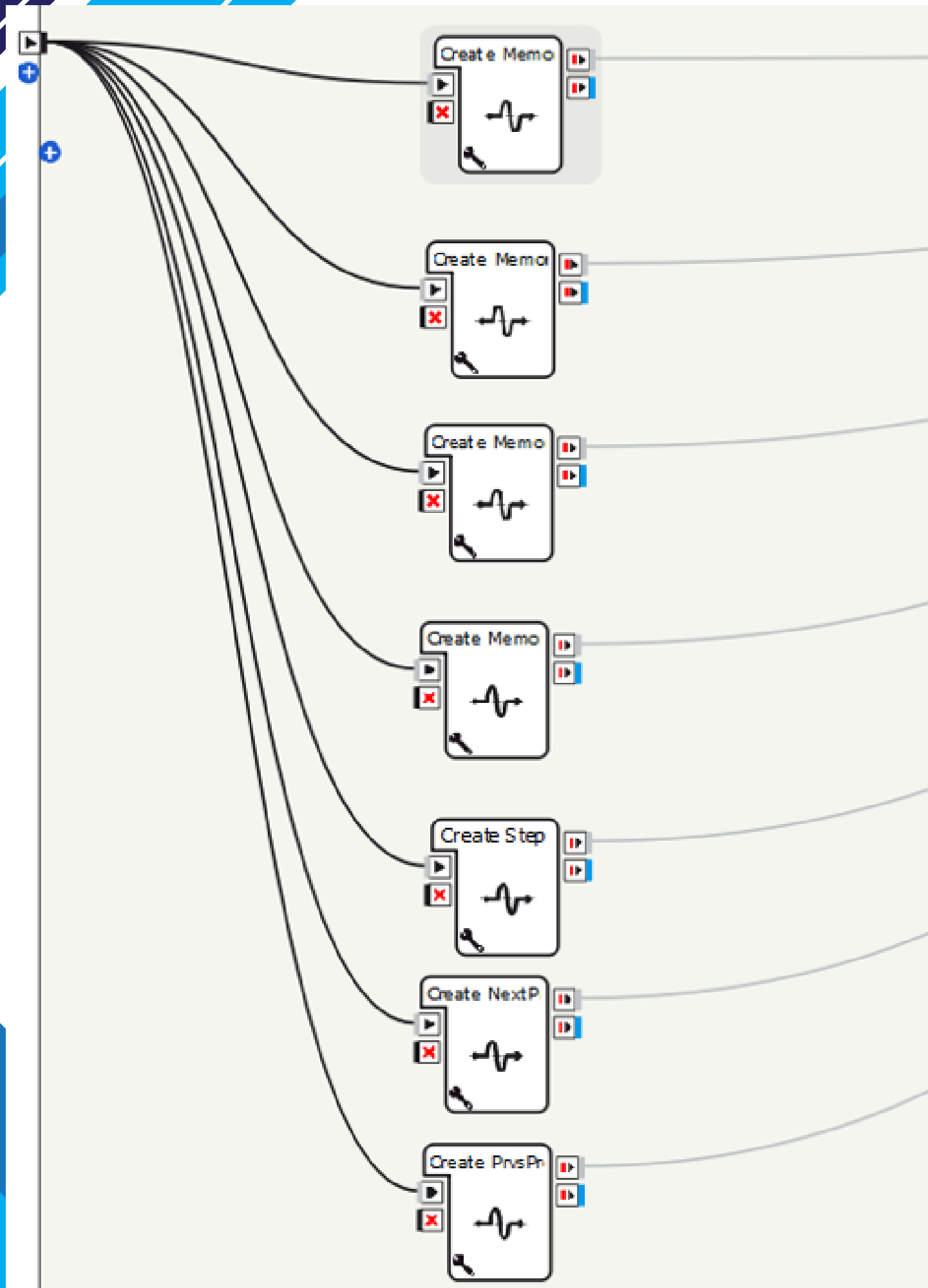
- UC01- Request a Recipe,
- UC02 – Recommend a Recipe
- UC-05 Play music or Tell a joke.



