



# Reporte Final de Estadía

José María Venegas Morgado

CPLASU



# Universidad Tecnológica del Centro de Veracruz

Programa Educativo  
Tecnologías de la Información y Comunicación / Tecnologías de la  
Información

Reporte para obtener título de  
Ingeniero en Tecnologías de la Información

Proyecto de estadía realizado en la empresa  
Solser Information Technology

Nombre del proyecto  
CPLASU

Presenta  
José María Venegas Morgado

Cuitláhuac, Ver., a 1 de abril de 2018.



## Universidad Tecnológica del Centro de Veracruz

Programa Educativo

Tecnologías de la Información y Comunicación / Tecnologías de la  
Información

Nombre del Asesor Industrial

Abraham Guerrero Sierra

Nombre del Asesor Académico

Lorena Alcudia Chagala

Jefe de Carrera

César Aldaraca Juárez

Nombre del Alumno

José María Venegas Morgado

## Contenido

|  |           |
|--|-----------|
| <b>AGRADECIMIENTOS .....</b>                         | <b>1</b>  |
| <b>RESUMEN .....</b>                                 | <b>2</b>  |
| <b>CAPÍTULO 1. INTRODUCCIÓN .....</b>                | <b>4</b>  |
| <b>1.1 Estado del Arte.....</b>                      | <b>4</b>  |
| <b>1.2 Planteamiento del Problema.....</b>           | <b>6</b>  |
| <b>1.3 Objetivos .....</b>                           | <b>6</b>  |
| <b>1.4 Definición de variables .....</b>             | <b>7</b>  |
| <b>1.5 Hipótesis.....</b>                            | <b>7</b>  |
| <b>1.6 Justificación del Proyecto .....</b>          | <b>7</b>  |
| <b>1.7 Limitaciones y Alcances.....</b>              | <b>7</b>  |
| <b>1.8 La Empresa Solser Sistem Information.....</b> | <b>8</b>  |
| <b>Historia De La Empresa .....</b>                  | <b>8</b>  |
| <b>Misión, Visión y Valores .....</b>                | <b>8</b>  |
| <b>Procesos.....</b>                                 | <b>8</b>  |
| <b>Mercado De Impacto.....</b>                       | <b>9</b>  |
| <b>Impacto En El Área De Las TIC .....</b>           | <b>9</b>  |
| <b>[IMG03]Diagrama de proceso SCRUM Solser .....</b> | <b>10</b> |
| <b>CAPÍTULO 3. DESARROLLO DEL PROYECTO .....</b>     | <b>11</b> |
| <b>CAPÍTULO 4. RESULTADOS Y CONCLUSIONES.....</b>    | <b>17</b> |
| <b>4.1 Resultados.....</b>                           | <b>17</b> |
| <b>4.2 Trabajos Futuros .....</b>                    | <b>17</b> |
| <b>4.3 Recomendaciones .....</b>                     | <b>17</b> |



ANEXOS ..... 18

Bibliografía ..... 18

## **AGRADECIMIENTOS**

Agradesco primeramente a la Universidad Tecnológica del centro de Veracruz por haber fungido su función como casa de estudios, en la cual obtuve conocimiento, actitudes y aptitudes para desarrollarme como profesionista en el ámbito de las Tecnologías de la Información.

También agradezco a la Organización Solser Technology Information por brindar la oportunidad de realizar el desarrollo de mi estadía en sus instalaciones, por ceder el apoyo de recursos materiales y de asesores que expandieron mi campo de conocimiento en el ámbito de la Tecnologías de la Información.

Dirijo también mis agradecimientos a mi asesor académico Mcc. Lorena Alcudia Chagala y a mi asesor Industrial Lic. Jesús Carlos Cornejo, por estar pendiente de mi desempeño durante el desarrollo de mi estadía y su apoyo brindado en el transcurso

Agradezco también a todos lo que fueron mis amigos y compañeros a lo largo de todos los niveles de la carrera, al compañerismo, amistad y apoyo que brindaron el cual apporto bien estar ético y emocional en mi trayecto como estúdiante en Ingeniería.

Final mente agradezco a mi familia por ser uno de los principales motores de apoyo para poder alcanzar mi meta como Ingeniero en Tecnologías de la Información, agradezco a mis padres José Venegas Valdez y María Esther Morgado Gómez por brindarme una formación responsablemente moral, ética y sólida para poder desenvolverme en la sociedad correctamente, agradezco a mis hermanos Ángel Estuardo Venegas Morgado y Leonardo Venegas Morgado por su apoyo y darme el sentido de la responsabilidad y empatía hacia los demás.

## **RESUMEN**

El proyecto CPLASU consiste en una migración de un sistema de ventanilla utilizado por una empresa bancaria. Este sistema se encarga de realizar diversas transacciones operadas por el personal de la empresa bancaria.

La empresa bancaria (el cliente) ha cedido el código fuente de su sistema de ventanilla actual, acompañado con una serie de documentos de análisis de cada una de las transacciones con las que cuenta el sistema, esto con la finalidad de realizar una migración con las mismas funciones y procesos del sistema anterior con optimizaciones de en los procesos mediante el código. Cabe mencionar que dicho software está desarrollado en el lenguaje de programación Visual Basic 6.0.

El proyecto en la actualidad cuenta con el cumplimiento de varios objetivos principales, tales como desarrollo de Ventana Principal, Desarrollo e implementación de módulos de Caja (mixta y principal), Captación, Crédito, Servicios al cliente, Transferencias, Transferencias electrónicas y la elaboración de diversas transacciones y componentes genéricos.

