# Rubicon Global Azure Bootcamp 2018

# LAB 3: Step 1 – Extend LUIS APP

Before we continue customizing the bot from LAB 2, we want to add more ‘intents’ to LUIS to broaden its understanding.

1. Go to Luis.ai
2. Click on Sign In
3. Sign in with the credentials from LAB 2 (if necessary)
4. Select the application (in our example “GAB2018BOT-<key>”)
5. Select Intents
6. Click [Create new intent]
7. Type “Authenticate”
8. Click [Done]
9. After the intent is created, you can start adding sentances to train it. Before we do this, we will first create the other intents for this LAB. Repeat the following and create intents named BrokenBoiler, ErrorCode and PlanDate:
   1. Select Intents in the left menu
   2. Click [Create new intent]
   3. Type the name
   4. Click [Done]
10. Select Intents in the left menu
11. You should see 4 intents.

Before we continue, we want to add some prebuilt Entities to our App before we start adding sentinces to train it.

1. Select Entities in the left menu
2. Click [Manage prebuilt entities]
3. Check the following entities:
   1. Number
   2. Email
   3. Date

Let’s continue training our app

1. Select Intents in the left menu
2. Click on “Authenticate”
3. Type the following sentence in the first input field:

*“My email is* [*test@test.com*](mailto:test@test.com)*”*

1. LUIS should have automatically detected the email, and replaced the sentence with “My email is **email**”
2. Type the following sentence in the first input field:

“test@test.com”

1. Type the following sentence in the first input field:

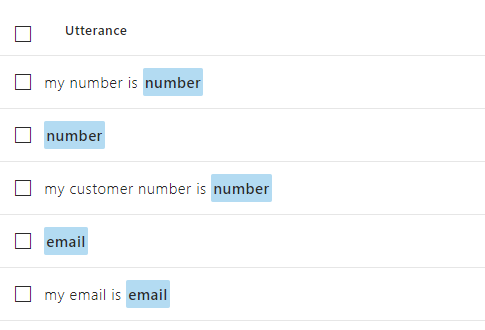
*“My customer number is 1234567”*

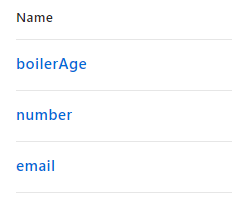
1. LUIS should have automatically detected the number, and replaced the sentence with “My customer number is **number**”
2. Type the following sentence in the first input field:

“1234567”

1. Click on “1234567”
2. Select ‘Browse prebuilt entities’ in de dropdown.
3. Search for number
4. Select number
5. Type the following sentences in the first input field:

*“My number is 1234567”*

1. You should have 5 sentences looking like this:  
   
2. And 3 entities (variables) like this:



1. We are ready with this intent.

Let’s add another one…

1. Select Intents in the left menu
2. You should see 4 intents.
3. Click on “BrokenBoiler”
4. We are not going to add any entities here, so just type the following 5 sentence in the first input field:

“My boiler is broken”

“I have no heat”

“I have no hot water”

“It is broken”

“It does not work”

(Add any sentences that you think can help identify how people would tell you their boiler is broken. The more examples LUIS has, the better it can predict when people mean this.)

1. Select Intents in the left menu
2. You should see 4 intents.
3. Click on “ErrorCode”
4. We are going to introduce only one new entity here, called “ErrorCode”. Add the following sentences and replace, E01, E10, E120, E1 and E4 with the entity “ErrorCode”. Note, with the first sentence you need to create the entity first. It is a **simple** entity.

“It says E01”

