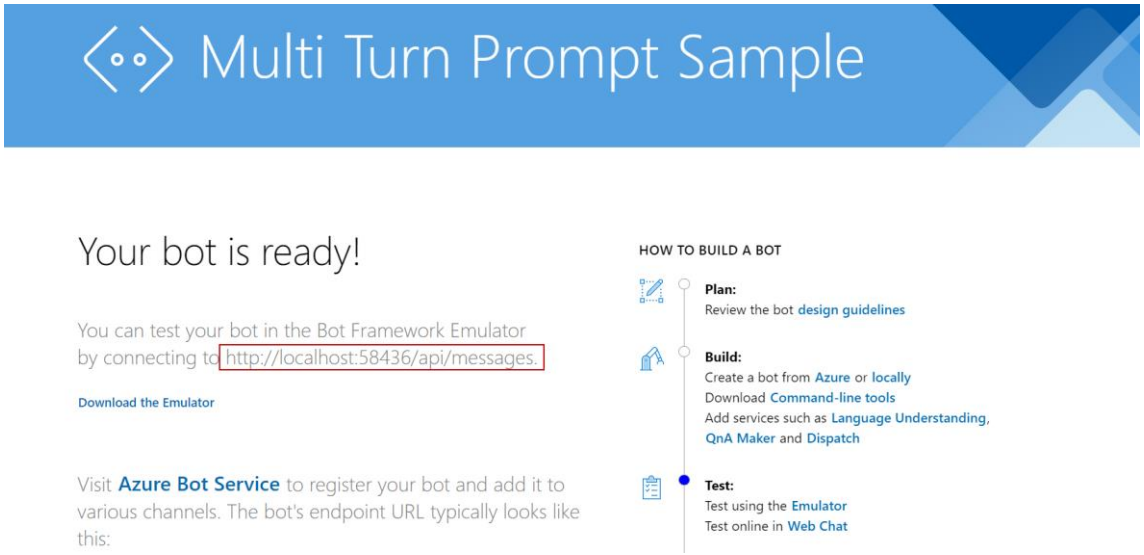
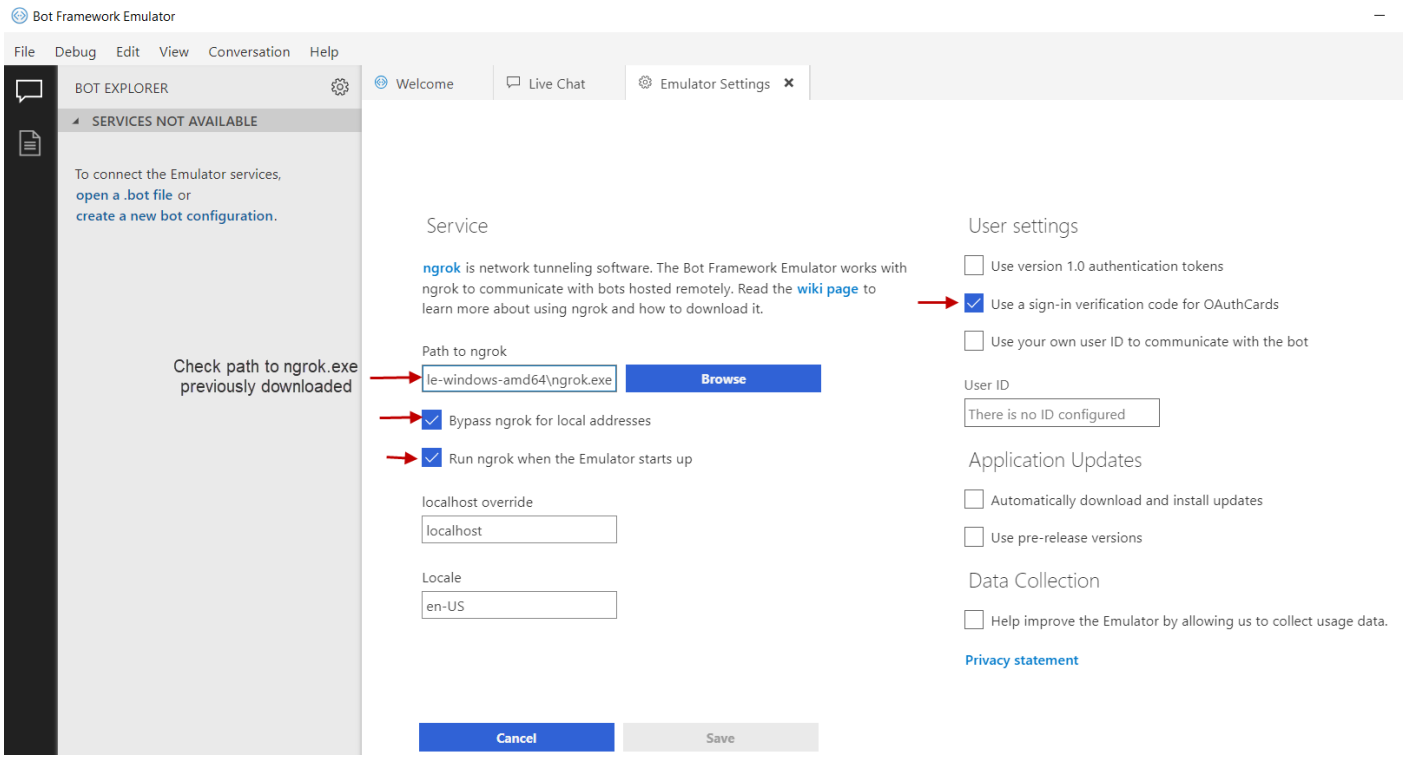
Test your bot locally.

