**PROCEDIMIENTO PARA SUBIR LA PALICACION BACK-END A LOS SERVIDORES DE HEROKU**

1. En una consola de comandos, situarse en la carpeta raíz del proyecto p.e(C:\Users\Administrador\Documents\workspaces\Eclipse-workspace\SpringMicro)
2. Asegurarse de tener instado heroku cli para que el usuario pueda ser reconocidos desde línea de comandos, escribir en la línea comandos heroku login ***(credenciales: usuario:*** [***mecano\_mcs@hotmail.com***](mailto:mecano_mcs@hotmail.com)***; password: mecano007)*** se mostrara usuario logeado.
3. Versionar el proyecto con ***git***, esto generara una carpeta oculta ***.git***; Escribir en la consola de comandos ***git init***
4. En la raíz del proyecto se encuentra el archivo ***.gitignore*** (creado por spring boot – de lo contrario crearlo) el cual contiene los archivos que serán ignorado cuando git envíe todos los chivos de la aplicación a su servidor spring
5. Antes de subir al archivo heroku requiere un archivo de nombre Procfile sin extensión situado en la raíz del proyecto, el cual indica donde va a estar la ruta del archivo .jar una vez que se compile en maven el cual contiene únicamente la línea siguiente: ***web: java -Dserver.port=$PORT -jar target/SpringMicro-0.0.1-SNAPSHOT.jar***
6. Cumplidos los dos puntos anteriores, en la consola de comando escribir lo siguiente para poner todos los archivos del proyecto en un espacio temporal  ***git add .***
7. Comprobamos el estatus de los archivos a enviar al servidor de heroku con el comando  ***git status***
8. Para confirmar los cambios que se enviaran en la line de comandos escribir: ***git commit –m “Primer commit”***
9. Comprobamos el estatus de los archivos en local que serán enviados al servidor de heroku con el comando  ***git status***
10. Crear un proyecto en los servicios de heroku para referenciar el código que se encuentra en local al proyecto en heroku, escribir en la línea de comandos ***heroku apps:create micro-mito*** damos enter y se conectara con los servicios de heroku para crear el espacio del proyecto proporcionando una ruta. Nota: El nombre tiene que ser único en el mundo.
11. Para enviar el código del proyecto local a la rama principal del proyecto creado el paso anterior, en la línea de comando escribimos ***git push heroku master***
12. Si tenemos un estatus de Build success, significa que el proyecto ya está en heroku
13. Para ejecutar el código del proyecto en heroku en la línea de comando escribimos heroku open
14. Testeamos el proyecto de internet Explorer en la ruta: <https://micro-mito.herokuapp.com/persona/leer/1>, obtenemos la cadena de json: {"id": null,"nombre\_completo":"null","correo":"null","pais":"null","bio":"null","fotoURL":"null"}, los resultados null es porque aunque la base de datos se crea cuando se levanta la aplicación no existen datos todavía; Pero el proyecto está funcionando.
15. Por default heroku trabaja con Postgre, aunque se pueden otras bases de datos pero tienen un costo. Para suministrar datos a la base de datos en la línea de comandos escribimos y la cerramos con Ctr+c: ***heroku pg:psql***
16. El nombre de la base de dados la proporciona heroku, además de una línea de comandos para ingresar scripts de base de datos, para pegar dichos scripts y poblar la base de datos; Hecho esto verificamos desde la url de heroku de la siguiente forma: <https://micro-mito.herokuapp.com/persona/leer/1> y debería devolver ***{"id":2,"nombre\_completo":"Martin Cano","correo":"mecano\_mcs@hotmail.com","pais":"Mexico","bio":"JAVA FULL STACK","fotoURL":"C:\\Users\\Administrador\\Pictures\\VARIOS\\shrek 1.png"}***

**PROCEDIMIENTO PARA SUBIR LA PALICACION FRONT-END A LOS SERVIDORES DE HEROKU**

1. Para que heroku detecte que esta es una aplicación en HTML5, agregamos un archivo index.php que contiene solo una línea ***<?php header( 'Location: /index.html' ) ; ?>***
2. Se procede a hacer la modificación del front-end porque actualmente utiliza localhost:8080 y debería invocar a la url del proyecto en heroku. Por lo tanto todo lo que diga localhost:8080 en el front-end deberá ser remplazado por micro-mito.herokuapp.com; por lo tanto si ejecutamos el index.html del front-end mostrara los datos de la base de datos desde heroku.
3. Asegurarse de estar logeado a heroku desde la consola de comandos y situarse en la carpeta raíz del proyecto p.e(C:\Users\Administrador\Documents\CURSOS\Curso Spring Developer (On Line)\codigo\Codigo casa\dia 7\Spring-micro-ui) y ejecutar el comando git init
4. Ejecutar el comando ***git add .***
5. En este caso no hay archivos que se van a ignorar, ejecutar el siguiente comando ***git commit –m “Primer commit ui”***
6. Verificar si hay algo por confirmar, ejecutando el siguiente comando: ***git status***
7. Ejecutamos el comando: ***heroku apps:create micro-mito-ui***
8. Ejecutamos el comando para enviar el código: ***git push heroku master***
9. El paso anterior nos regresa una url: ***https: //git.heroku.com/micro-mito-ui.git***
10. Para abrir y probar la aplicación front-end desde un internet Explorer ejecutamos el siguiente comando: ***heroku open***
11. Si tratamos de ejecutar el comando heroku pg:psql nos indicara que no hay una base de datos asociada porque se trata de una aplicación de tipo html5.
12. **Si el código del front-end o back-end cambian, solo tenemos que hacer lo siguiente, desde la carpeta raíz del proyecto en cuestión, ejecutar el comando (git add .), después (git commit –m “nuevo cambio”) finalmente (git push heroku master).**