

1-4-2022 – 07/06/2022

# Toy Computer Corporation

Desarrollo de comparador de procesadores.

Rafael Hermosilla Cabello

Desarrollo de Aplicaciones Multiplataforma (DAM)

Tutor: José Manuel Gras

IES Serra Perenxisa (2021-2022)



## Toy Computer. Software de comparación de procesadores

Este trabajo, así como el código que aparece en él está sujeto a una licencia creative commons que permite su difusión uso comercial reconociendo siempre la autoría de su creador.

Se garantiza permiso para copiar, distribuir y modificar este documento según los términos de licencia **Creative Commons BY-NC-SA 4.0**



## Agradecimientos

Me gustaría en primer lugar agradecer a las diferentes personas que me han dado su apoyo y ayuda en este último año para conseguir mi titulación.

En primer lugar, a todo el profesorado del IES Serra Perenxisa por darme una excelente formación en este curso. Sobre todo, a mi tutor del proyecto final, José Manuel Gras, por haberme guiado y ayudado en este último paso.

En segundo lugar, quería agradecer el apoyo recibido por diferentes familiares, en especial a Juanjo y Silvia por darme diferentes puntos de vista e ideas que me han podido ayudar a enriquecer mi proyecto.

## Resumen del proyecto

El proyecto consiste en una aplicación para poder comparar todos los procesadores del mercado y además de poder crear, editar o eliminar tus propios procesadores para poder compararlos con los que ya viene en la aplicación.

## Resum del projecte

El projecte consisteix en una aplicació per poder comparar tots els processadors del mercat i a més de poder crear, editar o eliminar els teus propis processadors per poder comparar-los amb els que ja ve a l'aplicació.

## Project summary

The project consists of an application to be able to compare all the processors on the market and also to be able to create, edit or delete your own processors to be able to compare them with those that already come in the application.

## Índice

<b>1)Introducción .....</b>	<b>5</b>
1.1) Descripción .....	5
1.2) Ideas Principales .....	5
<b>2)Estado del arte .....</b>	<b>6</b>
2.1) Ideas.....	6
2.2) Novedades no existentes .....	6
<b>3)Estudio de viabilidad.....</b>	<b>7</b>
3.1) DAFO .....	7
3.2) CAME .....	7
3.3) Estudio de mercado .....	8
3.4) Recursos necesarios .....	8
3.5) Viabilidad temporal y planificación .....	10
<b>4)Análisis de requisitos .....</b>	<b>11</b>
4.1) Requisitos del cliente .....	11
4.2) Requisitos de las tecnologías seleccionadas .....	11
4.3) Requisitos del tiempo de desarrollo .....	12
4.4) Diagrama general de casos de uso .....	12
4.5) Diseño .....	13
<b>5)Codificación .....</b>	<b>15</b>
5.1) Tecnologías elegidas .....	15
5.2) Documentación interna .....	19
5.3) Manual de usuario .....	20
<b>6)Despliegue.....</b>	<b>29</b>
6.1) Desarrollador .....	29
6.2) Cliente.....	29
6.3) Herramientas de apoyo.....	29
6.4) Gestión de pruebas .....	30
<b>7)Conclusiones .....</b>	<b>30</b>
7.1) Conclusión personal.....	30
7.2) Posibles ampliaciones y mejoras.....	31
<b>8)Bibliografía .....</b>	<b>32</b>
8.1) Apuntes.....	32
8.2) Páginas Web .....	32

# 1 Introducción

## 1.1 Descripción

Este proyecto va a consistir en una aplicación para que cualquier usuario pueda comparar cualquier procesador del mercado y para ello voy a utilizar el lenguaje **Java** con una tecnología denominada **JavaFX**. Esta aplicación tendrá su propio comparador que contendrá toda la información de los dos procesadores, un gráfico de barras para saber que característica en específico gana al otro procesador y un gráfico circular para ver generalmente que procesador es mejor. También tendrá una gestión de usuarios que servirá para poder cambiar el nombre, contraseña, tipo y verificación del usuario en la aplicación y toda esta información se guardará en la Base de datos.

Mas adelante implementare una seguridad para iniciar sesión en el que consistirá en una verificación por correo electrónico para que cada vez que inicies sesión te mande un código que tengas que poner para poder iniciar sesión. También tendrá un registro de usuarios para poder acceder a la aplicación.

Implementare el envío de correo electrónico tanto para iniciar sesión si el usuario quiere y para el registro de usuario que eso sí que será obligatorio.

Cuando un usuario quiera crear más de un procesador deberá “comprar” la extensión que le dejara crear todos los procesadores que quiera y cuando haga eso se lo generara un informe que será una factura, para poder crear el informe pase mi proyecto a **Maven**.

Además, creare una página web desarrollada con **HTML**, **CSS** y **JavaScript** para que el usuario se pueda descargar la aplicación y vea de forma simple en que consiste la aplicación, además de tener una ayuda en la propia página web para que el usuario si en algún apartado tiene alguna duda de cómo utilizar la aplicación acuda a la página web.

Posteriormente voy a implementar una Base de Datos con **MySQL** que tendrá los procesadores por defecto de la aplicación, los usuarios registrados de la aplicación con toda la información proporcionada en el registro de la aplicación, las valoraciones de las comparaciones o de la misma aplicación y los procesadores creados por los usuarios.

Mas adelante implementare que la información más delicada como las contraseñas de los usuarios se envíen de forma encriptada a la Base de Datos para tener mayor seguridad.

Y por último creare un servidor hecho con Sockets e hilos con Java para que haga las peticiones a la Base de Datos y le envíe la información al usuario de la Base de Datos o le envíe correos al usuario.

## 1.2 Ideas principales

El principal objetivo de este proyecto es que el usuario que tenga la aplicación tenga libertad a la hora de usar la aplicación (dentro de sus limitaciones) como por ejemplo crear los procesadores que quiera (dependiendo del tipo de usuario que sea) y compararlos con los del mercado. También que tenga libertad de expresión a la hora de valorar la aplicación o la comparativa de los procesadores que haya puesto el usuario.

## 2 Estado del arte

### 2.1 Ideas

Buscando ideas para mi proyecto encontré un patrón muy similar y es que mayoría de los productos tienen una página web ya sea porque el comparador es la página web o para exponer su aplicación al público y es por ello por lo que creare una página web para exponer mi proyecto a los usuarios para que vean las diferencias con los otros productos y para que se puedan descargar la aplicación.

Algo en el que tome buenas ideas de otros productos es a la hora de distribuir la información a la hora de comparar ya que lo primero es mostrar la información general de cada procesador y luego ya hacer la comparación más en detalle y más adelante más general para ver cuál es mejor.

Además, vi que la mayoría de los comparadores no son aplicaciones de escritorio, sino que son páginas web por ello pensé en crear la aplicación de escritorio para estar pensada en que una vez se descargue el usuario siempre la tenga para que en un futuro no esté buscando páginas web hasta encontrar la idónea, sino que con un solo clic ya tenga el comparador sin ninguna preocupación.

### 2.2 Novedades no existentes

Las novedades que tiene mi proyecto frente a otros del mismo estilo es que el usuario pueda **crear su propio procesador** ya sea inventado o alguno que falte por incorporar. Por ello al dejarle tanta libertad al usuario para que no colapse la base de datos ante tantos datos pensé en crear un tipo de cuenta en la que sería la principal fuente de ingresos de la aplicación ya que dependiendo del tipo de cuenta podrás crear uno o varios procesadores y en este caso hay dos tipos de cuentas que son los de tipo prueba y tipo premium. El tipo prueba será gratuito y se podrá crear un procesador y el tipo premium tendrá un coste de 10 euros, pero se podrá crear todos los procesadores que el usuario quiera

Además de incorporar una **gestión de usuarios** que es algo también novedoso en este territorio ya que para acceder a nuestra aplicación el usuario tendrá que crearse una cuenta en Toy Computer con la que más tarde podrá editar toda la información indicada.

