# Practica de Front End

Iván Castañeda

Para iniciar el proyecto se requieren los paquetes

* flask
* flask\_sqlalchemy
* ninja2

en caso de no contar con ellos, se deben instalar.



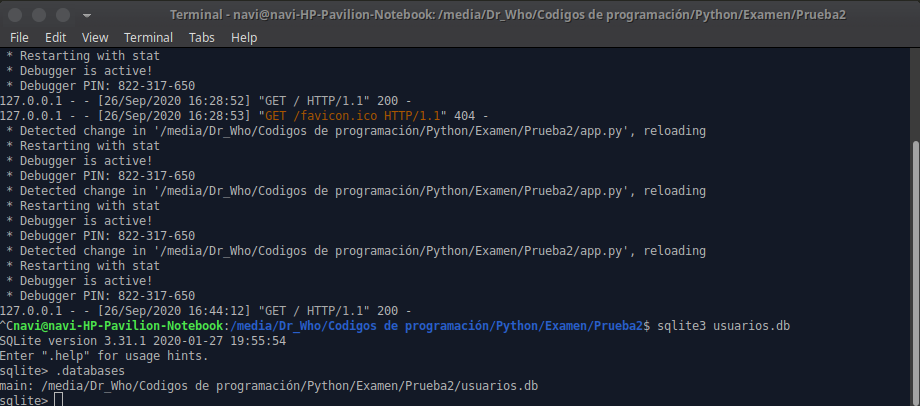
La interfaz uno de la pagina web contara con 5 @app.route()

* /
* /add\_contact
* */*detalle<id>
* /delete<id>
* /back

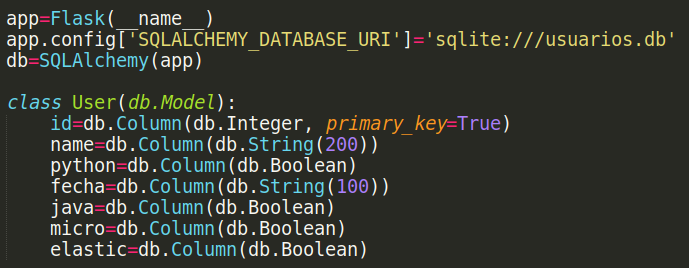
Se crea una base de datos en sqlite3 la cual tendra en total 6 columnas

* Nombre
* Fecha
* Python
* Java
* Elastic
* Microservicios

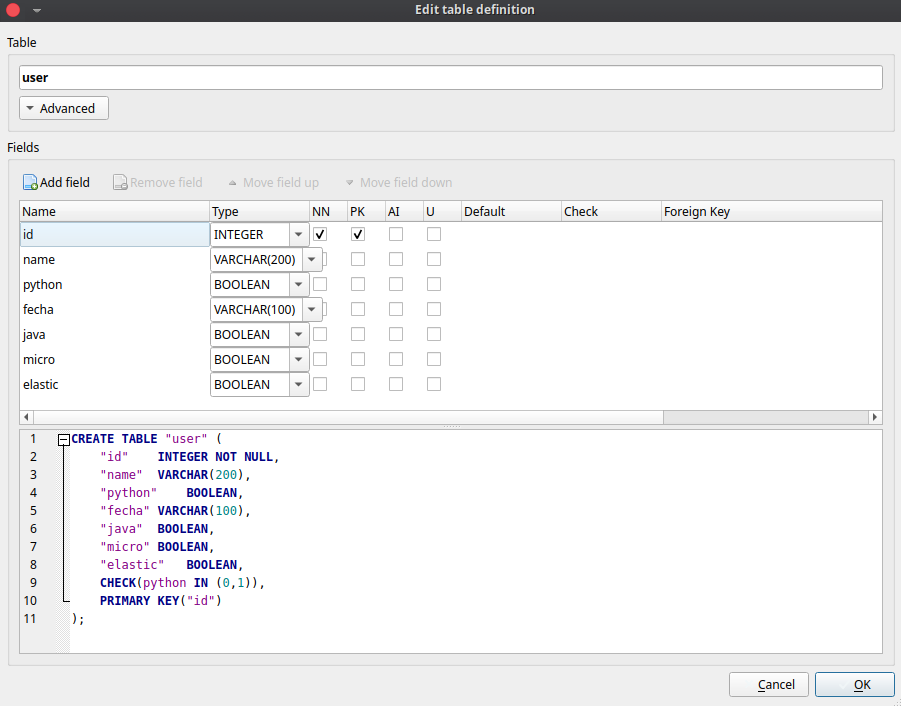
Para crear la base de datos, en la terminal se escribe sqlite3 usuarios.db y seguidamente se escribe .databases



