



ugr | Universidad
de **Granada**

TRABAJO FIN DE GRADO
GRADO EN INGENIERÍA INFORMÁTICA

Proyecto SinTime

Aplicación Web Gamificada de Docencia

Autor
Álvaro Fernández-Alonso Araluce

Directores
Juan Manuel Fernández Luna
Isaac José Pérez López



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍAS INFORMÁTICA Y DE
TELECOMUNICACIÓN

—
Granada, Junio de 2018



Proyecto SinTime

Aplicación Web de Docencia Gamificada

Autor

Álvaro Fernández-Alonso Araluce

Directores

Juan Manuel Fernández Luna
Isaac José Pérez López

Proyecto SinTime: Aplicación Web Gamificada de Docencia

Álvaro Fernández-Alonso Araluce

Palabras clave: SinTime, #SinTime_wARs, docencia, gamificación

Resumen

En este documento se desarrollará una idea dirigida a incrementar la atención, participación, dedicación y esfuerzo por parte del alumnado en sus obligaciones como estudiante.

Las actividades lúdicas provocan que focalicemos toda nuestra atención en superar retos de forma constante provocando incluso, por medio de historias; escenarios y experiencias, la inmersión total de la persona que las realice. La idea consiste en extraer todo ese mecanismo e implantarlo en la educación haciendo que los retos a superar sean retos educativos.

En este trabajo se presenta un escenario inmersivo en el que científicos han conseguido detener el gen de crecimiento. Dado que ya nadie fallece por causas naturales se implanta en cada persona un reloj regresivo con 10 años de vida. A partir de ese momento el trabajo se cobra en tiempo, los alimentos cuestan tiempo, toda la economía se basa en el tiempo. Si un reloj llega a cero provocará un infarto instantáneo.

El escenario anteriormente descrito es una breve sinopsis de la película **InTime** del director neozelandés **Andrew Niccol** y es en esta trama donde van a verse inmersos **los estudiantes de Fundamentos de la Educación Física** de los cursos 2016/2017 & 2017/2018 por medio de una **Aplicación Web** que va a simular este escenario.

En este caso cada estudiante (ciudadano de SinTime a partir de ahora) va a comenzar su experiencia con 15 días de vida en distritos independientes, cada segundo que pasa es un segundo menos que tienen y tendrán que realizar diversos retos para obtener más tiempo.

\$inTime Project: Gamified Web Application for Teaching

Álvaro Fernández-Alonso Araluce

Keywords: SinTime, #SinTime_wARs, Gamification, Teaching

Abstract

This document will develop an idea aimed at increasing attention, participation, dedication and effort on the part of students in their obligations as a student.

Playful activities cause us to focus all our attention on constantly overcoming challenges, even provoking them through stories; scenarios and experiences, the total immersion of the person who makes them. The idea is to extract all this mechanism and implement it in education, making the challenges to be overcome educational challenges.

In this work an immersive scenario is presented in which scientists have managed to stop the growth gene. Since no one dies due to natural causes, a regressive clock with 10 years of life is implanted in each person. From that moment work is charged in time, food costs time, the whole economy is based on time. If a clock reaches zero it will cause an instantaneous infarction.

The scenario described above is a brief synopsis of the film **InTime** by the New Zealand director **Andrew Niccol** and it is in this plot where the **students of Fundamentals of Physical Education will be immersed** 2016/2017 & 2017/2018 through a **Web Application** that will simulate this scenario.

In this case each student (citizen of SinTime from now on) will start their experience with 15 days of life in independent districts, every second that passes is a second less than they have and they will have to perform various challenges to get more time.

Yo, **Álvaro Fernández-Alonso Araluce**, alumno de la titulación **TITULACIÓN de la Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación de la Universidad de Granada**, con DNI 75167394J, autorizo la ubicación de la siguiente copia de mi Trabajo Fin de Grado en la biblioteca del centro para que pueda ser consultada por las personas que lo deseen.

Fdo: Álvaro Fernández-Alonso Araluce (alumno)

Granada a Mayo de mes de 2017.

D. **Juan Manuel Fernández Luna**, Profesor del Área de XXXX del Departamento YYYY de la Universidad de Granada.

D. **Isaac José Pérez López**, Profesor del Área de XXXX del Departamento YYYY de la Universidad de Granada.

Informan:

Que el presente trabajo, titulado ***SinTime, Aplicación Web de Docencia Gamificada***, ha sido realizado bajo su supervisión por **Álvaro Fernández-Alonso Araluce**, y autorizamos la defensa de dicho trabajo ante el tribunal que corresponda.

Y para que conste, expiden y firman el presente informe en Granada a X de Septiembre de 2018.

Los directores:

Juan Manuel Fernández Luna Isaac José Pérez López

Agradecimientos

A **Soraya** por su apoyo incondicional.

A **Juan Manuel Fernández Luna** por su paciencia y consejos.

A **Isaac José Pérez López** por contar conmigo para desarrollar una idea tan original y a **sus alumnos de Fundamentos de la Educación Física** por la experiencia vivida.

Índice general

Introducción	IX
1.1. Motivación[6, 9, 7, 8, 10, 11, 12, 13, 14]	IX
1.2. Diccionario de acrónimos	X
1.3. La economía en SinTime	X
1.4. Los retos en SinTime	XI
1.5. Objetivos de un padawan SinTime	XI
1.5.1. Seguimiento de maestro Jedi	XI
1.5.2. Pruebas de nivel	XII
1.5.3. Dominio de la fuerza	XII
1.5.4. Víveres	XII
1.5.5. Audiencia ante el Senado Galáctico	XII
1.5.6. Viajes Interplanetarios	XIII
1.5.7. Retos de Felicidad	XIII
1.6. Secciones	XIV
1.6.1. Organización de secciones [Rol=Guidoogway]	XIV
1.6.2. Funciones de cada sección [Rol=Guidoogway]	XV
1.6.3. Organización de secciones [Rol=Padawan]	XIX
1.6.4. Funciones de cada sección [Rol=Ciudadano]	XX
Planificación	XXIII
2.1. Trello	XXIII
2.2. Sprints	XXIV
2.3. Técnica Pomodoro (Productividad) [1]	XXIV
2.3.1. Los objetivos	XXV
2.3.2. Los principios	XXV
Presupuestos	XXVII
Plan de entregas	XXIX
4.1. Breve descripción del alcance del sistema	XXIX
4.2. Listado inicial de Historias de Usuario	XXIX
4.3. Cálculo de la velocidad del equipo	XXX
4.4. Descripción de las entregas	XXX

4.5. Lista Inicial del Producto	XXXI
4.6. Plan de entrega 0	XXXI
4.6.1. Objetivos de la entrega	XXXI
4.6.2. Listado de HU a desarrollar	XXXI
4.6.3. Descomposición en tareas de desarrollo	XXXII
4.6.4. Carga prevista en los desarrolladores	XXXIII
4.6.5. Planificación temporal de las tareas	XXXIV
4.7. Plan de entrega 1	XXXV
4.7.1. Objetivos de la entrega	XXXV
4.7.2. Listado de HU a desarrollar	XXXV
4.7.3. Descomposición en tareas de desarrollo	XXXV
4.7.4. Carga prevista en los desarrolladores	XXXVII
4.7.5. Planificación temporal de las tareas	XXXVII
4.8. Plan de entrega 2	XXXVIII
4.8.1. Objetivos de la entrega	XXXVIII
4.8.2. Listado de HU a desarrollar	XXXVIII
4.8.3. Descomposición en tareas de desarrollo	XXXVIII
4.8.4. Carga prevista en los desarrolladores	XL
4.8.5. Planificación temporal de las tareas	XL
4.9. Plan de entrega 3	XLI
4.9.1. Objetivos de la entrega	XLI
4.9.2. Listado de HU a desarrollar	XLI
4.9.3. Descomposición en tareas de desarrollo	XLI
4.9.4. Carga prevista en los desarrolladores	XLIV
4.9.5. Planificación temporal de las tareas	XLIV
Tecnologías usadas	XLVII
5.1. Google Analytics	XLVIII
5.2. Rollbar	XLIX
5.3. Port Monitor [2]	LI
5.4. Runtastic	LI
5.5. Twitter	LII
Diseño	LIII
6.1. Diagrama de clases	LIII
Análisis	LV
7.1. Historias de usuario	LV
7.2. Tarjetas de Servidor de Producción	LVI
7.2.1. Instalación de Ruby on Rails	LVI
7.2.2. Configuración de la Base de Datos	LVI
7.2.3. Configuración de gitignore	LVII
7.2.4. Sistema de alertas Rollbar	LVII
7.2.5. Desplegar cambios locales a producción	LVIII

7.2.6. Sistema de alertas Port Monitor	LIX
7.3. Sistema	LX
7.3.1. Configuración de namespaces	LX
Prensa	LXI
8.1. Prensa digital	LXI
8.2. Radio	LXII
8.3. Prensa escrita	LXIII
8.3.1. 27/06/2017 - Granada Hoy	LXIII
8.3.2. 28/06/2017 - Ideal	LXIV
8.3.3. 27/07/2017 - Ideal	LXV

Índice de figuras

1.1. Objetivo 1 - Cuenta regresiva	XX
1.2. Objetivo 2 - Ejemplos de avisos	XX
1.3. Objetivo 2 - Ejemplos de avisos	XXI
2.4. Trello	XXIV
5.5. Google Analytics	XLVIII
5.6. Rollbar - Notificación email	XLIX
5.7. Rollbar - Rollbar detalle Traceback	L
5.8. Rollbar - Rollbar detalle Ocurrences	L
5.9. Rollbar - Rollbar detalle Browsers/OS	L
5.10. Twitter - Ejemplo	LII
8.11. Prensa 27/06/2017 - Granada Hoy	LXIII
8.12. Prensa 28/06/2017 - Ideal	LXIV
8.13. Prensa 27/07/2017 - Ideal	LXV

Índice de cuadros

1.1. Sistema de alertas	XXI
7.2. Historias de usuario	LV
7.3. Historia de usuario - Instalación de Ruby on Rails	LVI
7.4. Historia de usuario - Configuración de la Base de Datos	LVI
7.5. Historia de usuario - Configuración de gitignore	LVII
7.6. Historia de usuario - Sistema de alertas (Rollbar)	LVII
7.7. Historia de usuario - Desplegar cambios locales a producción .	LVIII
7.8. Historia de usuario - Sistema de alertas Port Monitor	LIX
7.9. Historia de usuario - Configuración de namespaces	LX

Introducción

El objetivo de este proyecto es favorecer la participación del alumnado tanto en las sesiones presenciales como en casa. Para ello se ha desarrollado una aplicación Web gamificada en Ruby on Rails a modo de plataforma de docencia donde alumnos y profesores interactúan en todo momento.

La aplicación va a simular un escenario donde los alumnos tendrán en cada una de las pantalla un reloj regresivo, todos los alumnos comenzarán con el mismo tiempo de vida (TdV en adelante) y no llegar a cero será el objetivo principal de cada uno de ellos. El TdV, por tanto, se convertirá en una moneda sobre la que emergerá toda la economía de la aplicación, si además añadimos que cada ejercicio realizado reportará una cantidad de TdV sobre el contador del que lo realice en función de la resolución de éste tenemos como resultado una competición entre alumnos que provocará que ellos mismos pidan realizar todos los ejercicios/entregas posibles. Las formas de obtener TdV se detallarán más adelante en la sección **Objetivos de un padawan Sintime**.

1.1. Motivación[6, 9, 7, 8, 10, 11, 12, 13, 14]

Uno de los principales problemas que sufre la Universidad en la actualidad es la falta de motivación y compromiso de los estudiantes a la hora de participar activamente en su aprendizaje. Esta falta de interés y motivación puede estar determinada, en gran medida, por el rol pasivo que desempeñan los estudiantes en las metodologías de enseñanza tradicionales (Martí-Parreño, 2015), especialmente las generaciones más jóvenes, los nativos digitales, esto es, estudiantes que hacen un uso intensivo de la tecnología y de la interactividad digital (Prensky, 2001). Por tanto, la necesidad de mejorar el atractivo de la docencia y adoptar nuevas metodologías de enseñanza-aprendizaje activas, que favorezcan la motivación e implicación del alumnado, se ha convertido en todo un reto para la Universidad que su profesorado deberá afrontar cuanto antes (Contreras y Eguia, 2016; De Jorge et al., 2011; Kiryakova, Angelova, y Yordanova, 2014; Martínez González, 2011).

En este sentido, una poderosa estrategia para motivar y favorecer el aprendizaje del alumnado es la gamificación, entendida como el uso de elementos de diseño del juego en contextos no lúdicos con la finalidad de motivar a los participantes (Deterding et al., 2011). Para ello incorpora elementos de los videojuegos como el contexto, los desafíos o las recompensas, aumentando la interacción del alumnado con el entorno de aprendizaje y acabar modificando su conducta (González González y Mora, 2015; Prieto et al., 2014).

La gamificación nos ofrece ciertas ventajas frente a las metodologías de enseñanza tradicionales:

- Motivación del alumnado.
- Desarrollo de habilidades mediante el crecimiento gradual de la dificultad para realizar una tarea.
- Fomenta la competencia y ofrece un reconocimiento (Rankings).
- Fomenta la conexión social, ya que los estudiantes viven juntos una experiencia estimulante.

1.2. Diccionario de acrónimos

- Tiempo de Vida (TdV)
- Profesor en el papel de “Guardián del Tiempo” (Maestro Guidoogway)
- Alumno en el papel de “Ciudadano SinTime” (Padawan)
- Sistema en el papel de “Metronomista” (Metronomista)
- Ejercicio (Reto)
- Calificación (Badge)

1.3. La economía en SinTime

Anteriormente se ha hablado de forma general sobre la economía en SinTime. Podría decirse que ésta es la pieza central de la plataforma. Para gestionar esta moneda, cada alumno (Padawan en adelante) tendrá una “cuenta corriente” en la que se verán reflejadas todas sus operaciones bancarias. Para cada operación tendrán la información de cuándo se ha realizado, el motivo del ingreso/gasto y la cantidad de TdV implicada.

1.4. Los retos en SinTime

Los retos en SinTime se podrían reducir a ejercicios/entregas en su definición más simple aunque se le da un significado totalmente diferente. Son retos porque se desafía al alumno a realizarlos, y se le da la oportunidad de ir mejorándolo hasta alcanzar la máxima “calificación”.

Como en un juego, al realizar una entrega de un reto el alumno consigue un score/puntuación y, por medio de la reentrega conseguimos que el alumno quiera superarse, reintentarlo hasta conseguir el máximo feedback posible. Este feedback

La relación entre un reto y un alumno tendrá uno de los siguiente estados:

- **Comprado:** Un alumno que ha comprado ese reto con su TDV
- **Entregado:** Despues de la compra podrá entregar el reto.
- **Calificado:** El docente ha calificado la entrega y asignado un badge.

1.5. Objetivos de un padawan SinTime

El objetivo principal de un padawan SinTime será obtener el máximo TdV que pueda para no caer en el lado oscuro y obtener así una buena calificación en su nota final. Son muchas y variadas las formas por las que podrán obtener/perder TdV. Algunas de ellas son:

- Seguimiento de maestro Jedi
- Pruebas de nivel
- Dominio de la fuerza
- Víveres
- Audiencia ante el Senado Galáctico
- Viajes Interplanetarios
- Felicidad

1.5.1. Seguimiento de maestro Jedi

Para este tipo de reto se ha desarrollado una integración de la plataforma con Twitter. El reto consiste en guardar un mínimo de Tweets diarios que tengan cierta relevancia en el mundo de la Educación Física.

Si al final del día el alumno ha guardado un número mínimo de Tweets (especificado por el docente), se le ingresará en su cuenta bancaria un TDV (especificado por el docente).

1.5.2. Pruebas de nivel

Consiste en un **reto gratuito** compuesto por un enunciado en el que el alumno realizará una entrega de un fichero con una presentación que desarrollará en clase frente a los compañeros. La presentación que hará estará marcada por el enunciado del reto.

El profesor marcará un número máximo de alumnos que puedan “pedirse” el reto, sólo podrán realizarlo los X alumnos primeros en marcar el reto como "Pedido"(comprado internamente).

1.5.3. Dominio de la fuerza

El docente podrá lanzar a sus alumnos retos sorpresas, este tipo de reto resta 1 día de vida a todos los alumnos al ser lanzado recuperable únicamente si contestan correctamente al cuestionario.

1.5.4. Víveres

Para hacer la experiencia más inmersiva creamos a los alumnos la necesidad de alimentarse e hidratarse. Para ello lanzamos dos tipos de retos; **Comida** y **Agua**. La necesidad de realizar estos tipos de retos surge al crear una barra de energía para la sección de comida y otra barra de energía para la sección de agua. Estas barras se irán vaciando conforme pasen días sin entregas realizadas.

Estos retos son comprados por el alumno (tienen un coste en tiempo), al ser corregidos por el docente se le realizará un ingreso correspondiente al badge recibido.

Además, estos retos podrán ser grupales (realizados por todo el clan) o individuales. Añadir que para la sección de Agua, los retos individuales no podrán ser elegidos directamente, el alumno solicitará un reto individual y el sistema le asignará un reto aleatoriamente.

1.5.5. Audiencia ante el Senado Galáctico

Este reto se lanza la última semana de la asignatura. En esta sección, el alumno realizará la entrega de su proyecto final, en el que reflejará todo lo aprendido durante toda su experiencia.

1.5.6. Viajes Interplanetarios

Es un tipo de reto deportivo. Durante cada semana se especifican unos requisitos para cubrir durante la semana bien corriendo o bien en bicicleta.

