

Examen Final de Sistemas Distribuidos

Nombre: Rubén Romero Ortega

C.U. 174178

El examen es individual.

(2 puntos) Explica bajo qué requerimientos usarías cada una de las siguientes tecnologías:

Sockets TCP/UDP

Cuando se tiene una conexión en el Sistema de forma que se pueda comunicar por medio de puertos entre dos equipos dentro del Sistema distribuido. El medio de comunicación existe internamente en el Sistema, se conocen las direcciones IP de los servidores y de los clientes.

Además, esta tecnología asegura una comunicación o llegada de paquetes rápida en comparación con las otras tecnologías del Sistema. Pues tienen menores sobrecargas.

Java RMI

Necesidad de diseñar un sistema distribuido que invoque remotamente métodos, usando programación orientada a objetos necesariamente en Java. Se pueden acceder a los servicios gracias a los métodos que ofrecen los objetos que están regidos bajo este modelo de programación.

Los servicios se publican de forma que el cliente pueda acceder a las interfaces públicas y se pueda utilizar el servicio por medio de una invocación remota.

Servicios web SOAP

Cuando se necesite adaptar el modelo de programación web en aplicaciones que no necesiten tecnología de los navegadores sin importar la heterogeneidad. Se necesita conocer la descripción del servicio web que proporciona la página web (WSDL), para conocer las características de los métodos o servicios.

Comunicación indirecta

En un ambiente altamente volátil, donde los usuarios puedan cambiar o no se conozcan. Son tecnologías donde el receptor no necesita conocer al emisor o viceversa, los participantes salen y entran a un ritmo distinto que las otras tecnologías. También se necesita un desacoplamiento temporal ya que los participantes no necesitan estar conectados todo el tiempo ni al mismo tiempo para lograr recibir o transmitir.

(1 punto) En una página web ¿por qué es importante dónde se llaman/cargan los archivos externos de javascript?

Es importante saber colocar un script de javascript en HTML porque el orden de ejecución puede hacer que los sistemas no se carguen como se desean, también se debe recordar que los archivos externos pueden tener 3 configuraciones: la normal, la async y la defer.

Si un script se coloca en el header se espera que tenga funciones que se vayan a usar a lo largo de la página, si se coloca dentro de la página se espera que tenga una interacción directa donde se inserte código o imágenes. Las configuraciones van a afectar el desempeño de la ejecución de los Scripts, mientras que el normal para la descarga y parseo de HTML para descargar y ejecutar el script, en async se descarga el HTML y el Script al mismo tiempo y sólo se detiene HTML para la ejecución del Script. Finalmente Defer deja la ejecución del script hasta el final de la descarga de HTML y el Script.

(2 punto) Crea un archivo de descripción de un servicio web (es decir, un WSDL) original de una tienda en línea que tenga una operación con cuatro parámetros de entrada. El WSDL lo puedes hacer manualmente o utilizando Netbeans u otras herramientas similares. Copia el contenido del WSDL en este documento y explica en qué consisten cada una de sus secciones.

- 1) Definitions tiene el schema de toda la descripción del servicio web o del WSDL.
- 2) En message name vemos el nombre de la función “compraFunko”
- 3) En portType name observamos el nombre de las operaciones que podemos observar en nuestra página web OperacionesTiendaLinea, junto con sus inputs y sus respectivos outputs.
- 4) En binding observamos el formato del mensaje, asimismo observamos el protocolo que lo rige, siendo en este caso SOAP. El binding style se puede notar que es de documentos.

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Published by JAX-WS RI (http://javae.github.io/metro-jax-ws). RI's version is Metro/2.4.2 (UNKNOWN-17d59c9; 2018-09-26T15:16:30-0700) JAXWS-RI/2.3.1 JAXWS-API/2.3.1 JAXB-RI/2.3.1 JAXB-API/2.3.1 svn-revision#unknown. -->
<!-- Generated by JAX-WS RI (http://javae.github.io/metro-jax-ws). RI's version is Metro/2.4.2 (UNKNOWN-17d59c9; 2018-09-26T15:16:30-0700) JAXWS-RI/2.3.1 JAXWS-API/2.3.1 JAXB-RI/2.3.1 JAXB-API/2.3.1 svn-revision#unknown. -->
- definitions name="OperacionesTiendaLinea" targetNamespace="http://webservices/" xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:tns="http://webservices/"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata" xmlns:wsp1_2="http://schemas.xmlsoap.org/ws/2004/09/policy" xmlns:wsp="http://www.w3.org/ns/ws-policy"
  xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
  <types>
    <xsd:schema>
      <xsd:import schemaLocation="http://localhost:32093/SD_ExFin_TiendaLinea/OperacionesTiendaLinea?xsd=1" namespace="http://webservices/" />
    </xsd:schema>
  </types>
  <message name="compraFunko">
    <part name="parameters" element="tns:compraFunko"/>
  </message>
  <message name="compraFunkoResponse">
    <part name="parameters" element="tns:compraFunkoResponse"/>
  </message>
  <portType name="OperacionesTiendaLinea">
    <operation name="compraFunko">
      <input message="tns:compraFunko" wsam:Action="http://webservices/OperacionesTiendaLinea/compraFunkoRequest"/>
      <output message="tns:compraFunkoResponse" wsam:Action="http://webservices/OperacionesTiendaLinea/compraFunkoResponse"/>
    </operation>
  </portType>
  <binding name="OperacionesTiendaLineaPortBinding" type="tns:OperacionesTiendaLinea">
    <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name="compraFunko">
      <soap:operation soapAction="">
        <input>
          <soap:body use="literal"/>
        </input>
        <output>
          <soap:body use="literal"/>
        </output>
      </operation>
    </binding>
  </service name="OperacionesTiendaLinea">
    <port name="OperacionesTiendaLineaPort" binding="tns:OperacionesTiendaLineaPortBinding">
      <soap:address location="http://localhost:32093/SD_ExFin_TiendaLinea/OperacionesTiendaLinea"/>
    </port>
  </service>
</definitions>
```

En xsd=1 vemos los tipos de datos que va a permitir la función de compraFunko y cuales son sus características.

