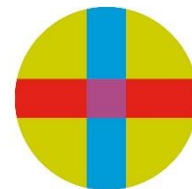


PROYECTO 1: GESTOR ALQUILER DE PRODUCTOS.



**1º en Ingeniería de Sistemas de
la Información**

Escuela Politécnica Superior

Universidad San Pablo CEU

David Sánchez Simón

Fecha de Finalización 15/12/2016



- **Resumen Ejecutivo**
- **Gestión del proyecto**
 - Recursos utilizados.
 - Diagrama de GANTT.
 - Diagrama de PERT.
 - Diagrama de Esfuerzo.
- **Diseño**
 - Diagrama de Paquetes.
 - Diagrama de Clases (Más importantes).
 - Diagramas de interacción
- **Implementación**
- **Lecciones Aprendidas**
- **Líneas Futuras**
- **Bibliografía**

Resumen Ejecutivo.

El proyecto que hemos llevado a cabo se trata de un Gestor de alquiler de productos, concretamente clientes y coches, por lo tanto, podríamos decir que el uso está enfocado hacia, talleres, concesionarios, o empresas de alquiler automovilístico.

El funcionamiento del programa consistirá básicamente en un menú para cada apartado. Cliente, coche y alquiler. Escogeremos una opción y nos enlazara con otro menú para cada apartado, en el que podremos añadir, borrar, modificar, buscar o mostrar.

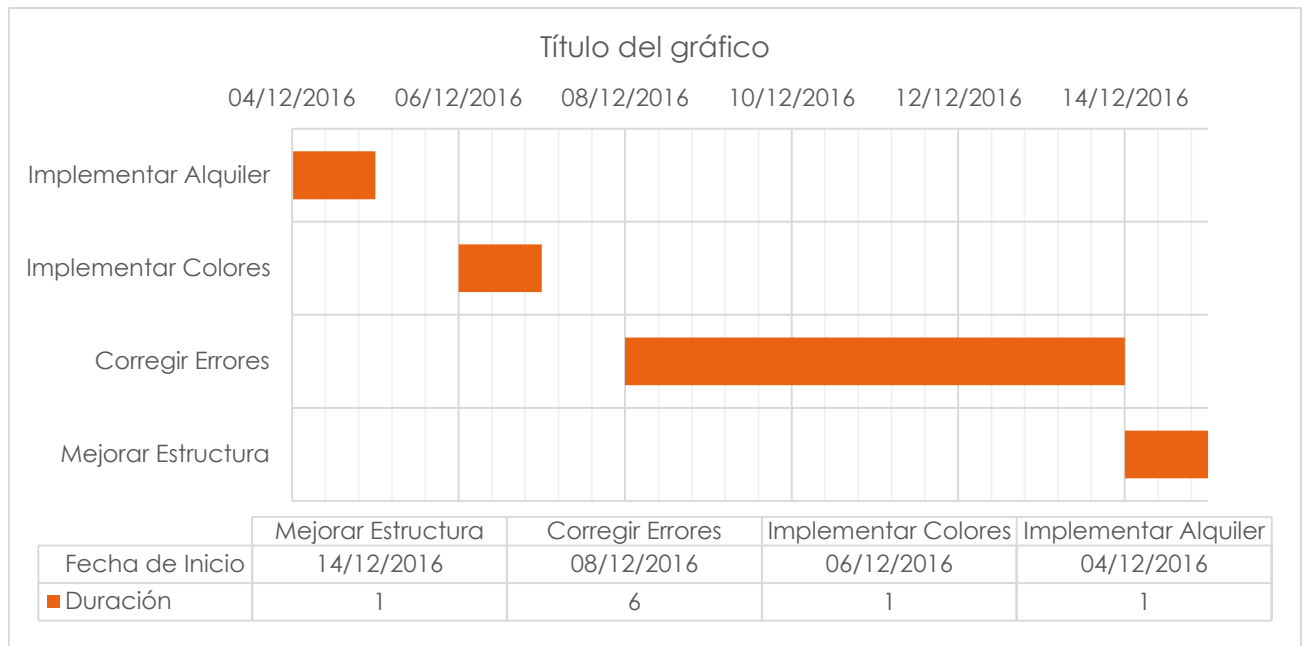
La funcionalidad de manera más sintetizada consistiría en la capacidad de introducir un producto u o servicio y almacenarlo en una hoja de cálculo, y poder consultarlo más tarde desde la propia hoja.

Concretamente en esta entrega desarrollaremos la Capacidad de añadir, borrar y modificar un alquiler de una hoja de cálculo, Además implementaremos colores a cadenas de caracteres y corregiremos y mejoraremos ciertos aspectos de la aplicación.

