**Proyecto**

“Programming Cloud”

**Grupo**

“Braineering”

"DOCUMENTO DE ESPECIFICACIÓN"

Hito: 0

Fecha entrega: 19-10-2016

Versión: 1

Componentes:

* Jesús Sánchez de Pablo
* Antonio Candela Ibáñez
* George Valentin Cristian Ututui
* Rubén Sánchez Cuenca

**Contenido**

[Contenido](#_gjdgxs)

[1. Introducción](#_1fob9te)

[1.1. Propósito](#_3znysh7)

[2. Descripción general](#_2et92p0)

[2.1. Ámbito del sistema (Perspectiva del producto).](#_tyjcwt)

[2.2. Funcionalidades generales.](#_3dy6vkm)

[2.3. Características de los usuarios.](#_1t3h5sf)

[2.4. Interfaces externas.](#_4d34og8)

[2.5. Requisitos (suposiciones y dependencias).](#_2s8eyo1)

[2.6. Restricciones.](#_17dp8vu)

[2.7. Requisitos futuros.](#_3rdcrjn)

[3. Requerimientos específicos](#_26in1rg)

[3.1. Requerimientos funcionales.](#_lnxbz9)

[3.2. Requerimientos no funcionales.](#_35nkun2)

[4. Apéndices](#_1ksv4uv)

[4.1. Definiciones, acrónimos y abreviaturas](#_44sinio)

[4.2. Referencias](#_2jxsxqh)

# Introducción

En esta sección se proporcionará una introducción a todo el documento de Especificación de Requisitos. Este es un documento para la asignatura de Proyectos Multimedia para el grado de Ingeniería Multimedia de la Universidad de Alicante.

## Propósito

Este documento tiene como objetivo definir las especificaciones funcionales y no funcionales para la creación de un sistema de aprendizaje online que permitirá el estudio de lenguajes de programación por estudiantes,profesores y autodidactas.

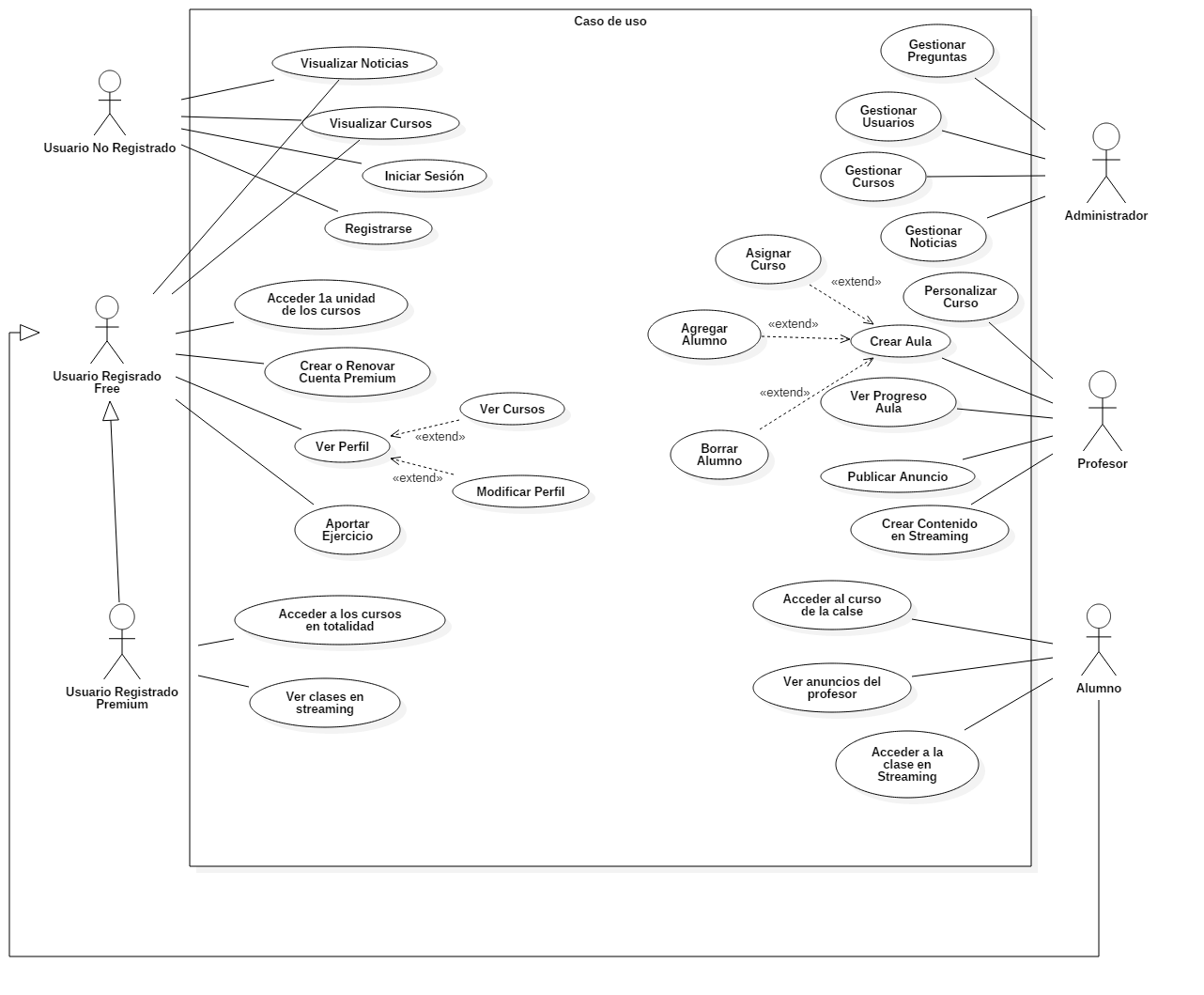
# Descripción general

En esta sección se definirán todos aquellos matices que afectan al producto y sus requisitos. Aquí lo que se describe es el contexto de los requisitos del sistema. De esta forma se permitirá precisar con más detalle los requisitos y hacer que sean más fáciles de entender.

## Ámbito del sistema (Perspectiva del producto).

El sistema Programming Cloud será una plataforma creada para trabajar en entorno web, lo que permitirá un fácil y rápido acceso a ella.

## Funcionalidades generales.

Los principales usuarios del sistema van a ser los alumnos y los profesores de dichos alumnos. Las funcionalidades del sistema serán diferentes en función del usuario que está utilizando el sistema. Un profesor puede crear lecciones, invitar a sus alumnos a dichas lecciones y ver sus progresos. Mientras que la ocupación del alumno será la de cursar dichas lecciones propuestas por el profesor. Estas clases van a tener una serie de PDFs, vídeos ejercicios y una evaluación final.

## Características de los usuarios.

En esta subsección vamos a describir las características generales de los usuarios de nuestro sistema, incluyendo datos relevantes como su nivel educacional y experiencia técnica.

|  |  |
| --- | --- |
| **Tipo de Usuario** | Profesor |
| **Formación** | Grado en Informática, Grado en una Ingeniería, Educador, Experto en algún lenguaje de programación |
| **Actividades** | Gestionar y Crear Lecciones, Evaluar Alumnos |

|  |  |
| --- | --- |
| **Tipo de Usuario** | Alumno |
| **Formación** | Secundaria |
| **Actividades** | Participación en los cursos compartidos por los profesores |

|  |  |
| --- | --- |
| **Tipo de Usuario** | Autodidacta |
| **Formación** | Sin especificar |
| **Actividades** | Asistir a los cursos disponibles en nuestra web |

|  |  |
| --- | --- |
| **Tipo de Usuario** | Administrador |
| **Formación** | Gestor de la página |
| **Actividades** | Controlar, Dirigir y Actualizar el sistema |

## Interfaces externas.

### Interfaces de usuario

En la página de inicio aparecerá el logo y la imagen corporativa de la empresa.

Estará disponible un tutorial para el uso de la página mediante un avatar.

### Interfaces de hardware

Acceso al almacenamiento interno del dispositivo para almacenar las credenciales del usuario.

Uso de ratón y teclado para interactuar con la página.

### Interfaces de software

Deberá tener acceso a la API de Google+ para poder utilizar la función de inicio de sesión.

Deberá tener acceso a la API de Facebook para poder utilizar la función de inicio de sesión.

### Interfaces de comunicación

Uso del asistente para conferencias Web para gestionar el streaming de las clases en directo.

Uso de los canales de Youtube para colgar lecciones, tutoriales y videos promocionales.

## Requisitos (suposiciones y dependencias).

Al utilizar Apache, el número de llamadas que es capaz de recibir nuestro servidor de manera simultánea es bastante limitado, en caso de que percibir un aumento creciente de la actividad de nuestra web, deberíamos cambiar a herramientas como node.js, la cual aumentaría la escalabilidad de nuestra página y la cantidad de conexiones simultáneas a nuestro sistema.

Para ahorrar memoria en nuestro servidor, nos valdremos de herramientas externas como Youtube o Big Blue Button. Por contrapartida, esto supondrá que dependeremos de ellas, por lo tanto en caso de que fallara su sistema o dejaran de permitirnos su uso, nos veríamos obligados bien a prescindir de algunos de nuestros contenidos, o bien a almacenarlos dentro de nuestro servidor, lo cual aumentaría considerablemente los costes de mantenimiento de nuestra página.

La inclusión de nuestro sistema de aprendizaje mediante audiolibros y la versión como aplicación android, son añadidos a nuestra página, esto quiere decir que aunque tengamos la intención de incluirlos dentro de nuestro producto final, podemos llegar a prescindir de ellos en caso de necesidad, bien sea por falta de tiempo o por problemas inesperados surgidos durante el curso.

## Restricciones.

Programming Cloud será una página web, lo cual implica que deberemos adecuarnos a los buscadores más utilizados para que nuestra página web pueda ser lo más versátil posible, es decir, que cuide el SEO, posicionamiento, accesibilidad de contenidos para que los buscadores nos posicionen bien en sus índices, además de que la web ha de ser adaptable, usable y cómoda para funcionar bien en cualquier navegador.

Utilizaremos javascript, PHP y WebGL como lenguajes de programación.

## Requisitos futuros.

En caso de crear una versión móvil para Android, deberemos acomodar dicha aplicación a su SO.

Además como mejora del contenido, se podría ampliar el campo de las asignaturas; por ejemplo relacionando otras con el ámbito de la programación como pueden ser las matemáticas o incluso el dibujo técnico, aportando ejercicios en el que haya que dibujar alguna figura o incluso teniendo que realizar algún cálculo exponencial, todo esto mediante código.

# Requerimientos específicos

## Requerimientos funcionales.

En esta sección veremos las distintas funcionalidades que puede desempeñar cada uno de los usuarios dentro de nuestra página web, y los requerimientos necesarios para poder realizarlos. Para ello hemos seguido las operaciones indicadas en el diagrama de casos de uso mostrado anteriormente.

### Requerimientos para Usuario invitado

* **Visualizar Noticias**

|  |  |
| --- | --- |
| **Requerimiento** | RF1 |
| **Nombre del requerimiento** | Visualizar Noticias |
| **Descripción del requerimiento** | Los usuarios pueden ver el panel de noticias relacionadas con la página web en la página principal |
| **Prioridad del requerimiento** | Media |

**Justificación:** Para que el usuario pueda saber más información sobre nuestra página y se interese en acceder a nuestros cursos.

**Precondiciones:** Entrar a la página principal de la web.

**Datos de entrada:** No consta.

**Descripción:** En la página de inicio el usuario podrá ver la siguiente información sobre nuestros cursos:

* Ejercicios más destacados y el usuario que los ha subido (Este no tiene porqué ser un administrador).
* Comentarios de usuarios respecto a la página (ekomi).
* Descripción de nuestra web: quienes somos, que ofrecemos, a quien se lo ofrecemos y con qué finalidad lo hacemos.

**Pruebas de aceptación:** No consta.

**Postcondiciones:** El usuario se ha informado sobre nuestra página, lo que posiblemente aumentará su interés en acceder a los servicios que esta ofrece.

**Errores:**  No consta.

* **Visualizar Cursos**

|  |  |
| --- | --- |
| **Requerimiento** | RF2 |
| **Nombre del requerimiento** | Visualizar Cursos |
| **Descripción del requerimiento** | Los usuarios pueden acceder a una descripción de los cursos del que dispone nuestra plataforma de aprendizaje |
| **Prioridad del requerimiento** | Media |

**Justificación:** Para que el usuario pueda saber más información sobre los cursos que proporciona nuestra web.

**Precondiciones:** Entrar a la página principal de la web.

**Datos de entrada:** No consta.

**Descripción:** El usuario podrá acceder a una breve descripción de cada uno de los cursos impartidos. Además proporcionaremos información sobre el funcionamiento de estos cursos, por ejemplo, cómo los profesores podrán personalizar estos para la estructuración de su plan de estudios.

**Pruebas de aceptación:** No consta.

**Postcondiciones:** El usuario se ha informado sobre nuestra página, lo que posiblemente aumentará su interés en acceder a los servicios que esta ofrece.

**Errores:**  No consta.

* **Contactar con nosotros**

|  |  |
| --- | --- |
| **Requerimiento** | RF3 |
| **Nombre del requerimiento** | Contactar con Administrador |
| **Descripción del requerimiento** | Los usuarios podrán ponerse en contacto con los administradores de la página |
| **Prioridad del requerimiento** | Media |

**Justificación:** Lo implementaremos principalmente para que instituciones o profesores puedan ponerse en contacto con nosotros y darse de alta en nuestra web.

**Precondiciones:** Entrar a la página principal de la web.

**Datos de entrada:**

* Nombre del usuario.
* Institución a la cual representa, en caso de que lo haga.
* Email del usuario.
* Asunto del mensaje.
* Mensaje.

**Descripción:** El usuario podrá ponerse en contacto con nosotros mediante un sistema de mensajería.

**Pruebas de aceptación:** No consta.

**Postcondiciones:** El usuario ha enviado su mensaje correctamente tras haber confirmado su correo electrónico.

**Errores:**  No consta.

* **Registrarse**

|  |  |
| --- | --- |
| **Requerimiento** | RF4 |
| **Nombre del requerimiento** | Registro de Usuario |
| **Descripción del requerimiento** | El usuario podrá darse de alta dentro de nuestra página web, lo cual le permitirá acceder a algunos materiales de nuestro cursos. |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Para que el usuario pueda acceder a los cursos, subir ejercicios, recibir notificaciones y en general, acceder a los contenidos de nuestra página.

**Precondiciones:** El usuario no debe de tener una cuenta ya creada. Por otro lado, si desea enlazar su cuenta con la de Facebook o Google+, deberá tener una cuenta en ellas.

**Datos de entrada:**

* Nombre del usuario.
* Institución a la cual representa, en caso de que lo haga.
* Email del usuario.
* Contraseña
* Repetir Contraseña
* Pregunta de confirmación (en caso de que quiera cambiar su contraseña)
* Enlazar con facebook o google+

**Descripción:** El usuario no registrado podrá darse de alta en nuestra web rellenando los datos solicitados. También podrá enlazar su cuenta a facebook o google+.

**Pruebas de aceptación:**

* La contraseña debe de tener más de seis caracteres e incluir letras y números.
* La contraseña repetida deberá ser idéntica a la anterior.
* Para verificar el registro a nuestra página, el usuario deberá hacer click en el enlace que tendrá en el mensaje enviado al correo electrónico especificado anteriormente.
* La pregunta de confirmación es un campo obligatorio.
* No pueden haber dos usuarios con el mismo email.

**Postcondiciones:** El usuario se ha registrado correctamente y se ha almacenado su información en el sistema.

**Errores:**  Si el usuario no consigue crear la cuenta después de 5 intentos, podrá solicitar ayuda al administrador mediante un mensaje.

* **Recuperar Contraseña**

|  |  |
| --- | --- |
| **Requerimiento** | RF5 |
| **Nombre del requerimiento** | Recuperar Contraseña |
| **Descripción del requerimiento** | En caso de haber olvidado su contraseña, el usuario podrá completar un formulario para recuperarla. |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Para que el usuario pueda recuperar su contraseña en caso de haberla olvidado.

**Precondiciones:** El usuario debe de tener una cuenta creada.

**Datos de entrada:**

* Email
* Respuesta a pregunta de confirmación

**Descripción:** El usuario podrá recuperar su contraseña tras haber rellenado correctamente el formulario.

**Pruebas de aceptación:**

* El email establecido debe estar registrado en la base de datos.
* La respuesta a la pregunta de confirmación deberá coincidir con la establecida por el usuario a la hora del registro.

**Postcondiciones:** El usuario recibirá vía email su antigua contraseña.

**Errores:**  Si el usuario no consigue validar después de 5 intentos, podrá solicitar ayuda al administrador mediante un mensaje.

