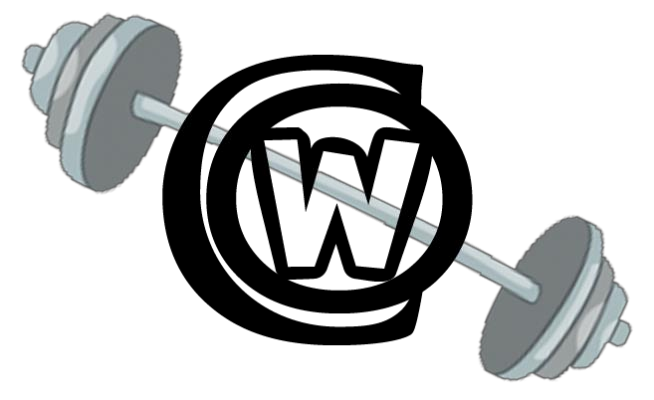
WorkCloseOut



Arquitectura e Integración de Sistemas Software Grado de Ingeniería del Software

Curso 2º

María Inmaculada Martín Galván (inma.martin.galvan@hotmail.com) Alfonso Soldado Caro (alfonsosoldado23@hotmail.com)

José Manuel Lara Villa (lara\_villa10@hotmail.com)

José Félix Gómez Rodríguez (gomezrodriguezjosefelix@gmail.com)

Tutor: Adela Del Río Ortega Número de grupo: 3

Enlace de la aplicación: [https://workcloseout.appspot.com](https://workcloseout.appspot.com/) Enlace de proyecto en projETSII:

<https://projetsii.informatica.us.es/projects/ge43rfzxqwgkq2tjzxx>

**HISTORIAL DE VERSIONES**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Detalles** | **Participantes** |
| 20/03/2016 | 1.0 | Incluye introducción, prototipos de las interfaces de usuario y diagramas UML de componentes y despliegue. | Mª Inmaculada Martín Galván Alfonso Soldado Caro  José Manuel Lara Morilla  José Félix Gómez Rodríguez |
| 11/04/2016 | 1.1 | Empezamos a implementar las API’s de google mapas y spotify, pero observamos que la dificultad es mayor de lo esperada y no funcionan. | Mª Inmaculada Martín Galván Alfonso Soldado Caro  José Manuel Lara Morilla  José Félix Gómez Rodríguez |
| 09/05/2016 | 2.0 | - Dejamos de lado las API’s de Facebook por su dificultad y GoogleMaps por no concordar bien con la idea.  - Conseguimos implementar la API de youtube y flickr. | Mª Inmaculada Martín Galván Alfonso Soldado Caro  José Manuel Lara Morilla  José Félix Gómez Rodríguez |
| 18/05/2016 | 2.1 | Conseguimos el token para GoogleFit pero no conseguimos que nos funcione y obtener los datos. | Mª Inmaculada Martín Galván Alfonso Soldado Caro  José Manuel Lara Morilla  José Félix Gómez Rodríguez |
| 25/05/2016 | 2.2 | Todas las API’s funcionan y realizamos la APIRest. | Mª Inmaculada Martín Galván Alfonso Soldado Caro  José Manuel Lara Morilla  José Félix Gómez Rodríguez |
| 29/05/2016 | 3.0 | Con toda la implementación de las API’s realizadas y con la APIRest terminada, terminamos el trabajo ultimando los detalles del css. | Mª Inmaculada Martín Galván Alfonso Soldado Caro  José Manuel Lara Morilla  José Félix Gómez Rodríguez |

**AUTOEVALUACIÓN**

|  |  |  |  |
| --- | --- | --- | --- |
| **Alumno** | **Entregable 1** | **Entregable 2** | **Final** |
| Alfonso Soldado Caro | 100% | - | 100% |
| María Inmaculada Martín Galván | 100% | - | 100% |
| José Manuel Lara Morilla | 100% | - | 100% |
| José Félix Gómez Rodríguez | 100% | - | 100% |

