El sistema de evaluación inteligente estará conformado por varios módulos. El desarrollo será por fases hasta tener un producto funcional e integrado.

LISTO(back) NO LISTO(back) FRONT

# Fase 1. Módulo de gestión de usuarios y Módulo de gestión de imágenes.

1. Los usuarios que tendrá el sistema son de 3 tipos: administradores, profesores *(superusuarios)* y estudiantes.
2. Los usuarios podrán darse de alta y automáticamente serán estudiantes.
3. Un usuario es profesor o administrador solo si un administrador cambia su estado. Cualquier usuario y un administrador puede modificar la información de su perfil.*(por el momento solo nombre*) Todos los usuarios pueden ver a todos los usuarios registrados *(solo los admins pueden ver a los admins).*
4. El listado de usuarios tendrá la posibilidad de ordenarlas por fecha de creación, nombre, tipo.
5. Se podrá buscar usuarios por cualquiera de sus metadatos.
6. Un administrador o profesor puede eliminar o suspender una cuenta de un estudiante. Un administrador o profesor puede crear grupos de estudiantes para monitorear su avance.(*delfin)*

Datos básicos para los usuarios:

Nombre de usuario Contraseña

Correo electrónico

Tipo de usuario (administrador|profesor|estudiante)

Datos básicos para los grupos Nombre

Descripción

1. Las imágenes que se usarán en el sistema solo pueden ser gestionadas por los administradores y profesores.
2. Cualquier profesor o administrador podrá cargar imágenes y quedarán asociadas a él. Solo el propietario de una imagen puede editar sus metadatos
3. Cada imagen tendrá una colección de metadatos que la describen. Una imagen solo podrá ser borrada por su propietario.
4. El listado de imágenes tendrá la posibilidad de ordenarlas por fecha de carga o por propietario.
5. Se podrá buscar imágenes por cualquiera de sus metadatos.
6. Las imágenes por defecto se verán pequeñas, pero al seleccionarlas se verán en su tamaño real.
7. Por defecto las imágenes son privadas, solo el propietario puede hacerlas públicas para su visualización.

Datos básicos para las imágenes Archivo

Propietario Título

Nombre original del archivo Formato

Descripción Palabras clave

La estructura inicial de la base de datos debe modelar las siguientes aseveraciones Un profesor o administrador es propietario de muchas imágenes

Una imagen solo tiene un propietario Una imagen tiene muchas palabras claves

Una palabra clave puede estar asociada a muchas imágenes Un grupo está formado por muchos estudiantes

Un estudiante puede estar en muchos grupos

# Fase 2. Módulo de gestión de preguntas y editor de imágenes.

1. Las preguntas serán de opción múltiple (5 opciones a lo más) y una de ellas será la respuesta correcta. Una pregunta podrá tener asociada una imagen del repositorio.
2. Las preguntas que se usarán en el sistema solo pueden ser gestionadas por los administradores y profesores.
3. Cualquier profesor o administrador puede crear nuevas preguntas. Solo el propietario podrá editarlas o borrarlas. Por defecto las preguntas son privadas, solo el propietario puede hacerlas públicas para su visualización.
4. Cada pregunta tendrá una colección de metadatos que la describen.
5. Si una imagen está asociada a una pregunta, dicha imagen no podrá ser borrada ni puesta privada.
6. El listado de preguntas públicas tendrá la posibilidad de ordenarse por el texto de la pregunta o por propietario.
7. Se podrá buscar preguntas por cualquiera de sus metadatos o texto.
8. La imagen seleccionada para una pregunta podrá tener texto, flechas y círculos (ver ejemplo en https://[www.drawerjs.com/)](http://www.drawerjs.com/)) para una mejor identificación. Esos elementos deberán asociarse a la pregunta.
9. El propietario podrá tener una vista previa de la pregunta en una ventana aparte. Los demás usuarios podrán tener la vista previa de las versiones públicas de las preguntas en el listado.

