

INFORME DE ACTIVIDADES - PRESTACIÓN DE SERVICIOS PROFESIONALES

DENIS VIVEROS PAYAN
CONTRATISTA

Santiago de Cali, 23 de febrero de 2026

Doctor:
SERGIO NICOLÁS GUZMÁN RICO
Director
Departamento Administrativo de Planeación Departamental
Gobernación del Valle del Cauca

Asunto: Informe de Actividades SEGUNDA cuota del contrato de prestación de servicios profesionales No. 1.110.17.13.4814 de enero 17 de 2026.

Cordial saludo,

Por su calidad de Supervisor del Contrato de Prestación de Servicios Profesionales No. 1.110.19.13.4814 de enero 17 de 2026, suscrito entre el Departamento del Valle del Cauca – Departamento Administrativo de Planeación y Denis Viveros Payan, presento a usted la relación de actividades correspondientes al pago de la SEGUNDA cuota del contrato.

OBJETO DEL CONTRATO: Prestar servicios Profesionales en el Departamento Administrativo de Planeación como ingeniero de sistemas en el desarrollo del proyecto denominado ACTUALIZACIÓN DEL SERVICIO DE INFORMACIÓN GEOGRÁFICA, GEODÉSICA Y CARTOGRÁFICA DEL VALLE DEL CAUCA.

Acta de Inicio: 21 de enero de 2026
Fecha de Terminación: 30 de junio de 2026

Para el cumplimiento del Contrato de Prestación de Servicios Profesionales No. 1.110.19.13.4814 de enero 17 de 2026, se ha estipulado en las obligaciones del contratista:

1. Desarrollar servicios y aplicaciones para la distribución de información actualizada a la entidad y terceros, que sea interoperable entre los sistemas de información geográfica de la Idee Valle.
2. Brindar soporte técnico, configuración, actualización y mantenimiento a los sistemas de información y/o base de datos geográficas del Departamento del Valle del Cauca
3. Asistir técnicamente a entidades territoriales, centralizadas y descentralizadas, en el funcionamiento y operatividad de los sistemas de información, sistemas y bases de datos del Valle del Cauca

