



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

SERVICIO WEB PARA LA GESTIÓN DEL CONTROL DE CALIDAD DE CARROCERÍAS

TRABAJO FIN DE GRADO

Grado en Ingeniería Informática

Autor: Sandra Moreno Olivares

Tutor académico: Antonio Martí Campoy

Tutor empresa: Francisco Monzó Gómez

Curso 2017-2018

Resumen

La empresa Ford España de Almussafes (Valencia) propuso actualizar el sistema de trabajo del Departamento de Control de Calidad de Carrocerías durante las prácticas de empresa del Grado.

Con el sistema actual, y así se ha venido haciendo durante cuarenta años, se efectúan controles de calidad en los que los operarios de cada punto de control indican sobre papel qué tipo de control de calidad se ha realizado, qué defectos se han encontrado en una determinada pieza de un vehículo y dónde están ubicados. Dado que, dependiendo de su localización, han podido ser causados por un robot determinado o haber sido originados por otro motivo, estas indicaciones permiten que, en caso de observarse reiteración del defecto, pueda repararse y corregirse la causa del defecto lo antes posible para evitar futuras pérdidas de producción y de tiempo.

En el proceso de producción de un vehículo, Ford realiza miles de pruebas de calidad y genera cantidades ingentes de documentos en papel que después deben digitalizarse y almacenarse, lo cual supone un gasto económico, de almacenamiento y de tiempo importantes.

Por estas razones, la empresa propuso desarrollar este proyecto, consistente en crear una aplicación web con la cual se puedan realizar controles de calidad eficientemente y de manera cómoda y asequible, así como con la que se alcance el objetivo primordial de reducir costes tanto temporales como económicos.

Principalmente, los lenguajes y tecnologías utilizados para el desarrollo de este proyecto han sido XAML, C#, .NET, SQL y Microsoft Access. Además, se ha empleado un proyecto de tipo *WPF Browser App* en el programa Visual Studio 2015 para crear un servicio web mediante el cual los operarios puedan efectuar controles de calidad desde una tableta o un ordenador con acciones como: seleccionar el tipo de test que van a llevar a cabo, determinar la pieza y modelo de vehículo objetos del test de acuerdo al plan de control de producción, así como recopilar y almacenar de forma dinámica y digital los resultados y observaciones que los operarios estimen oportuno efectuar al respecto.

Palabras clave: aplicación web, servicio web, control de calidad, operario, Ford, *WPF Browser App*, XAML, C#, SQL, Microsoft Access

Resum

L'empresa Ford Espanya de Almussafes (València) va proposar actualitzar el sistema de treball del Departament de Control de Qualitat de Carrosseries durant les pràctiques d'empresa del Grau.

Amb el sistema actual, i així s'ha vingut fent durant quaranta anys, s'efectuen controls de qualitat en què els operaris de cada punt de control indiquen sobre paper quin tipus de control de qualitat s'ha realitzat, quins defectes s'han trobat en una determinada peça d'un vehicle i on estan ubicats. Atés que, depenent de la seua localització, han pogut ser causats per un robot determinat o haver sigut originats per un altre motiu, estes indicacions permeten que, en cas d'observar-se reiteració del defecte, pugua reparar-se el més prompte possible per a evitar futures pèrdues de producció i de temps.

En el procés de producció d'un vehicle, Ford realitza milers de proves de qualitat i genera quantitats ingents de documents en paper que després han de digitalitzar-se i emmagatzemar-se, la qual cosa suposa una despesa econòmica, d'emmagatzemament i de temps importants.

Per aquestes raons, l'empresa va proposar desenvolupar aquest projecte, consistent a crear una aplicació web amb la qual es puguen realitzar controls de qualitat eficientment i de manera còmoda i assequible, així com amb la que s'aconsegueix l'objectiu primordial de reduir costos tant temporals com econòmics.

Principalment, els llenguatges i tecnologies utilitzats per al desenrotllament d'este projecte han sigut XAML, C#, .NET, SQL i Microsoft Access. A més, s'ha empleat un projecte de tipus *WPF Browser App* amb el programa Visual Studio 2015 per a crear un servei web mitjançant el qual els operaris puguen efectuar controls de qualitat des d'una tauleta o un ordinador amb accions com: seleccionar el tipus de test que van a dur a terme, determinar la peça i model de vehicle objectes del test d'acord al pla de control de producció, així com recopilar i emmagatzemar de forma dinàmica i digital els resultats i observacions que els operaris estimen oportú efectuar al respecte.

Paraules clau: aplicació web, servei web, control de qualitat, operari, Ford, *WPF Browser App*, XAML, C#, SQL, Microsoft Access

Abstract

Ford España company, located at Almussafes (Valencia) proposed an update of the workflow of the Quality Control Department of Bodywork plant during the intership.

In the current workflow, and also for 40 years ago, quality controls have been made by operators on paper. They write down which defects they found and where they found them. These data allow the supervisors to determine the reparation of one robot or work station in case of reiteration of the defect or damage in order to minimize future lost of production and time.

In the process of making a car, Ford executes thousands of quality tests and generates plenty of paper documents that should be digitalized and stored manually. This manual process implies an important waste of time and money.

For these reasons, the company proposed the development of the current project. It consists on a development of a web application that should be able to facilitate the quality process for the operators as long as reduce the detriment of time and money.

The languages and technologies mainly used for the development are: XAML, C#, .NET, SQL y Microsoft Access. Moreover, it has been developed using a project template for a WPF Browser App over Microsoft Visual Studio 2015 IDE. It had been chosen in order to create a web service through which an operator could easily make a quality control by tablet or PC with actions like: select the kind of test he want to make, select the piece and vehicle model over the test is going to being made, as long as collect and store dynamically the results and observations the operators consider pertinent.

Key words: web application, web service, quality control, operator, Ford, WPF Browser App, XAML, C#, SQL, Microsoft Access

Índice general

Índice general	VII
Índice de figuras	IX
Índice de tablas	XI
Glosario	XIII
Agradecimientos	XVII
<hr/>	
1 Introducción	1
1.1 Propuesta de Solución	2
1.2 Objetivos	3
2 Plan de trabajo	5
3 Especificaciones de requisitos	7
3.1 Introducción	7
3.1.1 Propósito	7
3.1.2 Ámbito del sistema	7
3.1.3 Referencias	8
3.2 Descripción general	8
3.2.1 Perspectiva del producto	8
3.2.2 Funciones del producto	9
3.2.3 Características de los usuarios	9
3.2.4 Restricciones	10
3.2.5 Suposiciones y dependencias	10
3.2.6 Requerimientos futuros	10
3.3 Requisitos específicos	11
3.3.1 Interfaces externas	11
3.3.2 Funciones	12
3.3.3 Requisitos de rendimiento	20
3.3.4 Atributos del sistema	20
4 Casos de uso	21
4.1 Diagrama casos de uso	21
4.2 Descripción casos de uso	22
5 Diseño	29
5.1 Arquitectura de tres capas	29
5.1.1 Capa de presentación	29
5.1.2 Capa de negocio	30
5.1.3 Capa de datos	30
5.2 Base de datos relacional	30
6 Implementación	35
6.1 Tecnologías y herramientas	35
6.1.1 XAML	35
6.1.2 C# y .NET	35
6.1.3 SQL	36

6.1.4	Visual Studio 2015	36
6.1.5	Internet Explorer	36
6.1.6	Microsoft Access	36
6.1.7	LaTeX	36
6.1.8	IIS	37
6.1.9	GIMP	37
6.1.10	StarUML	37
6.2	Estructura del proyecto	37
6.2.1	Capa de presentación	39
6.2.2	Capa de negocio	40
6.2.3	Capa de datos	41
7	Evaluación y pruebas	43
7.1	Prueba de validación de XAML	43
7.2	Prueba de uso en el navegador web	44
7.2.1	Inicio de sesión	45
7.2.2	Filtro principal	45
7.2.3	Parte de control de calidad vacío	46
7.2.4	Parte de control de calidad con datos	47
7.3	Prueba de distintas resoluciones	47
8	Conclusiones	51
	Bibliografía	53

Índice de figuras

1	Parte de control de calidad	XIV
2	Spot Pattern	XV
3	Etapa de carrocerías	1
4	Proceso del control de calidad	2
5	Plan de trabajo	6
6	Página de inicio	11
7	Control de calidad	12
8	Diagrama de casos de uso	22
9	Arquitectura de tres capas	29
10	Diagrama UML de la base de datos	31
11	Estructura del proyecto	38
12	Control apariencia superficial de la puerta derecha del Mondeo de techo normal	38
13	Control puntos de soldadura de la puerta derecha del Mondeo de techo normal	39
14	Control selladores de la puerta derecha del Mondeo de techo normal	39
15	Archivos de la capa de presentación	40
16	Archivos de la capa de negocio	41
17	Archivos de la capa acceso a datos	41
18	Validación de la página de inicio	43
19	Validación de la página del control de calidad	44
20	Validación del filtro principal	44
21	Validación del parte de control de calidad	44
22	Interfaz página de inicio	45
23	Interfaz filtro principal	46
24	Interfaz parte de control de calidad vacío	46
25	Interfaz parte de control de calidad con datos	47
26	Interfaz resolución 1366x768	48
27	Interfaz resolución 1280x768	48
28	Interfaz resolución 800x600	49

Índice de tablas

1	Tablas de requisitos	12
2	Casos de uso	23

Glosario

- **Base de datos:** [1] conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.
- **Body:** término empleado por los técnicos de Ford para clasificar en diversas secciones una o varias plantas encargadas de una etapa del proceso de producción. Este proyecto se centra en la planta de carrocerías de la factoría Ford España Almussafes. La planta de carrocerías se divide en tres secciones, que además se encuentran físicamente divididas en edificios separados. Dichas secciones se denominan *body 1*, *body 2* y *body 3*.
- **Canvas:** contenedor, lienzo o panel de control de la interfaz gráfica que permite posicionar elementos hijos en posiciones relativas con coordenadas explícitas. Este panel permite crear la ilusión de que en una imagen aparezcan formas como círculos o cuadrados cuando un operario selecciona un área o elemento de dicha imagen.
- **Checkbox:** se utiliza generalmente para listar opciones mediante casilla de verificación y el usuario puede escoger entre ellas
- **C#:** [2] lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET, aprobado posteriormente como un estándar por la ECMA (ECMA-334) e ISO (ISO/IEC 23270).
- **ComboBox:** elemento de control de la interfaz gráfica que muestra un cuadro de texto con información desplegable.
- **Grupo de ingeniería:** término utilizado por el personal de Ford para referirse a un área de trabajo dentro de las distintas secciones o *bodies* de una planta. Un grupo de ingeniería puede contener varias estaciones de trabajo que cuenten o no con robots.
- **LDAP:** siglas de *Lightweight Directory Access Protocol* [3], constituye un protocolo a nivel de aplicación que permite acceder a un servicio de directorio distribuido y ordenado para buscar diversa información en un entorno de red. Normalmente, almacena la información de autenticación (usuario y contraseña) y es utilizado para autenticarse.
- **OLE DB:** siglas de *Object Linking and Embedding for Databases*, tecnología para acceder a diferentes fuentes de información, o bases de datos, de manera uniforme desarrollada por Microsoft.
- **Operario de línea:** término usado en la factoría Ford España de Almussafes para identificar a los empleados sin cargo de mando que desempeñan su labor en la fábrica de la factoría; no están en despachos sino en estaciones de trabajo en las que se hallan los vehículos o piezas de estos en el proceso de fabricación.

- Parte de control de calidad:** documento usado por Ford para listar los defectos hallados en una pieza en concreto y la información relativa a estos defectos, como pueden ser: nombre del área o elemento anómalo, estación y robot que se encargan de su fabricación, etc. La figura 1 es un ejemplo de uno de estos partes.

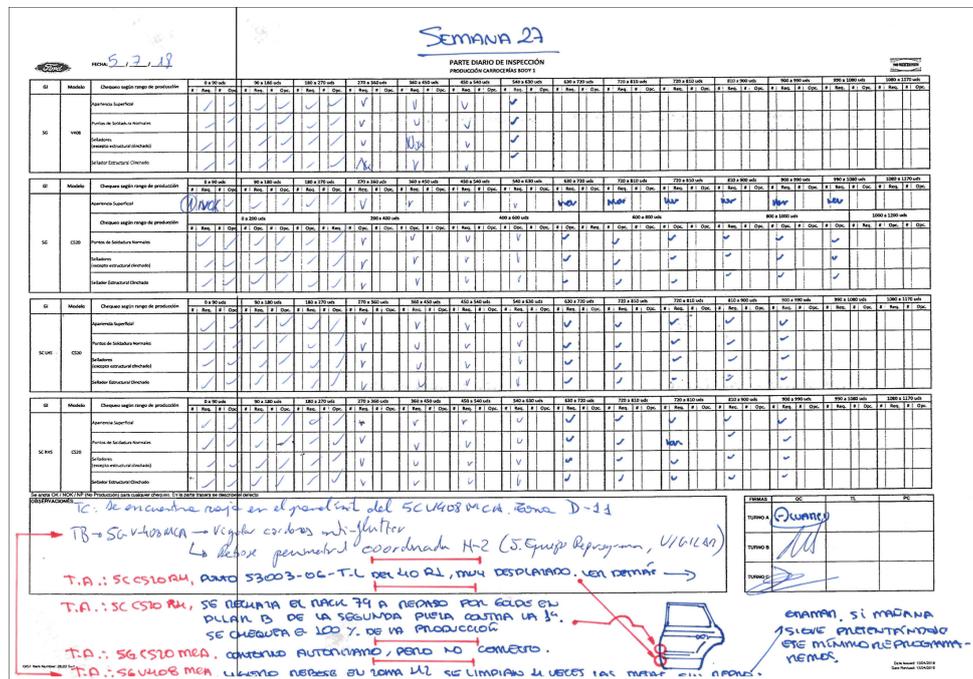


Figura 1: Parte de control de calidad

- Process coach:** término empleado en Ford para referirse al encargado de los operarios de línea.
- Navegador web:** programa, software o aplicación que permite el acceso a la web, se utiliza para interpretar la información de distintos tipos de archivos y sitios web a fin de que estos puedan ser visualizados.
- Sistema de gestión de bases de datos:** abreviado SGBD, es un conjunto de programas que permiten el almacenamiento, modificación y extracción de la información en una base de datos y proporcionan herramientas para añadir, borrar, modificar y analizar los datos.
- Software:** conjunto de rutinas y programas que permiten a los ordenadores realizar tareas específicas.
- Spot Pattern:** documento usado por Ford en el que se refleja el plan de control de una pieza determinada. Este documento contiene los distintos controles por los que pasa una pieza, el número de piezas del mismo tipo que se examinará en cada tipo de control, el material que deberá usarse y la frecuencia de estos exámenes. La figura 2 muestra un ejemplo de plan de control o *Spot Pattern*.

PLAN DE CONTROL DE PRODUCCIÓN (PCP)

Planta	Carrocerías	Nº Galga :	██████	VOME Body Construction Eng.:	██████
Modelo	Mondeo	Nº Área QC:	██████	VOME Weld Eng.Spv:	██████
Grupo Ing.	SB			Plant GRC:	██████
Nombre Pieza	Conjunto puerta delantera LH/RH			Plant Quality Manager:	██████
Proceso	Línea Automática				

CONTROLES RUTINARIOS		Apariencia Superficial	Puntos de Soldadura Normales	Puntos de Soldadura de Control	Selladores (excepto estructural clinchado)	Sellador Estructural Clinchado
Métodos	Estación de Control	██████	██████	██████	██████	██████
	Especificación / Procedimiento	██████	██████	██████	██████	██████
	Instrumentos de Control	Táctil, Visual y Aceite-Piedra	Martillo y Cíncel o Ultrasonidos	Martillo y Cíncel o Ultrasonidos	Visual	Visual / Dispositivo pelado
	Frecuencia	1 / 90uds	1 / 90uds	1 / 90uds	1 / 90uds (**)	1 / 90uds (**)
	Método de Análisis	Parte Diario de Inspección	Parte Diario de Inspección	Parte Diario de Inspección	Parte Diario de Inspección	Parte Diario de Inspección
	Hoja de Instrucción	██████	██████	██████	██████	██████
Plan Reacción:	██████	██████	██████	██████	██████	

CARACTERÍSTICAS DIMENSIONALES - CC (CRÍTICAS) / SIGNIFICATIVAS (SC) / ALTO IMPACTO (HI)										
Nº.	DENOMINACIÓN PIEZA	Características	Método							
	Nombre del proceso/ Descripción de la Operación		Tipo Defecto	Estación de Control	Especificación / Procedimiento	Instrumentos de Control	Persona Encargada	Frecuencia	Método de Análisis	Hoja de Instrucción / Plan Reacción:
1	CD391 SB LHRH	Fuera de especificación (HI) (***)	██████	-	██████	██████	██████	1 / día (*)	██████	██████

Figura 2: Spot Pattern

- **SharePoint:** consiste en una plataforma utilizada en sitios de web que permite acceder a espacios de trabajo compartidos, almacenes de información y documentos.
- **SQL:** [4] siglas de *Structured Query Language*, es un lenguaje de programación planteado para administrar sistemas de gestión de bases de datos relacionales. Una de sus características principales es el manejo del álgebra y el cálculo relacional para realizar consultas con el fin de recuperar, de forma sencilla, información de bases de datos, así como para realizar cambios en ellas.
- **TabControl:** elemento de control de la interfaz gráfica que muestra múltiples fichas o pestañas, como las etiquetas de un archivador. Estas fichas o pestañas, se denominan *TabItems* y pueden contener imágenes y otros controles o elementos gráficos de la interfaz.
- **Team leader:** término utilizado en Ford para referirse al jefe de equipo de la línea.
- **WPF:** siglas de *Windows Presentation Foundation*, es una tecnología de Microsoft. Permite el desarrollo de interfaces tomando características de aplicaciones Windows y de aplicaciones web.
- **XAML:** siglas de *eXtensible Application Markup Language*, constituye un lenguaje declarativo basado en XML que permite desarrollar aplicaciones web y está optimizado para describir gráficamente interfaces de usuarios.

