



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Gº en Ingeniería en Informática



Trabajo final del Gº Ing. Informática:

**App para el diagnóstico de fallos en
fresadoras**



Presentado por Juan Francisco Benito Cuesta
en junio de 2018
Tutor Raúl Marticorena Sánchez



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Gº en Ingeniería en Informática



D. Raúl Marticorena Sánchez, profesor del departamento de Ingeniería Civil, área de Lenguajes y Sistemas Informáticos.

Expone:

Que el alumno D. Juan Francisco Benito Cuesta, con DNI 45575555C, ha realizado el Trabajo final del Gº Ing. Informática titulado: App para el diagnóstico de fallos en fresadoras.

y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual, Se autoriza su presentación y defensa.

En Burgos a de junio de 2018

Raúl Marticorena Sánchez



Índice de contenido

Índice de ilustraciones.....	1	2.Alternativas estudiadas.....	5
Índice de tablas.....	1	2.1.Librería externa para parsear JSON. 5	
I -Introducción.....	3	V -Aspectos relevantes del desarrollo del pro- yecto	5
II -Objetivos del proyecto.....	3	VI -Trabajos relacionados	7
III -Conceptos teóricos.....	3	VII -Conclusiones y líneas de trabajo futuras 7	
1.JSON.....	4	VIII -referencias.....	7
2.API REST.....	4	Bibliografía.....	7
IV -Técnicas y herramientas.....	4	Índice alfabético.....	8
1.Metodología usada.....	5		

Índice de ilustraciones

Ilustración 1: Ejemplo de la alarma 713.....	4	Ilustración 2: Modelo Scrum.....	6
--	---	----------------------------------	---

Índice de tablas

Tabla 1: Alternativas estudiadas.....	5
---------------------------------------	---





Resumen

Aplicación en Java para Android que diagnostica los fallos de fresadoras a partir de las alarmas que producen estas. Realiza una serie de preguntas, dependiendo de la alarma que se haya activado, y el usuario debe ir eligiendo las respuestas según el estado de la fresadora.

Una vez finalizadas las preguntas, el problema se habrá resuelto o la aplicación te dirá una posible solución.

Estas alarmas con sus respectivas preguntas y respuestas están definidas en ficheros JSON. La aplicación obtiene de un API REST la alarma o alarmas que se han activado y parsea los ficheros JSON correspondientes para obtener las preguntas y respuestas de dichas alarmas.

Abstract

Application in Java for Android that diagnoses the failures of milling machines from the alarms that produce these. It asks a series of questions, depending on the alarm that has been activated, and the user must choose the answers according to the state of the milling machine.

Once the questions are finished, the problem will have been solved or the application will tell you a possible solution.

These alarms with their respective questions and answers are defined in JSON files. The application obtains from an REST API the alarm or alarms that have been activated and parses the corresponding JSON files to obtain the questions and answers of those alarms.



I - INTRODUCCIÓN

Este trabajo está pensado para los empleados de una empresa que trabajan con fresadoras. La aplicación que se realiza en este proyecto sirve para ayudar o incluso resolver los problemas y fallos que producen dichas fresadoras. Se trataría de un asistente que te ayuda realizando preguntas y que cualquier empleado puede tener en su teléfono móvil.

El trabajo consta de un proyecto de Android Studio, varios ficheros JSON y una serie de imágenes.

- El proyecto es la aplicación en sí, que consta de la funcionalidad: los activity[Activity] de cada pantalla y el modelo de clases de las alarmas, y de la interfaz gráfica: los layouts. [Layouts]
- Los ficheros JSON representan las diferentes alarmas que pueden saltar en las fresadoras. Cada fichero es una alarma y consta de su número, título, descripción, preguntas, respuestas y sus imágenes asociadas.
- Las imágenes son ayudas visuales que se aportan para resolver la situación. Cada alarma puede tener una o varias imágenes.

Esta memoria se divide en unos objetivos que se persiguen con la realización del proyecto, unos conceptos teóricos para explicar algunos detalles del trabajo, las técnicas y herramientas utilizadas en la elaboración del proyecto, aspectos relevantes a mencionar, trabajos que pueden tener relación con este, una serie de conclusiones y la bibliografía.[Bibliografía]

II - OBJETIVOS DEL PROYECTO

Los objetivos principales del proyecto son:

- Desarrollar una aplicación para que los empleados de una empresa puedan diagnosticar fácilmente los fallos de fresadoras.
- Utilizar Java para Android para desarrollar dicha aplicación.
- Realizar una interfaz gráfica usable y amigable para el usuario.
- Que dicha interfaz tenga los mismos colores, la misma fuente y logo de la empresa.
- Que la aplicación conecte con una API REST para la obtención de las alarmas activadas.
- Representar el título, descripción, preguntas y respuestas de las alarmas en ficheros JSON.
- Que la aplicación parsee dichos ficheros.[Cod. archivo][Norma ISO][Leer JSON]

III - CONCEPTOS TEÓRICOS

A continuación se explican brevemente algunos conceptos que se usan en el trabajo y sirve de ayuda entenderlos.





1. JSON

JSON (JavaScript Object Notation) es un formato de texto ligero que representa datos estructurados y se usa para su intercambio.

Puede ser utilizado independientemente de JavaScript, ya que muchos contextos de programación tienen la capacidad de parsearlo y generarlo. Es una alternativa al lenguaje XML, siendo JSON más fácil de usar.

Los tipos de datos que puede representar son: números, cadenas (string), booleanos, null, arrays de cualquiera de los tipos anteriores y propios objetos JSON.[JSON wikipedia][Intr. JSON][Info JSON]

```
{
  "Número": 713,
  "título": "AUTOMÁTICO BOMBA ASPIRACIÓN SALTADO",
  "Descripción": "La entrada E461 ha cambiado a 0, indicando que el térmico G11-Q21 se ha saltado.",
  "Preguntas": [
    {
      "Id": "1.0",
      "Texto": "¿Físicamente, el térmico G11-Q21 se ha saltado por consumo?",
      "Imagen": "",
      "Respuestas": [
        {
          "Id": "A",
          "Texto": "Sí, está saltado.",
          "Camino": "1.1",
          "Mensaje": ""
        },
        {
          "Id": "B",
          "Texto": "No, no está saltado.",
          "Camino": "0.0",
          "Mensaje": "Comprobar cableado circuito auxiliar del térmico y la entrada del automático."
        }
      ]
    },
    {
      "Id": "1.1",
      "Texto": "Si se rearma a mano, ¿se vuelve a saltar de nuevo al poco tiempo?",
      "Imagen": "",
      "Respuestas": [
        {
          "Id": "A",
          "Texto": "Sí, se vuelve a saltar.",
          "Camino": "0.0",
          "Mensaje": "Comprobar consumo de bomba y/o cambiarla."
        },
        {
          "Id": "B",
          "Texto": "No, no se vuelve a saltar.",
          "Camino": "0.0",
          "Mensaje": "Dejar bomba en observación."
        }
      ]
    }
  ],
  "Imágenes": [
    "a713",
    "a713_2"
  ]
}
```

Ilustración 1: Ejemplo de la alarma 713

2. API REST

API REST (Application Programming Interface Representational State Transfer) es una interfaz entre sistemas que utiliza el protocolo HTTP para obtener datos o realizar operaciones sobre esos datos en cualquier formato posible, como los mencionados anteriormente JSON y XML.

Las operaciones más importantes que podemos hacer son: POST (crear), GET (leer y obtener), PUT (modificar) y DELETE (eliminar). Para acceder o manipular esos datos u objetos de la REST se utiliza la URI, que es su identificador único.[API REST][REST][API REST 2][API REST 3]

IV - TÉCNICAS Y HERRAMIENTAS

La herramienta de desarrollo principal que se ha usado para llevar a cabo el proyecto ha sido



el programa Android Studio.[AS] Se trata de un entorno de desarrollo integrado que fue anunciado por Google en el año 2013, y por tanto es el IDE oficial para la creación de aplicaciones para Android. Con ella se ha programado la funcionalidad de las diferentes pantallas y su diseño gráfico.[Botones][Creación App][SplashScreen][Agregar botones][Spinner][Cambiar texto Spinner]

También se ha utilizado Eclipse para el desarrollo de un prototipo inicial de la aplicación, y StarUML para su diagrama de clases correspondiente. Se trataría de una versión en modo texto para comprobar la correcta navegación entre las preguntas de cada alarma.

Para la creación de los ficheros JSON se ha usado los editores de código fuente Notepad++ y Sublime Text.

Y para la realización del prototipo de la interfaz gráfica de la aplicación se ha utilizado el programa Pencil.

1. Metodología usada

La metodología usada para el desarrollo del software ha sido Scrum, planificando y realizando tareas cada dos semanas. Este control de las tareas se ha llevado a cabo con un repositorio en GitHub, realizando un commit por cada tarea.

Además, en este repositorio se encuentran todos los archivos que se han ido utilizando durante el trabajo.

Para almacenar y estructurar la información de las alarmas se ha utilizado los ficheros JSON mencionados en los otros apartados.

Para representar una alarma en el proyecto de Android Studio se ha empleado un modelo de clases Answer, Question y Alarm, con sus diferentes atributos y métodos.

Y para guardar las alarmas que se van parseando se ha creado una clase singleton, para que de esta forma se tenga un acceso global a la misma. Dicha clase utiliza un LinkedHashMap, siendo las claves los números de las alarmas y los valores los objetos alarmas. Se ha decidido que sea esta estructura de datos para que las alarmas se guarden en el mismo orden en el que se parsean. [Enviar datos]

2. Alternativas estudiadas

Alternativa	
Librería externa para parsear JSON	

Tabla 1: Alternativas estudiadas

2.1. Librería externa para parsear JSON

Al principio me descargué de Internet una librería para parsear ficheros JSON, llamada “json-simple-1.1.jar”[Librería JSON], que la utilicé en la aplicación en modo texto desarrollada en Eclipse. Cuando empecé con Android Studio mi intención era también usarla, pero descarté esta opción porque Android ya tiene librerías propias para parsear JSON.



V - ASPECTOS RELEVANTES DEL DESARROLLO DEL PROYECTO

El ciclo de vida utilizado para el desarrollo del proyecto ha sido Scrum, que es un tipo de



método ágil. Este ha consistido en que a partir de unos requisitos para el avance de la aplicación, se han ido realizando sprints o tareas cada dos semanas para el cumplimiento de dichos requisitos. De esta forma se obtiene un “entregable” después de ese tiempo y se va avanzando en el desarrollo de la aplicación. Después de los dos primeros meses el tiempo entre sprints pasó a ser de una semana, para llevar un mayor control del proyecto.

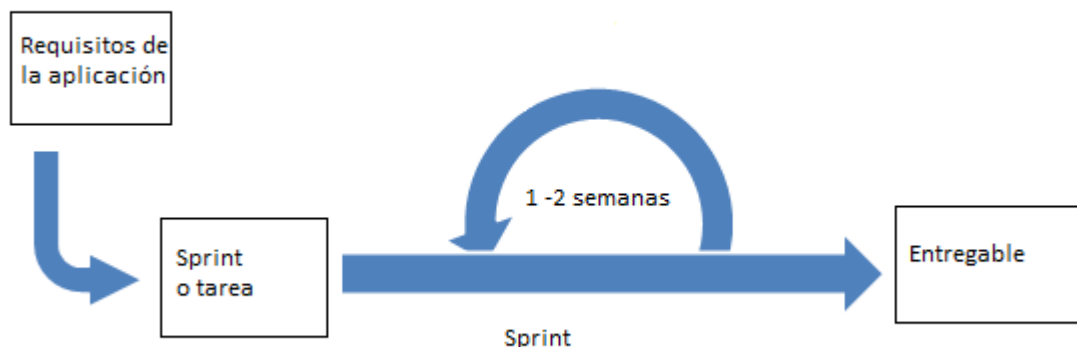


Ilustración 2: Modelo Scrum

Durante este desarrollo se ha seguido una serie de fases:

- Fase de análisis: en esta primera fase se toman las decisiones de cómo hacer el proyecto y qué pasos se han de seguir. Como por ejemplo de qué forma representar y estructurar las alarmas, cómo debe ser la interfaz de la aplicación, etc.
- Fase de diseño: en esta segunda fase se realiza el prototipo de la aplicación con la herramienta Pencil y se elabora un primer proyecto en Eclipse que representa la aplicación en modo texto.
- Fase de implementación: en esta última fase se realiza el desarrollo de la aplicación en la herramienta Android Studio, teniendo en cuenta todo lo que se ha hecho anteriormente. [Scrum]

Un aspecto importante del proyecto a comentar es la “traducción” de la información de las alarmas contenida en una hoja de cálculo a la representación de dicha información en ficheros JSON:

Las alarmas son identificadas por un número, poseen un título y una descripción. En la hoja de cálculo algunas alarmas tienen una nota a tener en cuenta. Esta nota está representada en el fichero JSON dentro del apartado “Descripción”, como si formara parte de esta.

Las preguntas están identificadas también por un número, que en el JSON es de tipo double, y que gracias a esto podemos diferenciar dos tipos de preguntas: las principales que serían las 1.0, 2.0, 3.0, etc., y las preguntas que llevan en el enunciado una posible solución y derivan de las principales, que serían las 1.1, 1.2, 2.1, etc.

De las respuestas comentar que en el JSON se identifican por letras: A, B, C, etc. y que cada una de ellas tiene un camino a seguir. Este camino es un valor de tipo double que puede ser 0.0 si la respuesta ya no tiene continuación, en ese caso se mostraría un mensaje final, o puede ser el id de una pregunta, que en este otro caso se mostraría dicha pregunta[Long]. Esta información en la hoja de cálculo se muestra al lado de la respuesta: en el primer caso es una frase que sería el mensaje final, y en el segundo caso aparece la palabra “continuar” si se debe seguir con la siguiente pregunta o directamente el número de la pregunta a la que se debe saltar.



También los ficheros JSON contienen un array con el nombre de los archivos de las imágenes que están asociadas a la alarma. Una pregunta también puede tener asociada una imagen, y está representada en el fichero de la misma forma.

VI - TRABAJOS RELACIONADOS

VII - CONCLUSIONES Y LÍNEAS DE TRABAJO FUTURAS

Las conclusiones sobre las herramientas utilizadas durante el proyecto son:

- El lenguaje JSON es de mucha utilidad a la hora de estructurar datos. Es más sencillo, más fácil de comprender y de parsear que el XML.
- El IDE Android Studio me parece una potente herramienta para la creación y desarrollo de aplicaciones. Te permite una gran variedad de opciones y funcionalidades para construir tu proyecto.

Las conclusiones sobre los resultados del proyecto son:

- Me parece buena idea “traducir” la colección de preguntas y respuestas que tiene la empresa sobre cada alarma, en una aplicación sencilla que te permita ir respondiendo y navegando sobre esas preguntas hasta llegar a una posible solución final. Me parece útil para los empleados de la empresa que dispongan de esta aplicación y puedan usarla cada vez que falle una fresadora.

El proyecto se puede mejorar o continuar en los siguientes aspectos:

- Que un empleado pueda iniciar sesión en la aplicación y pueda ver un registro de su actividad: las veces que ha utilizado la aplicación, cuando lo ha hecho, las alarmas que ha consultado, etc.
- Que se puede buscar por palabras clave la alarma correspondiente. Un usuario introduciría varias palabras que tengan que ver con el fallo de la fresadora, y la aplicación te mostraría la alarma o alarmas que más se aproximen a lo que haya escrito el usuario.

VIII - REFERENCIAS

Bibliografía

Activity: TuTutorial, Cómo pasar de un Activity a otro en una aplicación en Android Studio, 2017, <https://www.youtube.com/watch?v=41SYosZP708>

Layouts: Código Alonso, Tutorial 010 Programación Android: Layouts, organización de componentes con Android Studio., 2013, <https://www.youtube.com/watch?v=DXhMcraA45q8&t=792s>

Bibliografía: Cristina Batallas, Base de datos bibliográfica en Writer, 2011, <https://www.youtube.com/watch?v=19N97pi0Ky8>