Índice

1. [Introducción 5](#_bookmark0)
   1. [Aplicaciones integradas 5](#_bookmark1)
   2. [Evolución del proyecto 6](#_bookmark2)
2. [Prototipos de interfaz de usuario 6](#_bookmark3)
   1. [Vista 1 6](#_bookmark4)
   2. [Vista 2 7](#_bookmark5)
3. [Arquitectura 8](#_bookmark6)
   1. [Diagrama de componentes 9](#_bookmark7)
   2. [Diagrama de despliegue 10](#_bookmark8)
4. [Implementación 11](#_bookmark9)
5. [Pruebas 12](#_bookmark10)
6. [Manual de usuario 13](#_bookmark11)
   1. [Mashup 13](#_bookmark12)
   2. [API REST 13](#_bookmark13)

[Referencias 14](#_bookmark14)

# Introducción

Con “WorkCloseOut” buscamos dar respuesta al cambio de tendencia por una parte de la sociedad, que busca nuevas formas de hacer ejercicio físico, de una forma más barata, efectiva y a la vez con menos tiempo para poder dedicarle al deporte, en una sociedad cada vez más ocupada. El deporte tan fundamental para la salud de cada uno y el conjunto de una sociedad saludable.

Para hacer llegar una nueva idea mediante este método, muchas veces encontramos que uno tiene la voluntad, pero no el conocimiento para la realización de las técnicas necesarias. Saber cuáles son los puntos situados en la ciudad dedicados para la práctica de este deporte y porque no, un extra de motivación llevando la música que más nos gusta y compartiendo con tus amigos tus marcas.

Desde WorkCloseOut pensamos en las necesidades de las personas, por ello y mediante este Mashup integraremos varias API’s, tales como GoogleFit, YouTube y Flickr. Pretendemos desarrollar un producto saludable, barato, diferente y motivador para que saques donde no lo hay, ese tiempo que siempre quisiste para tu entrenamiento personal.

## Aplicaciones integradas

|  |  |
| --- | --- |
| **Nombre aplicación** | **URL documentación API** |
| Flickr | https://www.flickr.com/services/api/ |
| YouTube | https://developers.google.com/youtube/?hl=es |
| GoogleFit | https://developers.google.com/fit/overview |

TABLA 1. APLICACIÓN INTEGRADAS

Desde YouTube se mostrará un vídeo con el ejercicio que el usuario desea realizar para usarlo de guía.

Flickr mostrará imágenes relacionadas con el ejercicio buscado en youtube, con imágenes motivadoras.

Finalmente, con GoogleFit podrás obtener los datos que tengas en tu cuenta sincronizada con google fit , una app especializada en monitorizar tu activadad física .

## Evolución del proyecto

Estas primeras semanas de proyecto estamos asentando los conocimientos necesarios para llevar a cabo nuestro trabajo, lo más fielmente posible a la descripción que realizamos en este documento. Seguiremos describiendo la evolución del mismo en este apartado en las siguientes entregas.

Tras la defensa del primer entregable realizamos las correcciones del mismo y empezamos con la búsqueda de las API’s que implementaremos. Empezamos su implmentación, pero no conseguimos que funcionen Facebook ni spotify.

Tras no poder hacer la entrega 2 porque su funcionamiento no era el correcto, empezamos con las API’s que teníamos de altenativas, las cuales eran Flickr y GoogleFit, las cuales al poco tiempo conseguimos realizar.

Utilizamos finalmente las APIS’s de youtube, flickr y googleFit, y utilizamos como complemento el javascript de Facebook para compartir los resultados.

En las ultimas semanas perfeccionamos las API’s y la última semana conseguimos realizar la APIRest y terminamos de realizar el CSS.