Agradecimientos

A mi tutor en Ford, don Francisco Monzó Gómez por guiarme durante este proyecto y a mi tutor de la universidad, don Antonio Martí Campoy por ayudarme con la redacción de esta memoria. Me siento muy afortunada de haberles tenido como tutores.

A David García Molino, José Buzón García y Daria Corina Bec por motivarme, ayudarme y estar conmigo en los buenos momentos y en los menos buenos.

Y a mis padres, por haberme apoyado para poder llegar hasta aquí: espero que estéis orgullosos.

CAPÍTULO 1

Introducción

Se ha realizado una estancia en prácticas de empresa en la multinacional Ford España Almussafes (Valencia) en el Departamento de Control de Calidad en Carrocerías. Este departamento necesita actualizar la forma en la que trabajan sus operarios, ya que todas las inspecciones o partes de calidad se realizan en papel. En estos partes, se hacen constar los defectos encontrados en la pieza correspondiente y la ubicación de cada uno de ellos.

Los controles de calidad se efectúan en cada una de las fases del proceso de producción, desde que se obtienen las distintas piezas que conforman un vehículo al prensarse la chapa, hasta que se completa su proceso de fabricación. Además, se realizan distintos tipos de controles de calidad: unos, sencillos, en los que observan con detalle distintas piezas de la carrocería o elementos del vehículo en busca de imperfecciones y otros, más generales, cuando ha finalizado una etapa de su proceso de producción.

Este proyecto se centrará en los controles de calidad que se realizan en la etapa de carrocerías, cuando se ensamblan las piezas del vehículo que acaban de ser prensadas. Esto incluye tanto el control de calidad una vez ensamblado por completo el vehículo como los controles de calidad por los que pasa cada de pieza del vehículo (laterales, puertas, suelos, techos, capos, etc). En la figura 3 puede observarse el lateral derecho de uno de los modelos.



Figura 3: Etapa de carrocerías

Actualmente, para llevar a cabo un control de calidad, los operarios efectúan las siguientes acciones: primero, acceden al *sharedPoint*, página web creada por los ingenieros de procesos que contiene los partes de cada pieza y de cada modelo. Seguidamente, buscan el que se requiere para su grupo de ingeniería y lo imprimen. Una vez impresos los partes necesarios, anotan en ellos a mano todos los defectos u observaciones que encuentran en las piezas examinadas. Además, dichos operarios proceden a apartar la pieza defectuosa para su posterior reparación o eliminación del proceso de producción. Cuando los controles han sido completados, los operarios localizan a su *process coach* (o encargado) para que dé el visto bueno al informe. Tras obtener la firma del *process coach*, es necesaria la validación y firma del *team leader*, el jefe de equipo de la línea, al que los operarios han de localizar personalmente. Cuando finalmente el operario recupera el informe de calidad, procede a archivarlo y a registrar los datos para posteriores controles, análisis, gráficos, auditorías, etc. En caso de que no se pueda localizar al *process coach* o al *team leader*, el parte se tramitará en la siguiente jornada laboral. Este procedimiento queda reflejado en la figura 4:

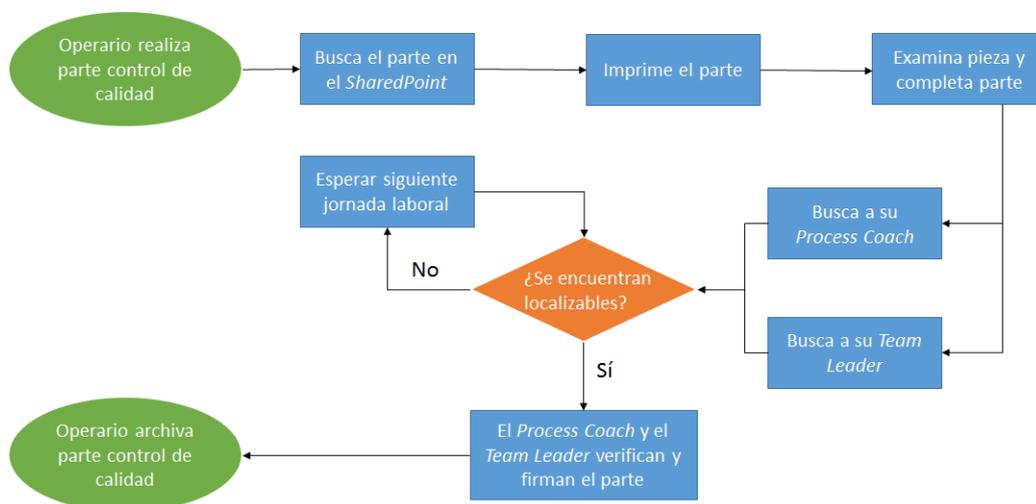


Figura 4: Proceso del control de calidad

1.1 Propuesta de Solución

Como se observa, este proceso es altamente ineficiente y comporta una considerable pérdida económica y temporal para la empresa. Con el objetivo de llegar a solventar en un futuro este problema, Ford propone realizar una aplicación web de gestión informatizada de partes de control de calidad de una pieza concreta de un modelo de vehículo en particular. El propósito de este proyecto es comprobar si sería viable aplicar este sistema de gestión a todo el sistema de control de calidad de carrocerías en la planta de Almusafes de Ford España dado que actualmente producen cinco modelos de vehículo, en los que cada vehículo cuenta con un gran número de piezas.

Sin embargo, el conjunto de datos para la gestión de controles de calidad con el que Ford cuenta en este momento se almacena en una base de datos no relacional denormalizada implementada en Microsoft Access. Por consiguiente, este almacenamiento presenta abundantes redundancias en áreas sensibles, carece de algunos datos importantes y escasea de la necesaria integridad que caracteriza a las bases de datos relacionales. Como consecuencia de este hecho y a fin de poder implantar adecuadamente la gestión piloto en el proceso de los controles de calidad, se requiere diseñar adicionalmente una base

de datos que optimice los tiempos de consulta y que dé soporte al sistema de gestión de partes.

1.2 Objetivos

El objetivo de este trabajo final de grado (TFG) es, por lo tanto, diseñar un programa informático de gestión de partes de control de calidad delimitado a un área determinada del proceso de producción de un vehículo con el propósito de verificar la viabilidad de su implantación en un futuro a todo el sistema de control de calidad. Por decisión de Ford, se va a emplear para ello en el programa Visual Studio 2015 un proyecto de tipo *WPF Browser App* que ofrecerá un servicio web soportado tanto por ordenador como por tableta. Previo al desarrollo de este proyecto, será necesario el diseño de una base de datos relacional que dé soporte a esta aplicación web de control de calidad. Por restricciones impuestas por Ford será implementada en Microsoft Access.

Este objetivo puede desglosarse en los siguientes:

- Analizar los distintos recursos y requerimientos necesarios para llevar a cabo el proyecto.
- Crear una base de datos relacional normalizada que pueda sustentar el programa que se va a desarrollar. Las tablas de esta base de datos cumplirán con la tercera forma normal.
- Diseñar la arquitectura e implementar el servicio web de gestión de partes de control de calidad.

Cabe precisar que este proyecto se implementará para dar funcionalidad completa al sistema de gestión del control de calidad de una puerta derecha del modelo de vehículo Mondeo con techo normal, para el grupo de ingeniería 5B en el área *body 3* de la planta de carrocerías de Ford España Almussafes (Valencia).

CAPÍTULO 2

Plan de trabajo

En este apartado, se va a definir el plan de trabajo consistente en establecer las diferentes etapas del proyecto y el tiempo que se dedicará a cada una.

En primer lugar, el tiempo definido para la identificación de requisitos y funcionalidades con las que contará este proyecto será de dos semanas, incluyendo las revisiones y validaciones por parte de Ford. Una vez todos los requerimientos y el comportamiento de la aplicación web hayan sido detallados y aprobados por la empresa, se dedicará una semana al planteamiento de los casos de uso y al diseño de su respectivo diagrama.

Posteriormente, se requerirán tres semanas para el diseño completo del proyecto; esta es una de sus etapas más importantes que, por lo demás, también exigirá numerosas revisiones y aprobaciones por parte de la empresa. En la etapa de diseño se estudiará cómo utilizar de forma óptima las tecnologías requeridas por Ford y se determinará la arquitectura más adecuada para estructurar este servicio web.

Una vez completado y aprobado el diseño por la empresa, se procederá a su desarrollo y al diseño e implementación de la base de datos sobre la que se sustentará. El desarrollo e implementación constituyen la etapa en la que debe invertirse la mayor cantidad de tiempo durante el proyecto. Por otra parte, han de tenerse en cuenta los diferentes tests y pruebas que se irán realizando conforme se vaya implementando el programa. Por tanto, la etapa de pruebas empezará al mismo tiempo que la etapa de implementación; sin embargo, las pruebas finalizarán después de la finalización del período de implementación debido a que se deberá comprobar el correcto funcionamiento de la aplicación una vez haya sido finalizado su desarrollo. Por estas razones, se requerirán no menos de seis semanas para su implementación y alrededor de siete para las pruebas finales.

Tal como se observa en la figura 5, el proyecto tiene una duración de trece semanas, lo cual representa tres meses de trabajo.

Tareas	Inicio	Fin	Duración	Semanas													
				1	2	3	4	5	6	7	8	9	10	11	12	13	
Requisitos y funcionalidades	23/04/2018	06/05/2018	14 días	■	■												
Casos de uso	07/05/2018	13/05/2018	7 días			■											
Diseño	14/05/2018	03/06/2018	14 días				■	■	■								
Implementación	04/06/2018	13/07/2018	30 días								■	■	■	■	■	■	
Pruebas	04/06/2018	20/07/2018	35 días								■	■	■	■	■	■	■

Figura 5: Plan de trabajo

CAPÍTULO 3

Especificaciones de requisitos

En este apartado se van a definir los requerimientos con los que cumplirá esta aplicación web. En concreto, se va a aplicar el estándar IEEE 830/1998 [5] para la Especificación de Requisitos Software (ERS).

3.1 Introducción

El presente apartado de especificaciones de requisitos se estructurará siguiendo el estándar IEEE 830/1998.

3.1.1. Propósito

Se explicará cada funcionalidad que ofrecerá este servicio web de forma estructurada y concreta. Asimismo, también se indicarán las funcionalidades que quedarán fuera del alcance del programa. De este modo, se planteará de forma que sea entendible para todo aquel que se interese por el proyecto.

3.1.2. Ámbito del sistema

El objetivo de este proyecto es desarrollar un programa que proporcionará un servicio web para la gestión de partes de control de calidad. Por restricciones de Ford, se implementará para el navegador Internet Explorer.

Un operario podrá iniciar sesión en esta aplicación web con su cuenta suministrada por Ford y su contraseña, sin necesidad de una específica. Inmediatamente después, la aplicación mostrará una ventana llamada Filtro Principal con *comboBoxes* (cuadros con elementos desplegable), que permitirá al operario seleccionar: el área en el que va a realizar el control de calidad, en este caso, *carrocerías*; el modelo del vehículo que va a examinar, el *Mondeo*; la pieza que va a inspeccionar, una *puerta derecha del modelo de techo normal*; y, finalmente, el grupo de ingeniería al que pertenece dicho usuario, *5B*.

Con los datos seleccionados, la aplicación cargará diferentes pestañas con los distintos partes de control por los que pasa esa pieza en concreto en ese punto de la producción según su *spot pattern* (plan de control). Dependiendo del parte de control que el usuario elija, se mostrarán unos datos u otros si bien el operario tan solo podrá ver y seleccionar la información relativa a los defectos que pertenezcan a su grupo de ingeniería, sin necesidad de que conozca los detalles de ese defecto, p.e. nombre de la zona, estación y robot

que la gestiona, etc.

Por otro lado, el parte que cargará la aplicación contendrá la imagen ampliable de la pieza con los datos que se estén examinando y el operario únicamente tendrá que indicar la zona o elemento que sufre una anomalía. Una vez un área o elemento haya sido seleccionado, se iluminará en la imagen cargada; a continuación, la aplicación abrirá un panel que identificará el nombre del área o elemento marcado y que contendrá un *checkbox* o casillas de verificación con los posibles defectos que pueden presentarse en ese parte. En caso de que el usuario observe un defecto no contemplado entre los posibles, podrá seleccionar una casilla especial que mostrará un cuadro de texto en el que podrá explicar el error hallado.

Así pues, el principal beneficio de esta aplicación es permitir al operario realizar *in situ* desde su tableta u ordenador cada uno de los controles de calidad de la pieza que examine, ahorrando un valioso tiempo que mejora la productividad y calidad del proceso de control de calidad. Estos partes serán guardados en una base de datos relacional escalable diseñada previamente para sustentar el programa.

Si Ford considerase oportuno en un futuro desarrollar por completo este servicio web piloto e implantarlo en su web privada o intranet de Ford Valencia: Primero debería añadir al programa algunos cambios, como distinguir distintos tipos de usuario según el rol que representen. Además, el programa debería permitir: a los operarios de línea completar los controles de calidad; a los encargados o jefes consultar los informes, firmarlos digitalmente y validarlos desde donde se encuentren; facilitar las auditorías y servir de base para nuevos programas que tomarán los datos de los informes para crear gráficas, estadísticas etc.

3.1.3. Referencias

IEEE Recommended Practice for Software Requirements Specification. ANSI/IEEE 830/1998.

3.2 Descripción general

En este apartado se describirán los factores y los elementos contextuales que pueden condicionar los requisitos de este proyecto, explicados con detalle en el punto siguiente.

3.2.1. Perspectiva del producto

La aplicación web funcionará en el entorno de un navegador web, dado que se planteó desde el primer momento la necesidad de que pudiera ser ejecutado tanto en tabletas como en ordenadores de sobremesa o portátiles. Además, a requisito de Ford, la aplicación deberá visualizarse correctamente en el navegador Internet Explorer puesto que la empresa lo emplea como navegador por defecto en todas sus instalaciones. Por otro lado, este programa se desarrollará sobre el sistema operativo Windows 10, pero también funcionará en otros sistemas operativos siempre y cuando cuenten con el navegador web Internet Explorer y soporten el elemento *canvas* de XAML. El elemento *canvas* es utilizado cuando el operario selecciona un área o elemento y éste se ilumina en la imagen cargada

en el parte del control de calidad.

3.2.2. Funciones del producto

En esta sección se van a definir a grandes rasgos las funcionalidades que deberá cumplir la aplicación web.

Al tratarse de una aplicación limitada en sus funcionalidades y acotada a una pieza concreta, se ha diseñado para que dé acceso a un único tipo de usuario, el operario, que será quien acceda al servicio web y complete los partes de control. El proyecto no contendrá más roles de usuario a petición de Ford; sin embargo, la base de datos ha sido diseñada de tal modo que podría dar soporte al proyecto completo si en un futuro la empresa decidiera implantarlo.

A continuación, se expondrán en líneas generales las funcionalidades que proporcionará el servicio web que ofrece la aplicación diseñada:

- Inicio de sesión: cualquier usuario con cuenta Ford podrá identificarse e iniciar sesión en el sistema.
- Seleccionar, modificar y cancelar datos del filtro principal: el usuario podrá seleccionar los datos que contiene la ventana del filtro principal. Deberá completar todos los datos del filtro principal siguiendo un orden concreto. Asimismo, el usuario podrá modificar las opciones elegidas o deseleccionar todos los datos escogidos.
- Mostrar y ocultar partes de control de calidad: a partir de las opciones seleccionadas en la ventana del filtro principal, se mostrarán diferentes pestañas con los distintos partes de control seleccionables según el *spot pattern* o plan de control de la pieza escogida. Si los datos del filtro principal son cancelados, dichas pestañas se ocultarán. El usuario podrá seleccionar una de las pestañas y la aplicación cargará los datos del parte escogido.
- Seleccionar, modificar y eliminar datos del parte de control escogido: el usuario podrá seleccionar las áreas o elementos en la imagen cargada del parte seleccionado que presenten anomalías. En cada área o elemento escogido, se podrán indicar los defectos encontrados y también modificarse. Además, el usuario podrá eliminar áreas o elementos seleccionados por error.
- Guardar y cancelar partes de control: el usuario podrá guardar cada parte completado, con lo que será almacenado en la base de datos relacional que soporta al programa. No obstante, el operario también podrá cancelar el parte.
- Cerrar sesión: el usuario contará con la opción de salir de la aplicación.