En este tipo de retos el docente especifica una velocidad media y duración mínimas (en caso de bicicleta) y un ritmo medio y duración mínimas (en caso de running). Los alumnos por su lado intentarán cubrir esa duración durante la semana en la modalidad que prefieran y en diferentes sesiones (mínimo 3 sesiones distintas). Si el alumno cumple los requisitos de la semana en la modalidad seleccionada se le hará un ingreso automático al final de la semana determinado por el docente.

1.5.7. Retos de Felicidad

Estos retos son más una labor humanitaria por parte de los alumnos, es un reto totalmente opcional sin coste alguno en el que un alumno propone una labor social a realizar durante 3 meses en una primera entrega. La segunda entrega que realice será una evidencia de la labor realizada.

Este reto tendrá como respuesta un badge con el ingreso de tiempo correspondiente al mismo.

1.6. Secciones

1.6.1. Organización de secciones [Rol=Guidoogway]

```
manager/home[Rol=Guidoogway]
├── Dashboard
│   ├── Distritos
│   ├── Padawans
│   │   ├── Entregas
│   │   ├── Movimientos bancarios
│   │   ├── Subir de nivel
│   │   └── Datos deportivos
│   └── Vacaciones
├── Ejercicios (Retos)
│   ├── Entregas
│   ├── Cuestionarios (Dominio de la fuerza)
│   ├── Comida
│   ├── Agua
│   ├── Retos deportivos
│   ├── Retos de felicidad
│   ├── Pruebas de nivel
│   └── Pruebas de audiencia
└── General
    ├── Calificaciones (Badges)
    ├── Constantes (Constantes de la app)
    ├── Rangos
    ├── Crear Padawans (Por lotes de DNI)
    ├── Chat
    ├── Cartas de privilegio
    ├── Apuestas
    └── Minas
```

1.6.2. Funciones de cada sección [Rol=Guidoogway]

Dashboard

Esta vista mostrará un resumen de las demás secciones, esto es; Usuarios, Retos, Información General. Hace la función de acceso directo para las secciones que se nos muestran en el sidebar.

Distritos

En esta sección podremos crear nuevos distritos además de editarlos, eliminarlos y mostrar información relevante. Además podremos ingresar TDV a todos los miembros del distrito/clan.

Padawans En esta sección podremos crear nuevos distritos además de editarlos, eliminarlos y mostrar información relevante.

Como funciones extra podemos destacar:

- Visualización de todas las entregas que ha realizado el usuario
- Visualización de todos los movimientos bancarios relativos al usuario
- Acción de subir de nivel de forma manual si el docente lo considera oportuno
- Visualización de todos los resultados deportivos del usuario.

Estas secciones anteriormente mencionadas y explicadas a continuación ofrecen información relevante de un usuario al docente cuando se encuentre con él en tutorías.

Entregas del usuario En esta sección, dentro del directorio del alumno podremos visualizar todas las entregas de todos los tipos de retos que ha realizado el alumno. Como información relevante se mostrará:

- El estado en el que está su entrega (Comprado, Entregado, Corregido)
- La visualización del fichero entregado (si el estado es Entregado o Corregido)
- Información sobre el reto (Enunciado)
- Información sobre el tipo de reto (Comida, Agua, Proyecto de Felicidad, Proyecto de innovación, Prueba de nivel, Audiencia ante el Senado Galáctico)

Movimientos bancarios de un usuario Esta sección es un show de los movimientos bancarios que tiene un usuario (Cobros e Ingresos).

Subir de nivel a un usuario Para que un usuario suba de nivel debe cumplir dos requisitos, haber superado el reto deportivo de la semana y

tener un badge mínimo especificado por el docente durante esa semana. Cuando esas dos condiciones se cumplen, el usuario sube de nivel e ingresa una cantidad de XP correspondiente al nivel que tenga. Esta tarea se ejecuta automáticamente al término de cada semana pero el profesor se reserva la opción de poder subir de nivel de forma manual si lo considera oportuno.

Datos deportivos de un usuario Esta sección muestra un resumen semanal de los resultados deportivos de un usuario en particular. Los datos que se muestran son, por semana, la fase del reto deportivo que se le aplica y si ha superado o no dicha fase.

Vacaciones

Esta sección ofrece la posibilidad al docente de hacer un parón de la app en períodos no lectivos. Al asignar un periodo de vacaciones indica a la app que el TDV y las barras de energía no disminuirán durante el periodo especificado.

Entregas

Esta sección nos muestra todas las entregas en estado “Entregado”. De esta forma el docente puede ir calificando esas entregas en una sola sección.

Cuestionarios

En esta sección podremos crear nuevos cuestionarios además de editarlos, eliminarlos y mostrar información relevante.

Al crear un cuestionario se le cobrará 1 día de vida a cada alumno recuperables si éste responde correctamente al mismo.

Comida

En esta sección podremos crear nuevos retos de comida además de editarlos, eliminarlos y mostrar información relevante.

Agua

En esta sección podremos crear nuevos retos de agua además de editarlos, eliminarlos y mostrar información relevante.

Retos deportivos

En esta sección podremos crear nuevas fases de retos deportivos además de editarlos, eliminarlos y mostrar información relevante.

Retos de felicidad

En esta sección podremos crear nuevos retos de felicidad además de editarlos, eliminarlos y mostrar información relevante.

Pruebas de nivel

En esta sección podremos crear nuevas pruebas de nivel además de editarlos, eliminarlos y mostrar información relevante.

Pruebas de audiencia

En esta sección podremos crear nuevas pruebas de audiencia además de editarlos, eliminarlos y mostrar información relevante.

Calificaciones (Badges)

En esta sección podremos crear nuevos badges además de editarlos, eliminarlos y mostrar información relevante.

Constantes

En esta sección podremos crear nuevas constantes además de editarlos, eliminarlos y mostrar información relevante.

Estas constantes nos sirven para modificar el comportamiento de la app, en esta sección podremos especificar:

- el tipo de interés que tienen los préstamos
- el número de días que debe transcurrir entre la entrega de un proyecto de felicidad y la entrega de las evidencias.
- la recompensa en segundos que se obtendrá al superar una fase deportiva.
- la recompensa en segundos por almacenar X Tweets diarios.

- el número mínimo de Tweets diarios que debe almacenar cada usuario.
- la velocidad a la que disminuye la barra de energía de comida.
- la velocidad a la que disminuye la barra de energía de agua.

Rangos

En esta sección el docente puede definir los rangos que van a existir en la app y la puntuación mínima en el ranking para conseguir ese distintivo.

Crear Padawans

En esta sección el docente puede dar de alta usuarios por lotes mediante una lista de DNIs.

Cartas de privilegios

En esta sección el docente puede ocultar o mostrar una carta de privilegios previamente implementada. Además de modificar el coste, imagen de la misma.

Apuestas

En esta sección el docente puede crear apuestas deportivas ante cualquier torneo deportivo entre clanes. Una vez publicado la apuesta los usuarios podrán apostar TDV en las diferentes opciones.

El docente puede cerrar la apuesta para que los alumnos no puedan apostar más. El siguiente paso es marcar la opción correcta y monetizar a los alumnos que hayan acertado.

Minas

En esta sección el docente puede crear una mina, especificar un código secreto que deben encontrar los alumnos mediante pistas que agrega el docente a la mina. Las pistas no son visibles a los alumnos a menos que las compren con cartas de privilegios.

1.6.3. Organización de secciones [Rol=Padawan]

```
padawan/home[Rol=Padawan]
├── Padawan
│   ├── Hoja de personaje (Datos)
│   ├── Rango Jedi (Clasificaciones)
│   ├── Nivel - Cartas de privilegio (Cartas de privilegio)
│   └── Historial de aprendizaje (Movimientos bancarios)
├── Formación
│   ├── Seguimiento de maestro Jedi (Twitter)
│   ├── Pruebas de nivel (Pruebas de superación)
│   ├── Dominio de la fuerza (Cuestionarios sorpresa)
│   └── Audiencia ante el Senado Galáctico (Proyectos de innovación)
├── Viveros
│   ├── Comida (Retos individuales o por clanes)
│   └── Agua (Retos individuales o por clanes)
├── Comunidad de aprendizaje
│   ├── RCI: Red de Comunicación Intergaláctica (Amigos)
│   │   ├── Hologramas (Chat)
│   │   └── Historial (Galería de fotos)
│   ├── Viajes Interplanetarios (Deporte - Runtastic)
│   ├── Detección de la fuerza (Apuestas deportivas)
│   └── Altruismo
│       ├── Donación (Donación de TDV)
│       └── Minas (Desactivación de minas)
├── Desconexión
│   ├── Cápsula del tiempo (Préstamos)
│   └── Retiro (Vacaciones)
└── Dagobah (Tutorías)
    └── Felicidad (Proyectos de felicidad)
```

1.6.4. Funciones de cada sección [Rol=Ciudadano]**Todas las secciones****Mostrar el TdV**

Figura 1.1: Objetivo 1 - Cuenta regresiva

Es un reloj regresivo que muestra al usuario el TdV que le queda en tiempo real. Como pieza clave del proyecto debe mostrarse en cada una de las secciones que el usuario visualice ya que el padawan tomará sus decisiones en función de éste.