## Tabla de ilustraciones

|   |    |
|---|----|
| Ilustración 1 Diagrama de inicio de proyecto Solser.....  | 9  |
| Ilustración 2 Diagrama de proceso SCRUM Solser.....   | 9  |
| Ilustración 3 Diagrama de proceso SCRUM Solser.....   | 10 |
| Ilustración 4 Herramienta de seguimiento de actividades MeisterTask.....  | 11 |
| Ilustración 5 Plan de proyecto de la OLA4 .....   | 11 |
| Ilustración 6 Diagrama de actividades correspondiente a las transacciones elaboradas (Esta imagen se muestra incompleta por motivos de confidencialidad)..... | 12 |
| Ilustración 7 Diagrama de actividades correspondiente a las transacciones elaboradas (Esta imagen se muestra incompleta por motivos de confidencialidad)..... | 13 |
| Ilustración 8 Estimación del tiempo basado en una lista de tareas.....  | 13 |
| Ilustración 9 Cronograma de actividades.....  | 14 |
| Ilustración 10 Prototipo de la interfaz gráfica de la transacción. ....   | 14 |
| Ilustración 11 Ilustración 9 Prototipo de la interfaz gráfica de la transacción.....  | 15 |
| Ilustración 12 Apartado de Incidencias en la herramienta MeisterTask. ....  | 16 |

# CAPÍTULO 1. INTRODUCCIÓN

A continuación, se describe el desarrollo del proyecto que tiene como nombre “CPLASU” el cual consiste en una migración de un sistema de ventanilla utilizado por una empresa bancaria, Este documento presenta una serie de proyectos similares existentes en el estado del arte, así como los objetivos, la problemática, la justificación y el desarrollo del proyecto se encuentra referenciado en una metodología, llamada SCRUM.

## 1.1 Estado del Arte

La migración de aplicaciones consiste en el proceso de mover uno o varios programas o aplicaciones de un entorno a otro. Estas migraciones se producen por diversas razones algunos ejemplos a mencionar son la actualización de hardware, el surgimiento de nuevas, optimas y potentes tecnologías, dificultad para brindar mantenimiento o simplemente la tecnología sobre la que están desarrolladas se están volviendo.

### **Visión general de un proyecto de migración**

Existen tres fases principales para el proyecto de migración: fase de planificación, fase de práctica y fase de migración de la producción. Cada una de estas fases requiere sus propias técnicas y recursos. Es muy importante asignar suficiente tiempo para cada actividad a fin de garantizar que no queda ninguna tarea clave sin realizar.

### **Fase de planificación**

En la fase de planificación, debe realizar un inventario de los activos actuales, como documentación, máquinas, software, instancias, bases de datos y código personalizado. Es muy importante saber qué personalizaciones se han llevado a cabo y dónde se integra el sistema con los sistemas externos

### **Fase práctica**

La fase práctica se divide en dos o más partes, en función de la empresa.

- Migración del entorno de desarrollo:
  - Intente migrar un entorno de desarrollo al principio del proyecto, a fin de poder identificar los cambios necesarios en el código personalizado. Commerce.
  - Es importante que migre satisfactoriamente un entorno de desarrollo al principio del proceso si tiene desarrolladores a la espera de escribir código en la nueva versión.
- Migración del entorno de prueba
  - Realice una migración de extremo a extremo para saber qué debe esperar durante la migración de producción.

- Este proceso reduce la incertidumbre y el tiempo durante el que el sistema no está disponible en el proceso de migración.
- Temporizar cada actividad de migración en el entorno de prueba le proporcionará información importante que puede utilizar para reajustar la planificación del proyecto para la fase de migración de producción.
- Un servidor de transición es un activo muy valioso en esta fase para poder probar la reconstrucción del servidor de transición a partir del servidor de prueba.

### **Fase de migración de producción**

En la fase de migración de producción, el sistema se migra y se conmuta para sustituir la instalación anterior. Esta actividad suele realizarse durante una fase planificada de producción en la que el sistema está fuera de servicio. El valor de la fase de planificación y de práctica queda plasmado en este momento, y a que el personal técnico ya llevado a cabo sus actividades con anterioridad. (IBM, s.f.)

### **Antecedentes de migraciones de proyectos**

#### **Proyecto Migración Ontología ↔ Mikrokosmos.**

El objetivo del proyecto realizado fue el de construir una aplicación sencilla que permite a un usuario migrar el contenido de una base de datos ontológica, cuyo fin es el de representar un modelo de la realidad en forma de grafo dirigido cuyos nodos son conceptos, a un sistema práctico de traducción multilingüe denominado Mikrokosmos. Para desarrollar esta aplicación, el primer paso necesario es analizar la estructura de ambos extremos de la migración, por un lado, la base de datos ontológica desarrollada por otros compañeros en años anteriores y, por otro, los ficheros en los que Mikrokosmos almacena la ontología. A partir de ahí, se pueden establecer los pasos que debe seguir la migración de datos y codificarlos. Por otro lado, la aplicación debe incluir una utilidad para verificar la base de datos ontológica de cara a saber si se ajusta a las restricciones que ésta ha de cumplir.

(Rodríguez)

#### **Proyecto de migración a desarrollar**

OFI es un sistema bancario de ventanilla el cual, en cargo de gestionar operaciones bancarias, está conformado por los módulos de Caja y captación. Los módulos a migrar son Eliminación de sobrantes de caja y Deposito Interestado en efectivo. El lenguaje en el que se encuentra originalmente desarrollado es Visual Basic ya que en los años de los 90 Visual Basic es sin duda un lenguaje verdaderamente popular. Su última versión Visual Basic 6.0 era utilizada por miles de personas para el desarrollo de aplicaciones de escritorio ya que era una manera sencilla y visual con una curva de aprendizaje suave esto hacia posible que cualquiera hiciera cosas útiles en poco tiempo.

Para la mala fortuna del lenguaje de programación los años han pasado y Microsoft dejó de darle la importancia debida para que se mantuviese al margen de los lenguajes competentes. A lo largo de estos años los paradigmas de programación han cambiado de acuerdo a los que había hace algunos años lo que ha hecho que este lenguaje ya no sea una opción viable para las empresas. En la actualidad hay diversas tecnologías que cuentan con un mayor número de ventajas que con desventajas, entre los lenguajes de programación una de las más populares es Java ya que es multiplataforma, multifacético y cuenta con características que lo hacen un lenguaje viable para las migraciones tales como arquitectura y frameworks por mencionar el utilizado en este proyecto que es Spring que se encuentra basado en el manejo de inyección de dependencias simplificando así el desarrollo de la aplicación lo que lo convierte en una buena opción.

La empresa bancaria a la cual se le desarrolla el proyecto de software cuenta con su antiguo Sistema de Ventanilla desarrollado en Visual Basic 6.0 con el paradigma orientado a eventos. Este software es ejecutado como aplicación de escritorio con una comunicación en un servidor local en la sucursal y posteriormente con una comunicación a un servidor central conectado a la red.

