

Instituto Tecnológico de Costa Rica

Documento del PoC

Prueba de concepto, proyecto semestral

Alumnos:

Josué Toruño
Alejandro Rosales

Aseguramiento de la Calidad del Software:

M. Sc. Saúl Calderón Ramírez

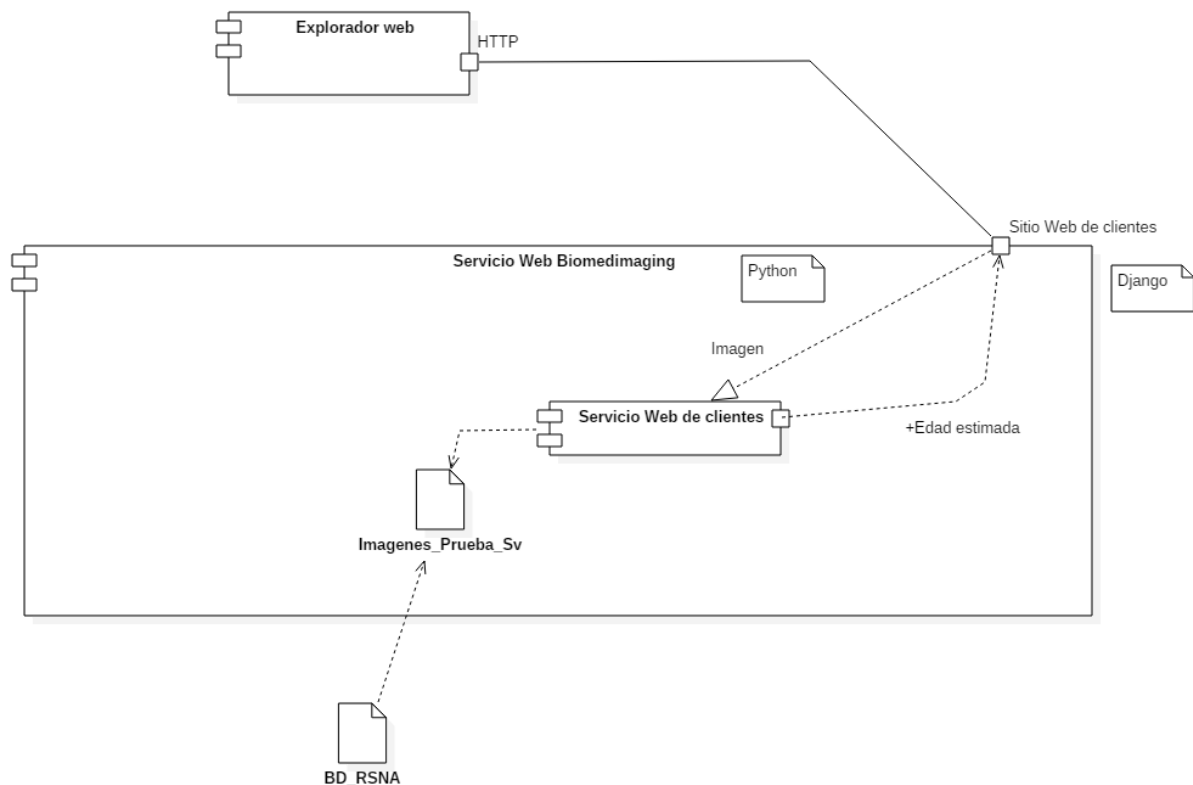
Fecha de Entrega:
05/03/2018

Tabla de Contenidos

Diagrama de Componentes	2
Herramientas Utilizadas	3
Trozos de Código	3
Justificación de Trozos	7
Problemas:	8

Diagrama de Componentes

A continuación se muestra el diagrama de componentes en alto nivel utilizado en este proyecto.



Donde se muestran los componentes principales, en este caso el explorador web, donde el usuario interactúa con el sistema, y por otra parte el servicio web, que contiene las distintas funcionalidades para poder funcionar de manera óptima y lograr obtener como resultado la edad estimada a partir del análisis de la muestra. Se utilizaron las tecnologías de Django y Python para interactuar entre lo que es el frontend y backend respectivamente.