Datos básicos de las preguntas Título

Pregunta Respuesta 1 a 5 Respuesta correcta Propietario Palabras clave Imagen asociada Mapa de objetos

Dificultad (1,2,3,4…)

-La estructura de la base de datos debe modelar las siguientes aseveraciones: Un profesor o administrador es propietario de muchas preguntas

Una pregunta solo tiene un propietario Una pregunta tiene muchas palabras clave

Una palabra clave puede estar asociada a muchas preguntas Una pregunta tiene asociada una imagen

Una imagen puede estar asociada a muchas preguntas

\*\*\* Para la búsqueda por texto deberán usar índices de tipo FULLTEXT y modificar la sentencia SQL de consulta

<http://www.hackingwithphp.com/9/3/18/advanced-text-searching-using-full-text-indexes> Esto permitirá hacer una búsqueda por relevancia para todos los elementos.

# Fase 3. Módulo de evaluación.

Un profesor puede crear una evaluación conformada por una colección de preguntas públicas. Una pregunta pública que forme parte de una evaluación no podrá ser borrada ni vuelta privada. Cada evaluación tendrá un nombre y una descripción para su identificación. Por defecto las evaluaciones son privadas, solo el propietario puede hacerlas públicas para su visualización.

Cada evaluación tendrá una colección de metadatos que la describen.

El listado de evaluaciones públicas tendrá la posibilidad de ordenarse por el texto de la evaluación o por propietario.

Se podrá buscar evaluaciones por cualquiera de sus metadatos o texto.

El propietario podrá tener una vista previa de la evaluación en una ventana aparte. Los demás usuarios podrán tener la vista previa de las versiones públicas de las evaluaciones en el listado. Para ello se listarán las preguntas que conforman la evaluación y entonces se podrá visualizar cada una de ellas.

Datos básicos de las evaluaciones Título

Descripción Palabras clave

Tiempo (duracion)

Dificultad (0,1,2,3,4..) 0= aleatorio

NUEVO

access

n\_questions

n\_aciertos

modo (training, mision, battle )

tiempo inicio

tiempo fin

La estructura de la base de datos debe modelar las siguientes aseveraciones: Un profesor o administrador es propietario de muchas evaluaciones

Una evaluación solo tiene un propietario Una evaluación tiene muchas palabras clave

Una palabra clave puede estar asociada a muchas evaluaciones Una evaluación tiene asociada muchas preguntas

Una pregunta puede estar asociada a muchas evaluaciones

# Fase 4. Módulo de juego.

La página principal del sistema tendrá la opción de registro e inicio de sesión. Además de un texto descriptivo acompañado de una captura de una pregunta de ejemplo. Existirá un medallero por cada modalidad de juego.

Al iniciar sesión el usuario podrá modificar sus datos de registro, ver sus estadísticas previas e iniciar una modalidad de evaluación.

Tendremos tres modalidades de juego

* Entrenamiento. El usuario escoge el tema (palabra clave), el número de preguntas y el tiempo total para contestar.
* Misión. El usuario selecciona un conjunto de evaluaciones preconfiguradas.
* Batalla. El sistema tratará de identificar el nivel de aprendizaje del estudiante

Para cualquier modalidad, el sistema registrará la actividad del estudiante en cada partida. Al final de una partida presentará los resultados. Dichos resultados podrán ser públicos en el medallero.

La actividad que se registrará por cada estudiante es Fecha/hora de inicio de la evaluación

Fecha/hora de finalización de la evaluación Fecha/hora de visualización de una pregunta Fecha/hora de selección de una opción Fecha/hora de selección del botón de siguiente Puntuación de cada pregunta

Puntuación de la evaluación

El modo batalla presentará de forma aleatoria 10 preguntas de un mismo nivel y luego de un % de aciertos presentará preguntas del siguiente nivel y así sucesivamente hasta que el estudiante no pueda avanzar.

La estructura de la base de datos debe modelar las siguientes aseveraciones:

Un estudiante tiene muchas partidas de algún tipo Cada partida tiene muchas acciones

Hay un medallero por cada tipo modalidad

# Fase 5. Módulo de monitoreo.

El estudiante podrá ver el desempeño en cada una de las partidas que haya realizado en la plataforma.

El profesor podrá ver los resultados de las partidas por alumno o grupo. Esta información se presentará en forma de tablas y gráficas.

Ejemplo:

