**Decisiones de diseño pendientes de documentar**

* **Realización de exámenes**
* **¿TipoTest // no tipoTest? Jerarquía de tipo de preguntas.**

# Decisiones de diseño

En esta sección describiremos las decisiones de diseño que se han tomado a lo largo del desarrollo de la aplicación que vayan más allá de la mera aplicación de patrones de diseño o arquitectónicos.

## Decisión 1: Añadir preguntas a un examen

### Descripción del problema:

Nos planteamos la adición de preguntas a un examen en la página. El problema reside en que no podemos saber de antemano el número de preguntas que tendrá ni de qué tipo serán, ya que no hay una respuesta fija.

### Alternativas de solución evaluadas:

Alternativa 1.a: Añadir las preguntas en la página de creación o edición del examen.

En la página de creación/edición del examen se colocaría un botón de añadir pregunta. Mediante Javascript se irían añadiendo secciones al formulario y se pasarán a la base de datos desde un único formulario.

**Ventajas**:

* Es la forma más rápida si se quieren añadir varias preguntas.
* Se realizan menos accesos a la base de datos.

**Inconvenientes**:

* Encontramos, el procedimiento, complejo de implementar, ya que no tenemos todos los conocimientos necesarios de las tecnologías requeridas para llevarlo a cabo.
* Puede resultar confuso para la navegabilidad del usuario medio que va a utilizar la aplicación.

Alternativa 1.b: Crear una sección enfocada exclusivamente a las preguntas del examen, en las que se puedan añadir y editar las preguntas de una en una.

**Ventajas:**

* Para un usuario promedio, al estar dividido en secciones más específicas, puede ser más sencillo de comprender.
* La implementación es sencilla.
* Permite editar una pregunta sin tener que modificar el resto del examen.
* Se pueden añadir preguntas de distinto tipo a un examen más fácilmente.

**Inconvenientes:**

* Múltiples accesos a la base de datos.
* Puede ser lento si se desean añadir muchas preguntas.

### Justificación de la solución adoptada

Alternativa 1b: Creemos que gran parte del público estará conformada por personas que no están acostumbradas a trabajar con sistemas de información de uso avanzado. Por tanto, prima que el funcionamiento de nuestra aplicación sea sencillo. Además, debido al plazo del que disponemos, nos hemos decantado por la solución que podíamos conocer mejor a primera vista.

## Decisión 2: Añadir opciones a una pregunta

### Descripción del problema:

El problema es similar que surge al añadir preguntas a un examen. Queremos añadir opciones a una pregunta. El problema reside en que no tienen un número fijo de opciones.

### Alternativas de solución evaluadas:

Alternativa 1.a: Añadir las opciones en el momento en el que se crea o edita una pregunta.

En el formulario de creación de pregunta, habrá un botón “Añadir opción” que, cada vez que sea pulsado, añadirá de manera asíncrona un formulario para una nueva opción.

**Ventajas**:

* Es la forma más rápida si se quieren añadir varias opciones.
* Se realizan menos accesos a la base de datos.

**Inconvenientes**:

* Al igual que con la decisión de añadir preguntas a exámenes, encontramos, el procedimiento, complejo de implementar. Presenta dificultades similares a la anterior decisión.
* Puede resultar confuso para la navegabilidad del usuario medio que va a utilizar la aplicación.

Alternativa 1.b: En la página donde se listan las preguntas del examen, al lado de cada pregunta, se muestra el botón “Añadir opción”. Este botón llevará a un formulario exclusivo para esa opción.

**Ventajas:**

* Para un usuario promedio, puede ser un procedimiento fácil de entender.
* La implementación es sencilla.
* Permite editar una opción sin alterar el resto de preguntas, de manera que es menos probable modificar otras opciones o preguntas por error.
* Es una solución más coherente a la ya decidida para añadir preguntas al examen.

**Inconvenientes:**

* Múltiples accesos a la base de datos.
* Puede ser lento si se desean añadir muchas opciones.

### Justificación de la solución adoptada

Alternativa 1b: Creemos que gran parte del público estará conformada por personas que no están acostumbradas a trabajar con sistemas de información de uso avanzado. Por tanto, prima que el funcionamiento de nuestra aplicación sea sencillo. Además, debido al plazo del que disponemos, nos hemos decantado por la solución que podíamos conocer mejor a primera vista.