Herramientas Utilizadas

Se está utilizando como herramienta de control de versiones Github, el proyecto se encuentra en <https://github.com/Odin2/poc-ACS>.

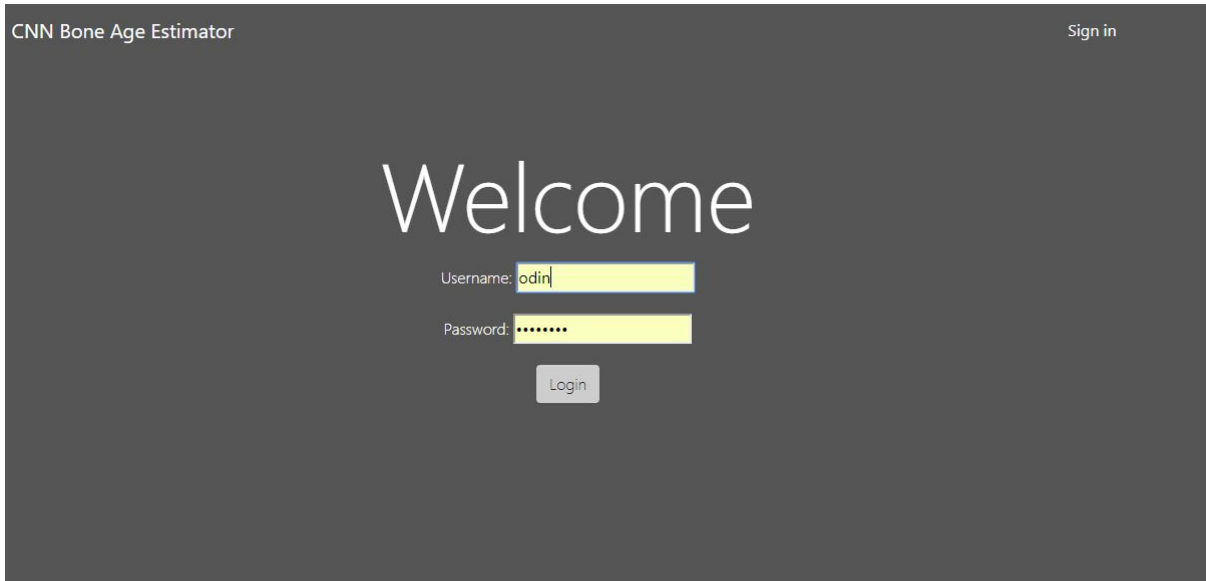
Como entorno de desarrollo se usa Eclipse, incorporando en él Github, para el control de versiones, así como Python y Django en el caso de backend y frontend. Para la realización del diagrama de componentes fue desarrollado en StarUML, la gestión de desarrollo del proyecto, se está llevando a cabo en projects.zoho.com, específicamente en <https://projects.zoho.com/portal/acsalerosales#mywork>

Las librerías utilizadas fueron numpy, Csv, OpenCv, Python Imaging Library (PILImage), Django las cuales fueron instaladas por medio del Python Package Index (PyPI).