(Yang)

(Fielding, 2000)

## 1.2 Planteamiento del Problema

El lenguaje en el que se encuentra desarrollado el actual sistema de la empresa bancaria es en Visual Basic 6.0 el cual ya es considerado un lenguaje obsoleto, además que trae consigo desventajas y una de ellas es que no es un lenguaje que soporte del todo la programación orientada a objetos o que no permite la creación de aplicaciones multihilado, esto trae problemas de rendimiento en el software. Además, como se mencionó anteriormente es un lenguaje obsoleto por lo cual dificulta su mantenibilidad y al correr del tiempo esto será un problema mayor. Ante esta problemática se optó por realizar una migración del sistema hacia el lenguaje Java utilizando la tecnología JavaFX y el framework Spring.

## 1.3 Objetivos

### **Objetivo general:**

Desarrollar los módulos de Eliminación de Sobrantes de Caja y Depósito Interestado en efectivo del sistema bancario Bancoppel en la empresa SOLSER INFORMATION TECHNOLOGY.

### **Objetivos específicos**

- Realizar un análisis detallado de los módulos
- Realizar la migración de Visual Basic 6.0 a Java.
- Respetar los procesos de funcionalidad y el diseño de interfaces.
- Considerar la arquitectura orientada a Web.
- Optimizar los procesos del sistema de los módulos.

## 1.4 Definición de variables

Independiente:

Los módulos a desarrollar están realizados en Visual Studio 6.0. Debido a que este lenguaje es obsoleto frente a las nuevas tecnologías se optó por realizar una migración de estos módulos en el lenguaje java.

Dependiente:

Desarrollar los módulos del sistema de ventanilla en el lenguaje Java utilizando la tecnología JavaFX y dándole un enfoque de aplicación web para su sencilla distribución en las sucursales de la empresa bancaria.

Constata:

Al realizar los módulos se deberá respetar el antiguo proceso de operación a nivel usuario esto con la finalidad de no perder recursos económicos, de personal y tiempo en la capacitación del uso del nuevo sistema.

## 1.5 Hipótesis

Con el uso de las tecnologías Java, Scene Builder, Spring. se pretende llevar a cabo la migración exitosa de los módulos Eliminación de Sobrantes de Caja y Deposito Interestado en Efectivo que se encuentran desarrollados en el lenguaje Visual Basic 6.0 mejorando así el desempeño, la eficiencia estética de las interfaces de usuario.

## 1.6 Justificación del Proyecto

Se debe realizar la migración de los de los módulos Eliminación de Sobrantes de Caja y Deposito Interestado en Efectivo de Visual Basic 6.0 a Java. La migración de los módulos aporta grandes ventajas tales como la abierta posibilidad de hacer mejoras en algún futuro y la facilidad de dar mantenimiento de los mismos. Por el lado del usuario ofrece una mejora en la rapidez de los procesos de las tareas y una interfaz de usuario más simple, cómoda e innovadora. Al ser una aplicación de java con orientación web facilita la distribución en las diversas sucursales y actualizaciones de forma óptima.

## 1.7 Limitaciones y Alcances

Limitaciones:

- No se pueden aportar mejoras al diseño gráfico diferentes a las actuales
- El sistema requiere la instalación de una plataforma Java.
- Se requieren versiones específicas de navegadores para el correcto funcionamiento del sistema
- El sistema es de uso exclusivo para la empresa bancaria.

Alcances:

- Mejoras en la estética de las interfaces gráficas, sin afectar la secuencia de los pasos para operar la transacción.
- Mejoras en los procesos internos a nivel código para la optimización en la ejecución de tareas.
- Permitir una fácil mantenibilidad del sistema.
- Dejar abierta la posibilidad de futuras mejoras.
- Mejoras en los módulos Caja (principal y mixta) y Captación.

## 1.8 La Empresa Solser Sistem Information

### Historia De La Empresa

Solser Information Technology nace en el año 2002, con la visión de convertirnos en una empresa exportadora de soluciones en Tecnología de Información hecha en México hacia el mundo. Nuestro Centro de Soluciones actualmente se ubica en la ciudad de Querétaro con una producción al año de más de 100,000 horas y con capacidad para 180,000 y contamos con oficinas comerciales en la Ciudad de México y Cancún Quintana Roo.

### Misión, Visión y Valores

**Misión:** Ser de las primeras empresas en brindar soluciones que ayuden a desarrollar e incrementar los negocios de nuestros clientes, a través del potencial de nuestros profesionales, la calidad y la innovación tecnológica desde México para el mundo.

**Visión:** En SOLSER buscamos consolidarnos como líderes en el mercado de Soluciones Tecnológicas en México y otros países, reconocidos por nuestra experiencia y compromiso en brindar soluciones tecnológicas de alto valor para hacer crecer los negocios de nuestros clientes, basados en las mejores prácticas de calidad.

### Valores:

- **Responsabilidad y Compromiso:** Cumplir con los objetivos de nuestros clientes es nuestra meta para alcanzar el éxito.
- **Dinamismo e integridad:** Creemos en la capacidad de lograr los más grandes retos, basados en el esfuerzo, la disciplina y la honestidad.
- **Rectitud:** Nos comprometemos a hacer siempre lo mejor que podemos basados en el principio de que cumplir con los clientes es lo más importante.
- **Respeto:** Tratamos con respeto a las personas dentro y fuera de la empresa, porque sabemos que es la base de la confianza en todo lo que hacemos.
- **Actitud positiva y trabajo en equipo:** En SOLSER creemos que tenemos la capacidad de pensar, hacer y lograr que todo sea posible a través del trabajo en equipo.

### Procesos

Al contar con la certificación en el modelo CMMI nivel 3 todos los procesos de la empresa se encuentran en una constante mejora, la cual requiere el estricto seguimiento de cada una de las actividades realizadas, existen diversos procesos dentro de la empresa cada

uno acorde a las necesidades de cada área, pero en este caso nos enfocaremos en el departamento de desarrollo ya que es considerado el que más flujo de trabajo genera.

Durante la concepción del proyecto se encuentran 3 procesos principales compuestos por diversas actividades estos procesos son.

