

Ejercicio

Realizar un wlog (Wiki+Blog) con las siguientes características:

- Si se intenta ver una página que no existe se crea una nueva con las credenciales del usuario actual. Si no hay usuario se solicita login.
- Las páginas pueden ser públicas o privadas:
 - Públicas: aunque tienen un propietario las puede modificar cualquiera (sólo el propietario puede cambiar la propiedad de publica a privada).
 - Privadas: solo se pueden modificar por el propietario.
- Todas las páginas son visibles por cualquier usuario, incluyendo al usuario sin identificar
- Solo un usuario puede borrar sus páginas.
- Se pueden añadir comentarios a una página desde la vista de detalle:
 - Para añadir un comentario no hace falta estar autenticado. En ese caso hay que pedir el nombre de usuario.
- Se debe poder añadir y quitar etiquetas a las páginas.
- Se podrán etiquetar las páginas públicas y las privadas que pertenezcan al usuario actual
- Configurar la interfaz de administración para que muestre los modelos de la aplicación:
 - Mostrar los comentarios dentro de la vista de edición de página en la interfaz de administración.
- En el listado de paginas habrá que poner un enlace que permita cargar los comentarios de una entrada por ajax.
- Para nota: mostrar un formulario de login en cualquier página y validarlo por ajax para que solo se recargue la propia página si la validación fue correcta.

Esquema de urls sugerido (pero no obligatorio):

- /
 - salta a /wiki/
- /wiki/
 - listado de todas las entradas de wlog en orden inverso a su publicación con enlaces tanto para ver cada entrada como para cargar por ajax los comentarios de cada una de ellas. Y un enlace para acceder a la pantalla "principal" del wlog (que será Start)
- /wiki/Start/
 - muestra la página inicial
- /wiki/<pagina>/
 - Mostrar pagina (y los enlaces que corresponda)
 - Enlace para cargar por ajax los comentarios
 - Formulario para insertar un nuevo comentario
- /wiki/<pagina>/edit/
 - Editar pagina (si es publica o si es propiedad del usuario conectado)
 - Si se trata del propietario, enlace especial para "eliminar comentarios"
- /wiki/<pagina>/delete/
 - Borra una página y todos sus comentarios.
- /wiki/<pagina>/comments/
 - Recupera los comentarios de la página (URL auxiliar para ajax)
- /user/<username>/
 - Muestra un listado con las últimas páginas editadas por dicho usuario
- /wiki/tags/<tagname>/

- Muestra un listado con las páginas con dicha etiqueta, ordenadas por fecha de modificación
- /accounts/login/
 - abre una sesión de usuario
- /accounts/logout
 - cierra sesión

Se valorará:

- Uso adecuado de entornos virtuales.
- Uso de Python 3 y Django 2.
- Uso de migraciones.
- El uso de vistas genéricas donde corresponda.
- El uso de paginadores donde proceda.
- El uso adecuado de la herencia de plantillas.
- Uso de Bootstrap 4 y SASS.
- Elementos visuales (animaciones, comunicación XHR, ...).
- Degradabilidad y usabilidad.
- Navegabilidad.

Extras:

- Incluir scripts de configuración para puesta en producción con nginx, uwsgi, ...
- Desarrollo de API con DjangoRestFramework.
- Crear UI usando el API.
- Script con GULP para compilar el SASS a CSS.

A entregar:

- Repositorio en GitHub con el proyecto y un volcado de la base de datos. Incluir instrucciones para poder clonar el repositorio y poder ponerlo en marcha y probarlo.

Fecha límite de entrega:

- 5/02/2019