



**Universitat Autònoma
de Barcelona**

Gestión de un Club de Pádel

Memoria del proyecto
De Ingeniería Técnica en
Informática de Gestión
realizada por
José Sánchez Roldán
y dirigida por
Yolanda Benítez Fernández

Escuela Universitaria de Informática

Sabadell, Julio de 2013

La firmante, ***Yolanda Benítez Fernández***,
profesora de l'Escuela de Ingeniería de la UAB,

CERTIFICA:

Que el trabajo que corresponde a la presente
memoria ha sido realizado bajo la dirección de

José Sánchez Roldán

y para que conste firma la presente.

Sabadell, ***Octubre*** de ***2013***

Firmado: ***Yolanda Benítez Fernández***

RESUMEN DEL PROYECTO DE FINAL DE CARRERA DE LA ESCUELA DE INGENIERIA INFORMÁTICA

Titulo del proyecto: Gestión de un club de pádel con control de acceso RFID.	
Autor: José Sánchez Roldán.	Data: <i>Octubre 2013</i>
Tutora: Yolanda Benítez Fernández.	
Titulación: Ingeniería Técnica en Informática de Gestión.	
Palabras clave: <ul style="list-style-type: none">• Català: Anàlisi, Programació, RFID, Software, BIOMAX, Control de Acceso, VB.NET, SQL Server• Castellà: Analisis, Programación, RFID , Software, BIOMAX, Control d' Acces, VB.NET, SQL Server.• Anglès: Analysis, Programing, RFID , Software, BIOMAX, Acces Control, VB.NET, SQL Server	
Resumen del proyecto: <ul style="list-style-type: none">• Català: Aquest projecte és un estudi que pretén realitzar una aplicació dirigida a centres esportius, implementant a través de Biomax l' accés a les instal · lacions utilitzant targetes de proximitat. A més l'aplicació permetrà gestionar de forma eficient als seus socis i instal · lacions. D'aquesta manera s'aconsegueix automatitzar - millorant en temps i qualitat - una tasca imprescindible de fer, control d'accessos de forma senzilla i amb un manteniment fàcil per part dels usuaris que la utilitzen. Per a això s'utilitzarà un dispositiu Kimaldi que ens permetrà gestionar el control d'accessos.• Castellà: Este proyecto es un estudio que pretende realizar una aplicación dirigida a centros deportivos, implementando a través de BIOMAX el acceso a las instalaciones utilizando tarjetas de proximidad. Además la aplicación permitirá gestionar de forma eficiente a sus socios e instalaciones.	

De esta manera se consigue automatizar - mejorando en tiempo y calidad -una tarea imprescindible de realizar, control de accesos de forma sencilla y con un mantenimiento fácil por parte de los usuarios que la utilicen.

Para ello se utilizará un dispositivo KIMALDI que nos permitirá gestionar el control de accesos.

- **Anglès:** This project is a study that seeks to make an application aimed at sports centers BIOMAX implemented through access to facilities using proximity cards. Also, the application will efficiently manage its partners and facilities.

This will get automated - better in time and quality-an essential task to perform, access control in a simple and easy maintenance by the users who use it.

This will work with KIMALDI device that allow us to manage access control.

Índice de contenidos

Capítulo 1 INTRODUCCIÓN	9
1. Presentación.....	10
1.1 Motivaciones personales	10
1.2 Estructura de la memoria.....	10
Capítulo 2 ESTUDIO DE VIABILIDAD	13
2. Estudio de viabilidad.....	14
2.1 Introducción.....	14
2.1.1 Presentación del proyecto.....	14
2.2 Descripción de la situación actual.....	14
2.2.1 Descripción de la situación a tratar.....	14
2.2.2 Perfil de Cliente/Usuario.....	15
2.2.3 Objetivos.....	16
2.3 Descripción del sistema a realizar.....	16
2.3.1 Descripción del proyecto.....	16
2.3.2 Recursos técnicos.....	18
2.3.3 Recursos económicos.....	18
2.3.4 Recursos humanos.....	19
2.3.5 Evaluación del riesgo.....	19
2.3.6 Organización del Proyecto.....	21
2.3.7 Alternativas de Desarrollo.....	21
2.3.8 Futuras Ampliaciones.....	21
2.4 Planificación del proyecto respecto al tiempo.....	22
2.5 Diagrama de Grant.....	24
2.6 Conclusiones.....	25
Capítulo 3 Plan de proyecto	27
3.1 Introducción.....	28
3.2 Estructura de Descomposición del Trabajo.....	28
3.2.1 Fases y actividades del proyecto.....	28
3.2.2 Diagrama WBS.....	29
3.3 Análisis de recursos.....	30
3.3.1 Recursos del proyecto.....	30
3.4 Calendario del proyecto.....	30
3.4.1 Calendario general.....	31
3.4.2 Dependencias.....	31
3.4.3 Cuadro de tareas del proyecto.....	31
3.4.4 Calendario temporal.....	32
3.5 Evaluación de riesgos.....	32
3.5.1 Lista de riesgos.....	33
3.5.2 Plan de contingencia.....	33
3.6 Presupuesto.....	34
Capítulo 4 Análisis de Requerimientos	35

4.1	Introducción.....	36
4.2	Funcionalidad.....	37
4.3	Requerimientos Funcionales.....	37
4.4	Requerimientos NO Funcionales.....	39
4.4.1	Restricciones de rendimiento.....	39
4.4.2	Restricciones de diseño.....	39
4.4.3	Objetivos de diseño.....	40
4.4.4	Descripción del entorno.....	40
	Capítulo 5 Fundamentos Teóricos.....	41
5.1	Introducción.....	42
5.2	Servidor.....	42
5.3	Lenguajes de programación y SQL Server 2005.....	43
5.4	Bases de datos relacionales (BDR) y lenguajes SQL	43
5.4.1	Conceptos Básicos.....	43
5.4.2	Reglas de integridad del modelo relacional.....	45
5.4.3	Introducción al Structure Query Language (SQL).....	45
5.5	Terminal biométrico Kimaldi BioMax2 FP.....	48
5.5.1	Tipos de autenticación.....	49
5.5.2	Ventajas de BIOMAX2... ..	49
5.5.3	Control de Acceso.....	50
5.5.4	Emulador de teclado RFID	52
5.5.5	Conclusión respecto al tipo de acceso Proximidad VS Biometrico.....	52
	Capítulo 6 Diseño del Proeyecto y Funcionamiento.....	53
6.1	Introducción.....	54
6.2	Esquema Base de datos Relacional.....	54
6.3	Interfacez gráficas.....	55
6.3.1	Pantallas de la aplicación.....	56
	Capítulo 7 Tests y pruebas.....	61
7.1	Introducción.....	62
7.2	Fase de pruebas por parte del programador.....	62
7.2.1	De unidad.....	62
7.2.2	De integradidad.....	63
7.2.3	De regresión.....	63
7.3	Fase de pruebas por parte del usuario.....	63
7.3.1	De aceptación.....	64
7.3.2	Del sistema.....	64
7.4	Ejemplos de pruebas realizadas.....	64
	Capítulo 8 Conclusiones.....	67
8.1	Conclusiones.....	68
8.2	Desviación sobre la planificación.....	69
8.3	Futuras ampliaciones del proyecto.....	70
	Capítulo 9 Definiciones, acrónimos y abreviaciones.....	71
	Capítulo 10 Bibliografía.....	73

Anexos Manual de usuario y instalación. (Adjuntos con la memoria).

Índice de figuras

Figura 2.1. Partes interesadas.....	15
Figura 2.2 Esquema sistema a realizar.....	17
Figura 2.3 Tabla de Fases/Actividades/hora.....	23
Figura 2.4 Diagrama de Grantt del Estudio de Viabilidad.....	24
Figura 3.1 Tabla de fases y actividades del proyecto.....	28
Figura 3.2 Diagrama WBS.....	29
Figura 3.3 Cuadro de tareas del proyecto.....	31
Figura 3.4 Diagrama de Gantt , fases y actividades del Plan de proyecto.....	32
Figura 3.5 Tabla de riesgos.....	33
Figura 3.6 Tabla plan de contingencia.....	33
Figura 3.7 Tabla de presupuesto final.....	34
Figura 5.1 Esquema servidor.....	42
Figura 5.2 Dispositivo BIOMAX.....	48
Figura 5.3 Dispositivo BIOMAX2 circuito y integrado en torno.....	48
Figura 5.4 Targetas de identificación de Profimidad.....	50
Figura 5.5 Targetas de identificación con chip.....	51
Figura 5.6 Huellas dactilares.....	51
Figura 5.7 Dispositivo emulador de teclado RFID.....	57
Figura 6.1 Esquema base de datos relacional.....	54
Figura 6.2 Formulario de Mantenimiento de Punto de Venta.....	57
Figura 6.3 Formulario de Mantenimiento de Reservas.....	58
Figura 6.4 Formulario Control de Accesos.....	59
Figura 8.1 Cuadro de desviación en el tiempo de las fases del proyecto.....	69

Capítulo 1

INTRODUCCIÓN

1.- Presentación

El proyecto final de carrera (PFC) [def] se desarrolla para un club de pádel, estas instalaciones estan recientemente en auge y son cada vez mas solicitadas por los consumidores.

En vista de esta situación, las instalaciones deportivas se ven en la obligación de disponer de una aplicación que les permita controlar el acceso a sus instalaciones de una forma rápida y eficiente, además de una gestión correcta de sus instalaciones a través de un sistema de reservas.

1.1 Motivaciones personales

La principal motivación del proyecto viene dada por el autoaprendizaje de nuevas tecnologías y lenguajes de programación, la llegada de nuevos dispositivos que revolucionan la informática, dispositivos que permiten desarrollar aplicaciones que permiten gestionar nuestros recursos de forma mas rápida y eficaz.

Siempre confié en crear una aplicación que permitiera gestionar algun tipo de empresa, pero no ha sido hasta el actual curso académico (2012-2013) que he dispuesto de tiempo suficiente para involucrarme totalmente en este proyecto y llevarlo a cabo.

Realizar este PFC no solo significa un logro sino que también representa un título importante a nivel personal.

1.2 Estructura de la memoria.

La presente memoria está organizada en diferentes capítulos en función de cada uno de los pasos que componen el proyecto desde su definición hasta la presentación de un producto final.

Capítulo 1: Presentación: Se presenta el proyecto y describe a grandes rasgos cuál es el alcance y los objetivos del mismo. Se detallan las motivaciones que han dado

Capítulo 2: Estudio de viabilidad: Se analiza la situación actual del sistema y se definen los objetivos y los requisitos del mismo, metodología de trabajo, a quien va dirigido, recursos económicos, recursos humanos. En base a toda esta información, se proponen una serie de alternativas para el desarrollo del proyecto; se escogerá una de ellas como solución óptima y se determinará si el proyecto es viable o no lo es.

Capítulo 3: Plan de proyecto: Se definirá la estructura del proyecto, que fases y actividades contempla, recursos necesarios y una evaluación de riesgos. Además se elaborará un presupuesto.

