

## Requisitos.

Control de código y tareas:

**GitHub:** acceso completo a todos los repos (si es que hay mas de uno)

- Poder abrir ramas, PRs, tags y releases sin bloquearme.
- Que me pasen las reglas de branching / naming (ej. Git Flow).

**Monday:** Permisos a todos los tableros de IT y a los demás tableros que sean necesarios

- Acceso al historial de tickets/bugs y plantillas de PR.

### Bases de datos y entornos:

- Usuarios y strings de conexión para dev, stage y prod (si es que existen)  
**[READONLY ACCESS]**
- Scripts de creación/migración y datos semilla.
- Diagrama ERD actualizado. (Si es que existe)
- Backups automáticos + cómo restaurarlos.
- Qué versión exacta de SQL Server corre cada ambiente.

### Infra y despliegues:

- Acceso al pipeline CI/CD para ver builds/tests/deployes. (si es que existen)
- Permisos en los servidores o servicios de hosting.
- Logs centralizados (Event Viewer, ELK, Grafana, etc.).

### Setup local:

- Laptop/PC con Windows.
- Versión exacta de Visual Studio / Visual Basic que usa el proyecto.
- SDK/Runtime .NET o Framework correcto.
- Drivers ODBC/OLE DB necesarios.

- Archivo .env de ejemplo o script de configuración.
- Plugins útiles (ReSharper, SQL Prompt, etc.).
- Gestor de paquetes listo (NuGet, npm, pip...).

#### **Credenciales y seguridad:**

- Variables de entorno sensibles bien documentadas.
- Usuario “normal” y otro admin para el SGI.
- Tokens de APIs externas, certificados TLS, etc.

#### **Productividad:**

- Correo @sierra para tener la suite Office.
- Teams o lo que esten usando para comunicarse, slack, grupo de whatsapp, etc.
- Calendarios compartidos (reuniones, ventanas de deploy).

#### **Docs y know-how:**

- Manual de arquitectura (diagramas, capas, flujos clave, o lo que sea que haya documentado para comenzar a trabajar en un modulo).
- Guías de estilo de código.
- Roadmap y changelog.

#### **Testing & QA:**

- Tests automáticos (unit, integration, UI) + reportes de cobertura.
- Casos de prueba manual prioritarios.

#### **Activos del SGI:**

- Licencias de componentes de terceros (VB controls, ActiveX, reportes, fuentes).
- Documentación de integraciones con otros sistemas

#### **Soporte & operaciones:**

- Procedimientos de emergencia: rollback, restore, hotfix.

## Git:

david.sandoval@sierratextiles.com

## Monday:

Pablo Hernandez lo estará viendo.

## Bases de datos:

### Usuario

david

### Contraseña

D4v!d2025.Gt\*Sierra

Host: 192.168.4.9  
Versión del servidor: 5.5.46-0ubuntu0.14.04.2 - (Ubuntu)  
SO del servidor: debian-linux-gnu  
HeidiSQL Versión: 12.2.0.6576

Se adjuntan 3 archivos en los que se muestran los diferentes conexiones con sus respectivos datos. Este punto se explicó en la inducción.

Los respaldos se realizan por medio de tareas, donde se crea la copia de seguridad y se guarda en otro equipo. Adicional se realiza un respaldo manual de estas copias en un disco duro.

## Infra y despliegues:

Iniciar session con google – (credenciales abajo)

<https://hpanel.hostinger.com/> **Business Web Hosting / File Manager / Public\_html / Imagenes**

usuario: [admin.karims@hilosyalgodon.com](mailto:admin.karims@hilosyalgodon.com)

contraseña: [V1\\$@16=y?U9](#)

## Setup Local

Microsoft Visual Studio Community 2019 - Versión 16.11.36

CR .net version 13.0.3500.0

Zxing .net version 0.16.9.0

## Credenciales y seguridad:

Usuario SGI a todas empresas (dsandoval, 12345 -pedirá cambio al iniciar sesión-, copiar accesos Misrra)

## Productividad:

David.sandoval@sierratextiles.com

**Docs y know-how**

**Testing & QA**

**Activos del SGI**

**Soporte y Operaciones**

- En caso de emergencia de Base de datos, se asegura que el servidor no esté comprometido (cambio de contraseñas), luego se restablecen las bases de datos, y se inician operaciones (si se pierde información será de horas y tendrán que registrar lo de esas horas nuevamente)