* **Iniciar Sesión**

|  |  |
| --- | --- |
| **Requerimiento** | RF6 |
| **Nombre del requerimiento** | Iniciar Sesión |
| **Descripción del requerimiento** | Un usuario podrá acceder a la plataforma mediante su usuario y contraseña. |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Para que el usuario pueda acceder a la plataforma.

**Precondiciones:** El usuario debe de tener una cuenta creada o tener cuenta de Facebook o Google+

**Datos de entrada:**

* Usuario
* Contraseña

**Descripción:** El usuario podrá acceder a los distintos servicios de la plataforma, mediante una cuenta ya creada o las distintas APIs, ya se por Facebook o Google.

**Pruebas de aceptación:**

* El nombre de usuario y la contraseña deben ser correctas, tanto si se accede con los datos de una cuenta creada o mediante una API externa.

**Postcondiciones:** El usuario accede a la plataforma.

**Errores:**  Si el usuario no consigue acceder después de 5 intentos, podrá solicitar recuperar la contraseña.

### Requerimientos Usuario Registrado (Free)

Además de los requisitos que aquí se encuentran especificados este usuario también puede realizar los requisitos RF1, RF2 y RF3 que son parte de los requisitos del usuario no registrado.

* **Interactuar con Avatar**

|  |  |
| --- | --- |
| **Requerimiento** | RFTAG |
| **Nombre del requerimiento** | Interactuar con un Avatar |
| **Descripción del requerimiento** | El usuario va a tener un avatar en todo momento en la pantalla para la ayuda |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Para que el usuario pueda interactuar de manera más cómoda y divertida con el sistema de nuestra plataforma.

**Precondiciones:** Debe estar registrado.

**Datos de entrada:** Se debe elegir el género del avatar.

**Descripción:** El avatar de nuestra plataforma realizará las siguientes actividades:

* Nos acompañará durante los distintos tutoriales presentes en nuestra web, por ejemplo en la introducción a la utilización de los cursos y también en el tutorial de bienvenida de la página web.
* Durante la realización de un ejercicio, podemos clicar sobre el avatar para acceder al menú de ayuda. Nos dará pequeñas pistas para que el usuario pueda completar el ejercicio.
* El avatar será el personaje que aparecerá y que controlaremos en los juegos presentes de la plataforma.
* Podemos elegir la localización del avatar en la pantalla, o incluso podremos ocultarlo.

**Pruebas de aceptación:** No consta.

**Postcondiciones:** El usuario podrá estar informado de forma divertida sobre la funcionalidad de la web.

**Errores:**  No consta.

* **Realizar tutorial de bienvenida a la plataforma (e-learning)**

|  |  |
| --- | --- |
| **Requerimiento** | RF7 |
| **Nombre del requerimiento** | Realizar tutorial bienvenida (e-learning) |
| **Descripción del requerimiento** | Un usuario podrá hacer el tutorial de bienvenida para conocer la plataforma |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Para que el usuario pueda entender el funcionamiento de la plataforma de una manera interactiva.

**Precondiciones:** Debe estar registrado.

**Datos de entrada:** No consta.

**Descripción:** El usuario elegirá si quiere hacer o no el tutorial de bienvenida basado en e-learning, si decide hacerlo: se le guiará de manera interactiva por la plataforma para que conozca su funcionamiento..

**Pruebas de aceptación:** No consta.

**Postcondiciones:** El usuario finaliza el tutorial y puede desplazarse por las diferentes opciones que ofrece la plataforma.

**Errores:**  No consta.

* **Acceder a primera lección de los cursos**

|  |  |
| --- | --- |
| **Requerimiento** | RF8 |
| **Nombre del requerimiento** | Acceder primera lección cursos |
| **Descripción del requerimiento** | Un usuario podrá realizar un número limitado de lecciones dentro de los cursos de nuestra colección. |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Como nuestra página web trata sobre lecciones online, es necesario que el usuario pueda realizar un número limitado de lecciones. Antes de comprar un curso, el usuario podrá acceder a las lecciones gratuitas del mismo.

**Precondiciones:** El usuario debe estar como mínimo registrado para poder llevar a cabo la realización de las lecciones gratuitas.

**Datos de entrada:** No consta.

**Descripción:** El usuario podrá acceder a cualquier lección gratuita de la página, si quiere ampliar las lecciones y poder participar en todas deberá estar registrado como premium o pertenecer a alguna institución.

Las lecciones estarán formadas por los siguientes elementos:

* Video introductorio a la lección.
* Documento pdf con las definiciones y conceptos más teóricos.
* Posibilidad de generar un audio en base al pdf anterior.
* Ejercicio de ejemplo o tutorial e-learning del videojuego de ejemplo.
* Ejercicios basados en la temática de la lección.
* Evaluación mediante test.

**Pruebas de aceptación:** No consta.

**Postcondiciones:** Tras acceder a su primera lección, se le mostrará un tutorial basado en e-learning sobre el curso, puede saltarlo y pasar directamente a la participación y finalización de la lección por parte del usuario.

**Errores:**  No consta.

* **Acceder a primera lección de los cursos - Realizar tutorial del curso (e-learning)**

|  |  |
| --- | --- |
| **Requerimiento** | RF8.1 |
| **Nombre del requerimiento** | Realizar tutorial bienvenida (e-learning) |
| **Descripción del requerimiento** | Un usuario podrá hacer el tutorial de bienvenida para conocer la plataforma |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Para que el usuario pueda entender el funcionamiento del curso de una manera interactiva.

**Precondiciones:** Debe ser el primer curso a realizar.

**Datos de entrada:** No consta.

**Descripción:** El usuario elegirá si quiere hacer o no el tutorial del curso basado en e-learning, si decide hacerlo: se le mostrarán las diferentes pantallas que aparecerán en el curso (donde programe y donde aparezcan los resultados) y habrá una breve explicación de cada una de ellas.

**Pruebas de aceptación:** No consta

**Postcondiciones:** El usuario finaliza el tutorial y puede comenzar con la lección

**Errores:**  No consta.

* **Comprar curso o pack de cursos**

|  |  |
| --- | --- |
| **Requerimiento** | RF9 |
| **Nombre del requerimiento** | Comprar curso o pack de cursos |
| **Descripción del requerimiento** | Un usuario podrá comprar uno o varios cursos. |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Necesario si el usuario quiere realizar algún curso completo de la web.

**Precondiciones:** Estar registrado en el sistema.

**Datos de entrada:**

* Cursos que haya escogido.
* Plazo de pago (mensual, trimestral, semestral o anual)
* El pago lo haremos a través de TPV luego la validación de la tarjeta del usuario no dependerá de nosotros.

**Descripción:** El usuario podrá comprar un curso o un pack de estos. Los packs tendrán un descuento en proporción a la cantidad de cursos de los que disponga.

**Pruebas de aceptación:** Debe haber rellenado todos los campos.

**Postcondiciones:** Una vez haya comprado un curso, podrá acceder a todas las lecciones disponibles en este, además pasará a ser un usuario de clase premium.

**Errores**: No consta.

* **Ver perfil**

|  |  |
| --- | --- |
| **Requerimiento** | RF10 |
| **Nombre del requerimiento** | Ver perfil |
| **Descripción del requerimiento** | Un usuario podrá revisar su perfil |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Para que el usuario pueda llevar un control de su cuenta.

**Precondiciones:** Debe estar registrado.

**Datos de entrada:** No consta.

**Descripción:** El usuario podrá acceder a su perfil y actualizarlo en cualquier momento.

**Pruebas de aceptación:** No consta.

**Postcondiciones:** El usuario habrá actualizado sus datos.

**Errores:**

* Si alguno de los datos actualizados contiene errores o palabras ofensivas, saltará un mensaje de error.
* **Ver cursos**

|  |  |
| --- | --- |
| **Requerimiento** | RF10.1 |
| **Nombre del requerimiento** | Ver cursos |
| **Descripción del requerimiento** | Un usuario podrá ver sus cursos. |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Para que el usuario pueda llevar un control de los cursos realizados o en progreso.

**Precondiciones:** Debe haber realizado o estar realizando un curso.

**Datos de entrada:** No consta.

**Descripción:** El usuario podrá comprobar los avances en cada curso, así como, todos los terminados hasta ese momento, además podrá visualizar las insignias ganadas.

**Pruebas de aceptación:** No consta.

**Postcondiciones:** El usuario estará satisfecho con las estadísticas individualizadas que se le ofrecen.

**Errores:**  No consta.

* **Modificar perfil**

|  |  |
| --- | --- |
| **Requerimiento** | RF11 |
| **Nombre del requerimiento** | Modificar perfil |
| **Descripción del requerimiento** | Un usuario podrá modificar su perfil |
| **Prioridad del requerimiento** | Alta |

**Justificación:** El usuario debe tener la posibilidad de modificar alguno de sus datos del perfil.

**Precondiciones:** Debe haberse registrado y tener un perfil propio.

**Datos de entrada:**

* Nombre y apellidos.
* Nombre de usuario.
* Contraseña
* Correo.
* Número de cuenta, en el caso de ser premium.
* Foto de usuario.

**Descripción:** El usuario podrá modificar cualquiera de los datos de entrada nombrados anteriormente.

**Pruebas de aceptación:**

* El nuevo nombre de usuario debe estar disponible y no contener nada de carácter obsceno.
* La nueva contraseña debe regirse por las expresiones regulares establecidas.
* La nueva foto no debe contener nada de carácter obsceno.
* El nuevo número de cuenta debe ser válido y con saldo suficiente.
* El nuevo correo no debe existir en la base de datos.

**Postcondiciones:** El usuario habrá modificado a su antojo su perfil.

**Errores:**  Si cualquiera de las pruebas de aceptación falla, habrá un mensaje de error indicando que la modificación no ha podido realizarse.

* **Modificar perfil - Darse de baja**

|  |  |
| --- | --- |
| **Requerimiento** | RF11.1 |
| **Nombre del requerimiento** | Darse de baja |
| **Descripción del requerimiento** | Un usuario podrá eliminar su cuenta |
| **Prioridad del requerimiento** | Alta |

**Justificación:** La opción de darse de baja debe existir para que un usuario pueda eliminar su cuenta cuando guste.

**Precondiciones:** Debe tener una cuenta activa.

**Datos de entrada:**

* Debe rellenar un pequeño formulario con cuestiones básicas sobre el motivo de su baja.

**Descripción:** Si un usuario se encuentra insatisfecho con la página o simplemente decide dejar el curso y borrar su rastro en la página, podrá hacerlo desde su perfil y se nos notificará de su baja.

**Pruebas de aceptación:** Contestar formulario de baja.

**Postcondiciones:** El usuario dejará de formar parte de nuestra base de datos y de nuestra web.

**Errores:**  No consta.

* **Aportar ejercicios**

|  |  |
| --- | --- |
| **Requerimiento** | RF12 |
| **Nombre del requerimiento** | Aportar ejercicios |
| **Descripción del requerimiento** | Un usuario podrá aportar ejercicios a la web. |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Para incentivar al desarrollo de la página y del usuario, habrá un apartado de subir ejercicios.

**Precondiciones:** El ejercicio debe de aprobar unos controles de calidad aplicado por los administradores.

**Datos de entrada:**

* Como se sube desde su cuenta, el ejercicio quedará grabado con el usuario que lo subió.

**Descripción:** Si un usuario quiere ayudar a la web, o simplemente ser reconocido dentro de la misma, puede hacer aportaciones de ejercicios, los cuales quedarán grabados en la página con el nombre del usuario y la fecha en la que lo subió.

**Pruebas de aceptación:**

* Un administrador debe dar validar dicho ejercicio.

**Postcondiciones:** Todos los usuarios registrados de la web, podrán visualizar el contenido subido por el usuario.

**Errores:**  Si el ejercicio no aprueba los exámenes de calidad, se le notificará al usuario indicando el motivo.

### Requerimientos Usuario Registrado (Premium)

Tenemos que especificar que un usuario registrado premium también hereda los requisitos RF1, RF2 y RF3 del usuario no registrado y también todos los requisitos que se presentan de un usuario Registrado Free.

* **Acceder a sus cursos**

|  |  |
| --- | --- |
| **Requerimiento** | RF13 |
| **Nombre del requerimiento** | Acceder sus cursos |
| **Descripción del requerimiento** | Un usuario premium accede cualquiera de los cursos que haya comprado. |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Representa la principal característica de nuestro sistema para los usuarios registrados.

**Precondiciones:** Para poder acceder a los cursos es necesario registrase en la página web y haberlos comprado.

**Datos de entrada**: No consta.

**Descripción:** En usuario premium tiene la posibilidad de cursar cualquiera de los cursos comprados por completo.

**Pruebas de aceptación :**

* Se necesita el pago de una cantidad de dinero para el acceso Premium a dicho curso.

**Postcondiciones:** Si es la primera vez que se accede a un curso en la plataforma tienes la posibilidad de realizar un recorrido de tipo tutorial interactivo para explicar la estructura de los cursos y lecciones.

**Errores:**  No consta.

* **Cursar Lección**

|  |  |
| --- | --- |
| **Requerimiento** | RF14 |
| **Nombre del requerimiento** | Cursar Lección |
| **Descripción del requerimiento** | Los cursos están divididos varias lecciones. |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Es necesario para completar un curso.

**Precondiciones:** En caso de los usuarios premium se puede acceder a todas las lecciones de los cursos.

**Datos de entrada:**

* Respuestas en cuestionarios tipo test.
* Respuestas en preguntas con varias opciones.

**Descripción:** Las lecciones son pequeños módulos que componen un curso. Es necesario completar todas las lecciones para dar por finalizado un curso

**Pruebas de aceptación :** No consta.

**Postcondiciones:** Si se ha completado la última lección del curso (y todas las anteriores) se dará por finalizado un curso y obtendrás como premio un badge.

**Errores:**  No consta.

* **Cursar lección - Generar audio**

|  |  |
| --- | --- |
| **Requerimiento** | RF15.1 |
| **Nombre del requerimiento** | Generar audio |
| **Descripción del requerimiento** | Se puede generar un audio de cada lección |
| **Prioridad del requerimiento** | Baja |

**Justificación:** El usuario tiene la posibilidad de generar un audio de la lección que desea.

**Precondiciones:** Hay que estar dentro de un curso, y en una lección para poder generar el audio.

**Datos de entrada:** No consta.

**Descripción:** Otra funcionalidad más del sistema es la posibilidad de generar un audio en base a una lección. De esta forma puede descargarse en formato MP3 y disfrutar del audio fuera de la plataforma.

**Pruebas de aceptación :** No Consta.

**Postcondiciones:** Al generarla se puede descargar y utilizar en cualquier dispositivo multimedia con un reproductor MP3.

**Errores:**  No consta.

* **Cursar lección - Realizar evaluación**

|  |  |
| --- | --- |
| **Requerimiento** | RF15.2 |
| **Nombre del requerimiento** | Realizar evaluación |
| **Descripción del requerimiento** | Se realiza una evaluación cada vez que se avanza |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Para poder avanzar a la siguiente lección o a un curso más superior.

**Precondiciones:** Se tiene que haber llegado al final de una lección para realizar la evaluación.

**Datos de entrada:** No consta.

**Descripción:** Siempre y cuando se realiza una lección al final de cada una de ellas hay una evaluación para dar por finalizada esa parte. En caso de ser la última lección se realizará la evaluación final del curso.

**Pruebas de aceptación :** No consta.

**Postcondiciones:** Si se pasa la evaluación de una lección se puede pasar a la siguiente. Si se para la evaluación del curso se puede acceder a uno superior.

**Errores:**  No consta.

* **Ver streaming**

|  |  |
| --- | --- |
| **Requerimiento** | RF16 |
| **Nombre del requerimiento** | Participar en streaming |
| **Descripción del requerimiento** | Un usuario premium puede asistir a las clases en streaming |
| **Prioridad del requerimiento** | Media |

**Justificación:** Para que el usuario premium pueda acceder a las conferencias o clases ofrecidas por los administradores.

**Precondiciones:** El usuario tiene que ser premium para poder ver el contenido de tipo streaming.

**Datos de entrada:** No consta.

**Descripción:** Ya sea porque se da una clase en streaming de algún curso que el usuario está cursando o porque el usuario acceder a las clases de streaming a través de los anuncios que publicamos en la sección de noticias.

**Pruebas de aceptación :** No consta.

**Postcondiciones:** Se hace mediante un link que lleva al usuario a la plataforma BigBlueButton, y es esa plataforma que se encarga de ofrecer todas las funcionalidades del streaming.

**Errores:**

* En caso de que el link no funcione se le notificará al usuario del fallo y al administrador del sistema.