INFORME DE ACTIVIDADES - PRESTACIÓN DE SERVICIOS PROFESIONALES

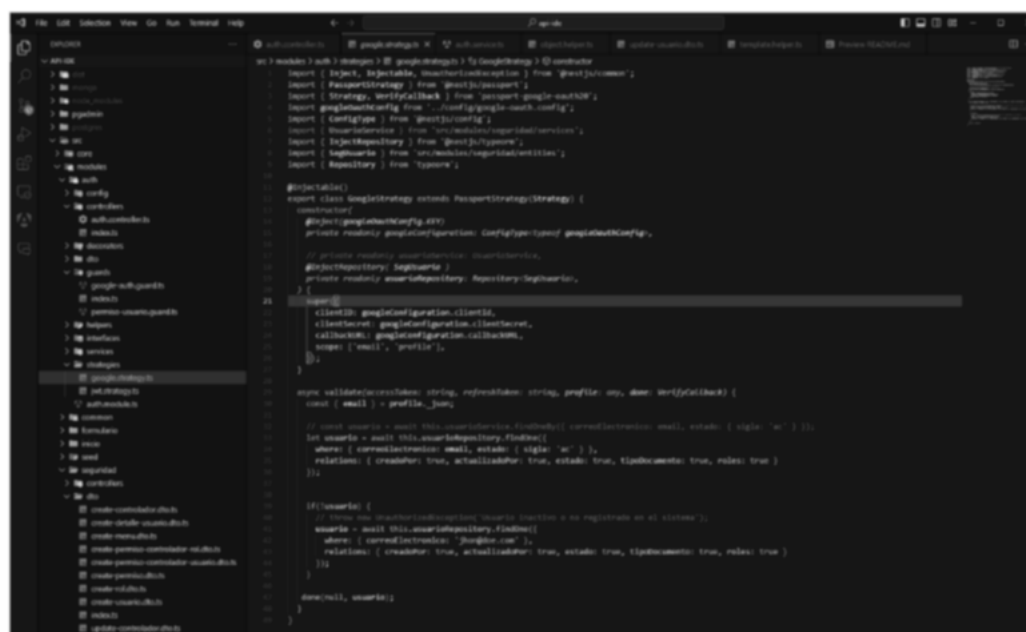
DENIS VIVEROS PAYAN
CONTRATISTA

4. Cumplir con las obligaciones a su cargo, así mismo, cumplir con las instrucciones que le imparta el supervisor del contrato que tenga relación directa con el objeto contractual.

En cumplimiento de lo anterior se ha avanzado en las siguientes acciones:

1. **Desarrollar servicios y aplicaciones para la distribución de información actualizada a la entidad y terceros, que sea interoperable entre los sistemas de información geográfica de la Idee Valle.**

- 1.1. Desarrollé endpoints en IDE-API *google/login* y *google/callback* encargados de la gestión del inicio de sesión con Google dentro de la plataforma de gestión de la IDEE. El desarrollo consistió en desarrollar la estrategia en el backend encargada de verificar la información del usuario entregado por Google, posteriormente permitir o restringir el acceso al usuario una vez verifique el estado del usuario dentro de la base de datos de la IDEE.



```
import { Injectable, UnauthorizedException } from '@nestjs/common';
import { PassportStrategy } from '@nestjs/passport';
import { Strategy, VerifyCallback } from 'passport-google-oauth20';
import { GoogleStrategyConfig } from 'src/config/google-strategy.config';
import { ConfigType } from '@nestjs/config';
import { UserRepository } from 'src/modules/user/user.repository';
import { AuthService } from 'src/modules/auth/auth.service';
import { User } from 'src/modules/user/user.entity';
import { Repository } from 'typeorm';

@Injectable()
export class GoogleStrategy extends PassportStrategy(Strategy) {
  constructor(
    @Inject(GoogleStrategyConfig.KEY) config: ConfigType<GoogleStrategyConfig>,
    private readonly userRepository: UserRepository,
    private readonly authService: AuthService,
  ) {
    super({
      clientID: config.clientID,
      clientSecret: config.clientSecret,
      callbackURL: config.callbackURL,
      scope: ['email', 'profile'],
    });
  }

  async validate(credentials: string, refreshToken: string, profile: any, done: VerifyCallback) {
    const { email } = profile._json;

    // Check if user exists in the database
    let user = await this.userRepository.findOne({
      where: { email },
    });

    if (!user) {
      // Create new user
      user = await this.authService.createUser({
        email,
        password: '12345678',
        role: 'user',
      });
    }

    // Generate token
    const token = this.authService.generateToken(user);

    done(null, user);
  }
}
```

Google.strategy.ts

DENIS VIVEROS PAYAN
CONTRATISTA

[illegible]

Auth.controller.ts – endpoints encargados de la gestion del inicio de sesion con Google

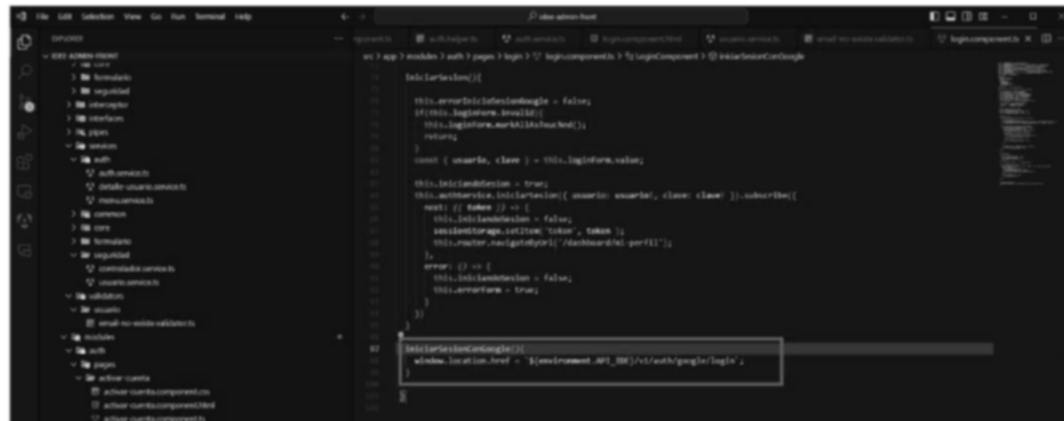
- 1.2.** Realicé integración de los endpoints desarrollados para el inicio de sesión con Google con el proyecto de gestión de la IDEE (IDEE-ADMIN-FRONT). Esta integración permite que los usuarios registrados pueden iniciar sesión en la plataforma a través de su cuenta de Google.