3.2.3. Características de los usuarios

Esta aplicación web se desarrollará para el ámbito empresarial. Los usuarios que utilizarán este servicio web serán operarios de línea, la mayoría de los cuales cuentan con formación básica, lo cual significa: por un lado, formación académica a nivel de estudios obligatorios como la Educación Secundaria Obligatoria (ESO) u equivalentes, ciclos medios o superiores; por otro lado, formación técnica adquirida en la empresa, ya sea por

haber cursado en ella ciclos de formación o mediante cursos ofertados. Además, existe una gran variedad de edades y experiencias entre los potenciales usuarios de la aplicación, tanto por lo que respecta al propio trabajo, es decir, realizar controles de calidad, como a la experiencia informática. Esta es la razón por la que esta aplicación deberá tener una interfaz sencilla y sin ambigüedad, con el fin de facilitar a los usuarios su manejo para mejorar y optimizar el proceso actual de gestión de partes de control de calidad.

3.2.4. Restricciones

Por exigencias de Ford, el diseño e implementación de este servicio web deberá cumplir las siguientes descripciones: se desarrollará con el programa Visual Studio 2015 usando el proyecto de tipo *WPF Browser App* para que funcione en el navegador web Internet Explorer. Además, tanto la página inicial con el formulario de autenticación como el código del programa y sus comentarios aparecerán en inglés. La interfaz gráfica se implementará en el lenguaje XAML, mientras que el comportamiento del programa se desarrollará usando el lenguaje C#. La base de datos relacional se diseñará en el sistema de gestión de base de datos Microsoft Access y la conexión con dicha base de datos se efectuará realizando consultas SQL mediante el conector OLE DB. Además, cuando la aplicación compruebe las identificaciones de los usuarios necesitará conectarse mediante un servidor LDAP a *FORDEU1*, que es el nombre del servidor de Ford España Almussafes.

El criterio de diseño de la interfaz deberá cumplir con la paleta de colores de Ford y contener el logo de la empresa en cada página de este servicio web. Por otro lado, el proyecto permitirá la concurrencia de varios usuarios ejecutando simultáneamente la aplicación. Por último, el servicio manejará las posibles incidencias que puedan producirse debido a errores humanos o problemas técnicos previsibles, por ejemplo, errores en la conexión con la base de datos, etc.

Por indicación de Ford y como ya se ha mencionado anteriormente, este proyecto se limitará al funcionamiento de la gestión de partes de control de calidad correspondientes a una puerta derecha del modelo de vehículo Mondeo con techo normal, para el grupo de ingeniería 5B en el área *body 3* de la planta de carrocerías de Ford España Almussafes (Valencia).

3.2.5. Suposiciones y dependencias

El proyecto se desarrollará para el sistema operativo Windows 10 y en el entorno de navegador Internet Explorer. Si se cambiase el sistema operativo no afectaría al funcionamiento de la aplicación siempre y cuando el nuevo sistema contara con el navegador web Internet Explorer. Sin embargo, sería necesario revisar los requisitos y realizar algunas modificaciones en el código en caso de que Ford decidiera cambiar de navegador web oficial, o añadir nuevos roles de usuario.

3.2.6. Requerimientos futuros

En un futuro, si Ford decidiese implantar esta aplicación web y ampliar su desarrollo a la totalidad de las piezas que requieren control de calidad, se requeriría implementar nuevos roles, tales como: administrador; *process coach* o encargado; *team leader* o jefe de equipo de la línea, entre otros. Además, sería necesario implementar la creación de los

informes de control de calidad y las opciones para consultarlos y firmarlos, con lo que quedaría completado el proceso de control de calidad informatizado.

3.3 Requisitos específicos

3.3.1. Interfaces externas

Se requiere que la interfaz del servicio web sea sencilla e intuitiva. Con este objetivo, se ha decidido separar la aplicación en distintas páginas.

La página inicial a la que los usuarios accederán contendrá el formulario de identificación e inicio de sesión, tal como puede observarse en la figura 6.

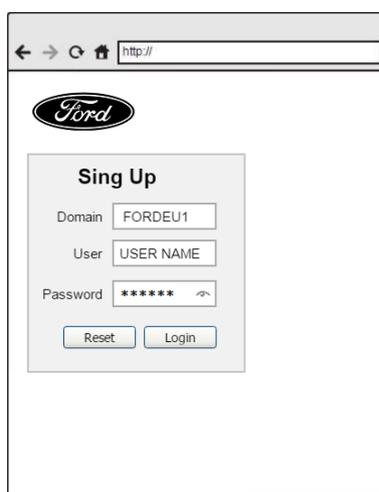
A screenshot of a web browser window. The address bar shows 'http://'. The page features the Ford logo at the top. Below the logo is a 'Sing Up' form. The form has three input fields: 'Domain' with the value 'FORDEU1', 'User' with the value 'USER NAME', and 'Password' with the value '*****'. There are two buttons at the bottom of the form: 'Reset' and 'Login'.

Figura 6: Página de inicio

La siguiente página a la que el usuario podrá acceder se dividirá en tres zonas gracias al uso de *frames* o marcos. Dichas zonas son las siguientes:

- **Lateral izquierdo:** contendrá el logo de la empresa y el filtro principal en el que el usuario elegirá los datos requeridos para realizar el parte de control de calidad correspondiente.
- **Zona superior:** aparecerá un *TabControl* con distintas pestañas que corresponderán a los distintos tipos de control de calidad seleccionables a los que conducen los datos introducidos en el filtro principal.
- **Zona central:** en cada pestaña o *TabItem* del *TabControl* que contiene dicha zona central se mostrará el contenido del tipo de control seleccionado, es decir, aparecerá la información del parte de control de calidad que completará el usuario. Este *frame* o marco contendrá: la imagen de la pieza dependiendo del control seleccionado y los datos filtrados; los botones de aumentar o disminuir el zoom de la imagen, aceptar o cancelar el parte de control o los posibles defectos del elemento seleccionado. Por otro lado, dicho marco también mostrará el panel con los posibles defectos que pueda hallar el usuario cuando seleccione un elemento u área de la imagen. Asimismo, el *frame* del parte de control dispondrá también de un panel con bloque de texto para las observaciones que se mostrará cuando el usuario seleccione la casilla especial denominada "Otro".

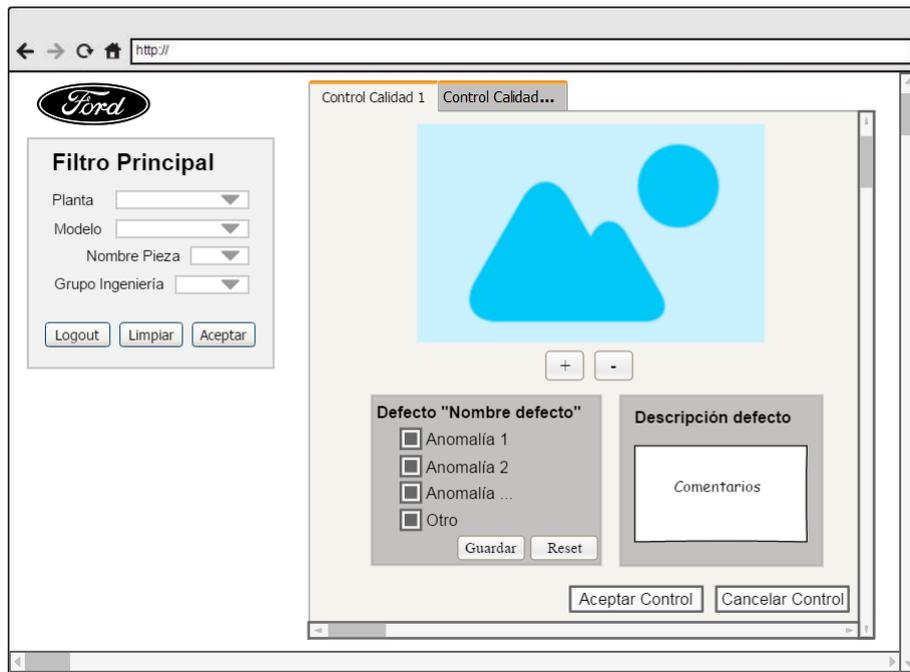


Figura 7: Control de calidad

3.3.2. Funciones

En este apartado se definirán las funcionalidades que cumplirá el proyecto para posibilitar su posterior implementación. Las funcionalidades se dividirán en diferentes listas para clarificar la explicación dependiendo de su objetivo final.

En primer lugar, se definirán aquellos requisitos que tienen como objetivo el funcionamiento de la página inicial con el formulario de identificación e inicio de sesión.

Tabla 1: Tablas de requisitos

Número	1
Requisito	Tomar usuario del sistema operativo
Prioridad	Alta
Descripción	La aplicación añadirá en mayúsculas el nombre del usuario registrado en el sistema operativo (Windows 10), en el campo de texto para el usuario de la aplicación.

Número	2
Requisito	Introducir usuario
Prioridad	Baja
Descripción	El operario podrá escribir su nombre de usuario en el campo de texto para la identificación del usuario. La aplicación internamente transformará a letras mayúsculas el usuario introducido.

Número	3
Requisito	Introducir contraseña
Prioridad	Alta
Descripción	El operario podrá escribir su contraseña en el campo de texto correspondiente.

Número	4
Requisito	Visualizar contraseña mediante múltiples entradas
Prioridad	Baja
Descripción	Si el usuario mantiene pulsada con el ratón o presiona con el dedo o bolígrafo de la tableta la imagen que contiene el bloque de texto de la contraseña, la aplicación mostrará los caracteres desprotegidos introducidos por el usuario, hasta que este deje de presionar la imagen.

Número	5
Requisito	Validar usuario y contraseña
Prioridad	Alta
Descripción	Cuando el usuario pulse el botón <i>Login</i> o presione la tecla de <i>Intro</i> , la aplicación se conectará a <i>FORDEU1</i> , nombre del servidor de Ford España Almussafes, mediante el protocolo LDAP, para comprobar si la autenticación es correcta. En caso de que la autenticación fallase, la aplicación mostrará un mensaje de error para que el usuario modifique los datos introducidos. Si por el contrario la autenticación es correcta, se mostrará la siguiente ventana de la aplicación.

Número	6
Requisito	Limpiar formulario de inicio de sesión
Prioridad	Baja
Descripción	Cuando el usuario pulse sobre el botón de <i>Reset</i> , la aplicación borrará los datos introducidos por el usuario y los mensajes de error que haya podido mostrar. Además, en el bloque de texto para introducir al usuario, volverá a aparecer el usuario registrado en la sesión activa del sistema operativo. Por último, el programa mostrará un mensaje para indicar al usuario que no ha sido autenticado aún.

Seguidamente, se definirán las funcionalidades que tienen como objetivo el funcionamiento de la ventana con el filtro principal.

Número	7
Requisito	Seleccionar planta
Prioridad	Alta
Descripción	Al cargar la página del filtro principal, la aplicación realizará una consulta SQL mediante el conector OLE DB a la base de datos relacional para obtener las distintas plantas seleccionables. El <i>comboBox</i> que pertenece a la planta vinculará su información desplegable con el resultado de la consulta a la base de datos. Finalmente, el usuario podrá elegir la planta en la que va a realizar el control de calidad, en el caso de este servicio web, únicamente mostrará la opción de <i>carrocerías</i> . Sin embargo, el programa y la base de datos están preparados para se puedan filtrar diferentes plantas.

Número	8
Requisito	Seleccionar modelo de vehículo
Prioridad	Alta
Descripción	Una vez el usuario haya escogido una planta, la aplicación consultará a la base de datos los distintos modelos de vehículo existentes en dicha planta. El <i>comboBox</i> perteneciente al modelo de vehículo vinculará su información desplegable con el resultado de la consulta SQL. Finalmente, el usuario podrá escoger entre varios modelos de vehículo el que corresponda al control que va a realizar. No obstante, como esta aplicación está delimitada, en la implementación sólo aparecerá la opción de <i>Mondeo</i> para las pruebas de este servicio web.

Número	9
Requisito	Seleccionar nombre de pieza
Prioridad	Alta
Descripción	Una vez el usuario haya escogido una planta y un modelo (en este orden en concreto), la aplicación consultará en la base de datos las distintas piezas que contiene dicho modelo de la planta marcada. El <i>comboBox</i> perteneciente al nombre de la pieza vinculará su información desplegable con el resultado de la consulta SQL. Finalmente, el usuario podrá escoger la pieza que va a examinar, en este caso, una <i>puerta derecha de techo normal</i> . Tanto la base de datos como el programa permitirán seleccionar entre diversas piezas, a pesar de que para las pruebas que se llevarán a término en este servicio web sólo debe aparecer la pieza mencionada anteriormente.

Número	10
Requisito	Seleccionar grupo de ingeniería
Prioridad	Alta
Descripción	Una vez el usuario haya escogido una planta y un modelo (en este orden en concreto), la aplicación consultará en la base de datos los grupos de ingeniería que trabajan con dicho modelo en la planta escogida. Por tanto, el grupo de ingeniería no depende de la pieza seleccionada para realizar la consulta. El <i>comboBox</i> perteneciente al grupo de ingeniería vinculará su información desplegable con el resultado de la consulta SQL. Finalmente, el usuario podrá escoger grupo de ingeniería al que pertenece, el <i>5B</i> en este servicio web.

Número	11
Requisito	Limpiar formulario de filtro principal
Prioridad	Baja
Descripción	Cuando el usuario pulse sobre el botón de <i>Limpiar</i> , la aplicación deseleccionará los datos introducidos por el usuario y ocultará posibles ventanas emergentes y mensajes de error que haya podido mostrar.

Número	12
Requisito	Validar datos del filtro principal
Prioridad	Alta
Descripción	Cuando el usuario pulse sobre el botón de <i>Aceptar</i> , la aplicación comprobará si todos los <i>comboBoxes</i> contienen un elemento seleccionado. En caso de que faltase algún dato, el programa mostraría un mensaje de error para que el usuario pudiera completar todos los datos. Sin embargo, si el operario ha introducido todos los datos, el programa efectuará una consulta SQL mediante el conector OLE DB a la base de datos para obtener los distintos controles de calidad por los que pasa la pieza escogida, del modelo seleccionado, en la planta marcada. Una vez obtiene la respuesta, muestra un <i>tabControl</i> o ventana dentro de la misma página del servicio web. Este <i>tabControl</i> está formado por distintas pestañas o <i>TabItems</i> , que representan cada uno de los controles seleccionables obtenidos en la consulta a la base de datos.

Número	13
Requisito	Cerrar Sesión
Prioridad	Media
Descripción	Cuando el usuario pulse sobre el botón de <i>Logout</i> , la aplicación comprobará si la información de la página en la que se encuentra el usuario tiene guardada toda la información. Si es el caso, el programa volverá a la página inicial de inicio de sesión. No obstante, si el operario se encontrara completando algún parte de control de calidad y dicho parte no hubiera sido previamente almacenado, el programa mostraría un mensaje de error y permanecería en la misma página.

Por último, se detallarán los requisitos con objetivo final la realización del parte de control de calidad.

Número	14
Requisito	Seleccionar parte de control de calidad a completar
Prioridad	Alta
Descripción	El control de calidad, o pestaña del <i>TabControl</i> mostrado, se abrirá cuando el usuario seleccione con el ratón, el dedo o el bolígrafo de la tableta, dicha pestaña.

Número	15
Requisito	Cargar los datos del parte de control de calidad a completar
Prioridad	Alta
Descripción	La aplicación mostrará los datos del parte de control de calidad escogido. Para ello, primero cargará la imagen correspondiente al tipo de control y pieza escogida por el usuario escalada a la mitad de su tamaño. Seguidamente, filtrará los elementos seleccionables en ella, por el grupo de ingeniería elegido, mediante una consulta SQL.

Número	16
Requisito	Aumentar zoom a la imagen mediante múltiples entradas
Prioridad	Alta
Descripción	La aplicación permitirá escalar la imagen mediante el movimiento con los dedos en la tableta, que ofrece el elemento <i>canvas</i> de XAML. Asimismo, si el usuario presiona sobre los botones de mayor o menor zoom, la imagen se escalará gradualmente dependiendo del botón seleccionado. El tamaño máximo de zoom es el tamaño original de la imagen y se establecerá un límite para minimizarla.

Número	17
Requisito	Seleccionar y destacar elemento o área de la imagen
Prioridad	Alta
Descripción	La aplicación permitirá al operario seleccionar un elemento o área de la imagen.

Número	18
Requisito	Destacar elemento o área seleccionado en la imagen
Prioridad	Alta
Descripción	El elemento o área de la imagen seleccionado se iluminará tras comprobar si existe en la lista de los elementos seleccionables según el grupo de ingeniería creada al cargar el parte. En caso de que dicho área o elemento pertenezca a unas coordenadas seleccionables, el elemento se iluminará.

Número	19
Requisito	Mostrar posibles defectos en un elemento o área de la imagen
Prioridad	Alta
Descripción	La aplicación abrirá un panel en el que estará identificado el área o elemento marcado y los posibles defectos que pueda presentar según el parte de control de calidad escogido.

