



NB

NoteBugs

DOCUMENTO DE ARQUITECTURA

Herramientas de desarrollo

Tipo de herramienta	Proveedor/Product o/Versión
Versionamiento	Github
Database	Microsoft SQL Server
GUI Development Tool	Visual studio community 2019
Host de base de datos	SmarterASP.NET
Testeo	Visual studio community 2019

Requisitos de Seguridad

Los usuarios deberán estar registrados y autenticado. Solo un usuario autenticado podrá ver el contenido de los modelos de los que es propietario.

Acuerdos de programación

El diseño del sistema y su construcción deberá contemplar la división entre la capa de acceso a la bd, la lógica de negocio, las entidades del negocio y la interfaz para optimizar la escalabilidad de la aplicación.

En general para los nombres se utilizará CamelCase, donde cada palabra comienza con una letra mayúscula

Métodos: Los nombres deben estar en mayúsculas y minúsculas. Usar verbos para describir lo que hace el método

Variables: Los nombres debe iniciar la primera letra en minúsculas; continuando deben en mayúsculas y minúsculas.

Los nombres deben representar lo que el valor de la variable representa.

Acuerdos de interfaz de usuario

- Coherencia de los elementos de la interfaz: Una apariencia coherente para aportarle armonía a la aplicación, los mismos colores que se visualizan en el código debían estar en toda la aplicación, la forma de los botones debe ser igual en todos los formularios, respetar el lugar y la forma de los controles de toda la aplicación, utilizando siempre la misma tipografía y tamaño.
- Diseñar pensando en la facilidad de uso: a la hora de testear lo que hacíamos pensábamos como si fuéramos el usuario final.

- Menú moderno: con opciones bien definidas además de contar con imágenes explicativas y desarrollado de la mejor manera para una buena interactividad

Problemas y solución

- Utilización de la misma BD entre los distintos integrantes del grupo: Decidimos colgar la BD en un servidor para poder utilizar la misma cadena de conexión todos y no tener que ir cambiándola, lo que nos sirvió para tener una consistencia entre los datos y la forma de manejarlos.
- En el transcurso del cuatrimestre un integrante del grupo decide dejar la carrera, por lo que pasamos a ser dos lo que nos llevo a tener que reorganizar la forma de trabajar.
- Problemas con el repositorio a la hora de programar los reportes y estadísticas, debido a configuraciones especiales que tenía el visual que dependían de cada máquina con sus extensiones de visual para reportes instalados, por lo que hicimos otro repositorio siendo mas exhaustivos en el gitignore.
Link al primer repositorio: https://github.com/frandacos/PAV_3K1_2020
Link al repositorio final: https://github.com/CanPe00/PAVI_3K1_GRUPO3

Aprendizajes

- Codificación de una aplicación visual con un paradigma orientado a objetos en donde además nos enfocamos en dividir en capas la misma cumpliendo con los requisitos de arquitectura.
- Conocer un nuevo lenguaje de programación, C# de la empresa Microsoft como parte de su plataforma .NET
- Utilización de un nuevo IDE, en nuestro caso, utilizándolo para la creación de aplicaciones web (Windows Form)
- Manejo de la persistencia de los datos, utilizando una base de datos (creada con los lineamientos de un paradigma estructurado en Microsoft SQL Server), a partir de una aplicación web creada con un paradigma orientado a objetos.
- Codificación en grupo (división de tareas, confianza en nuestros compañeros, cumplir plazos de entrega establecidos, etc.)

Distribución de roles en el equipo

Para afrontar cada entrega fuimos intercambiando los roles, es decir, intentamos rotar de acuerdo a las tareas a realizar, teniendo en cuenta que todos podamos hacer todo.

Antes de comenzar, planificábamos que parte debía hacer cada uno y poníamos una fecha limite para luego poder hacer un testeo de lo realizado y las modificaciones finales para mejoras.

La virtualidad y en especial, este trabajo practico, nos ayudó a aprender a trabajar en grupo, debido que hizo que tengamos muchos problemas, de comunicación, de tiempos, de entendimiento, etc. Y nos vimos obligados a esforzarnos por dejar las cosas claras, por ejemplo, en las tareas a realizar, en quien iba a tomar cada rol en cada entrega, en cómo queríamos que salgan las cosas, etc.

Los diferentes roles que hemos puesto en práctica son:

- Administrador de base de datos (La mayoría de las veces fue realizado por Cosci, Franco)
- Desarrollador
- Diseñador de interfaz de usuario
- Coordinador de proyecto (La mayoría de las veces fue realizado por Pérez, Candela)
- Tester
- Documentador