[illegible]

Componente login.component.html

INFORME DE ACTIVIDADES - PRESTACIÓN DE SERVICIOS PROFESIONALES

DENIS VIVEROS PAYAN
CONTRATISTA



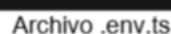
Componente login.component.ts



Visualización del componente de inicio de sesión. Boton de Inicio de Sesión con Google.

DENIS VIVEROS PAYAN
CONTRATISTA

2.1. Agregué al archivo `.env`, `.env.dev` y `docker-compose.yaml` las variables de entornos `DOMINIO_IDEEADMINFRONTEND`, `GOOGLE_CLIENT_ID`, `GOOGLE_SECRET`, `GOOGLE_CALLBACK_URL` necesarias para el proceso de configuración del inicio de sesión con Google dentro de la plataforma de gestión de la IDEE.

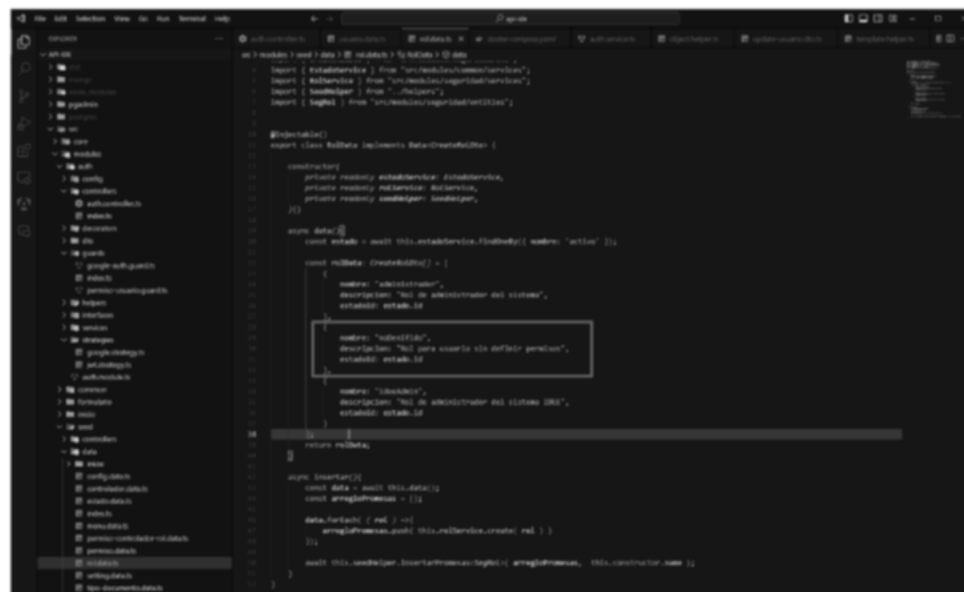


INFORME DE ACTIVIDADES - PRESTACIÓN DE SERVICIOS PROFESIONALES

DENIS VIVEROS PAYAN
CONTRATISTA

3. Asistir técnicamente a entidades territoriales, centralizadas y descentralizadas, en el funcionamiento y operatividad de los sistemas de información, sistemas y bases de datos del Valle del Cauca.