Capítulo 4: Análisis de Requerimientos: Este apartado define las funcionalidades, requisitos funcionales y no funcionales, así como las restricciones necesarias para la elaboración del proyecto.

Capítulo 5: Fundamentos Teóricos: Este apartado define los aspectos teóricos en que se basa la aplicación, servidores, lenguajes de programación, funcionamiento de la base de datos y el aspecto central de la aplicación, como es el dispositivo de control de accesos BIOMAX2.

Capítulo 6: Diseño del Proyecto y Funcionamiento: Se muestran el el diseño de la interfaz y funcionamiento de los módulos.

Capítulo 7: Diseño Tests y pruebas: Elaboración de las pruebas para verificar el correcto funcionamiento.

Capítulo 8: Conclusiones: Conclusiones finales una vez hemos desarrollado nuestro proyecto final de carrera.

Capítulo 9: Definiciones, acrónimos y abreviaciones: Lista al detalle de las definiciones, acrónimos y abreviaciones.

Capítulo 10: Bibliografía: Especifica la bibliografía necesaria así como las páginas de internet necesarias para recabar la información necesaria.

Capítulo 11: Anexos: Anexos con los diferentes manuales de instalación y manejo.

Capítulo 2

Estudio de

Viabilidad

2. Estudio de viabilidad

2.1 Introducción

2.1.1 Presentación del Proyecto

El objetivo de este apartado es demostrar la viabilidad del proyecto de fin de carrera. Se realizará el estudio del mismo evaluando los requisitos y necesidades que este ha de cumplir para llevarlo a cabo con éxito, beneficios y mejoras que en el caso de poderse aplicar, este aportará.

El proyecto que se presenta puede ser útil para llevar a cabo la gestión de un Club de Pádel, de tal forma que le permita tener un control exhaustivo de sus clientes, de la reserva de sus instalaciones por parte de los usuarios y sobretodo un control eficaz de su base de datos. A estos aspectos habría que destacar el acceso a las instalaciones.

A continuación presentaremos los objetivos principales que el PFC ha de conseguir, el método de planificación a utilizar y su planificación en función del tiempo con fechas orientativas para poder llevarlo a cabo. Describiremos los perfiles de usuario que utilizaran la aplicación, las partes interesadas del mismo, los recursos utilizados, riesgos que nos podríamos encontrar durante la elaboración del proyecto así como alternativas de desarrollo. Una vez realizado el estudio de viabilidad, se analizará toda la información para decidir si finalmente es viable o no ejecución del PFC.

2.2 Descripción de la situación actual

2.2.1 Descripción de la situación a tratar

Actualmente hay una gran variedad de empresas que se dedican a la salud y al cuidado de las personas, muchas de ellas facilitando actividades físicas deportivas, desde gimnasios, clubs de tenis, clubs de fútbol, etc.

El proyecto a desarrollar pretende proporcionar un software que permita la gestión de un club de Pádel, el cual gestione el mantenimiento de todos sus empleados y clientes.

El Club de Pádel necesita llevar una gestión eficiente sobre la reservas de sus pistas, que clientes son los que la han reservado y en qué día y hora esta realizada dicha reserva. Se pretende con esto optimizar al máximo los recursos y tener las pistas con el mayor porcentaje de ocupación posible. Por otra banda es interesante tener un control de accesos eficaz, que permita el acceso a las instalaciones a tanto a clientes como a otros usuarios como son sus empleados. En el caso de empleados este acceso permitiría saber al departamento de administración del Club las horas efectivas que sus empleados trabajan en las instalaciones. Se pretende entonces desarrollar una aplicación que pueda recoger toda esta información, ordenarla, actualizarla, mostrarla i gestionarla de forma sencilla y eficaz.

2.2.2 Perfil del cliente/usuario

El perfil de usuario que accederá a la aplicación será de dos tipos, por una parte *ADMINISTRADOR* ya que según el tipo de información ha de ser gestionada por un responsable, además de ser el encargado de gestionar los diferentes perfiles otorgados al resto de usuarios.

Otro perfil de usuario será el *USUARIO/RECEPCIONISTA*, que será aquel que utilizará constantemente la aplicación para gestionar directamente con los clientes el buen funcionamiento de las instalaciones, el cual le permitirá acceder a diferentes mantenimientos.

Partes interesadas	Acceso
Administrador	Acceso total a la aplicación
	Mantenimiento de empleados
	Total control de la aplicación
	Visualizar todo tipo de listados
Usuario/Recepcionista	Mantenimiento de clientes
	Mantenimiento de pistas
	Visualizar listados con acceso
	Gestión punto de venta
Cliente	Acceso a instalaciones mediante tarjeta
	Solicitud de pistas
	Opción de realizar compras

Figura 2.1 Partes interesadas.

2.2.3 Objetivos

A continuación se detallan los objetivos del proyectos, los cuales van a permitir disponer de toda la información en una misma aplicación con una Base de Datos Relacional.

OB 1.- Disponer de toda la información del Club de Pádel en una Base de Datos Relacional dentro de un servidor dedicado.

OB 2.- Mantenimiento y actualización periódica de la aplicación.

OB3.- Modificación, altas, bajas de los Empleados por parte del Administrador.

OB4.- Disponer de un control de acceso a las instalaciones, a través de tarjeta de proximidad.

OB5.- Disponer de un listado donde se lleve un control de los días de trabajo de cada empleado, así como otros tipos de listado.

OB6.- Creación de un pequeño punto de venta de artículos asociados a la actividad realizada, en este caso pádel.

OB7.- Mantenimiento de reservas.

OB8.- Gestión de las diferentes entidades que se manejaran en la aplicación, como son EMPLEADOS, SOCIOS, PISTAS.

2.3 Descripción del sistema a realizar

2.3.1 Descripción del proyecto

Como se ha comentado anteriormente el software ha de permitir gestión del Club de Pádel en un entorno intuitivo y fácil de usar, así como un acceso rápido a los menús.

La aplicación ha de permitir acceso desde cualquier punto de la red. La información de la base de datos estaría en un servidor y por otro lado tendríamos los diferentes terminales que accederían a ella a través de la aplicación.

La empresa dispone de una estructura de informática básica con tres ordenadores clientes conectados a un servidor. El directorio de datos compartido del servidor, que es accesible por todos los clientes.

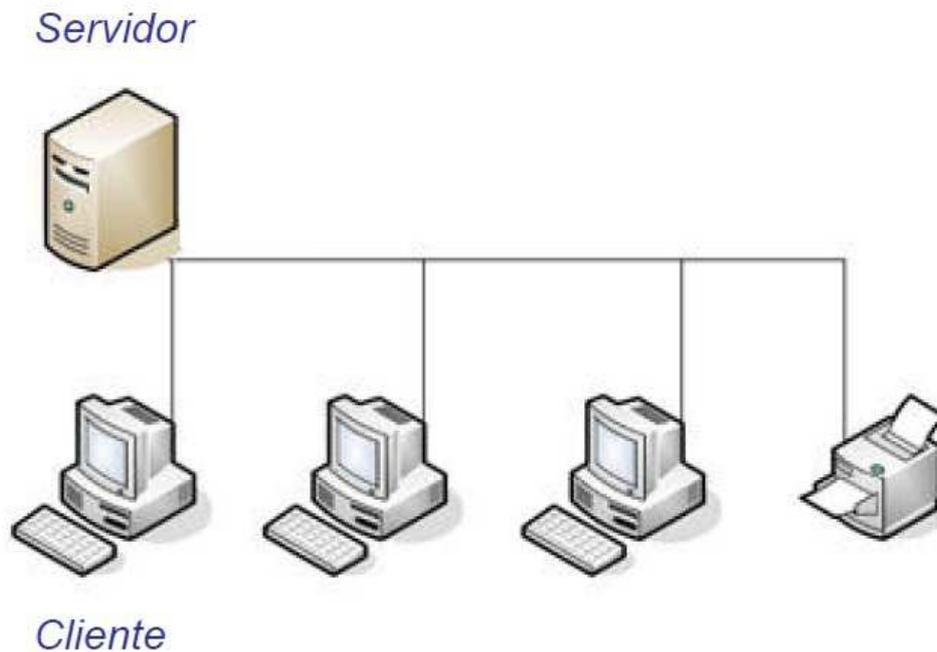


Figura 2.2 Esquema sistema a realizar.

El software a desarrollar ha de contemplar una serie de contenidos para alcanzar los diferentes objetivos.

1. El Administrador ha de poder gestionar todos los aspectos de la aplicación, el cual accederá mediante su contraseña, será el único que podrá crear *usuarios, tipos de acceso, puertas y más administradores.*
2. El perfil de USUARIO/RECEPCIONISTA gestionará básicamente el *mantenimiento de los socios, la reserva de pistas y la venta de artículos,* el cual se tendrá que validar también previamente con su contraseña.
3. Se ha de poder visualizar el estado de las pistas libres, reservadas, así como los diferentes tipos de listados.

Cada trabajador o usuario de las instalaciones ha de disponer de una tarjeta de acceso de proximidad para poder acceder al centro.

2.3.2 Recursos técnicos

Hardware:

Necesitaremos un PC desde el que se desarrollará la aplicación:

- Mínimo Procesador: Pentium IV a 1 GHz
- Memoria RAM: 1 GB
- Disco Duro: 250 GB
- CD-RW
- Monitor SVGA
- Terminal de Control de Acceso BIOMAX2
- Simulador de Teclado.

Software:

- Sistema Operativo: Windows 7
- Microsoft office para a la realización de la memoria y otros documentos.
- Lenguaje de programación: VISUAL BASIC.NET

2.3.3 Recursos económicos

En este apartado se mostraran los costes que tiene desarrollar el proyecto. Lo más costoso son los recursos técnicos, los cuales veremos su coste a continuación.

Por otro lado también utilizamos lenguaje de programación VISUAL BASIC.NET y una base de datos SQL SERVER.

Además de los recursos anteriores hay que destacar las horas invertidas del estudiante, que hará de analista y programador de la aplicación, así como el desarrollo de la memoria, tiempo estimado del cual será unas **312 horas** aproximadamente. Horas realizadas exclusivamente por el estudiante.

Las horas invertidas se analizaran más detalladamente en el capítulo siguiente. PLANIFICACION DEL PROYECTO RESPECTO AL TIEMPO.

Para el cálculo de las licencias y PC se tendrán en cuenta los costes de amortización.

Llegado a este punto podemos concluir que el coste del proyecto, iniciado desde cero, realizado por un universitario que cobrara unos 15 €/h sería de:

2.3.4 Recursos humanos

Además de los recursos técnicos y económicos, también son muy importantes las personas encargadas del análisis y desarrollo del proyecto.

- **Analista:** Persona encargada de definir las necesidades i viabilidad del proyecto, es el enlace entre cliente y proyecto.
- **Diseñador:** Es la persona encargada de la estructura, forma, color tipo de fuente, etc. De todo lo relacionado con la aplicación.
- **Programador:** Una vez realizada la estructura, será este quien realizará toda la aplicación, enlace con base de datos, etc.
- **Equipo de pruebas:** Tiene como función probar y testear todas las opciones de la aplicación para detectar posibles errores en la programación o posibles mejoras que se puedan realizar, este informará a los responsables para que lo puedan solucionar. Una vez se haya comprobado su buen funcionamiento estará lista para entregar al cliente final.

