

# Proyecto Viade\_es4a

Pelayo Rodríguez García

Raúl García Arenas

Antonio Fernández Flambó

Aique Montes Maestre

Adrián Díaz Moro

Israel Méndez Rodríguez

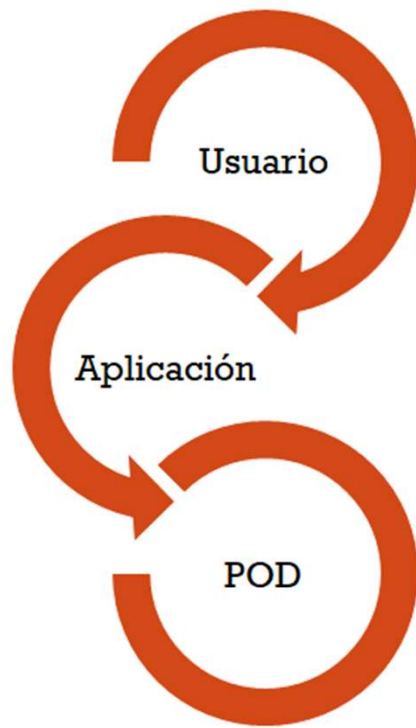
# Contenido

- ▶ Introducción
  - ▶ Solid
- ▶ Herramientas utilizadas
- ▶ Organización
  - ▶ Actas
  - ▶ Issues
- ▶ Explicación parte principal App
- ▶ Evolución del proyecto
- ▶ Demo viade\_es4a

# INTRODUCCIÓN: SOLID

- ▶ Proyecto que quiere cambiar la web
  - ▶ RE-descentralización de la web
- ▶ Ideado por Tim Berners-Lee
- ▶ Objetivos
  - ▶ Realizar aplicaciones descentralizadas basadas en LinkedData
  - ▶ Dar el control de sus datos al usuario
  - ▶ Interoperabilidad de aplicaciones
  - ▶ Aplicaciones modulares y extensibles

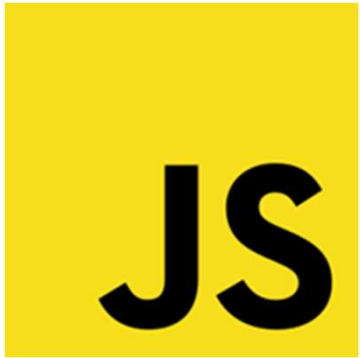




# FUNCIONAMIENTO

- ▶ La información se guarda en PODs
- ▶ Cada usuario tiene un POD
  - ▶ Tiene control absoluto sobre su información
- ▶ La aplicación interacciona con dichos POD

## HERRAMIENTAS UTILIZADAS: PÁGINA WEB

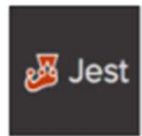




HERRAMIENTAS UTILIZADAS: ENTORNO



Chai



Jest

cucumber

HERRAMIENTAS UTILIZADAS: PRUEBAS

## HERRAMIENTAS UTILIZADAS: GIT

- ▶ GitHub: Control de versiones
- ▶ Travis CI: Integración continua



Travis CI



GitHub





# Organización

- ▶ Actas de reunión
- ▶ Issues
- ▶ Separación en ramas
- ▶ Pull Requests

# Organización (Actas de reunión)

## Acta 07 02 2020

Pelayo edited this page on 7 Feb · 1 revision

## Acta 07/02/2020

Hoy nos repartiremos el resto de documentación. Aunque no lleguemos a completar en profundidad todos los apartados, los dejaremos repartidos para ir haciendolos. Adri y Pelayo acabarán el 04,09 Raúl el 11, Israel la 10, Aique y Antonio la 08, El resto nos las repartiremos a lo largo de la semana según vayamos concretando cómo desarrollar el proyecto.

Asistencia: Antonio, Adri, Israel, Aique, Raúl y Pelayo.

