

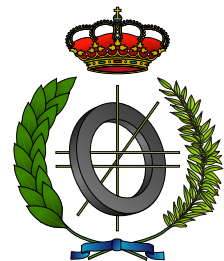


UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería en Informática



**TFG del Grado en Ingeniería Informática**

**Aplicación de soporte a personas con  
dificultades de visión**



Presentado por Bryan Reinoso Cevallos  
en Universidad de Burgos — Julio de 2014  
Tutores: José Francisco Díez Pastor,  
Dr. César I. García Osorio



**Resumen**

**Descriptores**



**Abstract**

**Keywords**



# Índice general

<b>1. Introducción</b>	<b>1</b>
<b>2. Objetivos del proyecto</b>	<b>3</b>
2.1. Objetivos funcionales . . . . .	3
<b>3. Conceptos teóricos</b>	<b>5</b>
<b>4. Técnicas y herramientas</b>	<b>7</b>
4.1. Técnicas de desarrollo . . . . .	7
4.2. Herramientas utilizadas . . . . .	7
4.2.1. Gestor de Tareas: VersionOne . . . . .	7
4.2.2. Gestor de Versiones: Git Hub . . . . .	7
4.2.3. IDEL de desarrollo: Android Studio . . . . .	8
4.2.4. Herramienta de Generación de Documentación: $\text{\LaTeX}$ . . . . .	8
4.2.5. Herramientas de Deep Learning: NeuralTalk . . . . .	8
4.2.6. Herramientas de Deep Learning: Caffe . . . . .	8
4.2.7. Herramientas de desarrollo de servidores: Flask . . . . .	8
4.2.8. Manual del Programador . . . . .	9
<b>5. Estado del arte</b>	<b>15</b>
<b>6. Aspectos relevantes del desarrollo del proyecto</b>	<b>17</b>
6.1. Resolviendo puzles en un ordenador . . . . .	17
<b>7. Conclusiones y líneas de trabajo futuras</b>	<b>19</b>
7.1. Dificultades encontradas . . . . .	19
7.1.1. Fallo con GSOAP y Apache . . . . .	19
<b>Anexos</b>	<b>20</b>





# Índice de figuras

4.1. Página de descarga del JDK de Java . . . . .	9
4.2. Instalación del JDK, paso 1. . . . .	10
4.3. Instalación del JDK, paso 2. . . . .	11
4.4. Instalación del JDK, paso 3. . . . .	12
4.5. Instalación del JDK, paso 4. . . . .	13



# Índice de tablas



# 1. INTRODUCCIÓN

---

El objetivo principal del proyecto es crear un prototipo de aplicación que se compondrá de dos partes.

La primera será el servidor, el cual recibirá una imagen y la procesará para devolver como resultado una frase que describa lo que la imagen representa.

La segunda parte será un cliente en Android que será con el que mandemos la imagen al servidor y recibiremos la respuesta. Esta será leída en voz alta gracias a un API de lectura.

La aplicación está destinada a personas con dificultades de visión para que puedan tener una idea algo concreta de lo que con la cámara han captado. Por tanto el cliente estará orientado en lo máximo posible a una interfaz de usuario que pueda ser usada por una persona con dificultades de visión.



## **2. OBJETIVOS DEL PROYECTO**

---

### **2.1 Objetivos funcionales**

El principal objetivo del proyecto es:





### 3. CONCEPTOS TEÓRICOS

---

En este apartado se profundizará...



## 4. TÉCNICAS Y HERRAMIENTAS

---

En este apartado se indicarán las técnicas y herramientas utilizadas durante la realización del proyecto.

### 4.1 Técnicas de desarrollo

En esta sección...

### 4.2 Herramientas utilizadas

En este apartado se mostrará las distintas herramientas utilizadas para el desarrollo del proyecto.

#### 4.2.1 Gestor de Tareas: VersionOne

Se ha estudiado entre varias posibles herramientas, entre ellas están:

- PivotalTracker
- FogBugz
- VersionOne

Se ha optado por la herramienta VersionOne, que ofrece unas condiciones notablemente mejores a las otras en su versión gratuita y además resulta bastante intuitiva y fácil de usar.

#### 4.2.2 Gestor de Versiones: Git Hub

Se ha estudiado entre varias posibles herramientas, entre ellas están:

- GitHub
- Bitbucket
- Sourcefoge

Finalmente se decidió que se iba a usar la herramienta GitHub porque se tenía experiencia previa en el uso de la misma, ofrece unas condiciones bastante razonables en su versión gratuita y se puede hacer un buen seguimiento del proyecto con ella.

### 4.2.3 IDEL de desarrollo: Android Studio

Se ha estudiado entre varias posibles herramientas, entre ellas están:

- Eclipse
- Android Studio

La elección de Android Studio ha sido porque no sólo es una herramienta exclusivamente dedicada a aplicaciones Android, sino que resultaba más prometedora que Eclipse; la cuál pensamos que puede quedar obsoleta para este tipo de aplicaciones.

### 4.2.4 Herramienta de Generación de Documentación: L<sup>A</sup>T<sub>E</sub>X

Se ha elegido esta herramienta debido a su facilidad de uso y a que optimiza automáticamente la estructura del producto final para ofrecer el mejor resultado visual.

### 4.2.5 Herramientas de Deep Learning: NeuralTalk

Se ha estudiado entre varias posibles herramientas, entre ellas están:

- Lib CCV
- Overfeat
- Deep Belief SDK

Esta herramienta ha sido elegida porque venía en el mismo idioma en el que se iba a programar el servidor, además de que su aplicación en nuestro proyecto era mucho mas práctico, pues devuelve una frase descriptiva de lo que en una imagen hay.

### 4.2.6 Herramientas de Deep Learning: Caffe

Esta herramienta es usada en nuestro proyecto debido a que la herramienta NeuralTalk tiene dependencia de este proyecto para extraer características de las imágenes y luego poder realizar una predicción con ellas. Además este proyecto está bien estructurado y en si se sigue correctamente su documentación es relativamente fácil de instalar.

### 4.2.7 Herramientas de desarrollo de servidores: Flask

Se ha estudiado entre varias posibles herramientas, entre ellas están:

- Axis2/C
- GSOAP
- Tomcat

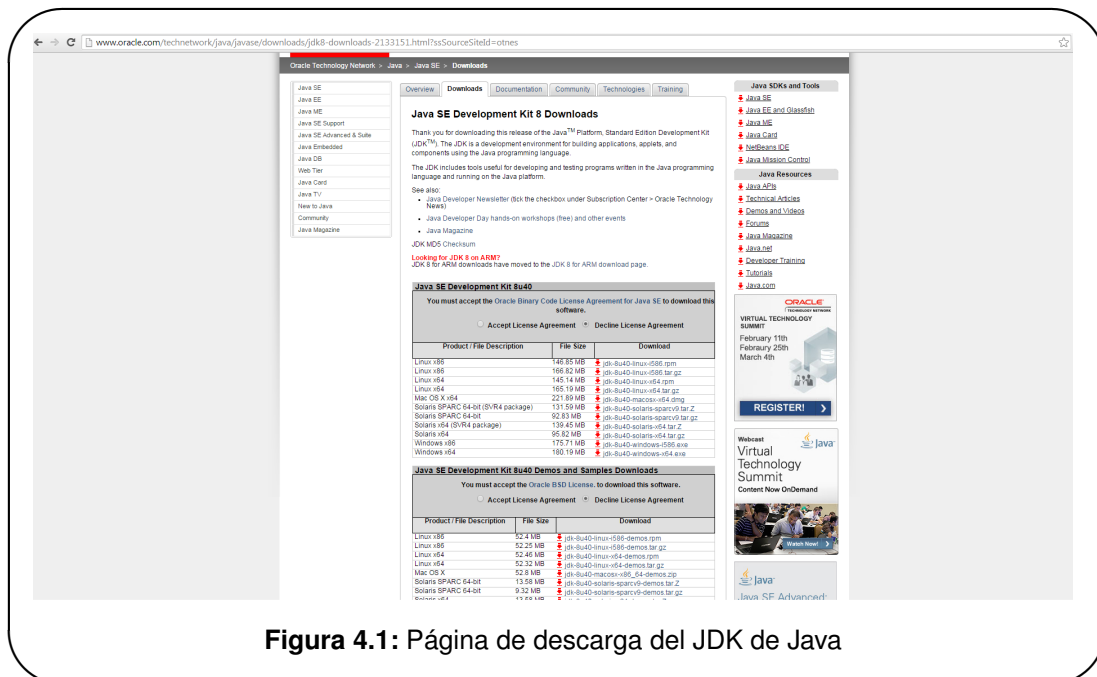


Figura 4.1: Página de descarga del JDK de Java

Se ha escogido esta herramienta en concreto porque sobre todas las demás su funcionamiento era muy inmediato y además se escribe en Python, que es un idioma muy versátil y fácil de usar. El hecho de que esta herramienta tenga un funcionamiento y programación tan sencilla la hace una herramienta que, a nuestro parecer, destaca sobre el resto y es interesante trabajar con ella.

#### 4.2.8 Manual del Programador

En esta sección se procederá a la explicación detallada de cómo instalar las herramientas necesarias y qué herramientas son necesarias para trabajar sobre este proyecto.

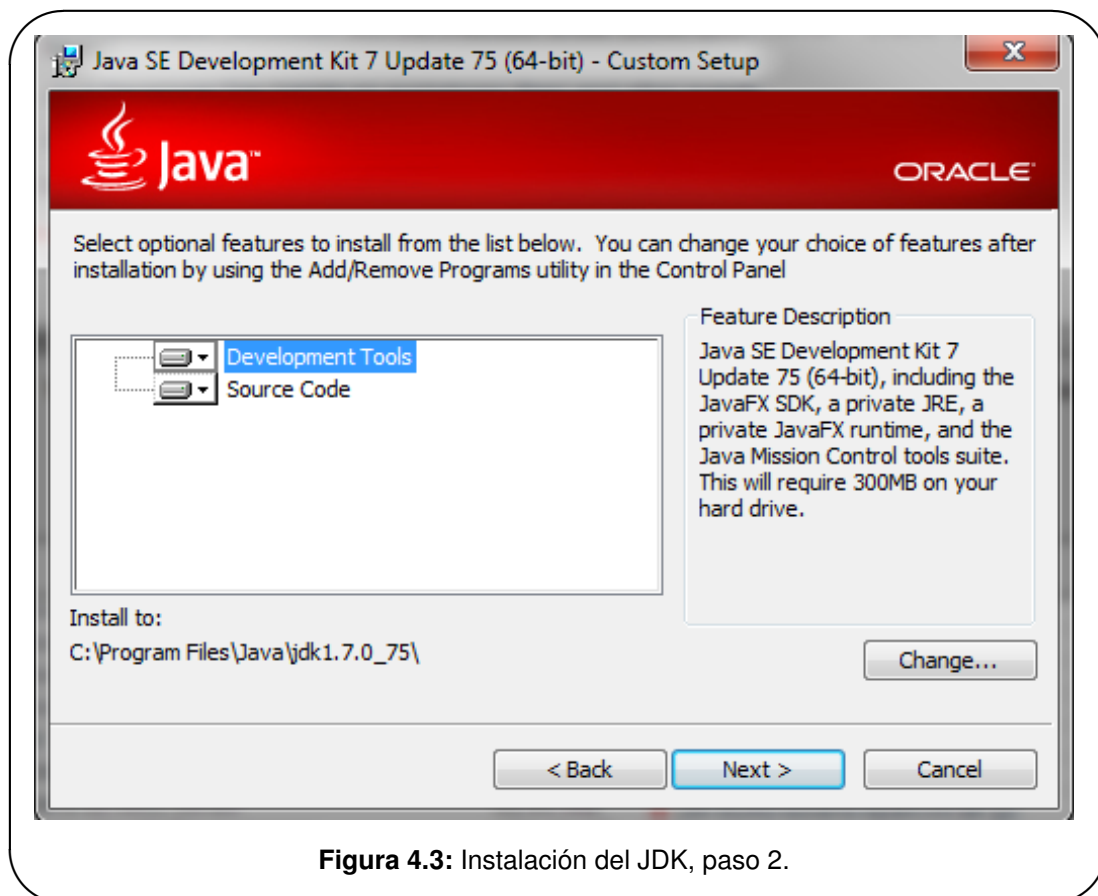
##### ■ Instalación del JDK

La primera, y más esencial de las herramientas, es el JDK de java, que es el set o conjunto de herramientas y librerías para los desarrolladores de java.

Primero deberemos ir a la página de **Oracle** en la que descargaremos el jdk, la página debería tener un aspecto más o menos como este

En dicha página tendremos que aceptar la licencia y posteriormente descargar el JDK que sirva para nuestra máquina. Una vez hemos descargado dicho archivo, lo ejecutamos. Una vez ejecutado seguimos los siguientes pasos para su instalación.





**Figura 4.3:** Instalación del JDK, paso 2.



**Figura 4.4:** Instalación del JDK, paso 3.





**Figura 4.5:** Instalación del JDK, paso 4.



## 5. ESTADO DEL ARTE

---

En este apartado...



## **6. ASPECTOS RELEVANTES DEL DESARROLLO DEL PROYECTO**

---

En este apartado se introducen los aspectos más relevantes del proyecto.

### **6.1 Resolviendo puzles en un ordenador**

La resolución...