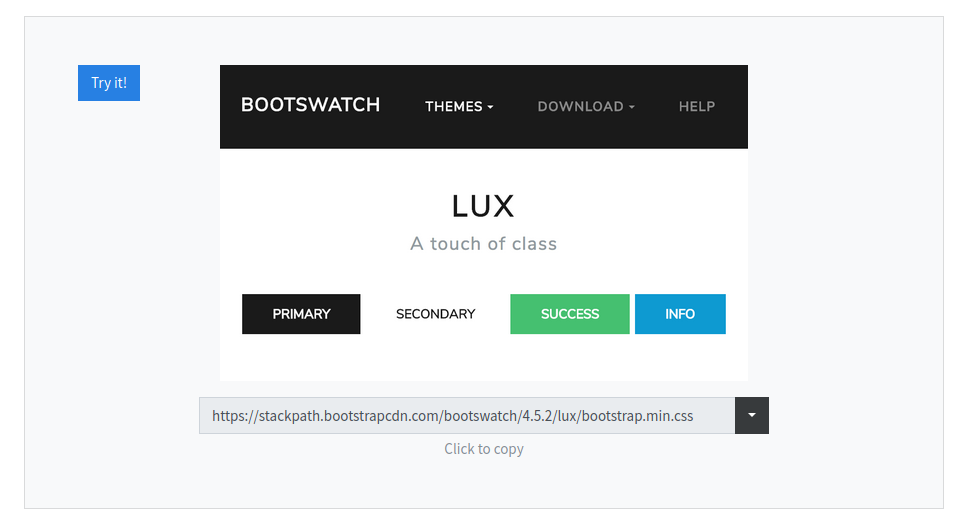
Una vez creada la base de datos, se crean las columnas con las siguientes lineas de codigo



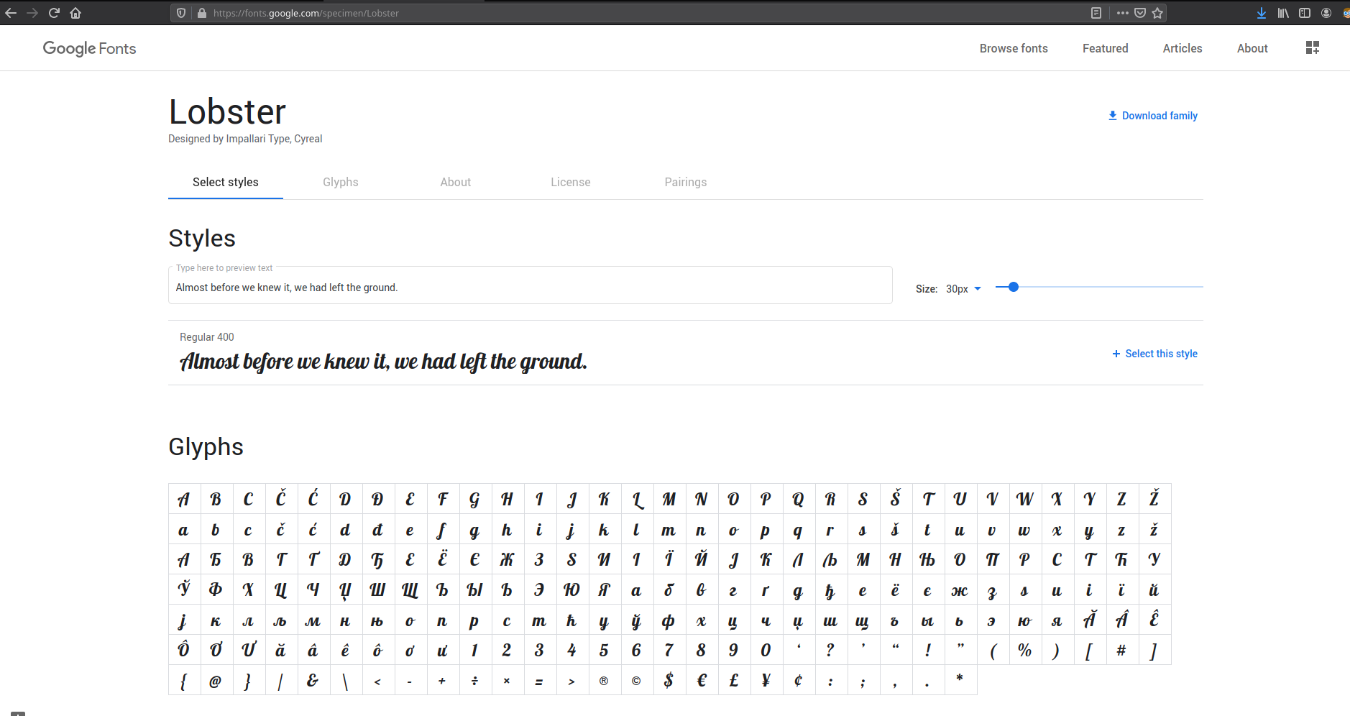
El tipo de valor de cada columna se verifica con DB Browser