Cod. archivo: Usuarios de Stack Overflow, ¿Qué codificación debería usar en un XML con texto en español?, 2015, <https://es.stackoverflow.com/questions/316/qu%C3%A9-codificaci%C3%B3n-deber%C3%ADa-usar-en-un-xml-con-texto-en-espa%C3%B1ol>

Norma ISO: Colaboradores de Wikipedia, ISO/IEC 8859-1, 2017, https://es.wikipedia.org/w/index.php?title=ISO/IEC_8859-1&oldid=102959948

Leer JSON: Gerard Coll, Leer archivo JSON localmente en Android, 2013, <http://dev4phones.blogspot.com.es/2015/12/leer-archivo-json-localmente-en-android.html>





JSON wikipedia: Colaboradores de Wikipedia, JSON, 2018, <https://es.wikipedia.org/w/index.php?title=JSON&oldid=106104589>

Intr. JSON: Colaborador de la página, Introducción a JSON, 2018, <https://json.org/json-es.html>

Info JSON: Juan Sepulveda, Trabajando con JSON, 2017, <https://developer.mozilla.org/es/docs/Learn/JavaScript/Objects/JSON>

API REST: BBVA Open4U, API REST: qué es y cuáles son sus ventajas en el desarrollo de proyectos, 2016, <https://bbvaopen4u.com/es/actualidad/api-rest-que-es-y-cuales-son-sus-ventajas-en-el-desarrollo-de-proyectos>

REST: Colaboradores de Wikipedia, Transferencia de Estado Representacional, 2018, https://es.wikipedia.org/w/index.php?title=Transferencia_de_Estado_Representacional&oldid=105107212

API REST 2: Fazt, Api Rest con Nodejs | ¿Que es Api Rest?, Clase 1, 2017, <https://www.youtube.com/watch?v=wMwON-gwyVM&t=1s>

API REST 3: Fazt, Api Rest con Nodejs | Verbos HTTP, Clase 2, 2017, <https://www.youtube.com/watch?v=ArdQcI2X1cc>

AS: Colaboradores de Wikipedia, Android Studio, 2018, https://es.wikipedia.org/w/index.php?title=Android_Studio&oldid=107398073

Botones: Miguel, Creación dinámica de botones, 2017, <http://umhandroid.momrach.es/creacion-dinamica-de-botones/>

Creación App: TuTutorial, Cómo crear una aplicación para Android con Android Studio, 2017, <https://www.youtube.com/watch?v=XKU7MlqmgWo&t=631s>

SplashScreen: TuTutorial, Cómo añadir un Splash Screen o pantalla de bienvenida a tu aplicación Android, 2017, <https://www.youtube.com/watch?v=fR1xcOxprSY>

Agregar botones: TuTutorial, Cómo agregar botones a tu aplicación en Android Studio, 2017, <https://www.youtube.com/watch?v=R-YMH5I677A>

Spinner: Cesar Augusto Perez Tafur, Tutorial 12 Como usar un Spinner en mi aplicacion Android, 2016, <https://www.youtube.com/watch?v=F9GV7vuIghw&t=273s>

Cambiar texto Spinner: Gonzalo Eduardo Pérez Correa, Android - Cambiar el tamaño del texto de un spinner, 2016, <https://www.youtube.com/watch?v=ciQnbxjP7SY&t=20s>

Enviar datos: TuTutorial, Cómo enviar datos de un Activity a otro en Android Studio, 2017, <https://www.youtube.com/watch?v=lAnpyZmrwX8>

Librería JSON: Desconocido, Download json-simple-1.1.jar : json simple « j « Jar File Download, 2017, <http://www.java2s.com/Code/Jar/j/Downloadjsonsimple11jar.htm>

Scrum: Colaboradores de Wikipedia, Scrum (desarrollo de software), 2018, [https://es.wikipedia.org/w/index.php?title=Scrum_\(desarrollo_de_software\)&oldid=107627551](https://es.wikipedia.org/w/index.php?title=Scrum_(desarrollo_de_software)&oldid=107627551)

Long: Usuarios de Stack Overflow, Convert Long into Integer, 2011, <https://stackoverflow.com/questions/5804043/convert-long-into-integer>

Índice alfabético

R Referencia.....7



Impreso en Burgos el jueves, 24 de mayo de 2018