

Una silla de tres patas

Aplicación web de preguntas y respuestas de Aergibide S.L.

Documentación

Equipo 4

Fecha de publicación 23 de noviembre del 2018

Versión 1.0

Contenido

|  |  |
| --- | --- |
| **Introducción** | **3** |
| Sobre nosotros | 3 |
| El reto | 4 |
| Logros del reto | 4 |
| **Inicio del proyecto** | **5** |
| Planteamiento y Reunión ARI | 5 |
| Reunión del equipo y Diseño inicial | 5 |
| Herramientas | 8 |
| **Desarrollo** | **9** |
| Diseño y desarrollo de la base del HTML | 9 |
| Investigación del Simulador y TIA Portal de Siemens | 9 |
| Desarrollo JavaScript y JQuery | 12 |
| Revisión del diseño inicial | 12 |
| Desarrollo HTML y CSS | 13 |
| Detalle de las funcionalidades | 14 |
| **Propuestas para el futuro** | **15** |
| Tareas pendientes | 15 |
| Mejoras futuras | 15 |
| **Bibliografía** | **15** |
| Documentar el proyecto | 15 |
| Información sobre las herramientas | 15 |

Introducción

Sobre nosotros

Somos Aitor Bringas Martínez, Jon Xu Jin y Sebastián Zawisza Abreu, miembros del equipo 4 del grupo 2GA-DAW de Egibide Arriaga.

Mediante contrato de equipo, formalizamos el equipo 4 y con denominación TRIPOD.

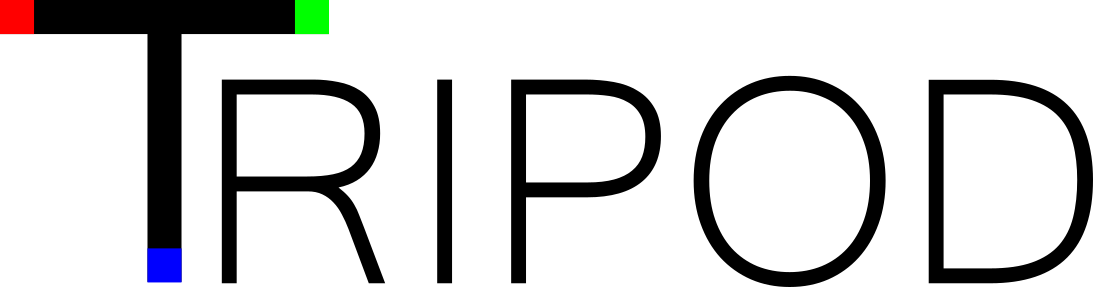
Documentamos el proceso completo en este documento del desarrollo de la aplicación web que maneja un autómata desarrollado por el equipo de ARI, para así dejar constancia de los procesos, detalles de las funciones, el uso de las herramientas y de los ensayos que nos ha permitido el desarrollo y el testeo de esta aplicación.

Nuestro repositorio Git:

<https://github.com/tripod-egibide/reto2>

También pueden visitarnos en:

<https://tripod-egibide.github.io/>



El Reto

Nuestro reto es el siguiente:

Hace tan solo unos meses que tuvisteis la idea de crear una empresa de desarrollo de

aplicaciones web y antes que la hayáis creado ya tenéis un primer cliente. La empresa del

sector aeronáutico Aergibide SL os ha contactado para desarrollar una aplicación que

solucione el grave problema que tienen en su departamento de I+D+I: la gestión del

conocimiento. Los ingenieros de la empresa (son más de 150 personas) trabajan de forma

muy autónoma, y en muchas ocasiones desconocen que hay otros compañeros que ya se

han enfrentado a esos problemas o son expertos del tema que podrían ayudarles.

Le habéis comentado a la empresa que afrontaréis el reto y que en tan solo unas

semanas tendréis desarrollada y operativa una aplicación web para resolver el problema.

Internamente tras el análisis realizado, habéis pensado que la mejor solución podría ser

una herramienta similar al ya conocido StackOverflow.

Las funcionalidades mínimas con las que deberá cumplir la herramienta son:

•Registro y Login de usuarios

•Publicación de preguntas (título, descripción, tema,...)

