FinBook

Contents

[**1. Explicación Del Ámbito del Proyecto** 2](#_Toc43138153)

[**1.1 Introducción Del Ámbito** 2](#_Toc43138154)

[**1.1.1 Hacia la Digitalización** 2](#_Toc43138155)

[**1.1.2 Liquidación Automatizada del IVA** 3](#_Toc43138156)

[**1.2 Factura** 5](#_Toc43138157)

[**1.2.1 Factura Genérica** 5](#_Toc43138158)

[**1.2.2 Factura Electrónica en México** 7](#_Toc43138159)

[**1.4 Explicación de Aplicación “Fintech”** 8](#_Toc43138160)

[**2. Motivación Y Objetivos Iniciales** 9](#_Toc43138161)

[**2.1 Motivación** 9](#_Toc43138162)

[**2.2 Objetivos Iniciales** 10](#_Toc43138163)

[**3. Desarrollo** 11](#_Toc43138164)

[**3.1 Metodología Aplicada** 12](#_Toc43138165)

[**3.2 Planificación** 12](#_Toc43138166)

[**4. Requisitos** 14](#_Toc43138167)

[**4.1 Requisitos de Hardware** 14](#_Toc43138168)

[**4.2 Requisitos de Software** 15](#_Toc43138169)

[**5. Arquitectura** 15](#_Toc43138170)

[**5.1 Arquitectura de FinBook** 15](#_Toc43138171)

[**5.1.1 Estructura Modular** 15](#_Toc43138172)

[**5.1.2 Arquitectura Publisher/Subscriber** 15](#_Toc43138173)

[**5.2 Arquitectura del Módulo desarrollado** 15](#_Toc43138174)

[**5.2.1 Arquitectura Multi-Agente** 15](#_Toc43138175)

[**5.2.2 Estructura Interna** 15](#_Toc43138176)

[**6. Competencias Especificas Cubiertas** 15](#_Toc43138177)

[**7. Generador Sintético de datos** 16](#_Toc43138178)

[**7.1 Introducción** 16](#_Toc43138179)

[**8. Conclusiones** 17](#_Toc43138180)

[**9. Bibliografía** 17](#_Toc43138181)

# **1. Explicación Del Ámbito del Proyecto**

## **1.1 Introducción Del Ámbito**

### **1.1.1 Hacia la Digitalización**

En la economía actual, las facturas se generan principalmente en papel en la mayoría de los sectores y países, esto se debe a que es el método mas usado durante las últimas décadas. En el siglo XX y sobre todo antes de la creación del ordenador y las redes de internet, era la mejor opción para generar facturas, sin embargo, ¿sigue siendo actualmente la manera óptima?, ¿o existen nuevas posibilidades para gracias a las nuevas tecnologías? Esa es una pregunta que se responderá durante la presentación de este trabajo.

Como respuesta se ha originado un nuevo tipo de factura, la factura electrónica. Esta factura al contrario de la común factura en papel se crea, se guarda y se distribuye virtualmente lo que permite una mayor facilidad y seguridad en mundo socioeconómico actual, como veremos a continuación. La factura electrónica funciona como cualquier fichero o archivo de datos, guarda información, pero en este caso guarda los datos de una factura. El formato de la factura electrónica más extendido es XML (eXtensible Markup Language) debido a su simplicidad y la claridad a la hora de leer una factura por una persona que conozca un poco el formato o para un programa que la procese. Este uso de formato genérico permite como acabo de nombrar, una “generalización” de la factura, es decir, permite o más bien obliga tener un conjunto de reglas rígido que seguir para la creación de las facturas formando una estructura común y constante entre todas ellas y por lo tanto una mayor facilidad a la hora de leerlas por una persona o procesarlas por un ordenador.

Esta ventaja de la factura electrónica ayuda enormemente a sectores como la fiscalía, hacienda o la industria, así como cualquier empresa de servicios o venta de productos, es decir cualquier entidad que genere facturas, debido a su facilidad de creación, distribución y uso de las mismas. Además, otra ventaja que tiene la factura electrónica es la seguridad que trae de por si el uso de estas ya que controla el flujo de facturas generados virtualmente y permite controlar las evasiones fiscales que trae consigo las facturas en papel que nunca son declaradas.

Por último, tenemos la ventaja que se va a aprovechar y demostrar en este proyecto, la capacidad de procesar y analizar estas facturas. Esto se debe como ya nombramos anteriormente, al formato que usa, que favorece el procesamiento y el análisis de estas ya que la estructura es única por lo que todas las facturas se pueden leer y obtener datos de la misma manera, de forma iterativa y constante de todas y finalmente generar un reporte de todas estas mucho más rápido que una persona experta en esto. Este sector que automatiza el análisis de datos se denomina Big Data, que como su nombre indica, permita analizar grandes volúmenes de datos de forma rápida y eficaz en todo tipo de sectores, uno de ellos la facturación. Esta es una de las grandes ventajas de la factura electrónica con respecto a la de papel, y es el Big Data el futuro en muchos sectores que trabajen con grandes volúmenes de datos.

La factura electrónica se esta empezando a extender actualmente y algunos países como México, Chile, Brasil e Italia están empezando a usarlas como en todos los sectores del país. En algunos como México están propiciadas por tener un mayor control de estas y evitar robos de dinero y fraudes fiscales. Este número de países sigue aumentando durante estos años y se espera que sea la factura mas usada y extendida en el futuro, como tantos otros avances que ha propiciado la ciencia y la tecnología.

### **1.1.2 Liquidación Automatizada del IVA**

El Impuesto de valor añadido es un tributo que grava la entrega de bienes y las prestaciones de servicios. La liquidación del IVA se ha de realizar trimestralmente, si bien las grandes empresas lo realizarán mensualmente. El modelo fiscal que hay presentar es el modelo 303 y se presenta telemáticamente a la Agencia Tributaria. Cada entrega de bienes y servicios estará gravada sobre el impuesto sobre valor añadido, de tal manera que la empresa cuando venda sus mercancías deberá consignar en la factura el importe del IVA repercutido. De otro lado, cuando la empresa compre bienes y servicios, el proveedor le entregará una factura por el importe gravado por el IVA, denominado IVA soportado. La base imponible del IVA se determina por el precio facturado menos todos los descuentos. A dicho importe se le aplicará un tipo de gravamen el cual dependerá del tipo de operación.

La liquidación del IVA consiste en determinar la diferencia entre el IVA repercutido y el IVA soportado. Tal diferencia, si es positiva, deberá ingresarse en la Agencia Tributaria a través del referido modelo 303. Si la diferencia es negativa, se generará un derecho de cobro que se compensará en la siguiente liquidación, a menos que nos encontremos al cierre del ejercicio económico.

La información fiscal que se desprende de las operaciones económicas que vienen gravadas por el IVA también se representan en la contabilidad, el IVA soportado que es deducible del IVA repercutido, es decir, que minora la base imponible del impuesto vendrá representado en una cuenta denominada hacienda pública, IVA soportado del grupo 4 del cuadro de cuentas del plan general de contabilidad. Cuando el IVA soportado no puede deducirse del IVA repercutido, formará parte del precio de adquisición y por tanto de la cuenta de gasto o de activo. Ello significa que la empresa al no poder reducir el IVA soportado no tendrá ninguna cuenta específica en los estados financieros. Además, los comerciantes minoristas que sean personas fijas pueden acogerse al recargo de equivalencia que supone pagar un IVA mas alto a cambio de no tener que presentar declaraciones del IVA a hacienda, de esta forma el IVA soportado se paga directamente por sus compras al proveedor y se simplifica la gestión porque no hay que llevar libros de IVA ni guardar facturas. Esto se relaciona con sector de la restauración ya que es un negocio minorista y es el sector que principalmente se va a enfocar mi Trabajo de Fin de Grado.

Tras una breve introducción de la liquidación, llega unos de los enfoques del Big Data en el sector fiscal, la liquidación automática del IVA, que consiste en el análisis y cálculo de IVA de repercutido y el IVA soportado de forma automática, rápida y eficaz del precio facturado de una empresa o de cualquier entidad económica que ofrezca bienes o servicios. Además de suponer una enorme facilidad para la Agencias tributarias a la hora comprobar esos datos, supone un ahorro de tiempo a la hora de presentar el modelo fiscal 303 ante la agencia tributaria telemáticamente.

Este avance permite un ahorro económico importante en las empresas y una facilidad para estas a la hora de hacer las desgravaciones ya que nos ahorramos “la mano de obra” que se encarga de realizar tal análisis.

