

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.

Existen los siguientes tipos de perfiles: visitante anonimo, visitante logueado, autor, moderador y administrador.

Cada visitante puede crear su propio usuario, enviandole un correo de confirmacion por medio de la pagina.

El visitante logueado puede tener activa una suscripcion o no. La fecha de vencimiento de suscripcion puede ser configurada por el administrador en una pantalla de administracion de usuarios, asi como inactivacion o activacion de usuarios y cambios de roles. Cada usuario por su parte tendra acceso a modificar la informacion de su propio perfil (nombre, correo, password, etc) y alli mismo contara con una opcion para ingresar sus datos de cobro y comprar una suscripcion 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 ultimos 10 articulos publicados, a nivel grafico mostrar un set de 4 destacados (a escoger por el administrador), una navegacion entre categorias y subcategorias, header y footer.

La revista tendra categorias de publicacion, cada una de esas categorias contara con un subhome y dentro de estas listara subcategorias y articulos, las paginas de categorías y subcategorias tendran 2 diseños a escoger con distintos elementos. Existen categorias con acceso gratuito, y categorias con acceso solo para usuarios pagados, en cada lugar que aparezca listado un articulo que puede ser accesado solo de pago, debera indicarse por medio de una etiqueta (premium, plus, pro). el administrador debera contar con una consola para crear y configurar las categorias/subcategorias y configurarlas.

El sistema debera contar con un sistema de comentarios recursivos, es decir, se podra vincular una respuesta de un comentario a un comentario en particular, debiendolo mostrar en subniveles. El visitante anonimo podra ver los comentarios, pero no podra crear nuevos. El visitante logueado podra crear comentarios y reportar comentarios inapropiados. El moderador recibira notificacion en una pagina de moderacion de estos comentarios, agrupados de manera tal que al revisar la pagina de moderacion debera listar los comentarios reportados agrupados, ordenados por el numero de reportes que ha tenido y un link hacia la pagina que lo lleve directamente al comentario a revisar. Tanto en el comentario como en la pagina de moderacion, el moderador podra censurar o descensurar un comentario, lo cual hara que aparezca un mensaje de "comentario censurado", sin embargo el comentario no sera borrado del sistema. tambien podra marcar todos los reportes de un comentario como ignorados en caso considere que no es

necesario censurar un comentario y no volvera a aparecer en los reportes de moderacion. Al final del dia, el sistema debera 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 dia de hoy como en general.

Si un comentario tiene mas de 25 reportes sin haber sido atendido por un moderador, el comentario se censurara automaticamente, pero al moderador se lo marcara como aun no revisado teniendo la posibilidad de descensurarlo.

Los autores podran crear articulos, a cada autor, dentro de su perfil, el administrador podra configurarle dentro de que categorias puede crear articulos, y cuando el autor cree un articulo unicamente estas categorias/subcategorias le apareceran disponibles. Al momento de crear un articulo, el autor contara con 3 plantillas para crear el articulo, que como minimo cada una debera contar con: una imagen, el nombre del autor, fecha de publicacion, titulo y texto del articulo. Sin embargo, deberan contar con elementos como , links a videos, carrousel de imagenes, diferentes subtítulos, mas imagenes, etc. Las plantillas seran propuestas por cada grupo (distintas para cada uno) y aprobadas por el catedratico.

Despues de crear el articulo, este no sera publicado, si no el autor podra dejarlo en draft o marcarlo para revision y publicacion. Los moderadores tendran dentro de su modulo la revision de articulos a publicar y si lo acepta sera publicado, de lo contrario le mandara un correo al autor indicandole que ha sido rechazado, volviendo el articulo a modo draft. En caso el autor quiera hacer una modificacion a un articulo publicado, podra entrar y modificarlo, pero volvera a enviar una solicitud de revision sin afectar el articulo anteriormente publicado hasta que un moderador lo apruebe, reemplazando asi el anterior. Unicamente el autor de un articulo (y los moderadores) pueden modificar un artículo.