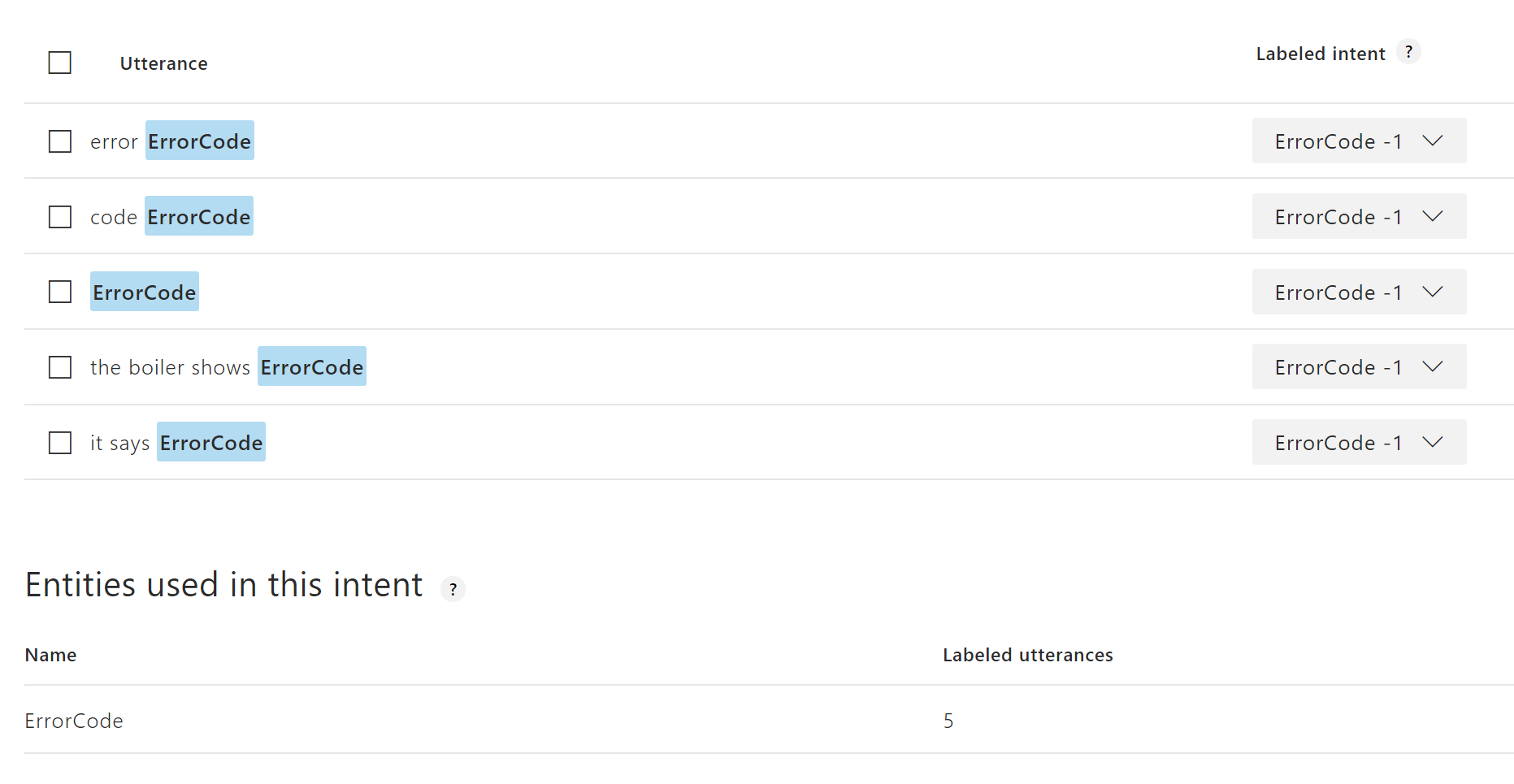
“The boiler shows E10”

“E120”

“code E1”

“error E4”

1. Your intent should look like this:



The last intent is PlanDate. Use the following sentences to train LUIS. We are going to use a built-in entity (just like the email).

1. Select Intents in the left menu
2. You should see 4 intents.
3. Click on “PlanDate”
4. Type the following sentence in the first input field:

*“I am home next Thursday”*

1. Luis should have detected that it is a date and replaced Thursday with **datetimeV2**
2. Now add the following sentences:

“next Wednesday”

“next week on Monday”

“November 13th”

“end of the week”

1. The last 4 should all be changed to **datetimeV2**

We can now train our app.

