

Servidor web y framework IoC

John David Ibañez Rodriguez,

I. INTRODUCCIÓN

SE implementara un servidor NO concurrente desplegado en heroku el cual tendrá como funcionalidades ser capaz de entregar páginas HTML e imágenes tipo JPG y así mismo este servidor debe proveer un Framework IoC para la construcción de aplicaciones Web a partir de POJOS por el momento se tendrá soportado un POJOs sin parámetro y uno con parámetro tipado como string.

La veracidad de este software se dará por medio de veracidad HTTP/Get, donde se le pedirán recursos ya predefinidos y guardados en una 'memoria' del software, dado el caso que no se encuentre un recurso específico, se le dará una al cliente.

II. ARQUITECTURA DE SOFTWARE

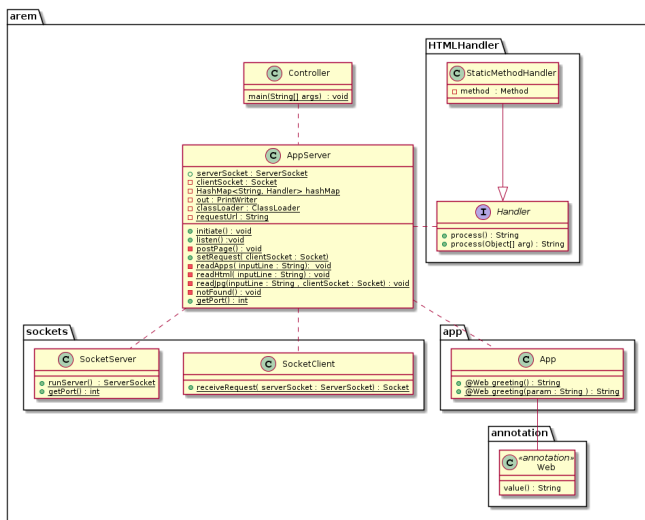


Figura 1. Diagrama de clases

Las capas de arquitectura implementadas se definieron con base a los requerimientos y necesidades de la implementación, donde cada capa de encuentra aislada de la otra, en busca de generar condiciones de estabilidad a corto plazo y probablemente extensibilidad a futuro

- App : Utilización de las etiquetas ya definidas en el capa de Annotation, los métodos que utilicen las etiquetas serán aquellos los cuales se utilizarán como /apps/some
- Annotation : capa que donde crean y se definen todas las anotaciones para las apps necesarias en el proyecto
- HTMLHandler : capa que se encarga en definir como se va a ejecutar cada método pre-definido al inicio de la aplicación,este se realiza con una interfaz para factibilidad y extensibilidad del proyecto para futuros métodos con diferentes parámetros

- Sockets : capa del proyecto que se encarga meramente de crear e inicializar los Sockets, tanto el del Cliente, como el del servidor; cada uno en su respectiva clase.

III. ARQUITECTURA DE DESPLIEGUE

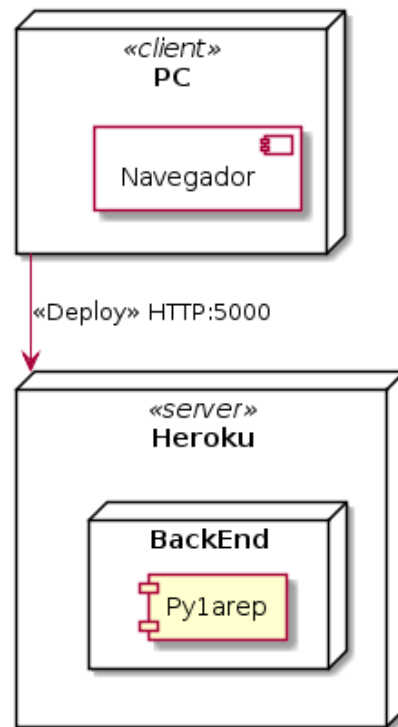


Figura 2. Deployment

Para hacer el despliegue se opto por requisitos del proyecto realizarlo en heroku, donde se tiene por el momento dos nodos, uno es el servidor de heroku donde tenemos almacenado nuestro servicio web y framework ioc y el otro es cliente-browser donde solo soportaremos peticiones get por parte de el.

