

On The Code

EVERY CODE HAS A STORY

On The Code

Iñaki Caballero
Julen Castillo
Julen Prieto
Jon Santos

ÍNDICE

- [EMPRESA E INICIATIVA EMPRENDEDORA\(EIE\)](#)
- [Idea de negocio](#)
- [Método Scamper](#)
- [DESPLIEGUE DE APLICACIONES WEB \(DAW\)](#)
- [Repositorio principal](#)
- [Esquema de ramas - GitHub](#)

EMPRESA E INICIATIVA EMPRENDEDORA(EIE)

IDEA DE NEGOCIO

La idea de negocio es el inicio de toda actividad emprendedora. Es el producto o servicio que queremos ofrecer al mercado. Constituye la oportunidad de negocio, y la clave del éxito.

Somos estudiantes de Desarrollo de Aplicaciones Web y nuestra empresa "On The Code" como no podía ser de otra forma, se dedica al desarrollo de aplicaciones y páginas web. Nuestro abanico de necesidades a cumplir es amplio, ya que hoy en día todos los negocios están informatizados, y toda empresa tiene una página web en la que publicitarse o dispone de alguna aplicación web con la que gestionar, automatizar o simplemente facilitar tareas repetitivas.

Nuestro objetivo es llegar a las pequeñas y medianas empresas , que necesitan una web o una aplicación, ya sea para publicitarse y llegar a más público o para facilitar las tareas de los trabajadores. Somos programadores serios y hacemos del trabajo de los demás algo más sencillo.

Para ello, nuestro primer proyecto lo haremos en colaboración con los miembros de ARI ("Sevron"). En él desarrollaremos una aplicación web para gestionar un tranvía. El propósito del proyecto es crear una aplicación web desde 0 con la cual recopilar datos importantes y poder gestionar el movimiento de un autómata. Para ello ponemos en juego nuestra creatividad y nuestras ganas de avanzar en la programación, ya que la aplicación tiene que ser simple, intuitiva y atractiva.

Hoy en día cualquier persona es capaz de generar una página web copiando una plantilla, o simplemente robando el código de terceros. Nosotros nos dedicamos a esto porque nos gusta comernos la cabeza y encontrar la mejor estructura e intentamos que cada página sea diferente a las anteriores.

Para diferenciarnos del resto utilizamos diseños minimalistas e intuitivos, las aplicaciones son generalmente para trabajar e intentamos que los diseños no estén sobrecargados para facilitar y hacer más ameno el trabajo de los operarios.

MÉTODO SCAMPER

- Definición del problema:

El problema a resolver puede ser encontrar o mejorar un producto o servicio, proceso, etc. Para la definición del problema se pueden utilizar otras técnicas, como checklist u otros similares.

- Formulación de preguntas:

En este punto, formulamos las preguntas utilizando los verbos de acción incluidos en la palabra SCAMPER y se procede a proponer respuestas a dichas preguntas.

-Sustituir:

¿Qué podemos sustituir ?

Podemos sustituir el sistema operativo y el software con el que trabajamos, y así lograr trabajar con un software libre y gratuito.

¿Podemos cambiar la forma, color, tamaño...?

Siendo una empresa de desarrollo web, podemos adaptar nuestro producto al gusto y las necesidades del cliente, ya sea el color o el dispositivo desde el que se va a trabajar.

¿Qué es posible sustituir para reducir costes?

Como ya hemos mencionado anteriormente, podemos usar aplicaciones y sistemas operativos gratuitos par abaratar el coste.

-Combinar:

¿Podemos combinar productos o partes de éstos?

Podríamos llegar a reutilizar código de distintos proyectos.

¿Qué combinación de elementos generarían una reducción de costes?

Una buena planificación y organización entre los grupos podía llevar a tener los proyectos finalizados en menos tiempo.

-Adaptar:

¿Podemos adaptarlo a otro uso o utilidad?

Este proyecto podríamos aplicarlo en cualquier tipo de transporte público y ferroviario.

-Modificar:

¿Qué podemos modificar para conseguir, obtener, ..?

