

Definición de Servicios Web

Juan Alberto Camacho Bolaños

16 de marzo de 2021

1. Caso de negocio

La aplicación será utilizada como una *ventanilla virtual* en la cual las personas que poseen una cuenta en un sistema existente (`prepaenlinea.mx`) podrán descargar información escolar y actualizar algunos de sus datos. El fin de la aplicación es acercarnos un poco al público que utiliza el sistema (chicos de entre 15 y 18 años).

2. Arquitectura de la solución

- Base de datos.
Existente, implementada en PostgreSQL y con un esquema definido. Se encuentra en su propio clúster.
- Repositorio de código.
Se utiliza GitLab para almacenar el código.
- Pipeline de integración continua.
Se tienen un par de servidores para implementar un pipeline de CI/CD para garantizar un mínimo de calidad en el desarrollo de proyecto.
- Servidor de pruebas.
Servidor que contiene un ambiente de pruebas idéntico al de producción, su base de datos puede ser destruida sin problemas. Tiene una ip pública que va cambiando. La base de datos de prueba se encuentra en este mismo servidor.
- Servidor de producción.
Ip pública y dominio definidos (`prepaenlinea.mx`) se conecta a la base de datos que se encuentra dentro de la misma nube privada virtual (VPC). Sólo las máquinas con autorización previa pueden comunicarse con el servidor, el acceso al shell del mismo se encuentra restringido y sólo se ha accedido a él cuando se prendió por primera vez. Toda la comunicación se hace a través del pipeline de CI/CD y los endpoints del API existente.

3. Tecnologías a utilizar

- **NodeJS**. Que sirve el API actual (REST).
- **PostgreSQL**. Como servidor de base de datos.
- **GitLab Runners**. Para implementar el pipeline de CI/CD.
- **Swift**. Para implementar el código de la aplicación en iOS.
- **Kotlin**. Para implementar el código de la aplicación en Android.

4. Modelo de datos

El modelo de datos se encuentra en producción por lo que el objetivo de la app será conectarse al mismo y obtener la información relevante, la figura 1 contiene un extracto de los datos que se utilizarán (aunque no es exhaustivo):

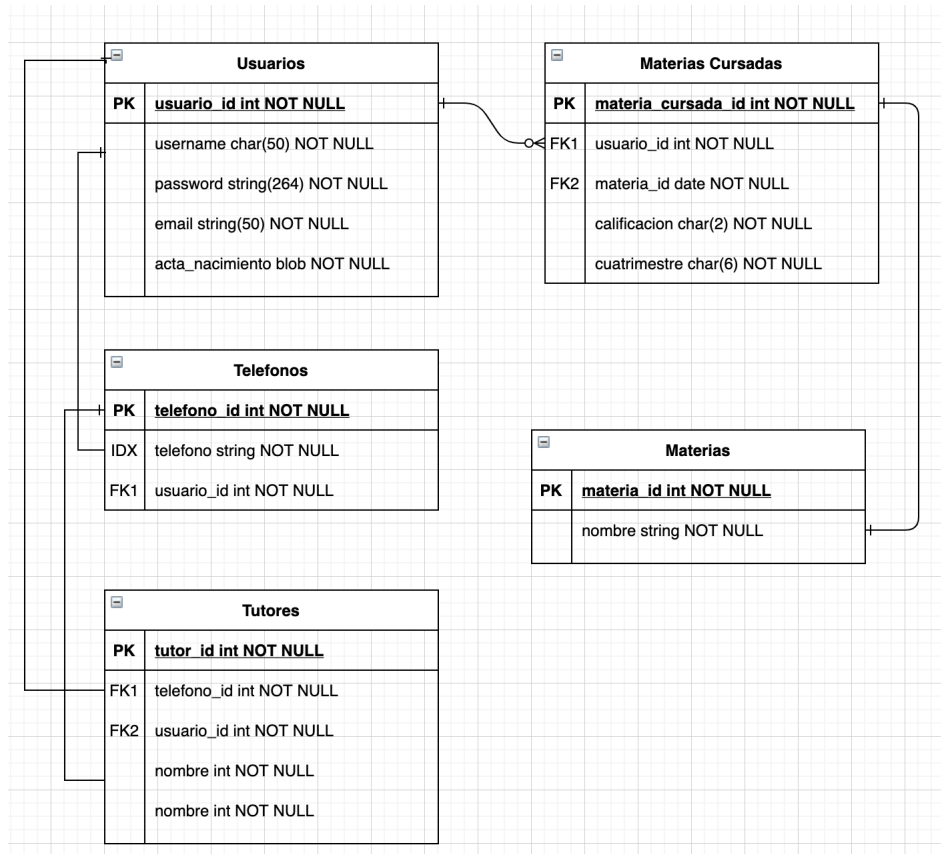


Figura 1: Diagrama del modelo de datos

5. Enpoints

Todos los enpoints incluyen las operaciones `list`, `create`, `update`, `read` y `delete` a menos que se indique explícitamente lo contrario.

- `/usuarios` Meta-modelo que contiene a los modelos Usuarios, Materias Cursadas, Tutores y Teléfonos.
- `/usuarios/id/tutores` Sirve para manejar los tutores de los usuarios.
- `/usuarios/id/telefonos` Maneja los teléfonos de los usuarios.
- `/usuarios/id/materias` Maneja las materias cursadas por el alumno.
- `/materias` Maneja las materias.