Acta 07 02 2020

Acta 13 03 2020













Acta 14 02 2020

Acta 20 03 2020

Acta 21 02 2020

Acta 24 01 2020

Acta 31 01 2020

<input type="checkbox"/>	 <b>Revisión prototipo</b> #29 opened on 6 Mar by cidinene		
<input type="checkbox"/>	 <b>Listado de amigos</b> <span>help wanted</span> #28 opened on 6 Mar by RaulGarciaArenas		
<input type="checkbox"/>	 <b>Upload Routes from geojson</b> <span>enhancement</span> #22 opened on 4 Mar by israelmnrq		 2
<input type="checkbox"/>	 <b>Rama pruebaEsqueleto</b> <span>enhancement</span> <span>help wanted</span> #19 opened on 19 Feb by Pelayo-Rodriguez-Garcia		 3
<input type="checkbox"/>	 <b>Revisión Arquitectura</b> #18 opened on 16 Feb by cidinene		 1
<input type="checkbox"/>	 <b>package.json</b> <span>enhancement</span> <span>good first issue</span> <span>question</span> #12 opened on 12 Feb by Pelayo-Rodriguez-Garcia		
<input type="checkbox"/>	 <b>12_glossary</b> <span>documentation</span> <span>help wanted</span> #7 opened on 11 Feb by RaulGarciaArenas		 1

# Organización (Issues)

# Organización (Separación en ramas)

- ▶ Rama/funcionalidad
- ▶ Funcionalidad/persona
- ▶ ¿Excesiva granularidad?

Active branches		
travis_fix	Updated yesterday by israelmnrq	✓
develop	Updated 3 days ago by Pelayo-Rodriguez-Garcia	✗
testRaul2	Updated 3 days ago by RaulGarciaArenas	✗
documentacion-V+	Updated 3 days ago by RaulGarciaArenas	✗
revert-37-cambiosCss	Updated 24 days ago by UO264695	✗

# Organización (Pull Requests)

- ▶ Todo el código ha de pasar por un pull request
- ▶ Probar antes de usar

1 Open

✓ 28 Closed

Documentacion v+

✖

#41 by RaulGarciaArenas was merged 3 days ago

Develop

✖

#39 by UO261072 was merged 23 days ago

Revert "Cambios css"

✖

#38 by UO264695 was merged 24 days ago

Cambios css

✖

#37 by UO264695 was merged 24 days ago

Revert "Cambios css"

✖

#36 by UO264695 was merged 24 days ago

Cambios css

✖

#35 by UO264695 was merged 24 days ago

Listado rutas

✖

#34 by RaulGarciaArenas was merged 24 days ago

# Desarrollo

- ▶ El desarrollo ha seguido varias etapas:
  - ▶ Consulta de tutoriales/documentación/información.
  - ▶ Construcción de la documentación.
  - ▶ Puesta en marcha.
  - ▶ Desarrollo feature after feature.
  - ▶ ¿Refinamiento?



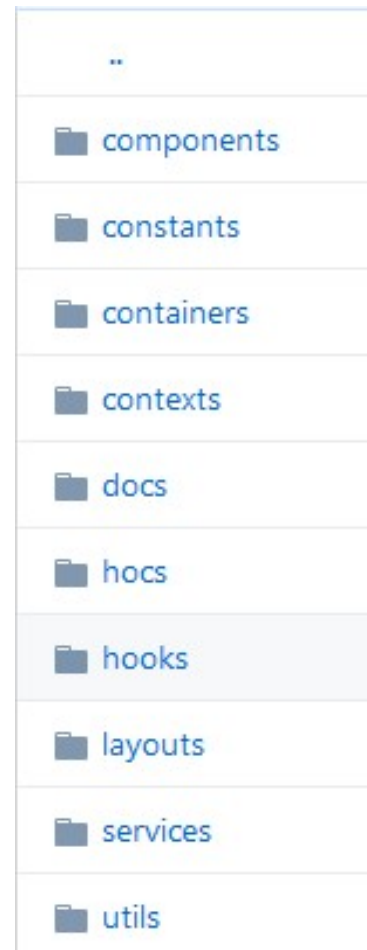
# Desarrollo (Esqueleto)

- ▶ Tras la consulta de documentación, se acuerda el uso de un esqueleto React/SOLID.
- ▶ Ventajas: Código ya desarrollado, estructura modular ya construida.
- ▶ Desventajas: Muy resistente al cambio, funcionalidad no usada.