## 7. CONCLUSIONES Y LÍNEAS DE TRABAJO FUTURAS

---

En este apartado introduciremos en primer lugar las dificultades encontradas durante el desarrollo del proyecto. Posteriormente, se hablará de las conclusiones que se obtienen tras el desarrollo del proyecto. Por último, se expondrán las líneas de trabajo futuras.

### 7.1 Dificultades encontradas

Durante el desarrollo del proyecto nos hemos encontrado varias dificultades que han hecho que el proyecto se retrase considerablemente y su avance no haya sido ni fácil ni rápido.

#### 7.1.1 Dificultades con GSOAP y Apache

Este fallo es el que más ha retrasado al proyecto y ha supuesto una dificultad enorme a la hora de llevar a cabo el mismo.

Empezamos con que para usar GSOAP se tuvo que estudiar una serie de cosas para poder adquirir los conocimientos necesarios para usar la herramienta, dichos conocimientos serán listados aquí:

- **XML:** Se tuvo que coger un nivel adecuado en el uso de XML ya que la herramienta GSOAP se basa en el uso de este tipo de archivos como medio de comunicación en las distintas peticiones y respuestas que procesa. Además el XML también es necesario para comprender el funcionamiento de SOAP y de WSDL, los cuales son completamente necesarios para entender el funcionamiento de la herramienta GSOAP.
- **SOAP:** Esta especificación se tuvo que estudiar para comprender el funcionamiento de la herramienta GSOAP y en qué se basaba su funcionamiento, entender el por qué debía funcionar la herramienta y como se realiza la comunicación gracias a ella. Aunque el SOAP no es usado directamente cuando usas GSOAP es necesario conocer esta especificación ya que GSOAP sí que usa WSDL, para el cual tenemos que tener un conocimiento básico, al menos, de SOAP para poder usarlo.
- **WSDL:** Esta otra especificación sí que se usa directamente en la herramienta GSOAP y básicamente con ella vertebras toda la aplicación que vas a hacer, de hecho tienes dos opciones:

La primera es usar un archivo WSDL donde especificas las operaciones que el servidor va a realizar, después con la herramienta GSOAP generas todos los stubs y documentos necesarios para hacer tu aplicación.

La segunda sería a través de un documento de tipo .h o una cabecera de C. Con el cuál también generas los stubs y documentos necesarios para programar tu servidor, entre dichos documentos se encontrará un archivo WSDL que contendrá la especificación de las operaciones que hay dentro del fichero cabecera que hayas usado. Pero incluso en esta opción necesitas entender SOAP y WSDL porque a través de comentarios tienes que especificar características que irán directamente al fichero WSDL, y que serán necesarios para el correcto funcionamiento del servidor.

Una vez se ha estudiado lo anterior se paso al estudio de la documentación de la herramienta GSOAP, además de el intento de hacer que funcionen sus ejemplos. Cuando se consiguió que funcionarían sus ejemplos se paso a la programación de un servidor propio, una vez se programo y se hicieron las pruebas de que estaba bien programado se procedió a intentar que este funcionará desde un cliente GSOAP. Para que funcionará con el cliente GSOAP se hizo una investigación de cómo hacer que el servidor funcionará en localhost y, siguiendo la recomendación que en la documentación de GSOAP encontramos, se instaló Apache y se inteto usar el módulo de Apache para su funcionamiento con GSOAP.

Como conclusión sacamos que, tras una larga investigación y mucho tiempo dedicada a esta herramienta, esta herramienta no tenia la documentación suficiente como para poder hacerla funcionar con Apache y, a pesar de haberlo intentado muchas veces, no conseguimos que el servidor GSOAP programado por nosotros devolviera alguna vez un resultado coherente al cliente. Finalmente desechamos la opción de trabajar con esta herramienta y le dimos un giro al proyecto con el que esperábamos tener avances más rápidos y mejores, optamos por la programación de un servidor en Flask.

### 7.1.2 Dificultades con DeepBeliefSDK

Al final no se pudo usar.

### 7.1.3 Dificultades en la instalación de Herramientas

Caffe tuvimos problemas de compatibilidad con open-cv Neuraltalk tuvimos problemas con la cantidad de dependencias que tenía.

### 7.1.4 Dificultades con NeuralTalk

Tuvimos problemas con la herramienta de MATLAB al ejecutar un script desde el servidor.

### 7.1.5 Dificultades con Android

Problemas al conectarnos con el servidor.



## **7.2 Conclusiones del Proyecto**

## **7.3 Lineas de trabajo Futuras**

