**Estructura del Proyecto**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Carpeta/archivo** |  |  |  |  |
| build |  |  |  | Aquí se guarda el código compilado (para producción) a JavaScript plano |
| doc |  |  |  | Aquí se guarda el código HTML que documenta el uso de la API (generado automáticamente por apidoc) |
| node\_modules |  |  |  | Aquí se guarda el código de las librerías usado mientras se desarrolla |
| src |  |  |  | Aquí se guarda el código principal de la API en format TypeScript |
|  | common |  |  | Aquí se guardan los archivos que los componentes de la API tienen en común (por ejemplo, utils.ts) |
|  |  | utils.ts |  | Funciones que son útiles para cualquier componente como format\_datetime, generate\_sql\_update\_entries, entre otros |
|  | db |  |  | Aquí se guardan los archivos para crear la base de datos y conectarse a ella |
|  |  | create\_database.ts |  | Archivo que crea la base de datos "app" |
|  |  | create\_table.ts |  | Archivo que crea las tablas activities y students en la base de datos "app" |
|  |  | pool.ts |  | Archivo que crea un pool de conexiones (para poder realizar múltiples conexiones) |
|  | endpoints |  |  | Aquí se guarda el código de los dos enpoints /api/activities y /api/students |
|  |  | activities |  | Aquí se guarda el código de los dos enpoints /api/activities |
|  |  |  | controller.ts | Archivo que contiene las funciones que realizan consultas SQL (ej. Select, delete, update, étc.) y dan formato al objeto de respuesta |
|  |  |  | model.ts | Archivo que contiene las interfaces y objetos necesarios para describir la estructura de los objetos utilizados en las consultas de SQL y consultas HTTP realizados en este endpoint |
|  |  |  | router.ts | Archivo que determina qué consultas SQL (funciones de controller.ts) se deben llamar para responder a distintas llamadas del endpoint (por ej. GET /activities, UPDATE /activities/:id) |
|  |  | students |  | Aquí se guarda el código de los dos endpoints /api/students |
|  |  |  | controller.ts | Archivo que contiene las funciones que realizan consultas SQL (ej. Select, delete, update, étc.) y dan formato al objeto de respuesta |
|  |  |  | model.ts | Archivo que contiene las interfaces y objetos necesarios para describir la estructura de los objetos utilizados en las consultas de SQL y consultas HTTP realizados en este endpoint |
|  |  |  | router.ts | Archivo que determina qué consultas SQL (funciones de controller.ts) se deben llamar para responder a distintas llamadas del endpoint (por ej. GET /students, UPDATE /students/:id) |
|  | index.ts |  |  | Archivo que agrupa todo y sirve como punto de entrada para que TypeScript sepa qué archivos compilar en /build |
| .gitignore |  |  |  | Archivo que contiene los formatos que no se deben incluir en el repositorio de git |
| apidoc.json |  |  |  | Archivo de configuración que le indica a apidoc como generar la documentación |
| LICENSE |  |  |  | Licencia del código |
| package-lock.json |  |  |  | Lista de librerías instaladas que se usan en la aplicación |
| package.json |  |  |  | Lista de librerías instaladas que se usan en la aplicación |
| README.md |  |  |  | Página home del proyecto en github |
| tsconfig.json |  |  |  | Archivo de configuración que le indica a typescript cómo compilar el código que se llama en index.ts |