Y por último una de las novedades es el **envío de correos electrónicos** ya que con el tenemos un nivel más de seguridad ya que el usuario puede indicar en la aplicación una verificación por correo, además que para registrarse debe poner un código que se le enviara por correo o incluso mandarle información de actualizaciones o de seguridad.

## 3 Estudio de viabilidad

### 3.1 DAFO



#### Debilidades



Poco tiempo de desarrollo



Nula visión Empresarial



Experiencia en el desarrollo de aplicaciones



#### Fortalezas



Producto muy libre para el usuario



#### Amenazas



Competencia mas experimentada



Poca visibilidad a la hora de encontrar clientes



#### Oportunidades



Novedades que no tiene la competencia



Uso para todo tipo de usuarios

### 3.2 CAME



#### Corregir



Hacer un curso de visión empresarial



Mas margen de tiempo para tener mas pulida la aplicación y tener mas experiencia



#### Mantener



Mantener la esencia de la aplicación para darle libertad al usuario



#### Afrontar



Mejorar pagina web y hacer publicidad



Aprender de la competencia y mejorar los puntos negativos



#### Explotar



Seguir sacando actualizaciones para competir



Seguir con la simpleza para que todo el mundo pueda usar la aplicación



## 3.3 Estudio de mercado

### ¿Es necesario nuestro producto?

Nos encontramos muchas veces que un usuario se va a comprar un nuevo dispositivo y para cualquier usuario ya sea experimentado o no, siempre nos entra la duda de que dispositivo comprar, ya que hay muchos procesadores en el mercado y no sabemos cuál es mejor, es por ello por lo que nuestra aplicación es necesaria, aunque sea de forma temporal para el usuario.

### ¿Tiene hueco en el mercado?

Lo bueno de la aplicación es que no hay muchas aplicaciones parecidas y las que hay no tienen lo nuevo que incorpora esta aplicación que es el poder crear tu propio procesador y poder editarlo y eliminarlo, es por ello por lo que claro que tiene hueco además de no tener coste y poder probarla cuando quieras.

### ¿A qué público objetivo interesa?

Lo bueno de este proyecto es que va dirigido a cualquier público ya sea experimentado o no en el sector, ya que en todo caso si el usuario se lía en la aplicación tiene una ayuda muy sencilla de entender de cómo usar la aplicación. Además, que puede usar la aplicación para ver información sobre su procesador y compararlos con otros por mera diversión o por si está interesado en comprarse un nuevo dispositivo o comprarse una nueva CPU.

## 3.4 Recursos necesarios

### Recursos Hardware necesarios

**-Por parte del desarrollador:** para elaborar el proyecto tuve un portátil Lenovo IdeaPad 5.

**-Por parte del usuario:** Como vi en el proyecto los recursos que consume la aplicación no son demasiados, es por ello por lo que le recomiendo al usuario un dispositivo de calidad de hardware bajos/medio.

### Requisitos Mínimos

Memoria RAM: 4GB

Procesador: i3-10100

Gráfica: NVIDIA GeForce GTX 770

Almacenamiento: 100 MB

## Requisitos Recomendados

Memoria RAM: 8GB

Procesador: I5-10400

Gráfica: NVIDIA GeForce GTX 1050

Almacenamiento: 100 MB

## Recursos Software necesarios

### -Por parte del programador:

- **Java**
  - Para poder desarrollar este proyecto use el lenguaje de programación Java que es el que aprendimos en este curso.
  - El coste fue de 0€.
- **XAMPP**
  - Para poder crear la Base de Datos y tener la pagina web activa use XAMPP.
  - El coste fue de 0€.
- **Eclipse**
  - Para poder desarrollar mi proyecto con Java use el entorno de desarrollo de Eclipse.
  - El coste fue de 0€.
- **Bibliotecas de correo y de conexión para la Base de Datos**
  - Para poder enviar y recibir información de la base de datos tuve que adquirir las bibliotecas adecuadas para ello y además también para poder enviar correos al usuario también tuve que adquirir una biblioteca.
  - El coste fue de 0€.
- **Visual Studio Code**
  - Para poder desarrollar la pagina web y además desarrollar los estilos del proyecto tuve que adquirir Visual Studio Code.
  - El coste fue de 0€.
- **Install4J**
  - Para poder crear un instalador para que el usuario se pueda instalar nuestra aplicación tuve que adquirir la aplicación Install4J.
  - El coste fue de 0€.
- **Scene Builder**
  - Para poder hacer el layout de la aplicación tuve que adquirir Scene Builder.
  - El coste fue de 0€.

- **Maven**
  - Para poder crear la factura para el cliente dentro de la aplicación tuve que adquirir Maven.
  - El coste fue de 0€.
- **GIT**
  - Para poder tener un control de versiones de mi proyecto tuve que adquirir GIT.
  - El coste fue de 0€.

#### -Por parte del usuario:

- **Sistema Operativo**
  - Por parte del usuario no necesita mucho uno de los pocos requisitos es tener un sistema operativo Windows o Linux.
- **Java**
  - Por último, requisito es tener instalado y actualizado Java.

## Recursos humanos

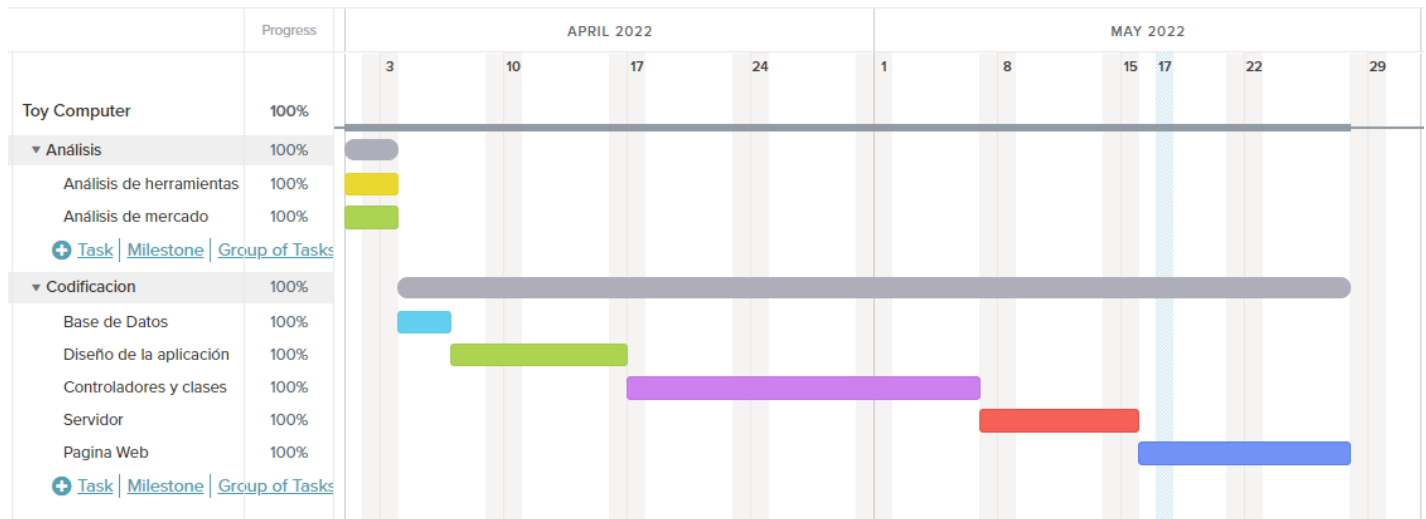
Rafael Hermosilla (programador).

## 3.5 Viabilidad temporal y planificación

El tiempo efectivo para la realización de este desarrollo fue de tres meses, entre abril y junio de 2022. Con este límite se planteó el desarrollo completo que estaba pensado, pero con más tiempo se puede hacer una aplicación más completa y mucho más compleja con mayor tiempo de codificación.

Para poder gestionar la viabilidad temporal del proyecto se utilizó la herramienta web **Team Gantt** (<https://www.teamgantt.com/>), que permite de forma gratuita hacer tu propio diagrama para poner lo que vas a hacer en los próximos días y poner si lo has completado o no.