### Requerimientos Usuario tipo Alumno

Para que el alumno pueda registrarse en nuestra plataforma, la institución a la que pertenece nos enviará el email del alumno. Nosotros seremos los encargados de enviarle un email a cada alumno (proporcionado anteriormente por la institución) con la contraseña y el usuario para acceder a nuestro sistema.

Un usuario de tipo Alumno además de los requisitos que se especifican en este apartado también también se incluyen los requisitos RF1, RF2, RF3 del usuario no registrado y todos los requisitos de un usuario Registrado Free. Asimismo hay que mencionar que un Alumno puede hacer los requisitos de un usuario Registrado Premium, accediendo a los cursos que haya comprado por su cuenta, y a los cursos que el profesor imparte.

* **Acceder a cursos de su aula**

|  |  |
| --- | --- |
| **Requerimiento** | RF17 |
| **Nombre del requerimiento** | Acceder al curso |
| **Descripción del requerimiento** | Acceder a los cursos que el profesor ha publicado en el aula |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Así es como los alumnos van a progresar y aprender dentro de la plataforma

**Precondiciones:** Tiene que ser un usuario dado de alta por el sistema como Alumno y estar dentro de un aula para hacer los cursos.

**Datos de entrada:** No consta.

**Descripción:** Es la tarea necesaria y suficiente para que el Alumno cumpla con lo que son las demandas del Profesor del aula.

**Pruebas de aceptación :**

* El administrador es quien da de alta un usuario de tipo Alumno
* Un Alumno es incluido por un Profesor en el aula

**Postcondiciones:** Cuando el Alumno ya ha empezado a cursar un curso, puede ver su progreso en dicho curso y llegar al final de este.

**Errores:**  No consta.

* **Ver notificaciones del profesor**

|  |  |
| --- | --- |
| **Requerimiento** | RF18 |
| **Nombre del requerimiento** | Ver notificaciones |
| **Descripción del requerimiento** | Ingresar en las notificaciones que el profesor publique |
| **Prioridad del requerimiento** | Media |

**Justificación:** El alumno tiene que estár al tanto con lo que el profesor le pide que realice en los cursos publicados por este

**Precondiciones:** Tiene que ser un usuario dado de alta por el sistema como Alumno y estar dentro de un aula para visualizar las noticias.

**Datos de entrada:** No consta.

**Descripción:** El Alumno tiene la posibilidad de sus notificaciones. En principio esta sección es para que el alumno pueda ver mensajes del profesor y también para ver noticias por parte del sistema

**Pruebas de aceptación :**

* El administrador es quien da de alta un usuario de tipo Alumno
* Un Alumno es incluido por un Profesor en el aula

**Postcondiciones:** Mediante enlaces se puede acceder a cada notificación en concreto.

**Errores:**  No consta.

* **Acceder a clases en streaming del profesor**

|  |  |
| --- | --- |
| **Requerimiento** | RF19 |
| **Nombre del requerimiento** | Ver clase en Streaming |
| **Descripción del requerimiento** | El alumno puede ver las clases en Streaming que el profesor imparte |
| **Prioridad del requerimiento** | Bajo |

**Justificación:** De esta forma un Alumno puede ver una clase en tiempo real impartida por el profesor.

**Precondiciones:** Tiene que ser un usuario dado de alta por el sistema como Alumno y estar dentro de un aula para entrar en las clases en streaming.

**Datos de entrada:** No consta.

**Descripción:** Un profesor puede publicar un anuncio con la hora y la clase que va a impartir. Y los alumnos de la clase podrán ver dicho anuncio y unirse a la presentación en streaming

**Pruebas de aceptación :**

* El administrador es quien da de alta un usuario de tipo Alumno.
* Un Alumno es incluido por un Profesor en el aula.

**Postcondiciones:** Todos los permisos y el software necesario para hacer las clases en streaming será parte de una plataforma externa a la nuestra, en concreto BigBlueButton.

**Errores:**

* En caso de que el enlace no funciona, serán los encargados de la plataforma BigBlueButton de lidiar con ese error.

### Requerimientos Usuario tipo Profesor

El Profesor tendrá también los requisitos RF1, 2 y 3 del usuario no registrado y también puede navegar por el sistema con la posibilidad de hacer todo lo que puede hacer un usuario Registrado Premium.

* **Crear Aula**

|  |  |
| --- | --- |
| **Requerimiento** | RF20 |
| **Nombre del requerimiento** | Crear aula |
| **Descripción del requerimiento** | Un usuario profesor Crear un aula donde puede introducir alumnos |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Para que el profesor pueda agrupar a sus alumnos y compartir el material que el crea ser más conveniente para cada una de las aulas creadas.

**Precondiciones:** Solo un usuario con permisos de profesor podrá crear sus aulas y personalizar sus cursos para compartirlos con los alumnos. El administrador es el que da los permisos de usuario tipo Profesor.

**Datos de entrada:**

* Nombre del aula
* Descripción del aula

**Descripción:** Hay que crear un aula para instruir a alumnos y compartir con ellos el material que el profesor haya creado o escogido dentro de la plataforma.

**Pruebas de aceptación :**

* El administrador debe dar de alta al profesor.

**Postcondiciones:** El aula se crea para introducir los alumnos en ella y así poder tener un seguimiento del progreso de las lecciones que el profesor comparte.

**Errores:**  No consta.

* **Crear Aula - Agregar Alumno**

|  |  |
| --- | --- |
| **Requerimiento** | RF20.1 |
| **Nombre del requerimiento** | Agregar Alumno |
| **Descripción del requerimiento** | El Profesor añade alumnos a la clase que ha creado previamente |
| **Prioridad del requerimiento** | Medio |

**Justificación:** Una clase sin alumnos no sirve de nada. hay que ir añadiendo alumnos a la clase hasta que el aula esté completa.

**Precondiciones:** Se debe de crear la clase para poder añadir alumnos en ella.

**Datos de entrada:**

* Nombre del alumno a introducir o email de esté.

**Descripción:** En crear el aula el Profesor tiene la opción de ir añadiendo alumnos a ella y poder impartir con ellos el material que les corresponde.

**Pruebas de aceptación :**

* Solo el Profesor puede añadir un alumno a su aula.

**Postcondiciones:** En cuanto se han introducido los alumnos a la clase se puede compartir con ellos el material que el profesor cree conveniente y así comenzar a cursar ese material.

**Errores:**  No consta.

* **Crear Aula - Eliminar Alumno**

|  |  |
| --- | --- |
| **Requerimiento** | RF20.2 |
| **Nombre del requerimiento** | Eliminar Alumno |
| **Descripción del requerimiento** | El Profesor borra un alumno de su aula |
| **Prioridad del requerimiento** | Medio |

**Justificación:** En caso de que un usuario ya no forma parte de su clase, por razones externas al sistema, el Profesor puede eliminar dicho alumno del aula. También puede darse el caso de que un profesor introduce un usuario que no es y quiere rectificar el fallo borrando dicho usuario.

**Precondiciones:** El alumno tiene que estar en dicha clase para poder ser eliminado.

**Datos de entrada:** No consta.

**Descripción:** En Profesor tiene una serie de alumnos en su clase y por alguna razón decide eliminar uno de esos alumnos del aula.

**Pruebas de aceptación :**

* Para poder eliminar un alumno de un aula tienes que ser el Profesor que ha creado ese aula.

**Postcondiciones:** Al borrarlo del aula ya no es parte de esta si se vuelve a convertir en un usuario registrado de la web.

**Errores:**  No consta.

* **Crear Aula - Ver progreso**

|  |  |
| --- | --- |
| **Requerimiento** | RF20.3 |
| **Nombre del requerimiento** | Ver progreso(aula) |
| **Descripción del requerimiento** | El Profesor ve el progreso de las lecciones publicadas de los alumnos |
| **Prioridad del requerimiento** | Medio |

**Justificación:** De esta forma el Profesor puede ver si los alumnos están haciendo lo que el profesor les está mandando.

**Precondiciones:** Tiene que haber al menos un alumno en la clase para poder ver el progreso de el/ellos.

**Datos de entrada:** No consta.

**Descripción:** Después de crear la clase con sus alumnos y publicar el contenido el profesor tiene la posibilidad de ver el progreso de sus alumnos en cada una de las lecciones que tiene publicadas en ese aula. De esta forma tiene un seguimiento de todos sus alumnos.

**Pruebas de aceptación :**

* Solo el profesor puede ver el progreso de su aula.

**Postcondiciones:** Si resulta que un alumno no está al día con lo que el profesor le ha pedido este puede tomar medidas al respecto.

**Errores:**  No consta.

* **Crear Aula - Asignar curso**

|  |  |
| --- | --- |
| **Requerimiento** | RF20.4 |
| **Nombre del requerimiento** | Asignar curso(al aula) |
| **Descripción del requerimiento** | El profesor podrá asignar el curso que desee a cualquiera de sus aulas. |
| **Prioridad del requerimiento** | Medio |

**Justificación:** El profesor podrá relacionar cada una de sus aulas a los cursos que el vea conveniente realizar.

**Precondiciones:** Tiene que estar creada el aula para poder asignar un curso a ella.

**Datos de entrada:**

* El aula a la que se le va a asignar el curso(aulas del profesor).

**Descripción:** Con el aula creada y todos los alumnos introducidos dentro de ella solo falta asignarles un curso y empezar a disfrutar de la plataforma aprendiendo. Este curso puede que se le asigna al aula puede ser uno ya creado del sistema o partir de uno personalizarlo por el profesor. Un aula puede tener asignada varios cursos.

**Pruebas de aceptación :**

* Solo el profesor puede asignar cursos a su aula.

**Postcondiciones:** Al asignar el curso al aula, los alumnos de esta pueden empezar a cursar ese material publicado.

**Errores:**  En caso de que el curso asignado al aula no sea del gusto del profesor puede borrarlo o cambiarlo a posteriori.

* **Personalizar curso**

|  |  |
| --- | --- |
| **Requerimiento** | RF21 |
| **Nombre del requerimiento** | Personalizar curso |
| **Descripción del requerimiento** | Se parte de un curso ya publicado en la web que se puede personalizar(con nuestro material) |
| **Prioridad del requerimiento** | Alto |

**Justificación:** Si el profesor ve un curso en nuestra plataforma que le gusta pero no por completo puede decidir adaptarlo a su gusto.

**Precondiciones:** Tiene que haber un curso creado para poder modificarlo.

**Datos de entrada:** No consta.

**Descripción:** El profesor decide añadir un curso a su clase. Escoge uno pero se da cuenta de que hay lecciones y/o ejercicios dentro de cada unidad que no están a su gusto por lo que decide personalizar dichos apartados. Solo se pueden eliminar o cambiar de orden las unidades y los ejercicios de cada unidad (que también ofrecemos nosotros).

**Pruebas de aceptación :**

* Solo un usuario de tipo profesor puede personalizar cursos.

**Postcondiciones:** Al acabar la personalización del curso se puede publicar en el aula del Profesor.

**Errores:**  No consta.

* **Acceder a curso tutorial**

|  |  |
| --- | --- |
| **Requerimiento** | RF21.2 |
| **Nombre del requerimiento** | Acceder a curso tutorial |
| **Descripción del requerimiento** | Los profesores podrán acceder a cursos orientativos sobre el uso optimizado de nuestra aplicación. |
| **Prioridad del requerimiento** | Medio |

**Justificación:** Para mejorar la experiencia de los profesores dentro de nuestra página, es necesario aprender a utilizarla adecuadamente.

**Precondiciones:** El profesor debe de haber sido de alta dentro del sistema..

**Datos de entrada:** No consta.

**Descripción:** Los profesores tendrán a su disposición unos minicursos que le ofrecerán la posibilidad de aprender a utilizar nuestra aplicación lo mejor posible, en ellos aprenderá a realizar las siguientes tareas:

* Personalizar sus cursos teniendo en cuenta aspectos como: las valoraciones de otros usuarios con respecto al material de nuestra página, la temática de cada uno de ellos, cuales son los más utilizados…
* Como crear un curso de forma estructurada para que el alumno pueda aprender de forma progresiva el contenido del mismo, esto servirá sobretodo a la hora de crear un plan de estudios adecuado.
* Controlar el progreso de los alumnos de cada aula dentro de cada curso. Esto será útil principalmente a profesores sustitutos que necesiten saber de la forma más rápida posible, qué es lo que cada aula ha aprendido hasta la fecha.
* Organizar las distintas aulas adaptándolas a las necesidades de los alumnos. Dentro de una misma clase no todo el mundo sigue un mismo ritmo, con este tutorial el profesor aprenderá a dividir la clase en distintas aulas y adaptar los contenidos de cada una de ellas a las necesidades de los alumnos que las compongan.
* Aprenderá a gestionar nuestra herramienta de autoevaluación. De esta forma podrá dejar en manos del sistema la evaluación de los alumnos, consiguiendo tiempo extra para desempeñar tareas más importantes.

**Pruebas de aceptación :**

* Solo un usuario de tipo profesor puede accedera estos tutoriales.

**Postcondiciones:** Tras completar estos tutoriales el profesor podrá hacer un uso más optimizado de nuestro sistema.

**Errores:**  No consta.

* **Eliminar Curso**

|  |  |
| --- | --- |
| **Requerimiento** | RF22 |
| **Nombre del requerimiento** | Eliminar Curso |
| **Descripción del requerimiento** | El profesor decide eliminar un curso que ha publicado en su aula anteriormente. |
| **Prioridad del requerimiento** | Medio |

**Justificación:** Ya sea porque el Profesor ya no quiere ese curso para sus alumnos o estos alumnos han finalizado dicho curso.

**Precondiciones:** Tiene que haber un curso publicado para poder eliminarlo de esta

**Datos de entrada:** No consta.

**Descripción:** El Profesor tiene una serie de cursos publicados en su aula. Se da cuenta de que ese curso en realidad no cumple con el temario que quiere impartir a sus alumnos y lo elimina. Por otro lado puede que los alumnos hayan finalizado todo el contenido y el Profesor decida deshacerse de el.

**Pruebas de aceptación :**

* Solo el profesor puede eliminar un curso.

**Postcondiciones:** Al eliminar dicho curso, sea personalizado o no, ya no está disponible para los alumnos de la clase.

**Errores:**  No consta.

* **Eliminar Aula**

|  |  |
| --- | --- |
| **Requerimiento** | RF23 |
| **Nombre del requerimiento** | Eliminar Aula |
| **Descripción del requerimiento** | Se borra el aula del Profesor |
| **Prioridad del requerimiento** | Alto |

**Justificación:** El curso se ha acabado y el Profesor decide eliminar dicha aula que ya no necesita.

**Precondiciones:** Tiene que haber un un aula creada para poder borrarla.

**Datos de entrada:** El aula a borrar.

**Descripción:** Un profesor en cualquier momento por las razones que sea, puede ser porque el curso se ha acabado o ya no es profesor de dicha clase, puede decidir eliminar un aula. Al eliminar un aula esta se borra con todos los alumnos de ella. Lo único que se quedará son los cursos personalizados en caso de que los tenga.

**Pruebas de aceptación :**

* Solo el profesor puede eliminar un aula.

**Postcondiciones:** Al eliminar un aula, el profesor ya no tiene acceso a esos alumnos ni a los progresos de ellos.

**Errores:**  No consta.

* **Publicar Anuncio**

|  |  |
| --- | --- |
| **Requerimiento** | RF24 |
| **Nombre del requerimiento** | Publicar anuncio |
| **Descripción del requerimiento** | El Profesor puede publicar anuncios para los alumnos de un aula en específico. |
| **Prioridad del requerimiento** | Medio |

**Justificación:** Va a ser la manera mediante la cual es Profesor se comunicará con los alumnos dentro del sistema.

**Precondiciones:** El profesor debe tener una clase creada con alumnos dentro para poder publicar un anuncio en la sección de Notificaciones de dicha aula.

**Datos de entrada:**

* Fecha y hora del evento.
* Situación geográfica del evento.
* Noticia en cuestión.

**Descripción:** Mediante los anuncios el Profesor puede comunicar a sus alumnos la tarea que estos deben hacer o cualquier información que este considere oportuna. También tiene la posibilidad de publicar un anuncio con la clase online que vaya a impartir.

**Pruebas de aceptación :**

* Es el usuario de tipo Profesor quién puede publicar anuncios.

**Postcondiciones:** Al publicar un anuncio esté para a ser visto por todos los integrantes del aula en donde lo ha publicado.