•Publicación de respuestas (pueden incluir archivos adjuntos)

•Votar respuesta favorita

•Perfil de usuario

No obstante, podréis considerar añadir otras funcionalidades como: marcar pregunta

como favorita, disponer de un perfil más completo de usuario (ver historial de preguntas y

respuestas publicadas, etc), notificaciones, sistema de puntuación/gamificación de

usuarios, gestor de documentos, ...

Al mismo tiempo deberéis constituiros como empresa cumpliendo con todos los requisitos

legales y desarrollando las actividades necesarias para la buena marcha posterior de

vuestro proyecto empresarial (valorar necesidades de financiación, hacer un estudio del

mercado, valorar posibles nuevos productos para vuestra cartera, decidir una estrategia

de marketing, etc).

Logros del reto

Desarrollo e instalación de una aplicación web completa (lógica de presentación, de negocio y de datos).

Inicio del proyecto

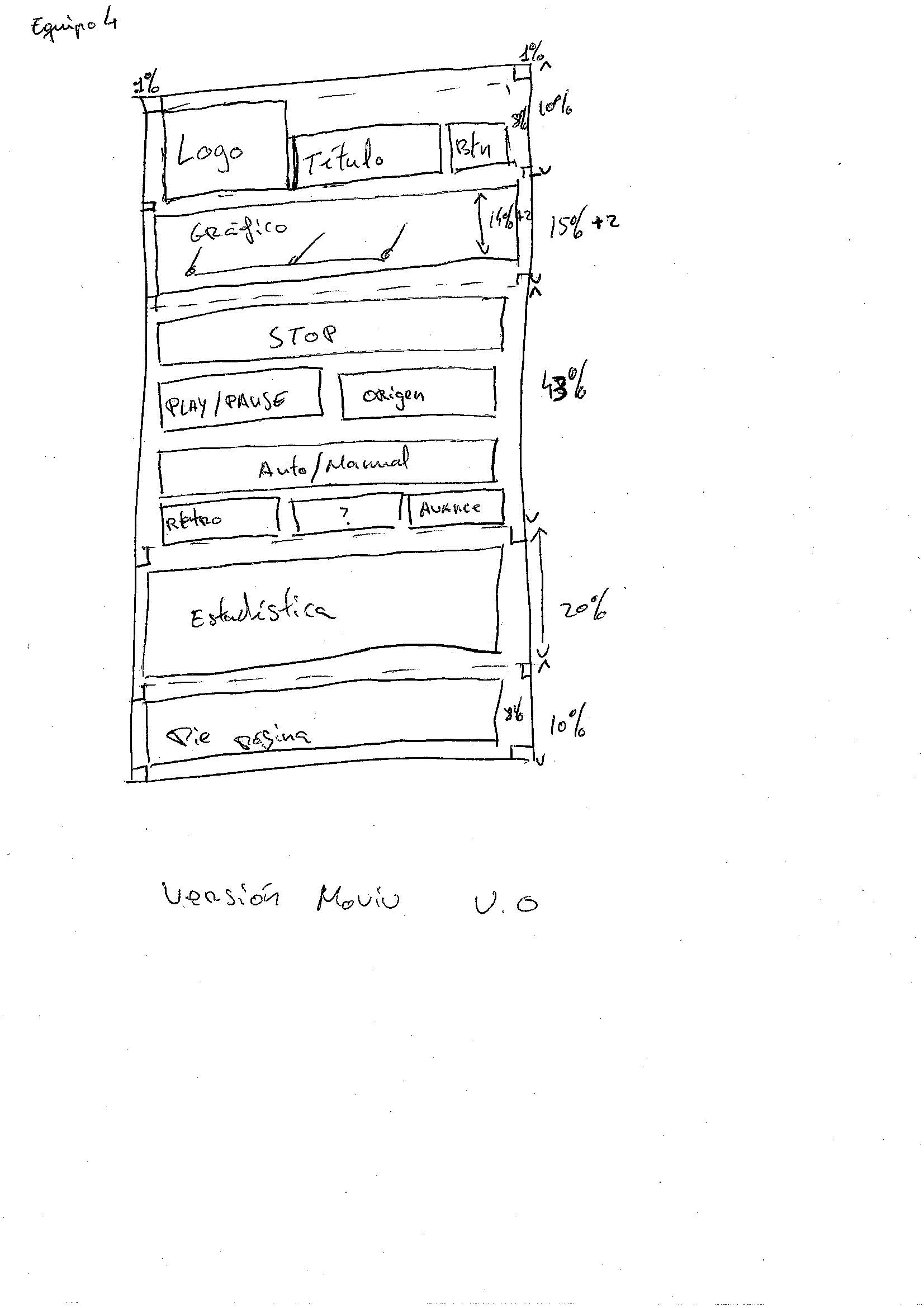
Planteamiento y Reunión de equipo

Realizamos una primera reunión de los miembros del equipo para investigar y explorar como usuario, la web de Stackoverflow, para obtener una primera idea de cómo será nuestra aplicación web.

