# CONTRATO DE PROYECTO

# Estructura

El contrato de proyecto de la asignatura tiene 5 partes diferenciadas:

* Identificación del grupo
* Objetivos del proyecto
* Funcionalidades a desarrollar
* Mecanismos de evaluación
* Materiales necesarios

# Identificación del grupo

Blue Bash:

1. Angel Cañal Muniesa: 716205
2. Roberto C. Claver Ubiergo: 720100
3. Alberto Mur Rodrigo: 697589
4. Víctor Fernández Melic: 651943

# Objetivos

Un objetivo del proyecto es una frase clara y concisa que define claramente qué se quiere aprender durante el desarrollo del proyecto. Es preferible usar objetivos precisos. Por ejemplo:

“*Los objetivos de este proyecto son los siguientes:*

* *O1. Ser capaces de desarrollar una aplicación Web que pueda ser desplegada en producción.*
* *O2. Afianzar capacidades de trabajo en equipo, es decir, trabajar en grupo y no agrupados.*
* *O3. Afianzar nuestro conocimiento en tecnologías nuevas y estandarizadas: Kotlin, Thymeleaf, Gradle, Git, Travis, Spring.*
* *O4. Seguir desarrollo TDD (Test Driven Development) .*
* *O5. Ser capaces de desplegar nuestra aplicación en un entorno cloud de producción (Google Cloud).*

# Funcionalidades

“*Las funcionalidades que van a ser implementadas para conseguir los objetivos marcados son las siguientes:*

* *F0. La aplicación será capaz de acortar, almacenar y recuperar URI’s.*
* *F1. La aplicación devolverá información sobre el sistema. El objetivo es aprender los conceptos asociados a la monitorización de una aplicación Web. La información como mínimo incluirá la carga del sistema.*
* *F2. La aplicación será capaz de limitar el número de redirecciones. El objetivo es aprender a controlar el consumo de recursos por parte de una aplicación Web. Entendemos por limitar el número de redirecciones el establecimiento de un límite al número de peticiones de redirección a una determinada URI que se pueden atender en un tiempo determinado.*
* *F3. La aplicación permite 2 tipos de uso, gratuito con limitaciones y de pago con acceso a todas las funcionalidades.*
* *F4. La aplicación permitirá elegir el nombre de la URI acortada a los usuarios de pago.*
* *F5. La aplicación permitirá generar un código QR asociado a la URI acortada.*
* *F6. La aplicación permitirá al usuario de pago recuperar estadísticas relacionadas con sus URI’s: Número de URI’s registradas, número de peticiones a cada URI…*
* *F7. La aplicación permite a los usuarios de pago mantener URI’s registradas de forma permanente, mientras que las URI’s de los usuarios gratuitos caducarán pasado un plazo determinado.*
* *F8. La aplicación verificará periódicamente que la URI a recortar es alcanzable, de tal forma que si no es alcanzable durante un determinado periodo, la URI será borrada del sistema. También se verificará que la URI esté disponible cada vez que se realiza una petición. El objetivo es no mantener información inútil en nuestro sistema.*
* F9: La aplicación permitirá opciones avanzadas en la generación de códigos QR para los usuarios de pago.

Una vez definidas hay que repartir 60 puntos entre dichas funcionalidades.

Este reparto se presentará utilizando una tabla como la siguiente:

|  |  |  |  |
| --- | --- | --- | --- |
| *ID* | *Peso* | *Funcionalidad* | *Objetivos* |
| *F0* | *15* | *La aplicación será capaz de acortar, almacenar y recuperar URI’s.* |  |
| *F1* | *10* | *La aplicación devolverá información sobre el sistema y estadísticas. (WS)* |  |
| *F2* | *10* | *La aplicación será capaz de limitar el número de redirecciones* |  |
| *F3* | *5* | *La aplicación permitirá elegir el nombre de la URI acortada (OK)* |  |
| *F4* | *5* | *La aplicación permitirá generar un código QR asociado a la URI acortada. (OK)* |  |
| *F5* | *15* | *La aplicación verificará periódicamente que la URI a recortar es alcanzable (RPC)* |  |

# Evaluación

En la tercera parte hay el grupo tiene que diseñar un método de evaluación automático que sea interesante, creativo y estimulante para cada funcionalidad. Tened la seguridad de que existe un método de evaluación automático para cada una de las funcionalidades indicadas.

Recordad la regla de evaluación general:

* 3 x corrección - ¿hace lo que se espera?
* 2 x diseño - ¿es robusto? ¿es mantenible? ¿es extensible? ¿tiene tests?
* 1 x estilo - ¿es un código legible? ¿está bien documentado? ¿buena UI?

Para que tengáis una orientación práctica hemos definido tres niveles de servicio: *Básico*, *Diseño* y *Estilo*. El primero correspondería a una solución correcta pero simple, el segundo a una solución que además de correcta refleja que hacer un buen diseño para implementarla, y el tercero sería una solución que además muestra que se han cuidado los detalles.

Por ejemplo, si la funcionalidad es “l*a aplicación verificará que la URI a recortar es alcanzable*”, los tres niveles de servicio pueden ser:

* *Correcto*: solo se puede crear una URI acortada si se verifica que una petición HTTP a la URI original es correcta.
* *Diseño*: se hará más robusta mediante la verificación periódica de que dicha URI original sigue devolviendo una respuesta correcta; si deja de devolver una respuesta correcta no se redireccionará hasta que vuelva a devolver un estado; toda la funcionalidad está cubierta por pruebas.
* *Estilo*: el API estará documentado.

El problema es que la propuesta que se haga puede ser un tanto imprecisa. Tras la entrega de la propuesta habrá una negociación con el profesor para precisar mejor los niveles de servicio:

* *Correcto*: solo se puede crear una URI acortada si se verifica que una petición HTTP GET a la URI original devuelve una respuesta con estado 200.
* *Diseño*: se hará más robusta mediante la verificación periódica de que dicha URI original sigue devolviendo una respuesta con estado 200; si deja de devolver el estado 200 no se redireccionará hasta que vuelva a devolver el estado 200; toda la funcionalidad está cubierta por pruebas automáticas de unidad u de integración.
* *Estilo*: el API relacionado está documentado utilizando Open API[[1]](#footnote-1), hay una consola Web interactiva que visualiza dicha documentación y que permite probar dicha API[[2]](#footnote-2) y hay documentación en AsciiDoc[[3]](#footnote-3) explicando el funcionamiento del código que ha sido convertida a HTML5 con Asciidoctor[[4]](#footnote-4).

Para cada funcionalidad la evaluación se puede presentar de la siguiente forma:

|  |  |
| --- | --- |
| *F0* | *La aplicación será capaz de acortar, almacenar y recuperar URI’s.* |
| *Correcto* | * *Tras acortar la URI, el valor almacenado en almacenamiento persistente es la URI del usuario y el hash devuelto al usuario es el identificador del valor almacenado en almacenamiento persistente.* |
| *Diseño* | * *La robustez del sistema radica en la imposibilidad de existencia de dos identificadores que redireccionen a dos URI’s distintas.* |
| *Estilo* | * *El código se documentará siguiendo la guía Java* |

|  |  |
| --- | --- |
| *F1* | *La aplicación devolverá información sobre el sistema.* |
| *Correcto* | * *Se puede conocer la carga actual del sistema* |
| *Diseño* |  |
| *Estilo* | * *El código se documentará siguiendo la guía Java* |

| *F2* | *La aplicación será capaz de limitar el número de redirecciones* |
| --- | --- |
| *Correcto* | * *Tras un número y un plazo determinado de redirecciones, la aplicación mostrará una página de error avisando que no se redireccionará más.* |
| *Diseño* | * *Tras el número especificado de redirecciones y antes de que se cumpla el plazo se recibe alguna petición más, el sistema devolverá un error 429 (Too Many Requests)* |
| *Estilo* | * *El código se documentará siguiendo la guía Java* |

| *F3* | *La aplicación permite 2 tipos de uso, gratuito con limitaciones y de pago.* |
| --- | --- |
| *Correcto* | * *Los usuarios gratuitos no podrán acceder a ninguna de las funcionalidades reservadas a los usuarios de pago.* |
| *Diseño* |  |
| *Estilo* |  |

| *F4* | *La aplicación permitirá elegir el nombre de la URI acortada a los usuarios de pago.* |
| --- | --- |
| *Correcto* | * *SI no hay ninguna URI con el nombre elegido por el usuario se creará una nueva URI con ese identificador* |
| *Diseño* | * *Si ya existe en el sistema una URI con el nombre que ha proporcionado el usuario, se denegará esa creación.* |
| *Estilo* | * *Si sucede un error, la página no perderá ni la URI ni el código asignado, siendo consistentes con las Heurísticas de Nielsen* |

| *F5* | *La aplicación permitirá generar un código QR asociado a la URI acortada.* |
| --- | --- |
| *Correcto* | * *Tras solicitar la creación de los códigos QR, el sistema devolverá un código QR que, una vez analizado redireccione a la URI generada por el sistema, no a la acortada (original).* |
| *Diseño* | * *Un usuario no registrado no va a poder generar un código QR, mostrando un error si esto ocurriese.* |
| *Estilo* | * *El código se documentará siguiendo la guía Java* |

| *F6* | *La aplicación permitirá al usuario de pago recuperar estadísticas* |
| --- | --- |
| *Correcto* | * *Si un usuario de pago solicita estadísticas, se devolverán aquellas estadísticas relacionadas con uno de sus enlaces.* |
| *Diseño* | * *Un usuario normal o no registrado no podrá recuperar estadísticas, mostrando un error si esto ocurriese.* |
| *Estilo* | * *El código se documentará siguiendo la guía Java. Además, las estadísticas se mostrarán mediante gráficas.* |

| *F7* | *La aplicación permite a los usuarios de pago mantener URI’s registradas de forma permanente.* |
| --- | --- |
| *Correcto* | * *Las URI’s de un usuario de pago se mantendrán de forma permanente en el almacenamiento persistente.* |
| *Diseño* | * *Si un usuario normal o no registrado crea una URI, será eliminada automáticamente al transcurrir un plazo determinado* |
| *Estilo* | * *El código se documentará siguiendo la guía Java.* |

| *F8* | *La aplicación verificará periódicamente que la URI a recortar es alcanzable* |
| --- | --- |
| *Correcto* | * *Solo se puede crear una URI acortada si se verifica que una petición HTTP GET a la URI original devuelve una respuesta con estado 200* |
| *Diseño* | * *Se verifica periódicamente que la URI original devuelve una respuesta con estado 200.* |
| *Estilo* | * *Toda la funcionalidad del API está documentada mediante utilizando Open API* |

| *F9* | *La aplicación permitirá opciones avanzadas en la generación de QR* |
| --- | --- |
| *Correcto* | * *Tras crear un código QR, los usuarios de pago podrán personalizar su código QR.* |
| *Diseño* | * *Los usuarios gratuitos y no registrados no tendrán la posibilidad de personalizar sus códigos QR* |
| *Estilo* | * *El código se documentará siguiendo la guía Java* |

# Autocomprobación

Antes de la negociación final con el profesor podéis utilizar estas preguntas para comprobar su adecuación

* ¿Los objetivos de aprendizaje son claros y realistas?
* ¿Se os ocurre algún otro que podáis considerar?
* ¿Las funcionalidades a implementar son razonables, apropiadas y eficaces para aprender lo que se desea?
* ¿Piensas que se debería incluir alguna más?
* ¿Los mecanismos de evaluación son válidos y están en consonancia con el nivel de servicio a evaluar?
* ¿Los mecanismos de evaluación de una funcionalidad están relacionados con los objetivos de aprendizaje?
* ¿Los materiales que hemos marcado como necesarios son los adecuados?

Es una buena idea que el grupo se haga estas preguntas para refinar la primera versión del contrato hasta que os sintáis satisfechos con él.

Una vez realizado estos pasos estáis en condiciones de negociarlo con el profesor.

# Negociación y firma

Tras entregar el proyecto se entrará en un proceso de negociación. Este proceso tiene tres objetivos:

* Que los proyectos de los equipos tengan un esfuerzo similar.
* Que funcionalidades similares tengan criterios de corrección similares.
* Que las evaluaciones sean no ambiguas para facilitar la autoevaluación.

El proceso se realizará una vez que el profesor tenga todos los proyectos. Se realizará mediante reuniones con cada grupo cuyo objetivo será elaborar un acuerdo consensuado.

Tras la negociación, el documento resultante será firmado por los miembros del grupo y el profesor.

1. https://www.openapis.org/ [↑](#footnote-ref-1)
2. https://springframework.guru/spring-boot-restful-api-documentation-with-swagger-2/ [↑](#footnote-ref-2)
3. http://asciidoc.org/ [↑](#footnote-ref-3)
4. http://asciidoctor.org/ [↑](#footnote-ref-4)