**Errores:**  No consta.

* **Ver cursos**

|  |  |
| --- | --- |
| **Requerimiento** | RF25 |
| **Nombre del requerimiento** | Ver cursos |
| **Descripción del requerimiento** | Un profesor podrá visualizar cualquiera de sus cursos activos |
| **Prioridad del requerimiento** | Alto |

**Justificación:** El profesor necesita estar al tanto de los cursos que tiene activos.

**Precondiciones:** Tener como mínimo un curso creado.

**Datos de entrada:** No consta.

**Descripción:** Cualquier profesor podrá ver todos los cursos que tiene activos, ya sea para revisarlos, modificarlos o incluso eliminarlos.

**Pruebas de aceptación :**

* Cada profesor podrá visualizar solo sus propios cursos o los genéricos ofrecidos por el sistema.

**Postcondiciones:** Una vez visualizado el curso tendrá la posibilidad de modificar el curso.

**Errores:**  No consta.

### Requerimientos para Administrador

* **Gestionar noticias**

|  |  |
| --- | --- |
| **Requerimiento** | RF26 |
| **Nombre del requerimiento** | Gestionar noticias |
| **Descripción del requerimiento** | El administrador podrá incluir un boletín de noticias en la plataforma. |
| **Prioridad del requerimiento** | Medio |

**Justificación:** Para promover la participación de los usuarios en eventos concertados.

**Precondiciones:** Formar parte del grupo administradores.

**Datos de entrada:**

* Fecha y hora del evento.
* Situación geográfica del evento.
* Noticia en cuestión.

**Descripción:** El administrador será el encargado de anunciar noticias o eventos globales en la página web, así como, información sobre eventos próximos que puedan ser relevantes en el ámbito de la programación o incluso concursos como **hackathones.**

**Pruebas de aceptación:** No consta.

**Postcondiciones:** La noticia será publicada en su correspondiente apartado y todos los usuarios que visiten la web podrán visualizarla.

**Errores:**  No consta.

* **Gestionar usuarios**

|  |  |
| --- | --- |
| **Requerimiento** | RF27 |
| **Nombre del requerimiento** | Gestionar usuarios |
| **Descripción del requerimiento** | El administrador tendrá acceso a la base de datos y podrá gestionar a los usuarios |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Para tener un control de los usuarios, es necesario que el administrador tenga acceso a la base de datos para gestionarlos.

**Precondiciones:** Ser administrador y conocer el funcionamiento de la base de datos.

**Datos de entrada:**

* Datos necesarios para iniciar correctamente la base de datos.

**Descripción:** Cualquier administrador tendrá control total sobre cualquier usuario (agregar, modificar, eliminar), de manera que si cualquier usuario infringe las normas de la web, será sancionado.

**Pruebas de aceptación:** Los datos introducidos en la base de datos sean correctos.

**Postcondiciones:** Un mensaje en pantalla aparecerá indicando la acción que ha realizado el administrador y se le enviará una notificación al usuario indicando la acción que ha realizado el administrador sobre ese usuario.

**Errores:**

* Tiempo de espera agotado.
* Datos inicio sesión base de datos incorrectos.
* Si un administrador intenta eliminar a otro administrador.
* **Gestionar cursos**

|  |  |
| --- | --- |
| **Requerimiento** | RF28 |
| **Nombre del requerimiento** | Gestionar cursos |
| **Descripción del requerimiento** | El administrador tendrá acceso a todos los cursos y podrá modificarlos. |
| **Prioridad del requerimiento** | Alta |

**Justificación:** Para la modificación de cursos activos, la introducción o eliminación de los mismos.

**Precondiciones:** Ser administrador.

**Datos de entrada:**

* En el caso de querer introducir un nuevo curso:
  + Nombre del curso.
  + Lecciones del curso.

**Descripción:** Un administrador podrá en cualquier momento introducir, modificar o eliminar cualquier curso que se encuentre en la web.

**Pruebas de aceptación:** El curso deberá aprobar las pruebas de calidad, antes de ser introducido o modificado. Además, antes de modificar, eliminar o añadir cualquier curso, la página estará offline mientras se modifique cualquiera de los cursos.

**Postcondiciones:** Si se ha introducido un nuevo curso, la lista de cursos se actualizará y aparecerá el nuevo curso, si se ha modificado aparecerá una breve descripción indicando los cambios, y si, el curso es eliminado desaparecerá de la lista de cursos disponibles.

**Errores:**  No consta.

* **Gestionar preguntas**

|  |  |
| --- | --- |
| **Requerimiento** | RF29 |
| **Nombre del requerimiento** | Gestionar preguntas |
| **Descripción del requerimiento** | El administrador podrá gestionar (responder, eliminar) las preguntas recibidas |
| **Prioridad del requerimiento** | Media |

**Justificación:** Debe haber un control de las preguntas enviadas por los usuarios.

**Precondiciones:** Existir preguntas activas y ser administrador.

**Datos de entrada:**

* Respuesta a pregunta realizada.

**Descripción:** Un administrador debe responder las preguntas realizadas por los usuarios de manera que se haga notar que hay un equipo detrás de la web y que se preocupa por sus usuarios (clientes).

**Pruebas de aceptación:** No consta.

**Postcondiciones:** La respuesta se le enviará al usuario que la formuló, y se le preguntará si se ha resuelto su problema para reflejar en las estadísticas de la web.

**Errores:**  No consta.

## Requerimientos no funcionales.

### Integridad

La integridad representa la seguridad de que nadie puede modificar datos a los que no está autorizado a acceder dentro de nuestro sistema. La información del perfil y de los cursos a los que esté inscrito un usuario, solamente podrán ser modificados por él y nadie más.

Por otro lado solo un usuario de tipo Profesor tiene la posibilidad de personalizar un curso y enlazarlo con las aulas (grupo de alumnos) que haya creado. A la hora de personalizar el curso, el profesor partirá de uno creado a priori por nosotros, el profesor simplemente elegirá qué lecciones quiere dar, en qué orden lo hará y que ejercicios tendrá cada una de ellas. Los ejercicios que el profesor usará para sus lecciones serán los que estén almacenados en nuestro repertorio, por ejemplo, en una lección hay tres ejercicios pero el profesor ve otro dentro de nuestra lista de materiales que le interesa más que los que ya están incluidos en la lección, el profesor podrá sustituir o añadir este ejercicio dentro de ella. En caso de que el profesor desee añadir uno hecho por él mismo, deberá enviarnos una solicitud para ello, los administradores se encargan de validarlo y subirlo al listado de ejercicios. Cualquier usuario es capaz de solicitar la inclusión de un nuevo ejercicio creado por el mismo.

Solo los administradores y gestores del sistema tienen pueden realizar cambios dentro de la web.

### Fiabilidad

Nuestro sistema será fiable porque tenemos en cuenta la recuperación de datos frente a posibles fallos en el sistema, ya que aseguramos que no se pierden los datos de perfil de usuario o el progreso de los cursos que a los que está registrado. Esto también incluye que no se perderá nada en la comunicación del cliente con el servidor o viceversa.

Asimismo vamos a tener copias de seguridad del material disponible y de toda la información de los usuarios tanto en la nube como en la base de datos. También dispondremos de otra copia en dispositivos físicos personales de la empresa. Tomamos estas precauciones para tener segura una recuperación de los datos en caso de un fallo del sistema nuestro o de terceros. También haremos uso de esclavos, los cuales nos permitirán mantener el sistema en funcionamiento en caso de que este se caiga de forma repentina.

### Disponibilidad

El servidor y las bases de datos van a estar en línea las 24 horas del día para proporcionar su uso constante a cualquier usuario y en cualquier momento cada vez que lo necesite. Para tener una alta disponibilidad vamos a tener en caso de fallo del sistema servidores duplicados de forma que si cae uno el otro entra en marcha. Todo usuario registrado va a tener acceso a las funcionalidades de la plataforma en cualquier momento.

### Mantenimiento

Del mantenimiento del contenido y de la gestión de los usuarios se encargan los administradores (gestores). Los cuales se encargan de publicar, actualizar y, si hace falta, borrar material en la web. También se preocupan de que todos los usuarios cumplan con los términos y condiciones de la web. Y por último se encargan de filtrar los ejercicios que los usuarios nos mandan y decidir cuales cumplen las características para estar en el sistema y cuales no se pueden utilizar.

Los desarrolladores y diseñadores van a ser los encargados del mantenimiento de toda la parte del Front-End que no realiza el Administrador. Serán los responsables de hacer cambios en la interfaz de la web, hacer cambios en la estructura e introducir nuevas funcionalidades a la plataforma.

Los desarrolladores Back-End son los encargados del mantenimiento en la parte del servidor. Van a ser los que se ocupan del correcto funcionamiento de los servidores, el mantenimiento de las bases de datos y de todos los aspectos relacionados con la seguridad del sistema.

### Rendimiento

Debido a que nuestro sistema no va a realizar cálculos complejos, responderá en un tiempo por debajo de los 2 segundos a cualquier tarea realizada por el usuario. Las tareas que más se solicitarán al sistema son las interacciones con la base de datos y la representación de los ejercicios que contienen gráficos avanzados. Para estos últimos dos es posible que con una baja velocidad de internet se aumenten los tiempos de espera.

### Privacidad

Los datos de todos los usuarios de Programming Cloud que están almacenados en las BBDD son de carácter personal. Estos datos serán privados y no se utilizarán con ningún fin fuera de la plataforma. También hay que mencionar que si un usuario se da de baja se guardarán todos los datos del sistema durante el tiempo que se estipula legalmente.

Los cursos personalizados por un profesor, solamente serán accesibles por los alumnos que estén incluidos dentro de las aulas que él mismo haya creado. A su vez, sólo él podrá hacer cambios dentro de estas aulas o cursos.

Los ejercicios creados por un usuario que sean publicados en la web, pasarán a formar parte del contenido de esta, ergo tendremos autorización para usarlos en los cursos que veamos conveniente de forma indefinida.

### Autentificación

A la hora de registrarse dentro de nuestra base de datos, el usuario deberá validar su correo electrónico anteriormente.

Utilizaremos el clásico sistema de usuario - contraseña para iniciar sesión en nuestra web, la cuenta podrá estar ligada a páginas como facebook o google+ mediante oauth, de esta manera dejaríamos esta tarea de autenticación en manos de terceros.

En caso de que el usuario desee recordar su contraseña, este deberá acertar la palabra secreta que especificó a la hora de registrarse en la web.

Para los pagos utilizaremos un TPV con el que nos comunicaremos con una entidad bancaria, la cual se encargará de verificar el estado de la tarjeta de crédito del usuario y de manejar el cobro de la compra. Usaremos TLS para cifrar los datos que enviaremos al TPV antes de realizar el pago (nuestra identificación y cuánto debe pagar el usuario).

Para evitar que programas informáticos creen cuentas dentro de nuestra web, utilizaremos el sistema [*No Captcha ReCaptcha.*](http://www.bbc.com/mundo/noticias/2014/12/141208_tecnologia_google_captcha_recaptcha_robot_humano_ig)

### Autorización

Los datos de todos los usuarios serán personales y sólo podrán ser modificados dichos datos por personas autorizadas para ello. Los perfiles de usuario dentro de nuestro sistema serán los siguientes: Usuario invitado, registrado free, registrado premium, profesor, alumno y administrador.

Un usuario invitado será aquel que visite nuestra página web sin que esté registrado. Puede ver los cursos que hay disponibles en la plataforma, la sección de noticias e ponerse en contacto con nosotros.

Para acceder a la parte privada del sistema el usuario tiene que introducir un usuario y una contraseña. Este usuario podrá crear un perfil, aportar ejercicios a la comunidad y acceder a las lecciones gratuitas de nuestros cursos.

Un usuario se convertirá en premium bien al comprar uno de los cursos o bien al comprar un pack de estos. En caso de comprar un curso, solamente tendrá acceso a las lecciones del mismo, el pago de estos se hará de forma mensual, trimestral, semestral o anual. Aplicaremos un descuento en los packs en función de la cantidad de cursos de los que está compuesto, cuanto más cursos tenga más barato será. Todo usuario premium podrá acceder a los streamings organizados dentro de la web.

Un profesor solamente podrá acceder a los cursos que su institución haya solicitado, sin embargo, podrá personalizar dichos cursos con cualquier ejercicio de los que estén subidos en el servidor. Cada profesor podrá organizar distintas aulas formadas por sus alumnos, estas aulas sólo pueden ser creadas y gestionadas por un usuario tipo profesor.

Un usuario del tipo alumno podrá acceder tanto a los cursos vinculados al aula de la que forma parte, como a los que él mismo desee comprar. Un usuario solamente puede llegar a convertirse en alumno siendo autorizado por un gestor, será la propia institución quien se pondrá en contacto con nosotros para dar de alta a los alumnos que necesite.

El administrador o gestor tiene acceso a todas las operaciones de altas, bajas,consultas y modificaciones de la aplicación. Asimismo puede hacer el rol de un usuario registrado y cambiar parámetros y contenido de la plataforma.

### Contabilidad

Vamos a monitorizar todos los datos que nos interesan y registrarlos para saber en todo momento quién, cómo y de qué manera se está utilizando nuestro sistema y cómo se comporta este.

Para ello utilizaremos la herramienta SEASLOG, que nos permitirá recopilar información en caso de fallos de personas (admin o usuarios) o de interfaces, caídas del servidor, temas de hardware, etc.

Por otro lado analizaremos el comportamiento de todos aquellos que acceden a nuestra web. Las principales métricas que vamos a estudiar de los usuarios no registrados son las siguientes: visitas, visitas exclusivas, tiempo medio de visita, tasa de rebote y la tasa de conversión de los usuarios (free y premium).

Para los usuarios no registrados vamos a tener otra serie de métricas: tiempo medio por curso y unidad de curso, tiempo medio en la aplicación, tasa de abandono de los cursos y análisis de partes específicas en donde el usuario tenga más dificultad de realizar. El feedback de estas métricas van a servir para mejorar de forma constante la plataforma haciendo que sea más usable y accesible para nuestros usuarios. Para el análisis y registro de estos datos utilizamos las herramientas Google Analytics y Klipfolio.

También analizaremos cuales son los ejercicios más empleados por los profesores a la hora de realizar ciertos cursos, a partir de estos datos podemos crear nuevo material dentro de la página que sea más acorde a la demanda de estos.

### Usabilidad

Representa la facilidad de uso y aprendizaje de nuestro sistema. En cuanto un usuario se registra y accede por primera vez a la página este será recibido por un tutorial interactivo para aprender donde están las funcionalidades más importantes de la página y cómo utilizarlas. Así todo nuevo usuario no tendrá el más mínimo problema en utilizar el sistema. También se puede omitir el tutorial si se desea.

Tenemos también otro tutorial interactivo de presentación que se le aparecerá al usuario al iniciar por primera vez un curso en nuestra web. El objetivo de este tutorial es presentar al usuario cual es la forma en las que se desarrollaran las lecciones, que componen cada lección y cada unidad y asegurarle que cumplirá el objetivo propuesto al registrarse en nuestra web; aprender de forma fácil, rápida e interactiva. Por supuesto este tutorial también incluye a los profesores, pues convendría explicarles cómo podrán personalizar sus cursos mediante el material que le proporcionamos.

### Accesibilidad

La utilización de la plataforma Programming Cloud se puede hacer por todo usuario con un mínimo conocimiento y experiencia de navegabilidad web. Además al inciar sesión por primera vez se da la posibilidad al usuario de realizar un tutorial interactivo que presenta al la forma de acceder a todas las herramientas disponibles para el usuario y donde se encuentran situadas. Además al iniciar por primera vez un curso también se tiene la oportunidad de realizar una presentación de la estructuras de los cursos y cada una de sus unidades.

### Portabilidad

Nuestra plataforma se podrá ejecutar desde diferentes terminales con diferentes sistemas operativos. Así mismo también se podrá acceder a la plataforma desde cualquier dispositivo móvil o tablet, ya que la web presenta un diseño responsive que se adapta al tamaño de la pantalla. Aquellos ejercicios con representación gráfica que puedan requerir interacción no se podrán realizar desde los dispositivos móviles o tablets.

Se pueden presentar leves cambios del HTML cuando cambiamos desde un navegador a otro. Pero que no afectan a disposición de los elementos de la web ni al funcionamiento de la misma.

### De Difusión

Nuestra web va a ser visible desde cualquier dispositivo electrónico, ya sea ordenador, móvil o tablet, ya que tendrá un diseño responsive. Además para complementar y tener un uso más personalizable en los dispositivos móviles (Android) vamos a crear una aplicación (.apk) donde se podrá visualizar el contenido de la plataforma de una manera más reducida y limpia.

Utilizamos APIs de distintas redes sociales con el objetivo de facilitar el registro dentro de nuestra plataforma. Las APIs que vamos a utilizar para el Inicio de Sesión van a ser las de Google y Facebook.