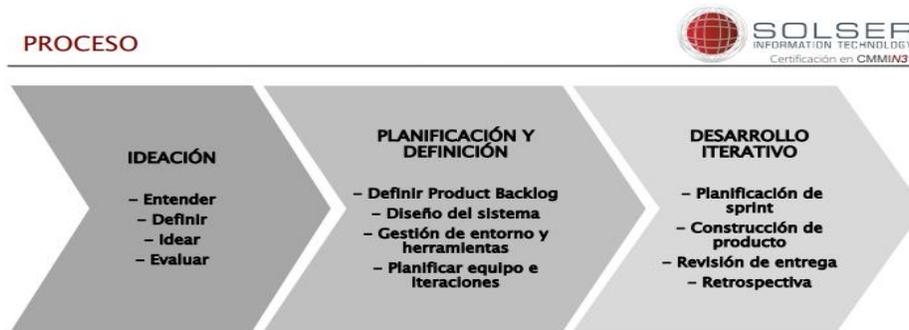


Ilustración 1 Diagrama de inicio de proyecto Solser.

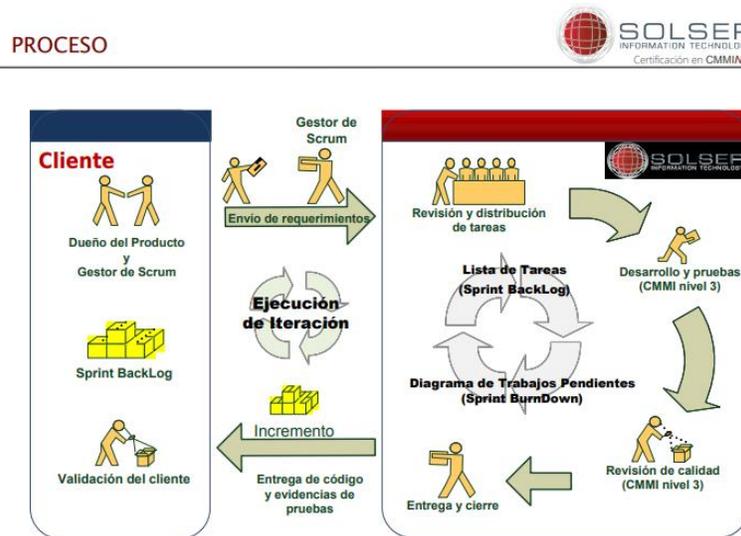


Ilustración 2 Diagrama de proceso SCRUM Solser

### Mercado De Impacto

Para la industria del software existen diversos mercados que pueden ser explotados, como el de entretenimiento, transporte, turismo, telecomunicaciones, Pymes y el sector financiero siendo este último el sector de mayor aprovechamiento para la empresa Solser, ya que cuenta con más de diez productos orientados a este ramo los cuales suelen ser los más lucrativos y de mayor reto.

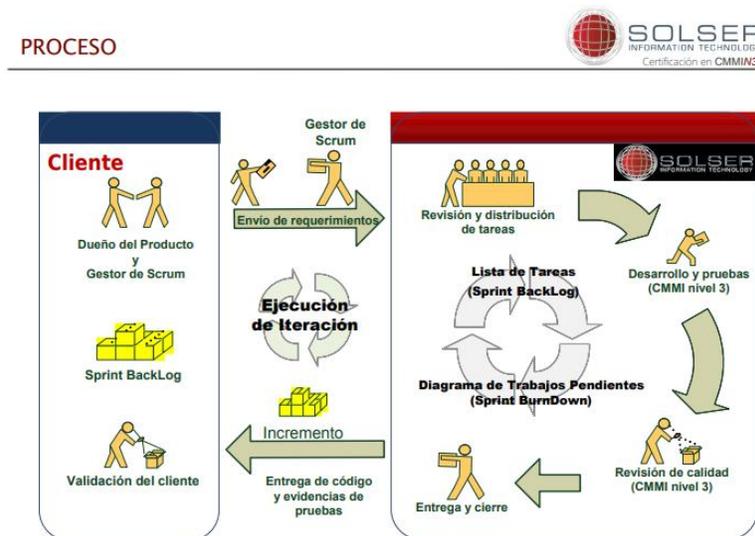
### Impacto En El Área De Las TIC

Dentro de la empresa bancaria el sistema de ventanilla es uno de los que más carga de trabajo tiene día con día, por lo cual es de vital importancia que este tenga un correcto funcionamiento y un nivel de eficiencia óptimo, es por eso que con la migración tecnológica serán ofrecidos múltiples beneficios para poder cumplir con las condiciones mencionadas

con anterioridad, otro añadido importante que viene con el cambio de lenguaje es el soporte del mismo, esto quiere decir que con el paso del tiempo el lenguaje contara con diversas APIS que ayudaran al sistema a realizar diversas acciones de maneras más simples y eficientes.

## CAPÍTULO 2. METODOLOGÍA

La metodología utilizada en la empresa Solser Information Technology para el desarrollo de software es SCRUM el cual contiene un proceso completo que aplica de manera constante buenas prácticas en el trabajo en equipo con la finalidad de obtener resultados óptimos en el desarrollo del proyecto de manera rápida y efectiva.



*Ilustración 3 Diagrama de proceso SCRUM Solser*

[IMG03]Diagrama de proceso SCRUM Solser

En el punto actual del desarrollo del proyecto ya se cuenta con las tareas a realizar, cada que se asigna una tarea a algún colaborador este debe llevar un seguimiento de su tarea a través de dos medios principalmente, estos son la bitácora y el MeisterTask, la primera consiste en un archivo donde se establece en que aplicativo se trabaja, a que fase del proyecto pertenece, en que consiste la actividad, el tipo, las fechas invertidas, así como sus horas he interrupciones.

La herramienta que es de suma importancia para el completo seguimiento del proyecto es el MeisterTask, dicha herramienta se encarga de supervisar el estado de los entregables del sprint por medio de ítems, así como de etiquetas de avance ya que cada una de las actividades tiene un tiempo

de entrega establecido que debe cumplirse en tiempo y forma, de esta manera la herramienta permite detectar que entregables llevan atraso, cuales ya fueron entregados y cuales ya han sido aprobados en calidad.