Reunión del equipo y Diseño inicial

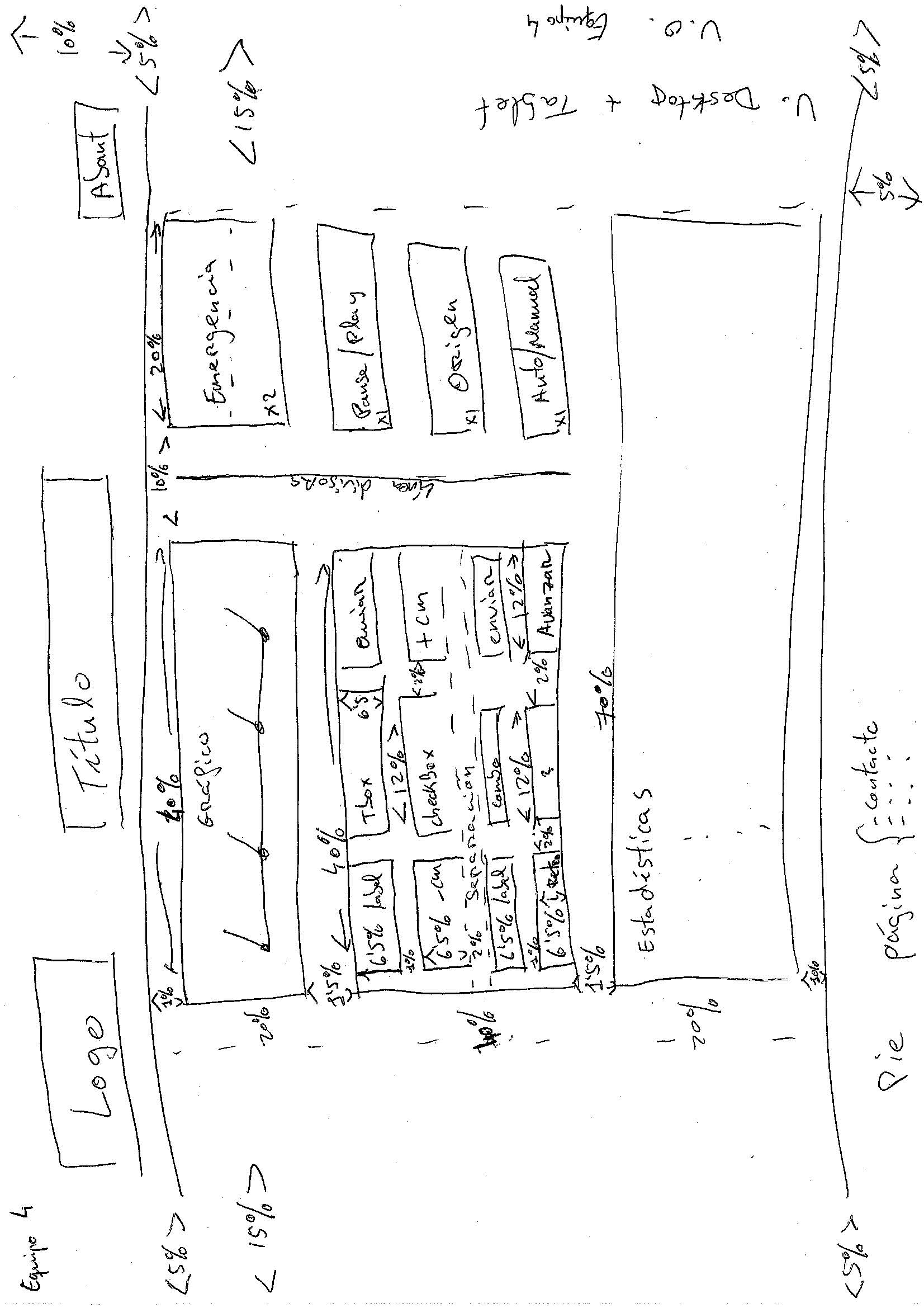
Tras la investigación de la web de Stackoverflow, realizamos una segunda reunión de los miembros del Equipo Tripod para por fin empezar a plantear cómo sería nuestra aplicación Web y diseñar un cuaderno de equipo para plasmar la planificación.