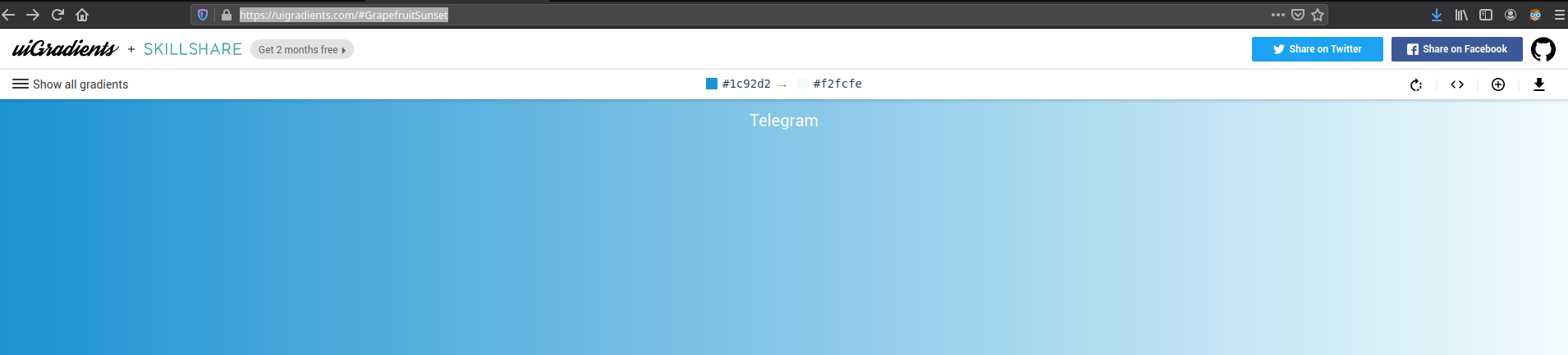
Se crea un archivo html para el diseño de la pagina web, el archivo tendra nombre de index.html, para la parte decorativa se utilizara se crea un archivo main.css para las etiquetas de html. De la pagina <https://www.bootstrapcdn.com/bootswatch/> se obtiene el diseño para el css

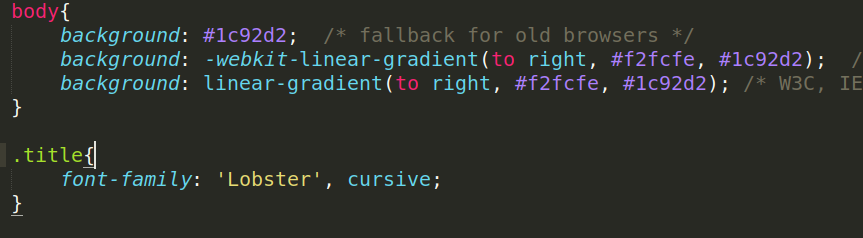


En la fuente tipografica para el titulo se utiliza lobster de google fonts



Por ultimo el fondo de la pagina se utilizara un degradado telegram de la pagina <https://uigradients.com/#GrapefruitSunset>



El archivo main.css queda de la forma

El archivo index.html tendra la siguiente estructura

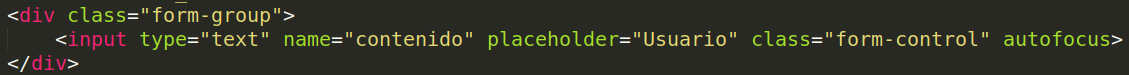
Titulo de la pagina “Usuarios”