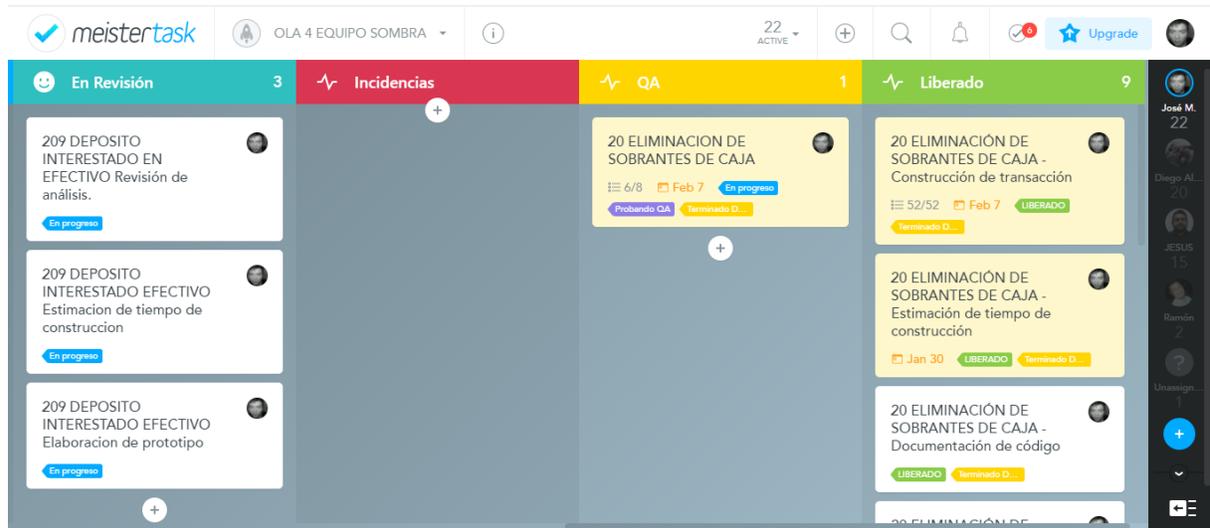


Ilustración 4 Herramienta de seguimiento de actividades MeisterTask.

Por medio de las herramientas anteriormente mencionadas es como el equipo de desarrollo detecta todos los riesgos, que se pueden presentar en forma de atrasos dentro de los sprints de SCRUM.

### CAPÍTULO 3. DESARROLLO DEL PROYECTO

Para llevar un correcto control de las actividades por realizar se realiza un plan de proyecto de las actividades asignadas por OLA en este caso es el plan de proyecto de la OLA4.

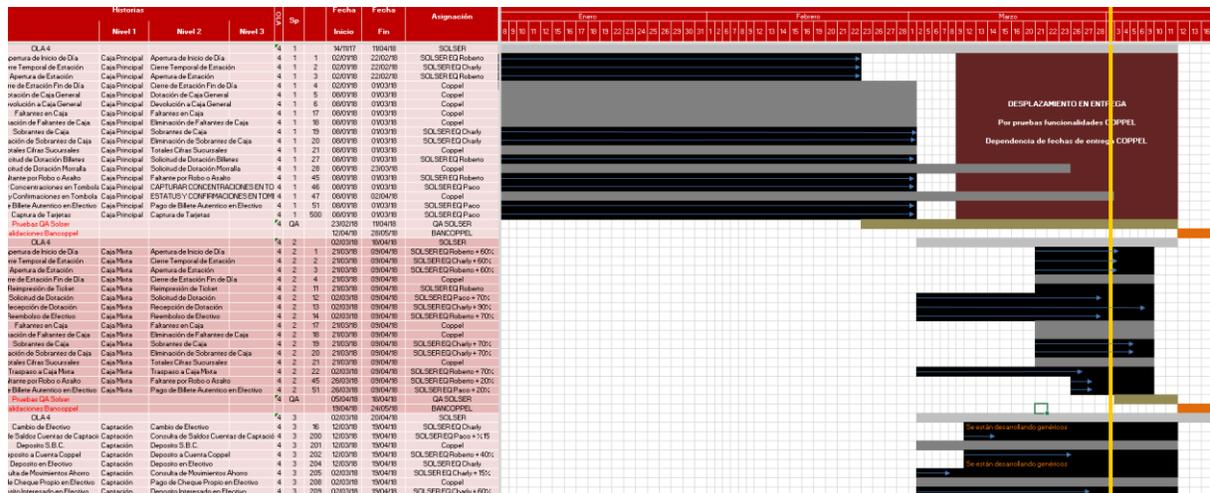


Ilustración 5 Plan de proyecto de la OLA4

Mediante el proceso de desarrollo del sistema se llevan a cabo la realización de múltiples tareas divididas en actividades menores con la finalidad de realizar una estimación del tiempo y medir el progreso de acuerdo a las tareas realizadas para así alcanzar un objetivo principal del Sprint.

El primer proceso durante el desarrollo de una transacción es la revisión del análisis del antiguo sistema dicho análisis es entregado por el personal de desarrollo de la empresa bancaria, para completar un análisis a fondo se debe realizar una depuración con el sistema de Visual Basic esto es debido a que el análisis entregado no se encuentra detallado del todo. Como resultado del análisis se obtiene un documento el cual es un reporte de las funcionalidades de la transacción.

Una vez finalizado el análisis y obtenida información sobre los procesos que realiza la transacción se realiza una estimación acerca del tiempo que llevara el desarrollo de la documentación, prototipo, codificación, pruebas y corrección de incidencias esto con la finalidad de llevar un control detallado del tiempo de trabajo.

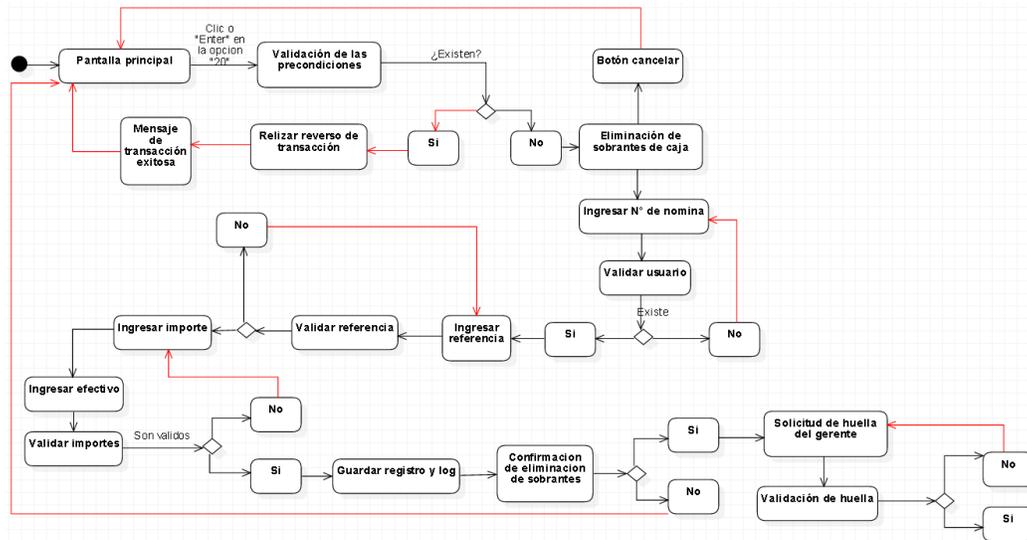


Ilustración 6 Diagrama de actividades correspondiente a las transacciones elaboradas (Esta imagen se muestra incompleta por motivos de confidencialidad)