## Planificación Temporal



## 4 Análisis de requisitos

### 4.1 Requisitos del cliente

Los requisitos del cliente es básicamente que pueda ver los procesadores actuales y que los pueda crear si quiere compararlos con los que ya están en la aplicación.

Otros requisitos del cliente incluyen el servicio de correo electrónico para mayor seguridad en la aplicación, facturación, gestión de clientes.

### 4.2 Requisitos de las tecnologías seleccionadas

En cuanto a los requisitos de las tecnologías seleccionadas para poder acceder a la información de la aplicación es necesario un servidor local.

Se utilizó XAMPP como servidor local para tener la Base de datos y la página web.

Java requiere las bibliotecas de conexión a la base de datos y la biblioteca para mandar correos a los usuarios. Además, necesitaremos que nuestro proyecto este hecho en Maven para poder crear el informe de la factura

### 4.3 Requisitos del tiempo de desarrollo

Con el escaso tiempo de desarrollo del que se disponía para realizar el proyecto se hizo todo lo propuesto con tiempo, aunque las ideas que se tuvo realizando el desarrollo no dio tiempo al incorporarlo por el escaso tiempo.

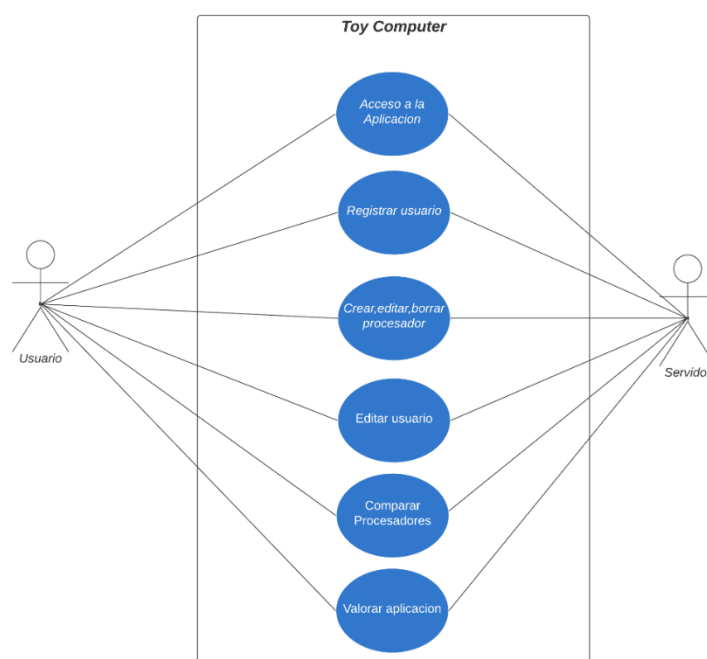
Estas que dieron tiempo incluyen la gestión de usuarios, el comparador, la gestión de procesadores, generación de informes, la gestión de correos, servidor, página web y la Base de Datos.

Las ideas que no dieron tiempo por el escaso tiempo fueron la de crear una nueva funcionalidad en la que el programa preguntaba una serie de preguntas y dependiendo de las opciones que eligiera le recomendaba una opción u otra, además crear otras extensiones que le recomendaran dependiendo del procesador una placa base y memoria RAM. Otra idea fue la de crear otra ventana más individualmente para el administrador de la aplicación en la que pudiera ver los usuarios creados por si hay alguno inapropiado y si lo hay borrarlo y lo mismo para los procesadores creados por los usuarios y sus valoraciones. Y por último trasladar la aplicación a una aplicación Android.

### 4.4 Diagrama general de casos de uso

A continuación, se muestra el diagrama de casos de uso genérico para la aplicación. El usuario de la aplicación accedera con sus credenciales y le mandara la información al servidor y si son correctas podrá continuar.

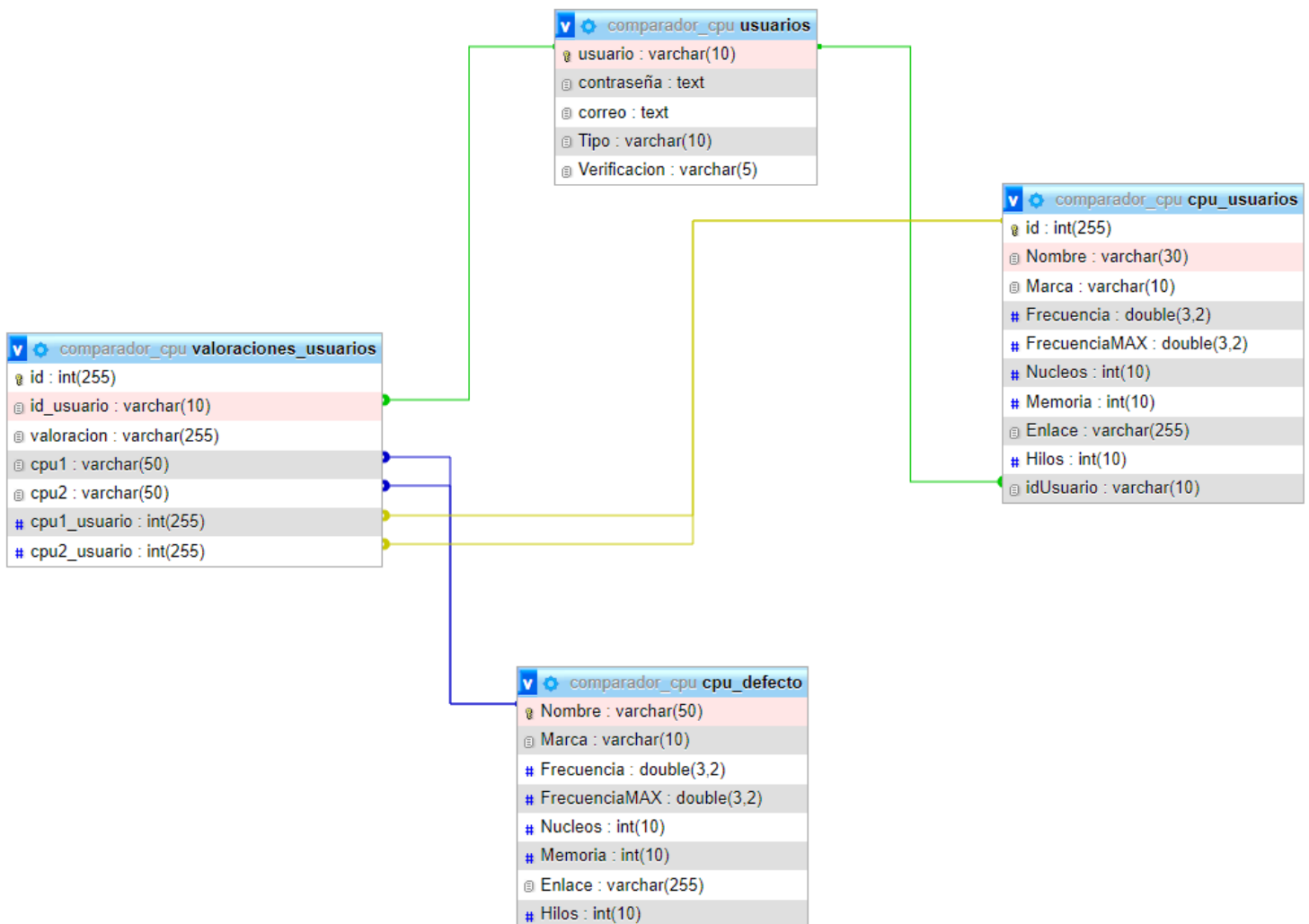
Una vez verificado en la aplicación puede acceder a todas las funcionalidades a través del menú (donde se puede crear, editar, borrar las CPU o ver/editar información de nuestro usuario o ver la ayuda de la aplicación) o del botón de comparar