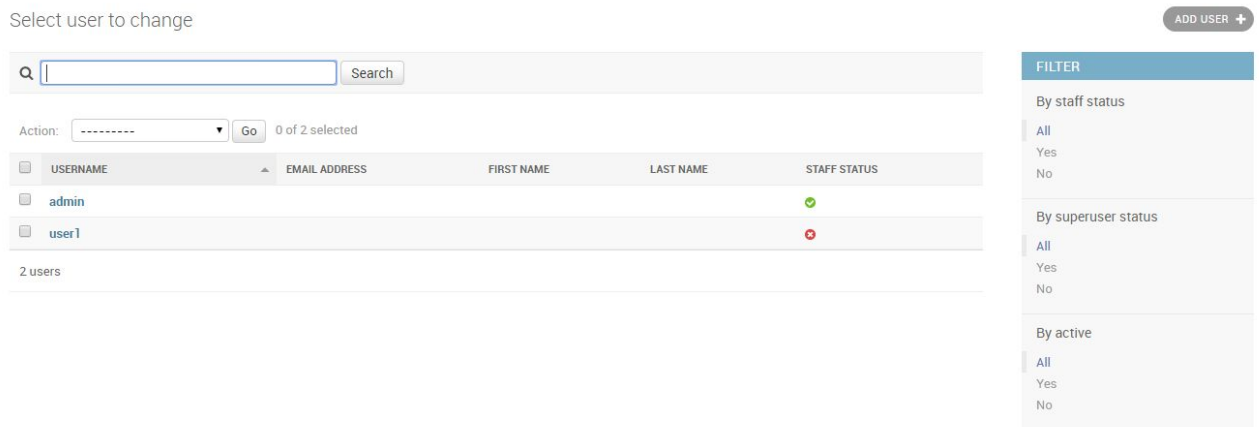
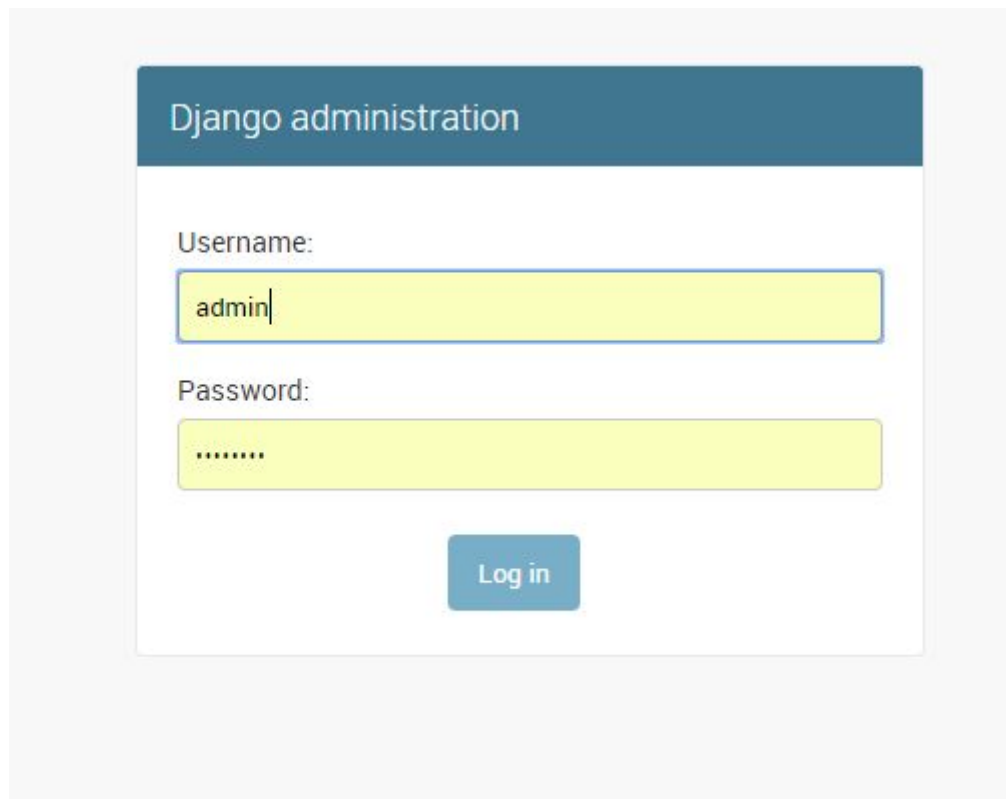
Trozos de Código

- **Login**

Sección de ingreso de usuarios, en la cual se puede iniciar sesión con los usuarios previamente registrados en el sistema.



Además, el usuario administrador puede agregar y editar usuarios en un sitio específico para la administración del sistema creado automáticamente por Django.