Label centrado “Usuarios registrados”

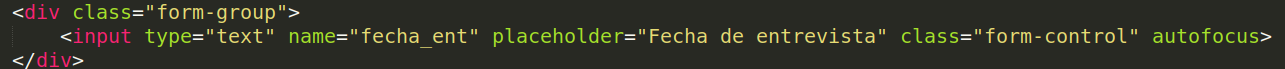


En el card header se tendra la información

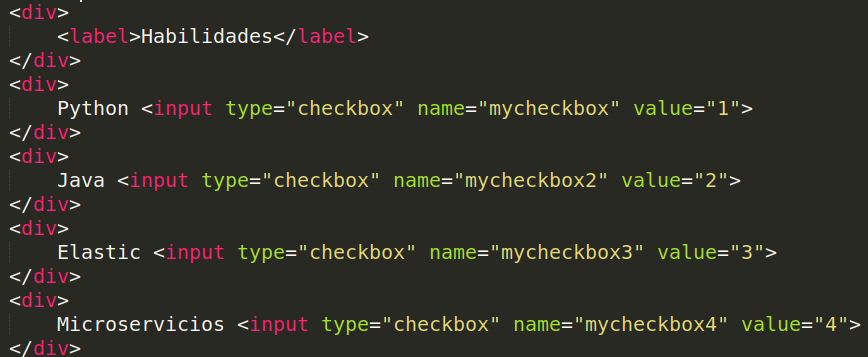
Nombre input



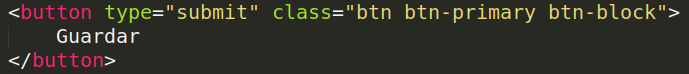
Fecha input



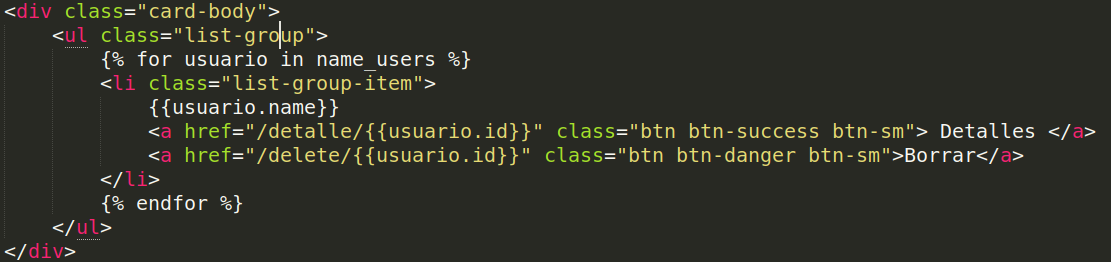
4 checkbox que contemplan las habilidades Java, Python, Elastic, Microservicios



Un botón para guardar la información



En el card-body se tendrán los usuarios registrados en forma de lista que se ira actualizando y cada uno de ellos tendra 2 botones en los cuales se podra acceder a su información o eliminarlo, la consulta se hara en base al id

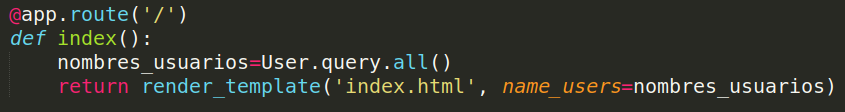


La pagina tiene el siguiente estilo visual

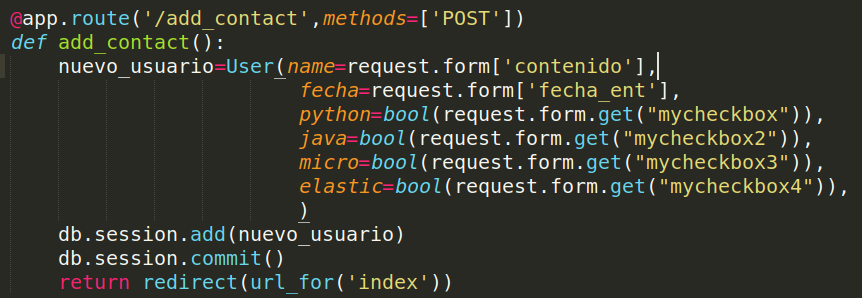


Antes de añadir un usuario se van a crear las app.route()

