Lab 3b: Binding services with manifest files

Description: This is the same lab as “Lab 3a: Services” but it uses manifest files to bind the application to the desired service.

Note that commands are in *italics* and the X should be replaced with a unique number or participant’s initials.

1. Using the same application and services you already have created on the previous lab, check the manifest file at the “manifest” directory of labs\_Java/artifcts.

*$ more manifest.yml*

*---*

*applications:*

*- name: pcfdemo*

*memory: 512M*

*instances: 2*

*host: pcfdemo*

*path: ../pcfdemo-us.war*

*services:*

*- rabbit*

Update the manifest file using your text editor of choice to reflect the application

name you want, number of instances desired, path of the war file, hostname, etc.

The manifest for the app being pushed by user 10 could look like:

*---*

*applications:*

*- name: map-userjfo*

*memory: 512M*

*instances: 2*

*host: mymap-userjfo*

*path: ../pcfdemo-us.war*

*services:*

*- myRabbit-userjfo*

The manifest above assumes you’ve already created the service

“myRabbit-userjfo”.

1. Now from the manifest file directory, just execute the cf push command:

*cf push*

The CloudFoundry CLI should verify the manifest file and re-push the app

using the options provided and bind to the service specified.

1. Check your application is running fine.

Login to the Web Console and verify the App health or just use the command line:

*cf app map-userX*

Acccess the application URL and verify it was deployed and bound to the service specified.