

## **Depot/U**

### *Supplemental Exercise 4*

### **Contact Keeper**

In this exercise, you will build an angular application consisting of two views. Routing will be necessary for this exercise. The first view is a welcome screen welcoming the user to the Contact Keeper. It also contains input boxes for the user to type in a name, email, and phone number, and an appropriately labeled button that will cause the data typed into the boxes to be added to an array of “contacts.” This array of contacts should also be shown on the first view. In short, the main page of this application will be a home page consisting of all your contacts (showing their name, email, and phone. You can use dummy data). The main page will also have a form where you collect this data, and a button that should add a contact record to the array containing the data typed in the form.

The second view will should display just the contact info for one contact. In other words, the welcome view should have a button that navigates to the second view, and passes some type of identifier as a route parameter. The identifier that you choose is up to you. It can be an array index, the email address, etc.

#### **Notes**

- For this to work, you will most likely have to store your data somewhere else, such as an Angular factory/service or a Platypus repository.
- In that “somewhere else”, you will need functions to get all contacts, as well as a function to get just one contact by whatever identifier you decide to use.

If you liked this exercise, complete it again using PlatypusTS.