

Formulario de Arquitectura

Programación de Aplicaciones Visuales

Fecha de entrega: 21/11/2018.

Integrantes:

Francisco Barafani, 72750

Marcos Mariatti, 72707

Boné Nicolás, 72624

Razón Social de la Empresa:

DJ Fran Barafani

Diseño por Capas

Dividimos los archivos en capas de datos, interfaces, modelos y reportes de forma tal de preservar los principios del paradigma orientado a objetos de la mejor forma posible.

Servidor DB

Cada uno descargo SQLServer Management Studio y SQLServer en sus computadoras personales, y cada uno maneja una base de datos personal.

Conexión DB

Para la conexión personal de cada uno con la DB decidimos agregar el archivo del DBHelper al gitignore para que no se subiera al repositorio cada vez que hacíamos un push, así cada uno tenía su string de conexión intacto.

Mantenimiento del Código

Desde su concepción, el proyecto fue subido a Github y mantenido por todos los integrantes. Los cambios se pueden encontrar en <https://github.com/MMariatti/ProyectoPav2018>.

División de Tareas

Utilizamos Trello, una herramienta web que permite crear tableros (en este caso “Falta Realizar”, “En Proceso”, “Terminado”), y dentro de los mismos, crear tarjetas con su descripción (cosas por hacer). Una vez creadas todas las tarjetas cada usuario puede asignarse tarjetas y desplazarlas a los tableros pertinentes. Esto nos ayudó mucho a dividir las tareas y organizar la igualdad de trabajo.

Flujo de Uso

Nuestro programa se centra en la navegación por medio de botones cuyas funciones quedan definidas por el texto visible al usuario. Buscamos que el programa fuera intuitivo de usar y requiriera la menor cantidad de interacción posible, para facilitar su uso y evitar errores innecesarios.

Estética

Intentamos buscar simetría en todas las interfaces que realizamos, además de minimizar la cantidad de objetos visibles al usuario. A pesar de todo esto, la interfaz del programa resultó más robusta de lo que esperábamos.

Avisos al Usuario

Utilizando properties(get set) en las clases de los objetos que representaban las entidades de la DB, desvinculamos la necesidad de comprobar los tipos de datos que se ingresaban al sistema ya que los objetos anunciarían que los tipos de datos ingresados son incorrectos al momento de su creación, sin necesidad de comprobarlo en la función del formulario (filtros aplicados al private set).

Configuración del Proyecto

Tuvimos múltiples problemas intentando instalar los paquetes de reportes, pero pudimos lograr solucionarlos. Todos utilizamos versiones de prueba del Visual Studio 2017.