```

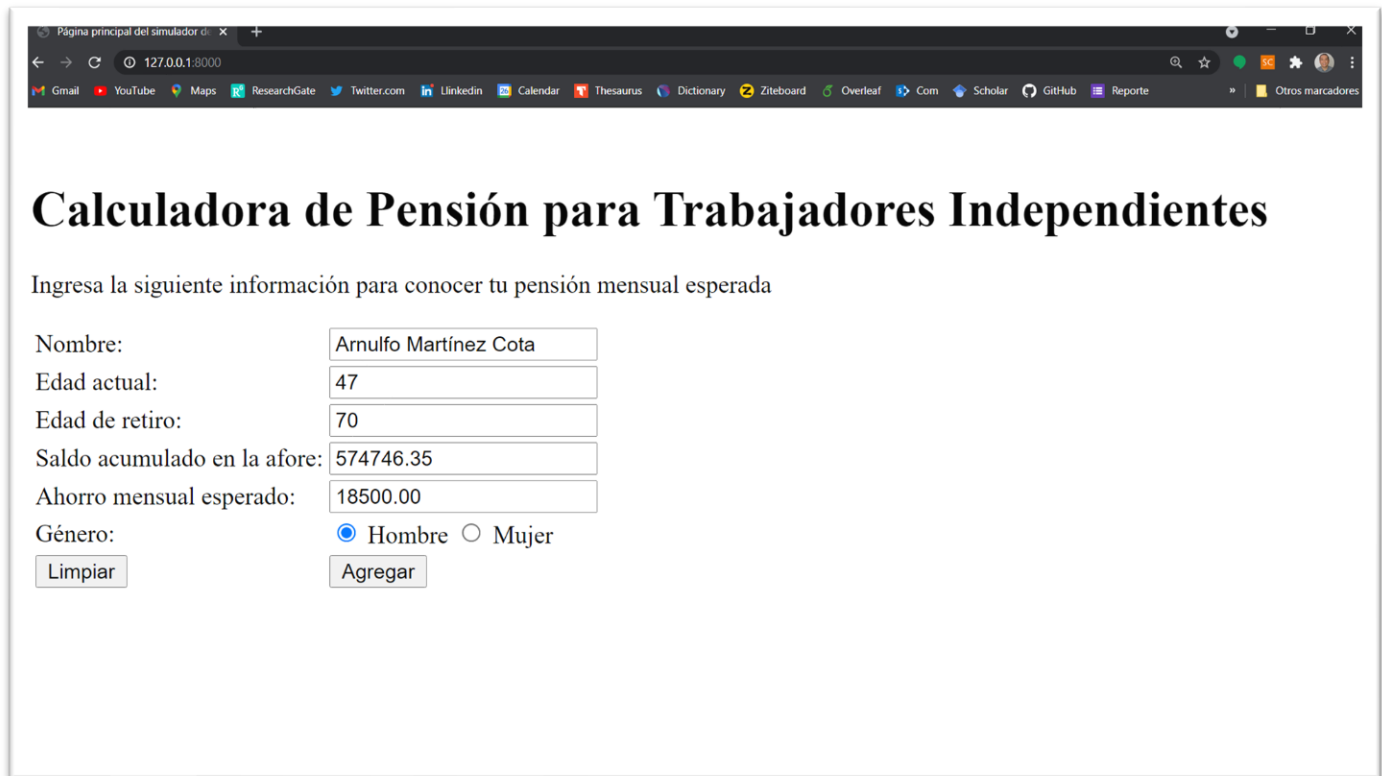
<?xml version="1.0" encoding="UTF-8"?>
<!-- Published by JAX-WS RI (http://jax-ws.java.net). RI's version is Metro/2.4.2
(UNKNOWN-17d59c9; 2018-09-26T15:16:30-0700) JAXWS-RI/2.3.1 JAXWS-API/2.3.1
JAXB-RI/2.3.1 JAXB-API/2.3.1 svn-revision#unknown. -->
<xs:schema targetNamespace="http://webservices/"
xmlns:tns="http://webservices/" version="1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="compraFunko" type="tns:compraFunko"/>
  <xs:element name="compraFunkoResponse"
    type="tns:compraFunkoResponse"/>
  - <xs:complexType name="compraFunko">
    - <xs:sequence>
      <xs:element name="nameFunko" type="xs:string" minOccurs="0"/>
      <xs:element name="precioFunko" type="xs:double"/>
      <xs:element name="direccionEnvio" type="xs:string"
        minOccurs="0"/>
      <xs:element name="tarjetaCredito" type="xs:string" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  - <xs:complexType name="compraFunkoResponse">
    - <xs:sequence>
      <xs:element name="return" type="xs:string" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>

```

Crea un sitio web utilizando DJANGO. Un gobierno de un país del G20 te contrata para hacer un sitio web que permita a los trabajadores independientes calcular su pensión a partir de i) su edad actual (int), ii) la edad de retiro esperada (int), iii) el saldo acumulado en la afore (float), iv) el ahorro mensual esperado (float) y su género (bool).

(2 puntos)

La calculadora de pensión se debe poder acceder indicando solo `http://127.0.0.1:8000/` sin agregar nada más. A continuación, se muestra la página de bienvenida que debes hacer y los datos que debes capturar:



The screenshot shows a web browser window with the address bar displaying `127.0.0.1:8000`. The page title is "Página principal del simulador". The main heading is "Calculadora de Pensión para Trabajadores Independientes". Below the heading, there is a prompt: "Ingresa la siguiente información para conocer tu pensión mensual esperada". The form contains the following fields and controls:

Nombre:	Arnulfo Martínez Cota
Edad actual:	47
Edad de retiro:	70
Saldo acumulado en la afore:	574746.35
Ahorro mensual esperado:	18500.00
Género:	<input checked="" type="radio"/> Hombre <input type="radio"/> Mujer
<input type="button" value="Limpiar"/>	<input type="button" value="Agregar"/>

El botón "Limpiar" debe reestablecer a los valores por defecto de todos los inputs. El botón "Agregar" debe guardar toda la información capturada en una base de datos y mostrar la siguiente página mostrando el nombre del trabajador independiente y la pensión mensual esperada.



(1 punto)

El cálculo de la pensión mensual esperada NO SE DEBE ALMACENAR en la base de datos. Siempre se debe calcular en función de los otros datos provistos por el trabajador independiente. El cálculo de la pensión mensual esperada se debe realizar en el modelo. Dentro del “modelo”, define una forma de calcular la pensión mensual esperada a partir de i) la edad actual, ii) la edad de retiro esperada, iii) el saldo acumulado en la afore y iv) el ahorro mensual esperado.

(1 punto)

Además, como parte del proyecto, te solicitan hacer una página donde se muestre un listado de todos los registros que se han capturado en el sistema. El listado debe poderse ordenar por nombre, edad actual, saldo acumulado y/o ahorro mensual esperado. El criterio de ordenamiento se debe pasar como parámetro en la URL de la página. La página se debe ver de la siguiente manera:

Listado de simulaciones de pensión

127.0.0.1:8000/nombre/listado/

Gmail

YouTube

Maps

ResearchGate

Twitter.com

LinkedIn

Calendar

Thesaurus

Dictionary

Ziteboard

Overleaf

Com

Scholar

GitHub

Reporte

Otros marcadores

Listado de registros de simulaciones de pensión

- Nombre: Arnulfo Martínez Cota | Edad actual: 47 | Edad de retiro: 70 | Género: Hombre | Saldo acumulado: \$574746.35 | Ahorro mensual: \$18500.0 | Pensión esperada: \$18616.517638888887
- Nombre: Dolores Carmen Zaragoza Ruiz | Edad actual: 19 | Edad de retiro: 63 | Género: Mujer | Saldo acumulado: \$20000.0 | Ahorro mensual: \$5000.0 | Pensión esperada: \$8855.555555555555
- Nombre: Luis Jiménez Pérez | Edad actual: 45 | Edad de retiro: 60 | Género: Hombre | Saldo acumulado: \$250000.0 | Ahorro mensual: \$1500.0 | Pensión esperada: \$1594.4444444444443
- Nombre: Rosa María López García | Edad actual: 57 | Edad de retiro: 70 | Género: Mujer | Saldo acumulado: \$480000.0 | Ahorro mensual: \$25000.0 | Pensión esperada: \$14333.333333333334

(1 punto)

Finalmente, te solicitan que habilites el “modelo” de los registros de simulación de pensión dentro del sistema para que un administrador pueda agregar, borrar, editar, y/o consultar los registros manualmente. El módulo de administración podría verse de la siguiente manera:

Select registro to change | Django

127.0.0.1:8000/admin/simulador/registro/

Gmail

YouTube

Maps

ResearchGate

Twitter.com

LinkedIn

Calendar

Thesaurus

Dictionary

Ziteboard

Overleaf

Com

Scholar

GitHub

Reporte

Otros marcadores

Django administration

WELCOME, ADMIN. [VIEW SITE](#) / [CHANGE PASSWORD](#) / [LOG OUT](#)

Home > Simulador > Registros

AUTHENTICATION AND AUTHORIZATION

Groups

+ Add

Users

+ Add

SIMULADOR

Registros

+ Add

Select registro to change

ADD REGISTRO +

Action: Go 0 of 4 selected

REGISTRO

Nombre: Arnulfo Martínez Cota | Edad actual: 47 | Edad de retiro: 70 | Género: Hombre | Saldo acumulado: \$574746.35 | Ahorro mensual: \$18500.0 | Pensión esperada: \$18616.517638888887

Nombre: Luis Jiménez Pérez | Edad actual: 45 | Edad de retiro: 60 | Género: Hombre | Saldo acumulado: \$250000.0 | Ahorro mensual: \$1500.0 | Pensión esperada: \$1594.444444444443

Nombre: Rosa María López García | Edad actual: 57 | Edad de retiro: 70 | Género: Mujer | Saldo acumulado: \$480000.0 | Ahorro mensual: \$25000.0 | Pensión esperada: \$14333.333333333334

Nombre: Dolores Carmen Zaragoza Ruiz | Edad actual: 19 | Edad de retiro: 63 | Género: Mujer | Saldo acumulado: \$20000.0 | Ahorro mensual: \$5000.0 | Pensión esperada: \$8855.555555555555

4 registros

INFORMACIÓN ADICIONAL

Las respuestas a este examen se deben enviar en un archivo comprimido a itam.examen@gmail.com y a octavio.gutierrez@itam.mx. Recomiendo que también envíen las respuestas con copia a alguno de sus correos para que sirva como acuse de recibo.

La hora límite para entregar el examen es 3:45 pm. En caso de entregar el examen posterior a esa hora, habría un descuento de 10% por cada tres minutos de retraso.

A más tardar mañana jueves antes de las 12h, te enviaré un mensaje en el chat de Teams con tu calificación final. Si deseas revisión, por favor contáctame a la brevedad pues subiré las actas oficiales al sistema del ITAM el 28 de mayo a las 11 am.