Número	20
Requisito	Seleccionar defectos de un elemento o área de la imagen
Prioridad	Alta
Descripción	La aplicación permitirá al usuario seleccionar mediante casillas de verificación las anomalías que encuentre mientras realiza la inspección de la pieza.

Número	21
Requisito	Introducir observación de un elemento o área de la imagen
Prioridad	Alta
Descripción	Si el defecto hallado por el operario no se encontrara entre las posibles opciones mostradas, el operario podría escribir sus observaciones en un bloque de texto que se abriría al seleccionar una casilla de verificación especial para este caso, denominada "Otro".

Número	22
Requisito	Listar defectos u observaciones de un elemento o área de la imagen
Prioridad	Alta
Descripción	La aplicación listará automáticamente los distintos defectos o el comentario que haya introducido el operario identificados por el elemento u área seleccionado.

Número	23
Requisito	Guardar elemento o área seleccionado y su información
Prioridad	Alta
Descripción	El operario podrá guardar los defectos y/o comentarios del elemento o área seleccionado.

Número	24
Requisito	Comprobar si el usuario ha guardado los datos del elemento seleccionado
Prioridad	Alta
Descripción	La aplicación comprobará si antes de escoger un nuevo elemento o área los cambios han sido guardados. En caso de no ser así, mostrará un mensaje de error hasta que el usuario guarde los datos.

Número	25
Requisito	Eliminar los defectos introducidos en el elemento seleccionado
Prioridad	Baja
Descripción	El operario podrá deseleccionar todos los defectos marcados y en caso de haber introducido algún comentario, este será eliminado también.

Número	26
Requisito	Listar elementos o áreas seleccionados de la imagen
Prioridad	Alta
Descripción	La aplicación listará por orden de selección los elementos o áreas que presentan anomalías escogidos por el usuario.

Número	27
Requisito	Eliminar elementos o áreas seleccionados de la imagen
Prioridad	Media
Descripción	El usuario podrá eliminar elementos o áreas seleccionados en la imagen guardados previamente. Por tanto, también se borrarán los defectos u observaciones que haya introducido en dicho elemento o área.

Número	28
Requisito	Guardar parte de control de calidad
Prioridad	Alta
Descripción	El usuario podrá guardar el parte de control de calidad realizado. La aplicación insertará los datos de los distintos elementos seleccionados en la base de datos mediante una consulta SQL.

Número	29
Requisito	Cancelar parte de control de calidad
Prioridad	Alta
Descripción	El usuario podrá cancelar el parte de control de calidad que completaba. Por tanto, la aplicación borrará todos los elementos seleccionados en la imagen y sus respectivos datos.

Número	30
Requisito	Adaptar el contenido de las páginas a la resolución de pantalla
Prioridad	Alta
Descripción	La aplicación se adaptará a las distintas resoluciones de pantalla que puedan tener los distintos aparatos electrónicos que la carguen.

Número	31
Requisito	Realizar <i>scroll</i> o desplazamiento en la página
Prioridad	Alta
Descripción	La aplicación permitirá realizar <i>scroll</i> o desplazamiento vertical y horizontal, tanto táctil como con el ratón. Si las barras de <i>scroll</i> o deslizamiento no son necesarias porque las páginas se muestran completas, dichas barras se ocultarán automáticamente.

3.3.3. Requisitos de rendimiento

El rendimiento final del proyecto dependerá en gran medida de la eficiencia del servidor web que aloje la base de datos y la aplicación.

Actualmente, la carga que soporta el servidor impuesta por el programa no es muy elevada, aunque en cualquier caso la base de datos ha sido diseñada para dar soporte a una eventual implantación del sistema a todo el proceso de producción potencialmente sometible a controles de calidad.

3.3.4. Atributos del sistema

El servicio web cumplirá con los atributos de fiabilidad, escalabilidad, mantenibilidad, portabilidad y seguridad.

La aplicación deberá ser fiable, es decir, cumplirá con los requisitos y funcionalidades detalladas en los apartados anteriores de forma correcta.

El sistema deberá ser escalable, a fin de adaptarse y soportar el volumen de datos y usuarios que se identifiquen concurrentemente.

El programa deberá ser mantenible, para facilitar la gestión de posibles errores futuros y adaptarse a entornos cambiantes, como cambios de resoluciones, sistemas operativos, etc.

La aplicación deberá ser portable y, por tanto, sencillo de instalar en otros servidores.

Finalmente y aquí es donde reside uno de los atributos más importantes, el proyecto deberá ser seguro e íntegro, es decir, deberá proteger la información privada de los usuarios y los datos que estos manejan cumpliendo con la Ley Orgánica de Protección de Datos.

CAPÍTULO 4

Casos de uso

En este apartado, se analizarán las funcionalidades de más alto nivel en el programa piloto. Se utilizarán diagramas de casos de uso para ilustrar el comportamiento de la aplicación mediante el lenguaje de modelado para sistemas software en UML *Unified Modeling Language*.

Los casos de uso definirán el comportamiento del sistema según lo que los usuarios puedan realizar en él y adoptan la forma de acciones y reacciones. Por tanto, en los mencionados diagramas aparecerán los actores que participarán en el servicio web en los distintos escenarios o secuencias de acciones que podrán llevar a cabo.

Un actor representa el rol desempeñado por un usuario que interactúa con el sistema. Tal y como se ha mencionado en el capítulo anterior, este programa piloto contará únicamente con un rol de usuario y, por tanto, con un único actor, el operario.

4.1 Diagrama casos de uso

La figura 8 muestra el diagrama de casos de uso de este proyecto que, posteriormente, será detallado.

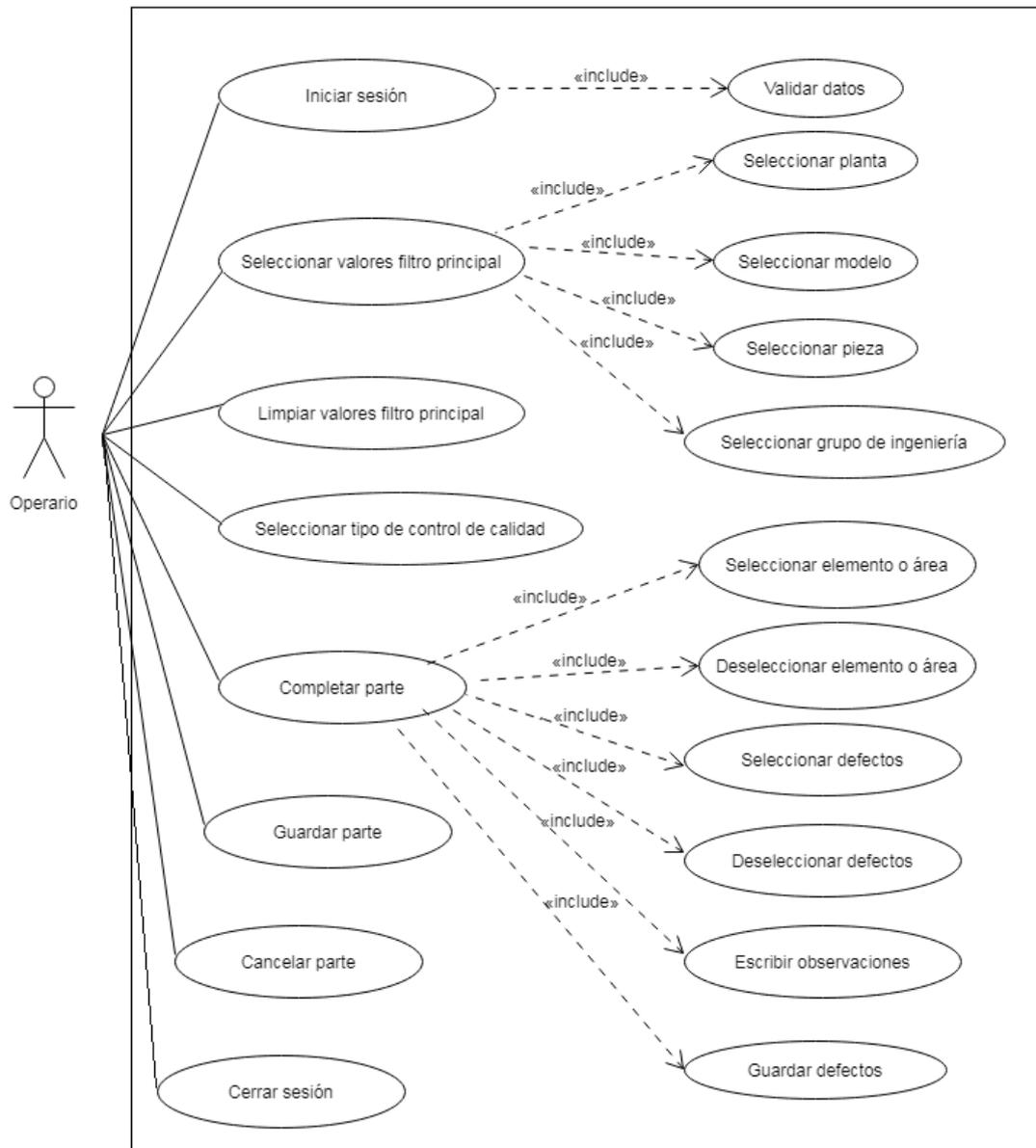


Figura 8: Diagrama de casos de uso

4.2 Descripción casos de uso

En este apartado se detallarán las funcionalidades mostradas en la figura anterior.

Cabe señalar que cada una de las siguientes tablas hará referencia a un requisito que deberá cumplir la aplicación. Además, la fila de *Secuencia normal* de dichas tablas listará por orden de ejecución las acciones que realizarán tanto el actor como el sistema en cada requisito. Sin embargo, en la fila *Excepciones* [6] de dichos requisitos, aparecerán únicamente los puntos de la lista anterior (*Secuencia normal*) que cambiarán el funcionamiento normal de la aplicación debido a las excepciones que se puedan producir en el sistema, pudiendo por tanto ser la numeración no continua.

Tabla 2: Casos de uso

Nombre caso de uso	Iniciar sesión
Actores	Operario
Propósito	Entrar en la aplicación.
Precondiciones	El operario debe trabajar en la empresa Ford a fin de contar con una cuenta en dicha empresa.
Secuencia normal	<ol style="list-style-type: none">1. El sistema cargará su primera pantalla y el nombre del usuario registrado en la sesión activa del sistema operativo.2. El operario introducirá su contraseña.3. El sistema verificará la validez del nombre de usuario y contraseña y permitirá al usuario acceder a la segunda pantalla del sistema.
Excepciones	<ol style="list-style-type: none">1. El operario podrá introducir otro nombre de usuario.3. Si el sistema no tiene constancia de ningún trabajador con la identificación proporcionada, el sistema comunicará la situación al operario permitiéndole corregir los datos.

Nombre caso de uso	Seleccionar valores filtro principal
Actores	Operario
Propósito	Seleccionar los datos para obtener los distintos controles de calidad correspondientes a los datos seleccionados.
Precondiciones	El operario deberá haber accedido a la segunda página del programa que contiene la ventana con el filtro principal.
Secuencia normal	<ol style="list-style-type: none"> 1. El sistema cargará su segunda pantalla y consultará en la base de datos las posibles plantas seleccionables de Ford Valencia. 2. El operario seleccionará la planta en la que vaya a realizar el control de calidad. 3. El sistema accederá a la base de datos para obtener los distintos modelos de vehículos que maneja la planta escogida. 4. El operario elegirá un modelo de vehículo entre las distintas opciones disponibles. 5. El sistema consultará en la base de datos las piezas y los grupos de ingeniería que existen para ese modelo marcado en la planta seleccionada. 6. El usuario seleccionará la pieza que debe examinarse y el grupo de ingeniería al que pertenece, o al revés. 7. El usuario aceptará los datos seleccionados. 8. El sistema comprobará los datos y mostrará los distintos controles de calidad que existen para esos datos en concreto.
Excepciones	<ol style="list-style-type: none"> 8. Si el usuario no ha introducido alguno de los datos requeridos, el sistema mostrará al operario un mensaje informándole de esta situación y permitirá que seleccione los datos faltantes.

Nombre caso de uso	Deseleccionar los datos del filtro principal
Actores	Operario
Propósito	El operario deberá haber accedido a la segunda página del programa que contiene la ventana con el filtro principal y haber seleccionado algún dato de dicho filtro.
Precondiciones	El operario deberá haber accedido a la segunda página del programa que contiene la ventana con el filtro principal.
Secuencia normal	<ol style="list-style-type: none"> 1. El usuario seleccionará la opción de limpiar los datos. 2. El sistema deseleccionará las opciones escogidas. 3. El sistema eliminará los datos obtenidos de las diversas consultas efectuadas a la base de datos, a excepción de la primera que obtenía las distintas plantas seleccionables de Ford Valencia.
Excepciones	<ol style="list-style-type: none"> 2. Si se han mostrado controles de calidad, el sistema ocultará dichos controles y deseleccionará las opciones escogidas en el filtro principal por el usuario.

Nombre caso de uso	Seleccionar tipo de control de calidad
Actores	Operario
Propósito	Escoger el tipo de control de calidad que se realizará.
Precondiciones	El sistema habrá cargado los distintos tipos de control de calidad según los datos seleccionados en el filtro principal.
Secuencia normal	<ol style="list-style-type: none"> 1. El usuario escogerá un control entre los posibles controles de calidad. 2. El sistema cargará el contenido del control de calidad seleccionado, es decir, mostrará el contenido del parte del control de calidad que el operario va a rellenar.
Excepciones	<ol style="list-style-type: none"> 1. Si no es el primer control de calidad que selecciona el operario y previamente no se ha almacenado el parte anterior, el sistema le comunicará al usuario la situación permitiéndole que pueda almacenar o descartar el anterior control de calidad.

Nombre caso de uso	Completar parte
Actores	Operario
Propósito	Anotar los defectos u anomalías hallados en la inspección a la pieza correspondiente en el parte de control de calidad.
Precondiciones	Haber seleccionado un tipo de control de calidad.
Secuencia normal	<ol style="list-style-type: none"> 1. El sistema cargará la imagen correspondiente a los datos seleccionados en el filtro principal y al tipo de control escogido. 2. El usuario seleccionará los elementos o áreas de la imagen cargada en el parte que presenten anomalías. 3. El sistema destacará cada elemento o área seleccionada en la imagen. Además, mostrará para cada área o elemento escogido un panel con los posibles defectos que pueden encontrarse en el tipo de control de calidad seleccionado. 4. El usuario escogerá los defectos que presente cada área o elemento seleccionado, y/o escribirá las observaciones que considere pertinentes y no contempladas por la aplicación. 5. El sistema guardará la información de los diversos elementos o áreas escogidos, lo cual significa que almacenará los defectos y comentarios pertenecientes a cada elemento o área seleccionado.
Excepciones	<ol style="list-style-type: none"> 2. Si no es la primera área o elemento seleccionado en la imagen, el usuario no podrá escoger uno nuevo sin haber almacenado previamente la información del seleccionado con anterioridad. 4. El usuario podrá deseleccionar o modificar los defectos escogidos previamente; incluso podrá deseleccionar áreas o conjuntos de elementos escogidos anteriormente. En caso de que se deseccione un área o elemento almacenado previamente, el sistema borrará las anomalías y comentarios de dicho defecto y lo desmarcará de la imagen. 5. El sistema no podrá guardar un área o elemento seleccionado si no se ha escogido al menos un defecto, ya que carece de sentido añadir en el parte elementos que no sufren anomalías.

Nombre caso de uso	Guardar parte
Actores	Operario
Propósito	Almacenar en la base de datos los defectos u anomalías marcados en el parte de control de calidad escogido.
Precondiciones	Haber seleccionado un tipo de control de calidad y añadido algún elemento.
Secuencia normal	<ol style="list-style-type: none"> 1. El operario seleccionará la opción de guardar el parte de control de calidad. 2. El sistema introducirá y, por tanto, almacenará en la base de datos que sustenta el programa, el conjunto de datos que conforman el parte. Estos datos incluyen: el tipo de control de calidad escogido, el usuario identificado en la aplicación que está realizando el parte; la fecha y hora del momento concreto en el que el usuario guarda el parte; los datos del filtro principal seleccionados; los elementos o áreas marcados en la imagen y por último, los defectos y comentarios de cada elemento escogido.
Excepciones	<ol style="list-style-type: none"> 1. Si el operario no ha escogido al menos un elemento o área defectuosos, o añadido algún comentario, el sistema mostrará un mensaje que avisará al operario de que no ha rellenado el parte de control de calidad y le permitirá completarlo.

Nombre caso de uso	Cancelar parte
Actores	Operario
Propósito	Borrar los defectos o anomalías marcados en el parte de control de calidad escogido.
Precondiciones	Haber seleccionado un tipo de control de calidad y añadido algún elemento.
Secuencia normal	<ol style="list-style-type: none"> 1. El operario seleccionará la opción de cancelar el parte de control de calidad. 2. El sistema desmarcará en la imagen los elementos o áreas seleccionados previamente y borrará la información constatada de estos.
Excepciones	<ol style="list-style-type: none"> 1. Si el operario no ha escogido al menos un elemento o área defectuosa, o añadido algún comentario, el sistema mostrará un mensaje que avisará al operario de que no ha realizado ninguna acción en el parte de control de calidad.

Nombre caso de uso	Cerrar sesión
Actores	Operario
Propósito	Salir de la aplicación.
Precondiciones	El operario debe haber entrado en la segunda página del servicio web.
Secuencia normal	<ol style="list-style-type: none">1. El operario seleccionará la opción de cerrar sesión.2. El sistema verificará que los datos cargados en la segunda página han sido correctamente guardados y permitirá al usuario acceder a la primera pantalla del sistema.
Excepciones	<ol style="list-style-type: none">2. Si el sistema detecta que el operario estaba realizando un parte de control de calidad y dicho parte no ha sido guardado previamente a que el operario escogiese la opción de cerrar la sesión; la aplicación comunicará al operario que debe guardar o cancelar el parte que estaba realizando.

CAPÍTULO 5

Diseño

En este capítulo, se describirán tanto la arquitectura utilizada para el diseño de esta aplicación como la base de datos relacional que lo sustenta.

5.1 Arquitectura de tres capas

En esta aplicación web, con el fin de maximizar la sencillez y la eficacia de su mantenimiento, se ha optado por seguir una arquitectura de tres capas: capa de presentación o interfaz de usuario, capa de negocio y capa de datos.

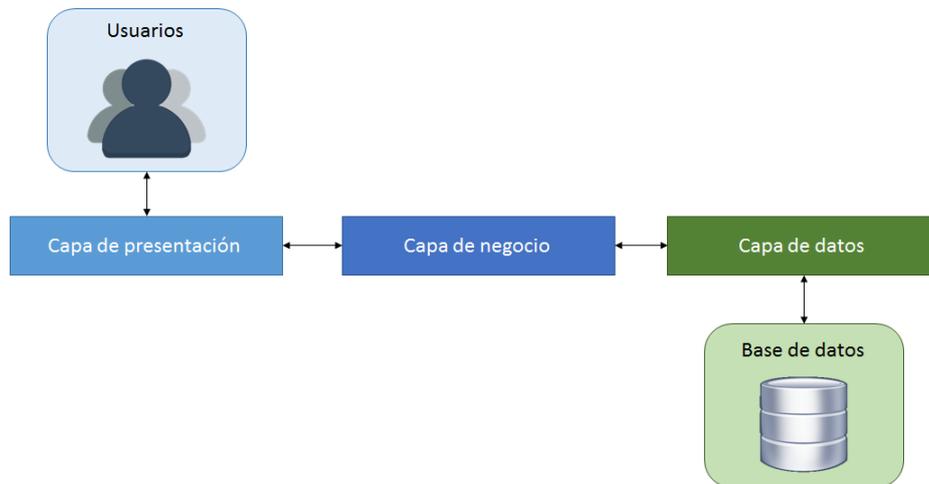


Figura 9: Arquitectura de tres capas

El beneficio de este modelo consiste en que su desarrollo se puede realizar en niveles diferenciados, de tal forma que en el momento en que sea preciso efectuar algún cambio este solo afectará al nivel correspondiente y no será necesario modificar el código de los otros niveles. En consecuencia, la aplicación resultante se mostrará más fácilmente escalable y mantenible.

A continuación, se describen con detalle las capas anteriormente mencionadas.

5.1.1. Capa de presentación

Esta capa muestra al usuario del programa la interfaz gráfica de la aplicación. Las funciones de este nivel consistirán en presentar el sistema gráficamente al usuario así como

captar y comunicar las acciones realizadas por el usuario sobre la capa de negocio. El diseño de las distintas páginas que compondrán la interfaz gráfica de este servicio web se puede observar en las figuras 6 y 7 del capítulo de especificaciones de requisitos.

5.1.2. Capa de negocio

Esta capa gestiona la lógica de la aplicación, lo cual requiere de la comunicación con los otros dos niveles ya que, por un lado, recibe las peticiones y acciones realizadas por el usuario de la capa de presentación y retorna los resultados de dichas acciones; por otro, se comunica con la capa de datos cuando accede al gestor de la base de datos para almacenar o recuperar información. Entre otras, también se encuentran aquí las funciones o métodos encargadas de validar la información introducida por el usuario antes de realizar consultas a la capa de negocio.

5.1.3. Capa de datos

La capa de acceso a los datos se encarga de acceder y almacenar la información requerida por la aplicación. Este servicio web contará con una base de datos relacional del sistema de gestión de bases de datos (SGBD) Microsoft Access.

5.2 Base de datos relacional

El servicio web requerirá de una base de datos relacional que almacene la información necesaria y la gestione de manera eficiente, íntegra y consistente.

Como ya se ha mencionado anteriormente en la memoria, el conjunto de datos para la gestión de controles de calidad con el que Ford cuenta actualmente se almacena en una base de datos no relacional denormalizada desarrollada con Microsoft Access. Dicha base de datos presenta redundancias, carece de algunos datos importantes y carece de la necesaria integridad que caracteriza a las bases de datos relacionales. Como consecuencia de este hecho se ha diseñado una base de datos relacional normalizada que dé soporte y permita el funcionamiento más eficiente posible del programa desarrollado.

El origen de las bases de datos relacionales computerizadas se remonta a la década de los años 70 del siglo pasado. En 1970, Edgar Frank Codd, de la empresa *International Business Machines Corporation* (IBM) desarrolló el modelo de base de datos relacional. Dicha base de datos consiste en una recopilación de elementos de datos con relaciones predefinidas entre ellos. Estos elementos se organizan en un conjunto de tablas compuestas por filas y columnas. El modelo relacional permite acceder a los datos de varias maneras sin reorganizar las propias tablas de la base de datos. Además, que la base de datos esté normalizada significa que se han aplicado una serie de reglas a las relaciones para evitar la redundancia de datos, proteger la integridad de los mismos y así disminuir problemas de actualización de los datos.

Una noción relevante en el diseño de bases de datos relacionales normalizadas es el de *forma normal*. Forma normal es el nombre que recibe un determinado grupo de requisitos de una base de datos; se establecen 6 formas normales sobre diferentes niveles de complejidad de los requisitos de las cuales únicamente interesan en este momento las tres primeras. Se asume que una base de datos cumple con la primera forma normal (1FN)

si se verifica que cada una de sus tablas contendrá una clave primaria única con valor no nulo; los elementos o atributos de las tablas serán simples e indivisibles, de forma que no aparecerán repetidos en otras tablas aunque deberán identificarse con una clave ajena para cumplir con la dependencia funcional, es decir: podrán ser referenciados por otras tablas; además, existirá una independencia del orden en dichos atributos, lo que implica que si los atributos de una tabla cambian de fila o columna no cambiará su significado. La segunda forma normal (2FN) implica que los atributos de las tablas de la base de datos que no sean clave principal dependerán únicamente de la clave principal de su propia tabla, de forma que no existirán dependencias parciales y se constatará una dependencia completamente funcional. La tercera forma normal (3FN) se alcanza cuando se añade a estos requisitos la inexistencia de dependencia funcional transitiva alguna entre los atributos que no son clave. Las tablas de la base de datos relacional desarrollada para el servicio web cumplirán con la tercera forma normal (3FN) lo cual, como ha podido observarse, implica el cumplimiento de las formas normales previas (2FN y 1FN).

La figura 10 representa en un diagrama UML la base de datos relacional normalizada que se implementará en Microsoft Access para sustentar la aplicación de servicio web; a petición de Ford, el idioma que se empleará será el inglés americano.

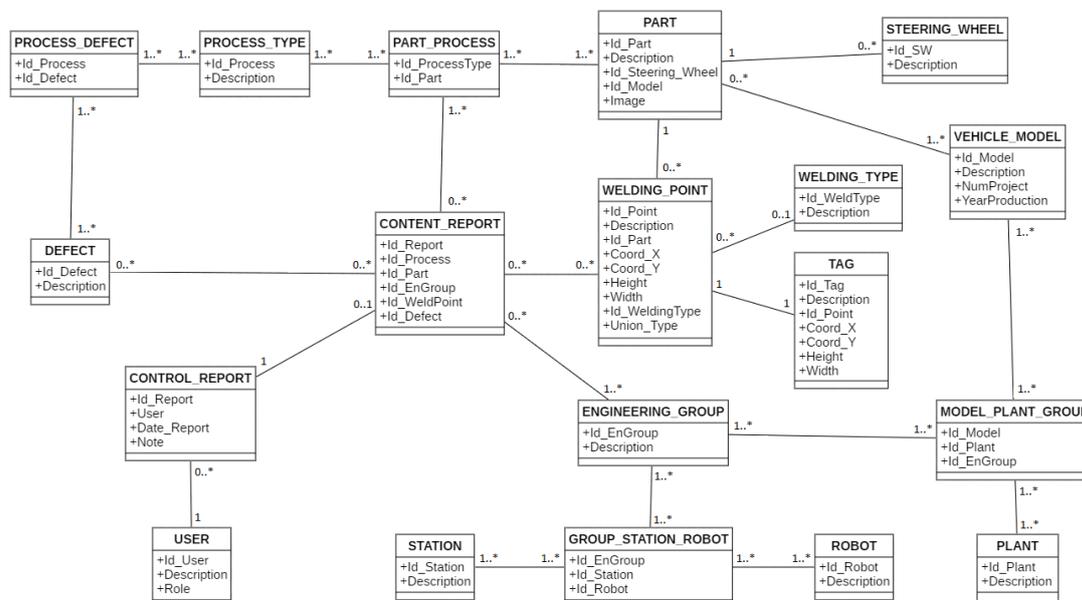


Figura 10: Diagrama UML de la base de datos

Las relaciones que se establecen entre tablas se muestran en este diagrama como líneas que contienen números con los que se expresan las restricciones de la relación entre ellas.

Se han definido como claves primarias los identificadores únicos de cada tabla (algunas de las cuales emplean varios identificadores) con el fin de posibilitar las relaciones entre ellas. Al mismo tiempo, para contribuir a su visualización se les ha añadido en el nombre la abreviatura *Id*. El resto de atributos de una tabla se han identificado como claves ajenas y constan a continuación de las claves primarias.

Las funciones de las distintas tablas serán detalladas a continuación:

Plant: contendrá la información relativa a las plantas que existen en Ford España Almus-safes; dicha información se compondrá del valor numérico que identificará la planta y el nombre común por el que es conocida por los trabajadores de Ford; por ejemplo: prensas, carrocería, pintura, etc.

Vehicle_Model: almacenará la información correspondiente a cada modelo de vehículo que se fabrica en Ford España Almussafes, como Mondeo, Transit, Kuga, S-Max y Galaxy, además de otras informaciones relativa a estos modelos.

Part: guardará los datos correspondientes a las piezas que contiene cada modelo de vehículo fabricado en Ford España Almussafes, incluidos el sentido del volante para la fabricación de coches ingleses entre otros y la dirección de la carpeta con las imágenes de la pieza correspondiente.

Steering_Wheel: en esta tabla se almacenan las distintas orientaciones del volante del vehículo; es decir, si el volante se colocará a la izquierda o a la derecha.

Welding_Point: contendrá los datos de todos los puntos pertenecientes a cada pieza de cada modelo; la ubicación o coordenada de estos y su tamaño; el tipo de punto; si son puntos de control que deben existir y no presentar defectos para que la pieza sea válida y el tipo de unión que ayudan a realizar.

Welding_Type: almacenará los distintos tipos de puntos que existen, como soldadura, selladores, etc.

Tag: Ford dispone de planos para que los operarios puedan realizar los distintos controles de calidad de las piezas correspondientes en los que cada elemento o área de dicha pieza está identificada por una etiqueta única con el nombre único de ese elemento o área y que señala su ubicación exacta en el plano. Esta tabla contendrá la información correspondiente a esas etiquetas, como la identificación de la etiqueta, el nombre que contiene, así como las coordenadas y tamaño del recuadro que conforma dicha etiqueta en el plano.

Engineering_Group: guardará el código identificativo y el nombre de cada grupo de ingeniería o subárea de trabajo de cada planta de Ford España Almussafes.

Station: contendrá el valor numérico identificativo de las distintas estaciones de trabajo que existen en cada planta de Ford España Almussafes.

Robot: almacenará la información relativa a los identificadores y nombre de los distintos robots existentes en las plantas de Ford España Almussafes.

Process_Type: guardará los datos de los distintos controles de calidad o procesos de calidad que se pueden realizar, como control de apariencia superficial, puntos de soldadura, soldadura de tuercas y casquillos, selladores, etc.

Defect: archivará la información correspondiente a los distintos defectos que se pueden hallar según los controles de calidad.

User: contendrá el identificador de los usuarios, la cuenta en Ford correspondiente y el rol que desempeña. No obstante, no almacenará la contraseña, ya que la aplicación la validará directamente en el servidor de Ford España Almussafes. Por otro lado, tal y como se ha mencionado anteriormente en la memoria, solo se implementará un único rol de usuario, el operario, por requerimiento de Ford.

Model_Plant_Group: puesto que dentro de Ford España Almussafes los mismos modelos de vehículo y grupos de ingeniería coexisten en más de una planta con los mismos identificadores y nombres, esta tabla sirve para crear una clave ternaria única que permitirá identificar exactamente la planta, modelo de vehículo y grupo de ingeniería sobre el que se desea trabajar.

Group_Station_Robot: del mismo modo que sucedía en la tabla *Model_Plant_Group*, la información correspondiente a estaciones de trabajo y robots aparece repetida en distintos grupos de ingeniería. Para solventar esta situación potencialmente problemática, esta tabla permitirá identificar de manera unívoca y mediante una clave ternaria el grupo de ingeniería, la estación y el robot con los que se desea interactuar.

Part_Process: cada pieza pasa por distintos controles de calidad, pero todas las piezas no sufren los mismos controles de calidad. Por esta razón, esta tabla almacenará mediante una clave binaria los distintos controles que sufre una misma pieza.

Process_Defect: como en cada control de calidad se pueden detectar varios defectos y algunos de estos defectos pueden ser comunes a otros controles de calidad, esta tabla contendrá una clave binaria con los distintos defectos que se pueden hallar en un único control de calidad.

Control_Report: almacenará los datos que hayan sido añadidos por el usuario al completar y guardar el parte de calidad de control de una pieza. Por tanto, contendrá el valor numérico automático que identificará el parte de control de calidad, la cuenta de Ford del operario que ha realizado dicho parte, la fecha y hora en que ha sido guardado y las observaciones o notas que haya añadido dicho usuario en caso de haber efectuado alguna.

Content_Report: esta tabla almacenará la información constitutiva del parte de control y los datos que haya seleccionado el operario tales como la planta en la que se efectúa el control, el modelo de vehículo seleccionado, la pieza escogida, el grupo de ingeniería al que pertenece dicho usuario, otros elementos anómalos hallados en la inspección y los distintos defectos de cada elemento anómalo hallado.

CAPÍTULO 6

Implementación

Una vez explicado el diseño utilizado en el desarrollo de este proyecto, se procederá a clarificar su implementación. En primer lugar, se comentarán las tecnologías y herramientas utilizadas para su desarrollo. Seguidamente, se detallará la estructura seguida por el proyecto.

6.1 Tecnologías y herramientas

En este apartado se describirán las diversas tecnologías y herramientas utilizadas en la implementación de la aplicación web.

6.1.1. XAML

XAML [7] siglas de *eXtensible Application Markup Language*, es un lenguaje de interfaz de usuario utilizado para implementar aplicaciones en *Windows Presentation Foundation* (WPF) y Silverlight.

Es un lenguaje basado en XML (*eXtensible Markup Language*) que ha sido optimizado para describir gráficamente interfaces de usuarios ricas desde el punto de vista gráfico, como las creadas por medio de Adobe Flash, XUL y UIML. Por otro lado, XAML se diseñó con el fin de que pudiera soportar las clases y métodos de la plataforma de desarrollo .NET.

Por estas razones, además de por la petición expresa de la empresa, este es el lenguaje empleado en la implementación de la capa de interfaz de usuario o presentación.

6.1.2. C# y .NET

C# [8] es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft. Dicho lenguaje permite a los desarrolladores crear una gran variedad de aplicaciones que se ejecutan en .NET Framework.

Por otro lado, .NET [9] es una plataforma desarrollada por Microsoft que ofrece un conjunto de herramientas, tecnologías y servicios que facilita el desarrollo de aplicaciones software. Asimismo, .NET proporciona soporte para diferentes lenguajes, como F#, Visual Basic, C++, C#, entre otros. La versión .NET Framework 4.5.2 proporciona una serie de características como lo son las mejoras en la generación de perfiles y depuración, cambios en el seguimiento de eventos, nuevas aplicaciones ASP.NET y nuevas propiedades

en el flujo de trabajo.

En este proyecto, ha sido utilizado .NET Framework 4.5.2 en la capa de negocio para su desarrollo e implementación. El lenguaje C#, sin embargo, ha sido empleado en todas las capas que componen la aplicación.

6.1.3. SQL

SQL (*Structured Query Language*) [4] es un lenguaje diseñado para administrar sistemas de gestión de bases de datos relacionales.

En esta aplicación, se ha utilizado en la capa de datos, ya que permite su acceso y manipulación, ya sea por medio de acciones como la creación, borrado y modificación de tablas, la inserción, consulta, borrado y actualización de datos, etc.

6.1.4. Visual Studio 2015

Visual Studio [10] es un entorno para el desarrollo integrado creado por Microsoft. Este entorno soporta una gran variedad de lenguajes de programación, tales como C++, C#, Visual Basic, F#, Java, Python, Ruby, PHP e integra .NET.

Los programadores de Ford trabajan en este entorno, ya que cuenta con una serie de herramientas muy útiles para la plataforma .NET. Para el desarrollo de este servicio web se ha empleado la versión 2015 puesto que Ford cuenta con licencias para la misma.

6.1.5. Internet Explorer

Internet Explorer [11] es un navegador web desarrollado por Microsoft. Cabe señalar que para desarrollar y visualizar una aplicación web se necesita utilizar al menos un navegador web, ya sea Chrome, Firefox, Internet Explorer, etc. Por otra parte, tal y como se ha comentado anteriormente, Ford emplea Internet Explorer como navegador principal y exclusivo en sus instalaciones incluso en los ordenadores o tabletas con el sistema operativo Windows 10. En consecuencia, la empresa requiere el desarrollo e implementación de este servicio web para ser explotado en este navegador. En concreto, el servicio web ha sido diseñado específicamente para la versión 11.0.9600.19002 de Internet Explorer.

6.1.6. Microsoft Access

Microsoft Access es el sistema de gestión de bases de datos (SGBD) en el que se ha implementado la base de datos relacional que da soporte a la aplicación web. La versión de Access empleada es la correspondiente al paquete Office 2016.

En el código del programa web, dentro de la capa de datos se ha utilizado el proveedor de Microsoft Microsoft.ACE.OLEDB.12.0 para establecer las conexiones con dicho SGBD.

6.1.7. LaTeX

LaTeX [12] (son las siglas de *Lamport TeX*) es un sistema de preparación de documentos, ideado por *Leslie Lamport*, utilizado en la redacción de la memoria de este proyecto. Los

documentos en LaTeX se construyen en texto plano en combinación con etiquetas de marcado y son estas etiquetas las que estructuran el texto para darle formato al documento.

6.1.8. IIS

Internet Information Services (IIS) [13] es un servidor web y un conjunto de servicios que se utilizan en el sistema operativo *Windows* de Microsoft. IIS permite que un ordenador se convierta en un servidor web para publicar páginas web tanto local como remotamente. Además, cuando se desarrollan proyectos web se requiere un servidor para ejecutarlos y Visual Studio permite utilizar este servidor web.

6.1.9. GIMP

GIMP (*GNU Image Manipulation Program*) es un programa libre y gratuito para la edición de imágenes.

Se ha empleado la versión GIMP 2.8.22. debido a que es la única que el soporte técnico de Ford permite instalar. Asimismo, se ha recurrido a este programa en la edición de imágenes que contienen tanto la aplicación web como la memoria de este TFG.

6.1.10. StarUML

StarUML [14] es una herramienta o programa de software libre para crear diagramas en los estándares UML [15].

StarUML ha sido utilizado en esta aplicación para crear el diagrama de la base de datos que puede observarse en la figura 10.

6.2 Estructura del proyecto

Como se ha mencionado anteriormente, la aplicación ha sido diseñada siguiendo la arquitectura de tres capas. Por tanto, la estructura de los ficheros que componen este servicio web han sido organizados siguiendo dicha arquitectura, tal como muestra la figura 11.

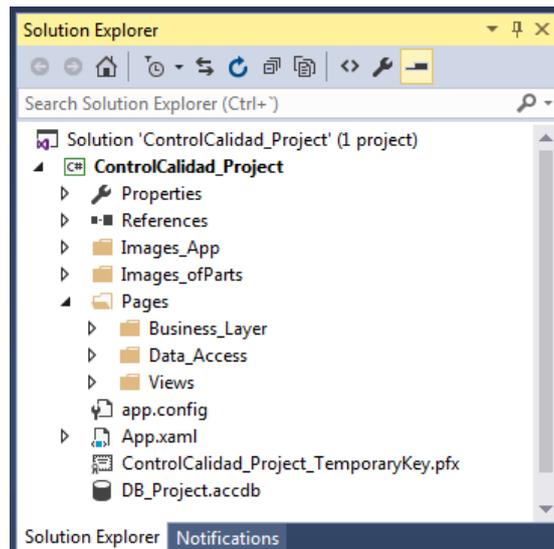


Figura 11: Estructura del proyecto

La carpeta de *Images_App* contiene todas las imágenes que forman parte del diseño de la aplicación, tales como logos, fondos o iconos.

En la carpeta denominada *Images_ofParts* se hallan las distintas imágenes de las piezas o *parts* que serán examinados. Cabe señalar que una única pieza puede dar lugar a diversas imágenes dependiendo del control de calidad que se le aplique. Por ejemplo, la puerta derecha del Mondeo de techo normal consta de tres imágenes, cada una de ellas con la información necesaria para su correspondiente control de calidad. Si dicho control examina la apariencia superficial de la pieza, su imagen mostrará la chapa de la pieza seccionada por una gradilla, es decir, dividida en filas y columnas. Si, por el contrario, en el control de calidad se examinan los puntos de soldadura la imagen correspondiente contendrá todos los puntos y etiquetas correspondientes; por último, si se trata de un control de calidad de inspección de los sellados, la imagen correspondiente de dicha pieza mostrará todos sus sellados. Las imágenes que se cargan en dichos controles de calidad se muestran en la figuras 12, 13 y 14.

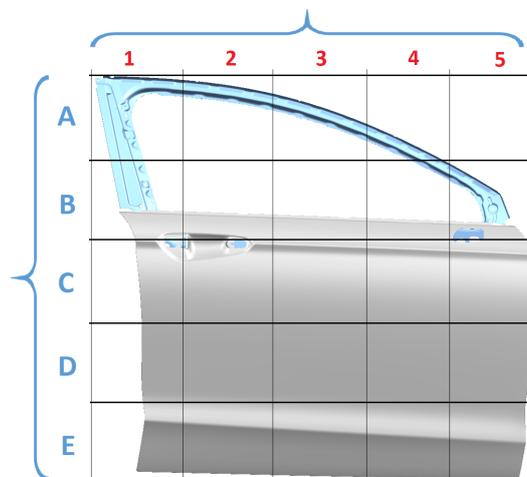


Figura 12: Control apariencia superficial de la puerta derecha del Mondeo de techo normal

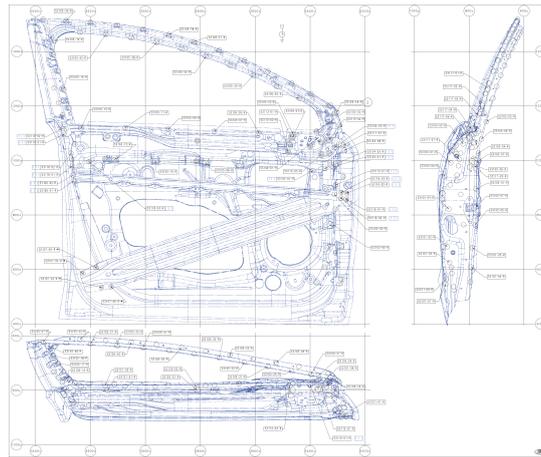


Figura 13: Control puntos de soldadura de la puerta derecha del Mondeo de techo normal

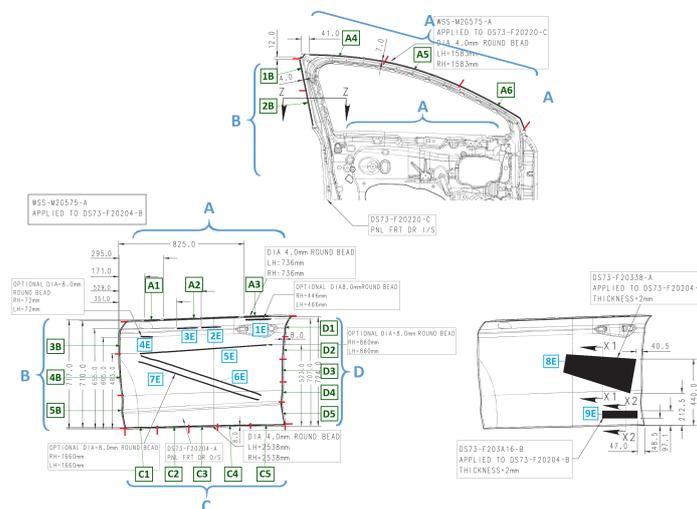


Figura 14: Control selladores de la puerta derecha del Mondeo de techo normal

En la carpeta *Pages*, se hallan las subcarpetas que diferencian las capas de presentación, negocio y acceso a datos con sus respectivos archivos. La implementación de dichas capas se comenta a continuación.

6.2.1. Capa de presentación

La capa de presentación se corresponde con la subcarpeta llamada *Views*. Dicha carpeta contiene todos los archivos *.xaml* y *.xaml.cs* que dan forma y gestionan los eventos que se reciben desde la vista de la aplicación. En el siguiente capítulo de evaluación y pruebas se muestran diversas figuras o ilustraciones de la interfaz gráfica de esta aplicación web.

Cuando se crea una interfaz en un proyecto de tipo *WPF Browser App* en Visual Studio 2015, son creados dos ficheros enlazados. El primer fichero contiene el código de la interfaz gráfica en XAML, por lo que su extensión es *.xaml*. Por otra parte, se crea el *code behind* que es el otro archivo. Dicho archivo presenta la extensión *.xaml.cs*, ya que está vinculado a la interfaz y se encarga de gestionar las acciones realizadas por el usuario mediante C# (cuya extensión es *.cs*). Estos hechos pueden observarse en la figura 15.

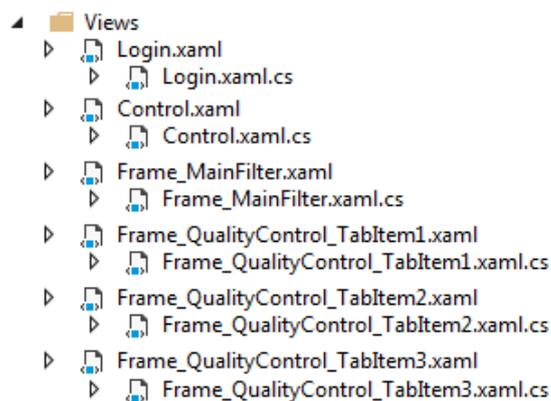


Figura 15: Archivos de la capa de presentación

Los archivos *Login.xaml* y *Control.xaml* contienen todos los controles y elementos gráficos de las páginas principales de la aplicación y los nombres de los eventos que se producen en ellos. No obstante, tal y como ya se ha comentado en el capítulo de requisitos en la figura 7, el control de calidad que corresponde al fichero *Control.xaml* se conforma a partir de otros ficheros. El primero de ellos es *Frame_MainFilter.xaml*, que corresponde a los elementos gráficos del filtro principal. Seguidamente, y dependiendo del control de calidad que el usuario haya seleccionado, se carga el contenido de dicho parte gracias a los ficheros que componen la interfaz gráfica del contenido de cada tipo de control. Si el control seleccionado resulta ser del primer tipo se cargará la interfaz del *Frame_QualityControl_TabItem1.xaml* y así sucesivamente.

Por otra parte, los ficheros *Login.xaml.cs*, *Control.xaml.cs*, *Frame_MainFilter.xaml.cs* y los distintos *Frame_QualityControl_TabItemX.xaml.cs* contienen las funciones que definen el comportamiento de los eventos relacionados con los elementos gráficos de sus archivos *.xaml* correspondientes. Dichas funciones reciben o envían datos a los módulos o archivos correspondientes de la capa de negocio en el caso de que un evento necesite consultar o almacenar información en la base de datos.

Por último, tal y como se mencionó en el capítulo de requisitos, la interfaz gráfica de todos los archivos *.xaml* que componen de la aplicación debe adaptarse a las distintas resoluciones de pantalla que contienen los ordenadores y tabletas de Ford. Con dicho objetivo, todos los documentos *.xaml* de este servicio web han sido implementados con la resolución 1366x768, ya que el programa Visual Studio 2015 se encarga de readaptar automáticamente la resolución y el resultado es mejor cuando transforma de resoluciones altas a bajas para pantallas más pequeñas. Asimismo, todos los archivos *.xaml* cuentan con barras de desplazamiento vertical y horizontales cuya visibilidad es automática, es decir, solo se muestran si la interfaz gráfica de la aplicación no puede mostrarse en su totalidad en la pantalla del dispositivo desde el que se está ejecutando dicho programa.

6.2.2. Capa de negocio

La subcarpeta *Business_Layer* contiene los archivos desarrollados en C# que sirven de intermediarios a las dos capas restantes, es decir, acceden a los archivos de la capa de presentación para obtener o insertar datos en la base de datos.



Figura 16: Archivos de la capa de negocio

En el archivo *Mediator_Login.cs*, se han programado las funciones que reciben la información necesaria del *Login.xaml.cs*, tales como los nombres de usuario e identificadores, a fin de validar dichos datos en el servidor de Ford y retornarlos al fichero *Login.xaml.cs*.

Mediator_MainFilter.cs está compuesto por las funciones que interactúan con las consultas de los datos seleccionados en el archivo *Frame_MainFilter.xaml.cs*, tales como planta, modelo, pieza y grupo de ingeniería seleccionado. La interacción consiste en el envío de dichos datos y la posterior recepción de la respuesta a la función del módulo correspondiente de la capa de acceso a datos.

Mediator_Control.cs contiene las funciones que reciben los datos del control de calidad seleccionado por el usuario y los seleccionados en el filtro principal. El objetivo de estas funciones consiste en efectuar una consulta a la función del módulo correspondiente de la capa de acceso a datos y enviar posteriormente la información pertinente al archivo *Control.xaml.cs* para que cargue el contenido del control de calidad seleccionado (es decir, al archivo *Frame_QualityControl_TabItemX.xaml.cs* correspondiente).

Mediator_TabItem.cs contiene las funciones que reciben los datos del filtro principal seleccionados y del control de calidad escogido y conecta con el módulo y funciones correspondientes de la capa de acceso a datos. El objetivo no es otro que efectuar la consulta y obtener la información que se cargará en el *Frame_QualityControl_TabItemX.xaml.cs* correspondiente cuando esta información sea enviada. Las funciones de *Mediator_TabItem.cs* obtendrán la imagen que se debe cargar y los datos correspondientes a esta, tales como puntos o áreas seleccionables y sus respectivas etiquetas, además de los posibles defectos del tipo de control seleccionado.

6.2.3. Capa de datos

Por último, la subcarpeta *Data_Access* contiene las clases o archivos que se comunican con la base de datos relacional que soporta el proyecto, llamada *BD_Project* y que presenta la extensión *.acddb*.

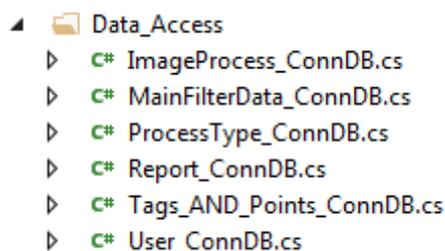


Figura 17: Archivos de la capa acceso a datos

El módulo *ImageProcess_ConnDB.cs* contiene las funciones necesarias en C# para conectarse con la base de datos que da soporte al programa mediante consultas SQL con el objeto *DataReader* de OLE DB. Estas funciones permiten averiguar la ubicación de la imagen correspondiente a los datos seleccionados en la página del control de calidad de la aplicación web.

En el archivo *MainFilterData_ConnDB.cs*, se hallan todas las funciones que se conectan a la base de datos mediante consultas SQL. Estas funciones utilizan el objeto *DataSet* (que contiene el conector OLE DB) con el fin de enlazar la información obtenida al contenido de los *comboBoxes* o elementos desplegable del filtro principal. Dicha información, como ya se ha mencionado anteriormente, se filtra a partir de los datos seleccionados por el operario de tal modo que, dependiendo de lo seleccionado, se irían mostrando en cascada por filtros sucesivos las plantas de Ford España Almussafes, los modelos de vehículo que fabrican en la planta escogida, las piezas que conforman el modelo seleccionado y los grupos de ingeniería que trabajan con dicho modelo.

El fichero *ProcessType_ConnDB.cs* contiene las funciones que se conectan a la base de datos por medio de consultas SQL (mediante el objeto *DataReader*) con la finalidad de obtener los diferentes procesos de calidad o controles de calidad por los que pasa la pieza seleccionada en el filtro principal.

Report_ConnDB.cs está constituido por las funciones que insertan en la base de datos toda la información generada por el usuario durante el control de calidad mediante consultas SQL.

Tags_AND_Points_ConnDB.cs contiene las funciones que consultan a la base de datos mediante consultas SQL los puntos o elementos seleccionables y sus respectivas etiquetas según la imagen cargada en el parte de control de calidad. De ese modo, la aplicación cuenta con la información necesaria para determinar si cuando un operario selecciona un punto en la imagen dicho punto se debe iluminar y mostrar sus datos o no.

Por último, el fichero *User_ConnDB.cs* consta de las funciones que acceden a la base de datos para obtener el nombre de usuario y el rol que le pertenece. No obstante, por limitaciones impuestas por Ford y por el propio alcance de este TFG, de momento este rol está limitado exclusivamente al de *operario*.

CAPÍTULO 7

Evaluación y pruebas

En este apartado, se evaluará el comportamiento de la aplicación, se verificará la calidad del producto y se constatará su correcto funcionamiento y el cumplimiento de las funcionalidades especificadas en los capítulos de requisitos y casos de uso. Conjuntamente, se presentarán los test y pruebas de uso realizadas durante la implementación.

7.1 Prueba de validación de XAML

Como se ha mencionado anteriormente, en la aplicación se han implementado varios archivos XAML para crear la interfaz gráfica de este servicio web. En este caso, se han empleado las pruebas del validador W3C para el código XML, puesto que el lenguaje XAML se basa en XML. En dicho validador se ha comprobado el código del que están constituidos estos archivos .xaml y se han ido corrigiendo hasta que todos ellos han superado las pruebas. Las figuras 18, 19, 20 y 21 muestran el resultado de la validación para las pruebas de código XML.

El enlace a la página en la que se ha validado el código XAML es el siguiente: http://validator.w3.org/#validate_by_upload

The screenshot shows the W3C Markup Validation Service interface. At the top, there is a blue header with the W3C logo and the text 'Markup Validation Service' and 'Check the markup (HTML, XHTML, ...) of Web documents'. Below the header, there are navigation links: 'Jump To:', 'Notes and Potential Issues', and 'Congratulations - Icons'. A green banner displays the message: 'This document was successfully checked as well-formed XML!'. Below this, a table shows the validation details:

Result:	Passed
File:	FolCalidad_Project\Pages\Login.xaml Examinar... <small>Use the file selection box above if you wish to re-validate the uploaded file Login.xaml</small>
Encoding:	utf-8 (detect automatically)
Doctype:	XML (detect automatically)
Root Element:	Page
Root Namespace:	http://schemas.microsoft.com/winfx/2006/xaml/presentation

At the bottom, there is an HP logo, a note that 'The W3C validators are hosted on server technology donated by HP, and supported by community donations. Donate and help us build better tools for a better web.', and a 'Flattr' button.

Figura 18: Validación de la página de inicio

W3C® Markup Validation Service
Check the markup (HTML, XHTML, ...) of Web documents

Jump To: Notes and Potential Issues Congratulations - Icons

This document was successfully checked as well-formed XML!

Result:	Passed
File:	<input type="text" value="iCalidad_Project\Pages\Control.xml"/> Examinar...
	<small>Use the file selection box above if you wish to re-validate the uploaded file Control.xml</small>
Encoding:	utf-8 <input type="button" value="(detect automatically)"/>
Doctype:	XML <input type="button" value="(detect automatically)"/>
Root Element:	Page
Root Namespace:	http://schemas.microsoft.com/winfx/2006/xaml/presentation

The W3C validators are hosted on server technology donated by HP, and supported by community donations. [Donate](#) and help us build better tools for a better web.

Figura 19: Validación de la página del control de calidad

W3C® Markup Validation Service
Check the markup (HTML, XHTML, ...) of Web documents

Jump To: Notes and Potential Issues Congratulations - Icons

This document was successfully checked as well-formed XML!

Result:	Passed
File:	<input type="text" value="Project\Pages\Frame_MainFilter.xml"/> Examinar...
	<small>Use the file selection box above if you wish to re-validate the uploaded file Frame_MainFilter.xml</small>
Encoding:	utf-8 <input type="button" value="(detect automatically)"/>
Doctype:	XML <input type="button" value="(detect automatically)"/>
Root Element:	Page
Root Namespace:	http://schemas.microsoft.com/winfx/2006/xaml/presentation

The W3C validators are hosted on server technology donated by HP, and supported by community donations. [Donate](#) and help us build better tools for a better web.

Figura 20: Validación del filtro principal

W3C® Markup Validation Service
Check the markup (HTML, XHTML, ...) of Web documents

Jump To: Notes and Potential Issues Congratulations - Icons

This document was successfully checked as well-formed XML!

Result:	Passed
File:	<input type="text" value="Frame_QualityControl_TabItem1.xml"/> Examinar...
	<small>Use the file selection box above if you wish to re-validate the uploaded file Frame_QualityControl_TabItem1.xml</small>
Encoding:	utf-8 <input type="button" value="(detect automatically)"/>
Doctype:	XML <input type="button" value="(detect automatically)"/>
Root Element:	Page
Root Namespace:	http://schemas.microsoft.com/winfx/2006/xaml/presentation

The W3C validators are hosted on server technology donated by HP, and supported by community donations. [Donate](#) and help us build better tools for a better web.

Figura 21: Validación del parte de control de calidad

7.2 Prueba de uso en el navegador web

Durante la etapa de implementación se han ido realizando comprobaciones de la funcionalidad de la aplicación con el objetivo de asegurar su eficiencia y capacidad de respuesta

para cada uno de los requisitos exigidos por el cliente. Además, como las pruebas se han llevado a cabo en el entorno proporcionado por el navegador Internet Explorer, se confirma que el servicio web funciona correctamente en dicho navegador. Asimismo, se ha verificado que los datos se almacenan correctamente en la base de datos que sustenta esta aplicación.

A continuación, se mostrarán las capturas de pantalla en las que se ilustra el funcionamiento global de la aplicación web. Dichas capturas se han tomado desde la pantalla de un ordenador con resolución 1366x768.

7.2.1. Inicio de sesión

El inicio de sesión permite acceder a la aplicación únicamente a trabajadores que cuenten con una cuenta en el servidor de Ford España Almussafes (Valencia). Además, el nombre de usuario que se carga de forma automática en la aplicación, como se mencionó en los requisitos, es el nombre del usuario registrado en la cuenta activa en el sistema operativo. En la figura 22 se observa la interfaz de la página de inicio del servicio web.

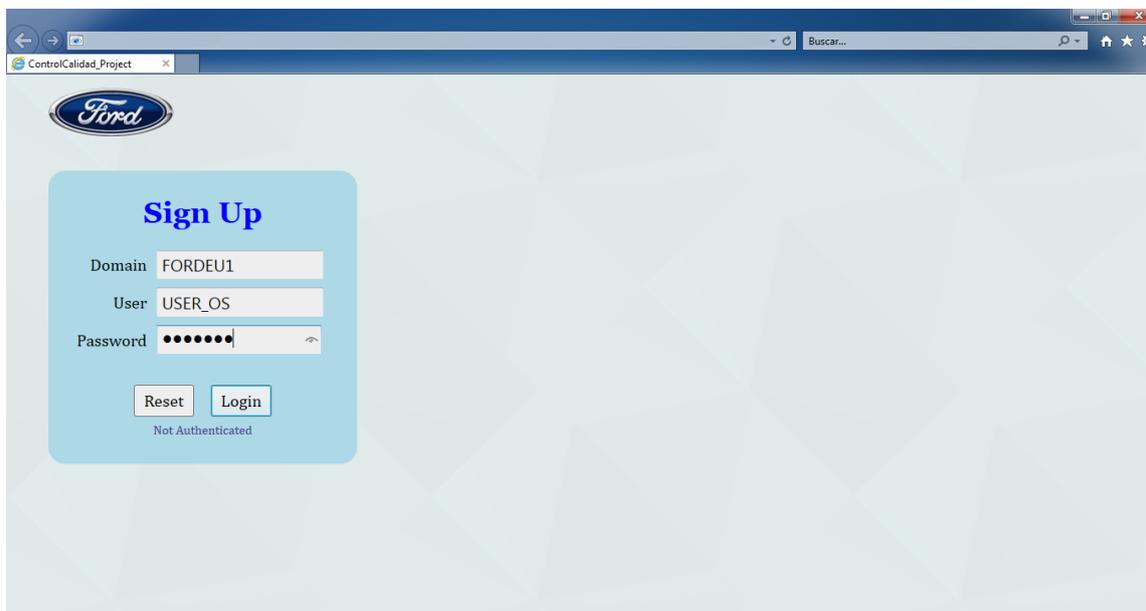


Figura 22: Interfaz página de inicio

7.2.2. Filtro principal

Antes de que los usuarios registrados puedan llevar a cabo un parte de control, primero han de filtrar los datos necesarios para cargar el parte correcto. La aplicación está delimitada de momento al siguiente caso específico: planta de *carrocerías*; modelo de vehículo *Mondeo*; como pieza susceptible del control, una *puerta derecha del modelo de techo normal* y, como grupo de ingeniería al que pertenece el usuario, el *5B*.

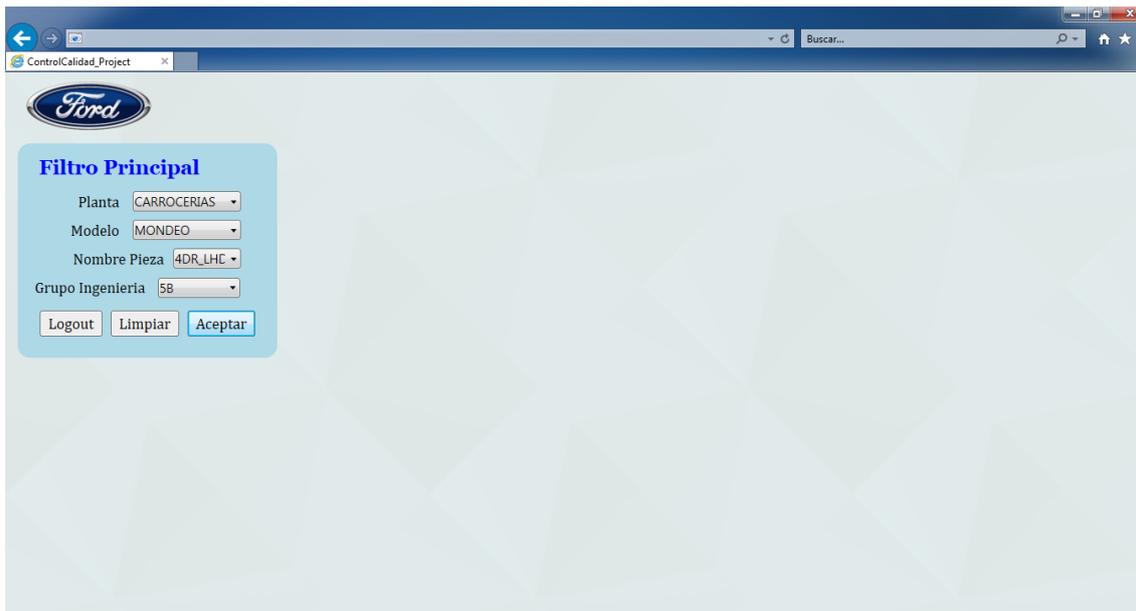


Figura 23: Interfaz filtro principal

7.2.3. Parte de control de calidad vacío

Una vez introducidos los datos, se cargan los distintos controles de calidad correspondientes a los datos seleccionados en el filtro principal. El usuario puede optar por cualquiera de los controles de calidad ofrecidos; en este caso, los controles disponibles son: Apariencia superficial; Puntos de Soldadura y Selladores. Dependiendo del control de calidad escogido por el operario, el contenido del parte correspondiente a dicho control cambiará, tal y como puede comprobarse en la figura 24.

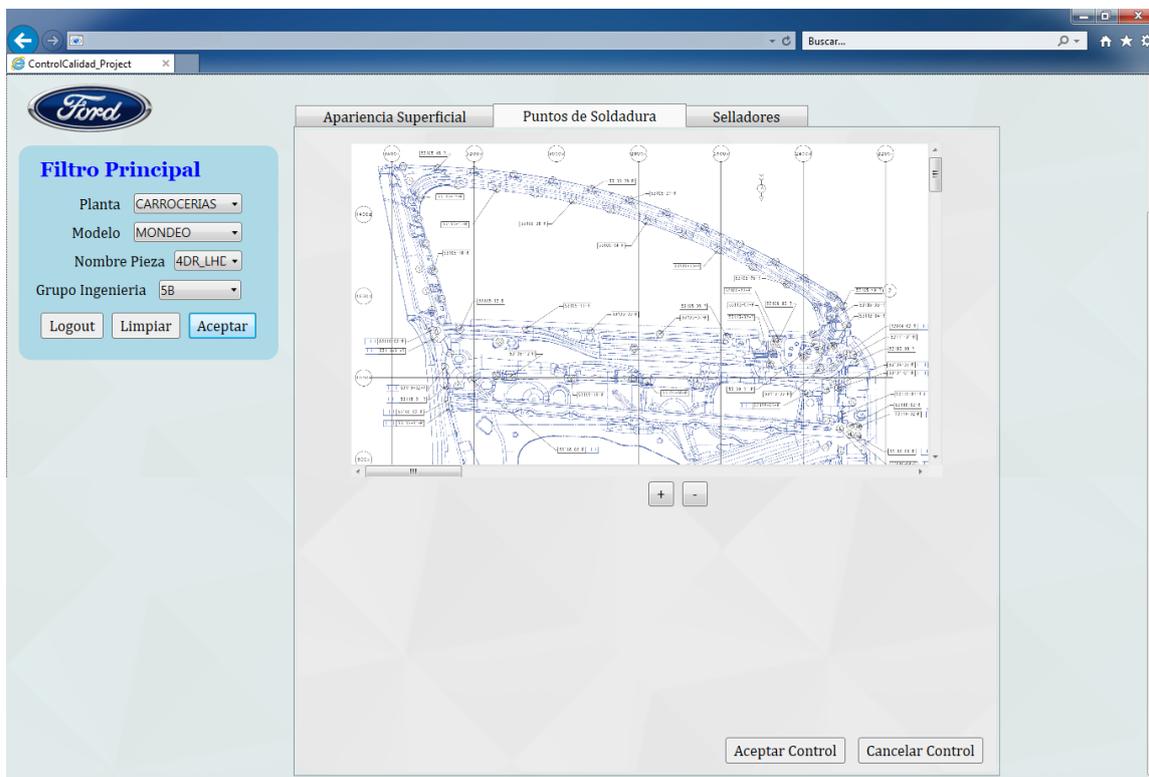


Figura 24: Interfaz parte de control de calidad vacío

7.2.4. Parte de control de calidad con datos

Tal como se observa en la figura 25, el usuario puede aumentar el tamaño de la imagen cargada, seleccionar diversos elementos de dicha imagen y seleccionar o indicar los defectos detectados.

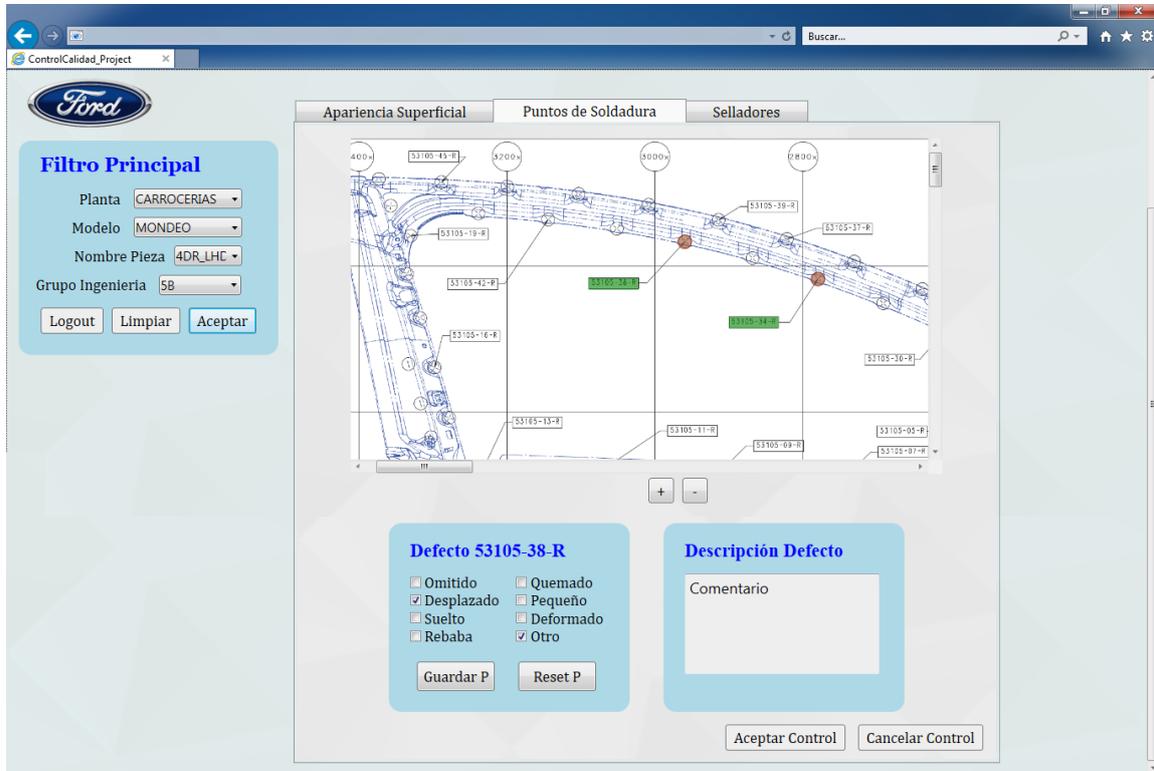


Figura 25: Interfaz parte de control de calidad con datos

7.3 Prueba de distintas resoluciones

Este proyecto se diseñó para que pudiera implantarse y ser utilizado correctamente en los dispositivos con los que cuenta Ford: portátiles, ordenadores de sobremesa y, especialmente, tabletas. Las resoluciones de pantalla de dichos dispositivos son muy diversas y, por otra parte, la visualización de la aplicación en tabletas puede presentarse tanto en vertical como en horizontal.

A continuación, se mostrarán unas capturas de pantalla de la aplicación web con las resoluciones más comunes con las que trabaja Ford.

- La resolución 1366x768 se muestra en la figura 26.

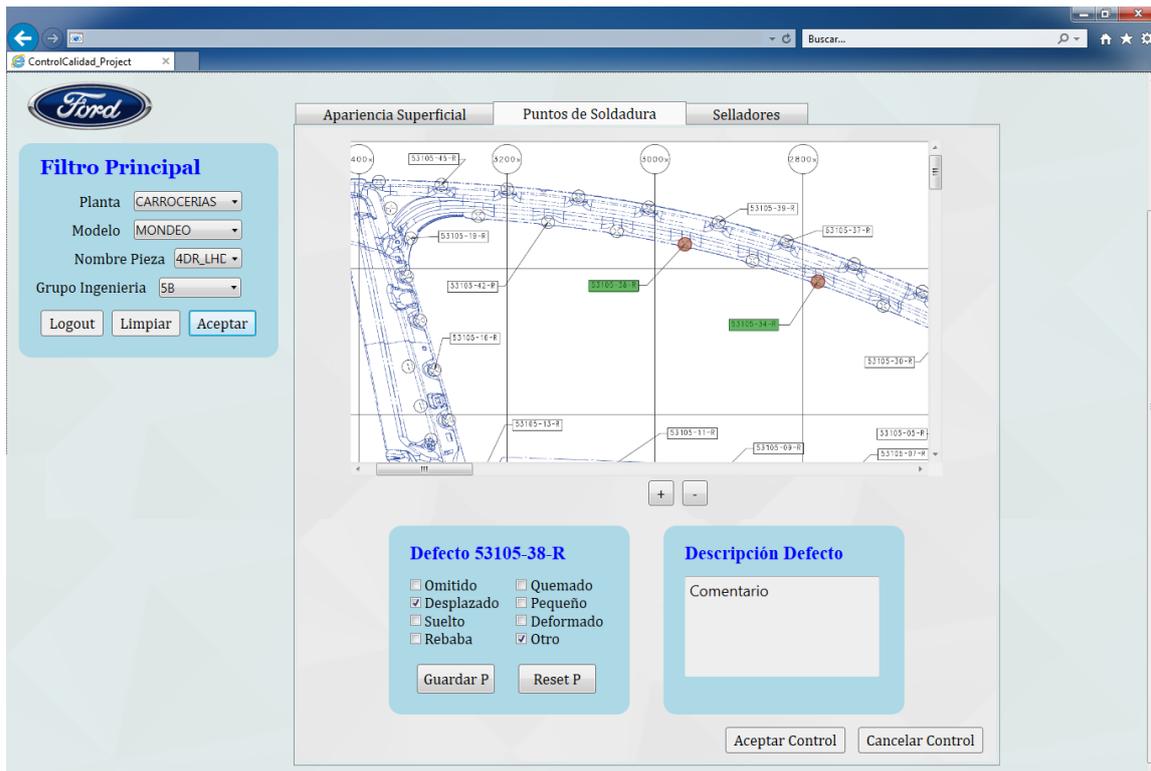


Figura 26: Interfaz resolución 1366x768

- La resolución 1280x768 se muestra en la figura 27.

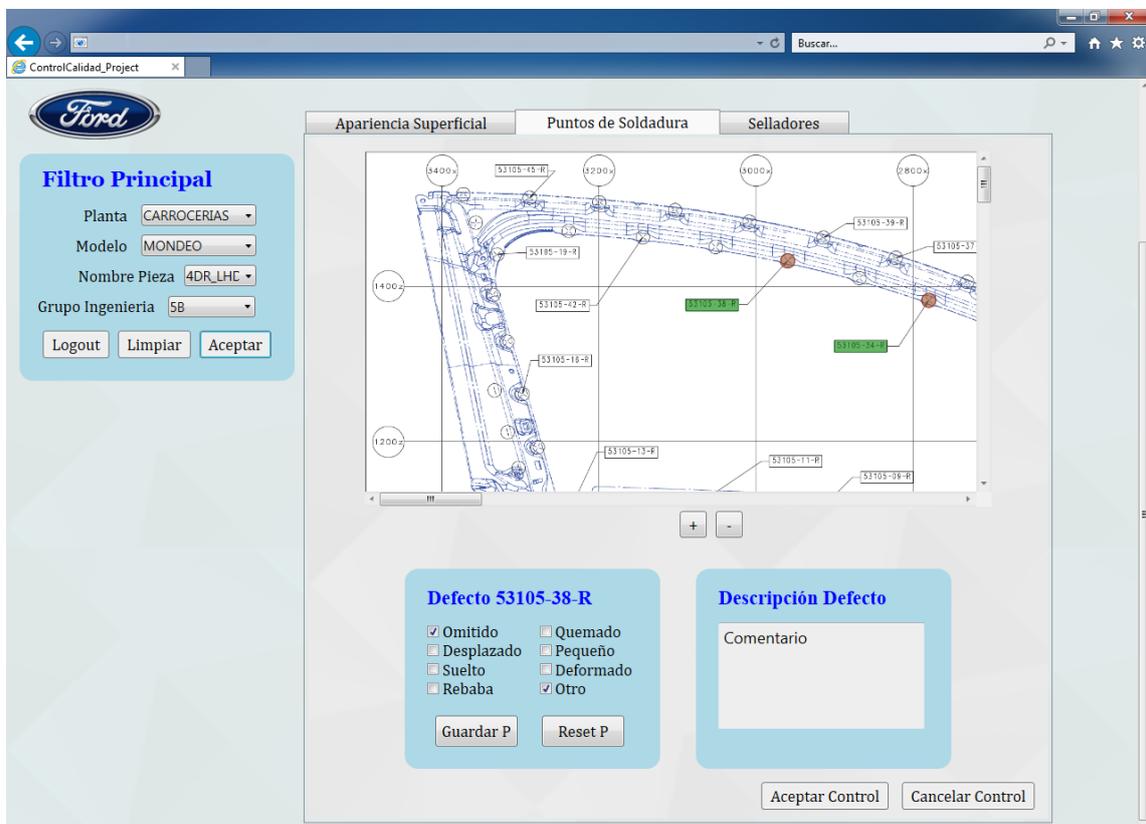


Figura 27: Interfaz resolución 1280x768

- La resolución 800x600 se muestra en la figura 28.

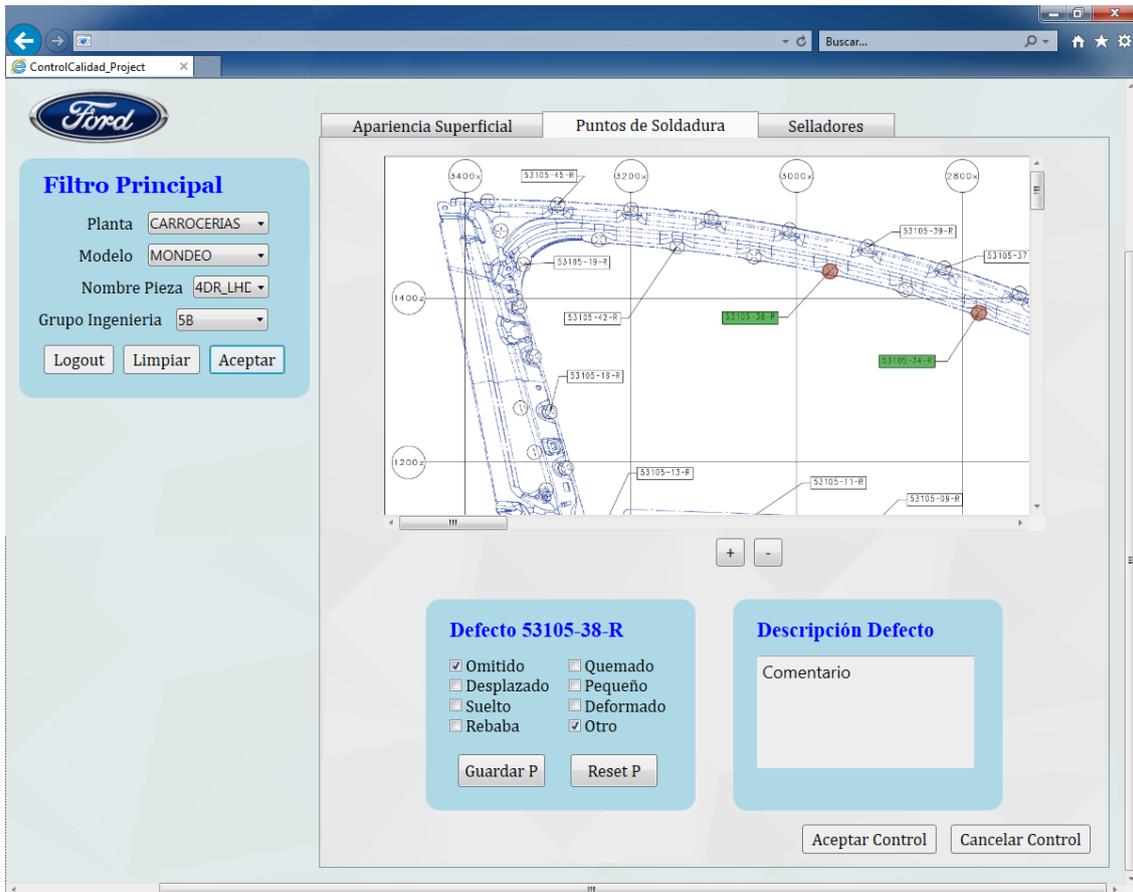


Figura 28: Interfaz resolución 800x600

CAPÍTULO 8

Conclusiones

La finalidad de este proyecto consistía en diseñar para la factoría Almussafes (Valencia) de la multinacional Ford un servicio web para la gestión de controles de calidad en el proceso de producción de un vehículo, en este caso delimitada al área específica de carrocerías.

Tal como se observa en el capítulo de evaluación y pruebas, la aplicación cumple con la especificación de requisitos expuesta en el apartado correspondiente de este documento. Además, se ha diseñado e implementado la base de datos que sirve de soporte a dicho proyecto. Asimismo, el servicio web desarrollado ha tenido en cuenta el propósito de resultar muy intuitivo y sencillo de utilizar ya que va dirigido principalmente para un operario de Ford con un nivel de formación técnica media. Por estas razones, se puede afirmar que se ha cumplido con las expectativas y objetivos descritos en los correspondientes apartados.

Además, en términos generales, se ha cumplido la planificación de trabajo realizada al inicio del proyecto en la estimación de los tiempos dedicados en cada etapa. Sin embargo, la fase de diseño ha exigido más tiempo del estimado en un principio. Esto se ha debido a los altos costes temporales de recopilar los datos y requisitos básicos para el diseño de la base de datos relacional necesaria como soporte de la aplicación y, especialmente, al gran tiempo invertido por Ford en evaluar y conceder los permisos de acceso a dichos datos. No obstante, las etapas de análisis de requisitos y casos de uso se han desarrollado con mayor celeridad de lo esperado.

En cuanto al apartado técnico, debo decir que esta ha sido la primera vez que he trabajado en Visual Studio 2015 como entorno de desarrollo y desconocía los lenguajes XAML, C# y .NET, y nunca antes había realizado un proyecto de desarrollo de software completo y de esta envergadura. Por estos motivos, necesité un periodo de adaptación y preparación específica para el adecuado desarrollo de este proyecto. Por otra parte, una vez superado dicho periodo, conseguí proceder con el trabajo con claridad, determinación y rapidez. Además de haber adquirido nuevos conocimientos y una buena dosis de confianza en mí misma, he averiguado que me apasionan realmente los estudios a los que me he estado dedicando estos años en el Grado y he descubierto con sorpresa mi interés incluso por ramas distintas a mi especialización en ingeniería de computadores.

El desarrollo de este proyecto ha supuesto todo un reto para mí. Para poder llevarlo a cabo me he tenido que apoyar, como es lógico, en muchas de las asignaturas estudiadas en el Grado y durante mi estancia como Erasmus en Polonia: para planificar el tiempo de

cada fase del proyecto he aplicado los conocimientos adquiridos en Gestión de proyectos; al desarrollar la base de datos relacional, necesité los conocimientos aprendidos en la asignatura de Base de datos y sistemas de la información; al programar el código de la aplicación, recurrí a todo lo aprendido en las diversas asignaturas de programación, especialmente en la asignatura de Estructuras de datos y algoritmos, que me resultó de gran utilidad para optimizar el código; por último, la experiencia Erasmus me proporcionó soltura y experiencia en el desarrollo del código de aplicaciones, además de que, desde un punto de vista emocional, me proporcionó los recursos para no desesperar en los momentos de bloqueo o incertidumbre y conseguir salir adelante y superarlos. Considero este hecho muy positivo, puesto que me he visto a mí misma sobrepasando los que consideraba mis límites y he descubierto que, a pesar de las adversidades, con esta convicción y esta determinación, puedo conseguir todo lo que me proponga.

Por último, como el resultado final de la aplicación web ha sido calificado por la empresa como muy satisfactorio, si Ford considerase oportuno en un futuro desarrollarlo por completo y expandirlo a todo el sistema de control de calidad de la factoría, simplemente deberían tenerse en cuenta unos pocos aspectos: primero, deberían añadirse al programa algunos cambios básicos, como distinguir diferentes tipos de usuario en el acceso a la aplicación, tales como: administrador; *process coach* o encargado; *team leader* o jefe de equipo de la línea, entre otros. Además, se deberían implementar otras funcionalidades, que permitiesen, entre otras: el volcado de los resultados de los controles de calidad en archivos de texto y la consiguiente creación de informes de control de calidad; a los encargados, o jefes, la realización de consultas a la base de datos y de informes de consultas, así como firmarlos digitalmente y validarlos online para culminar el proceso de control de calidad informatizado. Una vez informatizado todo el proceso de control de calidad de todas las piezas y modelos de vehículos, se facilitarían las auditorías y otros posibles estudios de mejora del proceso de producción a partir de los datos almacenados (mediante análisis gráficos, estadísticos, etc).

Bibliografía

- [1] Base de datos.
Consultado el 13 de julio de 2018.
<http://www.maestrosdelweb.com/que-son-las-bases-de-datos/>.
- [2] Luc Gervais - Sébastien Putier.
Lenguaje C#.
Ediciones ENI & Sons, 2016.
- [3] LDAP.
Consultado el 15 de junio de 2018.
<http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-rg-es-4/ch-ldap.html>.
- [4] Descripción SQL.
Consultado el 20 de julio de 2018.
https://geotalleres.readthedocs.io/es/latest/conceptos-sql/conceptos_sql.html.
- [5] Especificación de Requisitos según el estándar de IEEE 830 (2008).
Consultado el 19 de julio de 2018.
<https://www.fdi.ucm.es/profesor/gmendez/docs/is0809/ieee830.pdf>.
- [6] Plantilla casos de uso.
Consultado el 14 de agosto de 2018.
[users.dsic.upv.es/asignaturas/facultad/lasi/trabajos/032000\(2\).doc](users.dsic.upv.es/asignaturas/facultad/lasi/trabajos/032000(2).doc).
- [7] Descripción lenguaje XAML.
Consultado el 2 de junio de 2018.
<http://www.blackwasp.co.uk/XAML.aspx>.
- [8] Descripción lenguaje C#.
Consultado el 25 de mayo de 2018.
<http://www.larevistainformatica.com/C1.htm>.
- [9] Descripción .NET.
Consultado el 16 de junio de 2018.
https://es.wikipedia.org/wiki/Microsoft_.NET.
- [10] Bruce Johnson.
Profesional Visual Studio 2015.
John Wiley & Sons, 2015.
- [11] Internet Explorer.
Consultado el 17 de agosto de 2018.
<https://www.techopedia.com/definition/277/internet-explorer-ie>.

- [12] LaTeX.
Consultado el 28 de agosto de 2018.
<https://en.wikipedia.org/wiki/LaTeX>.
- [13] *Internet Information Services* (IIS).
Consultado el 25 de agosto de 2018.
https://es.m.wikipedia.org/wiki/Internet_Information_Services.
- [14] StarUML online.
Consultado el 21 de agosto de 2018.
<http://staruml.io/>.
- [15] Lenguaje UML.
Consultado el 21 de agosto de 2018.
https://en.wikipedia.org/wiki/Unified_Modeling_Language.
- [16] Ejemplo Arquitectura 3 capas.
Consultado el 22 de agosto de 2018.
<http://ltuttini.blogspot.com/2010/08/n-tier-desarrollo-en-capas-ejemplo.html>.