Vamos a estar presentes en las principales redes sociales de hoy en día: Facebook, Twitter y YouTube. En Facebook y Twitter mantendremos a nuestros seguidores informados sobre las actualizaciones de nuestro sistema y noticias lenguajes de programación, hackaton’s, educación y más. Por otra parte en Youtube vamos a tener vídeos promocionales de la plataforma y vídeos sobre nuestra empresa (Braineering) y nuestra web(Programming Cloud).

Trabajaremos con la conversión de formatos PDFs, ePub,... a formatos audio, preferiblemente MP3 para poner a disponibilidad de nuestros usuarios un formato multimedia más para disfrutar de los contenidos de la plataforma.

El contenido presente en la plataforma, la interfaz de usuario y la arquitectura del sistema son fruto de las mentes del equipo Braineering, por lo que nos corresponden todos los derechos de autor de los mismos. Crearemos estrategias para proteger todo contenido de propiedad intelectual de la web con el objetivo de no poner en riesgo los derechos de autor. Las principales medidas van a ser: registrar el nombre del dominio y la marca, patentar el formato de nuestras lecciones y tener discreción con los secretos de nuestro sistema.

* **Tecnología**

Nuestro servidor funcionará mediante Apache, javascript y php. Estudiamos la utilización de Node.js en un futuro.

Usaremos una base de datos relacional SQL.

Utilizaremos esclavos y discos duros extraíbles para el almacenamiento de copias de seguridad del contenido de nuestra página.

Emplearemos SEASLOG para manejar la contabilidad de nuestra web.

Para el tema de gráficos 3D, utilizaremos WebGL.

Utilizaremos OAUTH para relacionar las cuentas de Facebook y Google+ con los perfiles de los usuarios.

Los videos de nuestra plataforma estarán disponibles en Youtube. Los streamings se harán mediante la herramienta Big Blue Button.

Utilizaremos comunicación REST y https.

Seguiremos el estándar IEEE LOM para catalogar nuestros materiales educativos.

Usaremos el estándar de empaquetamiento SCORM, por lo tanto deberemos adaptar nuestra base de datos para almacenar ficheros de este estilo. A su vez utilizaremos Reload como editor de SCORM.

Para gestionar nuestra biblioteca digital utilizaremos Greenstone o DSpace.

Emplearemos xPerience para poder obtener fácilmente las estadísticas que necesitemos obtener de nuestros ejercicios.

También haremos uso de MERLOT para buscar materiales educativos en bases de datos ya creadas por otras instituciones.

# Apéndices

## Definiciones, acrónimos y abreviaturas

**API** - La Interfaz de Programación de Aplicaciones representan el subconjunto de rutinas funciones y procedimientos que ofrece cierta biblioteca de software.

**Software** - Representa el conjunto de programas y rutinas que permite la realización de las tareas.

**Hardware** - Son los conjuntos de elementos físicos y materiales que forman un sistema informático.

**Redes Sociales** - Representan comunidades de personas con intereses o actividades en común.

**HTML** - Es un lenguaje de programación que se utiliza en el desarrollo de ´paginas web.

**SIstema Operativo** - Representa el conjunto de órdenes y programas que controlan los procesos básicos de una computadora y permite el funcionamiento de otros programas dentro de este.

**Back-End -** El back-end es la parte que procesa la entrada desde el front-end. Representa toda la parte del funcionamiento del sistema.

**Front-End** - La parte del Front-End dentro del diseño del software representa la parte frontal del sistema que interactúa con el usuario.

**Hackathon** - Una hackaton representa un encuentro de programadores cuyo objetivo es el desarrollo de software.

**Streaming** - DIstribución digital de multimedia a través de una red de computadoras de manera que el usuario consume el producto en tiempo real.

**E-Learning** - Una plataforma e-learning es un espacio virtual de aprendizaje orientado a facilitar la capacitación online.

**JavaScript** - Es un lenguaje de programación orientado a objetos basados en prototipos, imperativo, débilmente tipado y dinámico.

**PHP -** Es un lenguaje de código abierto adecuado para el desarrollo web que puede ser incrustado en HTML.

**WebGL** - Permite mostrar gráficos en 3D en navegadores web. El WebGl permite mostrar gráficos en 3D acelerados por hardware.

**SQL -** Es un lenguaje declarativo de acceso a base de datos relacionales que permite especificar diversos tipos de operaciones en ella.

**Cloud** - El cloud es un paradigma que ofrece servicios de computación a tráves de internet.

**Ekomi -**  The Feedback Company, ha ayudado a más de 10.000 empresas a aumentar el tráfico online y las ventas, mediante la integración en su propia Web del sello eKomi, mostrando así al resto de usuarios las opiniones de clientes reales satisfechos. Una manera sencilla y práctica de potenciar la confianza, seguridad y transparencia de sus transacciones al mismo tiempo que reduce sus costes en marketing.

**Node.Js** - Es un intérprete JavaScript del lado del servidor que permite una gestión y personalización del servidor.

## Referencias

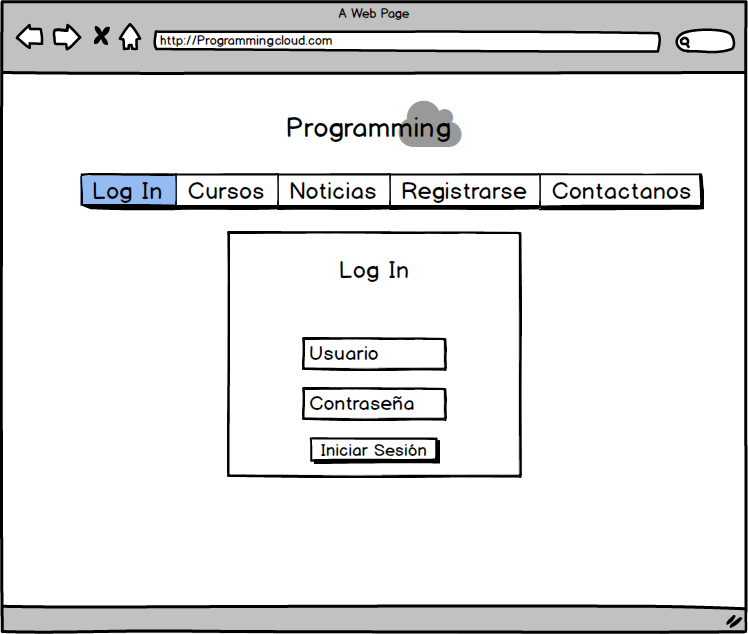
**Documento IEEE 830 :** <http://www.fdi.ucm.es/profesor/gmendez/docs/is0809/ieee830.pdf>

## Mockups del sistema

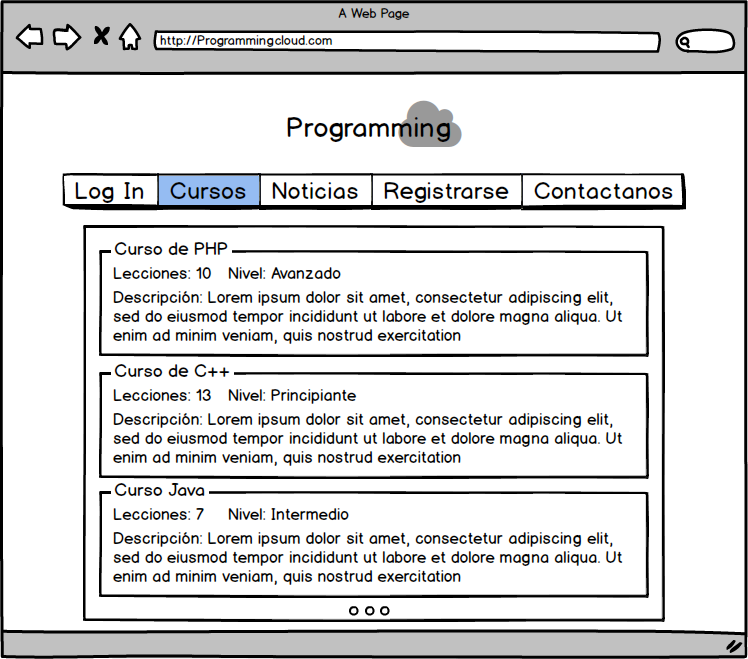
A continuación vamos a presentar los diseños de baja fidelidad de interfaces mediante mockups. Estos son prototipos que representan las funcionalidades más básicas de nuestro sistema.

### Usuario Invitado ([RF para Usuario invitado](#_xth5n9om1xbz))

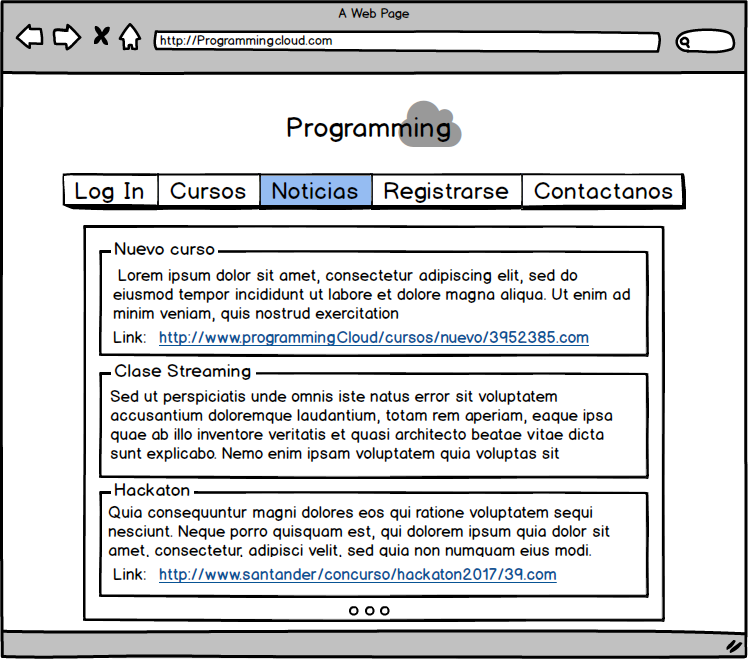
* Login
  + Está relacionado con el RF6



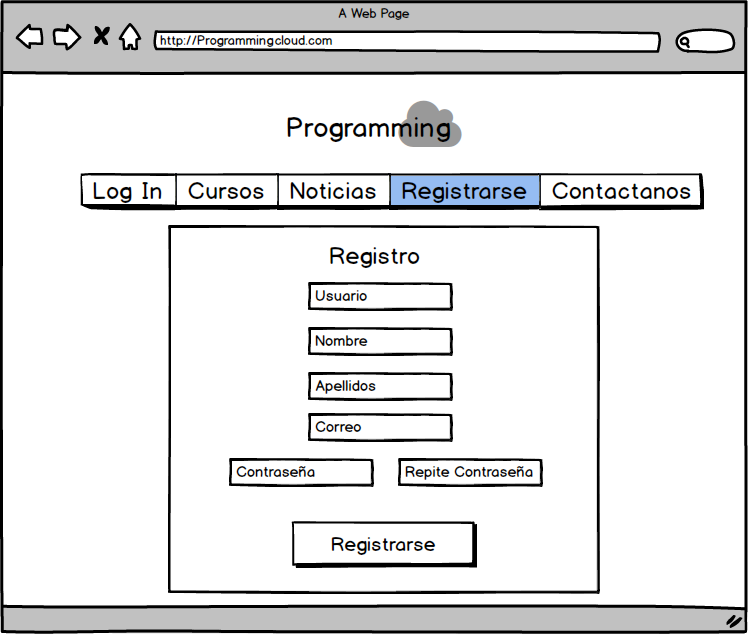
* Ver Cursos Disponibles
  + Está relacionado con el RF2



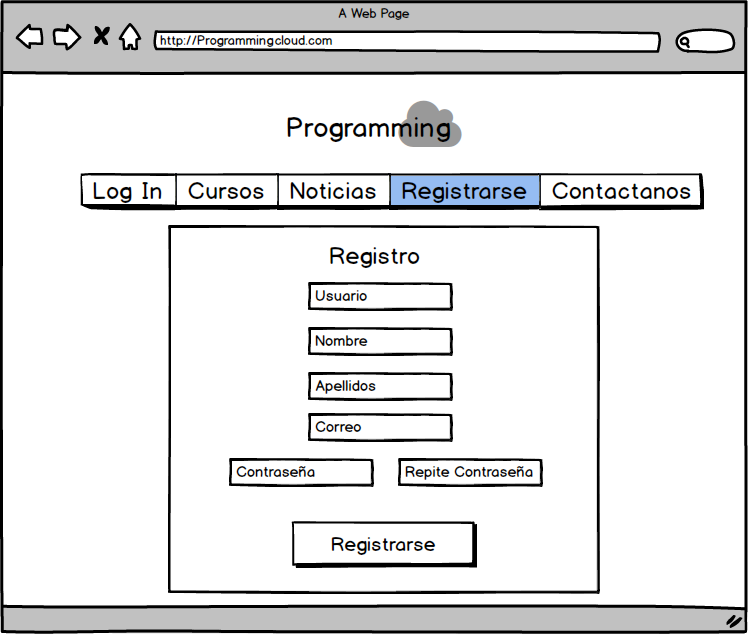
* Noticias
  + Está relacionado con el RF1



* Registrarse
  + Está relacionado con el RF4

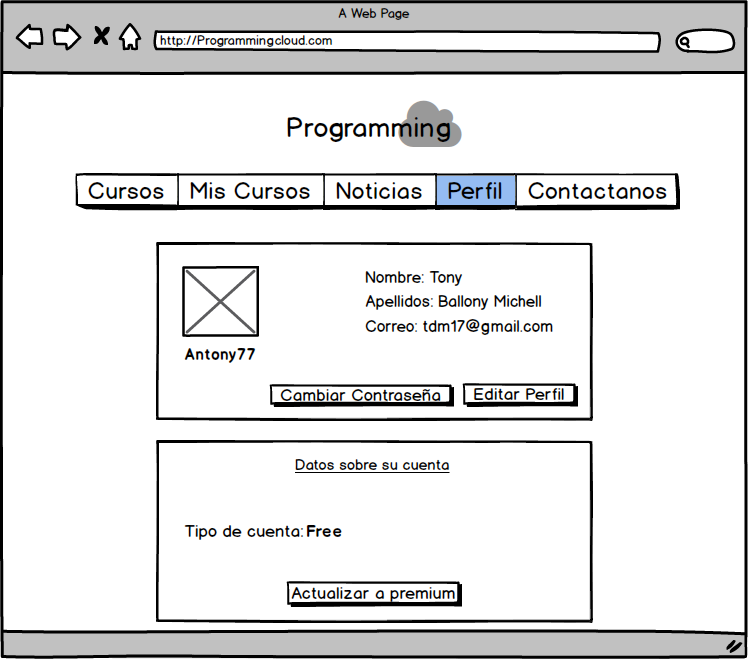


* Contactanos
  + Está relacionado con el RF3

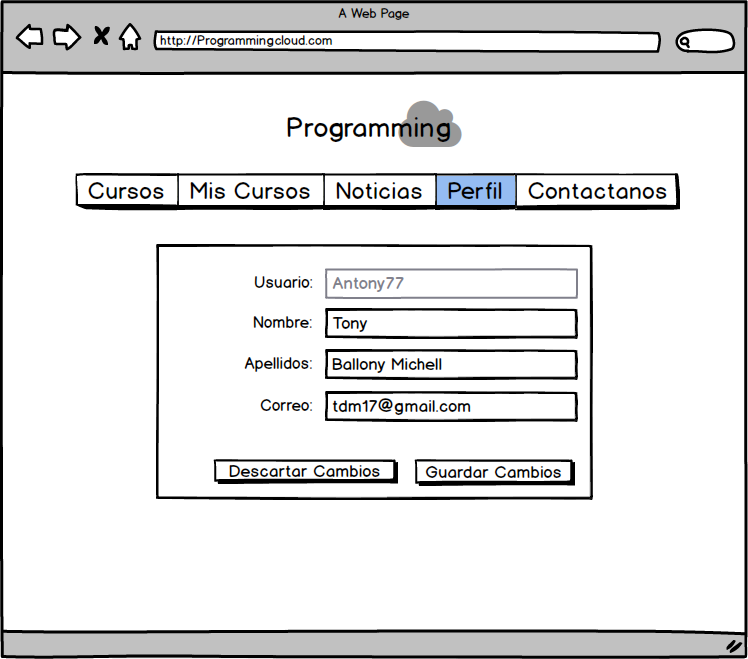


### Usuario Registrado Free ([RF Usuario Registrado (Free)](#_do3zjq1jehgc))

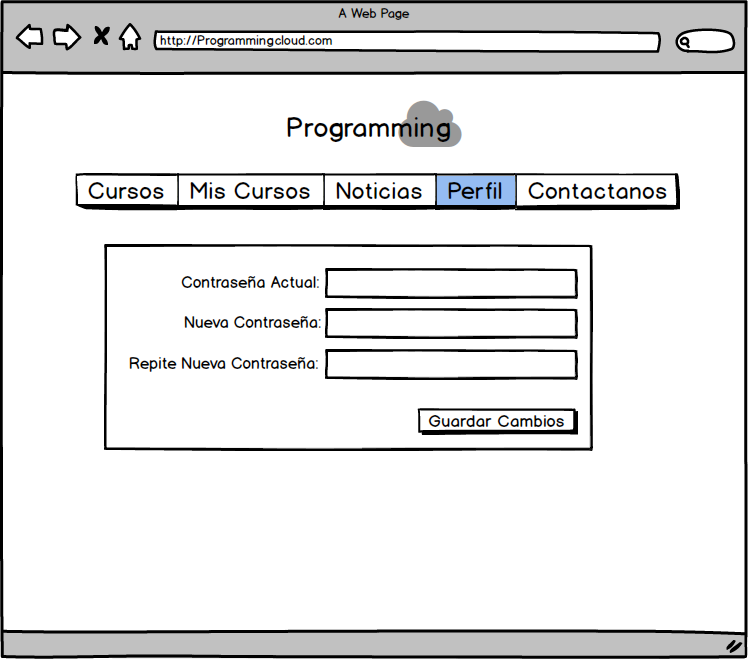
* Ver Perfil
  + Está relacionado con el RF10



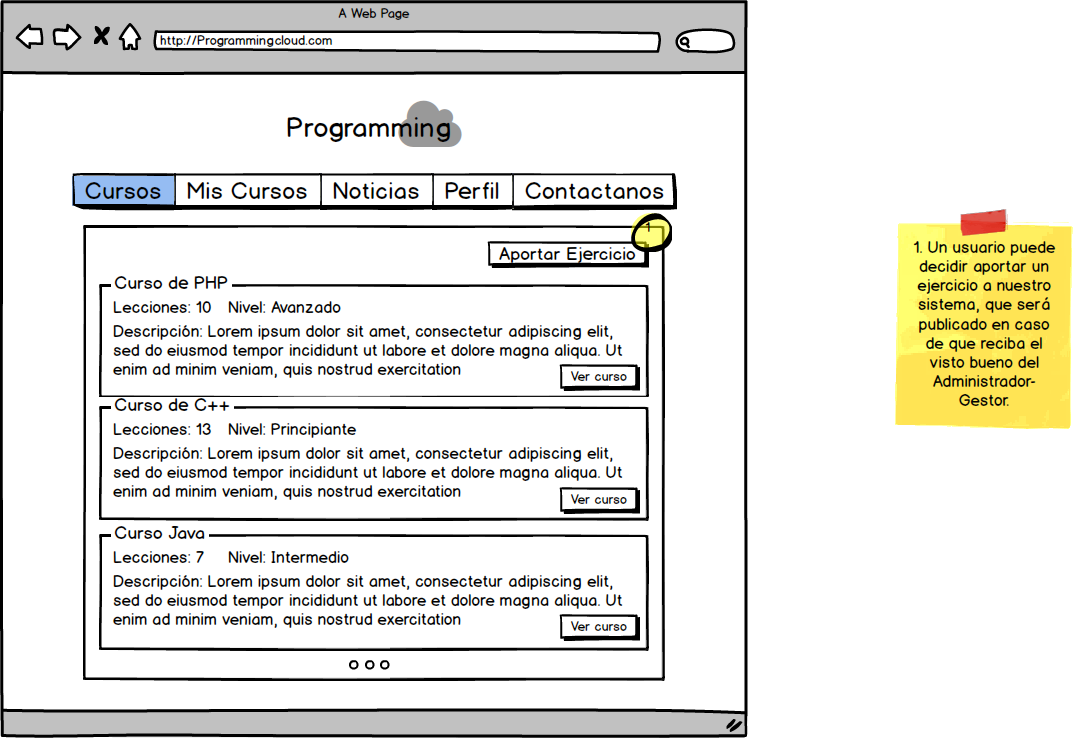
* Editar Perfil
  + Está relacionado con el RF11



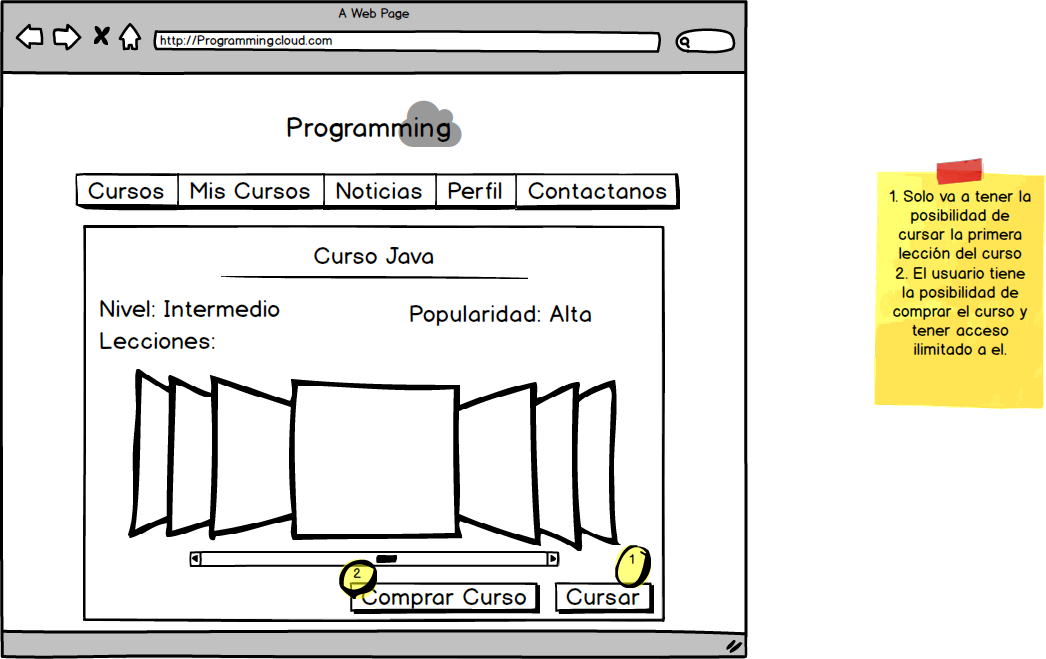
* Editar Perfil (Nueva contraseña)
  + Está relacionado con el RF11



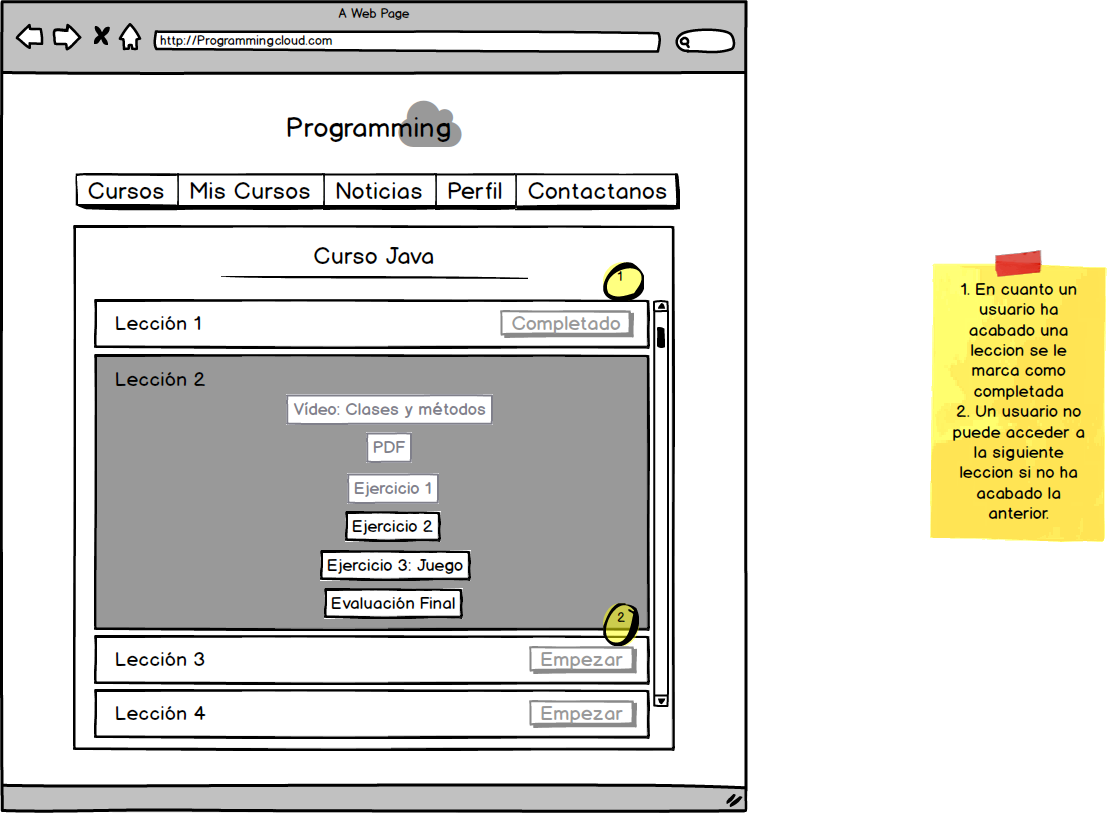
* Cursos
  + Está relacionado con el RF8



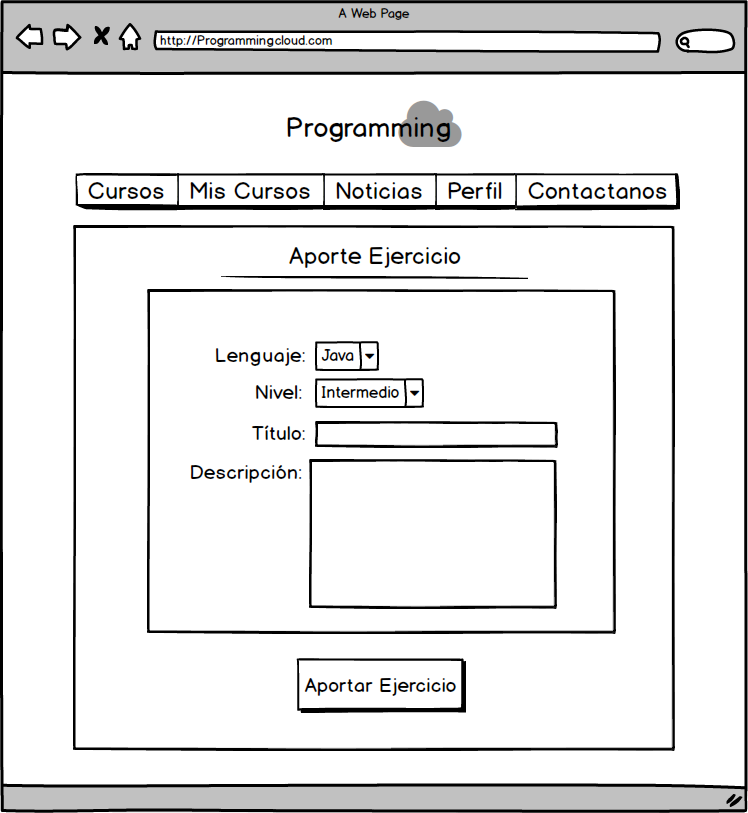
* Ver Cursos
  + Está relacionado con el RF8



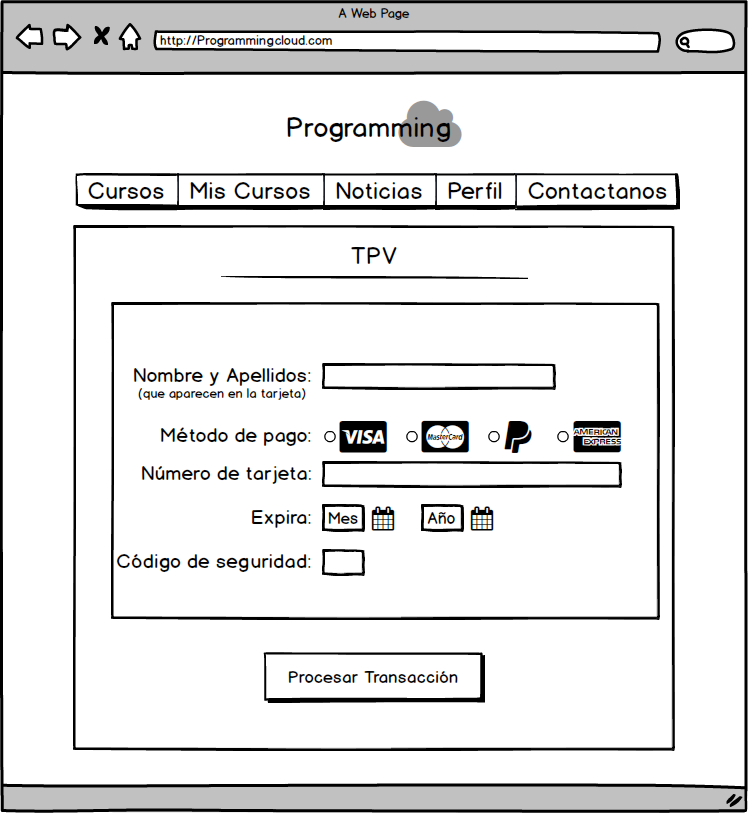
* Cursar Curso(boton al curso)
  + Está relacionado con el RF8



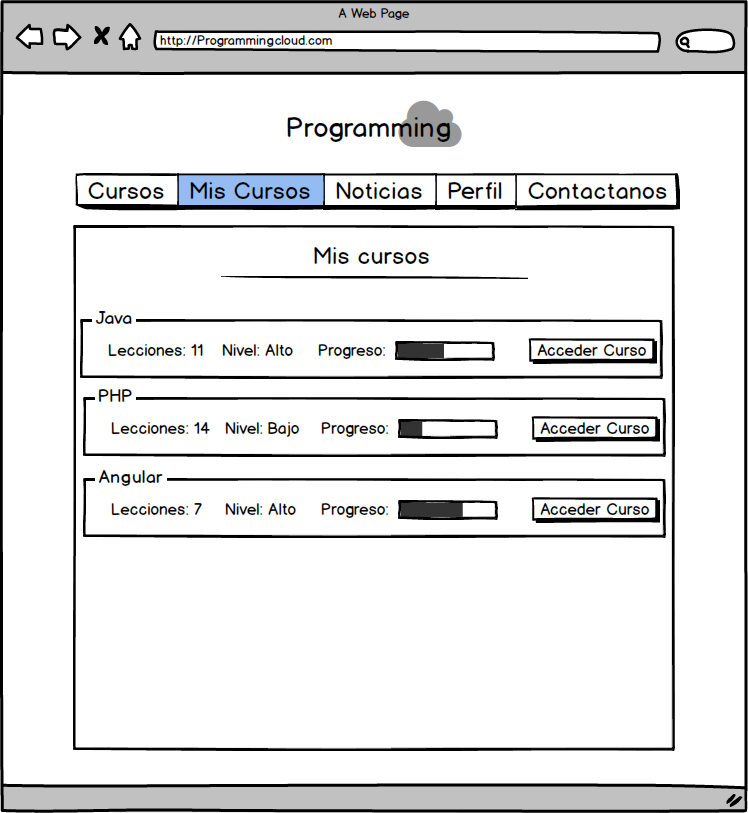
* Aportar Ejercicio
  + Está relacionado con el RF12