Gestión del proyecto

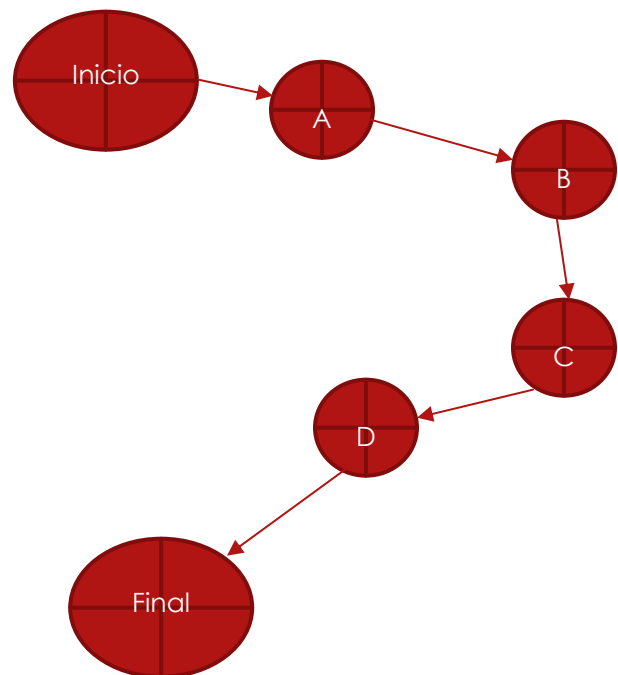
- Recursos utilizados
- Ordenador Packard Bell EasyNote TM85.
- Lenguaje de Programación: JAVA
- Entorno de trabajo: NetBeans

○ Diagrama de GANTT

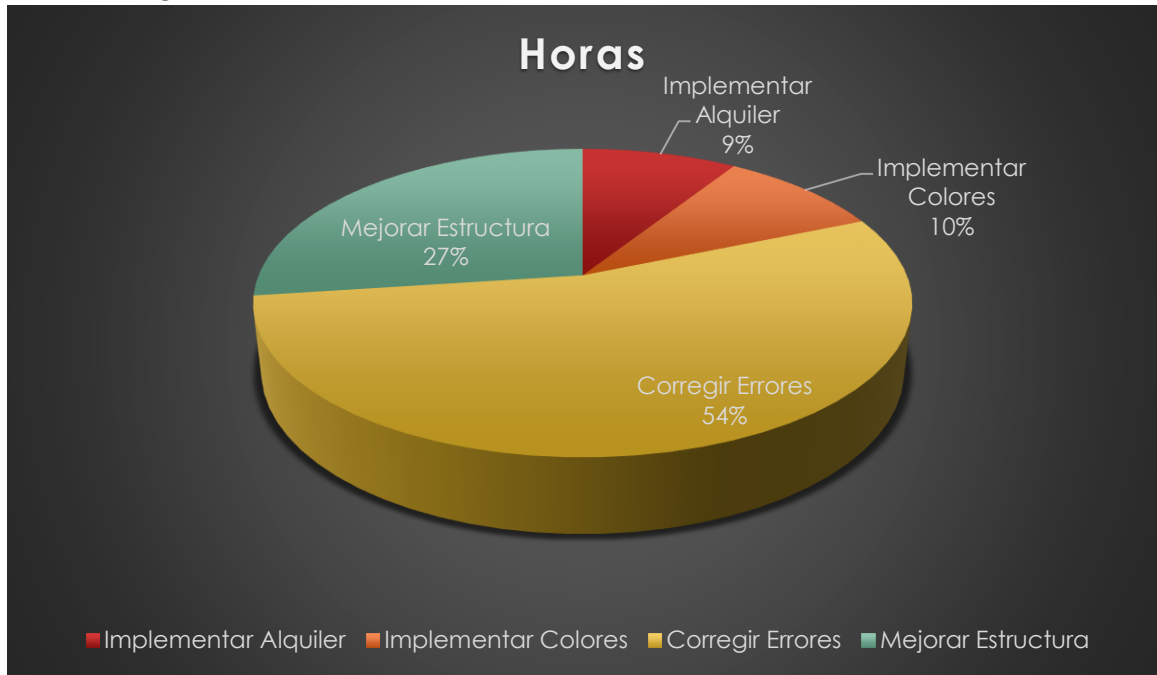


○ Diagrama de PERT

Actividad	Código	Duración
Implementar Alquiler	A	1
Implementar Colores	B	1
Corregir Errores	C	6
Mejorar Estructura	D	1

