

ANTEPROYECTO DEL TRABAJO DE FIN DE GRADO

INFORMACIÓN GENERAL

Alumno/a	Álvaro Yuste Moreno				
Titulación:	Grado en Ingeniería de Software				
Tutor/es:	Eduardo Guzmán De los Riscos				
Título	UrbanNav Madrid: Una herramienta visual sobre la movilidad en Madrid				
Subtítulo <i>(solo si en grupo)</i>					
Título en inglés	UrbanNav Madrid: A visual tool on mobility in Madrid				
Subtítulo en inglés <i>(solo si en grupo)</i>					
Trabajo en grupo:	Sí	<input type="checkbox"/>	No	<input checked="" type="checkbox"/>	
Otros integrantes del grupo:					

INTRODUCCIÓN

Contextualización del problema a resolver. Describir claramente de dónde surge la necesidad de este TFG y el dominio de aplicación. En caso de que el TFG se base en trabajos previos, debe aclararse cuáles son las aportaciones del TFG.

Diversos repositorios de datos abiertos ofrecen información relativa a la movilidad de las ciudades. Esta información, representada de forma visual, podría ayudar a los ciudadanos y a las autoridades a entender las dinámicas de movimiento, identificar puntos calientes, accidentes, etc. Este TFG intenta proporcionar una herramienta visual que permita analizar la información sobre la movilidad de una ciudad tan masificada como Madrid. A través de conjuntos de datos disponibles en <https://datos.madrid.es/portal/site/egob> [1] se busca desarrollar una aplicación web que muestre información detallada y análisis estadísticos sobre accidentes de tráfico, zonas de estacionamiento, ubicación de radares y otros datos.

OBJETIVOS

Descripción detallada de en qué consistirá el TFG. En caso de que el objeto principal del TFG sea el desarrollo de software, además de los objetivos generales deben describirse sus funcionalidades a alto nivel.

El objetivo es mostrar información al usuario sobre la movilidad, ya sea a través de un mapa urbano o de gráficos con información estadística sobre los datos. La aplicación permitirá, de forma fluida, navegar entre diversas funcionalidades. A continuación, se resumen cuáles serán las funcionalidades que la herramienta ofrecerá organizada en áreas:

Accidentes (de vehículos y bicicletas):

- Mostrar un mapa con zonas marcadas dependiendo de la cantidad y gravedad de los accidentes ocurridos en dicha zona.
- Mostrar una gráfica con estadísticas de accidentes agrupados por diferentes características: edad de los involucrados, tipo de vehículo, distrito, gravedad de la lesión, tipo de accidente, positivo en drogas y positivo en alcohol.
- Buscar accidentes concretos filtrando por diferentes parámetros: fecha, hora, rango de edad, sexo, distrito, estado meteorológico, tipo de vehículo, tipo de accidente,...

Estacionamiento (de vehículos varios, personas de movilidad reducida, motos y zonas de carga descarga):

- Mostrar todas las zonas de estacionamiento diferenciadas por colores, dependiendo del tipo de aparcamiento que sea.
- Buscar zonas de estacionamiento por: distrito, barrio, tipo de aparcamiento y tipo de color.

Radar:

- Mostrar todos los radares fijos.

Tráfico:

- Mostrar un mapa con zonas marcadas según la densidad media de tráfico. Podemos filtrar para ver la densidad en función de: un mes específico, un día específico y un periodo de tiempo (introduciendo un rango de días, horas, así como combinaciones de ambos).

Para cada funcionalidad se mostrará información textual que explique ciertas características que puedan generar dudas. Al seleccionar sobre un marcador del mapa, podremos observar información adicional. Por ejemplo, para el estacionamiento veremos el número de plazas, el tipo de aparcamiento (en línea o batería), etc.

ENTREGABLES

Listado de resultados que generará el TFG (aplicaciones, estudios, manuales, etc.)

Código fuente de la aplicación web UrbanNav

Memoria del TFG

MÉTODOS Y FASES DE TRABAJO

METODOLOGÍA:

Descripción de la metodología empleada en el desarrollo del TFG. Especificar cómo se va a desarrollar. Concretar si se trata de alguna metodología existente y, en caso contrario, describir y justificar adecuadamente los métodos que se aplicarán.

Se usará una metodología incremental iterativa.

Esta metodología es un enfoque que implica la división del desarrollo del proyecto en fases cíclicas donde en cada iteración se entregan funcionalidades incrementales del producto. Esto permite una gran flexibilidad a la hora de realizar cambios en los requisitos. También se obtiene una retroalimentación de las partes interesadas al final de cada iteración, lo que permite realizar ajustes durante todo el proceso.

FASES DE TRABAJO:

Enumeración y breve descripción de las fases de trabajo en las que consistirá el TFG.