* Comprar Curso
  + Está relacionado con el RF9



* Mis Cursos
  + Está relacionado con el RF10.1

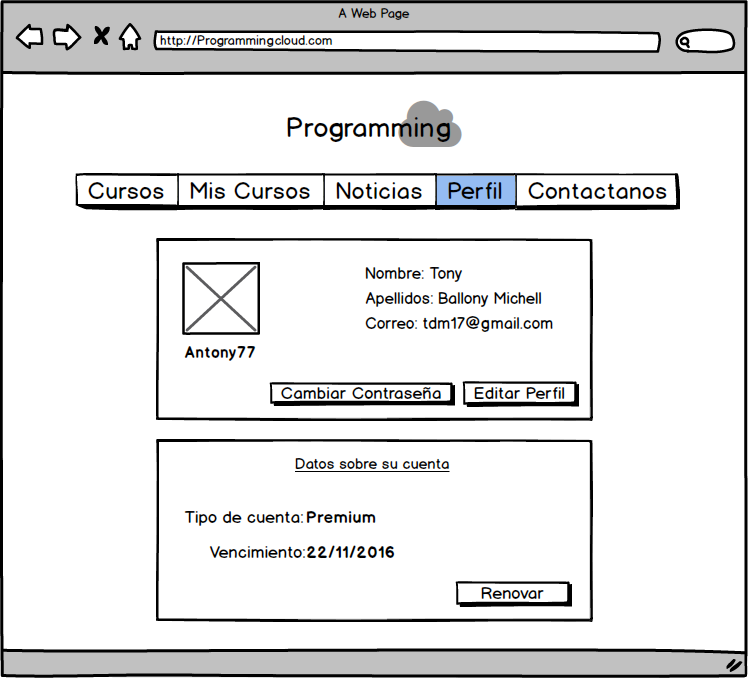


### 

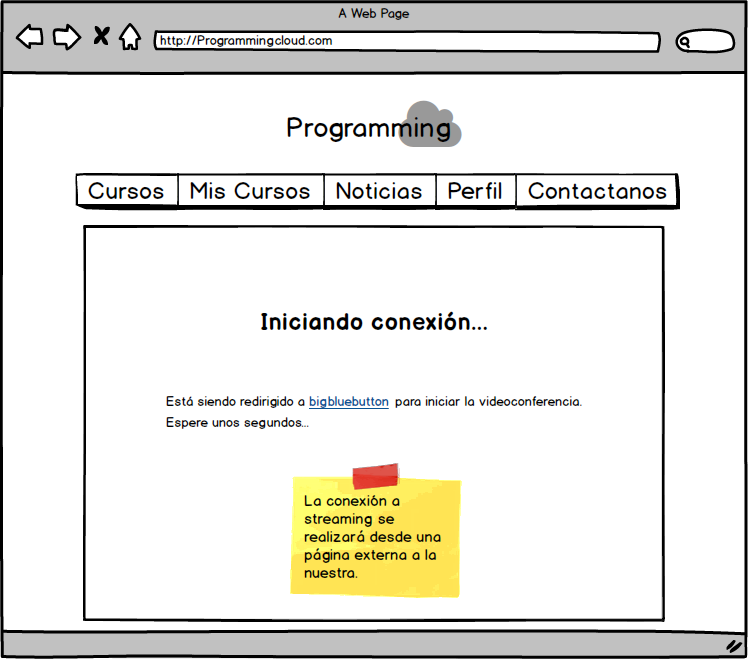
### 

### Usuario Registrado Premium ([RF Usuario Registrado (Premium)](#_6jfw6x23gobk))

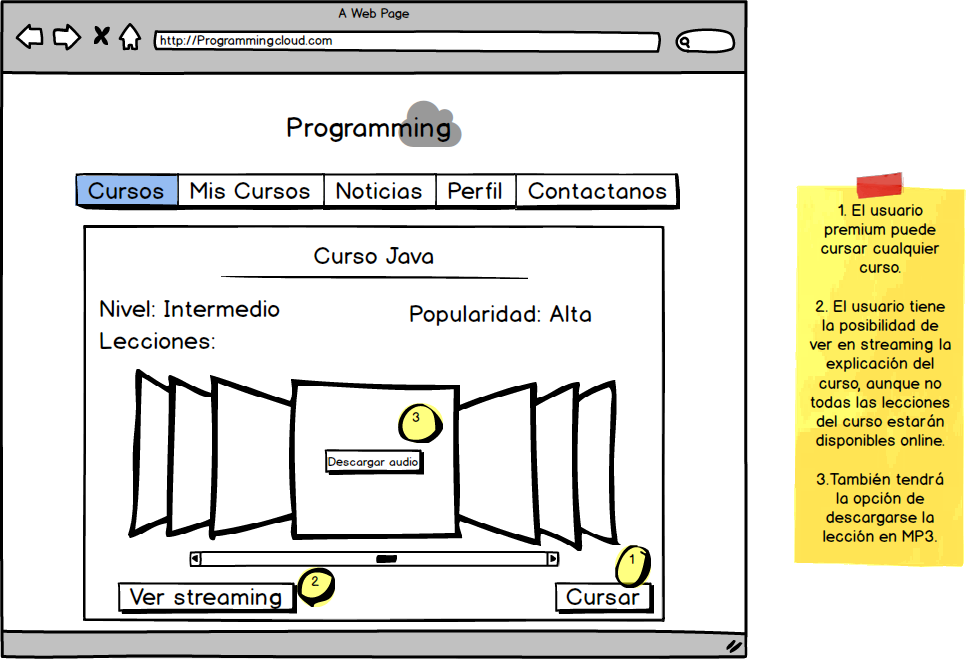
* Renovar Cuenta



* Acceder al Streaming
  + Está relacionado con el RF16



* Ver Mis Cursos
  + Está relacionado con el RF13 y RF15.1

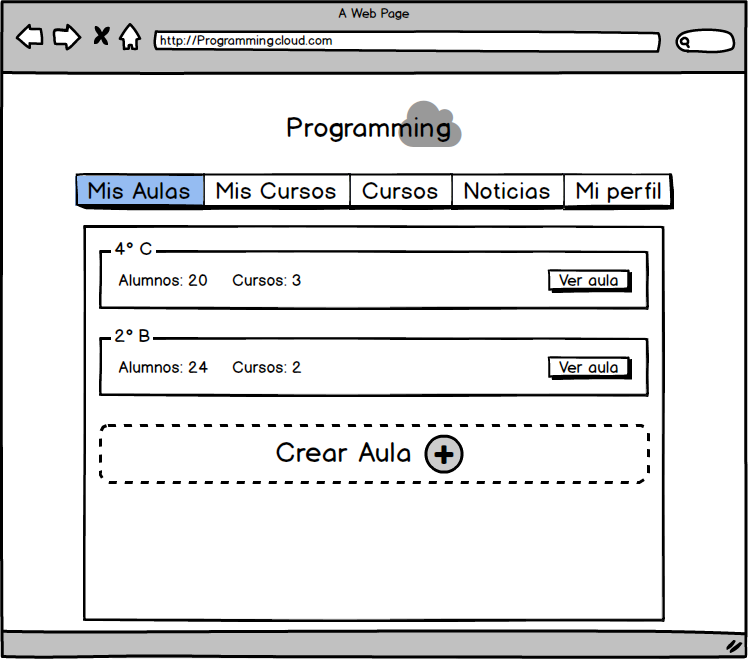


### 

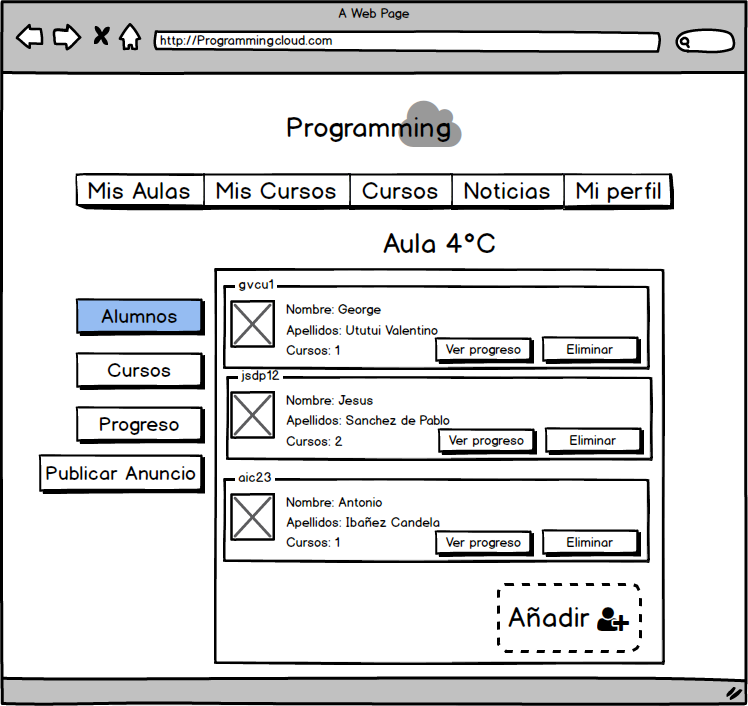
### 

### Usuario Profesor ([RF Usuario tipo Profesor](#_rlhceh6tca5c))

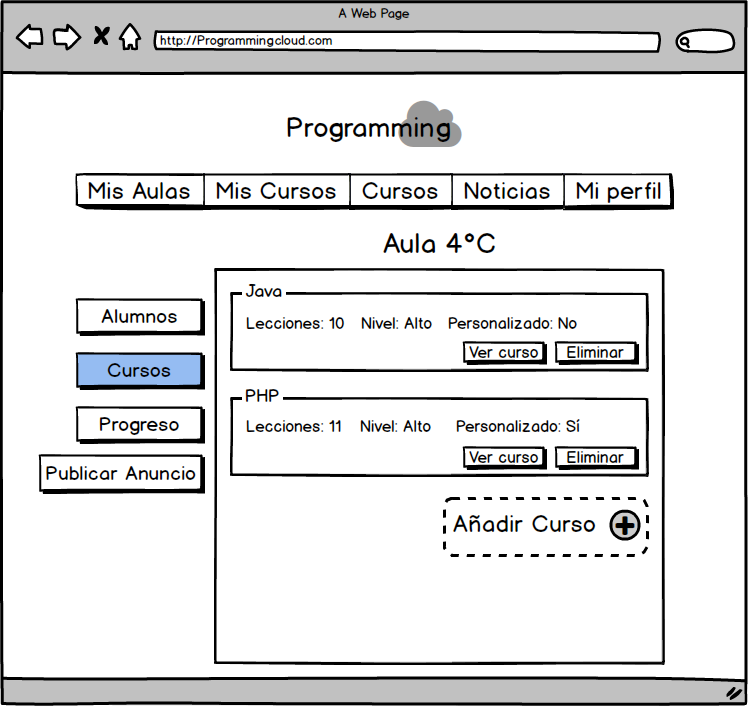
* Mis Aulas
  + Está relacionado con el RF20



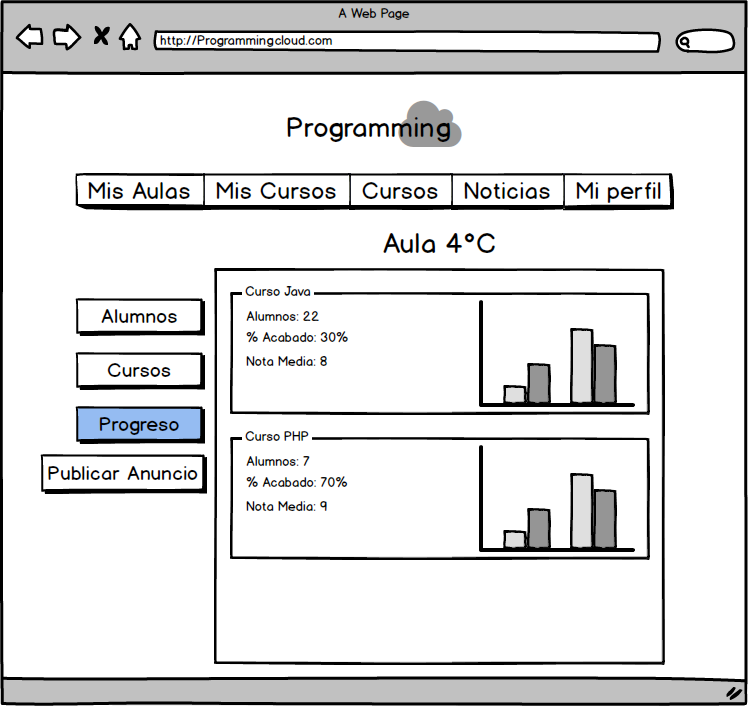
* Ver Aula - Alumnos
  + Está relacionado con el RF20.1



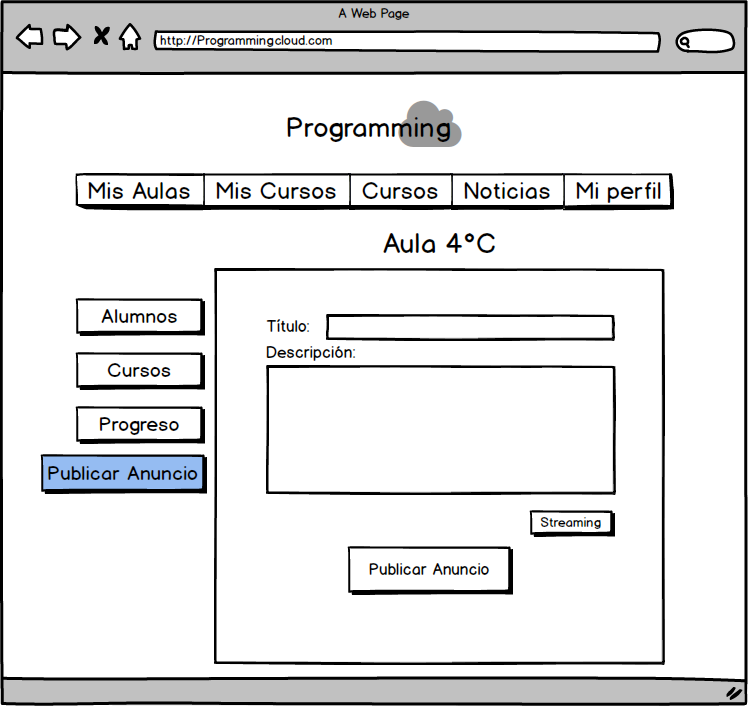
* Ver Aula - Cursos
  + Está relacionado con el RF23 y RF25



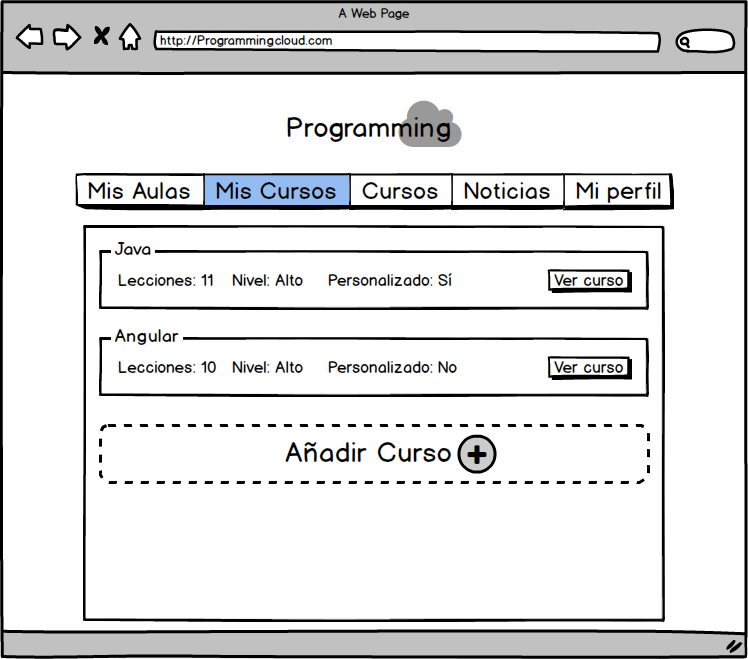
* Ver Aula - Progreso
  + Está relacionado con el RF20.3



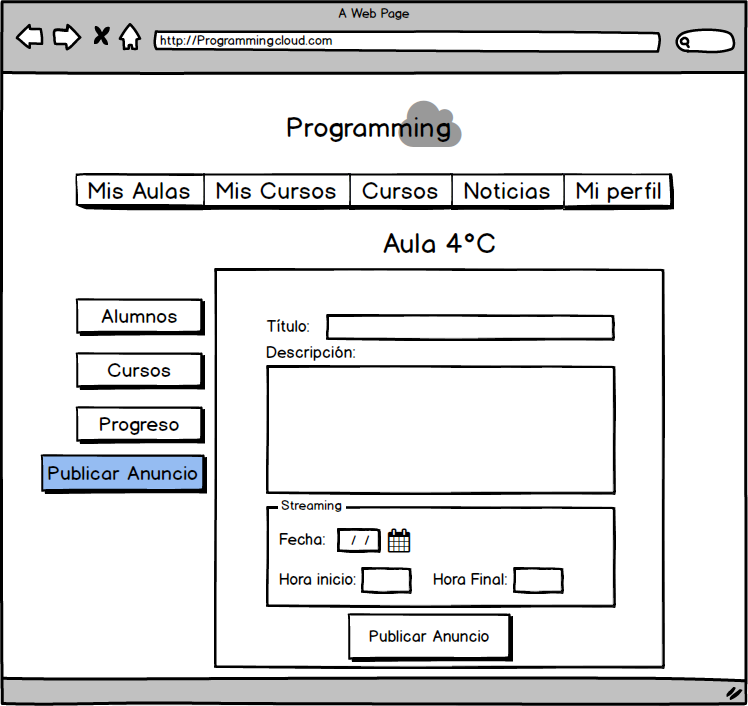
* Ver Aula - Publicar Anuncio
  + Está relacionado con el RF24