Tras una intensa lluvia de ideas sobre el diseño de las interfaces, optamos por una combinación de 3 ideas y plasmarlo en 1.

Realizamos varios Sketch intentando combinar las 3 ideas y el resultado fue lo siguiente:

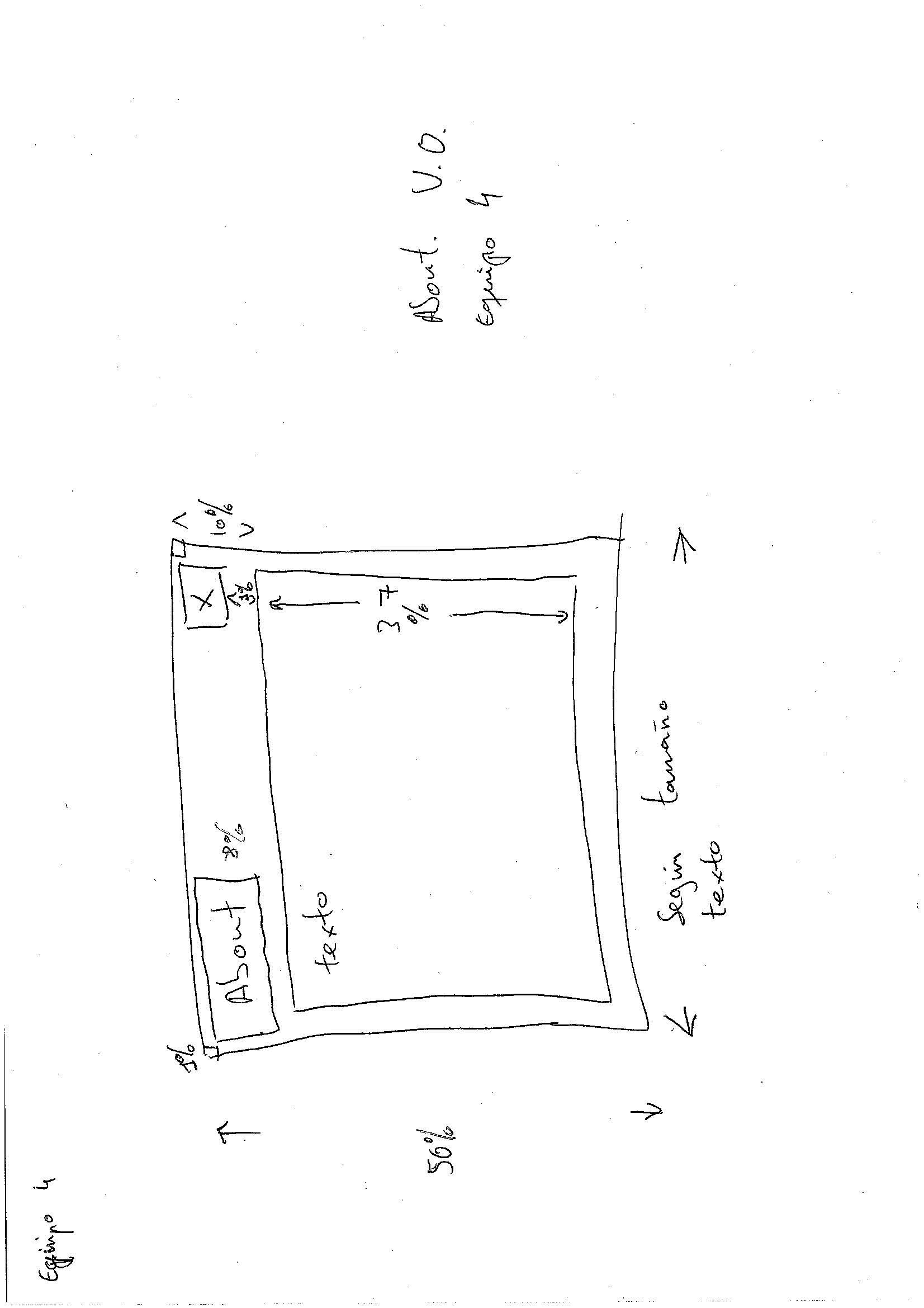
Versión SmartPhone y Tablet

Versión Ordenador



Versión común en todos los navegadores:

El about será una ventana flotante



Quedándonos todos conformes con el Sketch y llevarlo al desarrollo.