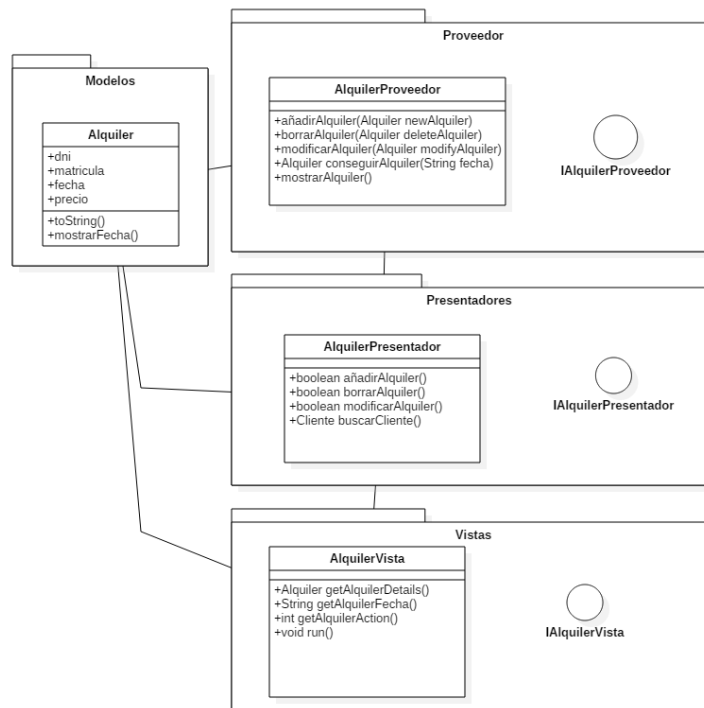


- Diagrama de Esfuerzo

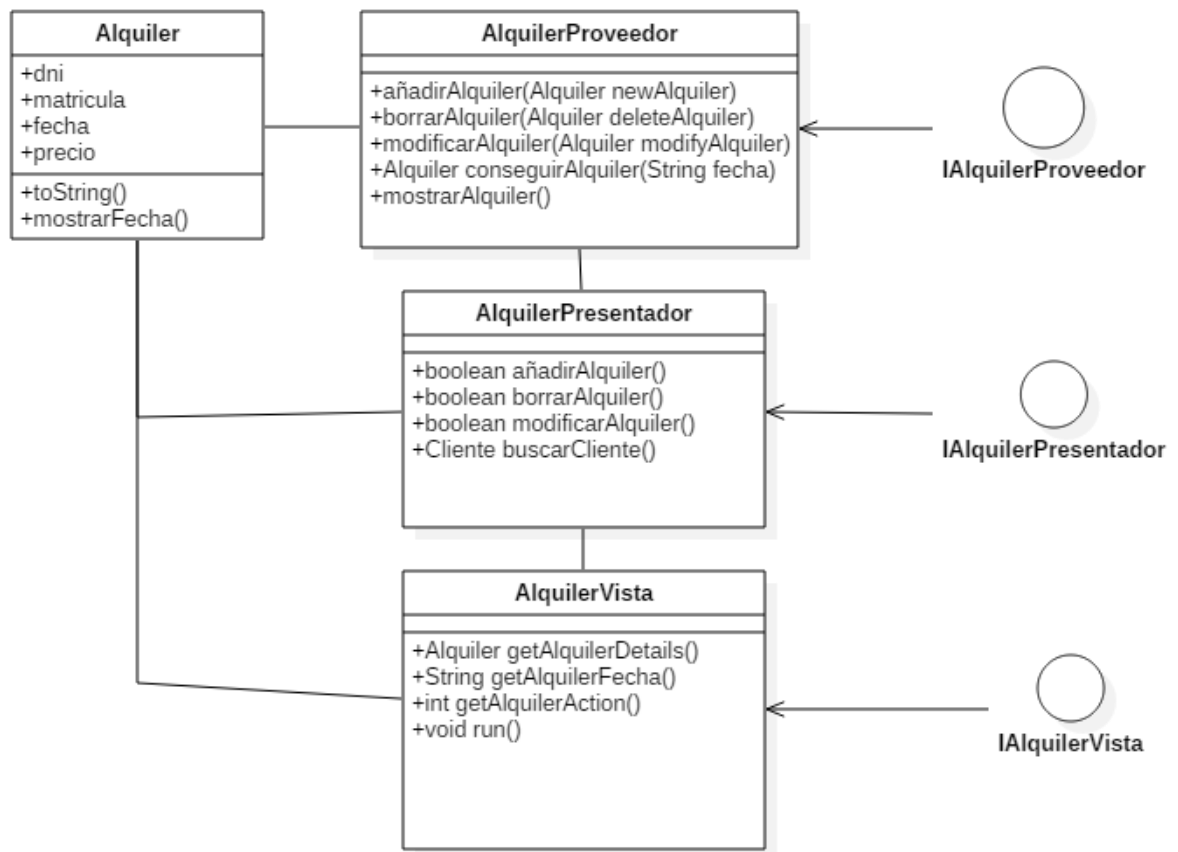


Diseño

- Diagrama de paquetes



- Diagrama de clases



Implementación

- Esta es una de las partes principales que destacaría de esta entrega (**"parte en negrita"**). En esta parte se encuentra la lógica para la