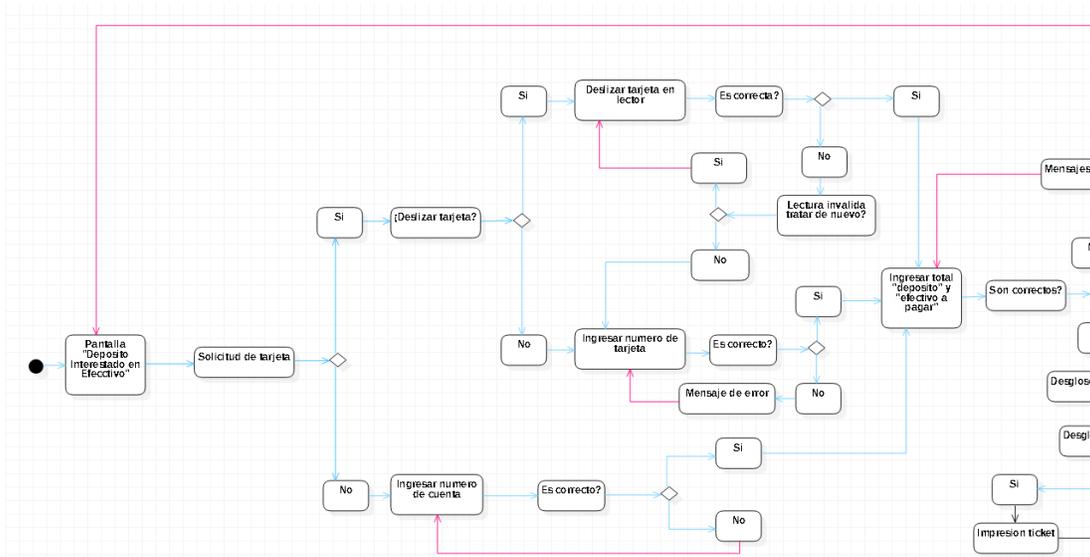


Ilustración 7 Diagrama de actividades correspondiente a las transacciones elaboradas (Esta imagen se muestra incompleta por motivos de confidencialidad)

Terminado el proceso de la estimación del tiempo se lleva a cabo el modelado de diagramas de caso de uso y actividades posteriormente se realiza el caso de uso de la transacción para así continuar con el desarrollo del prototipo de la interfaz gráfica el cual es realizado con la herramienta Scene Builder.

| Tarea   | Horas (Si aplica) | Justificación  |
|---|-------------------|--|
| Elaboración de caso de uso  | 3                 | No hay existencia del documento.   |
| Elaboración de prototipo  | 2                 | No hay existencia del prototipo.   |
| Validación de datos de pantalla   | 8.00              | La transacción contiene 4 controles que necesitan validación de longitud de campo, que no tengan valores nulos, validación de campos especiales, su tipo de carácter, en el caso de los primeros dos campos se debe validar su autenticidad y crear navegabilidad. |
| Validación de reglas de negocio   | 5.00              | Se debe verificar el No. de tarjeta y que el No. de cuenta sea existente.  |
| Consultas a base de datos DB2 (itran, itrandet, vdesgmon, varqueo, etc)                     |                   |  |
| Funciones DB2 a ejecutar  |                   |  |
| Afectación a tablas (incluir tablas varqueo, vdiarios, vjournal, vdesgmon y las necesarias) | 6.00              | Se estima este tiempo debido al análisis y comprensión de las tablas y la realización de los procesos de afectación.   |
| Integración de Web Services   | 20.00             | De acuerdo al número de procedimientos almacenados y la dedicación de tiempo al análisis de su función, realización y aplicación se estima este tiempo.  |
| Confirmación de fin de transacción  | 3.00              | Estimación a consideración del tiempo que conlleva la verificación del funcionamiento correcto de la transacción.  |
| Elementos (controladores, servicios, clases helpers, métodos, etc.)                         | 13.00             | Estimación realizada a consideración del tiempo que conllevará la realización de los productos.  |
| Documentación   | 3.00              | Se estima este tiempo de acuerdo al número de tareas a documentar.   |
| Pruebas unitarias   | 3.00              | Se estima de acuerdo al número de procesos a verificar y reportar su fallas.   |
| <b>Total de horas</b>   | <b>66.00</b>      |  |
| <b>DEPENDENCIAS</b>   |                   |  |
| DESGLOSE DE MONEDA NACIONAL   |                   |  |
| CAPTURAR TARJETA  |                   |  |

Ilustración 8 Estimación del tiempo basado en una lista de tareas.

