**API**

1. Nueva terminal
   1. dotnet new sln
2. En solution Explorer
   1. Add new project
   2. ASP .NET Core Web API
   3. C#
   4. Nombre Proyecto API
3. Instalar Extenciones con Nuget
   1. Ctrl + shift +P
   2. 
   3. 
   4. 
4. Cerrar VSCode, abrir de nuevo, clickar en barra navegación izquierda de solución (icono) y aceptar el cartel
   1. Texto

      Descripción generada automáticamente
5. Agregar Carpeta Comandos
6. Agregar Carpeta Resultados
7. Crear Carpeta Models
   1. Abrir terminal
   2. cd API
   3. dotnet ef dbcontext scaffold “Server=localhost; Database=Prog3-11- 05; Port=5432; User Id=ClaseProg3; Password=123456;” Npgsql.EntityFrameworkCore.PostgreSQL --output-dir Models
8. Configurar Program.cs
   1. Buscar en github
9. Configurar appSettings.json
   1. Buscar en github
10. Crear Controlador para Usuarios
    1. En carpeta Controllers crear UsuarioController.cs
    2. Copiar de Github
11. Crear Resultados para Usuarios
    1. Crear archivo ResultadoBase.cs
       1. Copiar de Github
    2. En carpeta Resultados crear Carpeta Usuarios
       1. Crear archivo ResultadoUsuarioGet.cs
       2. Copiar de Github
12. Crear Comandos para Usuarios
    1. En carpeta Comandos crear carpeta Usuarios
       1. ComandoUsuarioPost.cs
       2. Copiar de Github

**ANGULAR**

1. Crear nuevo Proyecto de Angular
   1. En consola: ng new nombreProyecto
2. Levantar proyecto (dentro de carpeta del Proyecto):
   1. En Consola: cd Frontend
   2. ng serve
3. Colocar en src/environments/environment.ts las urls de API
   1. Texto

      Descripción generada automáticamente
4. En src/index.html importar Bootstrap y otras librerías
   1. Texto

      Descripción generada automáticamente
5. En src/app/app.component.html dejar lo necesario
   1. Pantalla de computadora con letras

      Descripción generada automáticamente con confianza media
6. En src/app/app.component.ts programamos la funcionalidad del componente
   1. Pantalla de computadora con letras

      Descripción generada automáticamente con confianza media
7. Definir las redirecciones en src/app/app.routing.module.ts, de modo que según la rota especifico que componente me va a tener que abrir
   1. Pantalla de computadora con letras

      Descripción generada automáticamente con confianza media
   2. Buscar en Github
8. En src/app/app.module.ts declaramos todos los servicios que van a ser inyectados en nuestro código
   1. Texto

      Descripción generada automáticamente
   2. Buscar en Github e Importar
9. Crear carpeta de Providers en src
   1. Crear usuarioProvider.ts
   2. Copiar código de Github
   3. Importar Provider en appComponent
10. Crear carpeta de Interfaces en src/app
    1. Crear Usuario.ts como interfaz
    2. Copiar código de Github
11. En src/app/app.component.ts
    1. Programaccion del componente principal que se inyecta en el Index.html
12. En src/app/app.component.html
    1. Definir barra de navegación con los respectivos enlaces de las rutas definidas en el App Routing
13. Creacion de Componentes
    1. ng generate component nombreComponente
    2. Copiar de Github