[[1]](#footnote-1)

Proyecto Primer Corte AREP (Servidor Web)

Luis Fernando Pizza Gamba, Escuela Colombiana de Ingeniería, Servidor Web,

*Abstract*—El presente documento tiene como objetivo definir la arquitectura del primer proyecto realizado para la materia Arquitecturas Empresariales. En este se desarrollo un servidor web que fuera capaz de recibir y mostrar solicitudes de tipo web o imágenes de acuerdo a sus tipos de archivos, en este caso “html” y “png”. Lo que se buscó fue crear un servidor desde cero que pudiera interpretar los dos tipos de solicitudes más básicas que podemos ejecutar cuando usamos un servidor web, además de permitir el atener múltiples solicitudes no concurrentes. Adicional a esto el servidor provee un framework IoC para la construcción de aplicaciones web a partir de POJOS.

# Introducción

E

ste documento se toma como referencia ayudando a entender la implementación realizada en el proyecto realizado, gracias a esta se tendrá una mejor explicación a nivel de diseños, implementación, entender su código fuente y su ejecución. Todo esto tomando como base los requisitos planteados al momento de plantearse este proyecto. Se definirá todos sus componentes detalladamente, entre los cuales se pueden observar a grandes rasgos su arquitectura, diseño, implementación, documentación y variables definidas para el presente proyecto.

# Arquitectura

Se definieron las 3 abstracciones fundamentales para el desarrollo del presente proyecto estas son: Memoria, Interpretes y Enlaces de Comunicación, teniendo siempre presente la definición de dichos conceptos aprendidos en clase. En el presente proyecto se podrán encontrar dichas abstracciones distribuidas de la siguiente manera:

## Memoria:

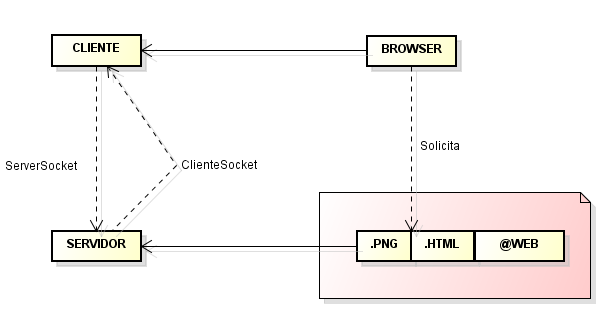
En Memoria podemos referenciar a los archivos del proyecto, los .class y .java donde se encuentra implementado nuestro proyecto, los archivos tanto .html, como .png y todos los recursos del proyecto.

## Interpretes:

En Interpretes podemos encontrar como abstracciones el Servidor, junto con la Máquina Virtual de Java, así como también el Browser que interpreta los htmls y las peticiones del servidor.

## Enlaces de Comunicaciones:

En enlaces de comunicación, se puede definir como abstracciones principales los Sockets que comunican al Cliente con el Servidor, También el protocolo http que comunica al servidor con internet, y el protocolo html que comunica a las páginas web del servidor con el brower.



# diseño

Las capas de diseño se implementaron con base a los requerimientos y/o requisitos y necesidades sobre la marcha de la implementación. Siempre buscando generar aspectos de extensibilidad y estabilidad. Dicho lo anterior en diseño para el diseño que se implemento al proyecto se implementaron 5 capas que se presentan a continuación.

## Socket:

## Se encarga meramente de crear e inicializar los Sockets, tanto el del Cliente, como el del servidor. Estos dos en sus clases correspondientes. Dichas clases son llamadas desde la clase principal, esta corre el servidor crenado instancias de los sockets de estos.

## httpServer:

Esta capa construye la clase principal del proyecto que en este caso es “HttpServer” desde aquí corre el servidor llamando a

los sockets y se empiezan a pedir solicitudes del cliente hacia el servidor. También se recorren los POJOS dentro de la capa de framework y extrae los métodos de estos alojándolos en un atributo de tipo Hash Map y extrae los parámetros de algún tipo de solicitud si este el caso.

## Paginas:

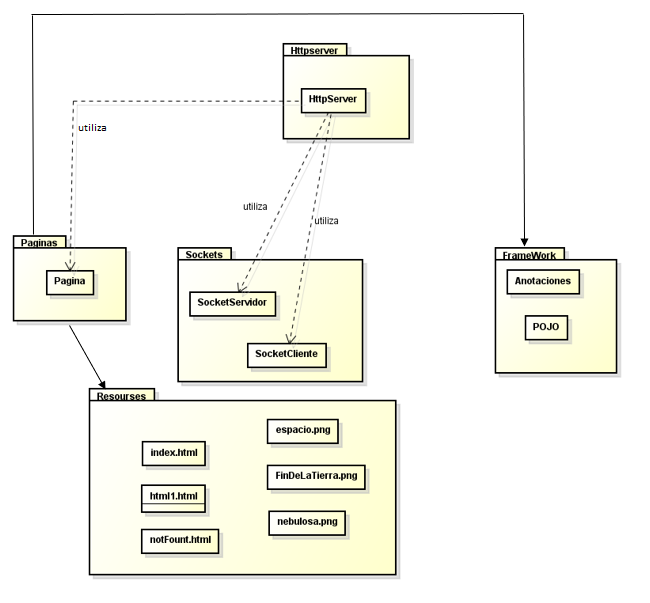
Esta capa es la que genera la visualización de todas las solicitudes de tipo HTML, png y los POJOS dentro del servidor con anotaciones de tipo @web. es acá donde tanto se leen, cómo se interpretan, visualizan y publican, los resultados de cualquier solicitud en el servidor.

## Resourses:

## En esta capa del proyecto, podemos hacer una referencia a la abstracción de memoria donde se encuentran alojadas los recursos html y png con los que cuenta el servidor para responder las múltiples solicitudes no concurrentes por parte de cualquier cliente.

## FrameWork:

En esta capa del proyecto se encuentran alojados los diferentes Pojos que se utilizaran para la construcción de aplicaciones web si el cliente lo solicita. Este será el único paquete en donde se podrán adicionar tantos Pojos como se quiera. Adicionalmente deberán de almacenar las diferentes anotaciones que se quieran implementar.



## 

# ESTRUCTURA DEL SERVIDOR

Este compuesto por los diferentes recursos, los cuales se podrán acceder desde la pagina principal que es (“/index.html”). De aquí se generan todos los vínculos a las demás dependencias del servidor, Incluyendo tanto las páginas HTML, como las imágenes usadas para visualizarlas. Para acceder a la visualización de los diferentes métodos con notación web (Pojos) se debe de colocar la extensión del método a llamar y si este necesita de parámetros a continuación se deberá colocar dos puntos y colocar los parámetros pero estos también deben de ir separados por dos puntos a continuación se presentaran dos ejemplos validos en el servidor: /suma:2:2, /cuadrado:2.

# conclusion

## Gracias al Desarrollo de este este proyecto se pudo apreciar de un modo mas claro la importancia de delegar, diseñar y estructurar de una forma eficiente en las diferentes capas la implementación del servido, dado que resultaría demasiado complejo la comunicación entre los diferentes sockets en solo una clase y la implementación de las diferentes solicitudes que se hagan. Se pudo observar la gran importancia que tienen los sockets a la hora de implementar y/o desarrollar un servidor web que, aunque si bien se sabe son de bajo nivel, ayudan notablemente en la comunicación Cliente-Servidor.

**Luis Fernando Pizza Gamba** nació en Bogotá, Cundinamarca, Colombia en 1998. Graduado de el Colegio Tomas Cipriano De Mosquera I.E.D en el 2015, inició sus estudios de Ingeniería de Sistemas en 2016.

1. . [↑](#footnote-ref-1)