Proyecto interdisciplinario de Bases de Datos 1 y Diseño Web (Draft)

Planteamiento del requerimiento:

Se solicita hacer un sistema web para una revista en línea que permita crear, listar, modificar, moderar y publicar artículos.[[1]](#footnote-1)

Existen los siguientes tipos de perfiles[[2]](#footnote-2): visitante anónimo, visitante logueado, autor, moderador y administrador.

Cada visitante puede crear su propio usuario, enviándole un correo de confirmación por medio de la página.

El visitante logueado puede tener activa una suscripción[[3]](#footnote-3) o no. La fecha de vencimiento de suscripción puede ser configurada por el administrador en una pantalla de administración de usuarios, así como inactivación o activación de usuarios y cambios de roles. Cada usuario por su parte tendrá acceso a modificar la información de su propio perfil (nombre, correo, password, etc.) y allí mismo contará con una opción para ingresar sus datos de cobro[[4]](#footnote-4) y comprar una suscripción por 6 meses o por un año indicando el precio de cada una.

El home debe contar por lo menos con un elemento que liste los últimos 10 artículos publicados, a nivel grafico mostrar un set de 4 destacados (a escoger por el administrador), una navegación entre categorías y subcategorías, header y footer.

La revista tendrá categorías[[5]](#footnote-5) de publicación, cada una de esas categorías contara con un subhome y dentro de estas listara subcategorías[[6]](#footnote-6) y artículos, las páginas de categorías y subcategorías tendrán 2 diseños a escoger con distintos elementos. Existen categorías con acceso gratuito, y categorías con acceso solo para usuarios pagados, en cada lugar que aparezca listado un artículo que puede ser accesado solo de pago, deberá indicarse por medio de una etiqueta (premium). el administrador deberá contar con una consola para crear y configurar las categorías/subcategorías y configurarlas.

El sistema deberá contar con un sistema de comentarios[[7]](#footnote-7) recursivos, es decir, se podrá vincular una respuesta de un comentario a un comentario en particular, debiéndolo mostrar en subniveles. El visitante anónimo podrá ver los comentarios, pero no podrá crear nuevos. El visitante logueado podrá crear comentarios y reportar comentarios inapropiados. El moderador recibirá notificación en una página de moderación de estos comentarios, agrupados de manera tal que al revisar la página de moderación deberá listar los comentarios reportados agrupados, ordenados por el número de reportes que ha tenido y un link hacia la página que lo lleve directamente al comentario a revisar. Tanto en el comentario como en la página de moderación, el moderador podrá censurar o des censurar un comentario, lo cual hará que aparezca un mensaje de "comentario censurado", sin embargo, el comentario no será borrado del sistema. también podrá marcar todos los reportes de un comentario como ignorados en caso considere que no es necesario censurar un comentario y no volverá a aparecer en los reportes de moderación. Al final del día, el sistema deberá enviar un mensaje de correo a todos los moderadores indicando un resumen de cantidad de comentarios reportados con sus respectivo resumen de veces reportado y que no haya sido moderado tanto para el día de hoy como en general.

Si un comentario tiene más de 25 reportes sin haber sido atendido por un moderador, el comentario se censurará automáticamente, pero al moderador se lo marcará como aun no revisado teniendo la posibilidad de des censurarlo.

Los autores podrán crear artículos, a cada autor, dentro de su perfil, el administrador podrá configurarle dentro de que categorías puede crear artículos, y cuando el autor cree un artículo únicamente estas categorías/subcategorías le aparecerán disponibles. Al momento de crear un artículo, el autor contará con 3 plantillas para crear el artículo, que como mínimo cada una deberá contar con: una imagen, el nombre del autor, fecha de publicación, titulo y texto del artículo. Sin embargo, deberán contar con elementos como, links a videos, carrusel de imágenes, diferentes subtítulos, más imágenes, etc. Las plantillas serán propuestas por cada grupo (distintas para cada uno) y aprobadas por el catedrático.

Después de crear el artículo, este no será publicado, si no el autor podrá dejarlo en draft o marcarlo para revisión y publicación. Los moderadores tendrán dentro de su modulo la revisión de artículos a publicar y si lo acepta será publicado, de lo contrario le mandara un correo al autor indicándole que ha sido rechazado, volviendo el articulo a modo draft. En caso el autor quiera hacer una modificación a un artículo publicado, podrá entrar y modificarlo, pero volverá a enviar una solicitud de revisión sin afectar el articulo anteriormente publicado hasta que un moderador lo apruebe, reemplazando así el anterior. Únicamente el autor de un artículo (y los moderadores) pueden modificar un artículo.

Todas las paginas deberán contar con un área de anuncio[[8]](#footnote-8) el cual mostrará una imagen[[9]](#footnote-9) con un link que escogerá al azar de un listado configurado por el administrador en un módulo de anuncios. El sistema deberá llevar control y poder entregar un detalle de cuantas veces se ha dado clic a un anuncio y en que página se le ha dado clic, pudiendo pedir un reporte por total de clics para un mismo anuncio o por total de clics por página (sin importar el anuncio en particular).

El sistema deberá contar con un buscador que filtre combinadamente por categorías, autores, fecha de publicación y/o texto y liste los artículos que cumplan con ese criterio, pudiendo ser ordenado por categoría, fecha de publicación o nombre del autor.

Deberá de llevarse un control sobre cuantas veces ha sido visitado un artículo, este dato aparecerá únicamente a los autores, administradores y moderadores dentro del mismo artículo.

Nota: la evaluación del proyecto, si bien es uno mismo, será separada en cada una de las áreas y conceptos vistos en cada clase, pudiendo obtener una nota distinta en cada área según su implementación enfocada a lo visto en clase.

Coloque especial atención en el uso de buenas prácticas al momento de desarrollar el proyecto.