En este caso, al tratarse de un estudio universitario, todos estos aspectos serán llevados a cabo por una única persona, el estudiante.

2.3.5 Evaluación del Riesgo

En este apartado trataremos los diferentes riesgos que podrían aparecer en la elaboración del proyecto.

- **Elección adecuada del método de acceso:** Para la elaboración del método de acceso a las instalaciones se disponen de varios tipos de acceso, como por ejemplo:

1. **Acceso biométrico:** *[def]* El acceso es mediante huella dactilar, permite al usuario de las instalaciones llevar siempre consigo su clave, la cual no puede extraviarse u olvidarse, además es difícil de robar. La validación biométrica es una buena opción, pero hay ocasiones en que los lectores fallan o su tiempo de respuesta es tardío ya que no todas las personas disponen de una huella dactilar nítida, como por ejemplo personas más mayores o simplemente aquellas personas que desempeñan trabajos en los que el desgaste de sus manos es mayor.
2. **Acceso de proximidad RFID:** *[def]* Este tipo de acceso es mediante una tarjeta, la cual solo ha de acercarse al terminal instalado junto al torno, tras su posterior validación dará acceso a las instalaciones a las que tenga permiso.

Planteados los dos casos de acceso más comunes actualmente, se opta por la elección de **acceso de proximidad**, ya que evitamos los problemas que surgen en las huellas, y la reposición de una tarjeta en caso de pérdida es baja. Por otro lado el acceso biométrico se descarta ya que en ocasiones la lectura de la huella en según qué personas da problemas de validación, como es en el caso de personas mayores o personas que puedan presentar alguna marca o herida en la zona de lectura.

- **Seguridad:** Respecto a la seguridad el problema planteado no es tan alto respecto a aplicaciones que trabajen con datos vía web, ya que trabaja a modo interno sin conexión al exterior, pero si es necesario gestionar copias de seguridad para evitar posibles pérdidas de datos. Esta tarea puede ser determinada según la política de Backup's de cada empresa.

2.3.6 Organización del Proyecto

La aplicación a desarrollar ha de pasar por una serie de pasos para realizarlo correctamente, para este proyecto el modelo a utilizar en su elaboración es el de ciclo de vida secuencial, la organización del cual es lineal en todas sus fases. Un proyectos con las fases tan marcadas es difícil de encontrar, pero el inicio i el final de sus fases han de estar bien definidas. La información generada en una fase es importante para la siguiente, es por eso que se escoge este tipo modelo.

2.3.7 Alternativas de Desarrollo

Es cierto que se han podido escoger otro tipo de lenguajes, pero hemos escogido VISUAL BASIC.NET porque es un lenguaje de programación fácil de manejar, robusto y el que permite realizar una aplicación robusta y fiable.

Respecto a la base de datos se podrían haber escogido bases datos como por ejemplo Mysql [*def*] pero optamos por SQLServer [*def*] ya que el funcionamiento es muy estable y funciona con solidez para aplicaciones en red como la que vamos a crear.

También se podría elaborar una Web en la que los clientes pudieran reservar sus las pistas de Pádel vía Online, pero esta ya sería en todo caso una ampliación a petición del cliente.

2.3.8 Futuras Ampliaciones

La aplicación estará diseñada en principio para un club de pádel, pero se podría ampliar si el cliente decidiera ampliar sus servicios, por ejemplo si este decidiera crear unas actividades o dispusiera de otro tipo de instalaciones a las pistas de Pádel.

Por otro lado una futura ampliación podría ser la creación de una Web del Club de Pádel, la cual tenga acceso a la base de datos y permita la reserva de pistas por parte de los socios vía online.

2.4 Planificación del proyecto respecto al tiempo

Para la elaboración del proyecto se han tener en cuenta las fases que este dispone y cuanto tiempo se dedicara a cada una de ellas.

- **Iniciación:** Esta fase es la primera de todas, en la que definimos el proyecto inicial, procedemos a la asignación con el tutor y matriculación del mismo.
- **Planificación:** Esta fase nos determinará si el proyecto es viable o no. Para conocer su viabilidad hemos de conocer la situación actual de trabajo, las ventajas e inconvenientes que puede tener y el cliente final a quien va dirigido. Una vez realizado esto es importante la planificación que se hará de los contenidos, hacer una valoración de todos los recursos que se utilizaran y el coste de los mismos.
- **Análisis y diseño:** En esta fase se analizaran los aspectos funcionales y no funcionales, para especificar al máximo los contenidos del proyecto. Se diseñara la base de datos y el programa en sí.
- **Codificación:** Fase en la que preparamos en entorno de desarrollo y confeccionamos el código.
- **Pruebas:** Fase dedicada a testear la aplicación y verificar su funcionamiento.
- **Memoria/Documentación:** Elaboración documentada del proyecto.
- **Finalización/Entrega:** Última fase del proyecto en la que se entrega el trabajo realizado durante el año y se expone ante un tribunal evaluador.

Fases	Actividades	Horas / Actividades	Horas / Fase
Iniciación	Definición	2 h	4 h
	Asignación	1 h	
	Matriculación	1 h	
Planificación	Estudio de Viabilidad	25 h	37 h
	Aprovisionamiento del EV	1 h	
	Plan de Proyecto	11 h	
Análisis y diseño	Análisis de requisitos	25 h	45 h
	Análisis y diseño de base de datos	20 h	
Codificación	Habilitar entorno de desarrollo	2 h	180 h
	Programar código	178 h	
Pruebas	Diseño tests de integración	2 h	20 h
	Desarrollo de pruebas	18 h	
Memoria/Documentación		25 h	25 h
Finalización/Entrega		1 h	1 h
Total horas			312 h

Figura 2.3 Tabla de Fases/Actividades/hora.

2.5 Diagrama de Grantt [def]

A continuación se muestra el diagrama de GRANT según su planificación en el tiempo.

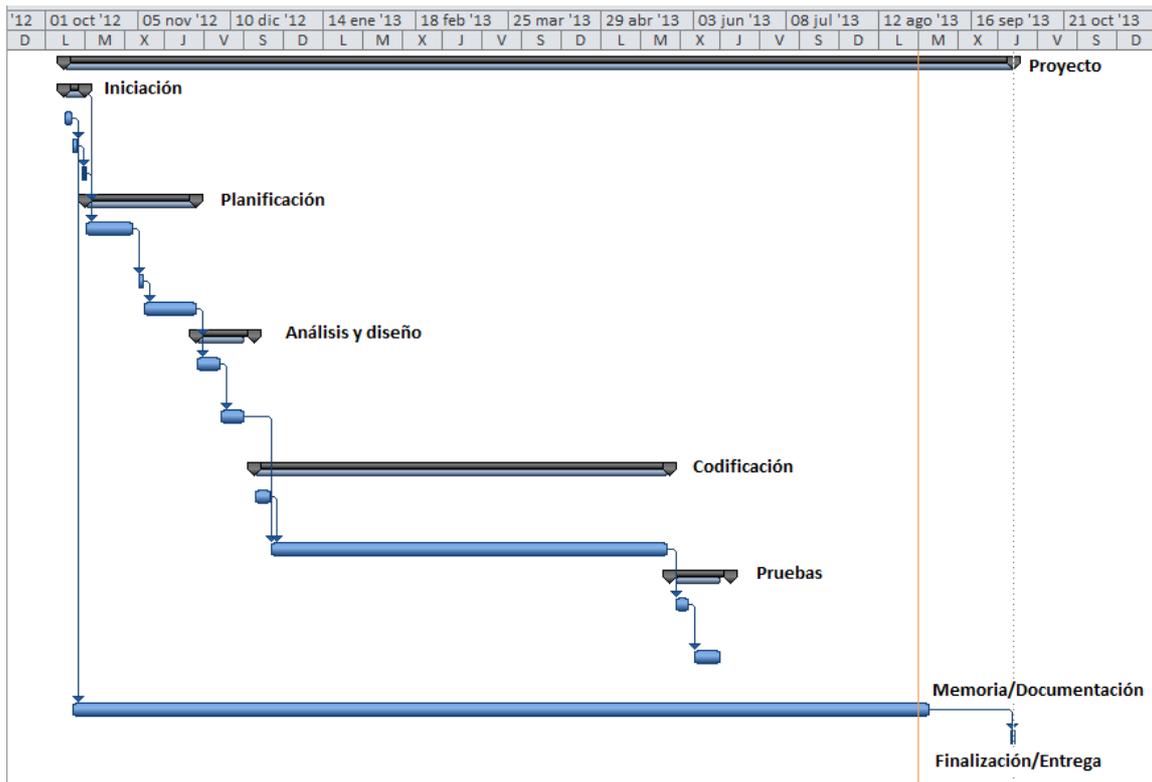


Figura 2.4 Diagrama de Grantt del Estudio de Viabilidad.

2.6 Conclusiones

Una vez realizado este estudio de viabilidad podemos deducir que:

La aplicación permitirá disponer de toda la información de la empresa con una mayor organización y accesibilidad a las personas encargadas de su gestión.

El coste de la aplicación no es muy elevado ya las horas a remunerar son de un estudiante, que es quien se encarga de todas las fases del proyecto.

El desarrollo del proyecto no es de gran duración y puede ser puesto en funcionamiento en un medio/corto plazo.

La instalación de este nuevo software permite al CLUB DE PÁDEL gestionar de forma más eficiente sus instalaciones.

Por todo lo expuesto anteriormente podemos afirmar que el desarrollo de este **proyecto es viable.**

Capítulo 3

Plan de Proyecto

3.1 Introducción

La instalación de este nuevo software permite al CLUB DE PÁDEL gestionar de forma más eficiente sus instalaciones. En este capítulo se detallan las fases de las que se compone el proyecto y se establece un calendario para el mismo, quedando el proyecto de esta manera enmarcado temporalmente mediante una fecha de inicio y final así como unas fechas intermedias que sirven como puntos de control. Se evalúan además cuáles son los posibles riesgos por los que el proyecto se puede ver afectado y se establece el correspondiente plan de contingencia. Es necesario también determinar un presupuesto exacto de los costes que supone llevar a cabo el proyecto para finalmente poder acabar formalizando una serie de conclusiones.

3.2 Estructura de Descomposición del Trabajo

La planificación que se expone a continuación se ha realizado mediante el software de desarrollo y control de proyectos Microsoft Project 2010 [def]. Para la elaboración del plan de proyecto se ha utilizado una metodología lineal, es decir, cada una de las fases que compone el proyecto debe estar finalizada antes de comenzar la siguiente.

3.2.1 Fases y actividades del proyecto

Fase	Descripción
Iniciación	Actividades de definición del proyecto, asignación de tutor y matriculación.
Planificación	Estudio de viabilidad, aprobación y plan del proyecto.
Análisis y diseño	Análisis de requisitos funcionales y no funcionales y arquitectura del sistema; incluyendo además el diseño de la implementación.
Codificación	Desarrollo del código de la aplicación.
Pruebas	Se valida el desarrollo mediante diferentes tipos de tests.
Memoria/Documentación	Realización de manual de usuario y memoria del proyecto.
Finalización/Entrega	El director del proyecto firma la aceptación y cierre del proyecto. Posteriormente, se defiende el proyecto delante de la comisión.

Figura 3.1 Tabla de fases y actividades del proyecto.

3.2.2 Diagrama WBS (Work Breakdown Structure) [def]

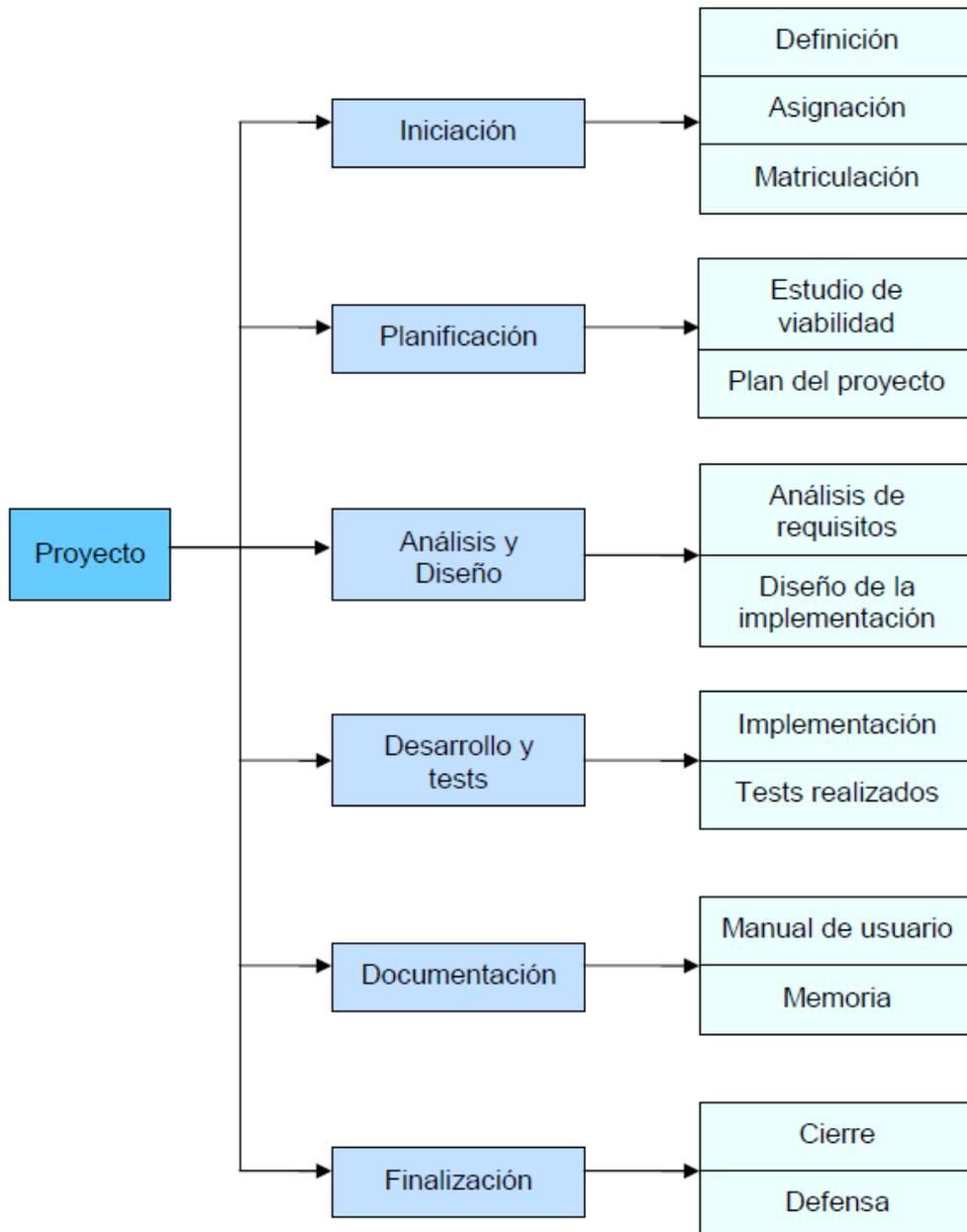


Figura 3.2 Diagrama WBS.

3.3 Análisis de recursos

Se estudian los recursos humanos y materiales que requiere el proyecto para cada una de las fases del mismo.

3.3.1 Recursos del proyecto

Recursos humanos:

Total = 312 horas * 15€/hora = 3780 €

Recursos materiales:

Biomax2 = 7550 €

Emulador de teclado = 120 €

PC = 50 € (lo que corresponde a la amortización)

Total = 920€

Visual Basic.NET = 50 €

SQL Server 2010 Express = 0 €

Microsoft Office(word/Visual Project) = 25 €

Windows 7 = 150 €

Total = 275€

Para el cálculo de las licencias y PC se tendrán en cuenta los costes de amortización.

TOTAL COSTE APLICACIÓN: 3780 € + 920€ + 225€ = 4925 €

3.4 Calendario del proyecto

A continuación se explica el calendario general del proyecto en donde se fijan el total de horas a dedicar y se tienen en cuenta las relaciones y los recursos asignados a cada una de las actividades del proyecto.

3.4.1 Calendario general

- **Fecha de inicio:** Octubre 2012
- **Fecha Finalización:** Septiembre 2013
- **Total de horas:** 312 horas

3.4.2 Dependencias

Todas las fases se desarrollan utilizando un modelo lineal. Por lo tanto, cada fase no empieza hasta que se ha finalizado la anterior.

3.4.3 Cuadro de tareas del proyecto

Nº	Descripción de la actividad	Duración	Predecesor
1	Iniciación	4 h	
2	Definición	2 h	1
3	Asignación	1 h	2
4	Matriculación	1 h	3
5	Planificación	37 h	
6	Estudio de Viabilidad	25 h	4
7	Aprobación del EV	1 h	6
8	Plan de Proyecto	11 h	7
9	Análisis y diseño	45 h	
10	Análisis de requisitos	25 h	8
11	Análisis y diseño de base de datos	20 h	10
12	Codificación	180 h	
13	Habilitar entorno de desarrollo	2 h	11
14	Programar código	178 h	13
15	Pruebas	20 h	
16	Diseño tests de integración	2 h	14
17	Desarrollo de pruebas	18 h	16
18	Memoria/Documentación	25 h	
19	Finalización/Entrega	1 h	

Figura 3.3 Cuadro de tareas del proyecto.

3.4.4 Calendario temporal

A continuación se muestra el *diagrama de Gantt* del proyecto, según el diagrama se muestra el periodo previsto de iniciación y finalización.

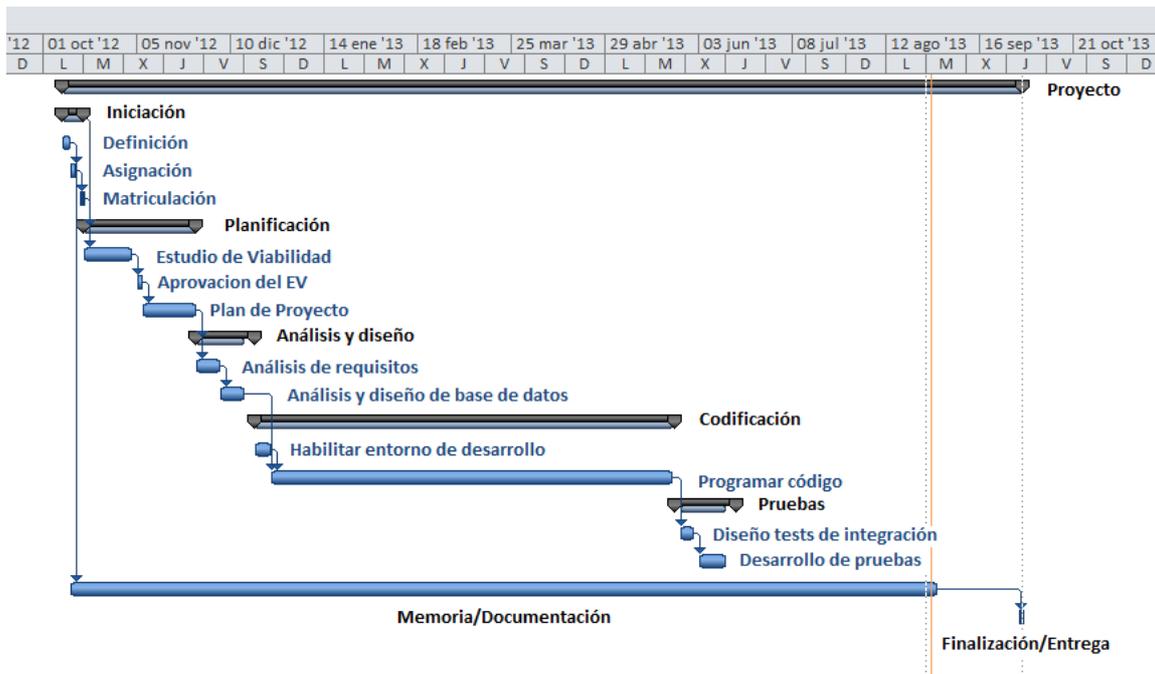


Figura 3.4 Diagrama de Gantt , fases y actividades del Plan de proyecto.

3.5 Evaluación de riesgos

Este apartado se crea con el objetivo de listar los posibles riesgos por los que el proyecto se puede ver afectado, previendo el impacto de los mismos sobre el proyecto. Se establece también un plan de coo caso de que el riesgo sucediera realmente.

3.5.1 Lista de riesgos

	Riesgo	Descripción	Consecuencias sobre el proyecto
R1	Planificación temporal optimista	El proyecto no se finaliza en los plazos previstos	Aumento de recursos y mayor presupuesto
R2	Falta de alguna tarea	Proyecto mal definido	Objetivos no cumplidos
R3	Presupuesto poco ajustado	Plan de proyecto mal definido	Menor calidad y posibles pérdidas
R4	Cambio de requisitos	Estudio de viabilidad y Análisis mal definidos	Retraso en los plazos de entrega
R5	Herramientas de desarrollo inadecuadas	Elección de Software erróneo	Retraso en los plazos de entrega
R6	Incumplimiento de legislación	Estudio de viabilidad mal definido	Incumplimiento de objetivos
R7	Abandono del proyecto	El alumno no finaliza el proyecto	Perdida de tiempo y dinero, decepción.

Figura 3.5 Tabla de riesgos.

3.5.2 Plan de contingencia

Riesgo	Solución
R1	Aplazar funcionalidades, entregar otras convocatorias
R2	Revisar plan de proyecto, modificar planificación
R3	Renegociar, afrontar pérdidas
R4	Renegociar, modificación planificación y presupuesto
R5	Prever herramientas alternativas
R6	Revisar normas y legislación, consultar a expertos
R7	Sin solución

Figura 3.6 Tabla plan de contingencia.

