

Desarrollo de Aplicaciones Distribuidas: Registrador de juegos

Rafael Gálvez-Cañero, Andreas Gerstmayr

Iteración 3 - 9 de Abril de 2015

Índice general

1. Datos generales	2
1.1. Miembros del grupo	2
1.2. Descripción del sistema	2
1.2.1. Funcionalidad observable	2
1.2.2. Servicios ofrecidos	2
1.2.3. Servicios demandados	3
1.3. Direcciones de descarga y planificación	3
1.4. Seguimiento	3
2. Modelado	4
2.1. Análisis del sistema	4
2.2. Arquitectura del sistema	4
3. Iteración 3	5
3.1. Objetivos de iteración	5
4. Iteración 4	6
4.1. Objetivos de iteración	6
4.1.1. Distinción de dominio, controlador y servicio	6
5. Iteracion 5	7
5.1. Objetivos de iteración	7
6. Iteracion 6	9
6.1. Objetivos de iteración	9
7. Iteracion 7	10
7.1. Objetivos de iteración	10

Índice de figuras

2.1. Modelo de despliegue del sistema	4
---	---

Índice de cuadros

1.1. Miembros del grupo	2
1.2. Datos generales del trabajo en grupo	3
1.3. Tabla de seguimiento	3

Capítulo 1

Datos generales

1.1. Miembros del grupo

Apellidos	Nombre	Correo-e	Grupo
Gálvez-Cañero	Rafael	galvesband@gmail.com	18
Gerstmayr	Andreas	andreas.gerstmayr@gmail.com	18

Cuadro 1.1: Miembros del grupo

1.2. Descripción del sistema

- **Tipo de sistema distribuido:** Sistema de información.
- **Nombre del proyecto:** Plataforma de juegos, Game Registry.
- **Breve descripción:** Sub-sistema para registrar sesiones de juego e información asociada.

1.2.1. Funcionalidad observable

- Registrar el inicio y el término de todas las sesiones de juego.
- Visualizar el historial de juegos.

1.2.2. Servicios ofrecidos

- Servicio de Registro: Capacidad de aceptar la información de una sesión de juego (juego ID, jugador ID y fecha de inicio y término).
- Servicio de Historial: Ofrece métodos para consultar el historial de sesiones.

1.2.3. Servicios demandados

- Servicio de Juego: avisar de término de una sesión de juego
- Servicio de Juego: recibir el título de un juego
- Servicio de Perfil: recibir el nombre de un jugador

1.3. Direcciones de descarga y planificación

Código fuente	https://repositorio.informatica.us.es/svn/lq3vqrtzfnh2nx9yhp
Planificación temporal	
Iteración 1	24/02/2015
Iteración 2	03/03/2015
Iteración 3	26/03/2015
Iteración 4	7/04/2015
Iteración 5	28/04/2015
Iteración 6	12/05/2015
Iteración 7	26/05/2015
Entrega Final	02/06/2015

Cuadro 1.2: Datos generales del trabajo en grupo

1.4. Seguimiento

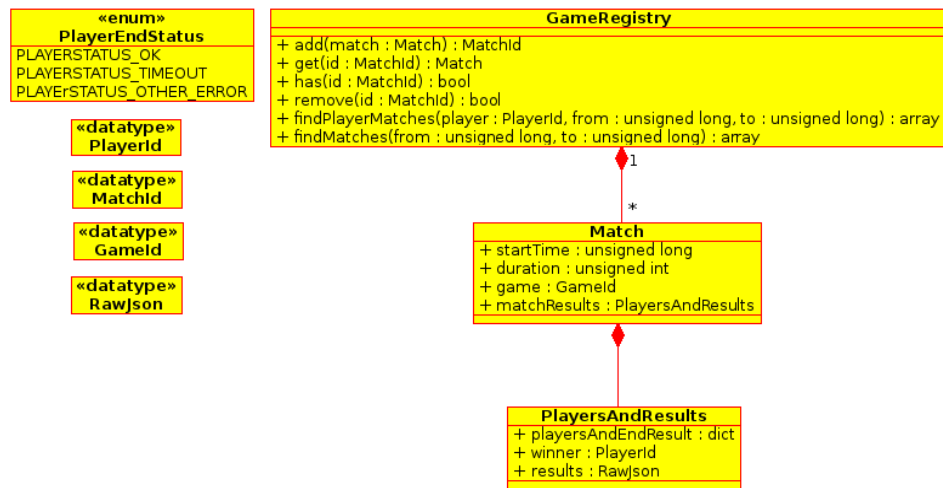
Estudiante	Iteración								Total	Pond.
	1	2	3	4	5	6	7	Final		
Rafael Gálvez-Cañero	5	5	5	5	-	-	-	-	10	1
Andreas Gerstmayr	5	5	5	5	-	-	-	-	10	1
Total	10	10	0	0	0	0	0			