Herramientas

Por decisión en conjunto de todos los miembros del equipo 4, se ha decidido emplear las siguientes herramientas para el diseño, desarrollo y ensayos de la aplicación web de control del autómata antes de su implementación al entorno de uso real:

|  |  |
| --- | --- |
| **Herramientas empleadas en el proyecto** | |
| Papel y boli | Diseño inicial de los Sketch (se ha empleado boli por falta de lápiz). |
| Trello | Planificar las tareas |
| Hoja de seguimiento | Seguimiento de las tareas y las asistencias de los miembros. |
| Atom | Aplicación para el desarrollo de los códigos de JavaScript de manera colaborativa y en vivo, permitiendo editar en caliente todos los desarrolladores. |
| WebStorm | Aplicación para el desarrollo de los HTML y CSS. |
| GitHub | Plataforma para el control de versiones de los códigos diseñados. |
| NotePad++ | Editor de textos y códigos en formato reducido, para corregir códigos sencillos para pruebas. |
| Google Docs | Para el diseño de todas las documentaciones. |
| Firefox  Google Chrome | Navegador predeterminado para el uso y ensayo durante todo el proyecto. |
| HotGloo | Aplicación web para el diseño del Wireframe, exportación del proyecto en .png y .html |
| Pack iconos | Obtenidos en Material.io para el decorado de la aplicación web. |
| Jquery-3.3.1.min.js | Librería JQuery para el funcionamiento de las comunicaciones con el autómata. |
| Google Fonts | Fuentes personalizadas del aplicativo. |

Desarrollo

Diseño y desarrollo de la base del HTML

Iniciando esta segunda semana, empezamos a desarrollar la estructura de la aplicación y la parte fija de nuestra aplicación web, una parte que no se modificará en ninguna de las páginas futuras, permitiendo mantener una estética fija de la cabecera y el pie de la página.

Estructura:

D

D

D

D

D

D

De la siguiente manera, en php, podemos añadir nuevas páginas sin necesidad de reescribir continuamente el header y el footer:

El Código debe estar dentro del <body>

Código justo debajo del <body>:

<?php

include "../partefija/header.php";

?>

Código justo encima del </body>

<?php

include "../partefija/footer.php";

?>

Desarrollo de HTML y CSS

El diseño de un html base como hemos indicado anteriormente, nos permite diseñar otros html sin apenas esfuerzo, ya que existe una carcasa que únicamente tendríamos que rellenarlo con nuevos códigos.

A continuación plasmamos una muestra de nuestro index, en ello hemos reflejado la mayor parte de los códigos de la web del proyecto para llamar las funciones del JavaScript, Ajax, JQuery y PHP.

Básicamente, aquí declaramos todos los campos y botones necesarias sin importar apenas cómo distribuirlos de manera gráficamente.

Código html:

<script type="text/javascript" src="js/botones.js"></script>

<div class="row">

<div class="col66">

<div class="animacion">

<div class="col100">

<button onclick="go()">Go</button>

<button onclick="stop()">Stop</button>

<button onclick="fase12()">fase12</button>

<button onclick="fase23()">fase23</button>

<button onclick="fase31()">fase31</button>

<div class="b" id="goAnimacion">o</div>

</div>

Código JavaScript, Ajax y JQuery:

Código PHP:

Código CSS:

Desarrollo JavaScript y JQuery

Empezamos a desarrollar los códigos en JavaScript y JQuery.

Así de primeras, tuvimos que acudir a San Google, para buscar información de cómo comunicarnos con JQuery, y pudimos dar con los siguientes códigos para enviar y leer información del simulador:

Enviar datos:

$($.ajax({

type: "POST",

data: '"webdata".' + variable + ' = ' + valor

}));

Leer datos:

$.get("htm/TargetPosition.htm", function(result) {

targetPosition = result.toString();

$("#tar\_pos").text(targetPosition);

});

Llamando desde una función y que lea desde otro html las variables que se desee

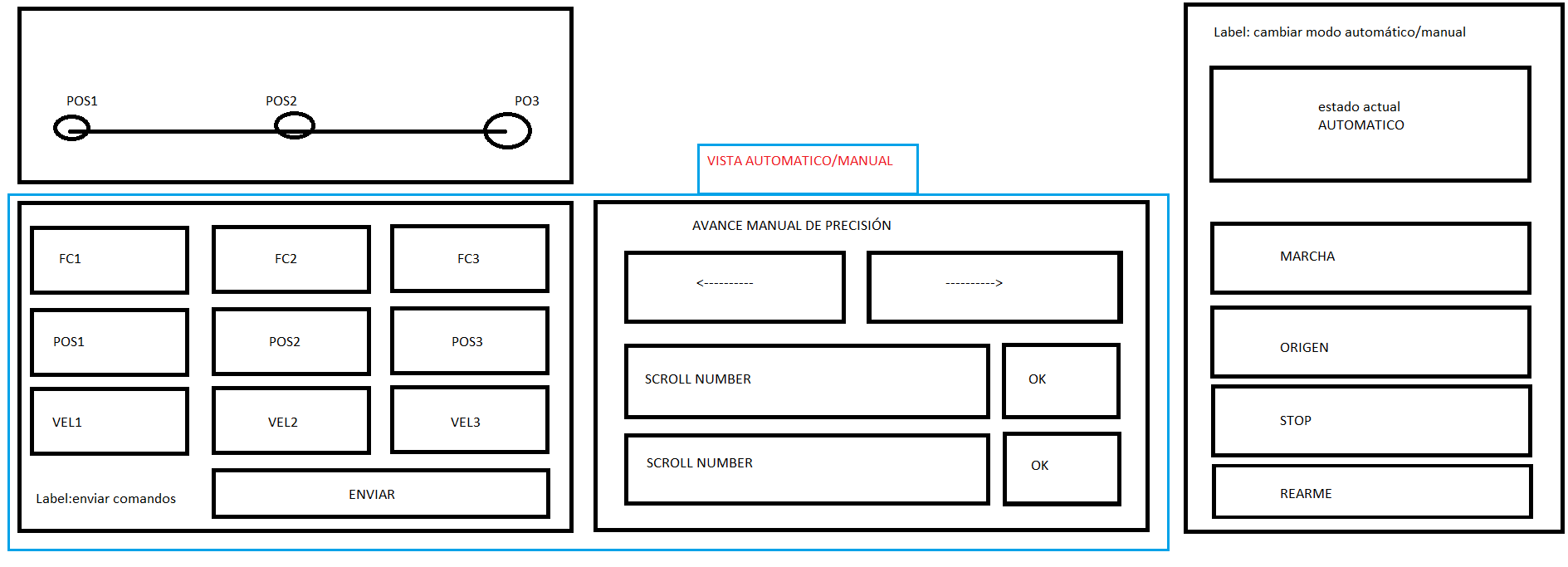
Lógicamente, hay más códigos, pero ya básicamente son de lógica para el manejo de las demás funciones, así como códigos claves, son los arriba indicados.

Aquí hemos plasmado una pequeñas muestras de cómo enviamos y obtenemos los datos, para comprender mejor y ver el código completo, recomendamos realizar una lectura del código fuente de la aplicación web, ya que está comentada y detallada de mejor manera.

Revisión del diseño inicial

Realizar cambios durante un proyecto es algo bastante habitual si dependemos de otros equipos u organizaciones, en nuestro caso, hemos tenido que modificar parte del proyecto por los cambios que han realizado el equipo de ARI.

A continuación, adjuntamos el nuevo Sketch, adaptado a las nuevas variables que nos han facilitado:



Detalle de las funcionalidades

Es habitual documentar durante el proceso del desarrollo de una aplicación.

Así como otros proyectos, pueden encontrar más información sobre las funcionalidades explorando nuestros proyectos.

//variables de lectura

let targetPosition, currentPosition, alarm1, auto;

//variables calculadas

let posPorcentage, contadorCiclos = 0,

ultimaAlarmaTiempo = null;

En este caso, todo el proyecto se puede explorar empleando en NotePad++, que lo pueden descargar desde el siguiente enlace:

<https://notepad-plus-plus.org/>

Propuestas para el futuro

Tareas pendientes

Al haber un plazo muy corto y problemas con el equipo de ARI, hubo parte de las funcionalidades que no se ha podido mejorar/corregir, tales como:

El entorno gráfico: la certeza de la flecha que señala la barra de progresos.