3.6 Presupuesto

Este apartado detalla el presupuesto final del proyecto, desglosando los diferentes tipos de costes.

Coste total del proyecto	Precio
Recursos personales	3.780 €
Dispositivos Hardware	920 €
Licencias	275 €
Coste Total	4.975 €

Figura 3.7 Tabla de presupuesto final.

Capítulo 4

Análisis de Requerimientos

4.1. Introducción

Uno de los primeros pasos del proceso de fabricación de software es el análisis de requerimientos. La función de esta actividad es comprender el problema que el software debe solucionar, la parte más importante se centra en identificar qué es lo que debe hacer el sistema, es decir, que necesidades debe cubrir. Hay que obtener todos los requisitos que la aplicación debe cumplir y las restricciones de diseño (por razones de seguridad, económicas, etc.).

El análisis de requerimientos consta de dos etapas:

- **Comprensión del problema:** El analista debe entender el sistema existente y las partes que se han de automatizar. Además debe tener muy claro qué datos se utilizan en el sistema y las diferentes entradas y salidas, ya que esta etapa es la base del proyecto.
- **Especificación de los requerimientos:** En este documento debe haber toda la información para resolver el problema planteado. Además debe comprobar que los requerimientos especificados a lo largo del proyecto cumplan todas las expectativas y reflejen las necesidades del proyecto.

Como primera instancia, constará de un apartado de funcionalidad del *Software* [def] donde se detallarán los diferentes apartados del que consta la aplicación. A continuación se realizará una descripción del entorno, así como una explicación de los objetivos del sistema, la especificación de los requerimientos funcionales y los no funcionales.

4.2. Funcionalidad

El propósito general que justifica el desarrollo de este proyecto es gestionar un centro deportivo en la mayoría de sus aspectos, intentando minimizar la gestión manual por parte del usuario, de una manera fácil e intuitiva.

GESPADEL es el nombre de la aplicación que corresponde a este proyecto. Se trata de una aplicación desarrollada básicamente en programación VB.NET bajo una base de datos SQL Server.

Es por ello que este apartado analizará las funciones que realiza la aplicación y así conseguir un buen funcionamiento de la misma.

4.3. Requerimientos Funcionales

Los requerimientos funcionales son aquellos que nos dan a conocer el comportamiento que se desea del software, controlando las posibles anomalías que se puedan producir:

- **Eficiencia:** LA aplicación tiene que estar muy bien estructurada, tiene que ofrecer todo aquello que el realmente el cliente necesita.
- **Tiempo de respuesta:** Este ha de ser el menor tiempo posible y constante por parte del servidor, la aplicación tiene que minimizar estos tiempos, para ello es necesario un un buen diseño de la misma.
- **Seguridad:** Es uno de los temas más importantes de cualquier aplicación donde haya datos que puedan comprometer la seguridad informática de una empresa, ya que no se puede permitir que cualquier usuario vea los datos. El acceso a la aplicación estará controlado mediante un login y un password. Se hace para implementar cierta seguridad para las bases de datos y evitar así cualquier intrusión que pueda hacer peligrar la seguridad de los servidores o de la red.

- **Facilidad:** EL uso de la aplicación debe ser sencilla e intuitiva, ha de informar al usuario donde se encuentra en cada momento y que hacer en caso de duda. Para ello es una buena gestion de los posibles errores que puedan surgir.
- **Calidad:** Una vez se han realizado una série de pruebas que se han pasado satisfactoriamente podemos considerar que el software es de buena calidad.

A continuación describiremos los requisitos que tendrá que cumplir cada perfil de usuario:

Administrador :

- Consultar, modificar, añadir o eliminar usuarios/información tanto de usuarios del sistema como de los socios de las instalaciones.
- Control de acceso de todos los usuarios de la aplicación.
- Recuperar cualquier tipo de información.

Usuario/Recepcionista:

- Consultar y modificar datos de la aplicación, por lo que respecta a los diferentes mantenimientos de la aplicación.
- Restriccion a items del menú como Usuarios.
- Gestión del punto de venta.
- Elaboración de informes.

Socio:

El socio no interactura con la aplicación, **simplemente hace uso de su targeta de identificación.**

4.4. Requerimientos NO Funcionales

Los requerimientos no funcionales son aquellas restricciones impuestas por el cliente por el problema en si y que afectan al diseño, estos acostumbran a ser cuantificables. Estos requerimientos se pueden clasificar en:

4.4.1 Restricciones de rendimiento

- **Restricciones estáticas:** Son aquellas restricciones que no dependen del entorno, como por ejemplo, la capacidad máxima de usuarios que pueden atacar la base de datos al mismo tiempo depende del sistema gestor de base de datos. En nuestro caso no será un problema puesto que el número sera muy bajo.
- **Restricciones dinamicas:** A diferencia de las anteriores estas si dependen del entorno, hemos de tener en cuenta las peticiones que se realizan al mismo tiempo en la base de datos, esto influye negativamente, pero al no depender de una conexión via internet, los accesos continuan siendo mínimos.

4.4.2 Restricciones de diseño

Las restricciones de diseño se basan en:

- **Cumplimiento de los standards:** La aplicacion se implementara bajo VB.Net, el cual es un lenguaje de programación muy potente con una alta fiabilidad asi como compatibilidad contrastada a nivel mundial.
- **Limitaciones de Hardware:** La aplicación necesitará para su funcionamiento un PC con una tarjeta para poder conectarse a la red y acceder a la aplicación. Por otro lado se precisará de un servidor con un mínimo de procesador IV a 1 GHz i 80 GB RAM.
- **Recuperación i fiabilidad antes posibles errores:** Los posibles errores que pueda causar el usuario seran controlados a traves del codigo de la aplicación, evitando los campos nulos o obligatorios en la base de datos.

- **Seguridad:** Para implementar la seguridad de la aplicación, la base de datos estará protegida por password. Para acceder a cualquier contenido de la aplicación el usuario ha de estar registrado y disponer de un login y un password. Se han de realizar copias de seguridad de la base de datos periódicamente.

4.4.3 Objetivos de diseño

En este apartado se comentaran las restricciones externas del proyecto, que afectan al substancialmente al resultado final del mismo.

- **Navegación intuitiva:** El diseño ha realizarse segun los estandars de la mayoría de aplicaciones en este entorno.
- **Posibles ampliaciones:** Se ha de realizar una buena estructura del diseño y una buena codificación del código, esto nos facilitará añadir futuros modulos ampliatorios a la aplicación.

4.4.4 Descripción del entorno

Al tratarse de una aplicación de la cual trabajaran diferentes usuarios se trabajará con un servidor que será el que tendra alojada la base de datos relacional, por tanto cualquier ordenador conectado a la red podrá acceder a la aplicacion mediante el usuario y password. El objetivo será tener centralizada, relacionada y actualizada toda la información referente a las instalaciones deportivas.

Capítulo 5

Fundamentos teóricos

5.1 Introducción

En este capítulo se explicaran todos los conceptos que necesitamos saber, tanto de hardware [def] como de software [def], para poder utilizar y comprender la aplicación, así como saber cual es su propósito.

5.2 Servidor

Un servidor sirve información a los ordenadores que se conecten a él. Cuando los usuarios se conectan a un servidor pueden acceder a programas, archivos y otra información del servidor.

En nuestro caso nos referimos al ordenador físico en el que funciona el software y almacenamos la base de datos, una máquina cuyo propósito es proveer datos de modo que otras máquinas puedan utilizar esos datos.

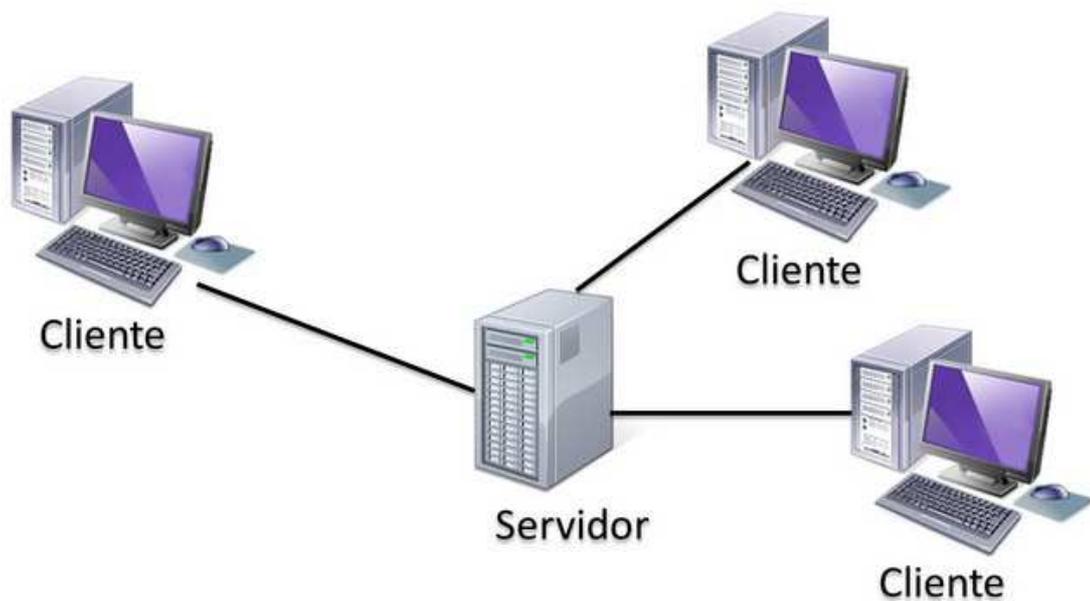


Figura 5.1 Esquema servidor.

5.3 Lenguajes de programación y SQL SERVER 2005

Un lenguaje de programación es un lenguaje formal diseñado para expresar procesos que pueden ser llevados a cabo por máquinas como las computadoras. Pueden usarse para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación humana.

Está formado por un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones.

- **VB.NET**

Visual Basic .NET (VB.NET) es un lenguaje de programación orientado a objetos que se puede considerar una evolución de Visual Basic. Este lenguaje dispone de significativos cambios respecto a su antecesor que lo hacen no sea compatible con el, por otro lado el manejo de las instrucciones es similar pero permite el desarrollo de aplicaciones más avanzadas.

- **SQL SERVER 2005**

Microsoft SQL Server es un sistema para la gestión de bases de datos producido por Microsoft basado en el modelo relacional

5.4 Bases de datos relacionales (BDR) y lenguajes SQL

En este apartado se explicaran: los conceptos básicos de bases de datos, las reglas de integridad del modelo relacional y se hará un breve resumen sobre Structure Query Language (SQL). [def]

5.4.1 Conceptos Básicos

En este apartado se explican los conceptos básicos para entender qué es y cómo se debe crear una base de datos:

- **Base de datos (BD):** Sistema informático diseñado para almacenar información (agrupación de datos, almacenados sin redundancia), mantenerla, recuperarla cuando se desee y de la forma que se quiera.
- **Base de datos relacional (BDR):** Es una base de datos, los usuarios de la que la ven como una colección de tablas, es decir, es un conjunto finito de información organizada en tablas que están relacionadas entre sí.
- **Tabla de una BDR:** Conjunto de datos organizados en una matriz bidimensional. Cada fila de la matriz es un registro que contiene información sobre una unidad de la tabla. Cada columna de la tabla contiene los diferentes campos de registro, y además los valores deben tener las mismas propiedades y formatos.

Por ejemplo, en la tabla Socios, cada fila contiene información sobre un socio, y esta fila está dividida en varios campos con información sobre el socio: Identificador, nombre, apellido, dirección, teléfono, etc.

Clave Primaria (Primary Key):

En el diseño de bases de datos relacionales, se llama clave primaria a un campo o a una combinación de campos que identifica de forma única a cada fila de una tabla. Una clave primaria comprende de esta manera una columna o conjunto de columnas. No puede haber dos filas en una tabla que tengan la misma clave primaria.

Una clave primaria debe identificar unívocamente a todas las posibles filas de una tabla y no solo a las filas que se encuentran en un momento determinado.

La clave primaria nunca puede tener valor NULL.

Una vez definidas todas las tablas de la base de datos con sus campos y sus claves, se determinará las relaciones entre las tablas. Existen diferentes tipos de relaciones: uno a varios, varios a varios y uno a uno.

Las relaciones uno a varios son las más comunes en las bases de datos relacionales. En este tipo de relaciones, el registro de una tabla puede tener más de un registro en otra tabla.

5.4.2 Reglas de integridad del modelo relacional

La función principal de las reglas de integridad es controlar la coherencia y corrección de los datos. Existen dos tipos:

- **Reglas de integridad específica:** Estas reglas son las que se aplican a los campos de las tablas: Longitud máxima de un campo, caracteres que acepta un determinado campo, etc. Es decir, todas las restricciones que tienen los campos de la base de datos, junto con sus características (longitud, tipo de campo, etc.).

- **Reglas de integridad generales: (Hay de tres tipos)**
 1. **Reglas de integridad de las entidades:** Ningún componente de la clave primaria puede aceptar valores Nulos.

 2. **Reglas de integridad referenciales:** Todos los valores de una clave externa (cuando toda ella es diferente a NULL) deben coincidir con un valor de la clave primaria de otra relación.

 3. **Reglas de integridad de los dominios:** Esta regla marca el formato o contenido de un campo. Todo valor debe pertenecer al dominio sobre el que está definido.

5.4.3 Introducción al Structure Query Language (SQL)

El lenguaje de consulta estructurado (SQL) es un lenguaje declarativo normalizado de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones sobre las mismas.

Una de sus características es el manejo del álgebra y el cálculo relacional permitiendo ejecutar consultas con el fin de recuperar, de una forma sencilla, información de interés de una base de datos, así como también hacer cambios sobre la misma .

El lenguaje SQL se compone de comandos, cláusulas, operadores y funciones de agregado. Estos elementos se combinan en las instrucciones para crear, actualizar y manipular las bases de datos.

Tipos de consultas:

- **Consultas de selección:** Las consultas de selección se utilizan para indicar al motor de datos que devuelva información de las bases de datos. Esta información se devuelve en forma de conjunto de registros. Este conjunto de registros se modificable. La siguiente consulta recupera el nombre y apellido, de un determinado socio de la tabla socios.

```
SELECT nombre, apellido
```

```
FROM socios
```

```
WHERE cp = 08210;
```

Si lo que se desea es recuperar todos los campos de una tabla no hace falta poner el nombre de cada uno de los campos; utiliza el carácter *. ejemplo:

```
SELECT *
```

```
FROM socios
```

```
WHERE cp = 08210;
```

- **Consultas de acción:** Las consultas de acción son aquellas que no devuelven ningún registro, son las encargadas de acciones como añadir, borrar y modificar registros. Hay tres tipos:

1. **DELETE:** Crea una consulta de eliminación, que elimina los registros de una o más tablas listadas en la cláusula FROM que satisfacen la cláusula WHERE. Esta consulta elimina los registros completos, no es posible eliminar el contenido de un campo en concreto mediante este comando. Su sintaxis es:

```
DELETE Tabla. * FROM Tabla WHERE criterio
```

Una vez que se han eliminado los registros utilizando una consulta de borrado, no se puede deshacer la operación.

2. **INSERT INTO:** Añadir un registro a una tabla. Se conoce como una consulta de datos añadidos. Para insertar un único registro se utiliza la sintaxis siguiente:

```
INSERT INTO Tabla (campo1, campo2, ..., campo N)  
VALUES (valor1, valor2, ..., valorN)
```

3. **UPDATE:** Crea una consulta de actualización que cambia los valores de los campos de una tabla específica basándose en un criterio específico. Su sintaxis es:

```
UPDATE Tabla  
  
SET Campo1 = Valor1, Campo2 = Valor2, ...,  
  
WHERE Criterio;
```

UPDATE es especialmente útil cuando se desea cambiar un gran número de registros o cuando éstos se encuentran en múltiples tablas. Puede cambiar varios campos a la vez.

5.5 Terminal biométrico Kimaldi BioMax2 FP

Para la realización del proyecto hemos utilizado un dispositivo llamado BioMAX2 [def] de la empresa KIMALDI [def], este dispositivo permite implementar los módulos de control de acceso para nuestra aplicación Gespadel.

Disponible con lector de proximidad (125 kHz y 13,56 MHz), lector de tarjetas magnéticas, chip, huella digital y combinación de éstos.



Figura 5.2 Dispositivo BIOMAX2.



Figura 5.3 Dispositivo BIOMAX2 circuito y integrado en torno.

5.5.1 Tipos de autenticación

- biométrica de huella digital
- Proximidad
- Banda magnética
- Código de barras
- Password

El terminal se suministra en carcasa o en versión OEM para integrar:

- Carcasa de acero inoxidable
- Carcasa de plástico.
- Versión OEM: los componentes se suministran por separado, ideal para integración en tornos, paneles o maquinaria.
- Acabado para exteriores: caja de acero inoxidable condicionada para uso en exteriores. Para un uso correcto del terminal se recomienda protegerlo mediante techado, sobre todo en terminales biométricos.

5.5.2 Ventajas de BIOMAX2

- Terminal on-line que ofrece la máxima flexibilidad y seguridad: controlado a tiempo real
- Admite múltiples configuraciones de hardware
- Incorpora lector de reconocimiento de huella dactilar FIM5360: sensor biométrico de gran calidad y durabilidad
- Autenticación combinada opcionalmente con proximidad, smart card, banda magnética o código de barras
- Conectividad básica RS-232 y Ethernet UDP, TCP/IP; opcionalmente a Wi-Fi
- Carcasa metálica, de plástico o en versión OEM para su integración
- Cuatro relés y entradas digitales
- Interfaz de usuario con display, teclado, beeper y leds
- Funcionalidades de control de acceso, control de asistencia, control de presencia y control horario

5.5.3 Control de Acceso

Los terminales de **control de acceso** son equipos electrónicos que gestionan de manera automática la apertura segura de puertas, barreras, tornos, verjas y otro tipo de accesos tanto para la entrada de personas como para la entrada de vehículos en parkings u otras áreas restringidas.

Existe una amplia gama de dispositivos de control de acceso con múltiples características diferenciadoras, pero normalmente estos terminales están compuestos por un hardware de control y una interficie con el usuario que en general consta de un teclado, display, indicadores luminosos (leds) y acústicos (beeper), entradas digitales y un lector. Los lectores permiten la autenticación de la persona a partir de una **tarjeta de proximidad (RFID), chip o banda magnética**. En un **control de accesos biométrico** las personas se identifican mediante la huella digital u otra característica física.

Tecnología RFID (Radio Frequency Identification) - Identificación por Radio Frecuencia: lectura y escritura a distancia. Utilice los lectores y grabadores de proximidad para crear soluciones más cómodas y duraderas para sus usuarios. La radiofrecuencia nos permite la transmisión de datos sin contacto físico entre el lector de aproximación y la tarjeta de aproximación.



Figura 5.4 Targetas de identificación de Profimidad.

Chip o banda magnética: Tarjetas plásticas de PVC: con banda magnética, de alta y baja coercitividad y con chip de memoria.

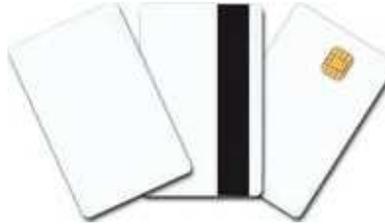


Figura 5.5 Targetas de identificación con chip.

Control de accesos biométrico: El acceso en este sistema es a través de la huella dactilar del usuario, se utiliza el dispositivo BIOMAX2, a diferencia del RFID la transmisión de datos exige un contacto físico entre el dispositivo y el identificador que sería huella dactilar.



Figura 5.6 Huellas dactilares

5.5.4 Emulador de teclado RFID

Dispositivo que nos permite la lectura de tarjetas, este obtiene su identificación interna que posteriormente nos servirá para asignarla a un determinado socio.



Figura 5.7 Dispositivo emulador de teclado RFID

5.5.5 Conclusión respecto al tipo de acceso Proximidad VS Biometrico

Por lo que respecta a estos tipos de acceso ambos son efectivos, pero consideramos que es mas factible el uso de tarjetas de proximidad puesto que en determinados casos la lecturas de las huellas dactilares van en función de cada persona, podemos encontrar personas que tengan problemas a la hora de ser identificadas. Un ejemplo seria una persona que tiene algun corte en el dedo, esto afectaria en la lectura de la huella.

Capítulo 6

Diseño del Proyecto y Funcionamiento

6.1 Introducción

El principal objetivo de esta fase es buscar una solución al problema planteado en la anterior fase, es decir, concretar el cómo realizar las necesidades de los apartados anteriores.

6.2 Esquema Base de Datos Relacional

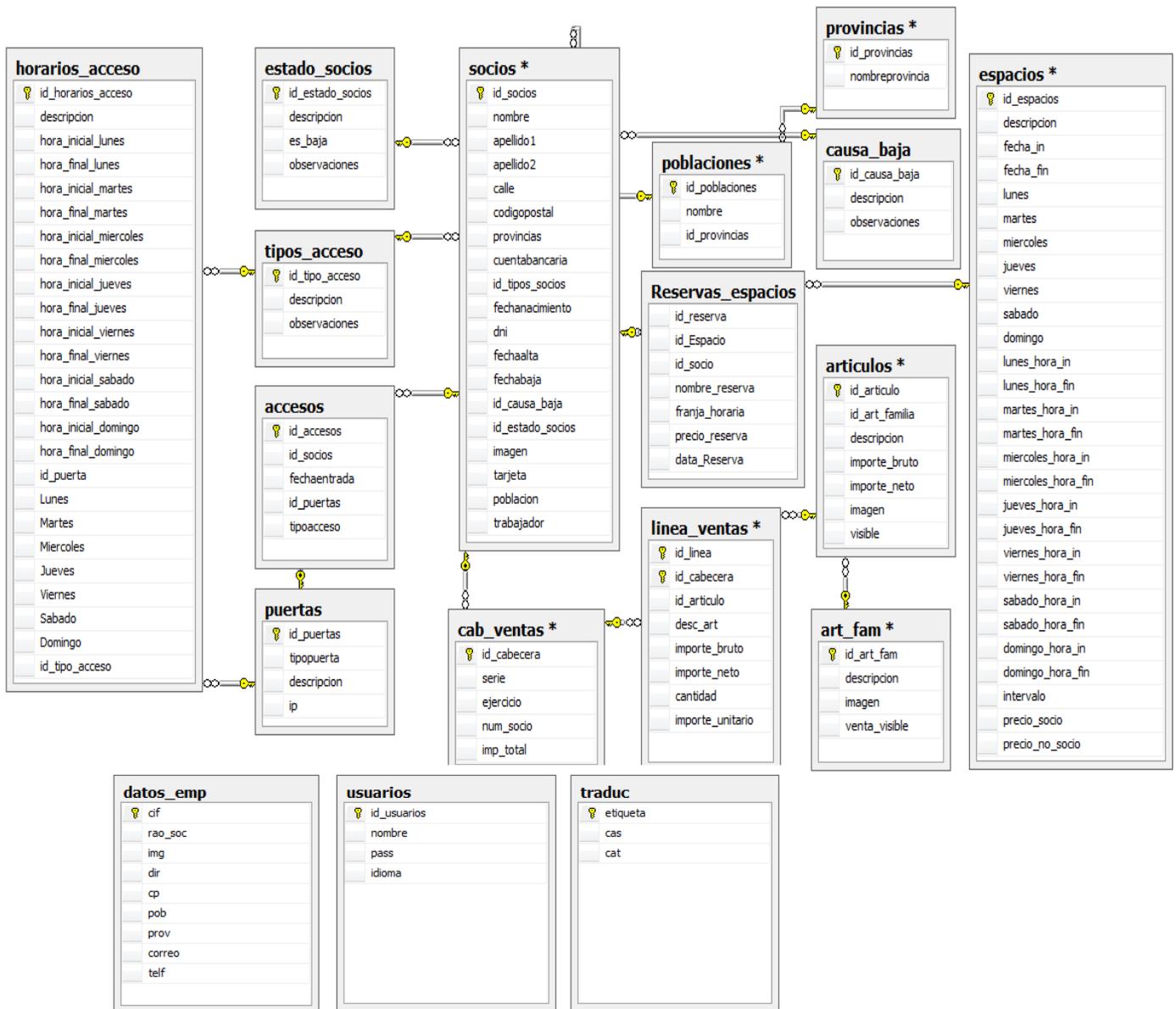


Figura 6.1 Esquema base de datos relacional.

6.3 Interfacez gráficas

A continuación se describirán algunos de los factores más importantes que sirven para mejorar la calidad de la interfaz de usuarios:

- **Facilidad de aprendizaje:** Corresponde al tiempo que los usuarios, en este caso la aplicación, tardarán en utilizar correctamente las diferentes opciones. Es por este motivo, que este es uno de los puntos que se ha tenido más en cuenta, ya que se quiere que sea una herramienta con la que se pueda trabajar inmediatamente y fácilmente.
- **Velocidad de uso:** Corresponde al tiempo que se tarda en realizar una tarea específica, es por ello que se ha intentado crear una aplicación intuitiva y rápida, vigilando mucho el tamaño de las imágenes e intentando aplicar el máximo el uso de estilos.
- **Frecuencia de errores:** Indica el número de errores por tiempo de uso de la aplicación. Así pues, en la aplicación se ha intentado eliminar y controlar todo lo que puede provocar errores por parte del usuario, tal como comprobación de caracteres, numérica, etc.
- **Facilidad de retención:** Indica el grado en que el usuario recuerda la utilización de la aplicación después de no interactuar durante un cierto tiempo. Para reducir este factor, y tal como ya hemos comentado anteriormente, la aplicación se ha realizado de forma que sea intuitiva para los usuarios.

Una vez expuestos los anteriores puntos, se puede observar que puntos como facilidad de aprendizaje o velocidad de uso son contrapuestos a frecuencia de errores, es por eso que se recomienda el uso frecuente, un aprendizaje gradual y no estar mucho tiempo sin utilizar la aplicación.

Los usuarios deben disponer de una buena interfaz para facilitar la interacción con la aplicación, es por eso que los módulos se han diseñado y programado respetando los siguientes puntos:

- **Consistencia:** La aplicación debe seguir una línea de uniformidad para hacer más cómoda la navegación y el aprendizaje al usuario.
- **Terminología:** Procurar que la misma palabra designe la propia acción, para toda la aplicación.
- **Formato:** Mensajes de error claros y siempre con el mismo formato, menús de apariencia similar, etc.
- **Dar Información:** Cada acción realizada por el usuario debe mostrar información sobre la resolución de la misma, ya sea correcta o incorrecta y siempre se pedirá confirmación cuando ésta conlleve la eliminación de registros de la base de datos.
- **Rectificación:** Permitir al usuario volver a modificar los datos que ya han sido introducidas, sin necesidad de volver a introducir todas de nuevo.
- **Estado actual:** El usuario debe saber en todo momento en qué parte de la aplicación se encuentra para evitar confusiones que puedan derivar en posibles errores.
- **Gestión de errores:** Se ha de evitar que el usuario pueda cometer errores que puedan causar problemas en la aplicación, así avisará al usuario cada vez que realice una acción errónea con mensajes, indicando de qué tipo de el error se trata, a fin de poder resolver de la forma más rápida posible.

6.3.1 Pantallas de la aplicación

En este apartado se mostrará la funcionalidad de la aplicación de forma gráfica y se comentará en detalle algunos módulos de la aplicación, para ver al completo el resto de mantenimientos consultar ANEXOS sobre el manual de usuario.

Mantenimiento punto de venta

Este mantenimiento nos permite vender a los clientes artículos que tengamos en venta, los artículos pertenecen a una determinada familia. Al seleccionar una familia se nos muestran los artículos disponibles que tenemos en venta, se puede seleccionar el artículo que pasará directamente a la línea de pedidos. Para ver al detalle el formulario consultar ANEXO de manual de usuario.

Articulo	Importe	id_linea
Barricade Team	100	001
Bebidas	5	005
*		

Figura 6.2 Formulario de Mantenimiento de Punto de Venta

Mantenimiento de Reservas

Este mantenimiento permite al usuario gestionar las reservas de las pistas, permite ver las que están disponibles para una fecha en concreto y realizar la reserva para un determinado socio, también se contempla el hacer la reserva para una persona que no sea socio o quiera disfrutar de las instalaciones. Para ver al detalle el formulario consultar ANEXO de manual de usuario.

pista 1	pista 2	pista
09:00 A 10:00	09:00 A 10:00	09:00 A 10:00
10:00 A 11:00	10:00 A 11:00	10:00 A 11:00
11:00 A 12:00	11:00 A 12:00	11:00 A 12:00
12:00 A 13:00	12:00 A 13:00	12:00 A 13:00
	13:00 A 14:00	13:00 A 14:00

Figura 6.3 Formulario de Mantenimiento de Reservas

Control de Acceso

El formulario de **Control de Accesos** es el más importante de la aplicación, este gestiona los accesos, el usuario ha de tener siempre activo este formulario, en el qual se podrá visualizar cada vez que el socio pasa su tarjeta de identificación, si esta dentro de su horario se pondra de color azul, mostrará su foto y dará acceso al socio, de no ser asi se pondrá en rojo y mostrará el motivo por el que no puede entrar, posiblemente esta fuera de horario.

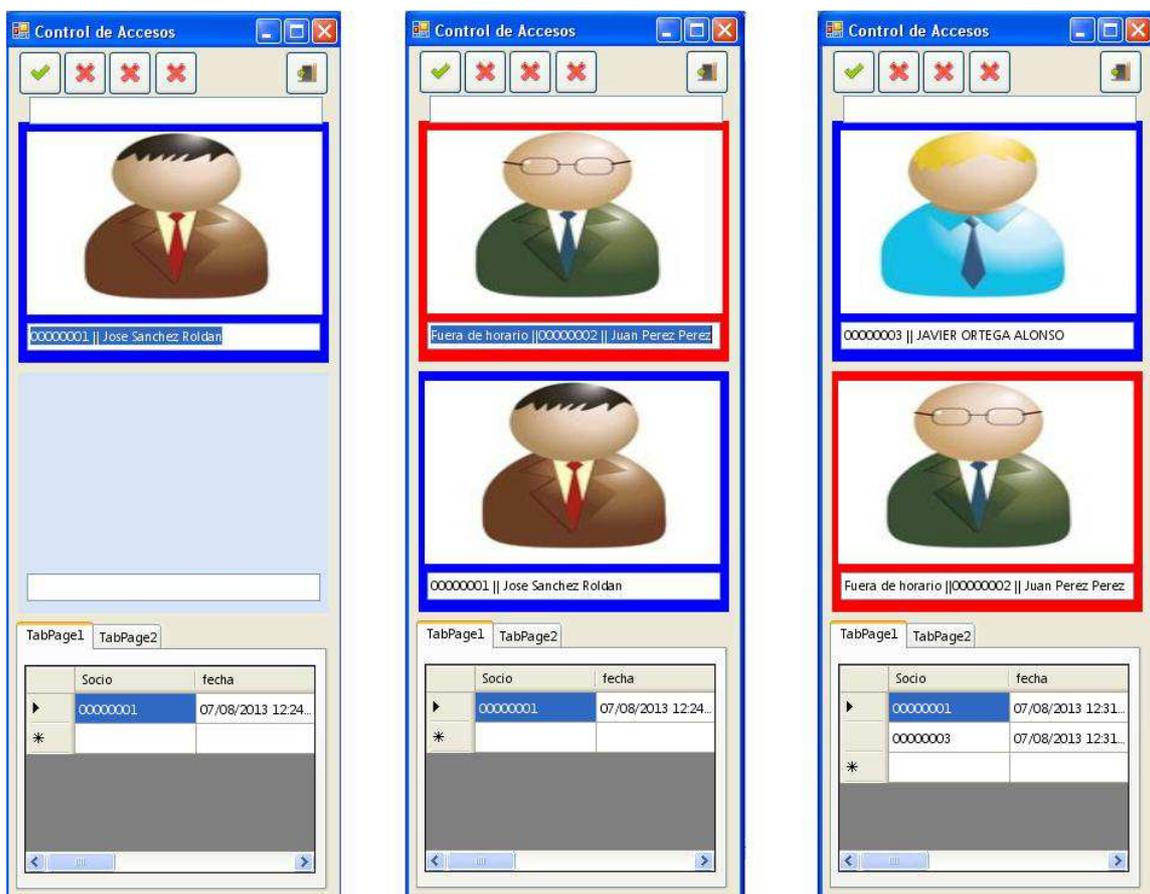


Figura 6.4 Formulario Control de Accesos

Capítulo 7

Tests y Pruebas

7.1 Introducción

En el desarrollo de cualquier software debe realizar una serie de pruebas para descubrir todos los errores posibles, aunque se considera imposible desarrollar un software con una tasa de fiabilidad del 100%.