WRITING SOLID APPLICATIONS WITH REACT

---

# Desarrollo (Esqueleto)





# Desarrollo (Funcionalidades principales)

- ▶ Se necesitará:
  - ▶ Poder loguearse en la aplicación con tu POD.
  - ▶ Un mapa en el que poder trazar una ruta.
  - ▶ Manejar esa ruta:
    - ▶ Subirla
    - ▶ Descargarla
    - ▶ Consultarla
  - ▶ Acceder a la lista de amigos

# Desarrollo del mapa

- ▶ Uso de la biblioteca “react-google-maps”.
- ▶ ¿Fallo de estudio previo?
- ▶ Documentación no tan extensa.

```
import React from "react";  
import { Marker, GoogleMap, withScriptjs, withGoogleMap, Polyline } from 'react-google-maps';
```

```
class MapWithMarkers extends React.Component {  
  state = {  
    places: [],  
    pathCoordinates: []  
  }  
  checkLines(e, newPlace) {  
    console.log(e);  
    const coordinates = {  
      lat: newPlace.lat,  
      lng: newPlace.lng  
    };  
    this.setState({  
      pathCoordinates: [...this.state.pathCoordinates, coordinates]  
    });  
  }  
}
```

# Desarrollo del mapa

Uso del concepto de State de React.

# Desarrollo del mapa

- Función render() de React.
- ¿Cómo integramos todo esto en nuestra arquitectura?

```
render() {  
  return (  
    <GoogleMap  
      onClick={this.addMarker.bind(this)}  
      defaultZoom={this.props.zoom}  
      defaultCenter={this.props.center}  
    >  
      {this.state.places.map(place => {  
        return (  
          <Marker  
            key={place.id}  
            position={{ lat: place.lat, lng: place.lng }}  
          />  
        );  
      })}  
    <Polyline  
      path={this.state.pathCoordinates}  
      geodesic={true}  
      options={{  
        strokeColor: "#ff2527",  
        strokeOpacity: 0.75,  
        strokeWeight: 2,  
        icons: [  
          {  
            icon: "lineSymbol",  
            offset: "0",  
            repeat: "20px"  
          }  
        ]  
      }}  
    ></Polyline> );  
  }  
}
```

# Desarrollo del mapa

```
▼ Mapa
  JS index.js
  JS mapa.component.js
  JS mapaFinal.component.js
```

```
import MyMapComponent from './mapaFinal.component.js';
export default MyMapComponent;
```

- ▶ Se exporta el componente de manera que sea accesible desde el enrutador
- ▶ El enrutador se exporta de manera que sea accesible desde App.js, punto de entrada de la aplicación.
- ▶ Hay una separación en capas estilo MVC, integrando React.
- ▶ ¿Es React el framework adecuado?

# Desarrollo del mapa (Enrutador)

```
const Routes = () => (  
  <Router>  
    <Fragment>  
      <Switch>  
        <NotLoggedInLayout component={Login} path="/login" exact />  
        <NotLoggedInLayout component={Register} path="/register" exact />  
        <NotLoggedInLayout path="/register/success" component={RegistrationSuccess} exact />  
        <PublicLayout path="/404" component={PageNotFound} exact />  
        <Redirect from="/" to="/welcome" exact />  
        <PrivateLayout path="/" routes={privateRoutes} />  
        <Redirect to="/404" />  
      </Switch>  
    </Fragment>  
  </Router>  
)  
);  
  
export default Routes;
```

# Desarrollo del mapa (App.js)

- ▶ Muchos componentes heredados del esqueleto.
- ▶ Mucho estudio previo.
- ▶ Escenario Greenfield vs Legacy.

```
const App = () => (  
  <Suspense fallback={<Loader />}>  
    <ThemeProvider theme={theme}>  
      <Fragment>  
        <Routes />  
        <Toaster  
          {...{  
            autoClose: 3000,  
            position: toast.POSITION.TOP_CENTER,  
            newestOnTop: true,  
            closeOnClick: true,  
            pauseOnVisibilityChange: true,  
            draggable: true,  
            className: 'solid-toaster-container',  
            toastClassName: 'solid-toaster',  
            bodyClassName: 'solid-toaster-body',  
            transition: Slide  
          }}  
        />  
      </Fragment>  
    </ThemeProvider>  
  </Suspense>  
)  
);
```