- **Estimador**

Sección presentada al usuario luego de haber iniciado sesión en el sistema, en la cual se proporciona la imagen que se desea analizar. Al subir la imagen se demuestra un ejemplo de conectar funciones del backend, como el que retorna el valor de la estimación (en este caso, es aleatorio), con el frontend del sistema.

CNN Bone Age Estimator

HomeHistorySign out

Upload Your Image


The image must be 255x255 in size

Patient name: John Smith

Patient age: 18

File: Choose File Xray_normal_hand_PA.jpg

SubmitSubmit



Estimate is:
8 months

CNN Bone Age Estimator

HomeHistorySign out

Upload Your Image

The image must be 255x255 in size

Patient name: John Smith

Patient age: 18

File: Choose File No file chosen

SubmitSubmit



Estimate is:
179 months

- **Cargar CSV**

Consiste en la carga del archivo csv que contiene las características y datos relevantes de los datos de entrenamiento del programa. Con el fin de analizar métricas del algoritmo. En este caso los atributos son: id, boneage, male. Se guarda como un objeto de Python. Esta funcion se encuentra en el archivo csv.py

```
id boneage male
1377 180 False
1378 12 False
1379 94 False
1380 120 True
1381 82 False
1382 138 True
1383 150 True
1384 156 True
1385 36 True
1387 138 True
1388 126 False
1389 138 True
1390 156 True
```

- **Utilización OpenCV para carga y conversión de imagen**

Se carga una imagen, y se convierte a blanco y negro. Esto con el fin de familiarizarnos con la librería de OpenCV y el manejo de imágenes.

Ejemplo de carga de una imagen tomada de internet y su respectivo almacenamiento.



Imagen original.

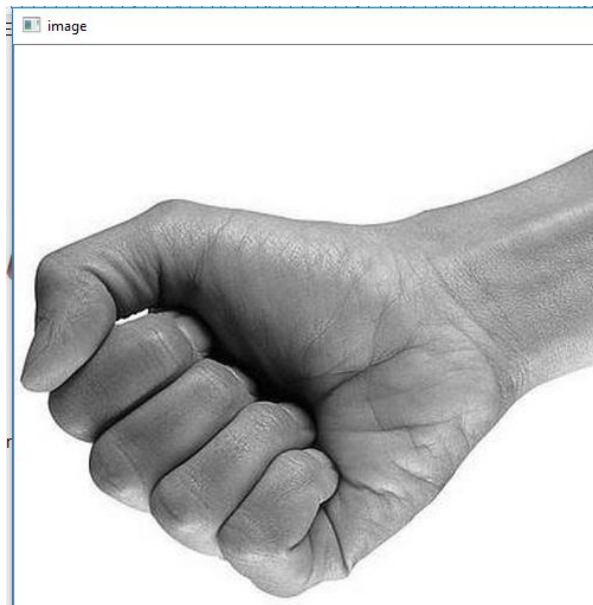


Imagen procesada y cambiada a grises por OpenCV.



Archivo original y generado luego de la corrida de código de prueba.

Justificación de Trozos

Se escogen estos trozos de código en específico, por lo siguiente:

- **Login:** Familiarizarnos con el framework de Django, y su manejo de datos persistentes, como el de los usuarios. Y más generalmente, el uso de “apps”, o páginas, y la interacción entre sí.
- **Estimación:** Probar la comunicación de funciones del backend del programa, con el frontend, también el método de subir archivos usando Django.
- **Carga CSV:** Practicar el manejo de datos desde un csv en python.
- **Utilización OpenCV para carga y conversión de imagen:** Familiarizarnos con funciones de OpenCV para el manejo de imágenes, ya que será una parte importante del sistema.

Problemas:

- Hubo dificultades al crear las páginas en HTML, se resolvieron usando una plantilla usando Bootstrap.
- Al retornar la imagen del backend para su despliegue en pantalla, hubo problemas con el manejo de los archivos recibidos por el usuario, en específico, fueron problemas convirtiendo la imagen en base64. Por lo que se tuvo que leer la documentación acerca del manejo de archivos en Django.