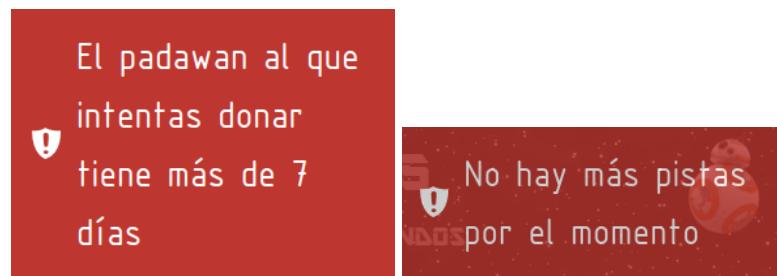
Feedback para cada interacción

Figura 1.2: Objetivo 2 - Ejemplos de avisos

Es importante que el alumno pueda visualizar estos avisos allá donde vaya para que sepa qué está ocurriendo en todo momento.

Mostrar alertas

El sistema de alertas permite avisar a un alumno de que se han producido nuevos eventos. Los eventos comunicados mediante este sistema de alertas son los siguientes:



Avisa de dos nuevas publicaciones de dominio de la fuerza.

Avisa de la publicación de un nuevo reto de prueba de nivel.

Avisa de que una nueva mina está disponible para ser desactivada.

Avisa del número de mensajes de chat pendientes de ser leídos.

Cuadro 1.1: Sistema de alertas



Figura 1.3: Objetivo 2 - Ejemplos de avisos

Planificación

Para este proyecto hemos seguido una metodología de desarrollo XP (eXtreme Programming). En el destacamos simplicidad en el desarrollo, un feedback constante con el usuario (incluso con los usuarios finales) y un desarrollo iterativo e incremental.

2.1. Trello

Para el proceso de planificación hemos decidido usar la herramienta online **trello**, que es un software de administración de proyectos con interfaz web basada en *cartas kanban*. Con esta herramienta dividimos tareas en columnas para crear un flujo de trabajo organizado, la organización de las tareas las dividiremos en las siguientes columnas:

- **Backlog:** En esta categoría agruparemos las tareas o mejoras que no sean urgentes pero estaría bien desarrollar. Estas tareas tienen una prioridad mayormente baja o una fecha de desarrollo estipulada en un futuro próximo.
- **ToDo:** En esta categoría agruparemos las tareas que tengamos pendiente de desarrollo ordenadas por nivel de prioridad de alta a baja.
- **Doing:** En esta categoría agruparemos las tareas que estemos desarrollando. Cada desarrollador tendrá asignada una y solo una tarea bajo esta categoría siempre. Cuando termine esa tarea la pasará a la siguiente categoría y escogerá otra tarea de la categoría **ToDo**. En este caso, como es un desarrollo de una sola persona siempre habrá una sola tarea en esta categoría.
- **QA:** En esta categoría agruparemos las tareas ya terminadas. La función de esta es la de revisar que el nuevo desarrollo se adapta bien al código de la rama master. Si supera los test se hará un PR y se pasará a la siguiente categoría. En caso contrario esta tarea volverá a la categoría **Doing**.

- **Done:** En esta categoría agruparemos las tareas terminadas que hayan pasado los test. Se acepta el Pull Request y se hace un merge con la rama master. Esta rama no se desplegará hasta que llegue la fecha final del Sprint.

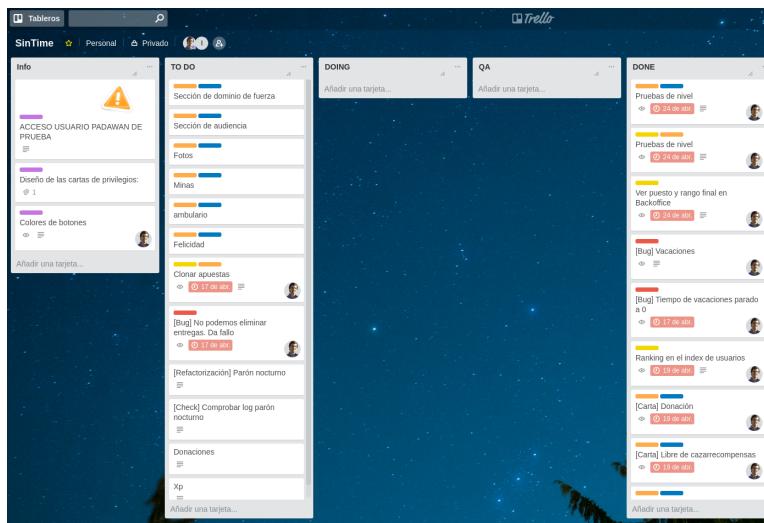


Figura 2.4: Trello

2.2. Sprints

Los sprint son *periodos de desarrollo* pactados con el cliente en el que definimos un paquete de nuevas funcionalidades. En este proyecto vamos a definir el Sprint tipo 2 + 1, esto es, dos semanas de desarrollo más una semana de testing.

2.3. Técnica Pomodoro (Productividad) [1]

En relación a la productividad del desarrollo del proyecto **SinTime**, se ha usado la técnica **Pomodoro** que es un método para gestionar el tiempo. Esta técnica se usa para combatir la ansiedad producida por la cantidad de tareas a desarrollar en un corto periodo de tiempo.

En lugar de trabajar con prisas a fin de conseguir terminar todo el trabajo en la fecha estipulada, esta técnica usa el tiempo como aliado para alcanzar el objetivo de un modo correcto y nos permite mejorar continuamente la manera en que trabajamos.

2.3.1. Los objetivos

La Técnica **Pomodoro** nos permite mejorar la productividad mediante un proceso para conseguir:

- Aliviar la ansiedad a comenzar las tareas.
- Aumentar la concentración disminuyendo las interrupciones.
- Impulsar la motivación y nos mantiene constantes.
- Refina el proceso de estimación.
- Mejora el proceso de trabajo.

2.3.2. Los principios

Cada día se seleccionan las tareas a completar y las colocamos en la lista **ToDo** de la que hemos hablado anteriormente. Podemos usar el cronómetro de nuestro reloj o un timer de **Pomodoro** para, a continuación:

1. Comenzar a trabajar: Iniciamos el timer con una duración de 25 minutos, que es el equivalente a una **sesión Pomodoro**. Una sesión Pomodoro es ininterrumpible por lo que no se debe pausar y reanudar, si se hace una pausa en mitad debe reiniciarse la sesión.
2. Cuando el timer llegue a cero comenzamos un periodo de descanso llamado **Short break** de 3 a 5 minutos, después de este descanso se comienza otra sesión Pomodoro.
3. Cada 4 sesiones Pomodoro completas se realiza un **Long Break**. Un tiempo entre 15-30 minutos.
4. Seguir trabajando hasta que la tarea haya terminado.

Presupuestos

Plan de entregas

4.1. Breve descripción del alcance del sistema

El desarrollo del proyecto **SinTime** consiste en la implementación de una aplicación web cuya implantación se realizará en el marco universitario con el fin de gamificar la experiencia del alumnado mediante la inmersión de ellos en un universo ficticio.

Los objetivos más importantes que debe cumplir la aplicación serán:

- Debe permitir la administración de todos los usuarios con independencia de su rol.
- Debe facilitar al docente la creación de todo tipo de retos para el alumnado.
- Debe facilitar al docente la información de los movimientos de un alumno dentro de la app.
- Debe permitir al alumno ser evaluado por el docente.
- Debe permitir al alumno ser evaluado por el sistema.
- El alumno debe poder obtener su calificación.

4.2. Listado inicial de Historias de Usuario

A continuación se muestran las Historias de Usuario obtenidas durante las reuniones de planificación y entregas del producto realizadas entre el cliente y el equipo de desarrollo. La lista se divide en 4 partes: un identificador, una descripción de la historia, una estimación en días ideales y una prioridad. La prioridad se medirá por el cliente en un rango de 0 a 100, siendo 100 la prioridad más alta.

Identificador	Historias de Usuario	Estimación	Prioridad
HU.1	Ejemplo	1	100

4.3. Cálculo de la velocidad del equipo

El equipo inicial está formado por un programador con una dedicación al proyecto del 50 %. La duración de cada Sprint será, como ya se ha comentado, de 3 semanas de las cuales dos semanas se destinarán a desarrollo y una a testing.

La estimación está basada en *días ideales*. Un día ideal se compone de 4 horas de trabajo por lo que el resultado en horas de cada sprint será:

$$1\text{Sprint} = 3\text{Semanas} \quad y \quad 1\text{Semana} = 5\text{días ideales}$$

$$1\text{día ideal} = 4\text{horas reales} \quad entonces \quad 1\text{Sprint} = 60\text{horas reales}$$

4.4. Descripción de las entregas

Esfuerzo total del proyecto = X PH **Duración del proyecto = X meses**
Velocidad del equipo = X a Y PH

En base a las estimaciones de velocidad del equipo y al esfuerzo necesario para el desarrollo se ha decidido realizar X entregas. La semana laboral dedicada al proyecto se dividirá en 5 días (Lunes a Viernes) con una duración de 4 horas diarias que suman un total de 20 horas semanales. El desarrollo del proyecto comenzará el día **1 de Octubre de 2017**.

El plan de entregas es el siguiente:

Entrega	Objetivo	Fecha de entrega
Sprint 1	Ejemplo largo Ejemplo largo Ejemplo largo Ejemplo largo	20 Octubre 2017
Sprint 2	Ejemplo largo	10 Noviembre 2017

4.5. Lista Inicial del Producto

La lista del producto, con las historias que se usaran en el inicio del desarrollo es la siguiente:

Identificador	Historias de Usuario	Estimación	Prioridad
HU.1	Ejemplo	1	100

4.6. Plan de entrega 0

4.6.1. Objetivos de la entrega

El objetivo principal de esta entrega se basa en la búsqueda y selección de las tecnologías y herramientas usadas para el desarrollo de este proyecto. Además se hará una primera implementación para crear un apartado básico de gestión de usuarios.

4.6.2. Listado de HU a desarrollar

Identificador	Historias de Usuario	Estimación
HU.1	Plataformas y tecnologías usadas	4
HU.2	Como administrador quiero preparar el entorno y las herramientas de uso	11
HU.3	Creación del esqueleto de la aplicación Web	15
HU.4	Como administrador quiero dar de alta a otros usuarios	10

4.6.3. Descomposición en tareas de desarrollo

A continuación incluimos un desglose de las HU en tareas de desarrollo junto con la estimación realizada de su duración.

HU.1	Plataformas y tecnologías usadas	4
Tarea 1.1	Diagrama de clases y Mapa mental	4

HU.2	Como administrador quiero preparar el entorno y las herramientas de uso	11
Tarea 2.1	Configuración de la herramienta y elementos necesarios para comenzar el desarrollo	11

HU.3	Creación del esqueleto de la aplicación Web	15
Tarea 3.1	Desarrollo de un prototipo básico real	15

HU.4	Como administrador quiero dar de alta a otros usuarios	10
Tarea 4.1	Creación en la plataforma Web del usuario tipo Maestro Guidoogway	3
Tarea 4.2	Creación en la plataforma Web del usuario tipo Padawan	3
Tarea 4.3	Creación en la plataforma Web del panel de gestión de usuarios desde el BackOffice	4

4.6.4. Carga prevista en los desarrolladores

A continuación mostramos la carga de trabajo prevista para cada uno de los desarrolladores en relación a las tareas definidas anteriormente:

Desarrollador	Velocidad inicial (días ideales)	Dedicación (% del tiempo)	Carga de trabajo (días ideales)	#Tareas aceptadas
Álvaro Fernández-Alonso Araluce	15	50 %	15	6

4.6.5. Planificación temporal de las tareas

Semana 1	Día 1)	Día 2	Día 3	Día 4	Día 5
Álvaro Fernández- Alonso Araluce	1.1	2.1	2.1	2.1 3.1	3.1

Semana 2	Día 1)	Día 2	Día 3	Día 4	Día 5
Álvaro Fernández- Alonso Araluce	3.1	3.1	3.1 4.1	4.1 4.2	4.3

Semana 3	Día 1)	Día 2	Día 3	Día 4	Día 5
Álvaro Fernández- Alonso Araluce	Testing	Testing	Testing	Testing	Testing

4.7. Plan de entrega 1

4.7.1. Objetivos de la entrega

El objetivo principal de esta entrega consiste en seguir desarrollando funcionalidades para los usuarios padawans y empezar a desarrollar el apartado de gestión de retos desde el BackOffice.

4.7.2. Listado de HU a desarrollar

Identificador	Historias de Usuario	Estimación
HU.5	Permitir al Maestro Guidoogway gestionar clanes entre los usuarios padawans	4
HU.6	Permitir al Maestro Guidoogway gestionar retos	64

4.7.3. Descomposición en tareas de desarrollo

A continuación incluimos un desglose de las HU en tareas de desarrollo junto con la estimación realizada de su duración.

HU.5	Permitir al Maestro Guidoogway gestionar clanes entre los usuarios padawans	4
Tarea 5.1	Permitir al Maestro Guidoogway agrupar usuarios Padawans en un clan	3
Tarea 5.2	Permitir al Maestro Guidoogway asignar un escudo a los clanes	1

HU.6	Permitir al Maestro Guidoogway gestionar retos	64
Tarea 6.1	Permitir al Maestro Guidoogway gestionar retos de Dominio de la Fuerza (Cuestionarios)	8
Tarea 6.2	Permitir al Maestro Guidoogway gestionar retos de Comida (Entregas individuales)	4
Tarea 6.3	Permitir al Maestro Guidoogway gestionar retos de Comida (Entregas de clanes)	4
Tarea 6.4	Permitir al Maestro Guidoogway gestionar retos de Agua (Entregas individuales)	4
Tarea 6.5	Permitir al Maestro Guidoogway gestionar retos de Agua (Entregas de clanes)	4
Tarea 6.6	Permitir al Maestro Guidoogway gestionar retos deportivos (Establecer parámetros deportivos a superar durante la semana)	8
Tarea 6.7	Permitir al Maestro Guidoogway gestionar retos de Felicidad	8
Tarea 6.8	Permitir al Maestro Guidoogway gestionar pruebas de nivel	8
Tarea 6.9	Permitir al Maestro Guidoogway gestionar pruebas de audiencia	8
Tarea 6.10	Permitir al Maestro Guidoogway asignar el coste en TdV de cada reto	4
Tarea 6.11	Permitir al Maestro Guidoogway asignar el beneficio obtenido por el Padawan en función de diferentes Badges	4

4.7.4. Carga prevista en los desarrolladores

A continuación mostramos la carga de trabajo prevista para cada uno de los desarrolladores en relación a las tareas definidas anteriormente:

Desarrollador	Velocidad inicial (días ideales)	Dedicación (% del tiempo)	Carga de trabajo (días ideales)	#Tareas aceptadas
Álvaro Fernández-Alonso Araluce	15	50 %	15	9

4.7.5. Planificación temporal de las tareas

Semana 1	Día 1	Día 2	Día 3	Día 4	Día 5
Álvaro Fernández-Alonso Araluce	5.1 5.2	6.1	6.1	6.2	6.3

Semana 2	Día 1	Día 2	Día 3	Día 4	Día 5
Álvaro Fernández-Alonso Araluce	6.4	6.5	6.6	6.6	6.7

Semana 3	Día 1	Día 2	Día 3	Día 4	Día 5
Álvaro Fernández-Alonso Araluce	6.7	Testing	Testing	Testing	Testing

4.8. Plan de entrega 2

4.8.1. Objetivos de la entrega

El objetivo principal de esta entrega consiste en seguir desarrollando funcionalidades para los usuarios padawans y empezar a desarrollar el apartado de gestión de retos desde el BackOffice.

4.8.2. Listado de HU a desarrollar

Identificador	Historias de Usuario	Estimación
HU.6	Permitir al Maestro Guidoogway gestionar retos	64
HU.8	Permitir al Maestro Guidoogway gestionar las vacaciones de los usuarios Padawans	8
HU.9	Permitir al Maestro Guidoogway gestionar los rangos de los usuarios Padawans	8

4.8.3. Descomposición en tareas de desarrollo

A continuación incluimos un desglose de las HU en tareas de desarrollo junto con la estimación realizada de su duración.

HU.8	Permitir al Maestro Guidoogway gestionar las vacaciones de los usuarios Padawans	8
Tarea 8.1	Permitir al Maestro Guidoogway gestionar las vacaciones de los usuarios Padawans globalmente	4
Tarea 8.2	Permitir al sistema gestionar la vuelta de las vacaciones de forma automática	4

HU.6	Permitir al Maestro Guidoogway gestionar retos	64
Tarea 6.1	Permitir al Maestro Guidoogway gestionar retos de Dominio de la Fuerza (Cuestionarios)	8
Tarea 6.2	Permitir al Maestro Guidoogway gestionar retos de Comida (Entregas individuales)	4
Tarea 6.3	Permitir al Maestro Guidoogway gestionar retos de Comida (Entregas de clanes)	4
Tarea 6.4	Permitir al Maestro Guidoogway gestionar retos de Agua (Entregas individuales)	4
Tarea 6.5	Permitir al Maestro Guidoogway gestionar retos de Agua (Entregas de clanes)	4
Tarea 6.6	Permitir al Maestro Guidoogway gestionar retos deportivos (Establecer parámetros deportivos a superar durante la semana)	8
Tarea 6.7	Permitir al Maestro Guidoogway gestionar retos de Felicidad	8
Tarea 6.8	Permitir al Maestro Guidoogway gestionar pruebas de nivel	8
Tarea 6.9	Permitir al Maestro Guidoogway gestionar pruebas de audiencia	8
Tarea 6.10	Permitir al Maestro Guidoogway asignar el coste en TdV de cada reto	4
Tarea 6.11	Permitir al Maestro Guidoogway asignar el beneficio obtenido por el Padawan en función de diferentes Badges	4

HU.9	Permitir al Maestro Guidoogway gestionar los rangos de los usuarios Padawans	8
Tarea 9.1	Permitir al Maestro Guidoogway gestionar los rangos de los usuarios Padawans	4
Tarea 9.2	Permitir al Maestro Guidoogway asignar un logo a cada rango	2
Tarea 9.3	Permitir al Maestro Guidoogway asignar la puntuación mínima necesaria para que un usuario Padawan consiga ese rango	2

4.8.4. Carga prevista en los desarrolladores

A continuación mostramos la carga de trabajo prevista para cada uno de los desarrolladores en relación a las tareas definidas anteriormente:

Desarrollador	Velocidad inicial (días ideales)	Dedicación (% del tiempo)	Carga de trabajo (días ideales)	#Tareas aceptadas
Álvaro Fernández-Alonso Araluce	15	50 %	15	9

4.8.5. Planificación temporal de las tareas

Semana 1	Día 1	Día 2	Día 3	Día 4	Día 5
Álvaro Fernández-Alonso Araluce	6.8 5.2	6.8	6.9	6.9	6.10

Semana 2	Día 1	Día 2	Día 3	Día 4	Día 5
Álvaro Fernández-Alonso Araluce	6.11	8.1	8.2	9.1	9.2

Semana 3	Día 1	Día 2	Día 3	Día 4	Día 5
Álvaro Fernández-Alonso Araluce	Testing	Testing	Testing	Testing	Testing

4.9. Plan de entrega 3

4.9.1. Objetivos de la entrega

El objetivo principal de esta entrega consiste en seguir desarrollando funcionalidades para los usuarios padawans y empezar a desarrollar el apartado de gestión de retos desde el BackOffice.

4.9.2. Listado de HU a desarrollar

Identificador	Historias de Usuario	Estimación
HU.10	Permitir al Maestro Guidoogway gestionar las misiones de desactivación de minas	12
HU.11	Permitir al Maestro Guidoogway gestionar los badges	4
HU.12	Preparar la sección de usuarios Padawans	20
HU.13	Control de accesos	4

4.9.3. Descomposición en tareas de desarrollo

A continuación incluimos un desglose de las HU en tareas de desarrollo junto con la estimación realizada de su duración.

HU.11	Permitir al Maestro Guidoogway gestionar los badges	4
Tarea 11.1	Permitir al Maestro Guidoogway gestionar los badges	3
Tarea 11.2	Permitir al Maestro Guidoogway asignar un logo a cada badge	1

HU.10	Permitir al Maestro Guidoogway gestionar las misiones de desactivación de minas	12
Tarea 10.1	Permitir al Maestro Guidoogway gestionar misiones de minas	4
Tarea 10.2	Permitir al Maestro Guidoogway asignar un código secreto para desactivar una mina	2
Tarea 10.3	Permitir al Maestro Guidoogway añadir pistas secretas para que los usuarios Padawans consigan la clave secreta	2
Tarea 10.4	Permitir al Maestro Guidoogway añadir un beneficio en TdV para los usuarios Padawans que consigan desactivar la mina	2
Tarea 10.5	Permitir al Maestro Guidoogway establecer la fecha y la hora de la explosión de la mina	2

HU.12	Preparar la sección de usuarios Padawans	20
Tarea 12.1	Creación en la plataforma Web del apartado para usuarios Padawan	4
Tarea 12.2	Colocación de las imágenes facilitadas por el cliente para cada sección	4
Tarea 12.3	Preparación de estilos, fondos y fuentes estipulados por el cliente	8
Tarea 12.4	Colocación del contador regresivo en cada sección	4

HU.13	Control de accesos	4
Tarea 13.1	Restringir el acceso a la sección de gestión solo para que puedan acceder usuarios Administradores	2
Tarea 13.2	Restringir el acceso a la sección de usuario solo para que puedan acceder usuarios Padawans	2

4.9.4. Carga prevista en los desarrolladores

A continuación mostramos la carga de trabajo prevista para cada uno de los desarrolladores en relación a las tareas definidas anteriormente:

Desarrollador	Velocidad inicial (días ideales)	Dedicación (% del tiempo)	Carga de trabajo (días ideales)	#Tareas aceptadas
Álvaro Fernández-Alonso Araluce	15	50 %	15	13

4.9.5. Planificación temporal de las tareas

Semana 1	Día 1	Día 2	Día 3	Día 4	Día 5
Álvaro Fernández-Alonso Araluce	10.1	10.2 10.3	10.4 10.5	11.1 11.2	12.1

Semana 2	Día 1	Día 2	Día 3	Día 4	Día 5
Álvaro Fernández-Alonso Araluce	12.2	12.3	12.3	12.4	13.1 13.2

Semana 3	Día 1	Día 2	Día 3	Día 4	Día 5
Álvaro Fernández- Alonso Araluce	Testing	Testing	Testing	Testing	Testing

Tecnologías usadas

En este proyecto se ha hecho uso de herramientas externas que nos han ayudado en diferentes aspectos durante el desarrollo del proyecto. A continuación vamos a nombrar algunas de ellas.

5.1. Google Analytics

Esta famosa herramienta nos va a ofrecer un feedback a tiempo real del tráfico y flujo de usuarios que interactúan con nuestra plataforma así como análisis y comparativas de esos flujos con respecto a diferentes períodos de tiempo.

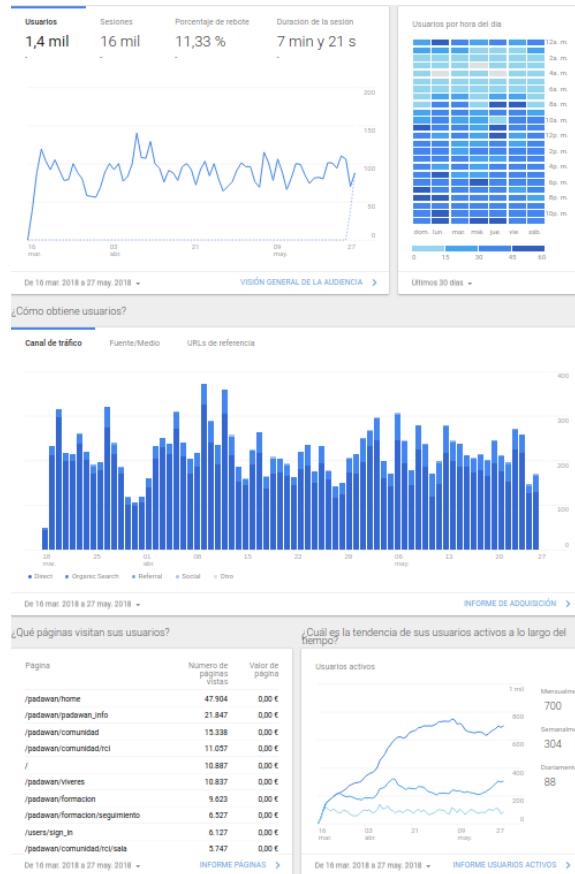


Figura 5.5: Google Analytics

5.2. Rollbar

Rollbar es un sistema de detección y diagnóstico de errores en tiempo real. En este proyecto está integrado mediante una **gema**, siguiendo una pequeña guía podremos conectar nuestra aplicación con su servicio. [4]. Debemos tener presente en todo momento que el ecosistema tecnológico está en continuo cambio por lo que cualquiera de estos podría afectar al correcto funcionamiento de nuestra plataforma, así que, una vez publiquemos nuestra aplicación deberíamos estar alerta de todos los errores que se suceden. Por ello, esta herramienta nos va a dar un feedback en tiempo real de los errores que sucedan así como entender cómo ocurren, por qué y dónde.

A continuación vamos a poner un ejemplo de un error en tiempo real y la potencia de esta herramienta.

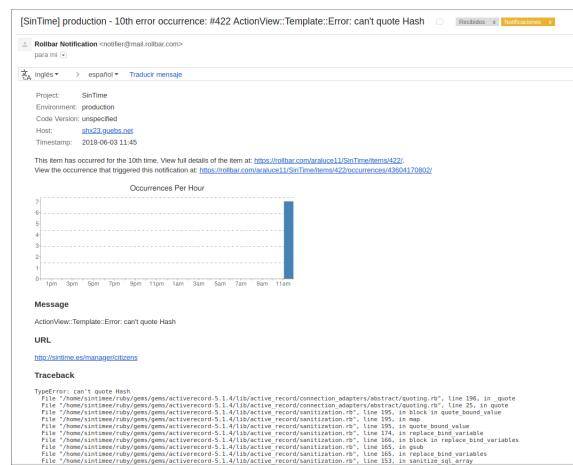


Figura 5.6: Rollbar - Notificación email

Si seguimos en enlace que nos marca la url podremos ver todos los detalles del error como la traza, las ocurrencias, navegadores y OS afectados y más detalles.

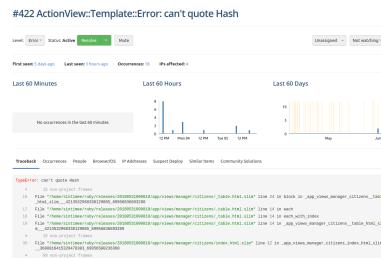


Figura 5.7: Rollbar - Rollbar detalle Traceback

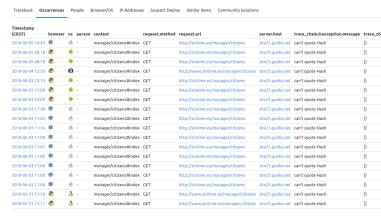


Figura 5.8: Rollbar - Rollbar detalle Occurrences

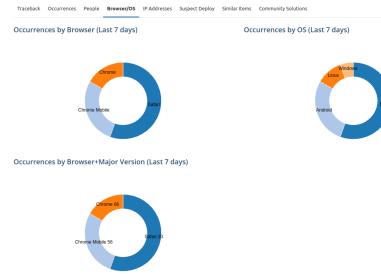


Figura 5.9: Rollbar - Rollbar detalle Browsers/OS

5.3. Port Monitor [2]

Servicio web creado por un **antiguo estudiante de la UGR**, Francis Yañez. Port Monitor es una herramienta fácil y en línea que supervisa el monitoreo del sitio web y del servidor para los usuarios las 24 horas del día, los 7 días de la semana. Registra el tiempo de actividad, los tiempos de respuesta del sitio web / servidor (rendimiento) y las causas del tiempo de inactividad. Genera informes personalizados y envía alertas instantáneas por correo electrónico o informes semanales / mensuales.

5.4. Runtastic

Ya que uno de los principales objetivos de la plataforma es mejorar el ritmo cardiorrespiratorio de los usuarios del sistema, debemos tener un servicio que monitorice los retos enfocados a los ejercicios físicos. En este proyecto actualizamos continuamente las sesiones que realizan cada uno de los padawans para evaluar semanalmente los resultados obtenidos.

Para ello necesitamos que el usuario nos facilite su usuario y contraseña de Runtastic, los almacenamos en la base de datos

5.5. Twitter

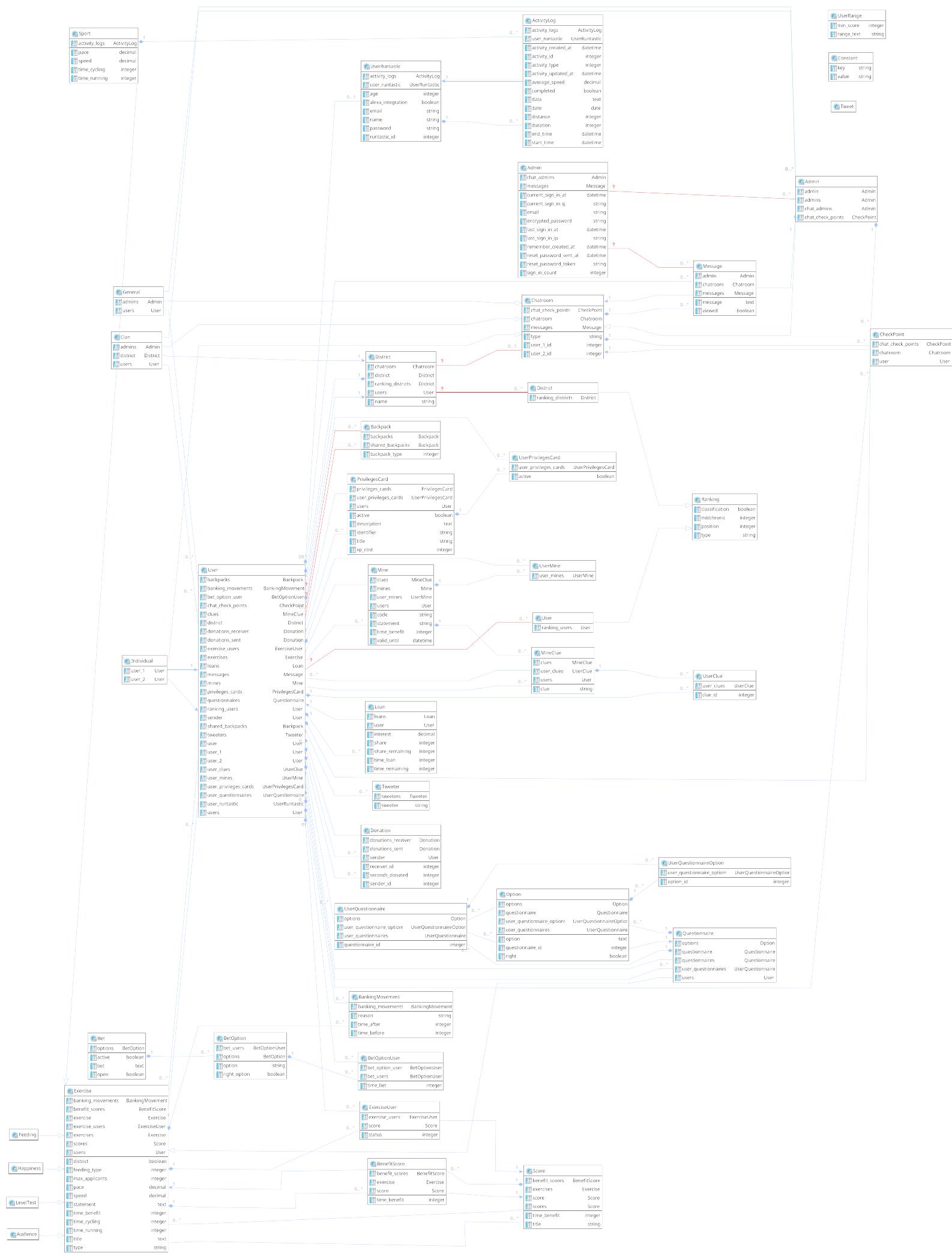
En esta aplicación hacemos uso de la API de Twitter ya que uno de los retos diarios consiste en almacenar 5 tweets diarios relevantes en el mundo de la Educación Física. Para ello tenemos que servir los tweets de usuarios de Twitter a los que los padawans sigan.



Figura 5.10: Twitter - Ejemplo

Diseño

6.1. Diagrama de clases



Análisis

7.1. Historias de usuario

#	Descripción	Dep.	Est.	Prio.
Servidor de producción				
1.1.	Instalación de Rails	-	5	1
1.2.	Configuración de la Base de Datos	1.1	1	1
1.3.	Git ignore	1.1 1.2	1	1
1.4.	Sistema de alertas (Rollbar)	1.1	4	2
1.5.	Desplegar cambios locales a producción	1.1 1.2 1.4	5	1
1.6.	Sistema de alertas (Port Monitor)	1.1	1	3
Sistema				
2.1.	Configuración de namespaces	-	1	1
2.2.	Gestión de sesiones con Devise	-	4	3
2.3.	Configuración de mailing	-	8	3
Alumnos				
3.1.	Reloj regresivo	-	2	1
Administración				
4.1.	Creación de usuarios(Lista DNIs)	-	4	1
4.2.	Creación de usuarios	4.1.	4	3
4.3.	Gestión de clanes	4.1.	2	2

Cuadro 7.2: Historias de usuario

7.2. Tarjetas de Servidor de Producción

7.2.1. Instalación de Ruby on Rails

1.1.	Instalación de Ruby on Rails						
Descripción							
Se debe instalar en el servidor el Framework Ruby on Rails. Proveeremos también de datos todos los ficheros de configuración del framework.							
<ul style="list-style-type: none"> ■ secrets.yml: que contendrá todas las contraseñas, o referencias a ellas, que necesite el sistema. ■ database.yml: que contendrá toda la información necesaria para acceder a la base de datos que hemos configurado en el servidor de producción. 							
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">Estimación</td><td style="padding: 2px;">5</td></tr> <tr> <td style="padding: 2px;">Prioridad</td><td style="padding: 2px;">1</td></tr> <tr> <td style="padding: 2px;">Dependencias</td><td style="padding: 2px;">-</td></tr> </table>		Estimación	5	Prioridad	1	Dependencias	-
Estimación	5						
Prioridad	1						
Dependencias	-						
<p>Pruebas de aceptación</p> <hr/> <p>-</p>							

Cuadro 7.3: Historia de usuario - Instalación de Ruby on Rails

7.2.2. Configuración de la Base de Datos

1.2.	Configuración de la Base de Datos						
Descripción							
Crearemos la base de datos del servidor de producción y obtendremos todos los datos necesarios para conectar con ella. La base de datos será de tipo MySQL.							
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">Estimación</td><td style="padding: 2px;">1</td></tr> <tr> <td style="padding: 2px;">Prioridad</td><td style="padding: 2px;">1</td></tr> <tr> <td style="padding: 2px;">Dependencias</td><td style="padding: 2px;">1.1</td></tr> </table>		Estimación	1	Prioridad	1	Dependencias	1.1
Estimación	1						
Prioridad	1						
Dependencias	1.1						
<p>Pruebas de aceptación</p> <hr/> <p>-</p>							

Cuadro 7.4: Historia de usuario - Configuración de la Base de Datos

7.2.3. Configuración de gitignore

1.3.	Configuración de gitignore
Descripción	
Configuraremos el fichero <i>gitignore</i> para que ignore los cambios en los ficheros de configuración anteriores. De esta forma evitaremos que se sustituyan estos ficheros de la parte local con los del servidor de producción, donde estarán las configuraciones reales.	
Estimación	1
Prioridad	1
Dependencias	1.1 1.2
Pruebas de aceptación	
-	

Cuadro 7.5: Historia de usuario - Configuración de gitignore

7.2.4. Sistema de alertas Rollbar

1.4.	Sistema de alertas (Rollbar)
Descripción	
Haremos la conexión de nuestro sistema con el servicio Rollbar para recibir alertas sobre errores en tiempo real. Para ello tenemos que registrarnos en su página web a través de la url [5] y seguiremos la guía de configuración facilitada en [4].	
Estimación	1
Prioridad	1
Dependencias	1.1
Pruebas de aceptación	
-	

Cuadro 7.6: Historia de usuario - Sistema de alertas (**Rollbar**)

7.2.5. Desplegar cambios locales a producción

1.5.	Desplegar cambios locales a producción
Descripción	
Configuraremos todas las acciones que va a realizar el servidor cuando demos la orden de despliegue. Para ello haremos uso de la gema Capistrano. Las acciones que vamos a realizar serán, por orden, las siguientes:	
<ul style="list-style-type: none"> ■ Conectará con la rama master de nuestro proyecto en github y lo descargará. ■ Instalará las nuevas gemas que hayamos incluido ■ Compilará los assets y los comprimirá ■ Ejecutará las migraciones pendientes de ejecutar ■ Cambiará el puntero del directorio principal de la web para que apunte a los nuevos cambios ■ Reiniciará la Web App 	
Si al ejecutar el despliegue se producen errores no se producirán cambios en el servidor de producción y rollbar nos notificará el error vía email.	
Estimación	5
Prioridad	1
Dependencias	1.1 1.2 1.4
Pruebas de aceptación	
-	

Cuadro 7.7: Historia de usuario - Desplegar cambios locales a producción

7.2.6. Sistema de alertas Port Monitor

1.6.	Sistema de alertas Port Monitor
Descripción	
Haremos la conexión de nuestro sistema con el servicio Port Monitor para recibir alertas sobre el estado de nuestro dominio. Para ello nos registraremos en su página web a través de la url [3] y seguiremos la navegación normal para que haga cheking de nuestro dominio.	
Estimación	1
Prioridad	1
Dependencias	1.1
Pruebas de aceptación	
-	

Cuadro 7.8: Historia de usuario - Sistema de alertas Port Monitor

7.3. Sistema

7.3.1. Configuración de namespaces

2.1.	Configuración de namespaces
Descripción	
Haremos una separación en la raíz del proyecto a partir de la cual van a crecer nuestros diferentes layouts. El sistema lo vamos a dividir en 3 partes:	
1.	Padawan: Será la sección a la que podrán acceder sólo los <i>usuarios de tipo estudiante</i> .
2.	Manager: Será la sección a la que podrán acceder sólo los <i>usuarios de tipo admin</i> , es la sección en la que el profesor podrá crear contenido para los estudiantes y gestionar gran parte de las funciones de la aplicación.
3.	Super Admin: Será la sección a la que podrán acceder sólo los <i>usuarios de tipo admin</i> . En ella podremos gestionar todos los aspectos posibles de la aplicación. Esta sección está reservada para el mantenimiento de la aplicación.
Estimación	1
Prioridad	1
Dependencias	-
Pruebas de aceptación	
- Ninguna de estas secciones podrá ser pública.	
- Sólo se podrá acceder a cada una de las secciones previo registro.	

Cuadro 7.9: Historia de usuario - Configuración de namespaces

Prensa

8.1. Prensa digital

PRENSA: Digital

- 20/05/2017 • **Granada Hoy:** <http://www.granadahoy.com/vivir/In-Time-mejora-fitness-caracteristicas-1148885257.html>
- 16/06/2017 • **Canal Sur Andalucía:** <http://www.canalsur.es/noticias/tecnicas-innovadoras/1177803.html>
- 26/06/2017 • **Canal UGR:** <https://canal.ugr.es/noticia/proyecto-gamificacion-in-time>
- 28/06/2017 • **Aula Magna:** <http://www.aulamagna.com.es/in-time-innovacion-docente-granada>
- 28/06/2017 • **Info Costa Tropical:** http://www.infocostatropical.com/mb/noticia.asp?id_noticia=1177803
- 07/07/2017 • **Esto no es Finlandia:** <https://estonoesfinlandia.wordpress.com/2017/07/07/estos-no-solo-son-jedis>
- 23/05/2018 • **Canal UGR:** <https://canal.ugr.es/noticia/los-primeros-jedi-ugr>
- 28/05/2018 • **Granada Hoy:** <http://www.granadahoy.com/vivir/Star-Wars-Episodio-Primero-1148885257.html>
- 28/05/2018 • **Europa press:** <http://www.europapress.es/andalucia/noticia-profesor-facultad-ciencias-20180523122857.html>
- 28/05/2018 • **Ideal:** <http://www.ideal.es/miugr/profesor-facultad-ciencias-20180523122857.html>
- 28/05/2018 • **Ahora Granada:** <https://www.ahoragranada.com/noticia/los-primeros-jedi-ugr-1148885257.html>

8.2. Radio

PRENSA: *Radio*

24/05/2018 • **Canal Sur Radio:** https://drive.google.com/open?id=1n_DUIsL96aHi8

8.3. Prensa escrita

8.3.1. 27/06/2017 - Granada Hoy



Figura 8.11: Prensa 27/06/2017 - Granada Hoy

8.3.2. 28/06/2017 - Ideal



Figura 8.12: Prensa 28/06/2017 - Ideal

8.3.3. 27/07/2017 - Ideal



Figura 8.13: Prensa 27/07/2017 - Ideal

Bibliografía

- [1] *Pomodoro Timer*. [Último acceso online el 1-Septiembre-2018] - <https://pomodoro-tracker.com>.
- [2] *Port Monitor*. [Último acceso online el 5-Junio-2018] - <https://es.port-monitor.com>.
- [3] *Port Monitor Sing Up*. [Último acceso online el 12-Junio-2018] - <https://es.port-monitor.com/users/sign-up>.
- [4] *Ruby Doc*. [Último acceso online el 5-Junio-2018] - <https://docs.rollbar.com/docs/ruby>.
- [5] *Ruby Sing Up*. [Último acceso online el 12-Junio-2018] - <https://rollbar.com/signup>.
- [6] CONTRERAS, R.S. Y EGUIA, J. *Gamificación en aulas universitarias*. Bellaterra: Institut de la Comunicació-Universitat Autònoma de Barcelona., 2016.
- [7] DETERDING, S., D. D. K. R. Y. N.-L. From Game Design Elements to Gamefulness: Defining “gamification”. In *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments*. Tampere: ACM., 2011.
- [8] GONZÁLEZ GONZÁLEZ CS, M. A. Técnicas de gamificación aplicadas en la docencia de Ingeniería Informática. *ReVision*, 8(1), 2015.
- [9] JORGE MORENO J.; GIL; L. MERINO, F. Y SANZ, M. Evidencia empírica de los motivos del absentismo en los estudiantes universitarios. *Revista de Investigación en Educación*, 9 (2), 2011, 76-90, 2011.
- [10] KIRYAKOVA, G., A. N. Y. Y. L. *Gamification in Education*. Stara Zagora: Proceedings of 9th International Balkan Education and Science Conference., 2014.
- [11] MARTÍ-PARRE NO, J., Q.-A. C. M.-I. N. E. G.-F. E. *El uso de la gamificación en la educación superior: el caso de trade ruler*. En xii

Jornadas Internacionales de Innovación Universitaria Educar para transformar: Aprendizaje experiencial., 2014.

- [12] MARTÍNEZ GONZÁLEZ, J. La motivación para aprender en el espacio europeo de educación superior. *Cuadernos de Educación y Desarrollo*, 3 (25)., 2011.
- [13] PRENSKY, M. *Digital Game-Based Learning*. New York: McGraw-Hill, 2001.
- [14] PRIETO A, DÍAZ D, M. J. R. E. *Experiencias de aplicación de estrategias de gamificación a entornos de aprendizaje universitario*. ReVision, 7(2), 27-43., 2014.