Cuadro 1.3: Tabla de seguimiento

Capítulo 2

Modelado

2.1. Análisis del sistema



2.2. Arquitectura del sistema

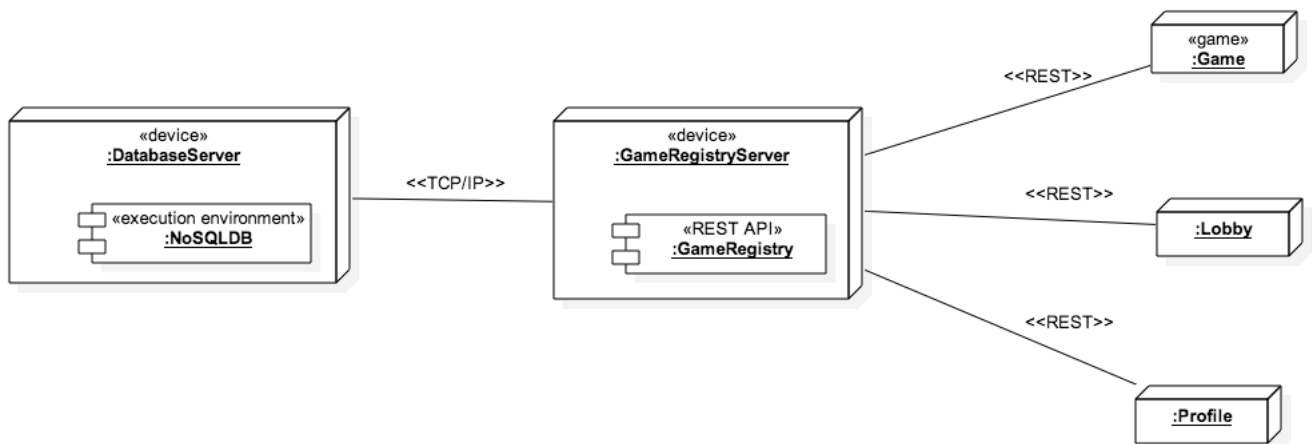


Figura 2.1: Modelo de despliegue del sistema

Capítulo 3

Iteración 3

3.1. Objetivos de iteración

- Integración de Gradle
- Servidor dockerizado
- Estructura inicial del Cliente vertex.

Capítulo 4

Iteración 4

4.1. Objetivos de iteración

- Integración de MondoDB (como contenedor docker)
- Cliente más avanzado. Servidor estructurado en Servicio / Controlador.

4.1.1. Distinción de dominio, controlador y servicio

Dominio

POJO clases con la misma esquema de la base de datos.

Controlador

Manejar los requisitos y respuestas y comunicar con el servicio.

Servicio

La lógica de negocio. Almacenar los sesiones en el base de datos.

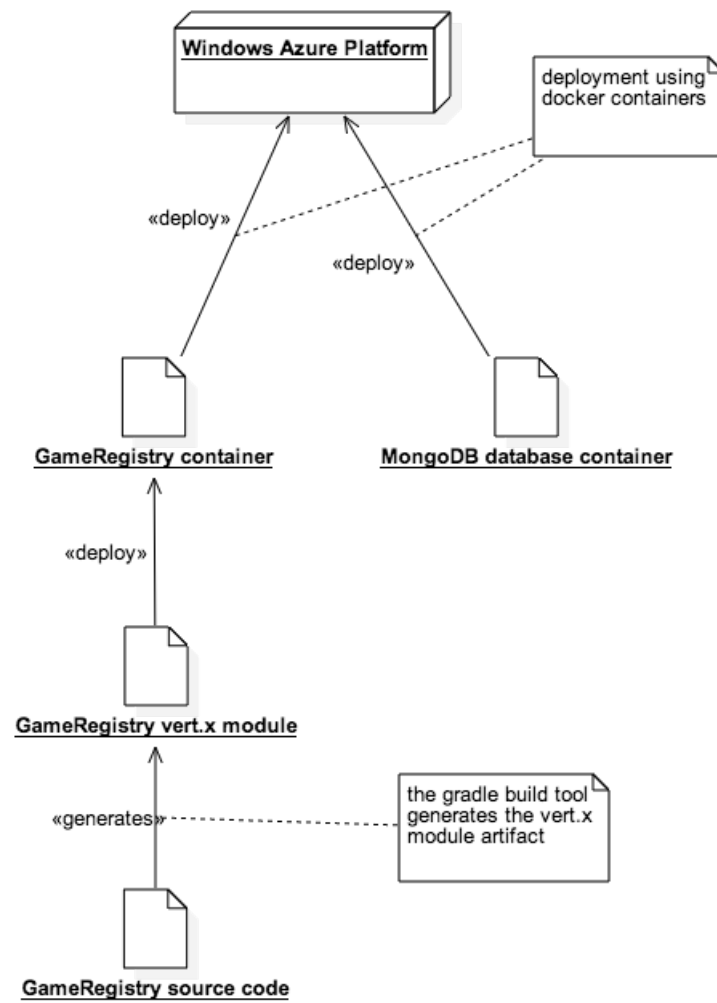
Capítulo 5

Iteracion 5

5.1. Objetivos de iteración

- Implementación inicial de API
- Primeros tests.
- Despliegue Azure plataforma
- ¿Integración continua?

5.2. Diagrama de despliegue



Capítulo 6

Iteracion 6

6.1. Objetivos de iteración

Final testing.

Capítulo 7

Iteracion 7

7.1. Objetivos de iteración

Subir a repositorio Maven.