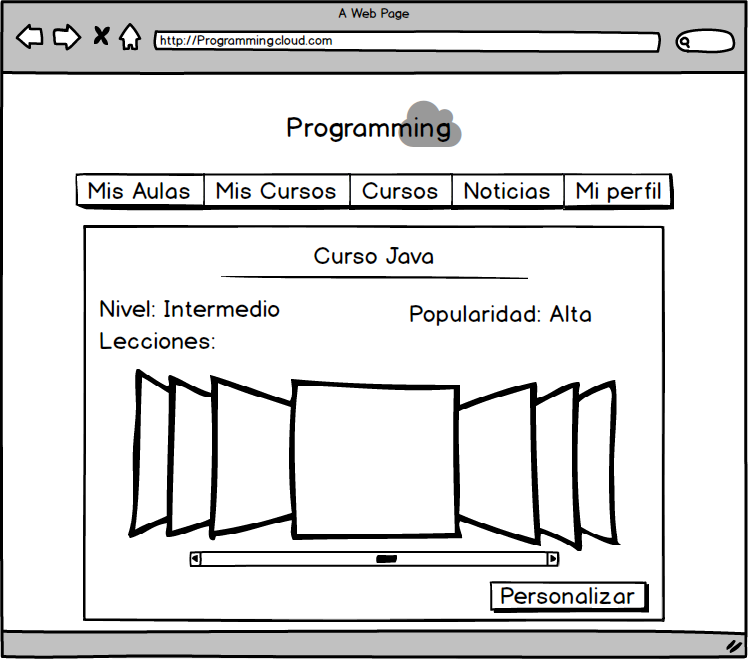
* Mis Cursos
  + Está relacionado con el RF25



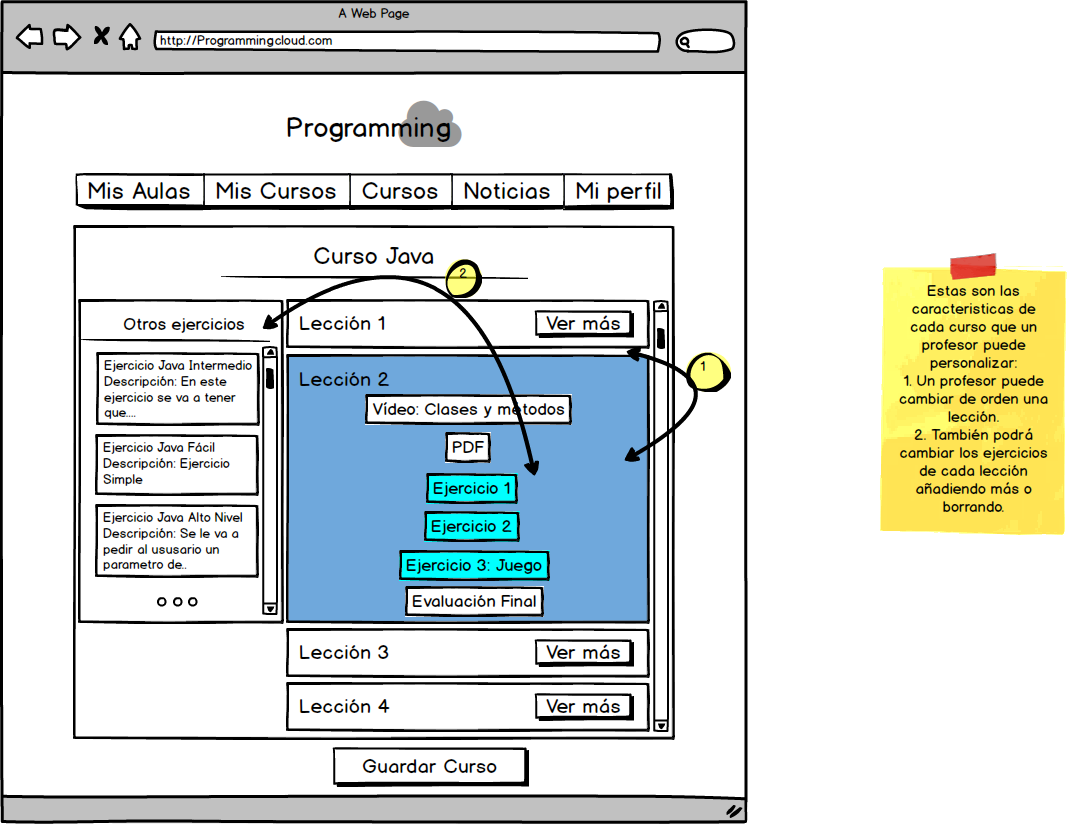
* Publicar Anuncio Tipo Streaming
  + Está relacionado con el RF24



* Ver Curso
  + Está relacionado con el RF25



* Personalizar Curso
  + Está relacionado con el RF21

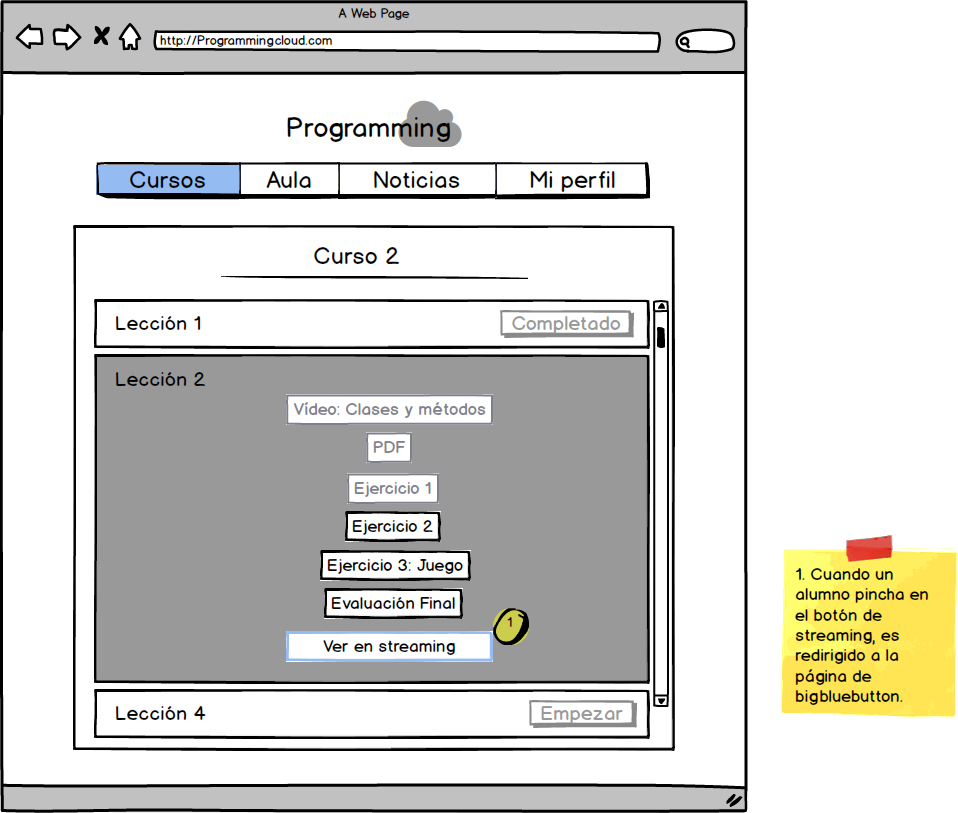


### 

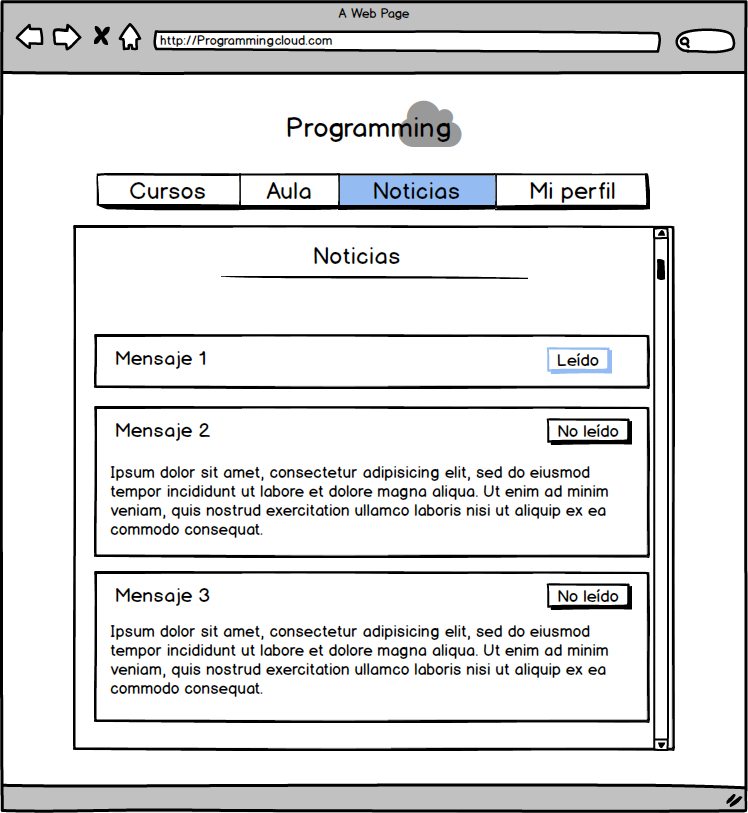
### 

### Usuario Alumno ([RF Usuario tipo Alumno](#_a6w4pv392wo2))

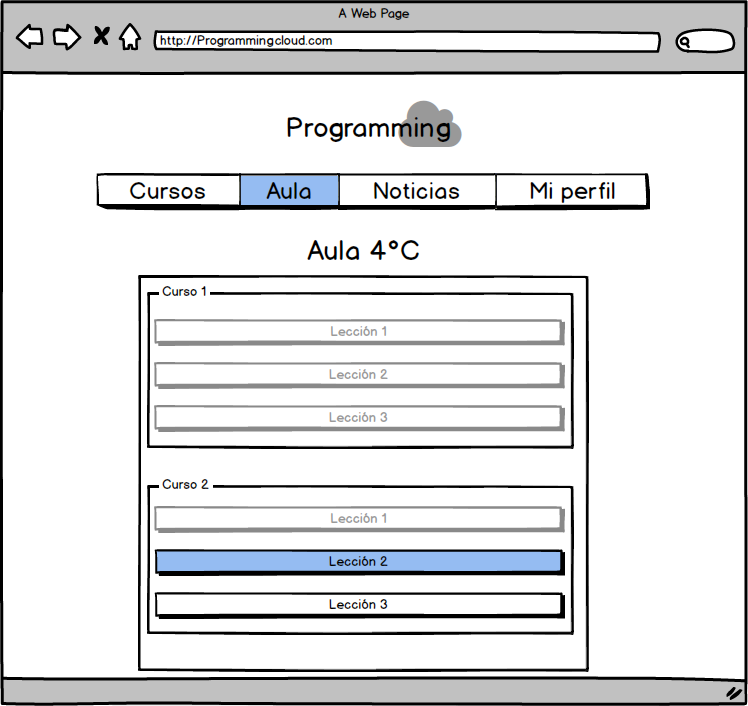
* Acceder a Streaming Profesor
  + Está relacionado con el RF19



* Ver Noticias Profesor
  + Está relacionado con el RF18



* Entrar en Aula
  + Está relacionado con el RF17



## 4.4 Diseño de las Bases de Datos

### Esquema Entidad-Relación

Debido a que la foto del esquema ER es demasiado grande no se ha podido incluir en este documento. Por lo que junto a la entrega de este documento está adjunta una foto con el esquema.

**Aclaraciones del modelo Entidad Relación:**

1. Una lección debe tener al menos un curso, pero en el modelo relacional no puede representarse una relación 0..N-1..N, cumpliremos esta restricción internamente mediante código.
2. También especificaremos mediante código el hecho de que un usuario premium no pueda acceder a los streamings de un profesor.
3. La entidad Instituto también representa a academias, por eso un profesor puede estar relacionado con un alumno desde su academia y su instituto en el caso de pertenecer ambos a la vez.
4. El LOM representado en el diagrama serán los metadatos que deberemos incluir al subir los ejercicios a nuestra biblioteca digital. Esto lo haremos para cumplir con el estándar SCORM.

### Tablas del sistema

* Usuario(Login,password,email,nombre,apellidos)

CP : Login

* Usuario Registrado(Login)

CP : Login

C. Aj. : Login -> Usuario

* Admin(login)

CP : Login

C. Aj. : Login -> Usuario

* Free(Login)

CP: Login

C. Aj. : Login -> Usuario Registrado

* Premium(Login)

CP: Login

C. Aj. : Login -> Usuario Registrado

* Alumno(Login)

CP: Login

C. Aj. : Login -> Usuario Registrado

* Profesor(Login)

CP : Login

C. Aj. : Login -> Admin

* Gestor(Login)

CP : Login

C. Aj. : Login -> Admin

* Streaming(id,fecha,descripcion,url,login\_adm)

CP: id

C. Aj. : login\_adm -> Admin

VNN: login\_adm

* Instituto(nombre, direccion, email)

CP: Nombre

* Aula(id,nombre,login\_profe)

CP: id

C. Aj. : login\_profe -> Profesor

VNN: login\_profe

* Curso(id, nombre, descripcion, nivel, lodin\_adm)

CD: id

C. Aj. : login\_adm ->Admin

VNN: login\_adm

* Leccion(id,nombre,url\_video,url\_pdf)

CP: id

* Ejercicio(id, valoracion, login\_usu, login\_ges)

CP: id

C. Aj. : login\_usu -> Usuario

C. Aj. : login\_ges -> Gestor

VNN: login\_usu, login\_ges

* Juego(id,url\_serv)

CP: id

C. Aj. : id ->Ejercicio

* Normal(id, pregunta, respuesta)

CP: id

C. Aj. : id -> Ejercicio

* Pack(id, descuento)

CP: id

A continuación vamos a presentar las tablas que surgen de relaciones Muchoas a Muchos o tablas que han surgido por la necesidad de guardar un dato en concreto al relacionar dos entidades.

* Sube(login\_usu, id\_ej, fecha)

CP: login\_usu

C. Aj. : Login\_usu -> Usuario

C. Aj. : login\_ej -> Ejercicio

VNN: id\_ej

* Leccion\_ej(id\_lec, id\_ej)

CP(id\_lej, id\_ej)

C. Aj. : id\_lec -> Leccion

C. Aj. : id\_ej -> Ejercicio

* Curso\_lec(id\_curso, id\_lec)

CP(id\_curso, id\_lecc)

C. Aj. : id\_curso -> Curso

C. Aj. : id\_lec -> Leccion

* Comprar(fecha, plazo, login\_usu, id\_pack)

CP(id\_pack, login\_usu)

C. Aj. : id\_pack -> Pack

C. Aj. : login\_usu -> Usuario

* C\_Insti(fecha, plazo, nombre\_inti, id\_pack)

CP(id\_pack, nombre\_insti)

C. Aj. : id\_pack -> Pack

C. Aj. : nombre\_insti -> Instituto

* Estar\_en(id\_pack, id\_curso)

CP(id\_pack, id\_curso)

C. Aj. : id\_pack -> Pack

C. Aj. : id\_curso -> Curso

* Ins\_alu\_prof(nombre\_insti, login\_alu, login\_prof)

CP(nombre\_insti, login\_alu)

C. Aj. : nombre\_insti -> Instituto

C. Aj. : login\_alu -> Alumno

C. Aj. : login\_prof -> Profesor

VNN: login\_prof

* Alumn\_aula(login\_alumno, id\_aula)

CP(login\_alumno,id\_aula)

C. Aj. : login\_alumno -> Alumno

C. Aj. : id\_aula -> Aula

* Aula\_curso(id\_aula, id\_curso)

CP(id\_aula, id\_curso)

C. Aj. : id\_aula -> Aula

C. Aj. : id\_curso -> Curso

* Prem\_strm(login\_premi, id\_stream)

CP(login\_premi, id\_stream)

C. Aj. : login\_premi -> Premium

C. Aj. : id\_stream -> Streaming

* Alumn\_strm(login\_alu, id\_stream)

CP(login\_alu, id\_stream)

C. Aj. : lodin\_alu -> Alumno

C. Aj. : id\_stream -> Streming