Hemos creado 4 prototipos para enviárselos al cliente y así poder adaptarlo a su gusto y necesidades.

-Poner en otros usos:

Puede ser usado por cualquier tipo de usuario preparado y formado para la gestión de transportes.

-Eliminar o minimizar:

¿Es posible reducir desperdicios?¿Cómo?

Al ser una empresa de desarrollo de aplicaciones web, no generamos desperdicios. Podemos reutilizar código entre proyectos y en cuanto a residuos electrónicos, podemos donar el material reutilizable y reciclar el obsoleto.

-Reordenar o Invertir:

¿Puede hacerse más grande/pequeño/ligero/pesado/...?

Nos esforzamos para hacer el código lo menos pesado posible para que las páginas no pesen más de la cuenta para que su ejecución sea óptima.

DESPLIEGUE DE APLICACIONES WEB(DAW)

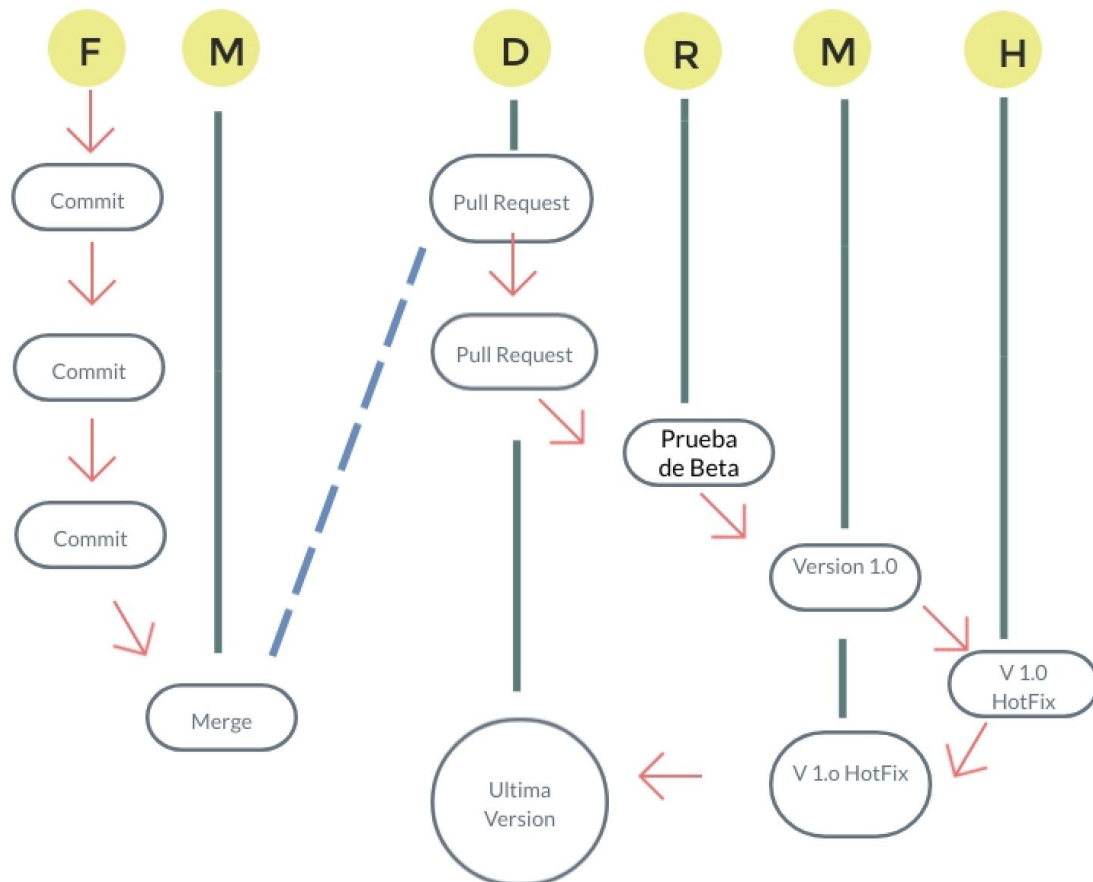
Repositorio principal

El repositorio principal se almacenará en la cuenta del grupo alojada en Github:
<https://github.com/OnTheCode-Co>

El nombre del repositorio se generará con la plantilla Reto-X donde X corresponderá al número del reto.

