# Angular Best Friends

## Module 4 Demo 2 – Creating and using an event bus

## Goals

The goal of this demo is to show the basic building blocks of an event bus service.

## Steps

1. Open the folder “module4demo2” in VsCode
2. From the terminal run **npm install** if running for the first time.
3. Run the app with **ng serve –open** just to make sure it is working
4. Go to “**core > services > event-bus.service.ts**”
5. Here there are a lot of things to be explained.
6. Make sure to show and outline that the event bus service is based on a **subject**. The subject is private since we don’t want to expose the entire subject to other components.
7. Then show the “**on**” function. Using this function other components will be able to subscribe to the event bus.
8. This function takes in an event type (in our case it is an enum. Could be however a simple string or whatever) and an action, which is actually a callback function.
9. Another interesting thing here is that in this function we’re basically subscribing to ourselves. Analogy, I’ll give you a phone and once something happens I’ll call you on that phone.
10. Then we use a pipe on the subject. This could be useful if the event bus handles several types of events. Then we could use pipes to filter out different things.
11. Then go to the **emit()** function. This function will be called by components that want to raise events.
12. This will emit an event of type **EmitEvent**. Show the class and notice that the value is optional.
13. Now let’s see how components emit events. For this, go to the “**customers-list.component.ts**” file.
14. Show that we’re injecting the event bus
15. Then show the **selectCustomer()** function. This function is triggered once a specific customer is selected in the list. Now, here we do two things. First we emit an event using the EventEmitter on an output property. But then, we emit a new EmitEvent object to the event bus.
16. Now from anywhere else in the app I would be able to now subscribe and get these events and eventually do something with them.
17. Let’s take “**app.component.ts**” as example.
18. Show that we inject the event bus in this component.
19. Also show that we subscribe to it using the “**on()**” function that we discussed earlier.
20. Go to the browser and show the effect of this. On the “Communication” menu item, go ahead and select one customer in the list.
21. We’ll notice that a customer information is displayed in the top right. Notice here that we have two totally different components that communicate through an event bus.