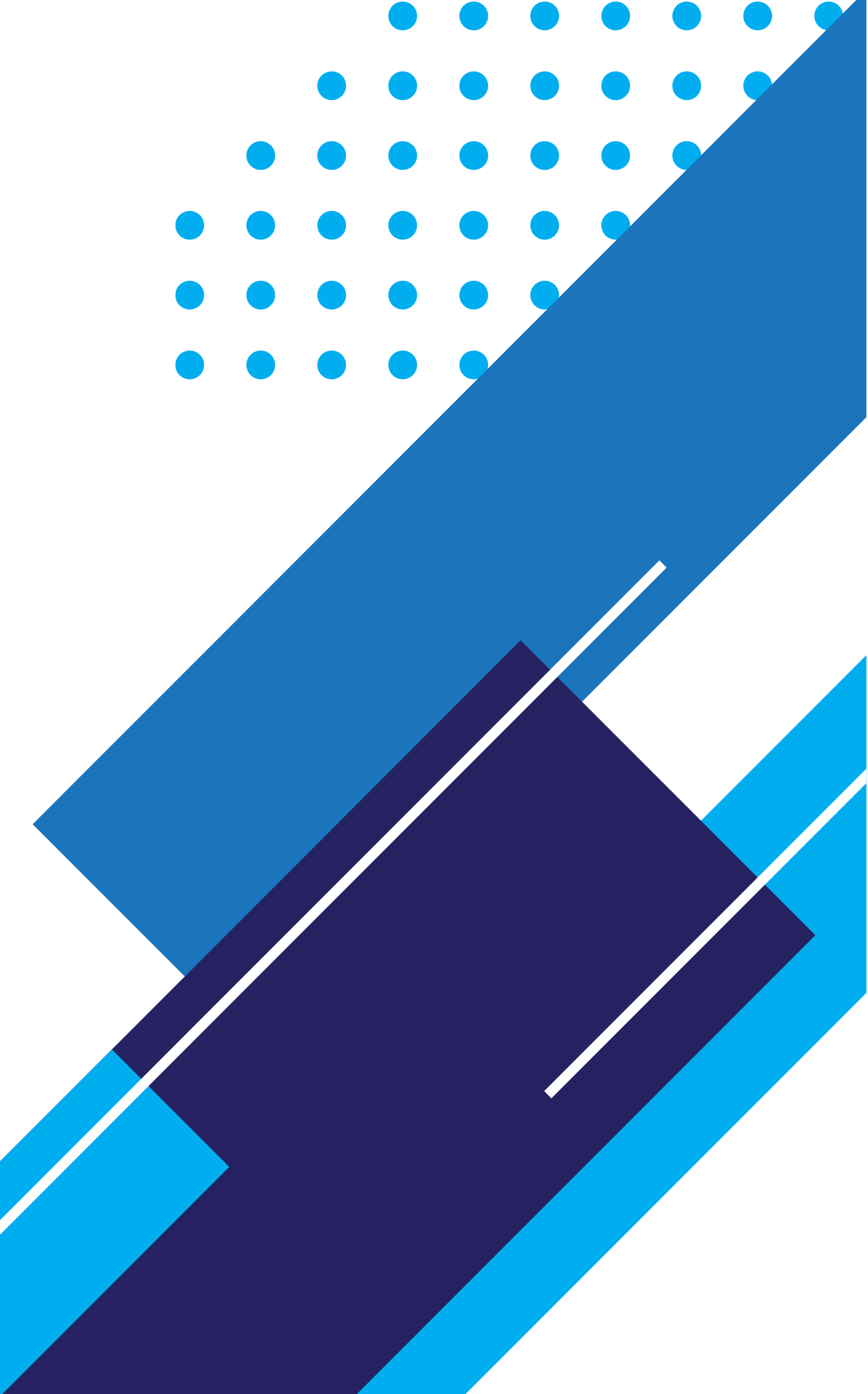
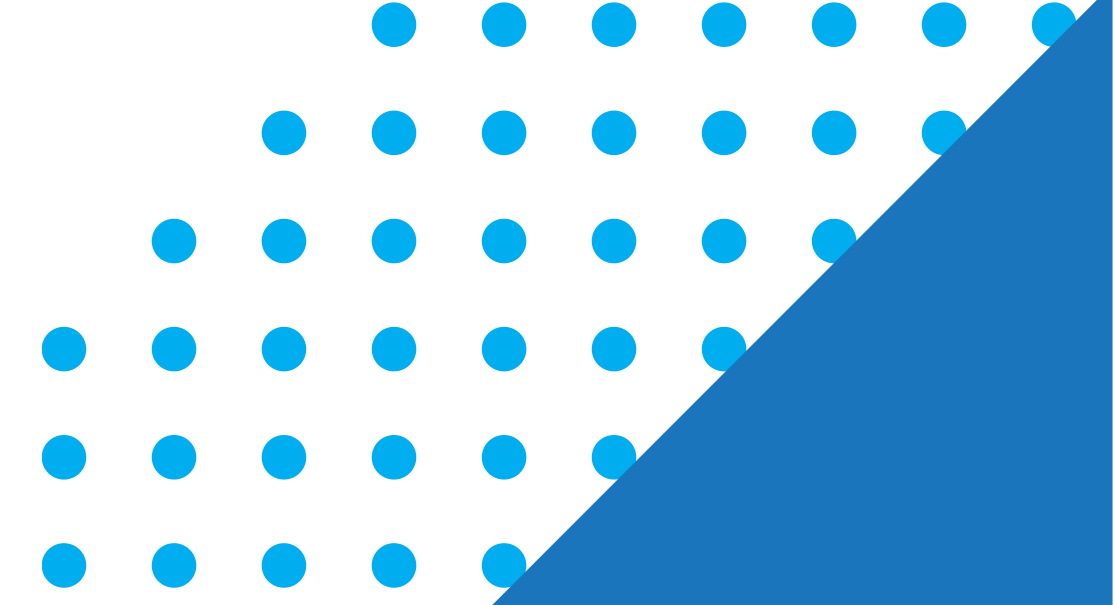
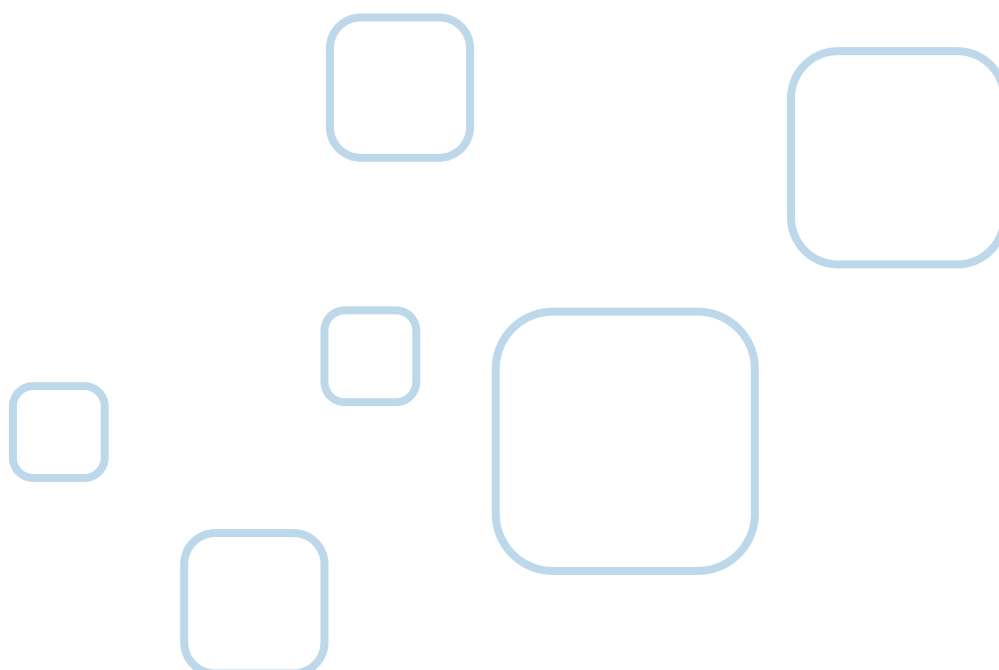
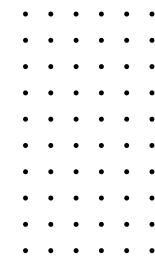
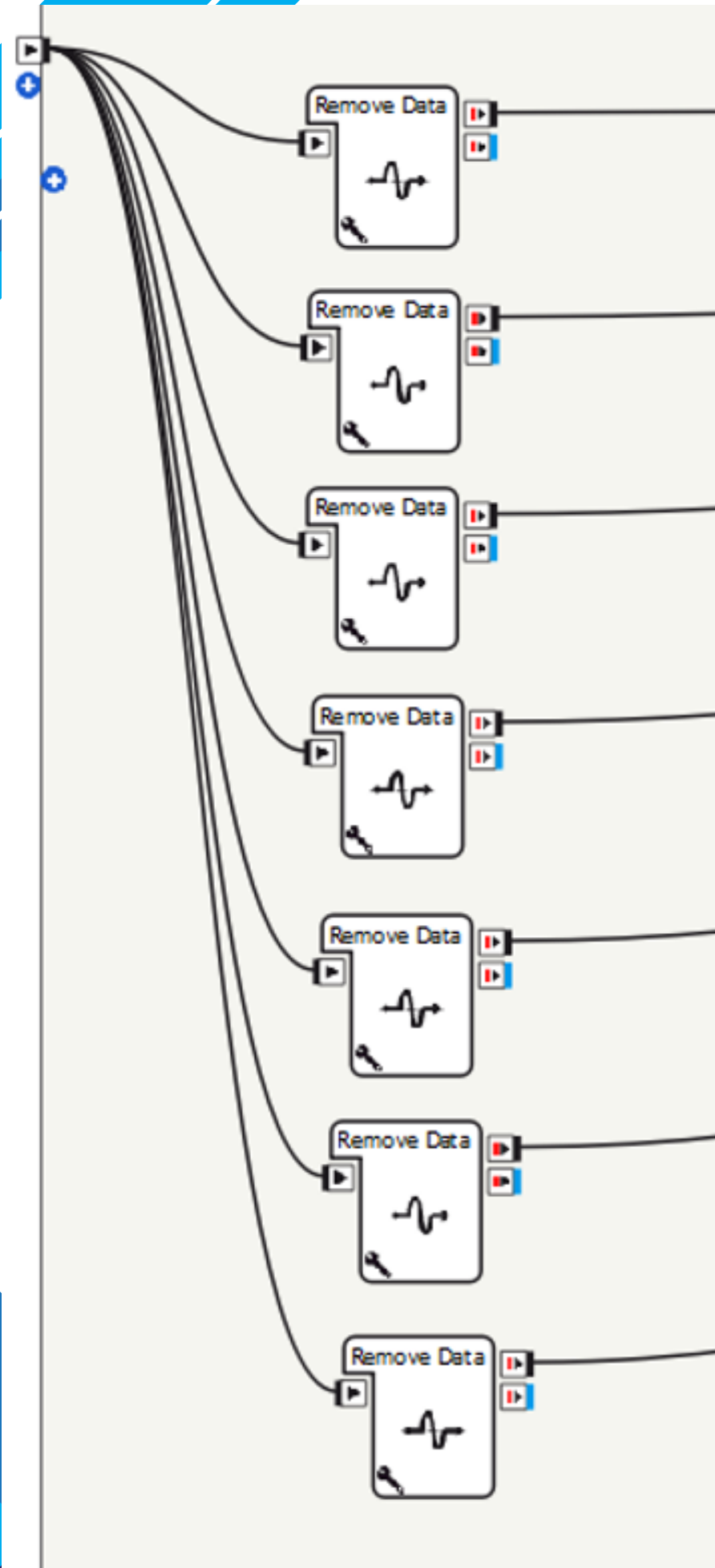


