**MVP**

**ADMINISTRADORES**

* CRUD Escuelas: se le asocia nº asignaturas y usuarios según licencia contratada.

**ESCUELAS**

* Las escuelas piden registro eligiendo licencia con el número de usuarios que desean, a través de un formulario.
* Las escuelas pueden pedir cambiar de licencia.
* Las escuelas crean usuarios (alumnos y profesores) y asignaturas hasta el límite de su licencia, deben poder hacerlo por lotes.
* Las escuelas asignan profesores a las asignaturas.
* Las escuelas asignan alumnos a las asignaturas.

**PROFESORES**

* Los profesores pueden subir documentación a las asignaturas que imparten.
* Los profesores deben poder crear ejercicios (ver la sección de ejercicios)

**ALUMNOS**

* Los alumnos deben poder ver y descargar el contenido de las asignaturas que cursan.
* Los alumnos deben poder completar los ejercicios de las asignaturas que cursan
* Los alumnos deben poder ver el ranking de la asignatura(Esto en verdad los profesores también, mover a usuarios en gral)

**USUARIO ANÓNIMO**

* Ver página de bienvenida
* Ver licencias
* Contacto
* Formulario de petición de alta

**FUNCIONALIDADES EXTRA FUTURAS**

* El profesor puede sumar o restar puntos a un alumno
* El profesor puede copiar contenido entre asignaturas que imparte.
* Sistema de recompensas.

**TECNOLOGÍA**

· Python

· Django

· MySQL

**LICENCIAS**

· Las licencias van por número de usuarios y asignaturas.

· Los paquetes tipo son:

o Básico: 1 asignatura 30 usuarios

o Medio: 5 asignaturas 150 usuarios

o Grande: 15 asignaturas 450 usuarios

o A medida: Contactar con el equipo de desarrollo

Alumno extra: Precio por definir

Asignatura extra: Precio por definir

**ROLES DEL EQUIPO Y COSTE**

· Project Manager: Alejandro Román 300/semana(neto)

· Business Analyst: Álvaro Domínguez 300/semana(neto)

· Ingenieros Software: El resto. 250/semana(neto)

Amortización de equipos: 7 equipos, 1000€/ordenador

Hosting.

Autónomo de 7 personas, 250€/persona

10 horas semanales de trabajo.

**Ejercicios**

1. **Tipos**
   1. Test
      1. Selección múltiple
      2. Selección simple
      3. Combo
      4. Cuadro arrastrable
   2. Programación
   3. Competición (Ver test, posibles: ii e iv)
      1. Preguntas secuenciales, el profesor dispara la siguiente pregunta
      2. Tiempo límite, puntos según velocidad.
2. **Opciones de modificación de los ejercicios**
   1. Intentos límite
   2. Fecha límite
   3. Evaluable por el profesor/Autoevaluable
3. **Creación de ejercicios**
   1. **Test**
      1. Se elige crear el ejercicio de tipo test, se elige el subtipo. Luego se añade el enunciado y se eligen las opciones que van a aparecer. Finalmente se selecciona la correcta y se confirma el ejercicio.
   2. **Programación**
      1. El profesor elige crear el ejercicio de programación. Luego añade el enunciado y el código de respuesta. El código es ejecutable y si es correcto, se puede confirmar el ejercicio. Al corregirlo es el profesor el que lo hace manualmente, aunque se pueda ejecutar lo que vayan haciendo los alumnos.
   3. **Competición**
      1. Igual que el de tipo test, pero añadiendo los parámetros como el tiempo de cada pregunta y los puntos.

**SPRINT 1**

Consiste en desarrollar la funcionalidad mínima que pueda entregarse a los usuarios pilotos con una UI decente.

Se desarrollará:

* Semana 1 (12-Marzo a 16-Marzo):
  + CRUD de las escuelas, usuarios y asignaturas:
  + Página principal
  + Formularios de creación de todos los usuarios
  + Investigación del ejercicio tipo Notebook
* Semana 2 (19-Marzo a 23-Marzo):
  + Subida y almacenamiento de documentación por parte de los profesores
  + Investigación del ejercicio tipo Notebook