- 3.1. Realicé modificación en el archivo *usuario.data* y *rol.data* dentro de del proyecto IDE-API. Dicha modificación consistió en incluir el rol “noDefinido” y un usuario que utilizara este rol “Jhon Doe” utilizado como comodín en los casos en que un usuario intente iniciar sesión en la IDEE y no se encuentre registrado o activo dentro del sistema.



```
import { Role } from '@prisma/client';
import { RoleData } from '@prisma/client';

export class RoleData implements RoleData {
  constructor(
    private readonly rol: Role,
    private readonly rolNombre: string,
    private readonly rolDescripcion: string,
    private readonly rolEstado: boolean,
  ) {}

  static async findRoleById(rolId: string): RoleData {
    const rolData = await this.rolData.findUnique({
      where: { rolId },
    });

    if (!rolData) {
      throw new Error('Rol no encontrado');
    }

    return rolData;
  }

  static async findRoleByRol(rol: Role): RoleData {
    const rolData = await this.rolData.findUnique({
      where: { rol },
    });

    if (!rolData) {
      throw new Error('Rol no encontrado');
    }

    return rolData;
  }

  static async findRoleByRolNombre(rolNombre: string): RoleData {
    const rolData = await this.rolData.findUnique({
      where: { rolNombre },
    });

    if (!rolData) {
      throw new Error('Rol no encontrado');
    }

    return rolData;
  }

  static async findRoleByRolDescripcion(rolDescripcion: string): RoleData {
    const rolData = await this.rolData.findUnique({
      where: { rolDescripcion },
    });

    if (!rolData) {
      throw new Error('Rol no encontrado');
    }

    return rolData;
  }

  static async findRoleByRolEstado(rolEstado: boolean): RoleData {
    const rolData = await this.rolData.findUnique({
      where: { rolEstado },
    });

    if (!rolData) {
      throw new Error('Rol no encontrado');
    }

    return rolData;
  }
}
```

Archivo rol.data.ts



```
import { User } from '@prisma/client';
import { UserData } from '@prisma/client';

export class UserData implements UserData {
  constructor(
    private readonly usuario: User,
    private readonly usuarioNombre: string,
    private readonly usuarioApellido: string,
    private readonly usuarioEmail: string,
    private readonly usuarioPassword: string,
    private readonly usuarioEstado: boolean,
  ) {}

  static async findUserById(usuarioId: string): UserData {
    const userData = await this.userData.findUnique({
      where: { usuarioId },
    });

    if (!userData) {
      throw new Error('Usuario no encontrado');
    }

    return userData;
  }

  static async findUserByUsuario(usuario: User): UserData {
    const userData = await this.userData.findUnique({
      where: { usuario },
    });

    if (!userData) {
      throw new Error('Usuario no encontrado');
    }

    return userData;
  }

  static async findUserByUsuarioNombre(usuarioNombre: string): UserData {
    const userData = await this.userData.findUnique({
      where: { usuarioNombre },
    });

    if (!userData) {
      throw new Error('Usuario no encontrado');
    }

    return userData;
  }

  static async findUserByUsuarioApellido(usuarioApellido: string): UserData {
    const userData = await this.userData.findUnique({
      where: { usuarioApellido },
    });

    if (!userData) {
      throw new Error('Usuario no encontrado');
    }

    return userData;
  }

  static async findUserByUsuarioEmail(usuarioEmail: string): UserData {
    const userData = await this.userData.findUnique({
      where: { usuarioEmail },
    });

    if (!userData) {
      throw new Error('Usuario no encontrado');
    }

    return userData;
  }

  static async findUserByUsuarioPassword(usuarioPassword: string): UserData {
    const userData = await this.userData.findUnique({
      where: { usuarioPassword },
    });

    if (!userData) {
      throw new Error('Usuario no encontrado');
    }

    return userData;
  }

  static async findUserByUsuarioEstado(usuarioEstado: boolean): UserData {
    const userData = await this.userData.findUnique({
      where: { usuarioEstado },
    });

    if (!userData) {
      throw new Error('Usuario no encontrado');
    }

    return userData;
  }
}
```