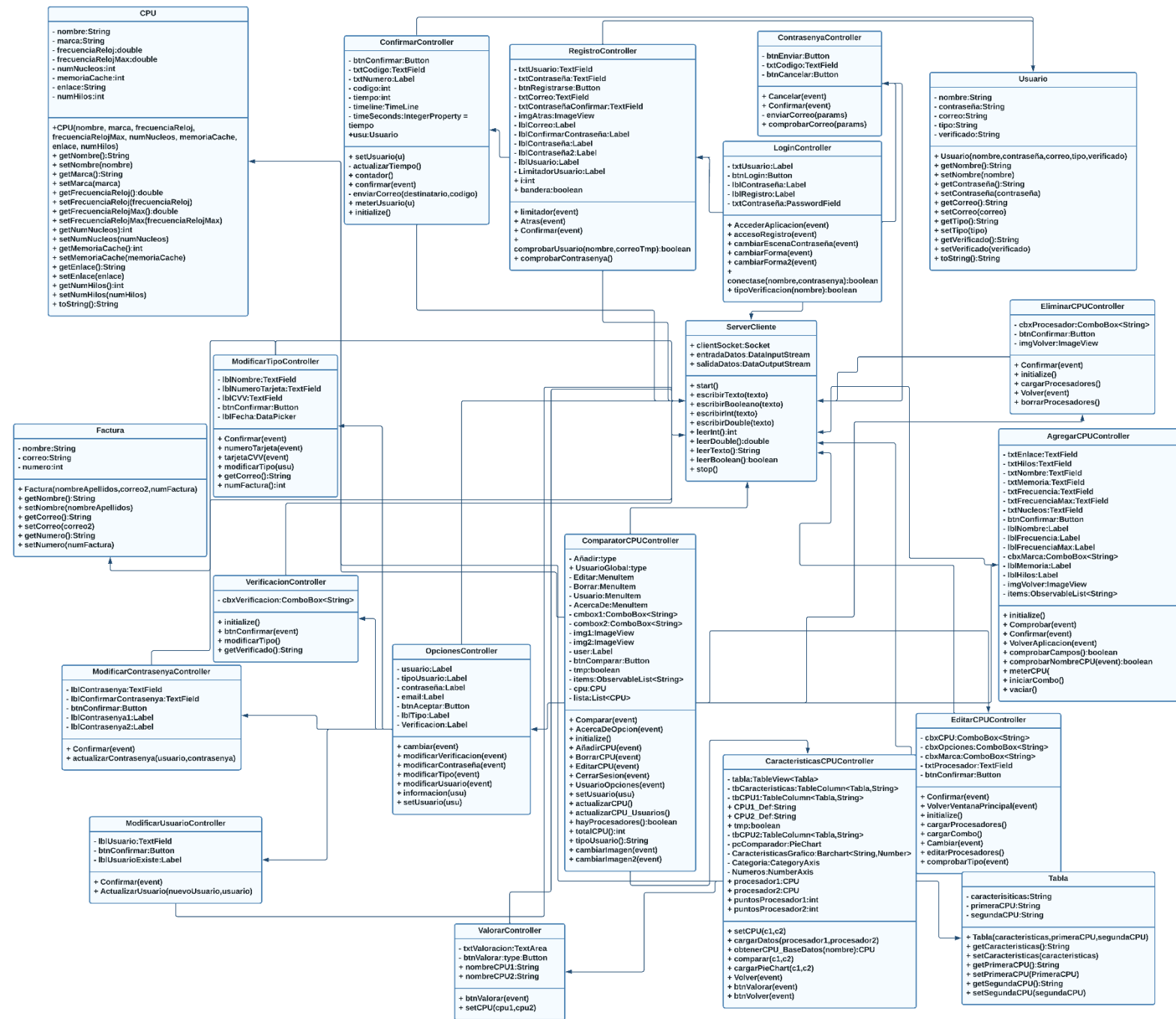
## 4.5 Diseño

### Diagrama lógico relacional

A continuación, se mostrará las tablas que forman parte de la aplicación de Toy Computer y las relaciones que tienen entre ellas para que la aplicación funcione perfectamente



## Diagrama de clases



## 5 Codificación

### 5.1 Tecnologías Elegidas

#### Java

Java es un lenguaje de programación de propósito general, concurrente, orientado a objetos que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Su intención es permitir que los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo.

#### ¿Por qué esta tecnología?

Básicamente porque está orientada a objetos y es un lenguaje de los más utilizados. Además de eso también es porque me permite crear mi propio servidor con Sockets y hilos, además de poder acceder a mi base de datos y poder enviar correos electrónicos. Y como plus tiene una herramienta llamada Maven con la que puedo crear mis propios informes



#### Maven

Maven es una herramienta de comprensión y gestión de proyectos de software. Basado en el concepto de un modelo de objetos de proyecto (POM), Maven puede administrar la construcción, los informes y la documentación de un proyecto desde una pieza central de información.

#### ¿Por qué esta tecnología?

Básicamente por que es de las pocas que te deja poder crear informes y en este caso como mi proyecto iba a tener un informe me decante por ella porque es única para Java.



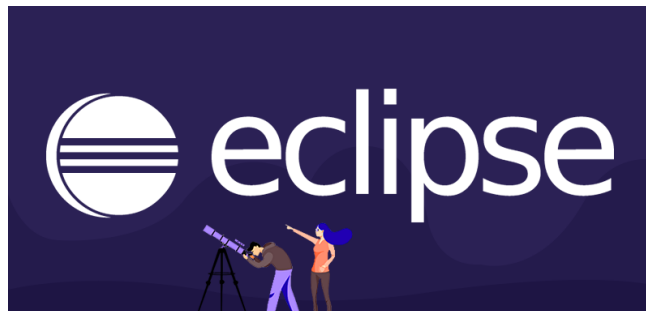


## **Eclipse**

Eclipse es una plataforma de desarrollo de código abierto, diseñada para ser extendida de forma indefinida a través de plug-ins. No tiene en mente un lenguaje específico, sino que es un entorno de desarrollo integrados genérico,

### **¿Por qué esta tecnología?**

Eclipse es una plataforma de desarrollo de código abierto, diseñada para ser extendida de forma indefinida a través de plug-ins. No tiene en mente un lenguaje específico, sino que es un entorno de desarrollo integrados genérico.



## **XAMPP**

XAMPP es un paquete de software libre, que consiste principalmente en el sistema de gestión de bases de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script PHP y Perl

### **¿Por qué esta tecnología?**

Elegí Eclipse ya que como dice la descripción es de código abierto, lo que quiere decir es que es gratis, además es uno de los más usados para programar Java

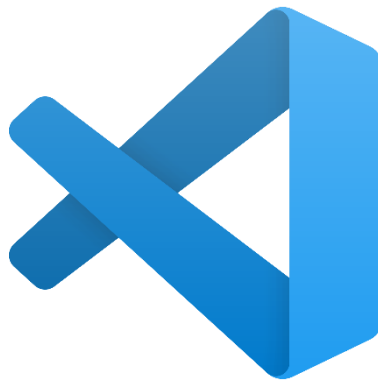


## **Visual Studio Code**

Visual Studio Code es un editor de código fuente desarrollado por Microsoft para Windows, Linux, macOS y Web. Incluye soporte para la depuración, control integrado de Git, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código.

### **¿Por qué esta tecnología?**

Básicamente porque necesitaba un editor para la creación de mis estilos para el proyecto y además para la creación de la página web y me decidí por este editor porque es la elección principal para desarrolladores web y JavaScript. Además de que es un editor muy elegante y completo.



## **Install4J**

Es un potente generador de instaladores Java multiplataforma que genera instaladores nativos y lanzadores de aplicaciones para aplicaciones Java

### **¿Por qué esta tecnología?**

Elegí Install4J ya que es uno de los pocos instaladores que tiene para Java además de ser gratis y visualmente su instalación es la que más me gusta en comparación a los otros productos gratuitos que hay en el mercado.



## **GIT**

Git es una herramienta que realiza una función del control de versiones de código de forma distribuida.