Una vez realizada toda la aplicación se han de llevar a cabo una serie de pruebas a diferentes niveles de la misma.

Los tests se realizan durante el desarrollo de la aplicación así como una vez finalizada la misma; pero es al final cuando están todos los módulos implementados que se intenta buscar algún posible error ya que la interacción entre ellos es más compleja. Estos aseguran que el aplicativo responde de manera correcta a una serie de resultados esperados. Si no es así, se busca una solución con el fin de presentar una aplicación totalmente funcional y coherente al análisis de requisitos realizado con anterioridad.

7.2 Fase de pruebas por parte del programador

En esta fase, el programador debe realizar las siguientes pruebas:

- De unidad
- De integridad
- De regresión

7.2.1 De unidad

Este tipo de pruebas son fundamentales cada vez que se modifica un módulo. De esta manera se asegura que las pruebas de integridad resulten exitosas.

Se han realizado las pruebas necesarias sobre todas y cada una de las diferentes opciones que componen la aplicación.

Estas pruebas se han ido haciendo a medida que se realizaba la codificación, para tener mayor facilidad para modificar y corregir los resultados negativos de la prueba de unidad.

7.2.2 De integridad

Una vez realizadas todas las pruebas de unidad, y haber solucionado los errores que han ido surgiendo, se han realizado las pruebas de integridad.

Estas, consisten en detectar los errores que aparecen cuando interactúan los diferentes módulos entre sí. De esta forma detectamos los errores cometidos con respecto al diseño de la aplicación.

7.2.3 De regresión

Al modificar partes de código para reparar o mejorar el software se pueden provocar nuevos errores, es por eso que la prueba de regresión consisten volver a realizar todas las pruebas realizadas hasta el momento, para comprobar si los cambios son correctos y detectar posibles nuevos errores

7.3 Fase de pruebas por parte del usuario

Esta fase la realiza el usuario final, pero con el apoyo del programador-analista que ha realizado la aplicación.

Se han realizado diferentes tipos de pruebas:

- **De aceptación**
- **Del sistema**

7.3.1 De aceptación

Una vez realizado el proyecto, se debe verificar que la aplicación hace lo que se había planteado en los requerimientos iniciales. Así pues, para llevar a cabo estas pruebas lo más usual es que el usuario final verifique el software o dejarlo probar a terceros (Beta-Testers).

Esta fase dura hasta que el cliente queda satisfecho con la aplicación, aceptando que todos los compromisos adquiridos durante la fase de análisis de requerimientos se han conseguido, y en muchos casos la funcionalidad solicitada en un principio ha sido superada por las características de la aplicación.

7.3.2 Del sistema

Consiste en ejecutar la aplicación en un entorno de trabajo real. Así pues se pone en marcha la aplicación estando en permanente vigilancia durante cierto tiempo para detectar los posibles errores del software.

7.4 Ejemplos de pruebas realizadas

En este caso todas las pruebas han sido realizadas por el estudiante, tanto a nivel de programación como de usuario final.

Se ha comprobado:

- La validación correcta de los usuarios de la aplicación, y verificar que el acceso a los módulos sea el correcto según los privilegios.
- Las modificaciones en la base de datos desde cualquier mantenimiento sean correctas, que los datos volcados sean totalmente fiables.
- La interacción entre los diferentes módulos funcione, que los datos manejados entre diferentes formularios sean coherentes y no existan duplicidades.

- Se ha comprobado la creación de reservas, tanto de socios como de clientes eventuales, y sobretodo se ha verificado que el programa no permita eliminar pistas con reservas activas.
- Comprobación en el control de accesos, que la lectura por parte del BIOMAX2 sea correcta asi como la verificación en la base de datos. Comprovar que la conexión con el terminal sea xorrecta a traves de su IP.
- La creación de articulos, asi como se sus familias se vean reflejadas en el punto de venta y la venta se ejecute correctamente.
- A nivel de usuario se han hecho simulacros en todos los mantenimientos, añadiendo, borrando y editando nuevos registros, controlando los posibles fallos que un usuario que no conoce la aplicación puede cometer, informando con mensajes de error el motivo para que este pueda hacer funcionar la aplicación correctamente.
- Respecto a la vinculación de la targeta RFID con cada socio se han hecho las comprobaciones de registro adecuadas, ya que un primer lugar el RFID devuelve valores hexadecimales los cuales se han de tratar como decimales ya que el BIOMAX2 trabaja en decimal.

Capítulo 8

Conclusiones

8.1 Conclusiones

Este es el momento de poner sobre la mesa el trabajo realizado en los últimos últimos meses y valorar las horas que he dedicado a su planificación, diseño y desarrollo:

- **Objetivos iniciales conseguidos:** Puedo asegurar que se han conseguido todos y cada uno de los objetivos, tanto los planteados al principio como durante el desarrollo de la aplicación.
- **Aprendizaje:** Destacar el aprendizaje sobre controles de accesos gracias a los dispositivos BIOMAX2, que permiten una amplia variedad de utilidades a desarrollar que de bien he podido comprobarson muy útiles en gran variedad de aplicaciones.
- **Aplicación de conocimientos:** urante la carrera, así como los conocimientos que obtuve tambien el CFGS (Ciclo Formativo Superior de Aplicaciones Informaticas).
- **Constante aprendizaje:** El mundo de Internet y de la Informática en general, está evolucionando continuamente, por lo que un programador de aplicaciones debe estar al día de las últimas novedades y aprender los nuevos lenguajes y tecnologías que vayan surgiendo. El autoaprendizaje suele ser una técnica de aprendizaje muy habitual.
- **Constante mejora del proyecto:** Debido a la aparición de nuevas tecnologías, el proyecto puede estar en constante cambio, aplicando las tecnologías que irán surgiendo en los próximos tiempos. Es posible que un más adelante surgan nuevos requisitos que la aplicación tenga que contemplar o simplemente la utilización de sistema de control de accesos más avanzados.

8.2 Desviación sobre la planificación

Fases previstas y reales

El siguiente cuadro muestra la desviación en el tiempo de algunas fases, la planificación en el tiempo de la codificación se alargó algo más de lo previsto, al haber desarrollado el proyecto de forma secuencial provocó un retraso en la siguiente fase, pero los plazos de entrega se han podido cumplir correctamente.

Nº	Descripción de la actividad	Duración	Predecesor	En plazo	Fuera plazo
1	Iniciación	4 h		v	
2	Definición	2 h	1	v	
3	Asignación	1 h	2	v	
4	Matriculación	1 h	3	v	
5	Planificación	37 h		v	
6	Estudio de Viabilidad	25 h	4	v	
7	Aprovisionamiento del EV	1 h	6	v	
8	Plan de Proyecto	11 h	7	v	
9	Análisis y diseño	45 h		v	
10	Análisis de requisitos	25 h	8	v	
11	Análisis y diseño de base de datos	20 h	10	v	
12	Codificación	180 h			X (2 semanas)
13	Habilitar entorno de desarrollo	2 h	11		x
14	Programar código	178 h	13		x
15	Pruebas	20 h			X (2 semanas)
16	Diseño tests de integración	2 h	14		x
17	Desarrollo de pruebas	18 h	16		x
18	Memoria/Documentación	25 h		v	
19	Finalización/Entrega	1 h		v	

Figura 8.1 Cuadro de desviación en el tiempo de las fases del proyecto

La fase de codificación del código se complicó un poco más de lo previsto debido a la complejidad del Biomax2, ya que nunca antes había programado este dispositivo. A causa de esto las tareas asociadas a esta fase y fase siguiente se vieron afectadas.

8.3 Futuras ampliaciones del proyecto

Algunas de las posibles futuras ampliaciones de este proyecto son las siguientes:

- Ampliar la forma de acceder los socios a través de acceso biometrico.
- Realización de reservas de pistas online a través de internet.
- Modificar aspectos de la aplicación que favorezcan la facilidad e integración del usuario.
- Creación de modulos de remesas y facturación.
- Control de aforo, con la posibilidad de ampliar otro terminal para controlar Entrada/Salida.
- Crear un visor de accesos a nivel público, visualizado en otro monitor para evitar posibles fraudes.
- Aumentar el número de reports más específicos, así como estadísticos.
- Ampliación del punto de venta en un kiosko virtual el cual permite la interacción directa entre el cliente y la aplicación, todo ello de una forma más visual.

Capítulo 9

Definiciones, acrónimos y abreviaciones

Definiciones, acrónimos y abreviaciones

- **Acceso biométrico:** Se refiere al acceso mediante verificación de huella dactilar.
- **Acceso de proximidad RFID:** identificación por radiofrecuencia.
- **BioMAX2:** Dispositivo de la empresa KIMALDI de control de accesos.
- **Diagrama de gantt:** Cronograma del proyecto.
- **Hardware:** Material informático.
- **KIMALDI:** Empresa que desarrolla dispositivos de control de accesos.
- **Microsoft Project 2010 :** software empleado para gestionar y planificar el desarrollo de proyectos.
- **Mysql:** Sistema de gestión de base de datos.
- **PFC :** proyecto final de carrera
- **Software:** Será el programa a diseñar.
- **SQLServer:** Sistema para la gestión de base de datos.
- **Structure Query Language (SQL):** Es un lenguaje declarativo de acceso a base de datos.
- **WBS (Work Breakdown Structure):** descomposición jerárquica de los procesos a seguir en un proyecto que permite definir y gestionar el proyecto en sí.

Capítulo 10

Bibliografía

Libros de texto:

- MS Sql Server 2005 Manual de Administrador - William R. Stanek - Editorial McGraw-Hill 2006
- Microsoft Sql Server 2005. Administración Y Análisis De Bases De Datos. César Pérez López - Ra-Ma Editorial 2006
- Microsoft Visual Basic .Net Lenguaje y Aplicaciones - F.Javier Ceballos Sierra - ALFAOMEGA GRUPO EDITOR 2007
- Visual Basic 2005 (VB.NET) Programe con Visual Studio 2005 - Thierry Groussard - 2006

Páginas de Internet:

- Desarrollo MSDN.
Disponible en: <http://msdn.microsoft.com/es-es/library>.
Fecha última consulta 05/06/2013
- Curso de Visual basic.
Disponible en: <http://www.aulaclic.es/visualbasic/>
Fecha última consulta 12/08/2013
- Cursos de Visual Basic.NET y SQL Server- La Web del Programador.
Disponible en: <http://www.lawebdelprogramador.com>
Fecha última consulta 12/08/2013
- Foros de ayuda MSDN:
Disponible en: <http://social.msdn.microsoft.com/Forums/es-ES>
Fecha última consulta 12/08/2013
- Foros de ayuda BV.NET:
Disponible en: http://www.elguille.info/foros_guille.aspx
Fecha última consulta 10/08/2013
- Servicio Kimaldi – Biomax2 -
Disponible en: <http://www.kimaldi.com>
Fecha última consulta 03/08/2013