Esquema de ramas - GitHub

- Hemos decidido trabajar con las siguientes ramas en equipo:
 - Master: Rama principal donde alojamos las versiones estables.
 - Development: Rama secundaria en la que subimos el trabajo diariamente. Será la última versión aunque esta no sea estable.
 - Release: Rama en la que probamos las versiones que consideremos estables para comprobar si realmente lo son, y si así lo fuesen se juntaría al master.
 - Hotfix: Rama de versión. La usaremos para hacer correcciones puntuales.
- En cuanto al trabajo personal cada uno de los integrantes hará un 'fork' a su GitHub. Usaremos las siguientes ramas:
 - Features: Aquí escribiremos nuestro código con nuevos métodos y características que queramos implementar.
 - Master: Una vez aseguremos que nuestro trabajo es apto para subirse con el resto del proyecto, juntaremos lo que tengamos en la rama feature con esta rama y la subiremos a la rama "Development" del GitHub conjunto.



F: FEATURES
M: MASTER
D: DEVELOPMENT
R: RELEASE
H: HOT FIXES

Al final del día, cada estudiante trabajará en su rama Features. Una vez acabado lo subirá a su rama master y de ahí a la rama Development del equipo.

Nomenclatura de los commit

Los commits se ejecutarán con la siguiente nomenclatura :
XXdescripción_breve , donde XX serán las iniciales de cada alumno. Se intentará explicar brevemente lo que se ha cambiado respecto a la versión anterior para saber qué se ha cambiado y en caso de error poder volver atrás.

Definiremos una versión nueva cuando esta sea una versión estable. Una vez definido el número de versión se le asignará el nombre de un monte vasco, ej: 1.3 Aitzgorri

Cómo marcamos qué momentos son una versión nueva.

Al final de la semana, nos reunimos para comentar los cambios que hemos realizado y definimos según los cambios si debemos hacer una versión pequeña o una versión grande, también

cuando hayamos completado los mínimos establecidos para cada apartado lanzaremos una versión estable del programa

Quién se encarga de ser el administrador: ¿Todos, cada vez uno?

Nos ocuparemos de gestionar el Github entre todos. En caso de conflicto nos reuniremos para solucionar los posibles errores y llegar a un acuerdo.

Comandos GIT

Inicializar el repositorio en Github. (Desde comandos -> git init)

Clonar el repositorio el los equipos -> git clone <https://github.com/OnTheCode-Co/Reto1>

Mostrar las conexiones al repositorio remoto -> git remote -v

Descargar cambios del repositorio remoto -> git pull

Añadir ficheros para Commit-ear (El "." significa todos los archivos) -> git add .

Commit de todos los archivos añadidos -> git commit -m "Esto es un commit"

Subir al repositorio remoto los cambios -> git push

Mostrar las ramas, las rojas son las remotas -> git branch -a

Moverte a una rama -> git switch nombreDeLaRama / git checkout nombreDeLaRama

DISEÑO DE INTERFACES WEB (DIW)

DISEÑO DE NUESTRA APLICACIÓN

Después de una primera reunión con los miembros de ARI, obtuvimos un breve resumen de cómo tenía que ser la aplicación. La idea tenía que ser simple, sin demasiados colores y funcional.

Una vez hecho el boceto a papel, se decidió hacer varios prototipos. Para ello trabajamos individualmente, haciendo cada uno una página distinta, sin coger ideas de los demás. El resultado fué el siguiente:

Diseño 1 - Jon Santos:

Gestión de tranvía

Logeado como: Administrador

Información sobre el tranvía:

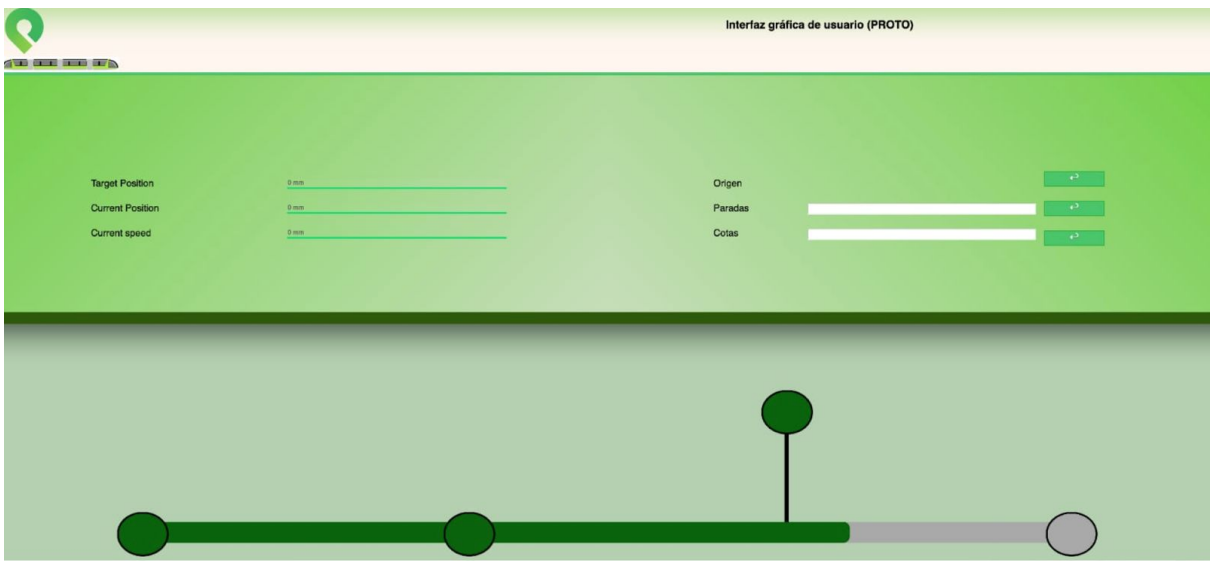
Current Position:	000
Current Speed:	000
Target Position:	000

ORIGEN

Introduce una parada (1-4):

Introduce una cota (0-100):

Diseño 2 - Iñaki Caballero:



Diseño 3 - Julen Castillo:



Diseño 4 - Julen Prieto:

GESTIÓN DE AUTÓMATA CON TIA PORTAL
GESTIÓN DEL AUTÓMATA

1 - 4

1 - 100

ERROR

BUSSY

READY

IN POSITION

Información

Destino del tranvía:

Posicion actual del tranvía:

Velocidad actual:

@OnTheCode.Co 2019/2020
 Product designed for ARI

Los prototipos fueron enviados al equipo de ARI para que eligiesen uno y añadiesen o quitasen elementos. Se enviaron todas las propuestas sin nombre y sin ningún tipo de indicación sobre su funcionamiento. Así se evitaban prejuicios y sabríamos si la aplicación era intuitiva o no.

Se eligió el último diseño y realizaron las correcciones sobre la imagen. Se nos pidió introducir una barra de progreso como en los otros diseños, cambiar la gama de colores y modificar la división de los errores.

modificar color, nos quema los ojos

GESTIÓN DE AUTÓMATA CON TIA PORTAL
GESTIÓN DEL AUTÓMATA

podeis poner todo del mismo color verde como este ->

1 - 4

1 - 100

ERROR

BUSSY BUSY

READY

IN POSITION

Información

Destino del tranvía:

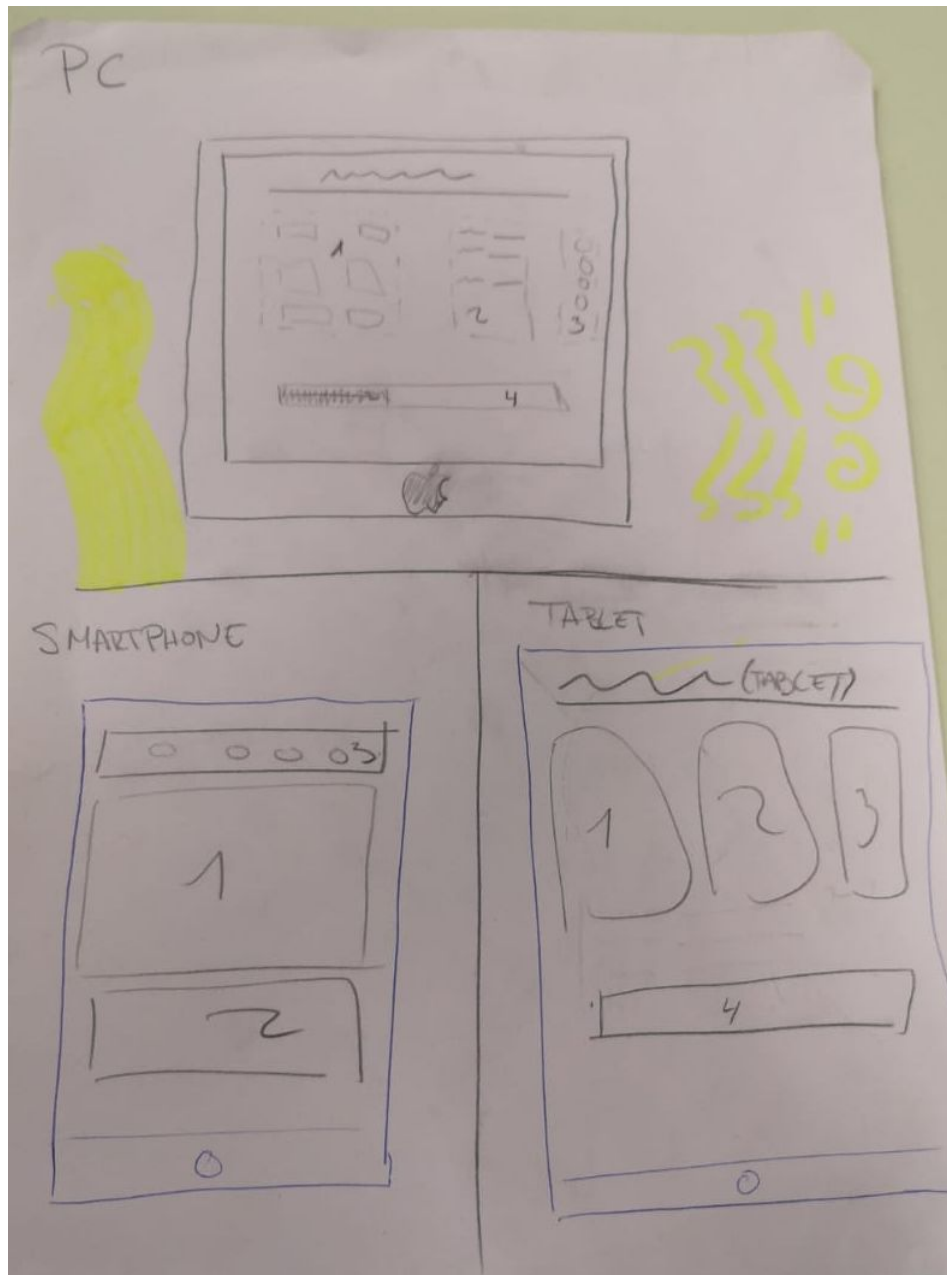
Posicion actual del tranvía:

indicar que es en mm

Velocidad actual:

solo 2 columnas (roja y verde)
 Se mueve? de ser asi, añadir indicadores de paradas como lo tienen los otros modelos.

Con las nuevas pautas se reorganizó la página en papel, se diseñaron las versiones de tablet y móvil. Para ello utilizamos papel y boli.



LAYOUT FINAL

Una vez realizados los prototipos individuales y el boceto a papel y boli, el resultado de la aplicación web es esta :

Como ya se ha dicho anteriormente, se buscaba que fuese intuitiva y simple, sin colores vivos. En la guía de estilos y en el manual de usuario se explicará el funcionamiento de la aplicación con más detalle.