### **¿Por qué esta tecnología?**

Elegí GIT por ser la mejor en control de versiones ya que es muy potente, es software libre y con ella podemos mantener un historial completo de versiones del proyecto, como es en mi caso que hice varias versiones del proyecto como por ejemplo una versión inicial sin Maven y otra versión un poco más completa con Maven y más adelante más versiones con ampliaciones.



## **Scene Builder**

Scene Builder te permite diseñar, mediante un interfaz gráfico, las estructuras de las ventanas de las aplicaciones que queramos desarrollar usando JavaFX.

### **¿Por qué esta tecnología?**

Elegí Scene Builder porque lo usa todo el mundo que programa en JavaFX y como mi proyecto es de JavaFX me decanté por ella ya que es gratuita y simple para poder hacer el trabajo que se pide.



## 5.3 Documentación interna

- **Envío de correos electrónicos**

- Como vemos en la imagen tenemos un método que manda un correo al correo que le digamos y básicamente lo único que necesitamos es el correo del remitente y la contraseña y el correo del destinatario. Después crearemos una variable llamada Properties donde configuraremos el SMTP además de poner el correo y contraseña del remitente. Y más adelante crearemos una sesión donde le pasaremos la configuración SMTP, después crearemos el mensaje donde pondremos el mensaje personalizado. Por último, creamos el transporte donde nos conectaremos al SMTP y le enviaremos el mensaje

```
String destinatario = entradaDatos.readUTF();
String usuario = entradaDatos.readUTF();
String contra = entradaDatos.readUTF();

Properties props = System.getProperties();
props.put("mail.smtp.host", "smtp.gmail.com"); //El servidor SMTP de Google
props.put("mail.smtp.user", "comparadorcpu@gmail.com");
props.put("mail.smtp.clave", "PonerContraseña"); //La clave de la cuenta
props.put("mail.smtp.auth", "true"); //Usar autenticación mediante usuario y clave
props.put("mail.smtp.starttls.enable", "true"); //Para conectar de manera segura al servidor SMTP
props.put("mail.smtp.port", "465");
props.put("mail.smtp.socketFactory.port", "465");
props.put("mail.smtp.socketFactory.class", "javax.net.ssl.SSLSocketFactory");
props.put("mail.smtp.ssl.protocols", "TLSv1.2");
Session session = Session.getDefaultInstance(props);
MimeMessage message = new MimeMessage(session);

try {
    message.setFrom(new InternetAddress("comparadorcpu@gmail.com"));
    message.addRecipients(Message.RecipientType.TO, destinatario);
    message.setSubject("Recuperar Contraseña...");
    message.setText("Tu usuario es "+usuario+" tu contraseña es "+contra);
    Transport transport = session.getTransport("smtp");
    transport.connect("smtp.gmail.com", "comparadorcpu@gmail.com", "PonerContraseña");
    transport.sendMessage(message, message.getAllRecipients());
    transport.close();
}
catch (MessagingException me) {
    me.printStackTrace();
}
```

- **Crear factura**

- Lo primero será generar un objeto de tipo Factura donde le mandamos la información la ventana, después creamos una lista con el objeto Factura, mas adelante cargamos el fichero del informe, después lo compilamos. Cuando creamos los el dataSource le metemos los datos de la lista de Factura, después se llena el informe con los datos del dataSource y finalmente exportamos el informe a PDF

```
Map<String, Object> parameters = new HashMap<String, Object>();
Factura factura = new Factura (lblNombre.getText(), getCorreo(), numFactura());

List<Factura> listfactura = Arrays.asList(factura);
InputStream is = Main.class.getResourceAsStream("Factura.jrxml");

JasperReport report = JasperCompileManager.compileReport(is);
JRBeanCollectionDataSource dataSource = new JRBeanCollectionDataSource(listfactura);
JasperPrint print = JasperFillManager.fillReport(report, parameters, dataSource);
JasperExportManager.exportReportToPdfFile(print, "Factura"+ComparatorCPUController.UsuarioGlobal+".pdf");
try {
    Runtime.getRuntime().exec("rundll32 url.dll,FileProtocolHandler "+System.getProperty("user.dir")+ "//Factura"+ComparatorCPUController.UsuarioGlobal+".pdf");
} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
```

## 5.4 Manual de usuario

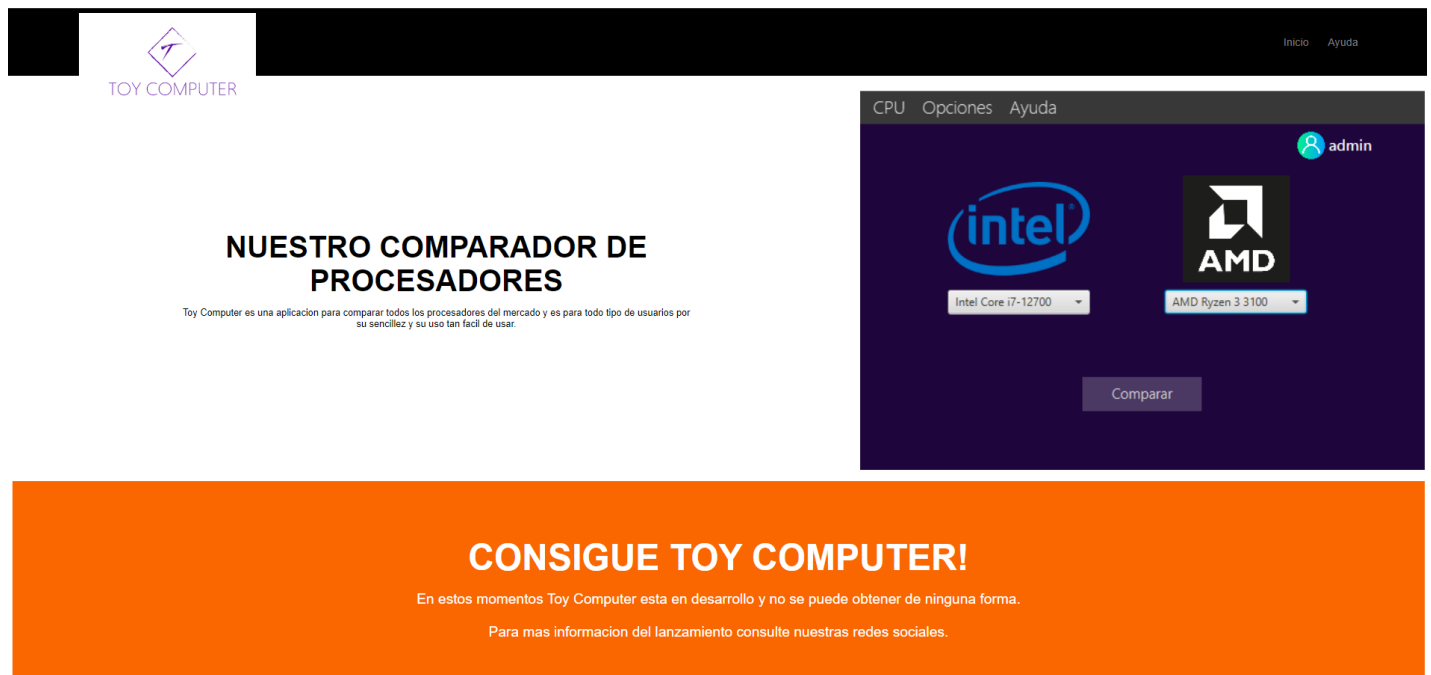
### Servidor

La aplicación a la que le damos al usuario está llamando constantemente a un servidor creado exclusivamente para la aplicación ya que tiene todos los métodos necesarios para que la aplicación vaya perfectamente cómo consultar, crear, editar o eliminar.

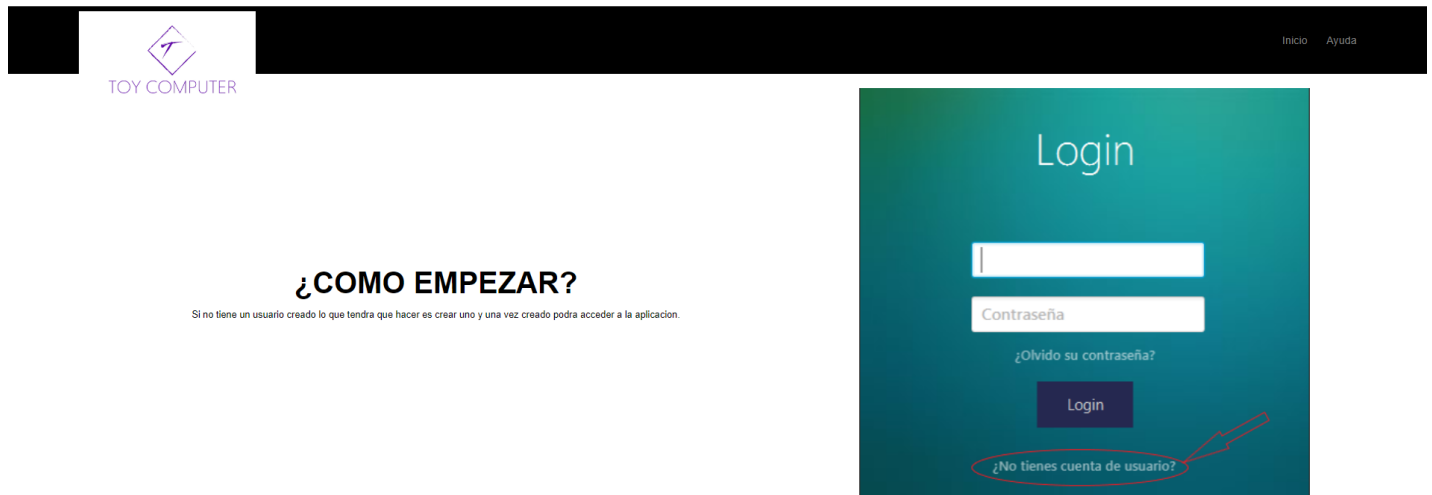
El servidor se dedicará a leer las peticiones del usuario y dependiendo de lo que le envíe el usuario le mandará una información u otra. Unos métodos interesantes que tiene el servidor es el envío de correos, agregar el usuario a la base de datos, agregar, actualizar, borrar los procesadores que ponga el usuario, comprobar si un usuario existe, etc....

### Página Web

1-Para poder consultar de que va nuestra aplicación deberemos ir a la página Web donde tendremos como página principal una introducción a la aplicación y más adelante la ayuda de usuario.

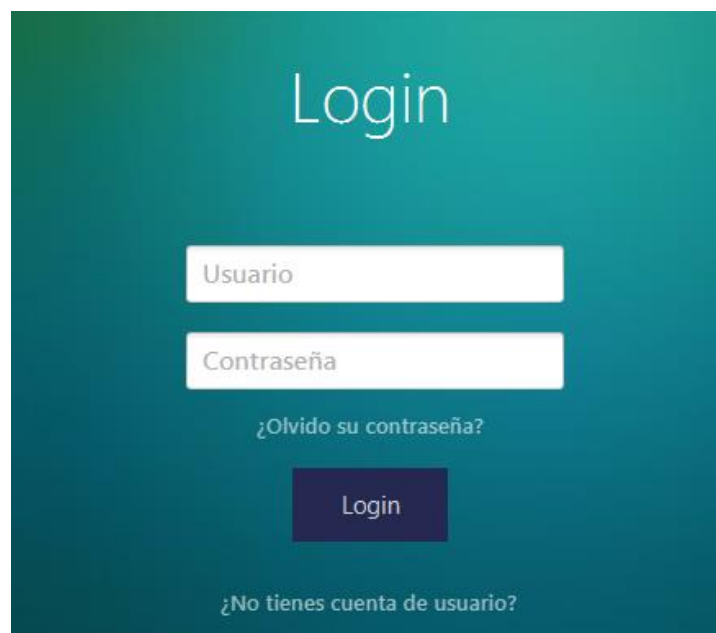


2-Después tendremos la ayuda para el usuario donde el usuario verá todo bien explicado por si tiene alguna duda de la aplicación



## Login y Registro

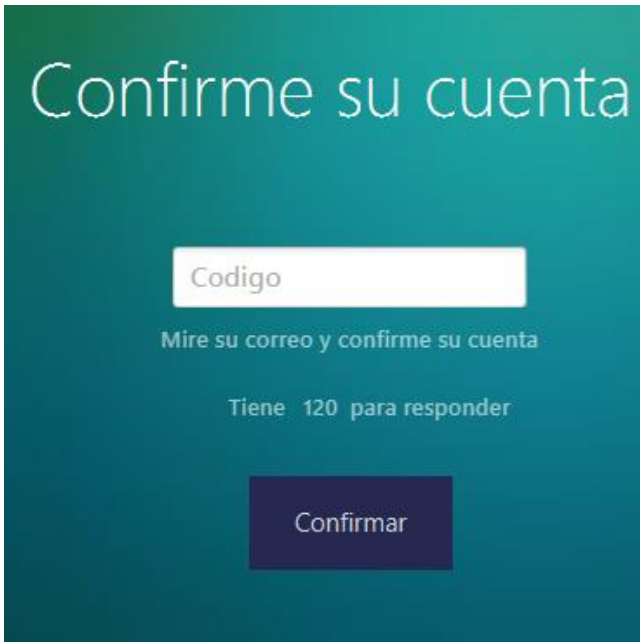
1-Lo primero que nos encontramos al iniciar la aplicación es el login, que si no tenemos ninguna cuenta creada deberemos ir a la ventana de registro de usuarios.



3-Una vez estemos en la ventana de registro deberemos poner nuestro correo, usuario y contraseña como pone en la ventana, además tendrá unas restricciones como que el usuario como máximo tendrá una longitud de 10 letras y la contraseña tiene que tener una longitud mínima y tener una mayúscula como mínimo y tener algunos atributos (.,@).

Una interfaz de usuario para el registro de un nuevo usuario. El fondo es un gradiente de verde azulado. En la esquina superior izquierda hay un botón circular con una flecha blanca hacia atrás. El título "Registrarse" está en blanco y centrado. Hay cuatro campos de entrada de texto blancos con bordes grises, uno encima del otro: "Correo Electronico", "Usuario" (con un contador "0/10" a la derecha), "Contraseña" y "Confirmar contraseña". Debajo de estos campos hay un botón rectangular de color verde oscuro con el texto "Confirmar" en blanco.

4-Una vez puestos los campos y le demos al botón, la aplicación le mandara los datos al servidor y el servidor nos mandará un correo con un código para verificar la cuenta y una vez pongamos el código se nos cerrara esa ventana y los datos se guardarán en la Base de Datos con la contraseña encriptada. Si no ponemos el código en 2 minutos se nos cerrara la ventana y si cancelara el registro.

Una interfaz de usuario para confirmar la cuenta. El fondo es un gradiente de verde azulado. El título "Confirme su cuenta" está en blanco y centrado. Hay un campo de entrada de texto blanco con el placeholder "Codigo". Debajo del campo, el texto "Mire su correo y confirme su cuenta" está en un color gris claro. Más abajo, el texto "Tiene 120 para responder" también está en gris claro. En la parte inferior hay un botón rectangular de color verde oscuro con el texto "Confirmar" en blanco.

## Recuperar Usuario

5-Si en algún momento se nos olvida el usuario o la contraseña la podremos recuperar yendo a la ventana de si no recuerda la contraseña y deberemos poner el correo, una vez puesto nos mandará un correo con toda la información de la cuenta.



Recuperar contraseña

Correo Electronico

Le enviaremos su usuario y contraseña al correo electronico

Cancelar Enviar

## Ventana Principal

6-Después de iniciar sesión tendremos la ventana principal de la aplicación donde tendremos los procesadores para compararlos entre ellos y un menú con diferentes opciones.