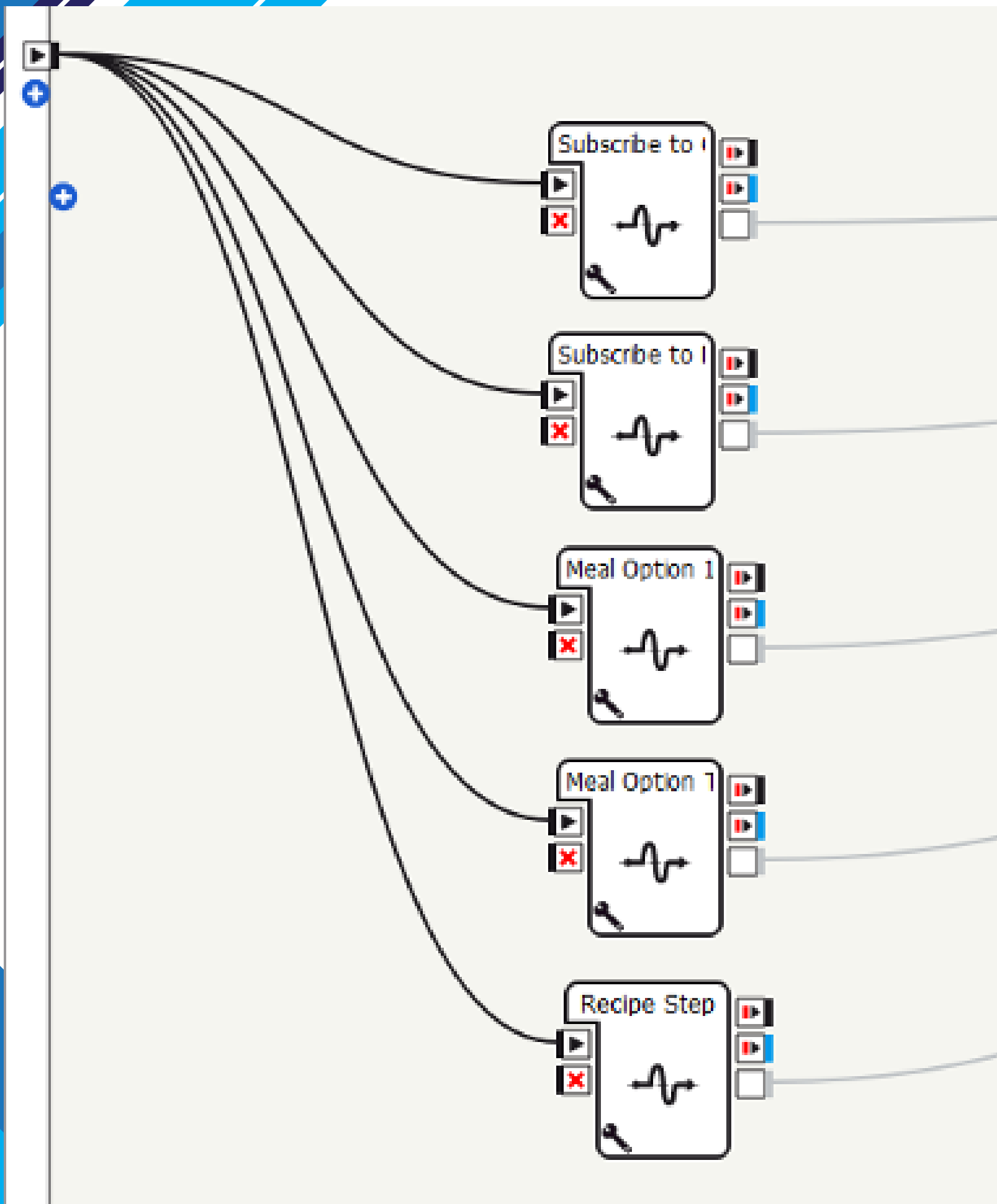
Folder Structure

Init Project Block



Clear Memory Block





Subscribe to Events

WebView Python Script

```
# create proxy on ALMemory
memProxy = ALProxy("ALMemory", "localhost", 9559)

#get data. Val can be int, float, list, string
cuisine = memProxy.getData("cuisine")
mealtype = memProxy.getData("mealtype")
option1 = memProxy.getData("option1")
option2 = memProxy.getData("option2")
nextinstruction = memProxy.getData("nextinstruction")
```

WebView Python Script

```
if tabletService:
    # use default robot IP address from the tablet
    robotIP = "198.18.0.1" + tabletService.getRobotIp()
    if nextinstruction:
        url = "http://{ip}/apps/{appName}/pages/step.html?nextinstruction={nextinstruction}".format(ip=robotIP, appName=appName, nextinstruction=nextinstruction)
    else:
        if option1 and option2:
            url = "http://{ip}/apps/{appName}/pages/choice.html?cuisine={cuisine}&mealtype={mealtype}&option1={option1}&option2={option2}".format(ip=robotIP,
appName=appName, cuisine=cuisine, mealtype=mealtype, option1=option1, option2=option2)
        else:
            if cuisine or mealtype:
                url = "http://{ip}/apps/{appName}/pages/displayinfo.html?cuisine={cuisine}&mealtype={mealtype}".format(ip=robotIP, appName=appName, cuisine=cuisine,
mealtype=mealtype)
            else:
                url = "http://{ip}/apps/{appName}/pages/index.html".format(ip=robotIP, appName=appName)

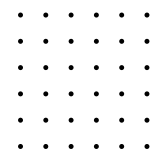
    self.logger.info(url)

    # Show the web view on the tablet
    tabletService.showWebview(url)
else:
    self.logger.warning("Couldn't find tablet service, so can't set application: %s" % appName)
```