IV. DESCRIPCIÓN APLICACIÓN WEB

V. PRUEBAS

V-A. Peticion html

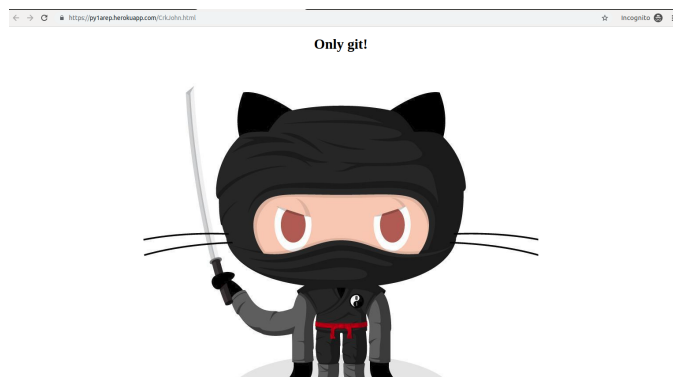


Figura 3. https://py1arep.herokuapp.com/CrkJohn.html

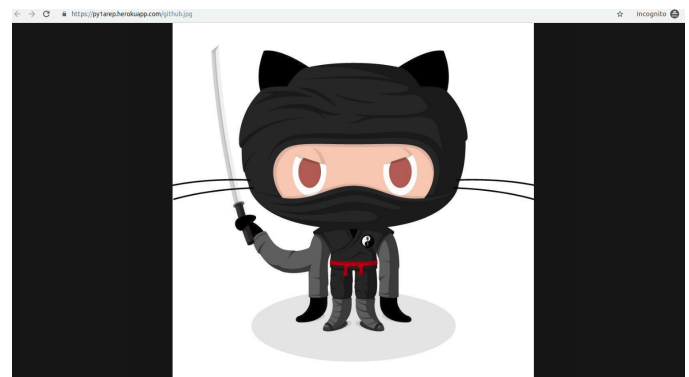


Figura 6.

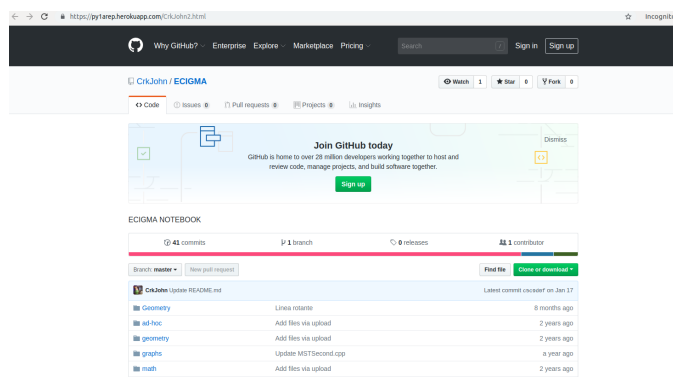


Figura 4. https://py1arep.herokuapp.com/CrkJohn2.html

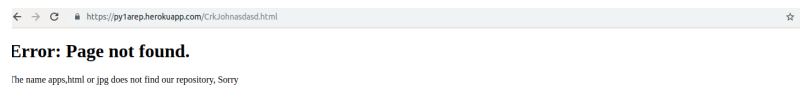


Figura 7. https://py1arep.herokuapp.com/Escue12la.html

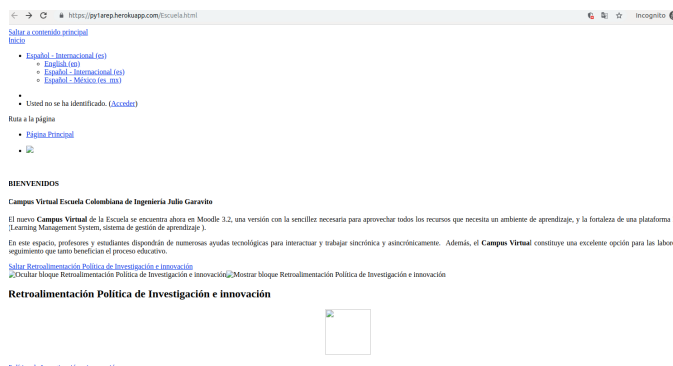


Figura 5. https://py1arep.herokuapp.com/Escuela.html

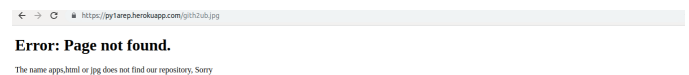


Figura 8. https://py1arep.herokuapp.com/Escuela.jpg

VI. CONCLUSION

Este servidor sería más funcional, si se tomara la información de cualquier URL tomada como parámetro y a su vez se pudiera especificar que tipo de recurso se quiere obtener de dicha URL que se solicito del mismo modo poder agregar aplicaciones Web a partir de POJOS de cualquier tipo de parámetro garantizando una disponibilidad aun más alta del software y como fin poder montar este servidor concurrente, obvivante sería un software complicado de hacer pero a su vez interesante como un proyecto del curso.