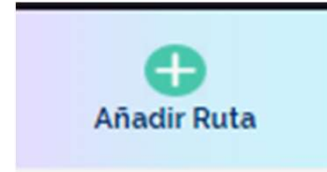
# Añadir Rutas

Breve explicación del añadido de rutas y parseado del JSON





# Interfaz de usuario



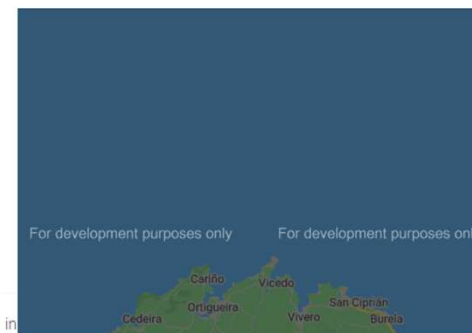
## Subir ruta

Upload file:  
 Ningún archi...seleccionado

## Add Route

Introduce el nombre:

Introduce la descripción de la ruta:



# El Codigo

- ▶ Se utiliza un formulario para introducir los datos
- ▶ Se solucionan los problemas de nombre, dandole el nombre de la fecha de subida.

```
const onSubmit= async (e)=>{  
  console.log(name);  
  var file=Parser(name,description,geoCoordinates);  
  console.log(new Date());  
  const {webId} = props;  
  console.log(webId);  
  
  const url=webId.split("profile/card#me")[0]+"public/"+new Date()+".json";  
  const fc  = new FC( auth );  
  await fc.createFile(url, file, "application/geo+json", {});  
  console.log("subido");  
}
```

# Creación del JSON

## Proposal for a route format standard #2



UO258425 opened this issue on 21 Feb · 10 comments

```
json='{ "@context":"http://schema.org",\n' +  
  '"@type": "Trip",\n' +  
  '"name": '+name+',\n' +  
  '"description": '+description+',\n' +  
  '"itinerary": {\n' +  
    '"@type": "ItemList",\n' +  
    '"numberOfItems": '+geoCoordinates.length+',\n' +  
    '"itemListOrder": "http://schema.org/ItemListOrderDescending",\n' +  
    '"itemListElement": [\n'  
if(geoCoordinates!=null){  
  for(let i=0;i<geoCoordinates.length;i++){  
    json=json+ '      {\n' +  
      '"@type": "GeoCoordinates",\n' +  
      '"latitude": '+geoCoordinates[i][0]+',\n' +  
      '"longitude": '+geoCoordinates[i][1]+'  
    }';  
    if(i<geoCoordinates.length-1){  
      json+=',\n';  
    }  
  }  
}  
]  
json+= '    ]\n' +  
  '  }\n' +  
  '}';
```

# Problemas encontrados al Añadir Rutas

- ▶ El Nombre de las rutas
- ▶ Obtener la dirección del pod del usuario
- ▶ Obtener las coordenadas del Mapa

# Listado de amigos

```
class FriendsComponent extends React.Component {
  state = this.savedState;

  get savedState() {
    return { expression: decodeURIComponent(window.location.hash.substr(1)) };
  }
  set savedState({ expression }) {
    window.location.replace(`#${encodeURIComponent(expression)}`);
  }

  componentDidUpdate(prevProps) {
    // mostrar webid si se registro
    const { webId } = this.props;
    if (webId && webId !== prevProps.webId && !this.savedState.expression)
      this.setState({ expression: `[${webId}].friends` });
  }

  onExpressionChange = expression => {
    this.savedState = { expression };
  }

  render() {
    return (
      <SubFriends expression={this.state.expression /*|| defaultExpression*/}
        onExpressionChange={this.onExpressionChange}/>
    );
  }
}
```