## **1.2 Factura**

### **1.2.1 Factura Genérica**

Una factura es un documento mercantil que refleja toda la información de una operación de compraventa. El sistema de facturación está basado en dos tipos de facturas:

• Factura completa u ordinaria: posee los siguientes campos:

* Número y, en su caso, serie. La numeración ha de ser correlativa dentro de cada serie.
* Fecha de su expedición.
* Fecha de realización de la operación documentada en la factura (si no coincide con la fecha de expedición).
* Número de identificación fiscal (NIF), nombre y apellidos o denominación social tanto del obligado a expedir la factura como del destinatario de las operaciones.
* Domicilio del expedidor y del destinatario.
* Descripción de las operaciones y todos los datos necesarios para la determinación de la base imponible y su importe y precio unitario sin impuesto, así como cualquier descuento o rebaja que no esté incluido en el precio unitario.
* Tipos impositivos aplicados a las operaciones.
* Cuota tributaria, que deberá consignarse por separado.
* Sede de actividad o establecimiento si alguna de las partes o ambas disponen de varias sedes fijas del negocio.
* Moneda
* Lengua, Las facturas pueden expedirse en cualquier lengua. Existe la posibilidad de exigir traducción al castellano u otra lengua oficial.
* Firma y localidad (opcional).
* Medios de expedición, papel o formato electrónico que permita garantizar la autenticidad de su origen, la integridad de su contenido y su legibilidad.

• Factura simplificada: Las facturas simplificadas, que sustituyen a los tiques, tienen un contenido más reducido que las facturas completas y por lo general, están permitidas solo para facturas no superiores a unos 400 euros.

También existe la denominada factura proforma que es simplemente un anticipo de la posterior factura real o la factura recapitulativa que se genera por periodos mensuales, es decir, pueden incluirse en una sola factura las operaciones realizadas en distintas fechas para un mismo destinatario. Deben ser expedidas, como máximo, el último día del mes natural en el que se realizaron las operaciones documentadas en la factura.

La factura electrónica es el documento electrónico equivalente a la factura en papel. Consiste en una transmisión telemática entre un emisor y un receptor por medio de dispositivos electrónicos capaces de generar las facturas adecuadas, que será la que estudiaremos más a fondo en el caso de México.

Las entidades o empresas deberán conservar los siguientes documentos:

• Las facturas recibidas.

• Las copias de las facturas expedidas.

La obligación de conservación puede cumplirse mediante la utilización de medios electrónicos. Las obligaciones se pueden cumplir materialmente por un tercero que actúe en nombre de la empresa. El Código de Comercio obliga a los empresarios a conservar los libros contables, la correspondencia, la documentación y los justificantes de sus operaciones durante seis años, a partir del último asiento realizado en los libros.

### **1.2.2 Factura Electrónica en México**

La factura electrónica en México es ya una realidad en este país, que debido a varias causas de índole fiscal y económica ha traído consigo indirectamente un gran cambio en la manera de crear y administrar las facturas.

La facturación electrónica en México se inició en el 2004, cuando el Servicio de Administración Tributaria (SAT) creó el marco legal que definió la implantación del Comprobante Fiscal Digital a través de Internet (CDFI). Con el tiempo se ha convertido en uso obligatorio para la contabilidad electrónica de México y actualmente es utilizado en el 100% de las transacciones económicas. El éxito de este sistema implementado en México ha hecho que otros gobiernos de Latinoamérica estén siguiendo sus pasos y trabajan en la implantación de una facturación electrónica propia. El sistema de facturación electrónica de México ha destacado gracias a su rápida difusión, que le ha permitido extenderse en pocos años por toda la sociedad mexicana.

El CFDI es un documento XML el cual está diseñado para cumplir con la especificación y exigencias del SAT y son las siglas que indican 4 características importantes de esta factura:

* Comprobante: justifica ante la administración fiscal que realmente se efectuó un proceso de compra venta y que se pagaron los impuestos designados.
* Fiscal: garantiza que el comprobante se encuentra en el marco legal, que tiene que ajustarse a la legislación aplicable y enviarse en plazo al SAT. Este punto es muy importante ya que era una las causas mas importantes que motivaban este cambio en la facturación debido a los fraudes fiscales y la evasión de impuestos.
* Digital: las facturas están creadas en un sistema binario y son almacenadas en un disco duro donde se puede hacer una búsqueda de una factura, de un archivo y encontrarla en segundos tan solo indicando su nombre, fecha o alguna otra característica de esta. Además, se puede mantener archivada fácilmente sin llegar a perderla en ningún momento.
* Internet: Las empresas que emiten CFDI, las emiten con su propio sistema de facturación lo que genera una Descentralizar la facturación y, por lo tanto, se ahorra un gran capital en términos de seguridad informática. Por lo tanto, este sistema permite emitir CFDI desde todo tipo de dispositivos electrónicos como un celular en un restaurante o en una venta hecha fuera de las oficinas.