| CRONOGRAMA DE ACTIVIDADES                                 | FASE           | TIEMPO |  |  |  |  |  |  |  |
|---|----------------|--------|--|--|--|--|--|--|--|
| ELABORACION DE PROTOTIPO                                  | FXML           | 3.00   |  |  |  |  |  |  |  |
| REVISAR ESTRUCTURA DE PANTALLA                            | FXML           | 0.50   |  |  |  |  |  |  |  |
| REVISAR ESTRUCTURA DE CONTROLLER                          | CONTROLLER     | 0.50   |  |  |  |  |  |  |  |
| CONSTRUIR VALIDACIONES Y PARAMETROS DE PANTALLA           | FXML           | 0.50   |  |  |  |  |  |  |  |
| CONSTRUIR VALIDACIONES DE PANTALLA EN CONTROLLER          | CONTROLLER     | 0.50   |  |  |  |  |  |  |  |
| CONSTRUIR NAVEGACION DE PANTALLA                          | CONTROLLER     | 1.00   |  |  |  |  |  |  |  |
| PONER FOCO EN CAMPO INICIAL                               | CONTROLLER     | 0.50   |  |  |  |  |  |  |  |
| CONSTRUIR VALIDACION DE LECTURA DE TARJETA                | CONTROLLER     | 1.00   |  |  |  |  |  |  |  |
| CONSTRUIR VALIDACION DEL CAMPO NO. DE TARJETA             | CONTROLLER     | 1.00   |  |  |  |  |  |  |  |
| CONSTRUIR VALIDACION DEL CAMPO NUMERO DE TARJETA          | CONTROLLER     | 0.50   |  |  |  |  |  |  |  |
| CONSTRUIR VALIDACION DEL NUMERO DE CUENTA                 | CONTROLLER     | 0.50   |  |  |  |  |  |  |  |
| CONSTRUIR LLAMADO A CONSULTA BUSQUEDA DE NUMERO DE CUENTA | SERVICE        | 0.50   |  |  |  |  |  |  |  |
| CONSTRUIR VALIDACION DEL CAMPO TOTAL DEPOSITO             |                | 0.50   |  |  |  |  |  |  |  |
| CONSTRUIR VALIDACION DEL CAMPO EFECTIVO RECIBIDO          |                | 0.50   |  |  |  |  |  |  |  |
| ESTATUS DE CAMPOS Y BOTONES                               |                | 0.50   |  |  |  |  |  |  |  |
| <b>SOLICITAR CREACION DE WEB SERVICES</b>                 | EXTERNO        | 1.00   |  |  |  |  |  |  |  |
| <b>cons_sdos2</b>   |                |        |  |  |  |  |  |  |  |
| CONSTRUIR LLAMADO A WEB SERVICE                           | CONTROLLER     | 0.5    |  |  |  |  |  |  |  |
| SOLICITAR CREACION DE WEB SERVICE                         | EXTERNO        | 0.5    |  |  |  |  |  |  |  |
| CONSTRUIR ESTRUCTURA DE LLAMDO WEB SERVICE                | SERVICE        | 0.5    |  |  |  |  |  |  |  |
| CONSTRUIR RESTCONTROLLER PARA LLAMADO WEB SERVICE         | RESTCONTROLLER | 0.5    |  |  |  |  |  |  |  |
| CONFIGURAR RESTCONTROLLER PARA LLAMADO WEB SERVICE        | RESTCONTROLLER | 0.75   |  |  |  |  |  |  |  |
| CONSTRUIR LLAMADO A SERVICE WEBSERVICE                    | SERVICE        | 0.75   |  |  |  |  |  |  |  |
| CONSTRUIR IMPLEMENTACION SERVICE WEBSERVICE               | SERVICE IMPL   | 0.75   |  |  |  |  |  |  |  |
| CONSTRUIR LLAMADO A WEB SERVICE                           | SERVICE IMPL   | 0.5    |  |  |  |  |  |  |  |
| CONSTRUIR BEAN DE REQUEST WEB SERVICE                     | SERVICE IMPL   | 0.5    |  |  |  |  |  |  |  |
| CONSTRUIR BEAN DE RESPONSE DE WEB SERVICE                 | SERVICE IMPL   | 0.5    |  |  |  |  |  |  |  |
| CONSTRUIR VALIDACIONES DE LLAMADO WEBSERVICE              | SERVICE IMPL   | 0.75   |  |  |  |  |  |  |  |
| CONSTRUIR ESTRUCTURA WEBSERVICE                           | BASEDATOS      | 0.5    |  |  |  |  |  |  |  |
| VERIFICAR EXISTENCIA WEBSERVICE                           | EXTERNO        | 0.5    |  |  |  |  |  |  |  |
| PROBAR WEBSERVICE   |                | 0.25   |  |  |  |  |  |  |  |
| CREAR LLAMADO WEBSERVICE                                  | BASEDATOS IMPL | 0.5    |  |  |  |  |  |  |  |
| CREAR MAPEO RESPONSE WEBSERVICE VS BEAN RESPONSE          | BASEDATOS IMPL | 0.5    |  |  |  |  |  |  |  |

Ilustración 9 Cronograma de actividades.

Realizado el prototipo de la interfaz es revisado para obtener el visto bueno y así proceder a realizar su controlador para poder agregar funcionalidad a la transacción, en este proceso se realizan las funciones requeridas de acuerdo al análisis.

Ilustración 10 Prototipo de la interfaz gráfica de la transacción.

*Ilustración 11 Ilustración 9 Prototipo de la interfaz gráfica de la transacción.*

Posteriormente realizada la construcción de la transacción de acuerdo a los requerimientos solicitados, se procede a la realización de las pruebas unitarias usando como guía las especificaciones en el caso de uso, este proceso también corresponde a la documentación de las pruebas con la finalidad de generar un registro de evidencias que respaldan el correcto funcionamiento del módulo.

Para la entrega de un trabajo de calidad una vez realizado el desarrollo y las pruebas unitarias de la transacción, el producto desarrollado será revisado por el departamento de QA este es el encargado de realizar revisiones de la funcionalidad del módulo en un ambiente tal y como el del usuario final. Una vez terminado el proceso de revisión por el departamento QA emitirá un juicio sobre el desempeño de la transacción, si fue un desempeño óptimo y sin fallas se liberará como terminado para la entrega del cliente, en caso de mostrar fallas se realizará un listado de las mismas y se agregaran a un sistema local de la empresa el cual es el encargado de gestionar las fallas cediéndoles el nombre de incidencias.

En caso de contar con incidencias en la transacción se procederá a revisar el listado de estas en el sistema COS que como se mencionó anteriormente este es el encargado de gestionar las incidencias. Las incidencias tienen 4 estados Abierta, Proceso, Resuelto y Cancelada.

Estados de las incidencias:

- Estado Abierta: Se refiere a que hay una incidencia por resolver
- Estado Proceso: Se refiere a que el desarrollador ha iniciado la corrección de la incidencia.
- Estado Resulto: Se refiere a que se ha dado solución a la incidencia y está lista para una nueva revisión.
- Estado Cancelado: Se refiere a que no existe en una incidencia como tal un error por parte del desarrollador y que la falla se debe a un factor externo al proceso de codificación.