Los íconos: Una mejor adaptación de los iconos.

Estadísticas: Falta las florituras.

Mejoras para el futuro

Así como plan de futuro, se podría corregir/implementar las tareas pendientes y además añadir nuevas funcionalidades, tales como:

Autentificación al acceder a la web.

Una BBDD para los datos de estadísticas.

Agregar accesibilidad para personas con discapacidad.

La versión final de la aplicación web que se ha entregado por el fin de plazos es funcional a nivel del enunciado, por lo que pensamos que cumple correctamente lo que nos exigen.

Bibliografía

Fuentes consultadas

Para el desarrollo de la aplicación web de preguntas y respuestas, se ha consultados las siguientes fuentes:

HTML y CSS:

<https://www.w3schools.com/>

JavaScript, Ajax y JQuery:

<https://developer.mozilla.org/es/docs/Web>

PHP:

<https://secure.php.net/manual/es/>

Gracias a las aportaciones de dichas fuentes, pudimos llevar a cabo el desarrollo del proyecto reto 2, por lo que agradecemos de corazón a las fuentes y personas que nos han ayudado.

Información sobre las herramientas

En este apartado, detallamos un poco de para qué se usan y dónde descargarlos.

Trello:

<https://trello.com>

Es un software de administración de proyectos con interfaz web para organizar proyectos, permitiéndonos registrar actividades del día a día, así como establecer fechas, etc

Hoja de seguimiento:

Básicamente es una hoja, física o virtual, para registrar las personas integrantes de un proyecto y así llevar un control de asistencia y puntualidad de los miembros.

Atom:

<https://atom.io/>

Es un editor de códigos de fuente de código abierto con soporte multiplataforma e integrado con Git para el control de versiones. Nos permiten trabajar en vivo con varios desarrolladores a la vez, permitiéndonos así desarrollar y revisar a la vez, uno desarrolla y otro revisa u optimiza el código.

WebStorm:

<https://www.jetbrains.com/webstorm/>

Es un software diseñado para el desarrollo de páginas webs en formatos html, php, etc, incluyendo el css y muchos más. Al ser un software diseñado para el diseño de webs, integra asistentes para las correcciones y sugerencias códigos, permitiéndonos reducir el tiempo de desarrollo y errores frecuentes de escritura de códigos.

GitHub:

<https://github.com/>

Es una plataforma online y gratuito que nos permite gestionar el control de versiones de las aplicaciones durante el desarrollo, así como crear ramas para trabajar con casi un sinfín de usuarios en un mismo proyecto.

NotePad++:

<https://notepad-plus-plus.org/>

Es una aplicación de código abierto de edición de códigos muy básico, con posibilidad de integrar pluggins que nos permiten facilitar la edición, comparación, etc. de códigos. En nuestro caso, lo hemos empleado para pequeñas correcciones, comparaciones y pruebas.

Google Docs:

<https://google.es/>

Ya conocemos esta funcionalidad de la gran empresa del internet. Su uso es completamente gratuito con previo registro, nos permiten crear documentos Word, Excel, presentaciones, etc. De manera fácil y cómo no, de manera cooperativa.

Firefox y Chrome browser:

<https://www.mozilla.org/es-ES/firefox/new/>

<https://www.google.com/chrome/>

Navegadores libres, sin costes y muy utilizados en todo el mundo. Hemos empleado estos 2 navegadores por ser los más utilizados, tanto en ordenadores, Tablet y SmartPhones, y así asegurarnos que nuestra aplicación sea compatible con navegadores diseñados con framework web de los más usados.

Iconos:

<https://material.io/>

Es una de las páginas con la más amplia en iconos para el diseño de webs, dándonos así un toque más atractiva a nuestras webs.

JQuery:

<https://jquery.com/download/>

Donde podemos descargarnos las librerías necesarias para el manejo de códigos en JQuery. En nuestro caso, hemos empleado la versión 3.3.1.

Google Fonts:

<https://fonts.google.com/>

Aquí podemos descargar fuentes de manera gratuita por si no nos gustase las fuentes existentes dentro de las herramientas de desarrollo que empleamos.

FIN

Imagen que contiene objeto

Descripción generada con confianza alta