Tanto las categorías, y subcategorías deben ser dinámicas y manejadas por tablas en la base de datos, por lo que, si se desea crear un registro nuevo de cualquiera de estos tipos, se podrá hacer dinámicamente.

Los datos de los usuarios, los datos de los artículos deberán ser propuestos y aprobados por el catedrático por cada grupo.

En las búsquedas únicamente aparecerá listado el título, auto, fecha de publicación, categoría y subcategoría.

Si un usuario no se ha logueado por más de un año será deshabilitado automáticamente.

Nota: podrá haber cambios conforme avance el curso ajustando el scope y funcionalidades. Ésta es una versión DRAFT.

Preentregas: 2 preentregas, una antes del tercer parcial, otra después del tercer parcial y entrega final el viernes antes de la semana de exámenes finales. Scope a definir para cada preentrega.

Out of scope: consideraciones estrictas de seguridad.

===========================================================================

Funcionalidades y evaluación DB:

1. Login, registro, cambio de password y administración de usuarios en general. Manejo de perfiles registrados o anónimos a nivel de datos y correcta implementación en el portal.
2. Creación de artículos.
3. Categorías, subcategorías: creación de páginas, administración y manejo dinámico.
4. Venta suscripciones y funcionalidad de paga.
5. Módulo de moderación con reporte diario enviado por correo implementado en procedimiento almacenado y censurado automático implementado con trigger.
6. Búsqueda con filtros y ordenamiento.
7. Subir por lo menos una fotografía (blob) para los artículos
8. Servicio para entregar los datos de un artículo, listado de categorías y subcategorías por medio de un json
9. Actualización de articulo.
10. Manejo de perfiles de usuario
11. Módulo de anuncios.
12. FNBC, llaves e índices creados adecuadamente según las buenas prácticas y funcionalidad.
13. Invalidación de usuarios al año de no loguearse automática.
14. Implementar un procedimiento almacenado con parámetros out y que tenga sentido dentro del uso del portal. (a ser aprobadas por el catedrático)
15. Implementar al menos una funcionalidad con queries dinámicos y con subqueries, por aprobar previamente por el catedrático.
16. Implementar al menos 2 vistas y 2 funciones que tengan sentido dentro del uso del portal. (a ser aprobadas por el catedrático)
17. Cualquier consideración y programación necesaria para cumplir con el planteamiento del requerimiento en un proyecto del mundo real.
18. Tecnologías: php y MySQL (no ORM)
19. Documentación:
20. Modelo entidad relación
21. Manual funcional y técnico. El manual funcional debe tener un enfoque de usuario visitante con casos de uso.
22. Nota: Implementado con el proyecto de diseño web.
23. Entregables: fuentes de php y script para crear base de datos subidas en repositorio de versiones. Listado de tareas efectuadas por cada integrante del grupo con % completado por cada uno para cada entrega siendo esto un prerrequisito de entrega, documentación completa. Cada alumno debe tener registrado al menos 10 commits por fase al momento de entregar cada fase.
24. Grupo máximo por afinidad de 2 personas.

===========================================================================

Funcionalidades y evaluación Diseño web:

Responsivo (móvil y desktop)

Template general del sitio

1. Homepage con listado de artículos y categorías
2. Sub home de categorías y subcategorías (2 plantillas)
3. Página de articulo en 3 plantillas distintas. Una de las plantillas de artículo debe implementar traer la información e inyectarse en la página por medio de un json y usando React
4. Página de creación de articulo
5. Página de registro y login
6. Módulo de moderación
7. Página de perfil de usuario
8. Página de información e historia de la revista (página web)
9. Página de contacto con la revista (página web)
10. Página de preguntas frecuentes.

Módulo de anuncios.

Look and feel de la aplicacion y flujo de los usuarios considerando una aplicacion del mundo real.

Deberá implementar al menos un mixin utilizando buenas prácticas de implementación y con sentido adecuado dentro del proyecto.

Entregables:

Fuentes y build en ambiente de desarrollo, fuentes versionadas en repositorio. Cualquier consideración y programación necesaria para cumplir con el planteamiento del requerimiento.

Documentación:

Manual funcional y técnico con wireframes. Descripción semántica de la estructura HTML de cada página.

Set de HTML funcionales generados con contenido de muestra y final según sea el caso con sus respectivas librerías. Entrega en ambiente de desarrollo funcional en web server local como el visto en clase.

Tecnologías: React, foundation, Sass.

Entregables: Fuentes en repositorio web (GitHub), cada alumno con 10 commits, documentación. Listado de tareas efectuadas por cada integrante del grupo con % completado por cada uno en cada tarea. Podría ser verificable en el historial de git y es requisito de entrega de cada fase.

Fases: F1-> preentrega 1, F2-> preentrega2, F3 -> entrega final. Fechas y alcance de cada fase por definir

En la fase final todas las funcionalidades y entregables deben estar completos, se restarán puntos por funcionalidades ya evaluadas y que no funcionen en la entrega final o en la segunda entrega.

Grupo máximo por afinidad de 2 personas. En caso sea necesario se considerará un único grupo de 3 con un alcance mayor por definir.

Imagen que contiene captura de pantalla

Descripción generada automáticamente

1. Tabla 1 [↑](#footnote-ref-1)
2. Tabla 2 (5) [↑](#footnote-ref-2)
3. Tabla 3 [↑](#footnote-ref-3)
4. Tabla 4 [↑](#footnote-ref-4)
5. Tabla 5 [↑](#footnote-ref-5)
6. Tabla 6 [↑](#footnote-ref-6)
7. Tabla 8 [↑](#footnote-ref-7)
8. Tabla 10 [↑](#footnote-ref-8)
9. Tabla 11 [↑](#footnote-ref-9)