

React Nivel Avanzado

Sesión sincrónica 6

Consumiendo servicios web



Bienvenida y actividad de bienestar

Duración: 10 minutos.

Nombre de la práctica	Plantea tus objetivos como metas de aproximación y replantea tus metas de evitación.
Descripción de la práctica	Con base en lo que propone Grenville (2012), en la práctica podrás definir diferentes tipos de metas y encontrar la mejor manera de conseguirlas.
Palabras clave	Objetivos, metas y planes.
Instrucciones para el aprendedor	<p>La autora Bridget Grenville-Cleave (2012), comenta que en el establecimiento de metas es importante distinguir los tipos que existen y menciona dos:</p> <ol style="list-style-type: none">1. Metas de aproximación (approach): son los resultados positivos (deseables, placenteros, benéficos o que nos gustaría tener) y hacia los cuales trabajamos.2. Metas de evitación (avoidance): son los resultados negativos (indeseables, dolorosos, dañinos, o que nos disgustan) y en los que trabajamos para evitarlos. <p>Ejemplo:</p> <p>Meta de aproximación:</p> <ul style="list-style-type: none">• Mejorar la eficiencia.• Ser amigable y extrovertido (a) en reuniones.• Asumir el rol de líder en el trabajo.

<p><i>Instrucciones para el aprendedor</i></p>	<p>Meta de evitación:</p> <ul style="list-style-type: none">● Dejar de aplazar.● Vencer la timidez en las reuniones.● No pasar desapercibido en el trabajo. <p>Las investigaciones que se han realizado respecto a este tema muestran que perseguir metas de evitación resulta en un detrimento del bienestar. Estos descubrimientos sugieren que establecer metas de aproximación o replantear las de evitación es benéfico. Reflexiona lo siguiente:</p> <ul style="list-style-type: none">● ¿Qué tipo de metas te has trazado?● ¿Hay algunas que puedas replantear en una forma más positiva?● ¿Cuándo las tendrás listas?
<p>Fuente</p>	<p>Grenville, B. (2012). <i>GOAL-SETTING SECRETS</i>. Recuperado de http://positivepsychologynews.com/news/bridget-grenville-cleave/2012013120696</p>

Actividad guiada

Parte 1

Duración: 75 minutos.

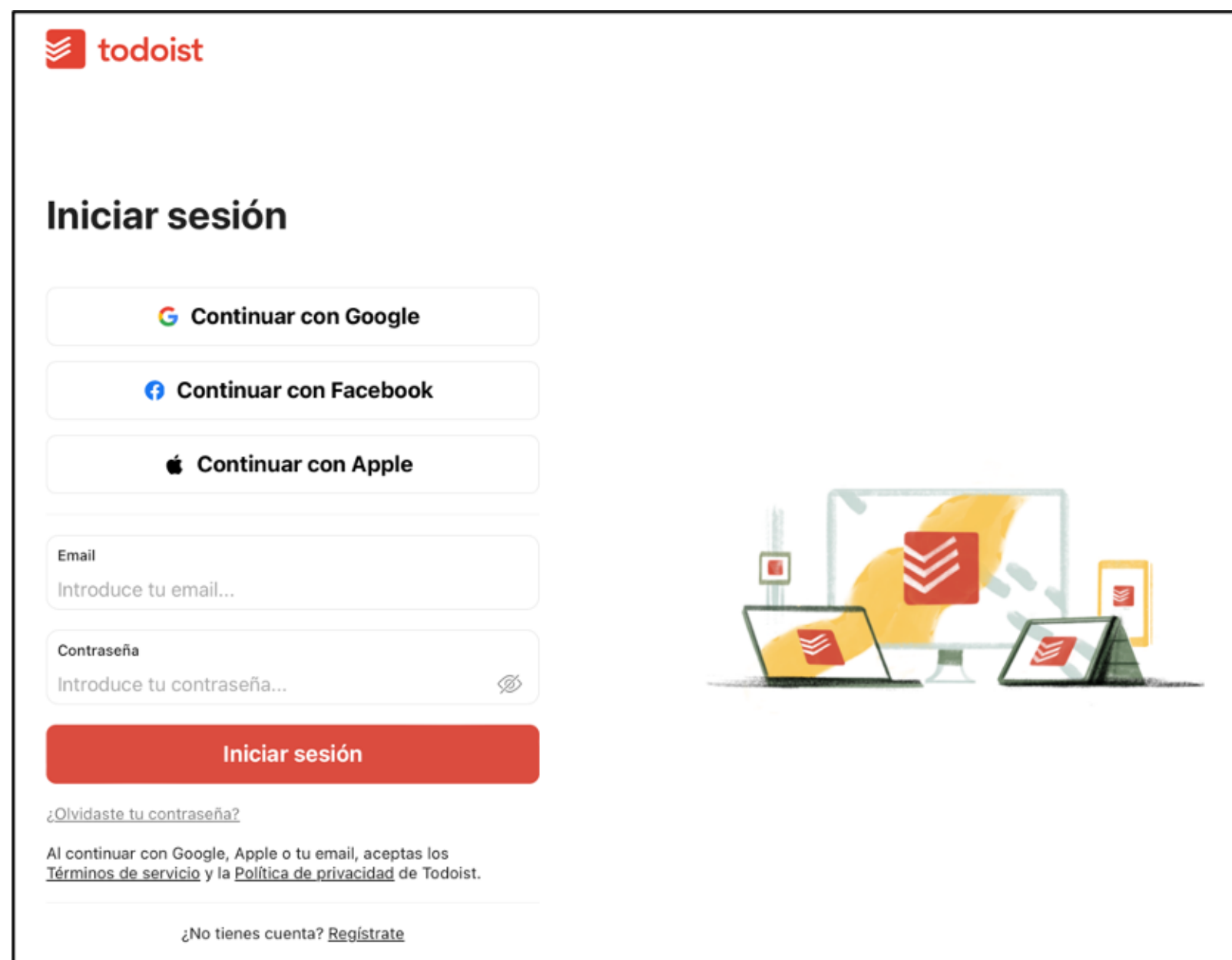
Ejercicio 1:

A continuación crearás una app de notas usando el servicio de Todoist, la app obtendrá y creará una lista de tareas mediante el uso de la API web de Todoist.

1. Crear el proyecto con el comando

```
npx react-native init todolist
```

2. Abrir la carpeta del proyecto en VS Code
3. Crear una cuenta en <http://todoist.com>

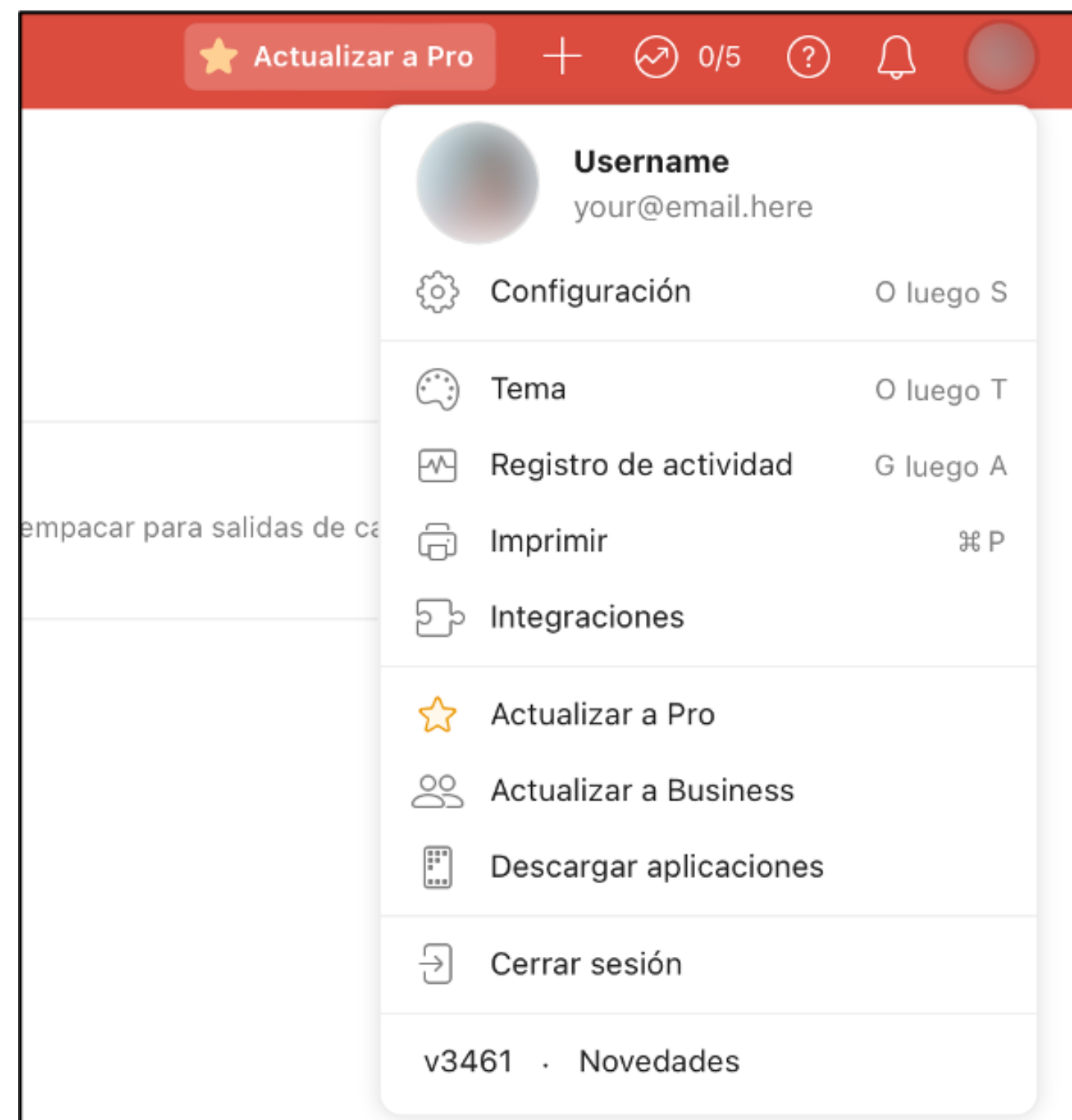


The screenshot shows the Todoist login interface. At the top left is the Todoist logo. Below it, the heading "Iniciar sesión" is displayed. There are three social login buttons: "Continuar con Google", "Continuar con Facebook", and "Continuar con Apple". Below these are two input fields: "Email" with the placeholder "Introduce tu email..." and "Contraseña" with the placeholder "Introduce tu contraseña..." and an eye icon for toggling visibility. A red "Iniciar sesión" button is positioned below the password field. To the right of the login form is an illustration of a laptop, a desktop monitor, and a smartphone, all displaying the Todoist app interface. At the bottom left, there is a link "¿Olvidaste tu contraseña?". At the bottom center, a small text block states: "Al continuar con Google, Apple o tu email, aceptas los [Términos de servicio](#) y la [Política de privacidad](#) de Todoist." At the bottom right, there is a link "¿No tienes cuenta? [Regístrate](#)".

Actividad guiada

Parte 1

4. Después de iniciar sesión, ir al icono de usuario para desplegar el siguiente menú.

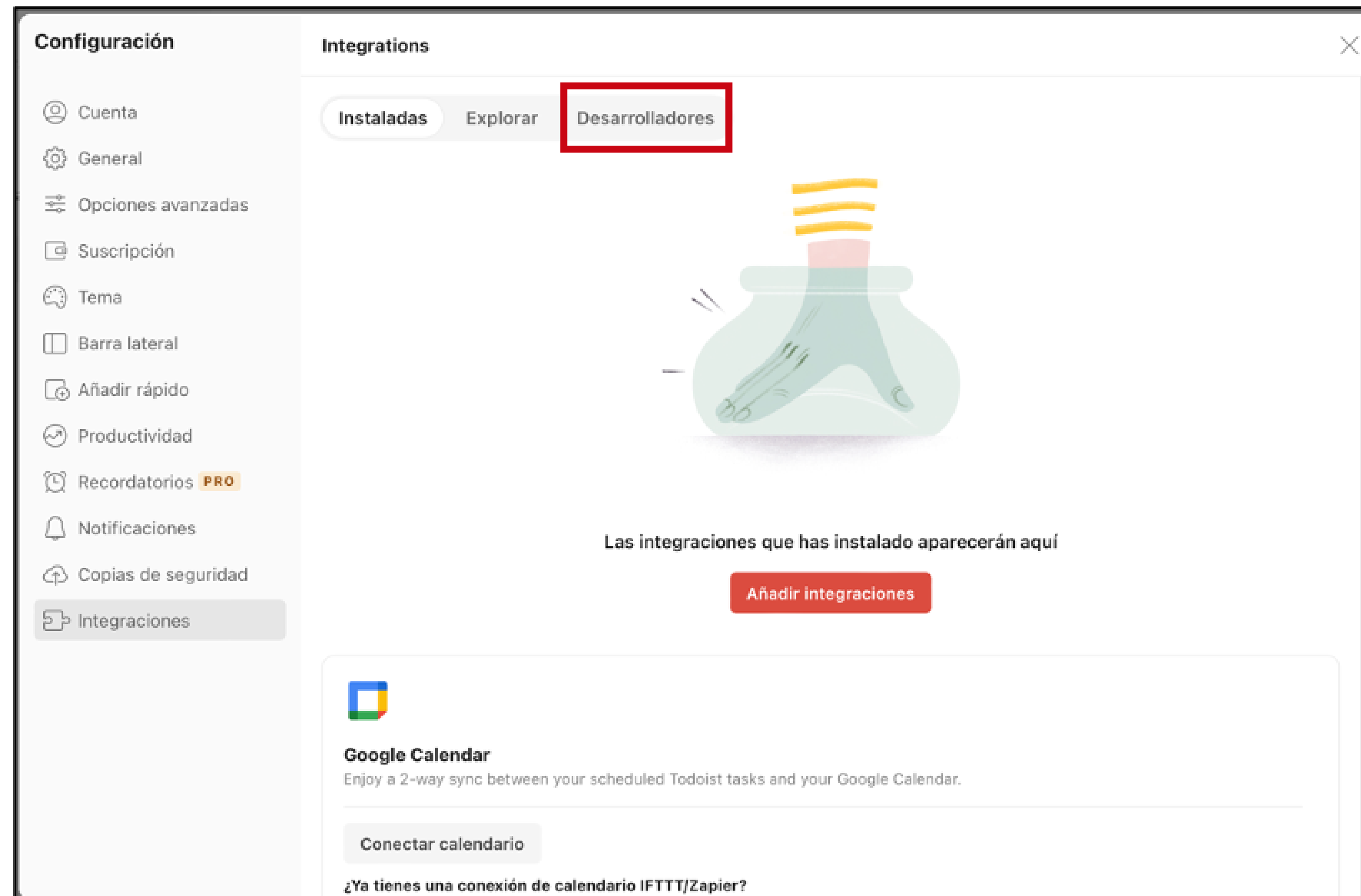


5. Elegir la opción **Integraciones**..

Actividad guiada

Parte 1

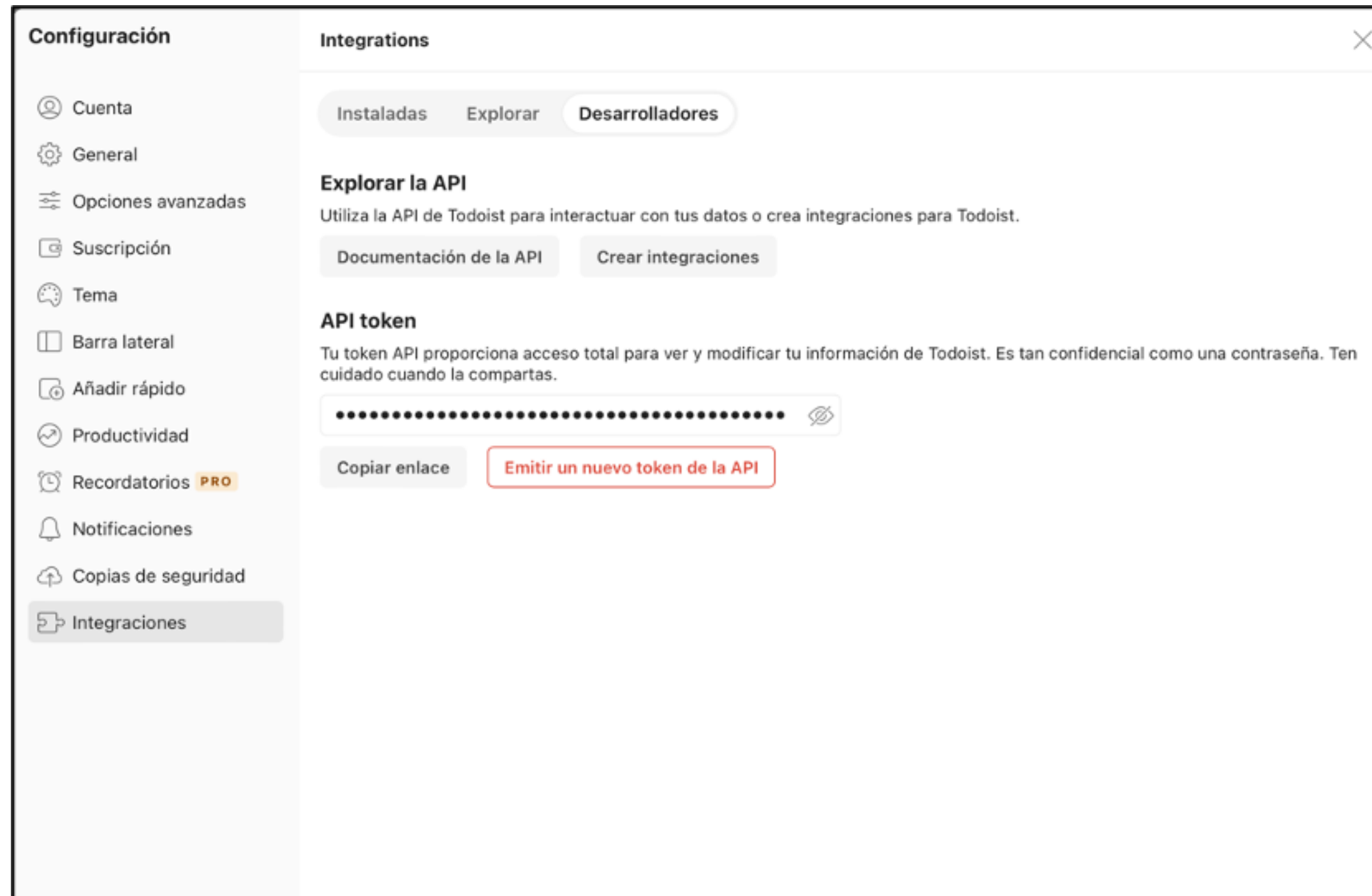
6. En la ventana que aparece, elegir **Desarrolladores** del menú de la parte de arriba.



Actividad guiada

Parte 1

7. Hacer clic en Añadir integraciones y copiar el API Token que se muestra.



8. Instalar en el proyecto el SDK de Todoist con el siguiente comando.

```
npm install @doist/todoist-api-typescript
```

Actividad guiada

Parte 1

9. Crear el archivo App.jsx en la misma carpeta que está el archivo App.tsx
10. Pegar el siguiente código en el archivo App.jsx (el que se acaba de crear)

```
import React, { useEffect, useState } from 'react';
import { SafeAreaView } from 'react-native';

import { TodoistApi } from "@doist/todoist-api-typescript";
import TaskList from '../components/TaskList';

// API Token for TODOIST
const TODOIST_TOKEN = "TU-TOKEN"

const api = new TodoistApi(TODOIST_TOKEN)

const App = () => {
  const [projects, setProjects] = useState([])
  const [tasks, setTasks] = useState([])

  useEffect(() => {
    // Obtiene los proyectos
    api.getProjects()
      .then((response) => setProjects(response))
      .catch(() => setProjects([]))

    // Obtiene la lista de tareas
    api.getTasks()
      .then((response) => setTasks(response))
      .catch(() => setTasks([]))
  }, [])

  return (
    <SafeAreaView>
      <TaskList tasks={tasks} projects={projects} />
    </SafeAreaView>
  )
}
export default App
```


Actividad guiada

Parte 1

11. Eliminar el archivo App.tsx
12. En el archivo App.jsx reemplazar el texto TU-TOKEN por el token que se encuentra en la sección **Api Token** de la ventana **integrations** del sitio de Todoist.
13. Crear la carpeta components en el mismo nivel que **App.jsx** y dentro crear el archivo **TaskList.jsx**
14. Pegar el siguiente código en el archivo **TaskList.jsx**

```
import React, { useEffect, useState } from 'react';
import { SectionList, StyleSheet, Text, View } from 'react-native';

const styles = StyleSheet.create({
  container: {
    paddingTop: 22,
  },
  sectionHeader: {
    paddingTop: 2,
    paddingLeft: 10,
    paddingRight: 10,
    paddingBottom: 2,
    fontSize: 22,
    fontWeight: 'bold',
    backgroundColor: 'rgba(247,247,247,1.0)',
  },
  item: {
    padding: 10,
    fontSize: 14,
    overflow: 'visible',
    height: 44,
  },
});

const TaskList = ({ tasks, projects }) => {
  const [projectList, setProjectList] = useState([])

  useEffect(() => {
    if (Array.isArray(tasks) && Array.isArray(projects)) {
      const allprojects = projects.map((project) => ({
        title: project.name,
        data: tasks.filter((f) => f.projectId == project.id).map(t => t.content)
      }));
      setProjectList(allprojects)
    }
  }, [projects, tasks])

  return (
    <View style={styles.container}>
      <SectionList sections={projectList}
        renderItem={({ item }) => <Text style={styles.item}>{item}</Text>}
        renderSectionHeader={({ section }) => (
          <Text style={styles.sectionHeader}>{section.title}</Text>
        )}
        keyExtractor={item => `basicListEntry-${item}`}
      />
    </View>
  );
};

export default TaskList;
```

Actividad guiada

Parte 1

15. Guardar los cambios y ejecutar el proyecto con **npm start**
16. Elegir el tipo de plataforma donde realizar las pruebas (Android o iOS).
17. Verificar que en la consola se muestre algo similar a esto.



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  node + v [icon] [icon] ... ^ x
LOG [{"color": "charcoal", "commentCount": 0, "id": "2312170614", "isFavorite": false, "isInboxProject": true, "isShared": false, "isTeamInbox": false, "name": "Inbox", "order": 0, "parentId": null, "url": "https://todoist.com/showProject?id=2312170614", "viewStyle": "list"}, {"color": "grey", "commentCount": 0, "id": "2312170620", "isFavorite": false, "isInboxProject": false, "isShared": false, "isTeamInbox": false, "name": "Educaci\u00f3n 🎓", "order": 1, "parentId": null, "url": "https://todoist.com/showProject?id=2312170620", "viewStyle": "list"}]
```

Consideraciones:

Debe:

1. Tener instalado React Native.
2. Haber instalado Android Studio.
3. Saber programar en JavaScript.
4. Realizar los ejercicios en orden.

Receso

React Nivel Avanzado

Duración: 10 minutos.



Actividad guiada

Parte 2

Duración: 75 minutos.

Continuar probando las demás propiedades del objeto **tasks** y del objeto **project** que la API devuelve.

Ejercicio 2:

1. Agregar estilos a los elementos usados hasta ahora.
2. Modificar el componente TaskList al gusto para hacer más claro el código.
3. Personalizar los encabezados de los proyectos.
4. Agregar más información en la lista de tareas, como fecha, descripción, etc.
5. Optimizar el código utilizado en el ejemplo anterior

Consideraciones:

Debe:

1. Tener instalado React Native.
2. Haber instalado Android Studio.
3. Saber programar en JavaScript.
4. Realizar los ejercicios en orden.

Duración: 10 minutos.

El uso de servicios web nos permite obtener datos y mostrar información de diferentes fuentes, Toda la interactividad dentro de las aplicaciones móviles se realiza mediante el *Touch* del usuario, por lo cual es sumamente importante que cuentes con un sistema de respuesta para los gestos que haga el usuario.

Recuerda que la interactividad es parte de la funcionalidad de la aplicación, sin esta, el usuario no podrá hacer nada, se frustrará y no volverá a usar tu desarrollo. Es importante que realices pruebas de interactividad para asegurarte que todo funciona correctamente.

El brindar al usuario información de lo que está haciendo en todo modo momento tu aplicación es importante, ya que esto impacta en la experiencia y por lo tanto, la opinión que tendrá tu usuario al usar tu desarrollo, eso hará que quiera volver a usarlo o no desee usarlo más.

La obra presentada es propiedad de ENSEÑANZA E INVESTIGACIÓN SUPERIOR A.C. (UNIVERSIDAD TECMILENIO), protegida por la Ley Federal de Derecho de Autor; la alteración o deformación de una obra, así como su reproducción, exhibición o ejecución pública sin el consentimiento de su autor y titular de los derechos correspondientes es constitutivo de un delito tipificado en la Ley Federal de Derechos de Autor, así como en las Leyes Internacionales de Derecho de Autor.

El uso de imágenes, fragmentos de videos, fragmentos de eventos culturales, programas y demás material que sea objeto de protección de los derechos de autor, es exclusivamente para fines educativos e informativos, y cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por UNIVERSIDAD TECMILENIO.

Queda prohibido copiar, reproducir, distribuir, publicar, transmitir, difundir, o en cualquier modo explotar cualquier parte de esta obra sin la autorización previa por escrito de UNIVERSIDAD TECMILENIO. Sin embargo, usted podrá bajar material a su computadora personal para uso exclusivamente personal o educacional y no comercial limitado a una copia por página. No se podrá remover o alterar de la copia ninguna leyenda de Derechos de Autor o la que manifieste la autoría del material.