Activate Windows

Go to Settings to activate Windows

QiChat Dialogue Sample



```
proposal: %cuisineprop Which cuisine are you in the mood for? I have Italian \pau=80\, French \pau=80\, Chinese \pau=80\ and Mexican!
u1:(Italian) %cuisine=italian Great! \pau=200\ Now, do you have a specific type of dish or meal time in mind from the Italian cuisine?
u2:(starter) \RSPD=40\ %mealtype=starter Perfect! For a cozy Italian starter, I'd recommend a Caprese Salad. For an adventurous one, how about Arancini Balls?
$option1=caprese-salad $option2=arancini-balls
u2:(dessert) \RSPD=40\ %mealtype=dessert A quick Italian dessert would be Panna Cotta. Feeling cozy? Try Tiramisu. $option1=panna-cotta $option2=tiramisu
u2:(breakfast) \RSPD=40\ %mealtype=breakfast A quick Italian breakfast would be a Cappuccino with a Cornetto. I also recommend a Frittata. $option1=cappuccino
$option2=frittata
u2:(lunch) \RSPD=40\ %mealtype=lunch For a cozy Italian lunch, try a Margherita Pizza or maybe a Spaghetti Carbonara? $option1=margherita-pizza $option2=spaghetti-
carbonara
u2:(dinner) \RSPD=40\ %mealtype=dinner Feeling adventurous for dinner? How about Lasagna al Forno? Feeling cozy? Try Osso Buco. $option1=lasagna-al-forno $option2=osso-
buco
```

JS Raise Event Function

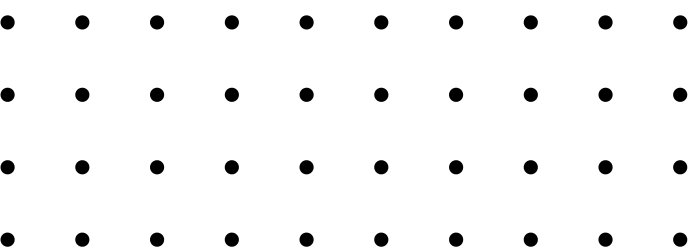
```
1  ✓ function raiseEvent(name, value) {  
2  ✓    QiSession(function (session) {  
3  ✓      session.service("ALMemory").then(  
4  ✓        function (mem) {  
5          mem.raiseEvent(name, value);  
6        },  
7  ✓      function (error) {  
8        console.log("An error occurred:", error);  
9      }  
10     );  
11   });  
12 }
```

JS Display Function

```
40 function displayPageInformation() {
41     document.getElementById("cuisine").innerHTML =
42     | "Cuisine: " + decodeURI(getUrlParam("cuisine", ""));
43     document.getElementById("mealtype").innerHTML = decodeURI(
44     | "Meal Type: " + getUrlParam("mealtype", "")
45     | );
46
47     imageName1 = getUrlParam("option1", "");
48     if (imageName1) {
49         document.getElementById("optionOneLabel").innerHTML =
50         | transformText(imageName1);
51         document.getElementById("option1").src = "../pics/" + imageName1 + ".png";
52         document.getElementById("option1").onclick = function () {
53         | raiseEvent("chosenOption", imageName1);
54         | };
55     }
56     imageName2 = getUrlParam("option2", "");
57     if (imageName2) {
58         document.getElementById("optionTwoLabel").innerHTML =
59         | transformText(imageName2);
60         document.getElementById("option2").src = "../pics/" + imageName2 + ".png";
61         document.getElementById("option2").onclick = function () {
62         | raiseEvent("chosenOption", imageName2);
63         | };
64     }
65 }
```

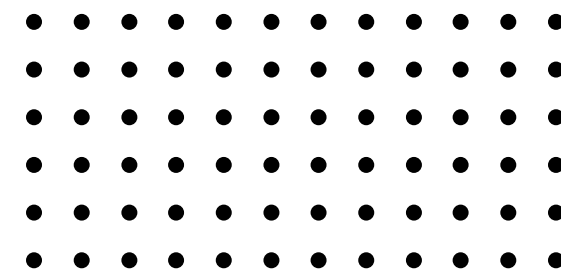
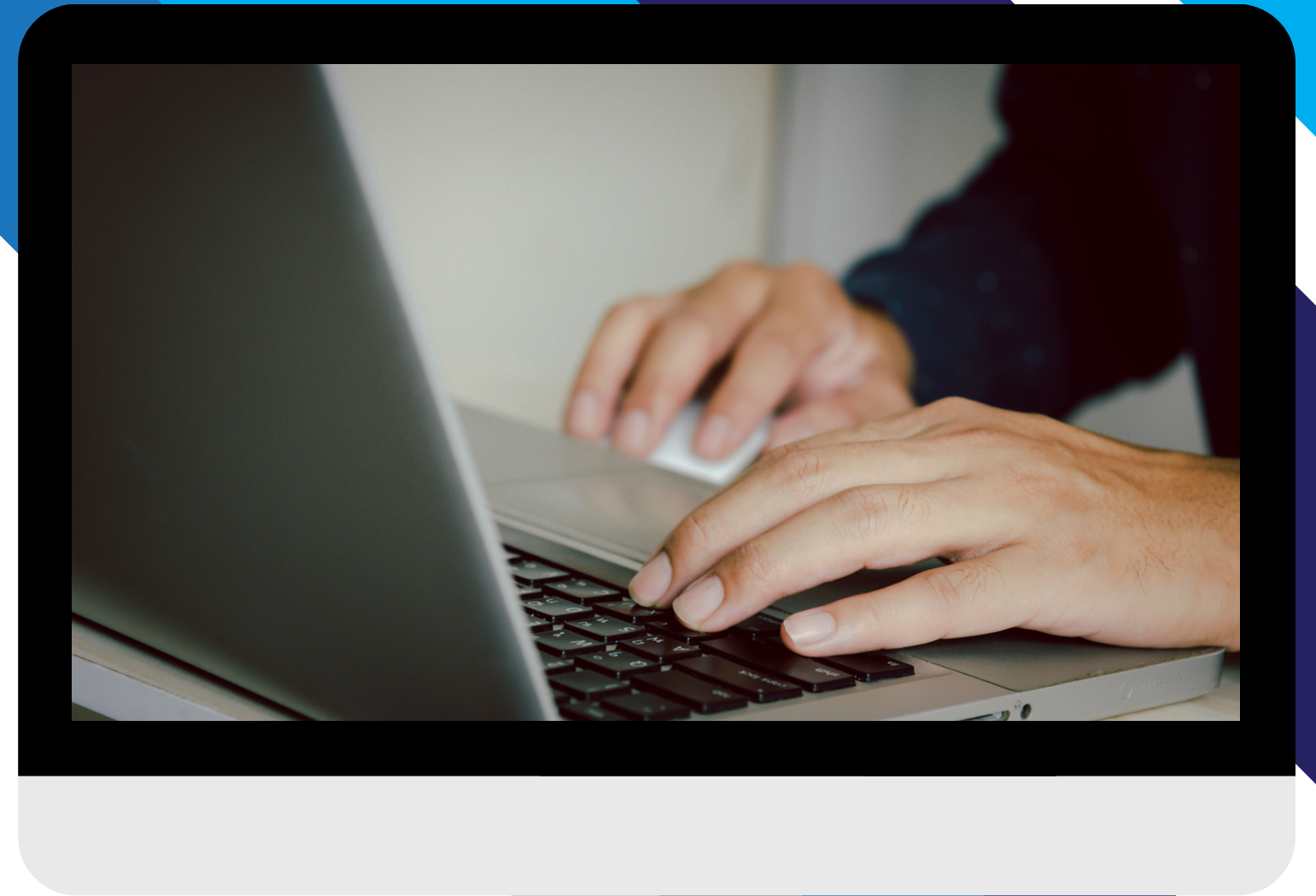
Challenges

- Choreograph syntax
- Limitation in time to learn the language
- Inconvenient time to work with the robot
- Difference in majors
- Merging the separate work of each subteam
- Difficulties in communication with the robot



Solution Challenges

- Split ourselves into groups of two in a way to facilitate the working process.
- Standardize the code
- Keep everyone involved in the changes.
- Working outside class hours
- Relied on dialogue chat (through typing) to ensure the acquisition of information
- Enhance our knowledge in qichat through YouTube videos and examples.





× × × ×



Future Work

- Access information from the internet
- Integrating AI into the robot for better communication with users.



Conclusion



× × × ×



Through this project, we implemented all the steps required to build a social robot, where it can communicate with users and provide a specific service.





Thank

you!