- Análisis de los datos disponibles en el open data.** El origen de la información usada en este proyecto es un open data bastante extenso sobre el que se hará un estudio para ver que datos se van a utilizar y como estructurarlos.
- Diseño del modelo de datos.** Se diseñará un modelo relacional para después crear la base de datos a partir del modelo e introducir los datos.
- Diseño arquitectura del servidor.** Decisiones sobre que herramientas y tecnologías utilizar, cómo organizar el código, cómo gestionar la escalabilidad y realizar la conexión con la base de datos.
- Implementación de la lógica de negocio.** Captura y estructuración de datos, manejo de solicitudes del cliente, validación de datos y cualquier otra lógica específica de la aplicación.
- Desarrollo de la API REST.** Realización de la API que permitirá al cliente hacer solicitudes HTTP para interactuar con la lógica de la aplicación en el servidor de acuerdo al estilo arquitectónico REST.
- Diseño de la interfaz de usuario (wireframes).** Diseño de la disposición de los elementos de la interfaz, sin entrar en detalles como colores o estilos.
- Desarrollo de la estructura de las páginas web.** Se define la estructura y contenido de las páginas web con HTML, mientras que con CSS se estiliza y da formato a estos elementos.
- Implementación de la lógica del frontend.** Agregación de funcionalidad interactiva a la interfaz de usuario mediante JavaScript. Esto puede incluir la validación de formularios, efectos visuales y cualquier otra interacción dinámica que requiera la aplicación.

9. **Integración con el backend a través de la API.** Esta fase recoge el uso de la API REST en el frontend. Se realizan consultas HTTP desde el frontend para interactuar con la lógica de negocio y los datos del servidor.
10. **Despliegue.** Configuración de servidores y despliegue de la aplicación en la nube. Se desplegará en Vercel[2] (tanto back como front).
11. **Documentación técnica.** Documentación del código (comentarios, estructura, ...) y documentación del entorno de desarrollo y despliegue.
12. **Elaboración de la memoria.** Descripción del modelo de datos, casos de uso implementados, cambios realizados respecto al anteproyecto (si procede),...

TEMPORIZACIÓN:

La siguiente tabla deberá contener una fila por cada una de las fases enumeradas en la sección anterior. En caso de tratarse de un trabajo en grupo, se añadirá una columna HORAS por cada miembro del equipo. Debe especificarse claramente el número de horas dedicado por cada alumno/a y la suma de horas individual deberá ser también de 296.

FASE	HORAS
	Álvaro Yuste Moreno
Análisis de los datos disponibles en el open data	30
Diseño del modelo de datos	30
Diseño arquitectura del servidor	10
Implementación de la lógica de negocio	70
Desarrollo de la API REST	10
Diseño de la interfaz de usuario (wireframes)	5
Desarrollo de la estructura de las páginas web	30
Implementación de la lógica del frontend	30
Integración con el backend a través de la API	5
Despliegue	10
Documentación técnica	16
Elaboración de la memoria	50
	296

ENTORNO TECNOLÓGICO

TECNOLOGÍAS EMPLEADAS:

Enumeración de las tecnologías utilizadas (lenguajes de programación, frameworks, sistemas gestores de bases de datos, etc.) en el desarrollo del TFG.

Lenguajes de programación: JavaScript (se utiliza tanto en el backend como en el frontend), HTML (se utiliza para la estructura y el marcado del contenido de las páginas web) y CSS (se utiliza para el diseño y la presentación visual de las páginas web)

Frameworks y bibliotecas: Express.js[3] (framework de JavaScript usado para el lado del servidor), React.js[4] (biblioteca de JavaScript para construir interfaces de usuario) y Leaflet [5] (biblioteca de JavaScript para mapas interactivos) entre otras.

Gestor de bases de datos: MySQL[6] (gestor de bases de datos relacional)

RECURSOS SOFTWARE Y HARDWARE:

Listado de dispositivos (placas de desarrollo, microcontroladores, procesadores, sensores, robots, etc.) o software (IDE, editores, etc.) empleados en el desarrollo del TFG.

Visual Studio Code[7]: entorno para desarrollar la aplicación web

MySQL Workbench 8.0 CE[8]: IDE para MySQL



REFERENCIAS

Listado de referencias (libros, páginas web, etc.)

- | |
|---|
| [1] Open Data:
https://datos.madrid.es/sites/v/index.jsp?vgnextoid=374512b9ace9f310VgnVCM100000171f5a0aRCRD&buscar=true&Texto=&Sector=transporte&Formato=&Periodicidad=&departamento=&orderByCombo=CONTENT_INSTANCE_NAME_DECODE |
| [2] Vercel: https://vercel.com/docs |
| [3] Express: https://expressjs.com/ |
| [4] React: https://es.react.dev/ |
| [5] Leaflet: https://leafletjs.com/ |
| [6] MySQL: https://www.mysql.com/ |
| [7] Visual Studio Code: https://code.visualstudio.com/ |
| [8] MySQL Workbench: https://www.mysql.com/products/workbench/ |

Málaga, 29 de febrero de 2024

Firma tutor/tutora:

Firma cotutor/a:

Firma tutor/a coordinador/a: