Trivial4a

Índice de contenidos

[Hoja de identificación 1](#_Toc418423638)

[Introducción 2](#_Toc418423639)

[Planteamiento del problema 2](#_Toc418423640)

[Identificación de Interesados 2](#_Toc418423641)

[Metodología usada 2](#_Toc418423642)

[Historias de usuario y roles 2](#_Toc418423643)

[Identificación de atributos de calidad 2](#_Toc418423644)

[Acercamiento a la solución 2](#_Toc418423645)

[Lista de atributos de calidad e interesados 2](#_Toc418423646)

[Descripción del negocio de la solución 2](#_Toc418423647)

[Escenarios de calidad 2](#_Toc418423648)

[Vistas 2](#_Toc418423649)

# Hoja de identificación

|  |  |
| --- | --- |
| Nombre del trabajo: | Trivial4a |
| Fecha: | Mayo 2015 |
| Autores: | |  |  | | --- | --- | | Nombre | UO | | Jorge Sierra Miranda | 226566 | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  | |

# Introducción

La arquitectura descrita en este documento pretende dar cabida al desarrollo de una plataforma de juego. Esta plataforma pretende imitar al juego “Trivial” además de ofrecer funcionamiento adicional.

En el documento se describirá la arquitectura del sistema comparándolo con los patrones arquitectónicos conocidos resaltando aquellas características que se hayan intentado obtener con dichos patrones.

# Planteamiento del problema

NoGame, una empresa del sector de los videojuegos está interesada en la creación de una aplicación con un funcionamiento similar al juego de mesa “Trivial”.

La aplicación no será sólo una plataforma de juego, sino que deberá además guardar registros de las actividades de los jugadores.

Debido a la naturaleza del juego las preguntas deberán poder añadirse, quitarse, y cambiarse; permitiendo jugar cada vez con un set de preguntas distinto.

Adaptándose a la las nuevas tecnologías el sistema estará disponible para jugar en aplicaciones de escritorio o a través de la web.

Por último, los datos recopilados por la información deber estar disponibles para los administradores del sistema. Esta funcionalidad permitirá la reutilización de la plataforma en entornos profesionales o educacionales para la realización de controles o estadísticas de personal.



Ilustración - Diagrama de contexto

# Metodología usada

El estudio de la arquitectura se realiza siguiendo el método de ADD (Atribute-Driven Design) (Bass, Clements, & Kazman, Software Architecture in Practice, Second Edition, 2003).

# Identificación de Interesados

Los interesados en la aplicación, también llamados *stakeholders* son los siguientes:

* Usuarios
* Administradores
* Desarrolladores

## Descripción de los interesados

Se tendrán en cuenta las distintas características de los *stakeholders* para el desarrollo del sistema. Cada tipo de *stakeholder* tendrá unos objetivos distintos en la aplicación.

### Usuarios

Son los que utilizarán el sistema participando en juegos y respondiendo preguntas.

Deberán estar registrados en el sistema.

Se guardarán los datos de cómo estos usuarios interactúan con el sistema.

### Administradores

Son los usuarios encargados del sistema que tendrán acceso a los datos recopilados por la aplicación.

### Responsables de NoGame

NoGame como empresa contratante es un interesado en la aplicación.

No participarán directamente en la aplicación pero sus necesidades son aquellas que la aplicación intenta cubrir.

### Desarrolladores

Son aquellas personas designadas al desarrollo de la aplicación.

Interactuarán con el sistema en la fase de desarrollo y tendrán como objetivo crear un sistema que sea fácil de mantener, con un coste reducido y abierto para la extensión de su funcionalidad.

# Roles e historias de usuario

## Roles

Los roles en el sistema se corresponden con dos tipos de interesados.

* Usuarios
* Administradores

## Historias de usuarios

|  |
| --- |
| Como administrador quiero poder añadir preguntas a la aplicación desde ficheros GIFT. |
| etc |
|  |
|  |
|  |
|  |
|  |

# Identificación de atributos de calidad

Los atributos de calidad identificados para este sistema son:

#### Disponibilidad

El sistema debe poder estar disponible las 24 del día.

#### Modificabilidad

**Facilidad de cambio**: El sistema tiene que poder sufrir cambios como nuevos ficheros fuente, nuevos formatos de representación y posibles cambios de base de datos sin que estos resulten costosos.

**Escalabilidad del sistema**: El incremento del número de usuarios y de las preguntas almacenadas no deben suponer cambio alguno para la aplicación.

#### Rendimiento

Las acciones interactivas realizadas sobre el sistema deben ser instantáneas a la vista de los usuarios.

#### Seguridad

Los accesos a la aplicación deben ser controlados con cuentas de usuario y contraseña.

Se debe asegurar la integridad de los datos almacenados.

#### Testabilidad

El funcionamiento de la aplicación se debe probar con facilidad.

#### Usabilidad

La aplicación debe ser amigable con el usuario, es decir, fácil de usar y entender.

# Acercamiento a la solución

La solución planteada para los requisitos del sistema es la realización de una aplicación basada en tres módulos relacionados.

[[Insertar imagen – Diagrama general del sistema]]

Los módulos comparten una única base de datos de forma similar a estilo arquitectónico de **base de datos compartida**, obteniendo de esta manera algunas de las ventajas de dicho estilo. Sin embargo los módulos de este sistema no son independientes lo que desvirtúa el patrón.

## Módulo 1 – Extract

Este módulo es el inicial del sistema. Desde este módulo se añaden preguntas a la base de datos desde ficheros en formato GIFT. Es decir, es un parser que lee ficheros y actualiza la base de datos. También se puede usar para escribir ficheros intermedios para el sistema.

El diseño de software interno consiste en la separación de responsabilidades. Independientemente de su diseño interno, este módulo tiene el comportamiento de una aplicación secuencial o batch. Por las características de este módulo y las necesidades que debe cubrir este comportamiento es el óptimo.

## Módulo 2 – Core

## Módulo 3 – Play Framework

# Lista de atributos de calidad e interesados

# Descripción del negocio de la solución

# Vistas

## Vistas de sistema

[[Diagrama del sistema]]

## Diagrama de componentes

[[Diagrama de componentes]]

## Diagrama de paquetes

[[Diagrama de paquetes general]]

{{Sobre el BPMN}}