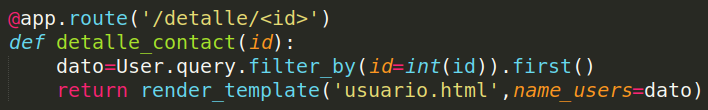
La app.route(/) va a mostrar la información de la base de datos en la pagína principal (index.html) y regresará una variable con la información de los usuarios



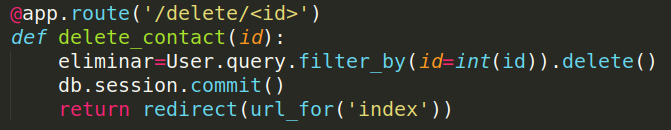
La app.route(/add\_contact) va a tener el método “POST” en la cuál recibiáa la información del usuario que se almacenaáa en la base de datos, al terminar la petición regresará a la pagína principal



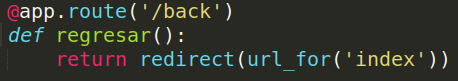
La @app.route(/detalle/<id>) llevará a otro html en donde se podrá consultar la información del usuario respecto a su id, esta redirecciona a un html llamado usuario, del cuál se hablará más adelante



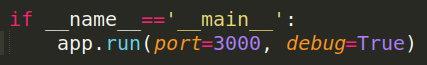
La @app.route(/delete/<id>) eliminará de la base de datos la información del usuario



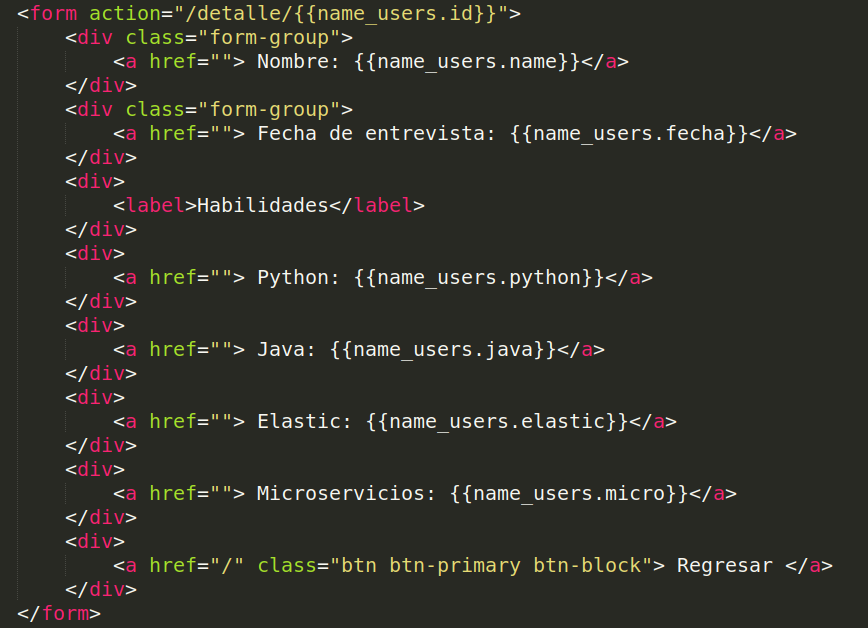
La @app.route(/back) redirecciona a la pagína principal, se úbica en la pagína del archivo usuarios.html



Por último la parte final hace referencia al puerto en dónde se ejecutaáa la pagina web, se habilito la opción debug, para visualizar los cambios en todo momento



El archivo usuarios.html, es una copia del index.html, solo que en esta se mostrará la información del usuario recibiendo como argumento el valor del id y esta construida de la siguiente forma



al registrar un usuario la pagína se observa de la siguiente manera



Al consultar la información del usuario, la pagína se observa de la siguiente forma



Al regresar a la pagina inicial se visulazi de la forma



Para subir la pagina web a para su consulta abierta se utiliso AWS en el servicio EC2 creando una maquina con ubuntu, en ella se utilize un ssh para acceder y hacer una copia de los elementos de trabajo, se instaló todo lo necesario para que su funcionamiento, por ultimo el link de acceso es el siguiente:

ec2-18-216-43-84.us-east-2.compute.amazonaws.com:3000

