

## Diario Alberto García Sánchez

Fecha	Actividad
11/04/16	Comenzamos el proyecto, nos ponemos de acuerdo en cómo va a ser nuestro diseño front-end mediante el prototipado web, también escogemos las tecnologías con las que trabajaremos, siendo finalmente nodejs y mongo. Y nos repartimos el trabajo. Se crea el proyecto en GitHub donde subiremos nuestros cambios.
12/04/16	Comienzo por configurar nuestro proyecto en la parte de Back-end con tal de ya poder empezar a trabajar. Lo subo a GitHub, para que así podamos seguir desde ahí tanto mi compañera, como yo.
13/04/16 - 14/04/16	Al principio escogimos trabajar con JADE, pero después de escuchar sugerencias de nuestros compañeros decidimos cambiarnos a EJS (Que resulto ser mucho más sencillo de usar y más intuitivo.). Empezamos a añadir nuevas funcionalidades a la web y optimizamos el código ejjs.
15/04/16	Decidimos crear un fichero .js de base de datos para mongo para así poder parte de una base de datos común para poder hacer las pruebas. Hablamos de los campos que contendrá nuestro modelo usuarios, y cuando estamos de acuerdo creamos el modelo schema de usuarios. Para poder realizar la acción dropdown, debo descargarme librerías de internet para así poder hacer posible esa acción mediante el hover y no el onclick. Empiezo a realizar el front-end que decidimos que yo me encargaría. Uso como tecnologías principales HTML5, CSS3 y Bootstrap. Subo los cambios al GitHub.
18/04/16	Empiezo a informarme sobre el funcionamiento del EJS, para poder saber cómo poder realizar un renderizado correcto con el objeto que le estoy enviado, y poder recorrerlo para poder obtener solo los datos necesarios. Al comprender el funcionamiento del EJS ya puedo empezar a trabajar con variables de EJS para poder enviar más variables mediante el renderizado para así poder hacer la página más dinámica y adaptable según el tipo de usuario que entra a la página web.

19/04/16	<p>Decido informarme sobre como poder comprobar los datos enviados a través del formulario son correctos. Encuentro que hay ciertas librerías de EXPRESS que me permitirían poder realizar-lo desde el lado del servidor. Al consultar ciertas dudas con compañeros me doy cuenta que la primera comprobación de si los datos del formulario son correctos debe de ser mediante el cliente primero. Por ello decido emplear una librería de internet conocida como BootstrapValidaor, que mediante js y css se encarga de decirte mediante unos mensajes personalizados en el archivo js, como debería ser el campo donde tienes el error. En el cual desconocía el motivo, pero no me mostraba los mensajes de error en el sitio que debía.</p>
20/04/16 - 25/04/16	<p>Comienzo a informarme sobre Passport, una herramienta de express que nos permitirá controlar el acceso de usuarios a nuestra página web. Visualizo bastantes videos con tal de poder comprender bien que es lo que se está realizando para así poder llevarlo a cabo correctamente. Incluso habiéndome informado encuentro complicaciones sobre el funcionamiento del Passport, desconocía el motivo por el cual me daba los errores, como "Done is not a functions", entre otros. Después de estar arduamente durante dos/tres días mañana tarde buscando porque podría ser el error por foros tanto en castellano como en inglés, decido darle de nuevo un vistazo al código, pero esta vez mas calmadamente. Y descubro que el problema era que no cerraba el ")" donde debía sino con un parámetro mas, ello hizo que me saltaran los errores.</p> <p>Después de poder solucionar estos problemas, pude concentrarme en realizar la página de perfil del usuario donde podría modificar sus datos, eso incluye tanto el fron-end como back-end. En esta parte me encontré con algunos pequeños contratiempos, uno de los cuales fue que cuando quería poner un mensaje por defecto como placeholder en caso de que ese campo estuviera vacío en la base de datos, en caso contrario quería que tuviera el value de la consulta. Pero como igualaba el campo a "" entonces cuando no le ponía datos me borraba los datos anteriores ya guardados referente a ese campo. Por ello me di cuenta que si lo igualaba a "undefined" entonces si lograba el efecto deseado.</p> <p>Decidimos que el usuario debía tener una imagen de perfil cuando hicimos el prototype, por ello me puse a indagar por foros, para poder encontrar la librería que nos diera la solución para poder realizar nuestros objetivos. Al final lo pude encontrar "connect-multiparty" y emplear correctamente. En este paso me encontré otro contratiempo y fue el siguiente, al escoger el destino de la imagen que el usuario subía a nuestra página, le estaba indicando el path a la carpeta, por ello al ejecutarlo me daba error, el cual me decía que el archivo seleccionado es una carpeta y no esta vacía, después de buscar en foros de stackoverflow encontré la solución y resulto ser que debía ponerle el nombre del archivo en el cual se iba a guardar la imagen.</p> <p>Preguntando a compañero e informándome decido que es mejor llevar el header, menú y footer a archivos separados y realizar un include allá donde se necesite, de esta manera hacemos que el código sea más óptimo y limpio.</p>

