

Proyecto: Transversal 02 Mis Viajes

Cliente: Sr. A. Perez

Deadline: 20 de Diciembre 2019

Doctype: Valoraciones previas

Segun lectura de documentación aportada por el cliente, después de un estudio inicial del proyecto, el equipo de trabajo propone ciertas observaciones y/o correcciones a tener en cuenta durante el desarrollo del proyecto para estructurar así sus fases y facilitar tanto la planificación como su evolución.

En un primer analisis, valoramos que la complejidad del proyecto no es excesivamente alta. Por ello creemos que algunos puntos del proyecto son facilmente ampliables o mejorables. Basamos esta impresión en el hecho de que en este proyecto podemos utilizar bases de datos y por lo tanto, mediante la creación de vistas en la base de datos, se pueden preparar las consultas select que devuelvan la información precisa y necesaria sin necesidad de realizar operaciones o funciones en programación.

Consideramos que con las vista, no solo aumentamos la seguridad en la base de datos impidiendo que se pueda conocer la estructura de tablas de esta, sino que ademas, nos facilita mucho el trabajo ya que con PHP solo usando las consultas de las vistas ya preparadas, obtenemos la información unicamente necesaria para su proposito que por JSON podemos mandar a javascript para que manipule o muestre dinamicamente

Por ello, emos concluido que era esencial que dedicasemos mucho tiempo antes de iniciar el proyecto a definir claramente que estructura tendria la base de datos y de que forma interactuarian frontend (JS) y backend (PHP) con esa concreta estructura. Asi como definir que nomenclatura usariamos tanto en la base de datos como en las variables, funciones, metodes... para evitar conflictos en la fusion de las partes (tareas) realizadas por cada miembro del equipo. De ello han surgido estos puntos

1. Seguridad
2. Timeing y reparto de responsabilidades
3. AJAX para conexión javascript php
4. Diseño frontend con Bootstrap
5. Base de datos con vistas

1. Seguridad

A todo el grupo nos preocupa realizar un proyecto que cumpla al máximo temas de seguridad y protección de datos. Dado que en este proyecto se nos permite el uso de bases de datos en una estructura realmente parecida a la que tendria una pagina web real, consideramos oportuno implementar filtros de seguridad en cada capa de acción del website

CAPA 1 Base de Datos

definir usuarios con permisos especificos a sus funcionalidades

Segmentar las tablas aunque eso conlleve mas JOINS

Crear las vistas concretas para cada uso real del website

CAPA 2 PHP

Crear páginas diferentes para admin y usuarios corrientes, con conexión a la base de datos usando los especificos usuarios.

CAPA 3 Apache

Crear carpeta admin (como en wordpress) con permisos restringidos

* Mirar si eso implica que los usuario admin deban autentificar-se dos veces (apache y login de PHP)

CAPA 4 Frontend

Como enlazaremos con PHP via AJAX y JSON, crear un ID de consulta (campo oculto) para evitar infiltraciones

2. Timing y reparto de responsabilidades

Nos encontramos ante la disyuntiva de elegir entre dos metodologías de trabajo distintas.

Opcion 1 Pedagògica:

contras

Assignar las tareas a los miembros del equipo, segun su menor dominio de conocimientos de manera que con la ayuda de los compañeros aprenda y realice el la tarea	Al estar solicitando ayuda constantemente a los compañeros se entorpeceria mucho el trabajo dificultando el cumplimiento de los plazos de entrega
---	---

Opcion 2 Garantista:

contras

Assignar las tareas a los miembros del equipo, segun su mayor dominio de conocimientos, aprovechando sus puntos fuertes y garantizando así una mayor eficiente y productividad.	El proyecto puede al final muy bonito, pero se corre el riesgo de la no compensación de la carga de trabajo, así como el desconocimiento absoluto de partes del proyecto
---	--

Consideramos que por el equilibrio de puntos fuertes de los distintos miembros del equipo de trabajo, podemos realizar el proyecto con cierta agilidad si designamos las tareas en base a una perspectiva garantista. Por ello:

- Javi domina mucho Javascript y por ello se encargaria de la parte AJAX. No obstante, como elegimos priorizar la parte dinámica del frontend en contra del backend, sabiendo la carga de trabajo que eso supondria y por el analisis de contras descrito anteriormente; decidimos que él realizaria la estructura base y aunque no constas en el Jira, Alexey y Roger complementarían activamente esa parte del proyecyo
- Sobre base de datos y PHP tanto Alexey como Roger se defendian, però a Roger se le da mejor temas de diseño y CSS, por ello decidimos que ambos realizarían tareas de PHP dejando a Alexey la base de datos y a Roger el diseño y documentación.

Aclarar que ese reparto de funciones es una declaración de intenciones que no implica que las partes sean exclusivas de una persona y sobretodo que al final, todos debemos aprender y observar de las partes de los otros

Con ello, mirando el backlog del cliente, decidimos que la primera semana la invertiríamos en la parte visual del usuario y la segunda semana a la parte del admin dejando para el final, si sobrase tiempo, posibles auditorias de seguridad y/o metrica de la eficiencia en uso y consumo del website.

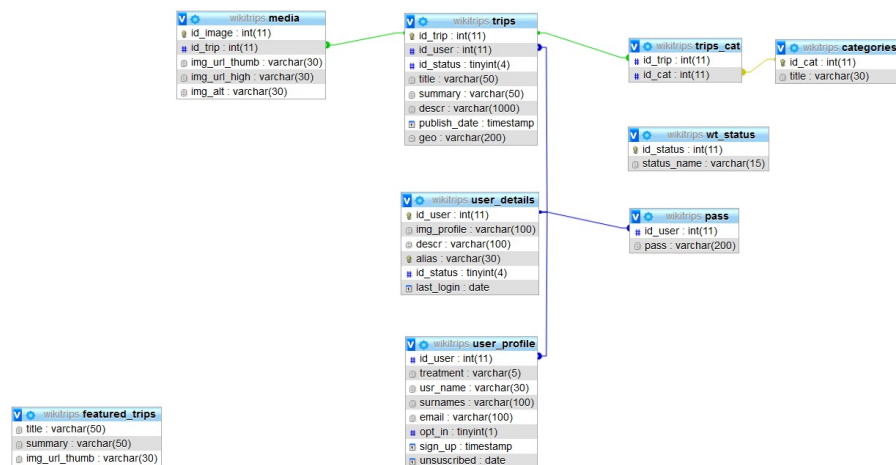
2. Estructura de la base de datos

Esta información esta incompleta, hay que actualizarla.

No se han credo en la BBDD las tablas ctal y como indicamos aquí. Los inidces no deberien ser INT(11), son muy largos.

Faltan los FK referenciados a las tablas, sino no podemos crear los JOIN con las vistas

Por ahira estamos aqui (exportacion de dissenyo de phpmyadmin, la del workbench es mejor)



usuario	pass	CREATE	ALTER	SELECT	INSERT	UPDATE	DELETE
root		SI	SI	SI	SI	SI	SI
ajr_media		NO	SI	SI	SI	SI	SI
admin				SI	SI	SI	SI
guest				SI	SI	SI	

TRIPS	AI	PK	FK	NULL	UNIQUE	OBSERVACIONES
id_trip	SI	SI		NO	SI	(índice de la tabla)
id_user			SI	NO		FK a USER_PROFILE (id_user)
id_status			SI	NO		FK a TRIP_STATUS (id_status) para borrador, publicado, reportado, usuario eliminado...
title				NO		Título (texto corto)
resum				NO		Texto para la pagina principal (resumen corto)
description				NO		Texto completo para la vista detallada
publish_date				NO		Mirar timestamp automático
geo				NO		Cazar IP?

TRIP_STATUS	AI	PK	FK	NULL	UNIQUE	OBSERVACIONES
id_status	SI	SI		NO		(índice de la tabla)
status_name				NO	SI	Borrador, Publicado, reportado, usuario de baja

TRIPS_CAT	AI	PK	FK	NULL	UNIQUE	OBSERVACIONES
id_trip		??	SI	NO		FK a TRIPS
id_cat		??	SI	NO	SI	FK a CATEGORIAS

CATEGORIES	AI	PK	FK	NULL	UNIQUE	OBSERVACIONES
id_categoria	SI	SI		NO		(índice de la tabla)
name			SI	NO	SI	FK a CATEGORIES (id_cat)

MEDIA	AI	PK	FK	NULL	UNIQUE	OBSERVACIONES
id_image	SI	SI		NO		(índice de la tabla)
id_trip			SI	NO		FK a TRIPS (id_trip)
img_thumb				NO		imagen pequeña para la vista lista
img_md						imagen media para vista featured
img_high				NO		imagen alta resolución para vista detalles
img_alt						para atributo ALT de etiqueta IMG

USER_PROFILE	AI	PK	FK	NULL	UNIQUE	OBSERVACIONES
id_user	SI	SI		NO		(índice de la tabla)
img_profile			SI	NO		url de la imagen de perfil
descr				NO		Descripción
alias				NO	SI	Será el mismo para nickname y login
id_status				NO		FK a USER_STATUS (id_status) para diferenciar usuario activo, baneado, borrado
last_login						último acceso

USER	AI	PK	FK	NULL	UNIQUE	OBSERVACIONES
id_user		SI	SI	NO		FK a USER_PROFILE (id_user)
treatment				NO		Sr, Sra, Señorita
usr_name				NO		Nombre usuario
lastname				NO		Apellidos
mail				??		Obligatorio??? Para recuperar contraseña? Limitar robots?
opt_in				NO		Booleano (0 no acepta, 1 acepta publicidad)
sign_up				NO		timestamp del momento en que se crea el usuario
unsuscribed				SI		timestamp del momento en que se da de baja un usuario

USER_STATUS	AI	PK	FK	NULL	UNIQUE	OBSERVACIONES
id_status		SI	SI	NO		FK a USER_PROFILE (id_user)
status_name						Activo, borrado, baneado

Vistas que hay que crear de sentencias SQL para TRIPS

- Trips destacados (featured) de la página principal [ultimos 4 / 4 mas votados / 4 mas cercanos]
- Lista de trips (version resumida) ??
- Lista de trips version detallada ??
- Información completa de un trip
- Lista de trips con filtros (un select distinto para cada filtro ??)

Vistas que hay que crear de sentencias SQL para USUARIOS

- Password de un alias de usuario concreto (WHERE)
- Datos perfil de usuario para apartado «editar perfil»

Vistas que hay que crear de sentencias SQL para ADMIN

- Lista de trips añadidos pendientes de aprobación
- Lista de trips reportados
- Lista de usuarios con trips reportados
- Creador de tablas para instalación
- Cargador de datos para demo