# Prototipos de interfaz de usuario

## Vista 1

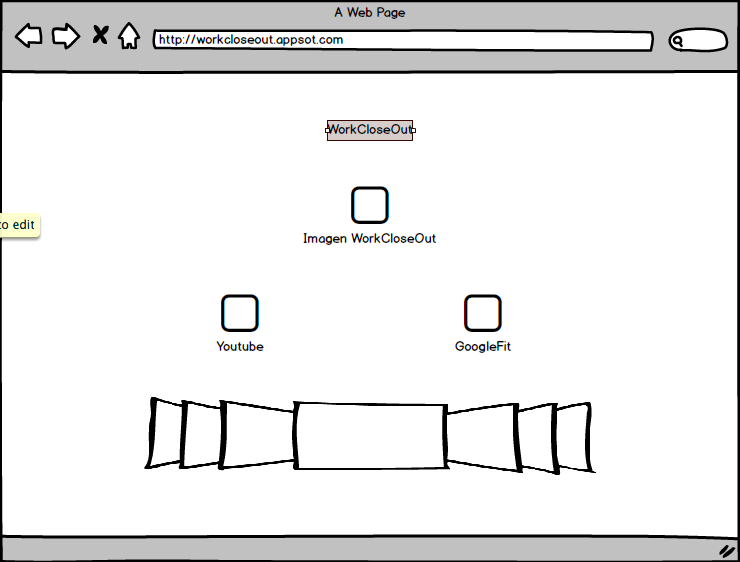


FIGURA 1. PROTOTIPO DE INTERFAZ DE USUARIO DE LA VISTA 1

Esta es la vista principal de la página donde se muestra en la cabecera el nombre y el logo de la aplicación. A continuación se muestra dos iconos, a la izquierda vemos el icono de youtube, en el cual te llevara a la vista dos. A la derecha vemos el icono de GoogleFit que nos llevará a la vista tres.

## Vista 2

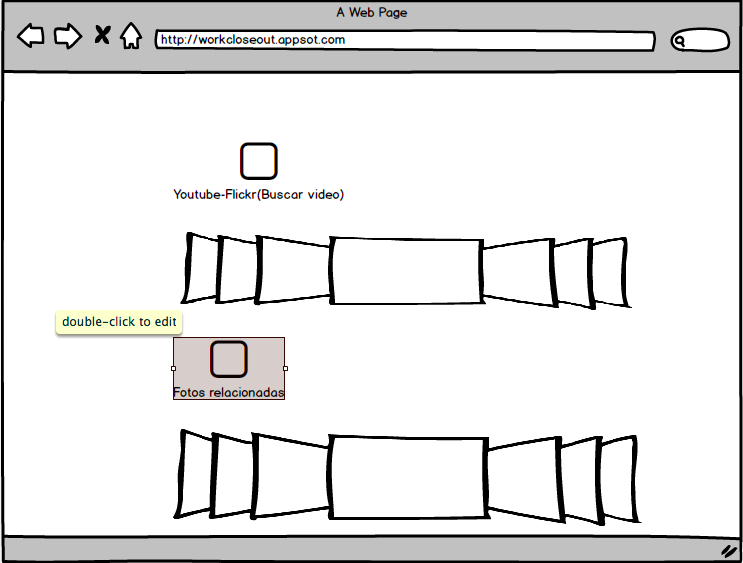


FIGURA 2. PROTOTIPO DE INTERFAZ DE USUARIO DE LA VISTA 2

Cuando el usuario selecciona youtube para buscar videos, obtiene videos relacioanda con la búsqueda realizada, y además obtendrá fotos motivadoras. Además podrá compartir en Facebook a través del icono los ejercicios hechos así como la url de los videos y las fotos que haya visto.

El nombre y el logo se mantendrá fijo.

* 1. Vista 3

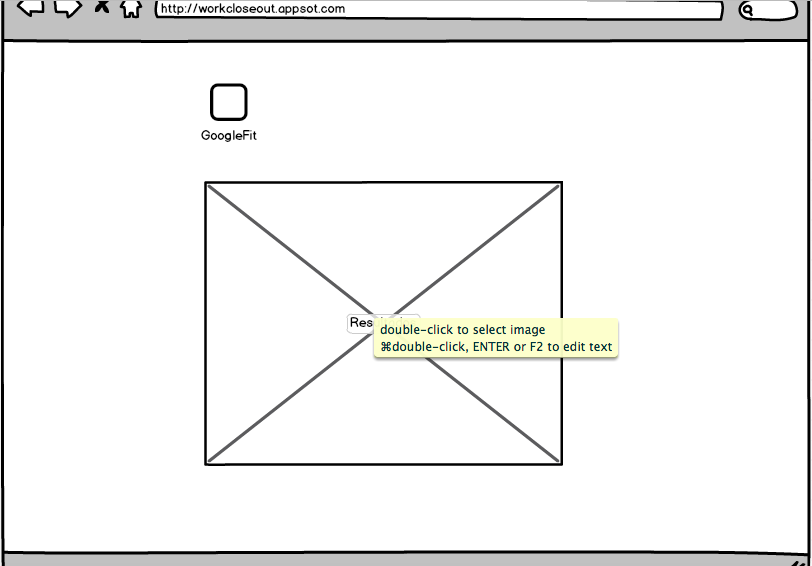


FIGURA 3. PROTOTIPO DE INTERFAZ DE USUARIO DE LA VISTA 3

Cuando el usuario clica en ‘GoogleFit se le llevará a esta vista donde podrá consultar todos sus ejercicios que haya realizado a través de google fit.

