



EventPlanner

Gestión de Eventos

Xabier Dendarieta
Sonia Elizondo
Paul Vega



Objetivo

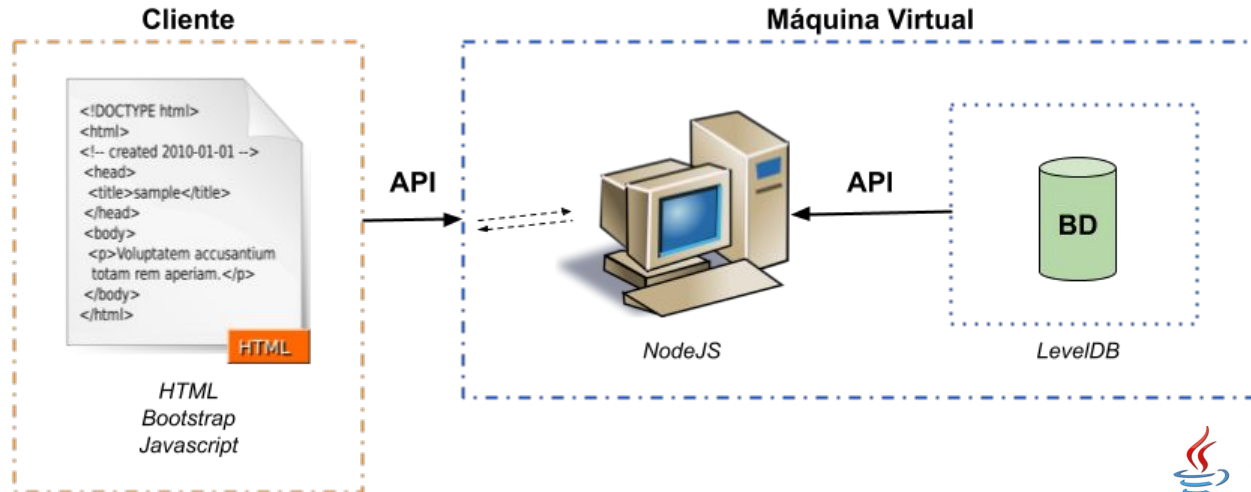
- Aplicación web **EventPlanner**
 - Gestión de asistencia a eventos sociales:
 - Comidas, celebraciones, reuniones familiares, ocio

EventPlanner

The solution you need to make events

Arquitectura del Software

- **3 capas:** Cliente Web \longleftrightarrow API \longleftrightarrow Base de Datos
 - Cliente web: **Bootstrap** junto a HTML/CSS y Javascript
 - API: **NodeJS**
 - Base de Datos: **LevelDB**





Decisiones Técnicas (ADR)

- Base de Datos:
 - **LevelDB** frente a ElasticSearch
 - Librería de Node.js
 - Uso sencillo con almacenamiento clave-valor
 - Equipo familiarizado con su uso



Decisiones Técnicas (ADR)

- Gestión de la Configuración:
 - **GitHub** frente a repositorios similares (Bitbucket)
 - Funcionalidades *Pro*: Paquete “Education”
 - Equipo tiene cuentas registradas
 - Integración con Jenkins



Decisiones Técnicas (ADR)

- Integración Continua:
 - **Jenkins**, frente a herramientas como Travis CI, Bamboo, GitLab CI
 - Herramienta gratuita
 - Compatible con GitHub
 - Posibilidad de tests de integración
 - Despliegue con Docker



Problemas Encontrados

- Instalación de librerías de Node.js
 - Uso de comandos
 - Sustitución de *zmq* por *zeromq*
- Ejecución automática de *pipeline* en Jenkins
 - Sin solución
- Despliegue en Docker
 - *Jenkinsfile* modificado para utilizar contenedor con Node.js
 - No se ejecuta dentro del contenedor, pero *pipeline* acaba en *SUCCESS*
- Funcionamiento de API
 - Mayor tiempo de desarrollo que el esperado
 - Excepciones eventuales al realizar peticiones a la base de datos
- Conexión Cliente-API
 - Bloqueo de petición por cross-domain



Funcionalidad

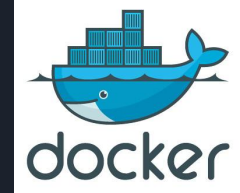
- **Registrarse / Identificarse** en el sistema
- **Crear** evento
- Visualizar **lista de eventos** actual
- Visualizar **información de un evento** en concreto
- **Asistir / dejar de asistir** a un evento
- Si el usuario es **organizador** de un evento:
 - **Eliminar** dicho evento
 - **Borrar asistentes** no deseados

Integración Continua



Pipeline en Jenkins:

1. Contenedor Docker: imagen de Node.js
2. Clonar repositorio GitHub
3. Instalar dependencias *npm* dispuestas en *package.json*
4. Test unitario a Base de Datos



Metodología

- Metodología personalizada: Ágil + Kanban
- Se intenta:
 - Reforzar la comunicación en todo momento
 - Reunión bisemanal fija
 - Reunión extra cuando los desarrolladores lo crean conveniente
- **Discord**
 - Herramienta de comunicación interna
 - Chat de texto general
 - Chat de texto para notificaciones
 - Chat de texto para planificación de reuniones
 - Chat de voz (posibilidad de vídeo) para reuniones
- **Trello**
 - Herramienta de tablón de tareas (Kanban)
 - Integración con **Clockify**: registro de tiempo



DISCORD