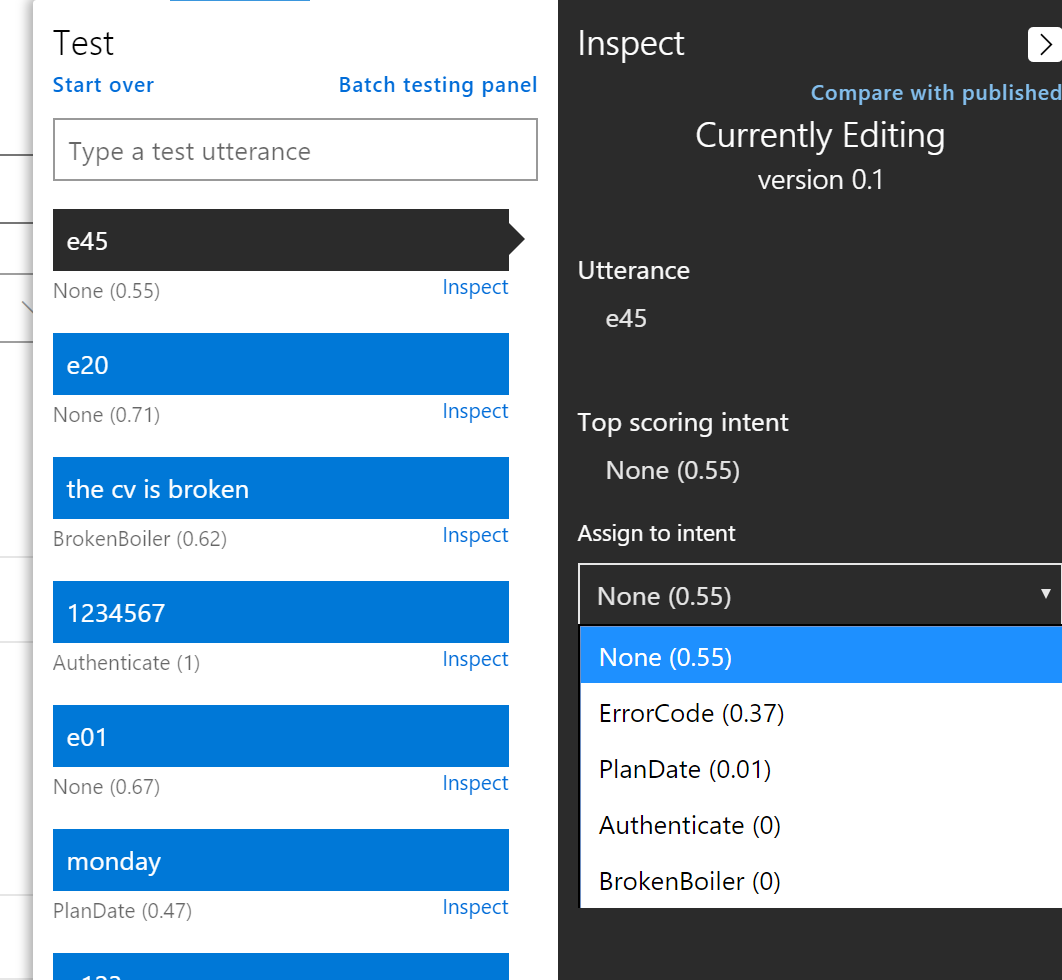
1. Click [Train] on the top menu
2. After training is done.
3. Click on [Publish] on top menu
4. Select Production in the Publish To option.
5. Click [ Publish to production slot]

You can now test your app

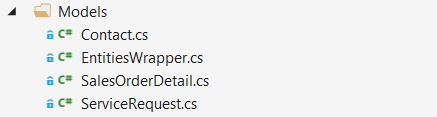
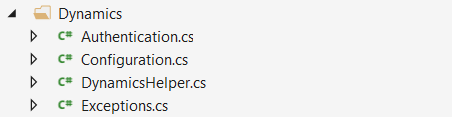
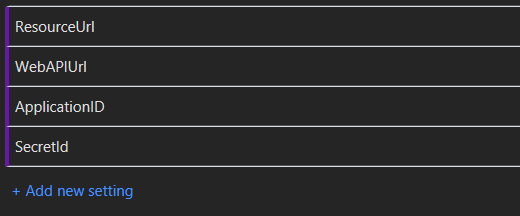
1. Click on [Test] in the top menu.
2. Type a sentence (different from the once you used to train)

NOTE:

* Typing an error code like “E345” can result in a none intent. If you click on the sentence, you can inspect the answer and assign it to a new intent and retrain your app. This enables you to finetune your app. In the example below “E45” is 55% undefined and “37%” ErrorCode intent.



# LAB 3: Step 2 - Connect to Microsoft Dynamics CRM

1. Open the bot solution from LAB 2 in Visual Studio
2. Go to Tools - NuGet Package Manager - Manage NuGet Packages for Solution...
3. Go to Browse and search for package Microsoft.IdentityModel.Clients.ActiveDirectory
4. Install the latest version of the package to the SimpleEchoBot project
5. Create a new folder “Models” under the SimpleEchoBot project and add the 4 classes from Lab3/Code/Models:  
     
   
6. Create a new folder “Dynamics” under the SimpleEchoBot project and add the 4 classes from Lab3/Code/Dynamics:  
     
   
7. Log in to the Azure portal and navigate to your web bot's App Service component.
8. Go to the Application settings blade under Settings
9. Scroll down to Application settings, and use Add new setting to add the following 4 settings. The values will be provided on site by Rubicon:  
     
   

# LAB 3: Step 3 - Add dialogs that interact with CRM

## Identify Customer dialog

1. Add a new dialog to your bot to identify the customer.
2. Have the dialog ask for the users e-mail address.
3. Create a new CRM request uri string with the email address returned by LUIS  
     
   
4. Use the Dynamics helper to execute the request to the Dynamics CRM backend:  
     
     
     
   You may need to add a reference to Newtonsoft.Json and using Newtonsoft.Json.Linq  
     
   **HINT**: you can test you Identify Customer dialog using the following e-mail addresses:  
   john.doe@example.com, jane.doe@example.com and jan.jansen@example.com

## Identify Boiler dialog

1. Add another dialog that will query CRM orders for boilers sold to the customer
2. If multiple boilers are found, ask the user to select the correct boiler  
     
   **HINT**: John Doe has 0 boilers, Jane Doe has 1 boiler and Jan Jansen has 2 boilers

## Make Appointment dialog

1. Add yet another dialog that will ask the customer for a timeslot to make a service appointment.
2. After getting the available time, create a query to make an appointment within CRM.