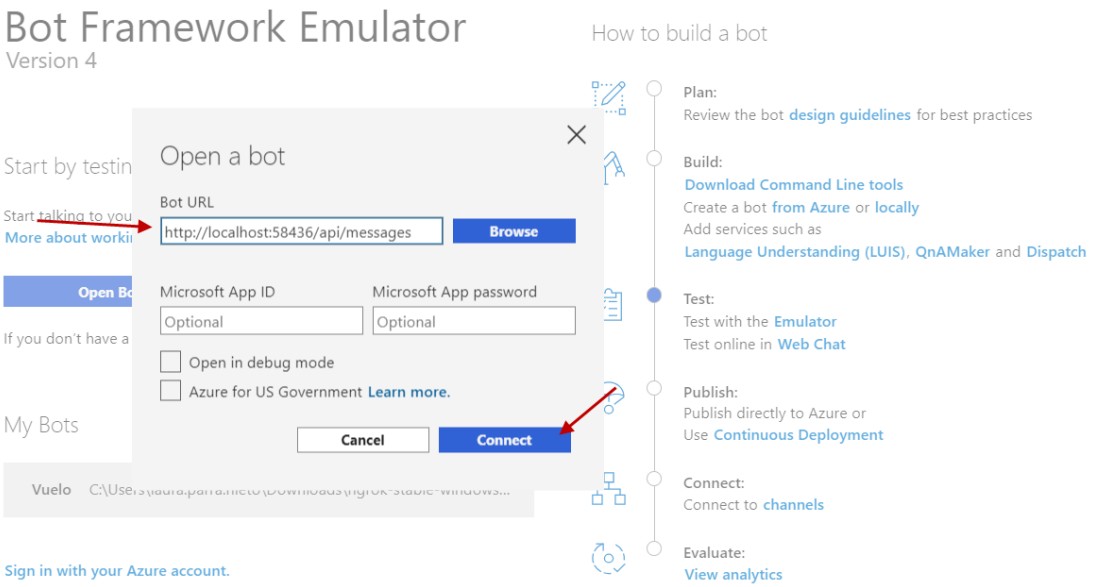
Step 1. Download or clone this repository, open this project with Visual Studio. Once, build solution and start debugging. New window (as below one) should be opened in your browser.



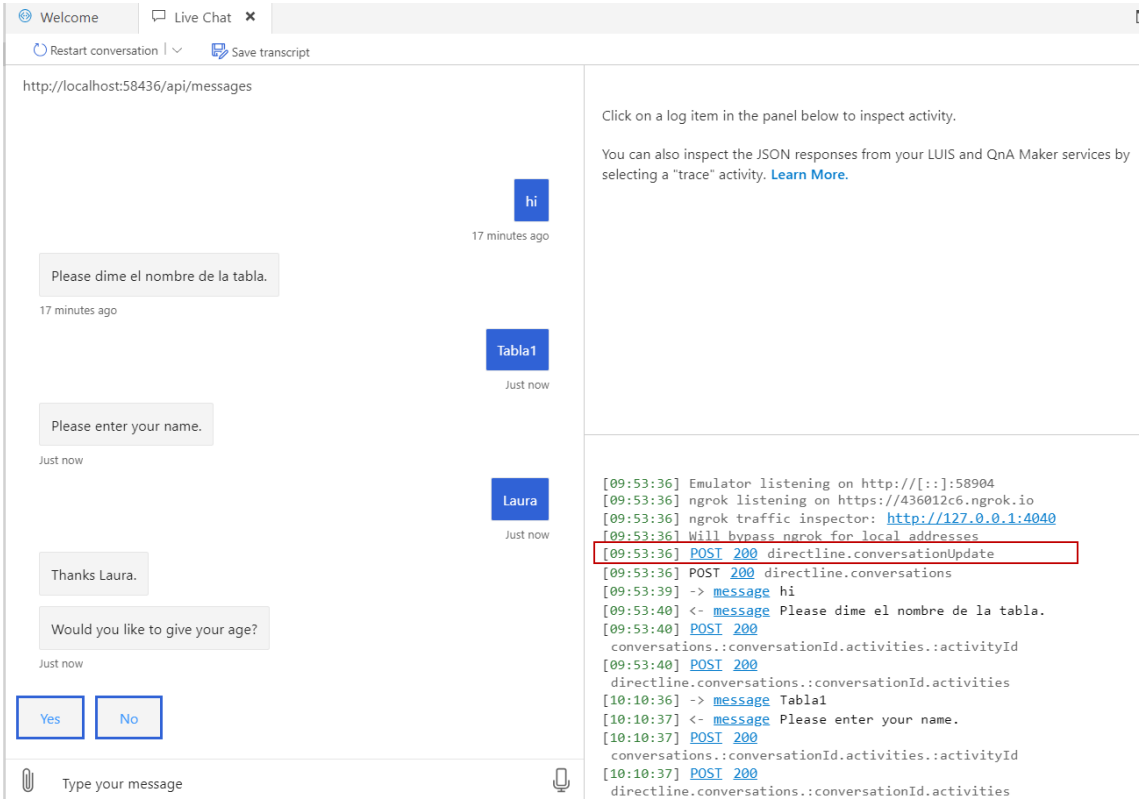
Step 2: Open Bot Framework Emulator and configure it. Find in next image how they should be marked. Before this step you should download ngrok (https://ngrok.com/download). Click on save.



Step3: Open your bot. Copy URL from the window of step 1, https:/localhostXXXX/api/message and click on “Connect”.



Once connection has been established, the message “POST 200 directline.conversationUpdate” should be showed. This message indicates that there is an activity in the conversation, then you can start to send messages.



Frequently errors:

Failed to spawn:

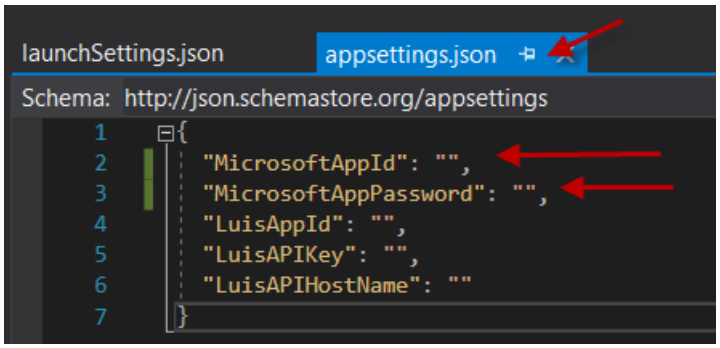
```
[10:28:49] Emulator listening on http://[::]:62100
[10:29:13] Failed to spawn
ngrok {"error_code":104,"status_code":503,"msg":"ngrok is not yet
ready to start tunnels","details":{"err":"a successful ngrok tunnel
session has not yet been established"}}
[10:29:13] POST 200 directline.conversationUpdate
[10:29:13] POST 200 directline.conversations
```

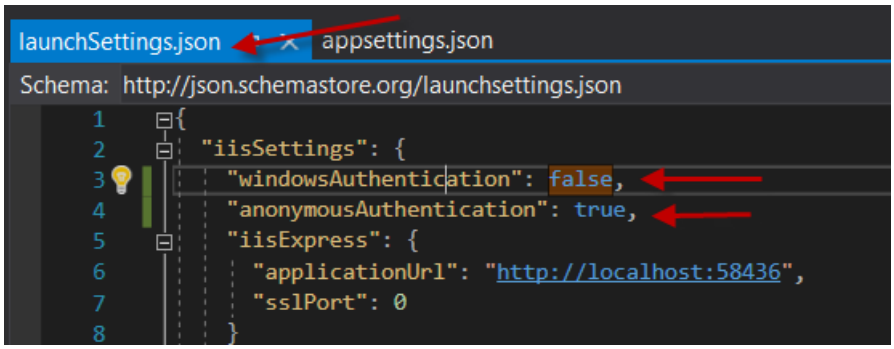
In our case it is not an actual problem, you can see below error message “POST 200 directline.conversationUpdate”. We are testing our bot locally. When you deploy your bot as Azure resource for example, you should create a new tunnel to your localhost to access it throw remote.

401. Cannot post activity. Unauthorized

```
[09:39:37] Emulator listening on http://[::]:57993
[09:39:37] ngrok listening on https://72d72312.ngrok.io
[09:39:37] ngrok traffic inspector: http://127.0.0.1:4040
[09:39:37] Will bypass ngrok for local addresses
[09:39:40]
[09:39:40] POST 401 directline.conversationUpdate
[09:39:40] POST 200 directline.conversations
[09:39:43] -> message hi
[09:39:45] Cannot post activity. Unauthorized.
[09:39:45] POST 401
 directline.conversations.:conversationId.activities
```

Solution: You should check launchsettings.json and appsettings.json files. Should be agreement between them. If there is no any Microsoft App Id and Password, Windows Authentication should be disabled and Anonymous Authentication available.




launchSettings.json
appsettings.json
Schema: http://json.schemastore.org/launchsettings.json
{
"iisSettings": {
"windowsAuthentication": false,
"anonymousAuthentication": true,
"iisExpress": {
"applicationUrl": "http://localhost:58436",
"sslPort": 0
}