devolución de los datos de un cliente cuando introducimos su DNI, y en el caso de que no existe nos lo diga.

```
public Alquiler GetAlquilerDetails () throws IOException
{
    Alquiler newalquiler = new Alquiler ();
    String continuamos;
    do
    {
        String dni = GetCIDni ();
        if (!"". equals(dni))
        {
            Cliente clienteEncontrado = buscarcliente(dni);
            if (clienteEncontrado == null)
            {
                System.out.println("El cliente no existe");
            }
            else
            {
                mostrarCliente(clienteEncontrado);
                newalquiler. SetDni(dni);
                newalquiler.SetMatricula(GetChMatri());
                newalquiler.SetFecha(GetAlquilerFecha());
                System.out.println("Introduce el precio");
                newalquiler.SetPrecio(console.readLine());
                System.out.println("¿Desea introducir un nuevo alquiler(S/N)?");
                continuamos = console.readLine();
            }
        }
    }
    while (continuamos.startsWith("s") || continuamos.startsWith("S"));
}
```

- Otra de las partes que destacaría es la forma de guardar y cargar datos de la propia hoja de cálculo.

```
public ArrayList<Alquiler> cargarAlquileresDesdeArchivo()
{
    ArrayList<Alquiler> listaalquiler1 = new ArrayList<>();
    try
    {
        Scanner sc = new Scanner(new File(this.nombreArchivo)).useDelimiter("\n");
        sc.next();
        while(sc.hasNext())

```

```

        {
            Alquiler alquiler = new Alquiler(sc.next());
            listaalquiler1.add(alquiler);
        }
    }
    catch(FileNotFoundException ex)
    {
        Logger.getLogger(AlquilerProveedor.class.getName()).log(Level.SEVERE, null, ex);
    }
    return listaalquiler1;
}

private void volcarAlquileres() //antes de volcar manera de borrar todo lo que haya
dentro/*porque lo que cargamos es la lista.*/
{
    try {
        FileWriter fw = new FileWriter(new File(this.nombreArchivo));
        fw.write("DNI;MATRICULA;FECHA;PRECIO\n");
        for (Alquiler alquiler : listaalquiler)
        {
            fw.write(alquiler.toString()+"\n");
        }
        fw.close();
    }
    catch (IOException ex)
    {
        Logger.getLogger(AlquilerProveedor.class.getName()).log(Level.SEVERE, null, ex);
    }
}

```

Lecciones aprendidas

- En esta última práctica he aprendido algo más sobre el funcionamiento de interfaces, y de los archivos CSV y su utilidad en este tipo de programas. Otra de las cuestiones que he aprendido, es que debido a la forma en la que realice la anterior práctica las modificaciones en esta entrega han sido prácticamente irrisorias, es decir que, desarrollando una aplicación eficiente desde el principio, ahorramos mucho tiempo a la hora de realizar futuras modificaciones en la misma. Por último, otra de las lecciones que he aprendido, es poner color a las cadenas de caracteres.

Líneas futuras

- Algunas de las nuevas funciones que podríamos implementar en el programa es un registro de usuarios, es decir, si la aplicación está dirigida a empresas de automovilismo, que cada una tuviera un usuario propio

con el que acceder a la aplicación. Dicho esto, otra de las funcionalidades que podríamos crear es la de que, en vez de guardarse en un archivo local, se almacenara en una base de datos, es decir en la red. De esta manera, en caso de que el archivo se corrompiera o el dispositivo en el que se encuentre fallara, los datos estarían en la red y por lo tanto podríamos acceder a ellos desde otro dispositivo o dispositivos. También podríamos añadir nuevos servicios como por ejemplo Compra-venta de manera que pudiéramos llevar un seguimiento de los productos que compramos y vedemos y de los que disponemos. Con estas nuevas funcionalidades el programa podría adaptarse a nuevos modelos de negocio como la administración de stock de empresas de materias primas o incluso un restaurante.

Bibliografía

- <http://es.stackoverflow.com/>
- <https://msdn.microsoft.com/es-es/library/dd831853.aspx>
- <https://www.youtube.com/watch?v=8Fae2wRLAd0>
- <https://www.youtube.com/watch?v=QgSeDCmB-qQ&list=PLw8RQJQ8K1yQDqPyDRzt-h8YIBj96OwMP>