La Plataforma Interna del SAT es la encargada de coordinar, administrar y sincronizar a todos los proveedores que están autorizados, también se encarga de administrar la información concerniente a cada uno de los contribuyentes y sus respectivas facturas electrónicas asociadas. Poseen un buen sistema de seguridad basado en el sistema de criptografía de clave pública, por lo que le otorga a cada contribuyente algunas claves, una pública y otra privada.

Como conclusión de este sistema, podemos ver que tenemos un sistema bien desarrollado y con vistas de futuro que con el tiempo el resto de los países seguirá y posiblemente se convierta en la base de la facturación en el mundo.

## **1.3 Explicación de Aplicación “Fintech”**

Tecnología Financiera o Fintech es una industria que aplica las nuevas tecnologías de la información, concretamente el creciente sector del Big Data, al sector financiero. Esta industria este dividida en cuatro segmentos:

* Herramientas de operación y medios de pago
* Conocimiento del cliente y Big Data
* Seguridad e identificación de personas
* Dinero electrónico

Las aplicaciones de Tecnología financiera (o Aplicación Fintech) es un tipo de aplicación financiera que proporciona herramientas tecnológicas a este sector para poder facilitar la labor de los usuarios de la aplicación a la hora de tratar con grandes volúmenes de datos.

Estas aplicaciones tienen como objetivo automatizar procesos que hoy en día se producen manualmente como la facturación, mediante la digitalización de los datos y el procesamiento automático de estos por parte de procesadores y células inteligentes. Para ello, esta entidad inteligente tiene la necesidad de poder procesar lenguaje natural y derivados, o leer archivos usados para el almacenamiento de datos, como XML o JSON, para poder realizar adecuadamente su labor, esto suele llevar detrás el desarrollo de una inteligencia que pueda entender tales lenguajes. Esta tecnología tiene diversos como:

* Pagos y remesas.
* Préstamos.
* Gestión de finanzas empresariales.
* Gestión de finanzas personales.
* Crowdfunding (financiamiento de proyectos).
* Gestión de inversiones.
* Seguros.
* Educación financiera y ahorro.
* Soluciones de scoring, identidad y fraude.
* Trading y mercados.

# **2. Motivación Y Objetivos Iniciales**

## **2.1 Motivación**

Este proyecto ha supuesto un enorme interés desde el inicio del desarrollo del mismo, debido a varios factores, principalmente por el hecho de crear por primera vez una aplicación Fintech, un campo completamente novedoso para mí, siguiendo una arquitectura de Big Data, otro sector completamente nuevo que, aplicado al sector financiero, supuso estudiar un nuevo enfoque de la programación que conllevaría aprender mucha información nueva para el desarrollo del proyecto.

Otro aspecto que destacar es el tamaño del proyecto completo, que ha sido un nuevo reto para mí, porque el haber desarrollado un módulo de este proyecto ha llevado mucho tiempo, además de la posterior integración con el resto de mis compañeros. Es sin duda el proyecto más ambicioso que he realizado hasta el momento y al que le he dedicado mucho esfuerzo y dedicación con entusiasmo y emoción.

Centrándonos en mi proyecto personal, el hecho de realizar una simulación en la que los elementos de este interactúen entre si realizando unas acciones realistas y aceptables dentro del marco de la normalidad, era de antemano, un reto, el cual me motivó enormemente ya que el poder hacer algo tan avanzado individualmente y solo con la ayuda de mi tutor era una idea que parecía imposible, debido a mi nivel inicial al empezar el proyecto.

Otra motivación proviene de poder realizar este proyecto partiendo desde cero, sin ningún tipo de trabajo ya hecho, teniendo yo la responsabilidad de la arquitectura y las dependencias a usar, realizando todas las funcionalidades necesarias de la simulación que se necesitaba. Además, siguiendo con las ideas del profesor, quería un proyecto con el mínimo de dependencias externas, ya que son factores que luego pueden afectar directamente al proyecto, por lo tanto, todos los distintos módulos y funcionalidades que iba a necesitar en el futuro fueran implementadas, en la medida de lo posible, todas por mí. Sin embargo, tenía claro que quería buscar un equilibrio en el que también se puedan ciertas librerías ya existentes que no supongan grandes dependencias en mi proyecto final.

## **2.2 Objetivos Iniciales**

Este proyecto tiene como propósito generar un Publisher (feeder) de la Plataforma de Datos, es decir, un generador de datos financieros realistas con el fin de crear un generador de datos sintéticos que aporte tanto facturas de clientes, como del personal o los proveedores de materias primas involucrados en el desarrollo económico de las empresas.