Estas incidencias proceden a liberarse en un espacio asignado después de terminar la transacción, este espacio de tiempo se encuentra contemplado en la herramienta MeisterTaks con el nombre de

“Incidencias”.

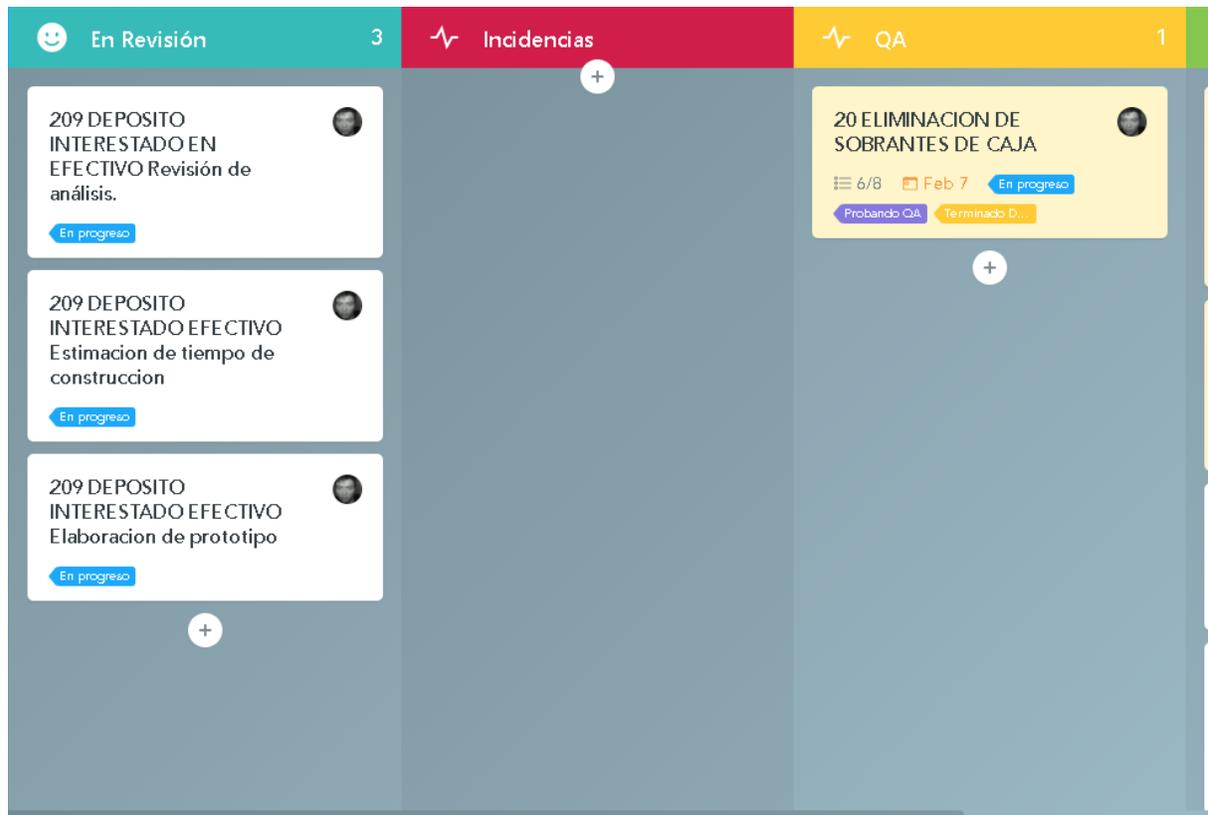


Ilustración 12 Apartado de Incidencias en la herramienta MeisterTask.

Para la resolución de las incidencias se tiene que llevar un proceso:

- Revisar el número de incidencias totales referentes a la transacción existen en el sistema COS.
- Revisar la descripción de la incidencia y revisar las evidencias de los errores para comprender la falla.
- Verificar que el funcionamiento actual coincida con el descrito en el caso de uso.
- Si la incidencia se debe a un error de funcionamiento de código se deberá corregir en la codificación, si por un mal seguimiento de la descripción del caso de uso se deberá adaptar el proceso de acuerdo al caso de uso tal como es la petición del cliente.
- Una vez resuelta la incidencia se marcará como resuelta en el sistema COS para una nueva revisión y así ser liberada como óptima.

Una vez liberadas las incidencias relacionadas a la transacción asignada se somete a una nueva revisión por el departamento de QA, este proceso se repetirá hasta que no haya alguna incidencia existente y poder así liberar el término de la transacción.

## **CAPÍTULO 4. RESULTADOS Y CONCLUSIONES**

### **4.1 Resultados**

Se logró migrar con éxito de Visual Basic 6.0 a Java los módulos Eliminación de Sobrantes de Caja y Deposito Interestado en Efectivo respetando el proceso de las operaciones y la similitud de las interfaces gráficas.

### **4.2 Trabajos Futuros**

Mantenimiento a los módulos Eliminación de Sobrantes de Caja y Deposito Interestado en Efectivo. Mejoras en el proceso de operación de los mismos y funcionamiento de interfaces.

### **4.3 Recomendaciones**

Tener un control correcto sobre los cambios realizados durante el desarrollo del proyecto, así como los cambios de arquitectura o la estructura de la organización del proyecto esto con la finalidad de generar una curva de aprendizaje suave y práctica para el futuro personal que posiblemente pueda ser contratado.

## ANEXOS

### Bibliografía

Fielding, R. T. (2000). *Architectural Styles and the Design of Network-based Software Architectures*. California: University of California, Irvine.

IBM. (s.f.). *IBM*. Obtenido de Visión general de un proyecto de migración:  
[https://www.ibm.com/support/knowledgecenter/es/SSZLC2\\_7.0.0/com.ibm.commerce.inSTALL.doc/refs/rmgovrview.htm](https://www.ibm.com/support/knowledgecenter/es/SSZLC2_7.0.0/com.ibm.commerce.inSTALL.doc/refs/rmgovrview.htm)

Rodríguez, L. A. (s.f.). *Proyecto Migración Ontología ↔ Mikrokosmos*. Madrid: Facultad de Informática Universidad Complutense de Madrid .

Yang, H. Y. (s.f.). *An Empirical Study into Use of Dependency Injection in Java*.