## Problemas al listar usuarios



Dificultades al principio con  
el webID



Acceder al POD del usuario

# Listado de rutas

```
const {webId} = props;
console.log(webId);
const url=webId.split("profile/card#me")[0]+"public";
console.log(url);
// I will use: https://github.com/jeff-zucker/solid-file-client
const fc = new FC( auth ) //With fc we can manage files
async function run(){

  var publicFolder = await fc.readFolder(url) //Get an item which has the url with the files(routes)

  for(var i=0; i < publicFolder.files.length; i++){ //Iterate over the files
    // var file = await fc.readHead(publicFolder.files[i].url) //Get the file's content
    if(publicFolder.files[i].name.includes(".json")){
      //console.log(publicFolder.files[i].name)
      var file=await fc.readFile(publicFolder.files[i].url);

      try{
        file=JSON.parse(file);
        //console.log(file)
        console.log(publicFolder.files[i].name)
        if(file.name!=null)
          console.log(file.name)
        if(file.description!=null)
          console.log(file.description)
      }catch (e) {

    }
  }
}
```

## Problemas al listar rutas



Acceder a la carpeta del  
POD

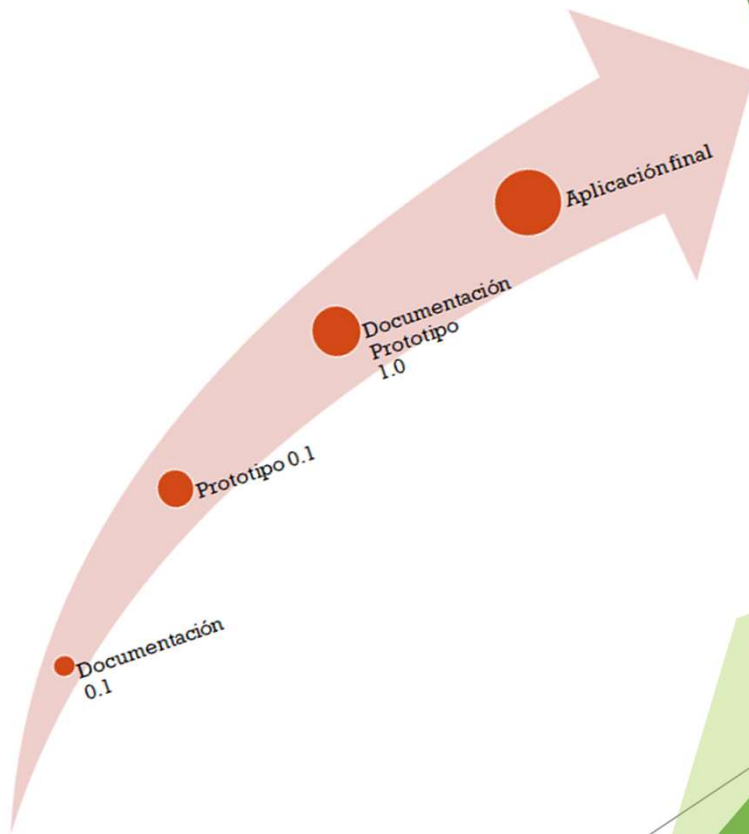


Usar cuenta del usuario en  
sesión



# EVOLUCIÓN DEL PROYECTO

- ▶ Cuatro entregables a lo largo del desarrollo
- ▶ Se puede equiparar a la realización de cuatro *sprints*



Documentación

Uso de arc42

Documentación  
muy básica

Lenguaje  
AsciiDoc

Primera entrega

Introduction and goals

Constraints

Context and scope

Solution strategy

Building block view

Runtime view

Crosscutting concepts

Architectural decisions

Quality requirements

Risks and technical debt

Glossary

# Segunda entrega

App funcional con las siguientes funcionalidades:

- Listado de amigos.

- Mostrar mapa.

- Subir archivo rutas.

- Test

# Tercera entrega



Mejoras en la subida de rutas.



Nuevos test.



Despliegue en gh-pages.



Versión final documentación.

# Entrega final



Mejoras en el mapa.



Parseado de rutas mejorado.

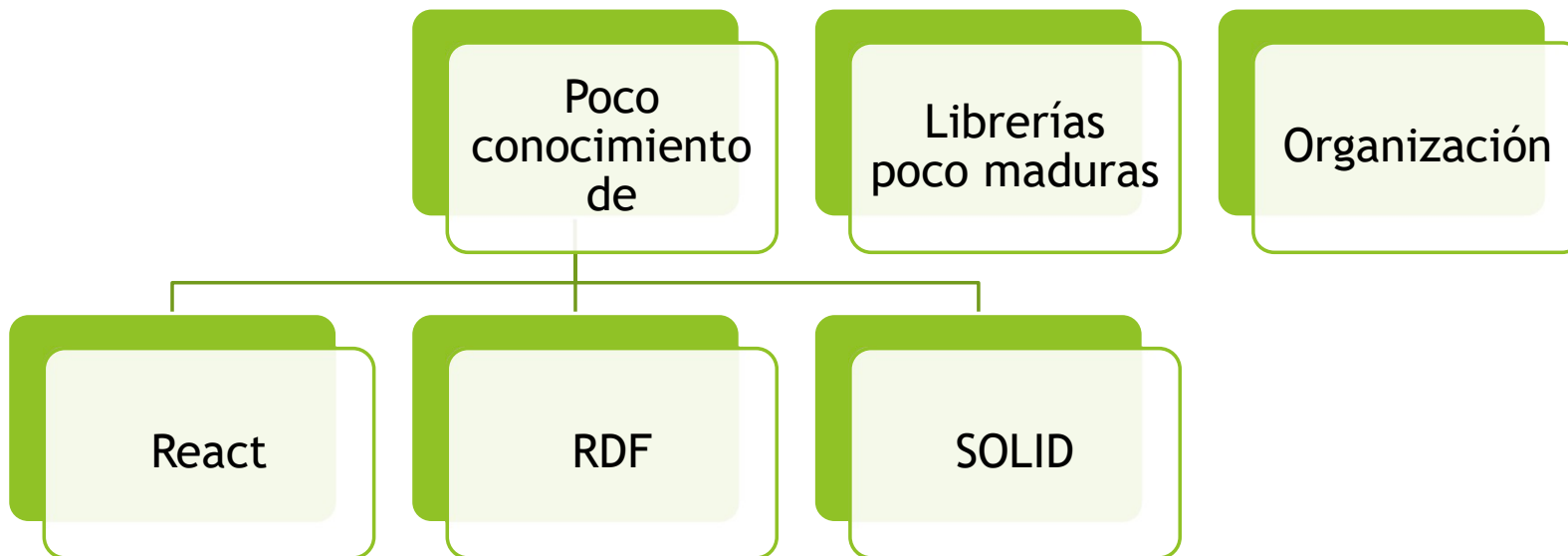


Nuevos test de aplicación.



Mejoras Codecov y code quality.

# Problemas encontrados



## Posibles mejoras futuras



Compartir  
rutas.



Compartir  
imágenes.



Mejora del  
rendimiento.



Interfaz más  
sencilla.



# Funcionamiento

A continuación, mostraremos como se suben rutas.







Bienvenida

Perfil

Añadir  
Ruta

Subir  
Ruta

Ver  
Ruta

Amigos

Listar  
rutas

Mapa



inrupt

Bienvenido, **UO261072**



## Subir ruta

Upload file:

Seleccionar archivo

Ningún archi...seleccionado

## Add Route

Introduce el nombre:

Intoduce la descripcion de la ruta:

Añadir

## Subir ruta

Upload file:

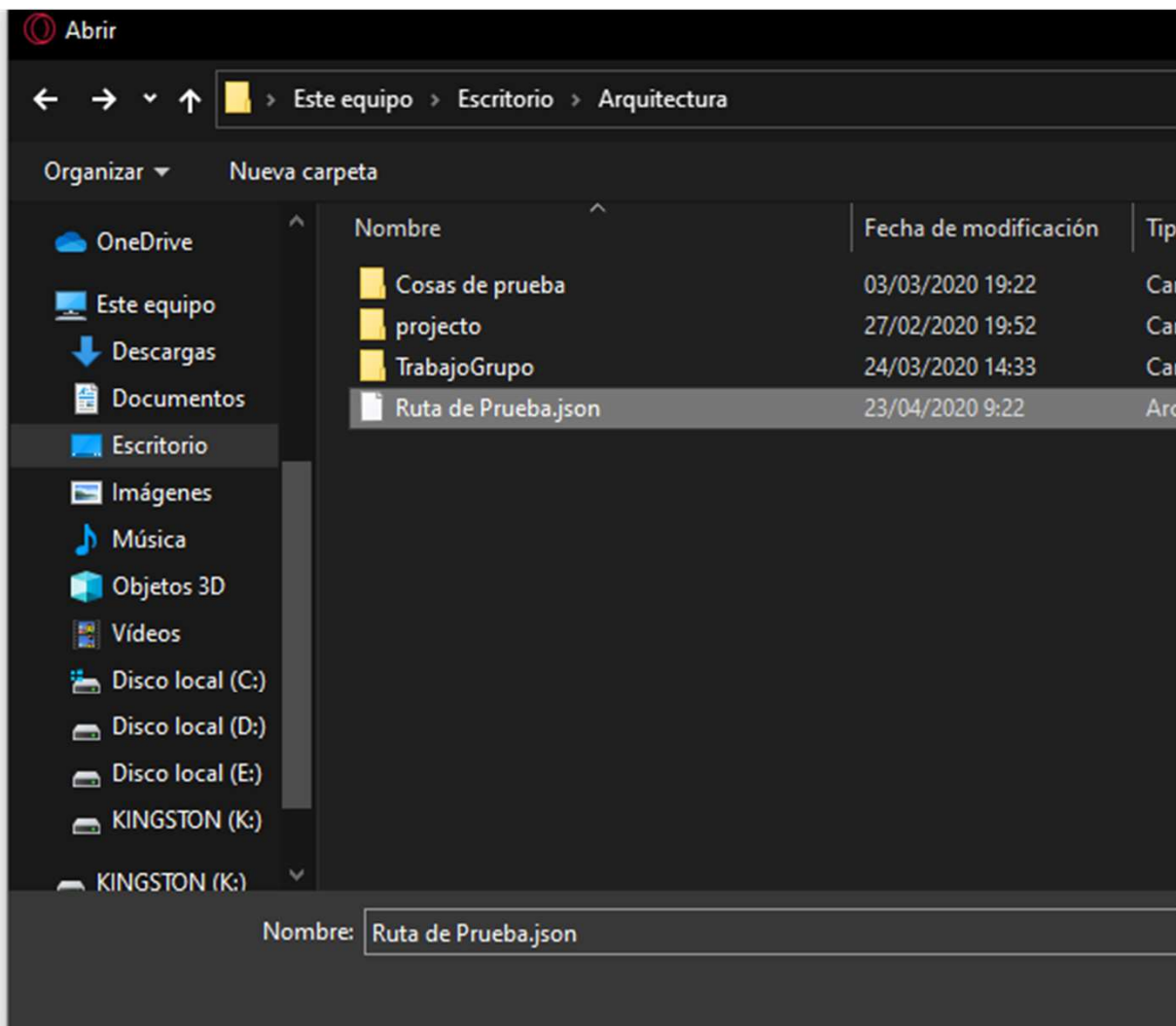
Seleccionar archivo Ningún archi...seleccionado

## Add Route

Introduce el nombre:

Intoduce la descripcion de la ruta:

Añadir



# Add Route

Introduce el nombre:

Introduce la descripción de la ruta:



Nombre: Ruta C

Descripcion: Mi ruta favorita

Coordenadas: -5.8512792 , 43.3551061

-5.8712749 , 42.015294

Nombre: Ruta H

Descripcion: Una ruta tranquila sin muchos problemas

Coordenadas: -5.182042 , 30.3123321

-4.2134512 , 29.47261