Estas empresas o agentes pondrán a disposición un gran volumen datos financieros que serán utilizados por otros proyectos posteriormente, para su estudio y análisis. Este objetivo será enfocado al ámbito de la restauración, y será posible a través de un ciclo iterativo basado en tres fases, Modelación, Simulación y Validación:

Modelación: Consiste en estructurar y abstraer los datos, elementos y eventos de la realidad para poder simularlos posteriormente, ya sea el consumo de restaurantes por parte de clientes y las facturas generadas, además de las nóminas y la compra de materias primas.

Simulación: añadir estos elementos dentro de la simulación y que interactúen entre sí mediante compra/venta de materias primas, contratación/despido de personal y consumo de los clientes.

Validación: Análisis de los datos recibidos de la simulación y comprobación de que lo datos recibidos son acordes y se reflejan en la realidad.

A partir de ahí, han surgido nuevos requisitos que cumplir, como la generación de facturas de servicios, de devolución de productos, además de añadir suficientes elementos a la simulación, como contratos de trabajadores, nacimiento y muerte de individuos y empresas, jubilación de los trabajadores, pago de impuestos por parte de las empresas, pago de hipoteca y concluyendo con aspecto muy importante en las simulaciones, como es la toma de decisiones adecuadas según la situación, por parte de los individuos y empresas de la simulación, todo esto para mejorar el realismo de la simulación.

# **3. Desarrollo**

## **3.1 Metodología Aplicada**

El lenguaje de programación que utilizaremos principalmente para elaborar la aplicación “FinTech” será Java, utilizando el IDE “IntelliJ”. Se hará uso de la herramienta “Intino”, que permite la creación de la plataforma de datos. Se usará una arquitectura de programación funcional, aplicando un desarrollo guiado por pruebas (TDD – Test-Driven Development) e iterativo, en el que se establecerán una serie de objetivos/hitos a cumplir en cada iteración. Cada iteración tendrá una duración aproximada de 2 semanas, en la que se le dedicará un total de 50 horas en cada una.

Se hará uso de un “bus” de mensajería proporcionado por “Intino” para comunicar las distintas partes funcionales del proyecto realizadas por los distintos participantes. Cabe destacar que “Intino” usa la tecnología JMS (Java Messaging Service) que consiste en una solución para el uso de colas de mensajes: es un estándar de mensajería que permite a los componentes de aplicaciones crear, enviar, recibir, y leer mensajes, y que permite una comunicación confiable de manera asíncrona.

Para el control de versiones del proyecto se utilizará la tecnología “Git”, y dada la naturaleza grupal del proyecto, se aprovechará el modelo de ramificación “GitFlow” para facilitar el desarrollo concurrente y organizado por parte de todos los integrantes del proyecto.

## **3.2 Planificación**

He dividido mi proyecto en 3 fases, estudio del ámbito y las tecnologías que se usaran, desarrollo e implementación del proyecto, y validación y prueba del proyecto en el entorno de usuario. Por ultimo, se encuentra la documentación final.

|  |  |  |
| --- | --- | --- |
| ***Fases*** | ***Duración Estimada (horas)*** | ***Tareas***  ***(nombre y descripción, obligatorio al menos una por fase)*** |
| Primera Iteración: Estudio previo y Análisis de las tecnologías y la metodología que se usaran. | 50 | Iteración 1.1: Estudio y búsqueda de un generador de personas aleatorias usadas como clientes de la simulación |
| Iteración 1.2: Estudio de la tecnología JSOUP para la obtención de datos de restaurantes la página web TripAdvisor, usados como restaurantes de la simulación. |
| Iteración 1.3: Análisis de datos relacionados con la parametrización de la simulación. Por ejemplo, distribuciones de sueldos, del tamaño del restaurante, del grado de consumo de restaurantes por parte de la población y el número de personas que son invitadas en función de sus relaciones sociales. |
| Iteración 1.4: Estudio de las facturas electrónicas generadas en México usando CFDI, que es un documento XML que cumple con las especificaciones proporcionadas por la SAT (Servicio de Administración Tributaria). |
| Segunda Iteración: Diseño, Desarrollo e Implementación de la Simulación | 250 | Iteración 2.1: Desarrollo de un módulo que extraiga de internet, una lista de restaurantes con sus datos para ser usados posteriormente en la simulación. |
| Iteración 2.2: Desarrollo de un módulo que extraiga de un archivo CSV, una lista de clientes y empresas secundarias con sus datos para ser usados posteriormente en la simulación. |
| Iteración 2.3: Guardar los datos de clientes, empresas secundarias y restaurantes en una base de datos fácilmente accesible para su rápida lectura durante la simulación. |
| Iteración 2.4: Diseño y desarrollo de un sistema de tiempo que permita controlar un calendario artificial con días meses y años, en el que sucederán todos los actos dentro de la simulación. |
| Iteración 2.5: Diseño y desarrollo de un sistema de simulables que interactuarán independientemente cada día de la línea de tiempo, todos los elementos que formen parte de la simulación serán simulables. |
| Iteración 2.6: Diseño e implementación de un sistema de consumo de restaurantes por parte de los clientes con cierta temporalidad parametrizada, que genere facturas de comidas y que sean incluidas en sus datos financieros para su posterior análisis, esto generará facturas de compras. |
| Iteración 2.7: Adición de empresas relacionadas con el sector de la Restauración como proveedores y empresas de servicios que provean de materias primas y servicios a los restaurantes, esto generará facturas de compras de productos, servicios y devoluciones. |
| Iteración 2.8: Desarrollo de un sistema de contrataciones de los trabajadores que trabajaran en los restaurantes a cambio de un salario que influya también en los datos financieros de la empresa, esto generará facturas de nóminas. |
| Iteración 2.9: Diseño e implementación de un sistema de estrategias a la hora de tomar decisiones en las distintas situaciones que se presentan. Por ejemplo, a la hora de contratar los trabajadores, los restaurantes elegirán quien contratar según la situación. |
| Iteración 2.10: Desarrollo de un sistema de contrataciones de los trabajadores que trabajaran en los restaurantes a cambio de un salario que influya también en los datos financieros de la empresa, esto generará facturas de nóminas. |
| Iteración 2.11: Implementación de un sistema de impuestos y pagos de hipoteca del local por parte de las empresas. |
| Iteración 2.12: Creación de una vista en web para mostrar la simulación en la que el usuario puede ver los eventos que se producen, e interactuar y cambiar los ajustes de este. |
| Tercera Iteración: Evaluación, Validación y Prueba de la Simulación en el Entorno de Finbook | 60 | Iteración 3.1: Testeo y búsqueda de errores. |
| Iteración 3.2: Integración del Publisher con la Plataforma de datos |
| Cuarta Iteración Documentación y Presentación del Proyecto realizado | 40 | Iteración 4.1: Documentar todo el proyecto realizado. |
| Iteración 4.2: Preparación de la presentación del trabajo. |

# **4. Requisitos**

## **4.1 Requisitos de Hardware**

* Ordenador de Sobremesa: herramienta fundamental y principal para el desarrollo de este proyecto, ya que ha sido el lugar de creación, desarrollo y almacenamiento de este, además del almacenamiento y uso de todos los programas necesarios para el desarrollo de la simulación.
* Ordenador Portátil: herramienta adicional para continuar con el desarrollo de este proyecto cuando no me encuentro cerca del lugar de trabajo principal.
* Conexión a internet: herramienta esencial en todo tipo de proyectos hoy en día, para la documentación y la descarga de aplicaciones necesarias.

## **4.2 Requisitos de Software**

* J**ava:** lenguaje de programación principal usado durante el desarrollo y la implementación de la simulación.
* **Javascript + JSP + CSS:** lenguajes y herramientas principales en el entorno grafico y visual del usuario, ya que he usado el entorno web para mostrar la simulación.
* **Intellij IDEA:** programa principal de desarrollo del proyecto tanto en la parte de servidor con Java como la parte del cliente en web.
* **Tomcat Server:** herramienta de Tomcat que he usado como servidor del proyecto.
* **Tomcat Web-socket:** herramienta de Tomcat para el desarrollo de los sockets entre el servidor y el cliente con el pasar información directa entre archivos de Javascript y archivos Java.
* **TripAdvisor:** página web usada para leer los datos de todos los restaurantes activos en Gran canaria, a través de la librería Jsoup. (9)
* **Jsoup:** librería para “Web Scrapping”, es decir, leer datos de los archivos html de las páginas web. (5)
* **Generador de datos aleatorios:** he usado un generador de datos para crear los perfiles de las distintas empresas y personas de la simulación, exceptuando los restaurantes. (6)
* **Jquery:** librería para complementar en Javascript en la administración de la web. (8)
* **DB Browser for SQLite:** programa para administrar leer, y guardar los datos que se manejaran durante la simulación, esto incluyen datos de clientes, empresas secundarias, facturas y restaurantes. (10)

# **5. Arquitectura**

## **5.1 Arquitectura de FinBook**

### **5.1.1 Estructura Modular**

### **5.1.2 Arquitectura Publisher/Subscriber**

## **5.2 Arquitectura del Módulo desarrollado**

### **5.2.1 Arquitectura Multi-Agente**

### **5.2.2 Estructura Interna**

# **6. Competencias Especificas Cubiertas**

Código/s:

- TFG01: Ejercicio original a realizar individualmente y presentar y defender ante un tribunal universitario, consistente en un proyecto en el ámbito de las tecnologías específicas de la Ingeniería en Informática de naturaleza profesional en el que se sinteticen e integren las competencias adquiridas en las enseñanzas.

- CII01: Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación y normativa vigente.

- CII06: Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.

- CII07: Conocimiento, diseño y utilización de forma eficiente los tipos y estructuras de datos más adecuados a la resolución de un problema.

- CII08: Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.

- CII013: Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.

- CII014: Conocimiento y aplicación de los principios fundamentales y técnica

- CII015: Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.

- CII016: Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la ingeniería de software.

- IS03: Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles.

- IS04: Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales.

# **7. Generador Sintético de datos**

## **7.1 Introducción**

La explicación de este proyecto realizado junto a sus funcionalidades se dividirá en 2 partes: la simulación y el entorno de usuario que interactuará con la simulación. Cabe destacar que la simulación y el modelo de este se ha desarrollado de tal manera que sea completamente independiente al exterior, es decir, que no tiene ningún tipo de dependencia con el servidor usado el cliente desarrollado, las acciones del usuario o las librerías usadas. No tiene ninguna dependencia exterior con las librerías usadas por lo que, si en el futuro alguna de las librerías o herramientas usadas se quedan obsoletas, no afectará en ningún modo al núcleo del proyecto, a la simulación.

En la primera parte hablaremos del núcleo de la simulación, el modelo del proyecto donde se ejecuta internamente todos los procesos de la simulación y en el que cada uno de los agentes de la simulación realiza sus acciones pertinentes.

En la segunda parte, mostraré la interfaz gráfica del usuario, las funcionalidades que tiene y las opciones en las que el usuario puede influir en la simulación.

Antes de empezar es importante aclarar una palabra que se usará a menudo durante esta explicación, que es el concepto de simulable. Un simulable es un agente de la simulación, pero usamos esta palabra ya que refleja de forma mas precisa el concepto, ya que los agentes o simulables van a simular elementos como por ejemplo un cliente, o un restaurante. Estas personas o empresas serán simulables que simularán ser esos elementos durante el transcurso de la simulación. Tomaran decisiones y realizaran acciones acordes al elemento que son, por ejemplo, el cliente ira a comer de vez en cuando en función de su poder adquisitivo a un restaurante que pueda permitirse.

## **7.2 Simulación**

### **7.2.1 Ciclo de Vida de la simulación**

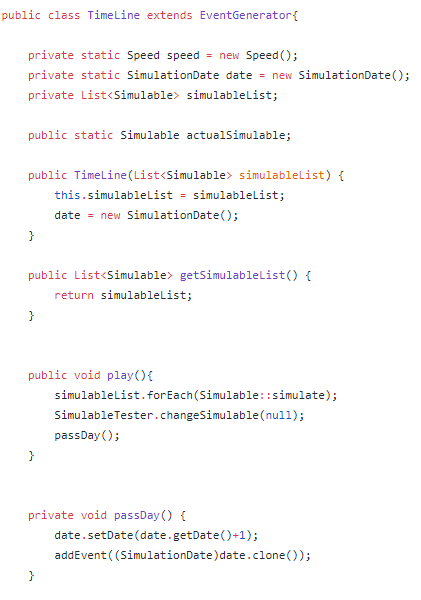
Para explicar esta simulación, es indispensable explicar el proceso principal de la simulación, el ciclo de vida. Este proceso lo lleva a cabo los administradores de la simulación, un conjunto de clases que se explicaran en el punto 7.2.4.

El ciclo es esencialmente un bucle constante que mantiene una fecha que se incrementa un día en cada iteración. Esto tiene como objetivo crear una línea de tiempo en que el ciclo o bucle es el “corazón que bombea” con los días que pasan de forma constante en cada iteración de este.

Dentro de cada simulación se produce cuatro procesos principales, en este orden:

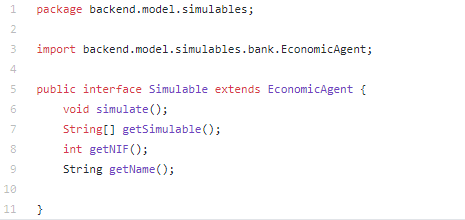
* Se llama a cada uno de los agentes para que realicen la acción que les toque (esto se explicará más adelante con detenimiento).
* Se pasa un día en la simulación.
* Los administradores de la simulación realizan ciertas tareas de mantenimiento y control (7.2.4)
* Se producirá una espera momentánea para que los días tengan una cierta duración de no más de 5 segundos.