Archivo usuario.data.ts

INFORME DE ACTIVIDADES - PRESTACIÓN DE SERVICIOS PROFESIONALES

DENIS VIVEROS PAYAN
CONTRATISTA

4. Cumplir con las obligaciones a su cargo, así mismo, cumplir con las instrucciones que le imparta el supervisor del contrato que tenga relación directa con el objeto contractual.

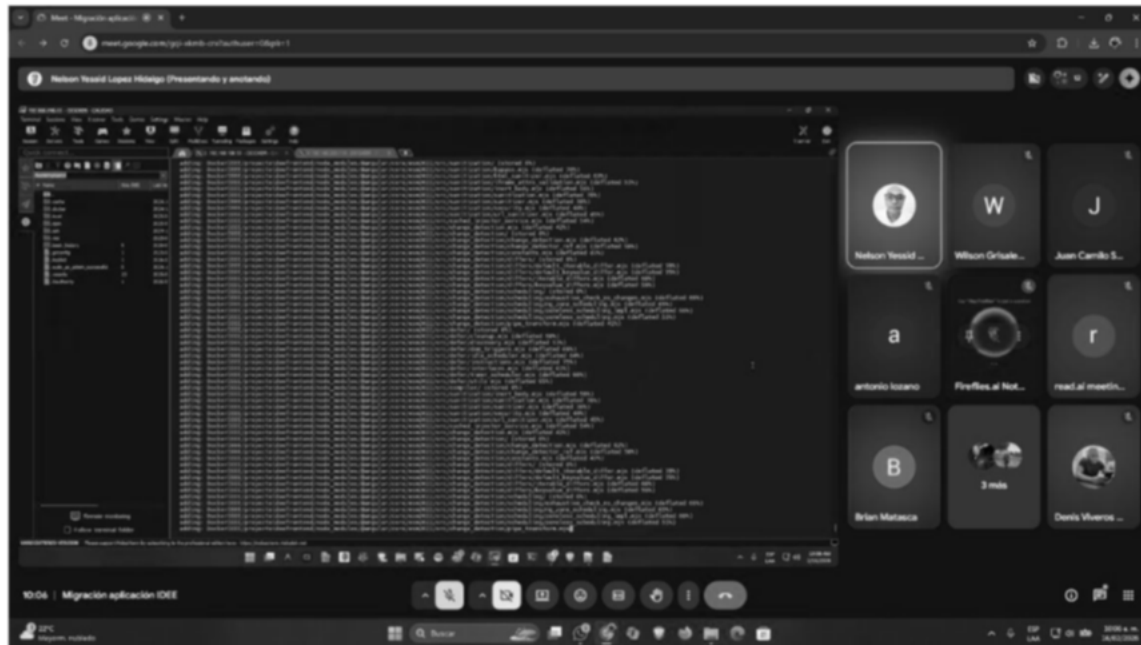
- 4.1.** Asistí a la mesa de trabajo virtual en la que se realizó la presentación del Brian Matasca como pasante en el grupo de la Idee; de manera posterior, se le hizo la presentación del desarrollo del proyecto IDEE.



- 4.2.** Asistí a la mesa de trabajo virtual solicitada por SETIC con el objetivo verificar y completar el despliegue de la aplicación IDEE desde calidad a producción, incluyendo sincronización con GIT, compilaciones y apuntes DNS.

INFORME DE ACTIVIDADES - PRESTACIÓN DE SERVICIOS PROFESIONALES

DENIS VIVEROS PAYAN
CONTRATISTA



Cordialmente,

Denis Viveros Payan
DENIS VIVEROS PAYAN
Contratista
C.C. 1.111.749.824
Cel. 3207599993.
denisvipa@homail.com