Todas las paginas deberan contar con un área de anuncio el cual mostrara una imagen con un link que escogera al azar de un listado configurado por el administrador en un modulo de anuncios. El sistema debera llevar control y poder entregar un detalle de cuantas veces se ha dado click a un anuncio y en que pagina se le ha dado click, pudiendo pedir un reporte por total de clicks para un mismo anuncio o por total de clicks por pagina (sin importar el anuncio en particular).



El sistema debera contar con un buscador que filtre combinadamente por categorias, autores, fecha de publicacion y/o texto y liste los articulos que cumplan con ese criterio, pudiendo ser ordenado por categoria, fecha de publicacion o nombre del autor.

Debera de llevarse un control sobre cuantas veces ha sido visitado un articulo, este dato aparecera unicamente a los autores, administradores y moderadores dentro del mismo artículo.

Nota: la evaluation del proyecto, si bien es uno mismo, sera 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 practicas 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 articulos deberan ser propuestos y aprobados por el catedratico por cada grupo.

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

Si un usuario no se ha logueado por más de un año sera deshabilitado automaticamente.

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

Preentregas: 2 preentregas, una antes del tercer parcial, otra despues 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:

Login, registro, cambio de password y administracion de usuarios en general. Manejo de perfiles registrados o anónimos a nivel de datos y correcta implementacion en el portal.



Creación de artículos.

Categorías, subcategorias: creacion de paginas, administracion y manejo dinamico.

Venta suscripciones y funcionalidad de paga.

Modulo de moderacion con reporte diario enviado por correo implementado en procedimiento almacenado y cesurado automatico implementado con trigger.

Búsqueda con filtros y ordenamiento.

Subir por lo menos una fotografia (blob) para los articulos

Servicio para entregar los datos de un articulo, listado de categorías y subcategorías por medio de un json

Actualización de articulo.

Manejo de perfiles de usuario

Modulo de anuncios.

FNBC, llaves e índices creados adecuadamente segun las buenas practicas y funcionalidad.

Invalidación de usuarios al año de no loguearse automatica.

Implementar un procedimiento almacenado con parámetros out y que tenga sentido dentro del uso del portal. (a ser aprobadas por el catedratico)

Implementar al menos una funcionalidad con queries dinamicos y con subqueries, por aprobar previamente por el catedratico.

Implementar al menos 2 vistas y 2 funciones que tengan sentido dentro del uso del portal. (a ser aprobadas por el catedratico)

Publicacion y dinamica de cambios.

Cualquier consideración y programación necesaria para cumplir con el planteamiento del requerimiento en un proyecto del mundo real.

Tecnologias: php y mysql (no ORM)

Documentación:

Modelo entidad relación

Manual funcional y técnico. El manual funcional debe tener un enfoque de usuario visitante con casos de uso.

Nota: Implementado con el proyecto de diseño web.

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 prerequisite de entrega, documentación completa. Cada alumno debe tener registrado al menos 10 commits por fase al momento de entregar cada fase.

Grupo maximo por afinidad de 2 personas.

=====

Funcionalidades y evaluación Diseño web:

Responsivo (móvil y desktop)

Template general del sitio

Homepage con listado de articulos y categorías

Sub home de categorias y subcategorias (2 plantillas)

Pagina de articulo en 3 plantillas distintas. Una de las plantillas de artículo debe implementar traer la informacion e inyectarse en la pagina por medio de un json y usando React

Pagina de creación de articulo

Pagina de registro y login

Modulo de moderacion

Pagina de perfil de usuario

Pagina de información e historia de la revista (pagina web)

Pagina de contacto con la revista (pagina web)

Página de preguntas frecuentes.

Modulo 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. Description semántica de la estructura html de cada pagina.

Set de htmls 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 restaran puntos por funcionalidades ya evaluadas y que no funcionen en la entrega final o en la segunda entrega.

Grupo maximo por afinidad de 2 personas. En caso sea necesario se considerara un unico grupo de 3 con un alcance mayor por definir.

=====

Fases: F1-> preentrega 1 - 27 septiembre- tarde

F2-> preentrega2 - 11 octubre - tarde

F3 -> entrega final. - 25 octubre - tarde

deben estar listos para presentar a la hora indicada por grupo. penalizacion 10 pts por cada 5 minutos de retraso. Codigo funcional tiene mayor pronderacion que la documentacion.

=====

Funcionalidades y evaluación Diseño web:

Responsivo (móvil y desktop) Desde F1

Template general del sitio F1



Homepage con listado de artículos y categorías F1

Sub home de categorías y subcategorías (2 plantillas) Fase 1 propuesta de plantillas F2 entrega

Página de artículo en 3 plantillas distintas. F1 entrega y aprobación, F2 implementación cambios. Una de las plantillas de artículo debe implementar traer la información e inyectarse en la página por medio de un json y usando React F3

Página de creación de artículo F1

Página de registro y login F3

Módulo de moderación F2

Página de perfil de usuario F1

Página de información e historia de la revista (página web) F3

Página de contacto con la revista (página web) F3

Página de preguntas frecuentes. F3

Módulo de anuncios. F2

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

Deberá implementar al menos un mixin utilizando buenas prácticas de implementación y con sentido adecuado dentro del proyecto. F1: propuesta, F2 implementación



Entregables:

Fuentes y build en ambiente de desarrollo, fuentes versionadas en repositorio. Cualquier consideración y programación necesaria para cumplir con el planteamiento del requerimiento. F1, F2 y F3

Documentación:

Manual funcional F2 y técnico con wireframes F2. Descripción semántica de la estructura html de cada página F3.

Set de htmls 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. F1, F2 y F3

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. F1, F2 y F3

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 maximo por afinidad de 2 personas. En caso sea necesario se considerara un unico grupo de 3 con un alcance mayor por definir.


=====

=====

Funcionalidades y evaluación DB:


Login, registro, cambio de password y administracion de usuarios en general. Manejo de perfiles registrados o anónimos a nivel de datos y correcta implementacion en el portal. F2

Creación de artículos. F1

Categorías, subcategorias: creacion de paginas, administracion y manejo dinamico. F1 

Venta suscripciones y funcionalidad de paga. F2

Modulo de moderacion con reporte diario enviado por correo implementado en procedimiento almacenado y cesurado automatico implementado con trigger. F3

Búsqueda con filtros y ordenamiento. F1 

Subir por lo menos una fotografia (blob) para los articulos F2

Servicio para entregar los datos de un articulo, listado de categorías y subcategorías por medio de un json F2


Actualización de articulo. F1

Manejo de perfiles de usuario F2


Modulo de anuncios. F3

FNBC, llaves e índices creados adecuadamente segun las buenas practicas y funcionalidad. F1

Invalidación de usuarios al año de no loguearse automatica. F3

 Implementar un procedimiento almacenado con parámetros out y que tenga sentido dentro del uso del portal. (a ser aprobadas por el catedratico) Propuesta F1, implementacion: F2

Implementar al menos una funcionalidad con queries dinamicos y con subqueries, por aprobar previamente por el catedratico. Propuesta F1, implementacion: F3

Implementar al menos 2 vistas y 2 funciones que tengan sentido dentro del uso del portal. (a ser aprobadas por el catedratico) Propuesta F1, implementacion: vistas F2, funciones F3, 

Publicacion y dinamica de cambios. F2

Cualquier consideración y programación necesaria para cumplir con el planteamiento del requerimiento

en un proyecto del mundo real. F3

Tecnologías: php y mysql (no ORM)

Documentación:

Modelo entidad relación F1

Manual funcional y técnico. El manual funcional debe tener un enfoque de usuario visitante con casos de uso. Funcional F2, Técnico: Fase 3

Nota: Implementado con el proyecto de diseño web.

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 prerequisite de entrega, documentación completa. Cada alumno debe tener registrado al menos 10 commits por fase al momento de entregar cada fase.

Grupo maximo por afinidad de 2 personas.

=====