En el primero, la línea de tiempo llama a cada uno de los agentes activos de la simulación. Estos agentes o simulables se encuentran guardados en una lista dentro de la línea de tiempo que serán siempre actualizados inmediatamente cada vez que se produzca el alta o la baja de algún simulable. A continuación, se puede ver la clase de la Línea de Tiempo, en la se puede visualizar la lista de los simulables, el método play() que llama a los simulables para que simulen, y el método passDay() que pasa el día.



Clase TimeLine (Línea de Tiempo)

La línea de tiempo no sabe que simulable es realmente, ya que mediante el principio de sustitución de Liskov, la línea de tiempo no conoce las implementaciones que se encuentran detrás y para esta, son simples “Simulables”, es decir una interfaz con el método simulate(), como podemos ver en la foto que se encuentra a continuación. El resto de los métodos son de identificación del simulable que tendrán siempre todos los ellos.



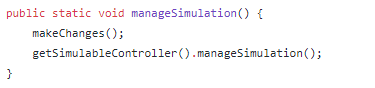
Interfaz Simulable

Los simulables serán llamados uno a uno por la línea de tiempo y realizarán la acción oportuna durante ese día. Estos simulables serán llamados por el método simulate() y en función de las circunstancias realizaran unas acciones u otras. Todos los tipos de simulables que hay y sus acciones serán explicadas durante el apartado 7.2.2.

Cuando terminen todos los elementos se terminará las acciones durante este día y se dará por finalizado este proceso.

Tras la fase de acción de los simulables, se llama al método passDay() que nombramos anteriormente, esta se encarga de actualizar la fecha que tiene la línea de tiempo, además de generar el evento de que ha pasado un día. Este tema de los eventos se explicará posteriormente en el apartado 7.2.5.

A continuación, nos encontramos con la fase de administración. En esta parte, se encarga de administrar la propia simulación y los simulables activos de esta. Debajo se encuentra el método principal de la fase de administración



Método Principal de administración

Este método como se puede observar se divide en dos partes, la primera se encarga de confirmar los cambios solicitados por los simulables y las bajas de simulables, el segundo se encarga de las posible altas en la lista de simulables.

En cuanto a la administración de cambios, esta fase concretamente se produce por el problema que trae consigo ciertos cambios en medio de la fase de acción de los simulables, por ejemplos las bajas de los propios simulables mientras la misma lista esta siendo iterada por la línea de tiempo traerá problemas de concurrencia en la ejecución de la simulación. Estos cambios de bajas de simulables se explicarán durante el apartado 7.2.3 y el resto de los cambios se explicarán durante la explicación de los simulables que solicitan esos cambios en el apartado 7.2.2.

En la segunda parte de la fase de administración, nos encontramos la sub-fase que controla las altas de simulables. Se encarga de comprobar si se quiere generar alguno de los simulables y si es afirmativo, se llamará a el módulo concreto de creación e inicialización de simulables y actualiza la lista de simulables de la línea de tiempo. Estas altas de simulables se explicarán con mas detenimiento durante el apartado 7.2.3.

### **7.2.2 Agentes Activos o Simulables**

### **7.2.3 Bajas y Altas de Simulables**

### **7.2.4 Administradores**

### **7.2.5 Eventos**

## **7.3 Entorno de usuario de la Simulación**

# **8. Conclusiones**

# **9. Bibliografía**

1. La facturación

<https://www.mheducation.es/bcv/guide/capitulo/8448614194.pdf>

2. CFDI

<https://es.wikipedia.org/wiki/CFDI>

3. Ejemplos CFDI

<https://facturador.zendesk.com/hc/es/articles/115012733308-Ejemplo-de-CFDI-Emisi%C3%B3n-3-3>

4. Tutorial Básico Ajax con JSP

<https://jarroba.com/ajax-con-jsp-y-servelts/>

5. Librería Jsoup

<https://jsoup.org/>

6. Generador de datos Aleatorios

<https://www.onlinedatagenerator.com/>

7. Fintech

<https://es.wikipedia.org/wiki/Tecnolog%C3%ADa_financiera>

<https://www.condusef.gob.mx/Revista/index.php/usuario-inteligente/educacion-financiera/763-que-son-las-fintech>

8. Jquery

<https://jquery.com/>

9. Tripadvisor

<https://www.tripadvisor.es/Restaurants-g187471-Gran_Canaria_Canary_Islands.html>

10. DB Browser for SQLite

<https://sqlitebrowser.org/>