CPU Opciones Ayuda

Usuario


intel intel

Comparar



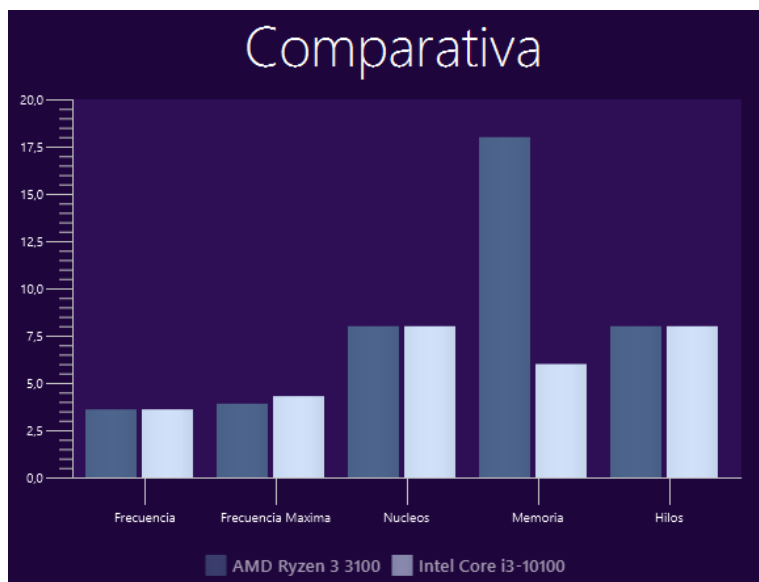
## Comparador

7-Si le damos al botón de comparar nos saldrá toda la información de los dos procesadores en detalle para ver lo que tiene cada uno, si bajamos más abajo tendremos un gráfico de barras para ver más en detalle qué característica es mejor. Por último, tenemos un gráfico circular para ver la comparación más en general y ver que procesador es mejor por lo general.



### Características

Características	CPU1	CPU2
Nombre	AMD Ryzen 3 3100	Intel Core i3-10100
Marca	AMD	Intel
Frecuencia	3.6	3.6
Frecuencia Maxima	3.9	4.3
Nucleos	8	8
Memoria	18	6
Hilos	8	8



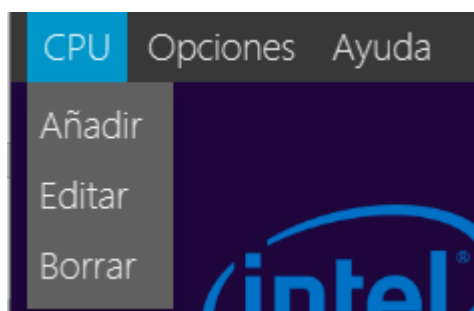
## Valoración

8-Una vez vista la comparación tenemos la opción de valorar la comparación donde tendremos una ventana para poner nuestra valoración que se guardará en la Base de Datos

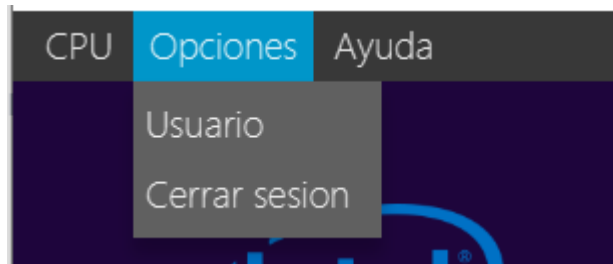


## Menú

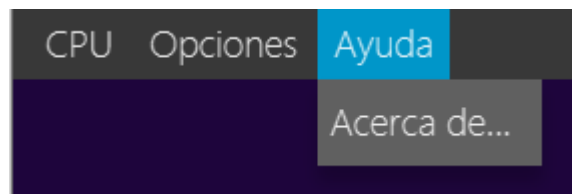
8-Si le damos a la primera opción del menú podremos gestionar los procesadores que tenga el usuario y podremos crear, editar o eliminar procesadores.



12-Si le damos a la segunda opción del menú podremos ver la opción de usuario o la de cerrar sesión.



15-Si le damos a la tercera opción podremos acceder a la ayuda de la aplicación que nos mandará a la página oficial de Toy Computer.

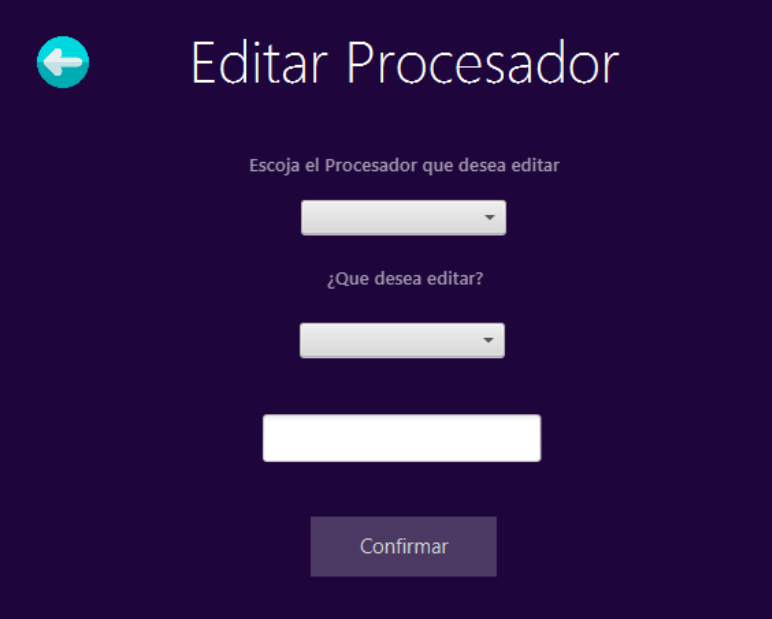


## Gestión de procesadores

9-Para poder crear un procesador deberemos rellenar los campos obligatorios que son todos menos la página web

A screenshot of a mobile application screen titled 'Añadir Procesador'. At the top left is a back arrow icon. Below the title, there is a vertical stack of input fields: 'Nombre', a dropdown menu, 'Frecuencia', 'Frecuencia Maxima', 'Nucleos', 'Memoria Cache', 'Hilos', and 'Enlace Web'. At the bottom of the form is a button labeled 'Añadir'.

10-Para poder editar un procesador deberemos poner primero que procesador queremos editar y más adelante poner que atributo deseamos editar y por último escribimos lo que vamos a editar



The screenshot shows a web form titled "Editar Procesador" with a back arrow icon in the top left. The form has a dark purple background. It contains three input fields: a dropdown menu labeled "Escoja el Procesador que desea editar", another dropdown menu labeled "¿Que desea editar?", and a text input field. A "Confirmar" button is located at the bottom right of the form.

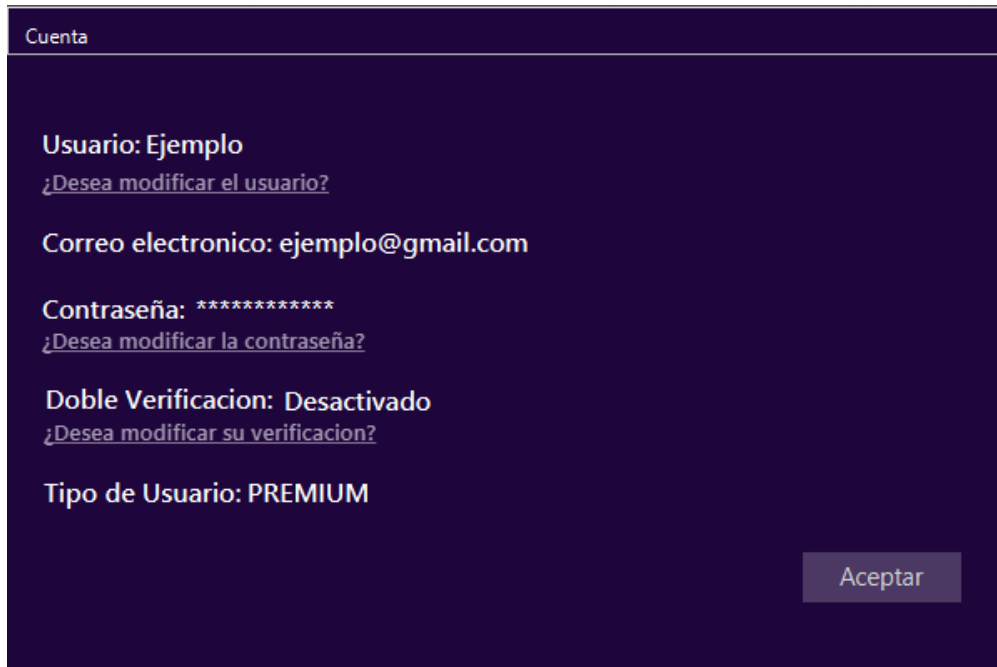
11-Para poder eliminar un procesador deberemos poner el procesador en cuestión y presionar el botón de confirmar



The screenshot shows a web form titled "Borrar Procesador" with a back arrow icon in the top left. The form has a dark purple background. It contains one input field: a dropdown menu labeled "Seleccione un procesador para poder eliminarlo". A "Confirmar" button is located at the bottom right of the form.

## Opción de usuario

14- En la información del usuario podremos editar la mayoría de las cosas del usuario como el nombre, contraseña, la verificación que es para que te manden correos con una verificación de correo o no y el tipo de cuenta que si tienes la cuenta de prueba significa que sólo puedes crear un procesador y si tienes el tipo premium puedes crear todos los procesadores que quieras.



14-Si le damos a modificar tipo nos mandara a una ventana donde tendremos que introducir los datos bancarios dónde se te cobrarán 10 euros y tendrás la opción de generar una factura o no.



## 7 Despliegue

### 7.1 Desarrollador

Para poder desplegar el proyecto he usado la herramienta llamada XAMPP en la que tengo configurada la Base de Datos y la página web.

Como primer punto tenemos la Base de datos tiene como uso guardar toda la información predeterminada que ve el usuario nada más iniciar la aplicación como la de guardar la información que nos da el usuario.

Y por último la página web que nos permite tener una visibilidad al público de nuestro producto y para que cualquier usuario pueda ver de qué se trata nuestra aplicación.

Además de lo comentado anteriormente deberemos iniciar el servidor personalizado creado en Java con sockets e hilos para poder recibir o enviar la información a la base de datos o a nuestro programa principal, ya que sin el servidor el cliente no podría hacer nada con la aplicación ya que para acceder a cada apartado el servidor debe recibir la información de la base de datos.

### 7.2 Cliente

Para poder desplegar la aplicación deberemos ir a la pagina web de nuestro proyecto y descargarnos el instalador que nos proporciona y una vez descargado lo que deberá hacer el cliente será iniciar el asistente de la instalación que es muy sencillo de seguir, una vez haya terminado el asistente se le habrá instalado la aplicación y tan solo deberá darle doble clic al ejecutable para poder desplegar nuestra aplicación.

### 7.3 Herramientas de apoyo

Para este proyecto use como herramienta de apoyo un sistema de control de versiones denominado GIT en el que por cada prueba exitosa que veía bien como nueva versión para el programa la subía a GIT para tener varias versiones de mi aplicación como por ejemplo tener la aplicación sin Maven y sin muchas implementaciones, otra versión con Maven y con más implementaciones, después otras versiones con más implementaciones y así sucesivamente.

## 7.4 Gestión de pruebas

El proyecto tuvo dos tipos de pruebas que son las pruebas de usuario de usuario y pruebas de rendimiento.

### Pruebas de usuario

A mediados de la aplicación se hizo una prueba de usuario para ver cómo iba un poco en general la aplicación en visión al usuario y se vio varios errores a cambiar para que el usuario se sintiera más cómodo y no se equivocara al crearse una cuenta o al añadir o editar un procesador y es por ello que se añadió más restricciones al proyecto y una versión más amigable como más imágenes, etc. Para finalizar realice una última prueba de usuario al finalizar la aplicación para ver si faltaba pulir algo más de la aplicación.

### Pruebas de rendimiento

Estas pruebas se realizaron a mediados de la aplicación para probar el sistema de correo y como se observó, al hacerlo en la misma aplicación se vio que tardaba en cargar considerablemente la ventana y es por ello que se optó en hacer un servidor a parte para organizar todos los métodos de la aplicación(incluyendo los de correo) y en la aplicación se hizo una clase llamada ClienteServidor donde tiene distintos métodos como la de iniciar el servidor, pararlo, leer los datos que llegan del servidor y mandar datos al servidor.

## 8 Conclusiones

### 8.1 Conclusiones Personales

Mi principal conclusión personal es la satisfacción de tener el proyecto realizado ya que desde que empecé a programar veía muy lejos el poder realizar este proyecto con tal nivel de dificultad.

En este proyecto he aprendido mucho, como la vista de desarrollador y cliente, el importante uso de GIT para tener un control de versiones del proyecto y un enfoque futuro del desarrollo para intentar no rehacer el proyecto desde un punto inicial por no haberlo previsto antes y sobre todo el refuerzo que te da el proyecto a nivel de programación ya que tienes que afrontar tu solo los problemas y subir el nivel y hacer cosas nuevas que antes no se te ocurrirían poner como por ejemplo en mi caso hacer una gestión de usuarios con una seguridad de envíos de correos.

Y por supuesto, también me ha servido para apasionarme aún más en el mundo de la programación y querer aprender cosas nuevas a diario para poder dedicarme a ello.

## 8.2 Posibles ampliaciones y mejoras

Se podrían hacer numerosas ampliaciones y mejoras. Como, por ejemplo:

- **Ampliaciones:**
  - Tanto el servidor de la aplicación como la base de datos y el servidor web no estuvieran en el mismo equipo.
  - Poder crear un instalador para el ejecutable ya que con el ejecutable que tengo del proyecto no se pudo hacer con Install4J ya que se estuvo haciendo pruebas y por la escasez de tiempo no se pudo llevar a cabo
- **Mejoras para el administrador:**
  - Una ventana especial para el administrador que pueda ver todos los usuarios, procesadores y valoraciones y que los pueda borrar o editar dicha información por si algún cliente tiene algún nombre inapropiado o pone procesadores que son inapropiados.
- **Mejoras en la interfaz gráfica:**
  - Usar las valoraciones para que los usuarios puedan dar su opinión de cómo mejorarían la interfaz ya sea de la aplicación de escritorio como la página web.
- **Nuevas funcionalidades:**
  - Por ahora tenía una en mente, básicamente es crear una nueva función en la que el usuario encuentre el procesador ideal para él, y para poder hacer eso la aplicación le preguntará varias preguntas y dependiendo de lo que responda se le recomendará uno u otro.



## 9 Bibliografía

### 9.1 Apuntes

#### [Tema 4 de programación de servicios y procesos](#)

Apuntes de José Zomeño Perona que me sirvieron de gran ayuda a la hora de crear el servidor para la aplicación

#### [Tema 9 de Desarrollo de interfaces](#)

Apuntes de José Manuel Gras que me sirvió de gran ayuda a la hora de crear el informe de la factura.

#### [Temario de Lenguaje de Marcas](#)

Apuntes de Carlos Tarazona Tarrega que me sirvió de gran ayuda a la hora de crear la página web del proyecto

#### [Tema 2 de Acceso a Datos](#)

Apuntes de Daniel Godoy que me sirvió de gran ayuda para resolver algunas dudas para poder acceder a la Base de Datos.

### 9.2 Páginas Web

#### <https://stackoverflow.com/>

Página muy interesante para poder buscar dudas generales de programación.

#### <https://docs.oracle.com/javafx/2/charts/css-styles.htm>

Página en la que me ayudó a la hora de hacer los estilos de los gráficos