

Manual técnico

1. Paquetes y librerías

En el desarrollo del software se utilizó el lenguaje de programación Python, el framework de desarrollo web Flask y una base de datos en MongoDB. Las librerías y paquetes utilizadas se listan a continuación:

- **Python3:** Python es un lenguaje de programación interpretado. Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, dinámico y multiplataforma.
- **Pip3:** pip es el instalador del paquete para Python. Puede usar pip para instalar paquetes del índice de paquetes de Python y otros índices.
- **Build-essential:** El paquete build-essential es una referencia para todos los paquetes necesarios para compilar un paquete Debian. Por lo general, incluye los compiladores y bibliotecas GCC / g ++ y algunas otras utilidades.
- **Librerías de servicios:** Estas librerías permiten a Python la encriptación de las contraseñas de los usuarios, el manejo de archivos PNG para la creación y lectura de códigos QR, la conexión a la base de datos de Mongo, el manejo de DNSs y la simulación de un servidor SMTP para el envío de correos electrónicos.
 - Flask-bcrypt
 - Pypng
 - Pyqrcode
 - Pillow
 - Pyzbar
 - Zbar-py
 - Pymongo
 - Dnspython
 - Smtplib
 - Uwsgi

2. Archivos fuente

Los archivos fuente que permiten el funcionamiento del sistema son:

- **Python:**
 - app.py: Todo el aplicativo web. Hecho en uso del framework Flask. Realiza la conexión con la base de datos. Administra toda la lógica tras las distintas operaciones realizadas dependiendo de las solicitudes de cada usuario.
 - mensajes.py: se definen previamente los mensajes para las respuestas automatizadas que se darán al momento de la respuesta de un administrador frente a una solicitud.
- **JavaScript:**
 - script_mi_cuenta.js: script que implementa distintas funciones usadas para la correcta funcionalidad del apartado "mi cuenta".
 - script_recuperacion_contraseña.js: script que implementa distintas funciones usadas para la correcta funcionalidad del apartado "recuperar contraseña".
 - script_registros.js: script que implementa distintas funciones usadas para la correcta funcionalidad de los apartados de registro para cada uno de los distintos tipos de usuario.
- **HTML:** Los archivos Html listados a continuación componen el apartado visual del software y su interfaz, visualizable en un navegador de internet:
 - templates/admin.html
 - templates/admin_cuenta.html
 - templates/admin_gestionarAdmins.html
 - templates/admin_gestionarBarrios.html
 - templates/admin_gestionarCategorias.html
 - templates/admin_gestionarEntidadesSanitarias.html
 - templates/admin_gestionarLocales.html

- templates/admin_gestionarSolicitudes.html
- templates/admin_gestionarSolicitudes_modificacion_entidadesSanitarias.html
- templates/admin_gestionarSolicitudes_modificacion_locales.html
- templates/admin_gestionarSolicitudes_modificacion_usuarios.html
- templates/admin_gestionarSolicitudes_registro_entidadesSanitarias.html
- templates/admin_gestionarSolicitudes_registro_locales.html
- templates/admin_gestionarSolicitudes_registro_usuarios.html
- templates/admin_gestionarUsuarios.html
- templates/admin_gestionarVisitas.html
- templates/contactanos.html
- templates/entidad_sanitaria.html
- templates/entidad_sanitaria_cuenta.html
- templates/entidad_sanitaria_registro.html
- templates/entidad_sanitaria_usuarios.html
- templates/index.html
- templates/inicio_de_sesion.html
- templates/local.html
- templates/local_QR.html
- templates/local_cuenta.html
- templates/local_no_QR.html
- templates/local_no_QR_des.html
- templates/local_registro.html
- templates/objetivo.html
- templates/recuperar_contrasena.html
- templates/usuario.html
- templates/usuario_cuenta.html
- templates/usuario_registro.html