26/04/16	Subo al GitHub todos los cambios realizados. Observo que mi compañera también sube el schema carrito, el que previamente ya discutimos como debía ser, la plantilla de pedido.
27/04/16 - 28/04/16	<p>Descubro por qué no se veían los mensajes del validador de formularios de bootstrap, el motivo era que en el HTML no utilizaba la class correcta para envolver cada input. Una vez hecho los mensajes ya aparecían, solo me quedo trabajar en el aspecto visual para que quedaran como queríamos.</p> <p>Observo que mi compañera ha tenido dificultades a la hora de estar realizando la parte del carrito por lo tanto decido ayudar-la. Al tener más experiencia con mongo y ejs, descubro la solución a realizar el carrito en un par de días, y su funcionamiento es parcialmente, dado que mostraba un artículo solamente y cuando ponía el siguiente desaparecía, eso era debido a que ponía un "[]" donde no tocaba, puede entenderlo y arreglarlo gracias a Jordi Andugar. Una vez realizado este cambio funciona perfectamente.</p> <p>Y entonces subo todo al GitHub.</p>
29/04/16	Me concentro en realizar un carrito visualmente decente por ello trabajo con bootstrap y css. También cambio el tipo de letra de "AlKa" por uno encontrado en internet (Gracias a las referencias de mi compañera Marta buena conocedora en temas de diseño).
2/05/16 - 6/05/16	Hago unas mejoras al código con tal de mejorar los roles administrador y usuario, de esta manera consigo que cada uno respete su rol y vea lo que puede ver. También me informo de como poder subir nuestro proyecto a Openshift (Referenciado por mi compañero Juan), sigo un tutorial que me referencia y puedo subirlo sin problemas, únicamente al subirlo me doy cuenta que la base de datos ya no es "AlKaTiendaInformatica" sino "alkatiendainformatica". Restando eso no hay más problemas (o eso pensaba).
9/05/16	Añado las funcionalidades pertinentes para poder crear los filtros para los productos de la página, corrijo el tamaño de la imagen del carrito, añadir el campo al schema de Puerto (dado que mi compañera de proyecto me pide que lo necesita para poder seguir con el proceso de factura del carrito). Al empezar a tratar de hacer el responsive correctamente en las 3 medidas conocidas como xs(Móviles), sm(Tabletas), lg(Ordenadores Restantes), me doy cuenta que no encaja bien el bootstrap por defecto ya que no hace ciertas funcionalidades bien en las tabletas, como mostrar la hamburguesa cuando debería, por ello me informo de cómo puedo llegar a solucionar este problema y veo que la solución es descargar y personalizar el bootstrap, cambiando las medidas max de las sm de 757px a 800px.

10/05/16	Un compañero de clase nos informa de un plugin para el nodejs que se encargaría de aligerar la carga al arrancar nuestra página web, este plugin se llama "Compresion". Hago mejoras en las funciones que tienen que ver con el rol Administrador.
11/05/16	Solucionado el problema que no dejaba subir productos, era una variable mal puesta. Corrección rutas administrador.
12/05/16 - 13/05/16	Ya que aparte del rol de Administrador tenemos, los roles de usuario registrado y no registrado, decidimos que los usuarios no registrados no deberían poder ver el carrito de la compra tanto en el menú como en los productos. Veo que mi compañera actualiza las funciones pertinentes al proceso de compra, recogiendo los datos del carrito ya pusheado con anterioridad.
14/05/16 - 17/05/16	Son los días que he dedicado a informarme y hacer que funcionara correctamente los buscadores usando los plugins "Typeahead" y "TypeSearch".
18/05/16 - 21/05/16	Son días empleados a crear una función nueva ConfiruaTuPC y su ejs respectivo, con el jquery respectivo. Tuve ciertos problemas al principio sobre cómo moverme por el DOM, pero tras unas cuantas pruebas poco a poco me fui adaptando más y entendí como navegar correctamente a través del (en parte gracias a ayudas puntuales de mi compañera), luego tuve problemas sobre cómo hacer para recoger todos los valores totales de los productosTotales al hacer click al cambiar la cantidad o el select. Al final lo pude lograr gracias a búsquedas por internet y aprender de nuevas funciones, las cuales desconocía anteriormente.
22/05/16	Actualizo el diario, soluciono los últimos problemas al realizar el ultimo merge. Y termino el diagrama de UML, también amplio el md del proyecto. Y grabo el video con la demostración del funcionamiento de la web.