# Arquitectura

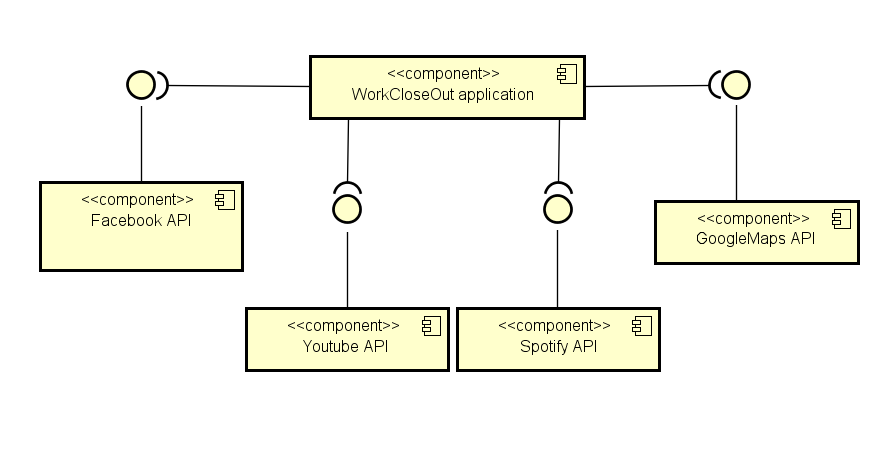
Para la realización de los diferentes diagramas hemos utilizado la aplicación Astah.

## Diagrama de componentes

Un diagrama de componentes se utiliza para representar la arquitectura lógica de un sistema. Un diagrama de tipo UML (Lenguaje unificado de modelado).

Muestran las interfaces por las que los componentes se relacionan.

Representan cómo un sistema de software es dividido en componentes que están conectados mediante sus respectivas interfaces.



El tipo de uml utilizado es el de vista de caja negra, donde los detalles de implementación quedan ocultos.

## Diagrama de despliegue

Un diagrama de despliegue es un diagrama del tipo UML.

Los diagramas de despliegues muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos.

La vista de despliegue representa la disposición de las instancias de componentes de ejecución en instancias de nodos conectados por enlaces de comunicación.

La imagen que proporcionamos a continuación muestra dos nodos, cliente y servidor, conectados por el protocolo HTTP, el cual se ejecuta el artefacto WorkCloseOut.war del sistema en un entorno de ejecución (en el caso del servidor).

# Implementación

Describir brevemente los aspectos de la implementación que creen da más mérito al trabajo. Añadir algún fragmento de código si se ve necesario.

# Pruebas

Documentar las pruebas realizadas a la aplicación. Se valorará muy positivamente automatizar pruebas con JUnit siguiendo el esquema visto en prácticas.

|  |  |
| --- | --- |
| **ID** | **Prueba 1** |
| **Descripción** | Prueba para la detección de errores al implementar búsquedas en Spotify usando servicios RESTful. |
| **Entrada** | Se hace uso de la librería XXX para invocar al servicio usando la URI YYY desde nuestra aplicación GWT. |
| **Salida esperada** | Los datos devueltos en formato JSON son mapeados a una clase Java y a continuación se muestran por pantalla. |
| **Resultado** | **EXITO** |

# Manual de usuario

## Mashup

Indique textualmente e incluyendo capturas de pantalla el manual de uso del mashup.

## API REST

Indique la documentación de la API REST (contrato) implementada [2]. Cómo mínimo, la API debería incluir:

* + 1. Protocolo de aplicación empleado por el servicio.
    2. URIs para invocar a las operaciones del servicio.
    3. Formato empleado para las representaciones de los recursos.
    4. Códigos de estado empleados por el servicio.
    5. Ejemplos de uso.

Está información también debe facilitarse en formato HTML como parte de la aplicación.

# Referencias

[1] *Balsamiq*. [http://balsamiq.com/.](http://balsamiq.com/) Accedido en Enero 2014.

[2] J. Webber, S. Parastatidis y I. Robinson. *REST in Practice: Hypermedia and Systems Architecture.* O'Reilly Media. 2010.