
Especificación de Requisitos de Software

MiPandilla

Versión <1.0> <Diseño>

**Alejandro Bermejo Vialas, Mariadalit Rosales Rodríguez,
Mariano González Salazar, Jorge Peláez González, Saad Ziyad Khalid.**

Universidad Carlos III de Madrid

<10/12/2015>

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

Contenido

1. Introducción	3
1.1 Propósito	3
1.2 Alcance	4
1.3 Visión de conjunto	4
1.4 Glosario de términos	5
2. Descripción general del sistema	6
2.1 Funcionalidad	6
2.2 Características de los usuarios	7
3. Requisitos específicos	8
3.1 Requisitos Funcionales	8
3.2 Requisitos No Funcionales	9
3.3 Requisitos de Dominio	10
4. Casos de uso	11

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

1. Introducción

El auge de las aplicaciones Web diseñadas para organizar eventos con el fin de que cualquier persona pueda acudir a ellos ha ocasionado que un gran número de personas se beneficie de estos servicios. Sin embargo, el desarrollo funcional de estas aplicaciones presenta un avance más lento, razón por la cual se generan una serie de barreras que dificultan la difusión de la información entre los miembros usuario de dichas aplicaciones.

Por este motivo, la oficina de atención al ciudadano contrata a TeLEDmíticos para que desarrolle un servicio que facilite la difusión de la información y el alcance de la misma a todo tipo de público.

Con el fin de subsanar este problema, la empresa TeLEDmíticos junto con la oficina de atención al ciudadano de este ayuntamiento va a desarrollar un servicio que facilite la difusión de la información y el alcance de la misma a todo tipo de público.

La oficina de atención al ciudadano proveerá de una sala de ordenadores a disposición del ciudadano, por los cuales podrán acceder a la aplicación MiPandilla.

MiPandilla es una aplicación desarrollada por la empresa TeLEDmíticos que servirá de plataforma gestora de eventos y usuarios. Desde esta aplicación, los usuarios podrán crear y gestionar su cuenta de usuario y crear eventos de carácter público. Los eventos permitirán la interacción por parte de los usuarios completando la plataforma. Además, el listado de eventos tendrá un carácter completamente público, por lo que cualquier ciudadano, registrado o no, podrá verlos.

Además, esta plataforma incluye un sistema de patrocinio mediante el cual se pueden patrocinar eventos de usuarios que contraten el servicio. Dichos eventos ocuparán un espacio reservado en las lista de eventos de modo que sean fácilmente visualizarles por cualquier usuario que vaya a ver el listado de eventos. De este modo diferentes empresas pueden generar ingresos o promocionar nuevos productos.

En este documento se describe el comportamiento del sistema a desarrollar, así como un conjunto de casos de uso que describe todas las interacciones que tendrán los usuarios con el software y los requisitos no funcionales.

1.1 Propósito

El propósito de MiPandilla es proporcionar a cualquier ciudadano el acceso a un servicio de organización de eventos. Una aplicación que cualquier persona pueda utilizar y mediante la cual se pueda organizar eventos y estar informada de futuros eventos.

El carácter general de esta aplicación hace que la audiencia sea amplia. Cualquier persona que cumpla los requisitos mínimos para utilizar un ordenador puede beneficiarse de la aplicación. Sin embargo, el carácter social de la aplicación hace que se vea más dirigida al sector joven de la población.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

1.2 Alcance

La aplicación tiene designado como nombre “MiPandilla”.

Las funcionalidades de los usuarios sobre los eventos incluyen la creación, modificación y eliminación de los mismos (siempre y cuando dicha modificación provenga del creador del evento), así como su seguimiento y un sistema de comentarios mediante el cual podrán expresar opiniones.

Las funcionalidades de los usuarios sobre su perfil u otros usuarios incluyen la modificación o eliminación del perfil propio, seguimiento de otros usuarios y la posibilidad de comentar en tableros de usuarios.

La aplicación hará posible que todas estas funcionalidades puedan ser llevadas a cabo.

Por otro lado, la aplicación no permitirá más que ver eventos a aquellos usuarios que no cuenten con una cuenta de usuario en MiPandilla.

La aplicación no interactuará con el contenido personalizable, es decir, no aplicará un control sobre el contenido que los usuarios suban al servidor de la aplicación. Esta labor será desempeñada por el personal de mantenimiento.

Una vez terminada la aplicación, los ciudadanos se verán beneficiados de una plataforma gestora de eventos con un carácter social. Los patrocinadores podrán sacar provecho de la situación y beneficiarse de los eventos promocionados así como publicitar nuevos productos.

En lo que respecta a requisitos del sistema que ejecutará la aplicación. Ya que la aplicación carece de complejidad de implementación y no necesita de un entorno gráfico más que una terminal de comandos, los requisitos mínimos pueden ser alcanzados por cualquier equipo que soporte distribuciones Linux.

1.3 Visión de conjunto

A partir de este punto se describirá de forma general el sistema (explicando las funciones y los diferentes tipos de usuario que la aplicación puede albergar), sus requisitos (tanto los requisitos funcionales como los no funcionales y de dominio) y por último los diagramas de casos de uso.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

1.4 Glosario de términos

1. *Contrato*: documento legal que establece un acuerdo entre cliente y distribuidor. Este incluye los requisitos técnicos y organizacionales, costes y un registro de un producto. Un contrato debe contener información informal pero útil como los compromisos o expectativas de las partes involucradas.
2. *Cliente*: La persona o personas que pagan por el producto y normalmente (pero no necesariamente) decide los requisitos. En el contexto de esta práctica recomendada, el cliente y el distribuidor suelen ser de la misma organización.
3. *Distribuidor*: La persona o personas que producen un producto para un cliente. En el contexto de esta práctica recomendada, el cliente y el distribuidor suelen ser de la misma organización.
4. *Usuario*: La persona o personas que operan o interactúan directamente con el producto. El usuario(s) y el cliente(s) a veces no son la misma persona(s).

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

2. Descripción general del sistema

2.1 Funcionalidad

MiPandilla está organizada siguiendo una estructura formada por dos bloques: usuarios y eventos.

En primer lugar se sitúan los eventos:

Todos los eventos siguen una misma estructura. Como parámetros tienen el nombre y ámbito del evento, descripción, el lugar y fecha de desarrollo, una lista de usuarios que van a acudir al evento y una sección de comentarios donde diversos usuarios registrados expresarán sus opiniones. A la hora de buscar eventos, se hará a través de un filtro basado en los parámetros de nombre, ámbito, lugar y fecha. A su vez, los eventos estarán divididos en dos listas de eventos: promocionados (estos eventos son los creados por los usuarios con privilegios de promotor) y no promocionados.

Por otro lado se encuentran los usuarios:

Estos pueden ser administrador, promotor, usuario registrado y usuario no registrado. También siguen una misma estructura que como parámetros tiene: nombre y apellidos del usuario, fecha de nacimiento, lugar de residencia, estado civil, frase favorita, una lista de seguidores y eventos a los que va a acudir y tablón de comentarios. Los usuarios registrados pueden seguir a otros usuarios y comentar en su tablón. A la hora de buscar usuarios también se realizará mediante un filtro que recoge los atributos de nombre, apellidos, lugar y fecha de nacimiento.

En el momento de representar eventos, se representarán de 10 en 10 (10 por cada página), y en cada página, tres de los eventos representados serán eventos promocionados (representándose en los dos primeros puestos de cada página).

La representación de usuarios se realiza de forma análoga.

Los objetivos del proyecto MiPandilla son:

- Proveer de la infraestructura necesaria para operar una red de eventos y usuarios.
- Proveer de un marco para patrocinadores para su generación de ingresos y promoción de nuevos productos.
- Proveer de un marco para usuarios que quieran expresar sus opiniones referentes a los diferentes eventos y usuarios.

Con respecto a los beneficios, los usuarios se beneficiarán de una plataforma difusora de eventos al alcance de todos y libre a la hora de la generación de los mismos. A su vez podrán organizar eventos promocionados. Los patrocinadores se beneficiarán de la generación de eventos patrocinados y podrán generar otros con el fin de publicitar nuevos productos o servicios.

Con respecto a los posibles casos de uso, entre ellos se encuentran: el registro de usuarios, inicio y cierre de sesión, visualización, eliminación y modificación de eventos y perfiles

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

de usuario, publicación de comentarios en tableros (dentro de perfiles de usuario) y eventos, inscripción en eventos y seguimiento de usuarios.

2.2 Características de los usuarios

El sistema de usuarios de MiPandilla se corresponde con un modelo muy sencillo de administrador, patrocinadores y usuario final.

La figura del administrador no dispone de un usuario registrado dentro de la aplicación. El administrador es aquel que puede acceder a los archivos y modificarlos según aconsejen las circunstancias del momento.

Las funciones del administrador son las siguientes:

- Realizar un seguimiento del contenido de la aplicación.
- Solucionar cualquier incidencia que pueda ocurrir con respecto al funcionamiento de la aplicación.
- Modificar o eliminar información errónea o inadecuada en caso de necesidad.

Los patrocinadores disponen de cuentas de usuario con privilegios en la generación de eventos. El sistema de patrocinio sigue el siguiente proceso:

1. Un usuario decide que quiere que un evento suyo llegue con más facilidad al público.
2. Entonces contacta con una empresa patrocinadora y establecen un acuerdo.
3. Dicha empresa crea el evento que cumple las características que el usuario contratante quiera.
4. Este evento pasará a la lista de eventos patrocinados (estos se representarán en un espacio reservado a los mismos en las listas de eventos).

Por otro lado, una empresa podrá generar eventos promocionales.

El usuario final hace referencia al ciudadano que accede a la aplicación.

El usuario final puede ser de dos tipos: usuario final registrado (aquel que disponga de un perfil y haga uso de la aplicación accediendo al mismo) y el usuario final no registrado (aquel que no disponga de una cuenta de usuario y acceda al servicio de ver eventos).

El usuario final registrado puede modificar su perfil, ver, comentar, crear, modificar, eliminar y seguir eventos y ver y seguir a diferentes cuentas de usuarios.

El usuario final no registrado no dispone de ninguna clase de derecho de interacción dentro de la aplicación, ya que únicamente puede ver el listado de eventos activos.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

3. Requisitos específicos

3.1 Requisitos Funcionales

RF1 Registro de usuarios: El sistema debe permitir el registro de usuarios y el almacenamiento de los mismos.

RF2 Tipos de usuarios registrados: Deberán existir dos tipos de usuarios registrados: normal y patrocinador. Cada usuario tendrá un nombre (identificador) único y una contraseña.

RF3 Seguridad en contraseñas: Las contraseñas deberán tener un mínimo de seguridad: tener una longitud no inferior a 6 caracteres e incluir como mínimo una letra en mayúscula y un número y no contener el nombre de usuario o las palabras 'password' o 'contraseña'.

RF4 Identificación: El sistema debe ser capaz de comprobar que el usuario y la contraseña coinciden. En caso de no coincidir, el usuario podrá volver a intentar entrar.

RF5 Accesibilidad: Dependiendo del tipo de usuario, los eventos podrán ser visualizados por los usuarios.

RF6 Perfil: El perfil de usuario debe tener como campos a rellenar (de forma opcional): nombre y apellidos del usuario, fecha de nacimiento, lugar de residencia, estado civil y frase favorita.

RF7 Gestión de perfil: Los usuarios deberán ser capaces de modificar o eliminar su perfil, entendiéndose modificar como el cambio de cualquiera de los campos a rellenar de su perfil de usuario.

RF8 Visualización de eventos: Los eventos aparecerán representados de diez en diez por página (sólo una página visible al mismo tiempo, para facilitar la comprensión). En la visualización se verán todos los parámetros salvo los comentarios y la descripción del evento, que aparecerán al entrar en la descripción del evento. Los eventos patrocinados saldrán ocupando las dos primeras posiciones y la aparición de estos será aleatoria. Cualquier usuario (tanto registrado como no registrado) deberá ser capaz de visualizar los eventos.

RF9 Visualización de usuarios: También aparecerán de diez en diez por página. Sólo podrán ser visualizados por usuarios registrados. En la visualización se verán todos los campos del perfil de usuario salvo los comentarios en el tablón, que aparecerán al entrar en el perfil del usuario.

RF10 Seguimiento de usuarios: El sistema debe permitir el seguimiento de usuarios por parte de usuarios. El seguimiento de usuarios sólo puede ser realizado por usuarios registrados.

RF11 Creación de eventos: El sistema debe permitir la creación de eventos. En caso de ser creados por un usuario patrocinador, este evento deberá ser almacenado en una lista de eventos patrocinados para que en su visualización aparezca en el lugar debido. La creación de eventos sólo puede ser realizada por usuarios registrados.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

RF12 Modificación de eventos: El sistema debe permitir la modificación de los parámetros de evento. La modificación de eventos sólo puede ser realizada por usuarios registrados y sólo por el usuario que creó dicho evento.

RF13 Eliminación de eventos: El sistema debe permitir la eliminación de eventos. La eliminación de eventos sólo puede ser realizada por usuarios registrados y sólo por el usuario que creó dicho evento.

RF14 Inscripción en eventos: El sistema debe permitir la inscripción en eventos por parte de usuarios. La inscripción en eventos sólo puede ser realizada por usuarios registrados.

RF15 Búsqueda de eventos: El sistema deberá proporcionar una búsqueda de eventos por filtración, de modo que puedan ser filtrados por sus parámetros (nombre, lugar, fecha y ámbito) para su posterior muestra.

RF16 Modificación de cuentas de usuario: El sistema debe permitir la modificación de cuentas de usuario. La modificación de cuentas de usuario sólo puede ser realizada por usuarios registrados y sólo por el usuario que inició sesión.

RF17 Eliminación de cuentas de usuario: El sistema debe permitir la eliminación de cuentas de usuario. La eliminación de cuentas de usuario sólo puede ser realizada por usuarios registrados y sólo por el usuario que inició sesión.

RF18 Búsqueda de usuarios: Asimismo, el filtraje podrá ser aplicado de la misma forma en los usuarios.

RF19 Comentario en eventos: Los eventos dispondrán de una sección de comentarios en la que diferentes usuarios podrán expresarse. La representación de los comentarios deberá incluir el nombre de usuario del autor del comentario.

RF20 Comentario en perfiles: Los perfiles de usuario dispondrán de un tablón de comentarios en la que diferentes usuarios podrán expresarse. La representación de los comentarios deberá incluir el nombre de usuario del autor del comentario.

3.2 Requisitos No Funcionales

RNF1 Usabilidad: El sistema debe ser fácil de usar con respecto al usuario. La aplicación puede ser usada por todo tipo de personas por lo que es importante que todas ellas dispongan de facilidades a la hora de comprender y usar la aplicación.

RNF2 Confiabilidad: El sistema ha de ofrecer medidas de seguridad como requerir una contraseña de usuario fiable (con un número mínimo de letras: 6 y como mínimo una de ellas en mayúscula). El almacenaje de datos ha de ser fácil de duplicar para evitar posibles pérdidas de datos.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

RNF3 Rendimiento: El programa debe interactuar consigo mismo de forma rápida y eficiente, de tal manera que el tiempo de procesamiento en el que el usuario tiene la aplicación bloqueada, no excede de 0.5 segundos. Un usuario puede trabajar al mismo tiempo por equipo y el sistema ha de ser capaz de almacenar el tamaño del archivo de almacenaje en ese momento más un margen de 50 megabytes. El almacenaje del sistema ha de estar abierto a ampliaciones periódicas (o por lo menos tener suficiente espacio como para poder reservar más espacio a lo largo del tiempo: 200 MB).

RNF4 Compatibilidad: La aplicación no tiene por qué requerir de alguna extensión para funcionar. El sistema será mantenido por los empleados de mantenimiento de la empresa contratante.

RNF5 Implementación: No existen restricciones en la plataforma de hardware a utilizar. No existen restricciones por parte del grupo de mantenimiento o de testing.

RNF6 Interfaz: La importación y exportación de datos se realiza a un documento de almacenamiento de datos. La interacción con otros sistemas se lleva a cabo mediante la actualización periódica de este documento en todos los sistemas. La interfaz de la aplicación será una terminal de comandos.

RNF7 Claridad: La representación de la información en la terminal deberá ser con letras negras sobre un fondo blanco. Esta característica aporta claridad a la aplicación y resulta más comprensible desde el punto de vista de los usuarios.

RNF8 Operación: durante la producción del sistema, la gestión será realizada por el equipo de implementación del mismo.

RNF9 Despliegue: El sistema será instalado por el equipo de mantenimiento de la empresa contratante. Una instalación por equipo será necesaria.

RNF10 Mantenible: El sistema debe aportar facilidades a la hora de realizar pruebas, cambiar cosa o realizar análisis del sistema.

RNF11 Legal: El sistema ha de ser legal, con una licencia de Creative Commons.

RNF12 Responsabilidades de fallo: Las responsabilidades en caso de fallo del sistema serán del equipo de implementación del mismo si el fallo corresponde a este proceso, o de la empresa contratante si el fallo es debido a algún problema de hardware o personal.

3.3 Requisitos de Dominio

RD1 Plataforma: La empresa contratante deberá aportar una serie de equipos mediante los cuales la aplicación pueda ejecutarse.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

4. Casos de uso

Identificador caso de uso:	UC-1
Nombre caso de uso:	Registrarse en la aplicación.
Actores:	<i>Iniciador 1:</i> UsuarioNormal <i>Iniciador 2:</i> Administrador Patrocinador se comunica con: Administrador.
Descripción:	Los diferentes tipos de usuario han de interactuar con el sistema para obtener su cuenta de usuario. Dos tipos de cuentas de usuario: normal y patrocinadora.
Precondiciones:	<ol style="list-style-type: none"> 1. UsuarioNormal no dispone de una cuenta de usuario en la aplicación. 2. Patrocinador ha sido contactado por un usuario. 3. El Administrador ha sido informado acerca del nuevo usuario patrocinador (Administrador).
Post-condiciones:	<ol style="list-style-type: none"> 1. UsuarioNormal dispone de una cuenta de usuario tipo normal. 2. Patrocinador dispone de una cuenta de usuario tipo patrocinadora.
Flujo básico:	<ol style="list-style-type: none"> 1. UsuarioNormal entra en la aplicación y selecciona la opción registrarse. 2. El sistema le pide al usuario que introduzca un nombre de usuario y una contraseña. 3. UsuarioNormal elige un nombre de usuario válido y una contraseña que cumpla los requisitos de seguridad. 4. El sistema comprueba que los datos son válidos (nombre de usuario disponible y contraseña segura). 5. UsuarioNormal obtiene una cuenta de usuario con sus datos de nombre de usuario y contraseña. 6. Administrador entra en los archivos y crea una cuenta de usuario tipo patrocinadora con los datos de nombre de usuario y contraseña elegidos por Patrocinador. 7. Patrocinador recibe su cuenta de usuario tipo patrocinadora.
Requisitos involucrados:	RF1, RF3, RNF1, RNF2, RNF6, RNF7 y RD1.
Flujos secundarios:	<ol style="list-style-type: none"> 4a. En el paso 4, si el nombre de usuario no está disponible: <ol style="list-style-type: none"> 1. El sistema muestra al usuario que el nombre de usuario está cogido. 2. El sistema vuelva al paso 2. 4b. En el paso 4, si la contraseña no es segura: <ol style="list-style-type: none"> 1. El sistema le muestra al usuario que la contraseña no es segura. 2. El sistema vuelva al paso 2.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

Identificador caso de uso:	UC-2.1.1
Nombre caso de uso:	Modificar perfil de usuario.
Actores:	<i>Iniciador 1:</i> UsuarioRegistrado
Descripción:	Tras iniciar sesión en la aplicación, se puede modificar el perfil de usuario del usuario que inició sesión.
Precondiciones:	1. UsuarioRegistrado dispone de una cuenta de usuario en la aplicación.
Post-condiciones:	1. UsuarioRegistrado modifica su perfil de usuario.
Flujo básico:	<ol style="list-style-type: none"> 1. UsuarioRegistrado inicia sesión y en el menú principal y ha seleccionado la opción de mostrar perfil de usuario. 2. Sistema representa el perfil del usuario. 3. UsuarioRegistrado selecciona la opción de modificar perfil. 4. Sistema muestra los campos del perfil de usuario. 5. UsuarioRegistrado completa los campos. 6. Sistema comprueba que los campos son correctos. 7. Sistema modifica el perfil del usuario.
Requisitos involucrados:	RF5, RF16, RNF1, RNF3 y RNF7.
Flujos secundarios:	6a. En el paso 6, si uno o varios de los campos son incorrectos: <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado los campos incorrectos. 2. El sistema vuelve al paso 4.
<<Includes>>:	<ol style="list-style-type: none"> 1. Iniciar sesión.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

Identificador caso de uso:	UC-2.1.2
Nombre caso de uso:	Eliminar cuenta de usuario
Actores:	<i>Iniciador 1:</i> UsuarioRegistrado
Descripción:	Tras iniciar sesión en la aplicación, se puede eliminar la cuenta de usuario del usuario que inició sesión en la aplicación.
Precondiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado dispone de una cuenta de usuario en la aplicación. 2. UsuarioRegistrado ha iniciado sesión en la aplicación.
Post-condiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado elimina su cuenta de usuario.
Flujo básico:	<ol style="list-style-type: none"> 1. UsuarioRegistrado inicia sesión y en el menú principal ha seleccionado la opción de mostrar perfil de usuario. 2. Sistema representa el perfil del usuario. 3. UsuarioRegistrado selecciona la opción de eliminar cuenta de usuario. 4. Sistema pregunta si está seguro de que quiere eliminar su cuenta de usuario. 5. UsuarioRegistrado confirma la eliminación de su cuenta de usuario. 6. Sistema elimina la cuenta de usuario y devuelve a UsuarioRegistrado a la pantalla principal de la aplicación.
Requisitos involucrados:	RF5, RF17, RNF1, RNF3 y RNF7.
Flujos secundarios:	<p>5a. En el paso 5, si la respuesta es errónea:</p> <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado que la respuesta es errónea. 2. El sistema vuelve al paso 4.
<<Includes>>:	<ol style="list-style-type: none"> 1. Iniciar sesión.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

Identificador caso de uso:	UC-2
Nombre caso de uso:	Iniciar sesión.
Actores:	<i>Iniciador 1:</i> UsuarioRegistrado
Descripción:	Proceso tras del cual UsuarioRegistrado entra en la aplicación con su nombre de usuario y contraseña.
Precondiciones:	1. UsuarioRegistrado dispone de una cuenta de usuario en la aplicación.
Post-condiciones:	1. UsuarioRegistrado inicia sesión en la aplicación.
Flujo básico:	<ol style="list-style-type: none"> 1. UsuarioRegistrado entra en la aplicación y selecciona la opción de iniciar sesión. 2. El sistema pide a UsuarioRegistrado que introduzca su nombre de usuario y su contraseña. 3. UsuarioRegistrado completa los campos con su nombre de usuario y su contraseña. 4. Sistema comprueba que la información es correcta. 5. El sistema muestra al usuario el menú y le indica que ha iniciado sesión correctamente.
Requisitos involucrados:	RNF4, RNF1, RNF3, RNF7 y RD1.
Flujos secundarios:	<ol style="list-style-type: none"> 4a. En el paso 4, alguno de los campos es incorrecto: <ol style="list-style-type: none"> 1. El sistema muestra al usuario el campo o los campos que son incorrectos. 2. El sistema vuelve al paso 2.
<<Includes>>:	1. Iniciar sesión.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

Identificador caso de uso:	UC-2.2
Nombre caso de uso:	Ver eventos.
Actores:	<i>Iniciador 1:</i> UsuarioNoRegistrado <i>Iniciador 2:</i> UsuarioRegistrado
Descripción:	Proceso de representación de los eventos dependiendo del tipo de usuario que quiera visualizarlos.
Precondiciones:	1. UsuarioRegistrado dispone de una cuenta de usuario en la aplicación.
Post-condiciones:	1. UsuarioRegistrado y UsuarioNoRegistrado visualizan los eventos.
Flujo básico:	<ol style="list-style-type: none"> 1. UsuarioNoRegistrado entra en la aplicación y selecciona la opción de ver eventos. 2. El sistema muestra la ventana de filtros. 3. UsuarioNoRegistrado completa los campos del filtro. 4. Sistema comprueba que la información es correcta. 5. Si la información es correcta se avanza al paso 10. 6. UsuarioRegistrado entra en la aplicación y selecciona la opción de iniciar sesión. 7. El sistema le pide a UsuarioRegistrado que introduzca un nombre de usuario y una contraseña. 8. UsuarioRegistrado introduce su nombre de usuario y contraseña. 9. El sistema comprueba que los datos son válidos (nombre de usuario existe y contraseña concuerda). 10. Registrado accede a la aplicación como usuario registrado y selecciona la opción de ver eventos. 11. El sistema muestra la ventana de filtros. 12. UsuarioRegistrado completa los campos del filtro. 13. Sistema comprueba que la información es correcta. 14. Sistema representa los eventos.
Requisitos involucrados:	RF5, RF8, RNF15, RNF1, RNF3 y RNF7.
Flujos secundarios:	<ol style="list-style-type: none"> 4a. En el paso 4, alguno de los campos es incorrecto: <ol style="list-style-type: none"> 1. El sistema muestra al usuario el campo o los campos que son incorrectos. 2. El sistema vuelva al paso 2. 9a. En el paso 9, si la información no concuerda: <ol style="list-style-type: none"> 1. El sistema le muestra al usuario que el nombre de usuario o la contraseña no son correctos. 2. El sistema vuelva al paso 7.
<<Includes>>:	1. Iniciar sesión (si UsuarioRegistrado).

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

Identificador caso de uso:	UC-2.3.1
Nombre caso de uso:	Filtrar usuarios
Actores:	<i>Iniciador 1:</i> UsuarioRegistrado.
Descripción:	T iniciar sesión en la aplicación, se pueden buscar usuarios.
Precondiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado dispone de una cuenta de usuario en la aplicación. 2. UsuarioRegistrado ha iniciado la aplicación.
Post-condiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado filtra los usuarios.
Flujo básico:	<ol style="list-style-type: none"> 1. UsuarioRegistrado inicia la aplicación. 2. UsuarioRegistrado selecciona la opción de filtrar eventos. 3. Sistema pide a UsuarioRegistrado que introduzca los campos de los usuarios a buscar que quiere filtrar. 4. UsuarioRegistrado completa los campos. 5. Sistema filtra los usuarios.
Requisitos involucrados:	RF5, RF9, RNF1, RNF3 y RNF7.
Flujos secundarios:	<p>5a. En el paso 5, si no se encontraron usuarios:</p> <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado que no se han podido buscar usuarios con esos campos de búsqueda. 2. El sistema regresa al menú principal o al de inicio.
<<Includes>>:	<ol style="list-style-type: none"> 1. Iniciar sesión.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

Identificador caso de uso:	UC-2.3.2
Nombre caso de uso:	Ver usuarios.
Actores:	<i>Iniciador 1:</i> UsuarioRegistrado
Descripción:	Proceso de representación de los usuarios.
Precondiciones:	1. UsuarioRegistrado dispone de una cuenta de usuario en la aplicación.
Post-condiciones:	1. UsuarioRegistrado visualiza los usuarios.
Flujo básico:	<ol style="list-style-type: none"> 1. UsuarioRegistrado entra en la aplicación y selecciona la opción de iniciar sesión. 2. El sistema le pide a UsuarioRegistrado que introduzca un nombre de usuario y una contraseña. 3. UsuarioRegistrado introduce su nombre de usuario y contraseña. 4. El sistema comprueba que los datos son válidos (nombre de usuario existe y contraseña concuerda). 5. Registrado accede a la aplicación como usuario registrado y selecciona la opción de ver usuarios. 6. El sistema muestra la ventana de filtros. 7. UsuarioRegistrado completa los campos del filtro. 8. Sistema comprueba que la información es correcta. 9. Sistema representa los usuarios.
Requisitos involucrados:	RF5, RF9, RF18, RNF1, RNF3, RNF7.
Flujos secundarios:	<p>4a. En el paso 4, si la información no concuerda:</p> <ol style="list-style-type: none"> 1. El sistema le muestra al usuario que el nombre de usuario o la contraseña no son correctos. 2. El sistema vuelva al paso 2. <p>7a. En el paso 7, alguno de los campos es incorrecto:</p> <ol style="list-style-type: none"> 1. El sistema muestra al usuario el campo o los campos que son incorrectos. 2. El sistema vuelva al paso 6.
<<Includes>>:	1. Iniciar sesión.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

Identificador caso de uso:	UC-2.2.1
Nombre caso de uso:	Crear eventos
Actores:	<i>Iniciador 1:</i> UsuarioRegistrado
Descripción:	Tras iniciar sesión en la aplicación, se pueden crear eventos.
Precondiciones:	1. UsuarioRegistrado dispone de una cuenta de usuario en la aplicación.
Post-condiciones:	1. UsuarioRegistrado crea un evento.
Flujo básico:	<ol style="list-style-type: none"> 1. UsuarioRegistrado inicia sesión y en el menú principal selecciona la opción de crear evento. 2. El sistema muestra a UsuarioRegistrado los parámetros a completar 3. UsuarioRegistrado completa los campos. 4. Sistema comprueba que los campos son correctos. 5. Sistema crea el nuevo evento.
Requisitos involucrados:	RF5, RF11, RNF1, RNF3, RNF7.
Flujos secundarios:	<ol style="list-style-type: none"> 4a. En el paso 4, si uno o varios de los campos son incorrectos: <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado los campos incorrectos. 2. El sistema vuelve al paso 2.
<<Includes>>:	1. Iniciar sesión.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

Identificador caso de uso:	UC-2.2.2
Nombre caso de uso:	Modificar eventos
Actores:	<i>Iniciador 1:</i> UsuarioRegistrado
Descripción:	Tras iniciar sesión en la aplicación, se pueden modificar eventos previamente creados por ese mismo usuario.
Precondiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado dispone de una cuenta de usuario en la aplicación. 2. UsuarioRegistrado ha creado el evento que desea modificar.
Post-condiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado modifica un evento.
Flujo básico:	<ol style="list-style-type: none"> 1. UsuarioRegistrado inicia sesión y en el menú principal y ha filtrado los eventos que quiere visualizar (entre ellos el que desea modificar). 2. Sistema representa los eventos filtrados. 3. UsuarioRegistrado selecciona el evento que desea modificar. 4. Sistema muestra la información del evento. 5. UsuarioRegistrado selecciona la opción de modificar evento. 6. Sistema muestra los campos del evento. 7. UsuarioRegistrado completa los campos. 8. Sistema comprueba que los campos son correctos. 9. Sistema modifica el evento.
Requisitos involucrados:	RF5, RF12, RNF1, RNF3 y RNF7.
Flujos secundarios:	<p>3a. En el paso 3, si el evento no existe:</p> <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado que el evento seleccionado no existe. 2. El sistema vuelve al paso 2. <p>5a. En el paso 5, si el evento no fue creado por UsuarioRegistrado:</p> <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado que no puede modificar ese evento ya que no es el creador del mismo. 2. El sistema vuelve al paso 4. <p>8a. En el paso 8, si uno o varios de los campos son incorrectos:</p> <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado los campos incorrectos. 2. El sistema vuelve al paso 6
<<Includes>>:	<ol style="list-style-type: none"> 1. Iniciar sesión. 2. Crear eventos.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

Identificador caso de uso:	UC-2.2.3
Nombre caso de uso:	Filtrar eventos
Actores:	<i>Iniciador 1:</i> UsuarioRegistrado o UsuarioNoRegistrado
Descripción:	En el menú de inicio y tras iniciar sesión en la aplicación, se pueden buscar eventos.
Precondiciones:	3. UsuarioRegistrado dispone de una cuenta de usuario en la aplicación. 4. UsuarioNoRegistrado o UsuarioRegistrado ha iniciado la aplicación.
Post-condiciones:	2. UsuarioRegistrado o UsuarioNoRegistrado filtra los eventos.
Flujo básico:	6. UsuarioRegistrado o UsuarioNoRegistrado inicia la aplicación. 7. UsuarioRegistrado o UsuarioNoRegistrado selecciona la opción de filtrar eventos. 8. Sistema pide a UsuarioRegistrado o UsuarioNoRegistrado que introduzca los campos de los eventos a buscar que quiere filtrar. 9. UsuarioRegistrado o UsuarioNoRegistrado completa los campos. 10. Sistema filtra los eventos.
Requisitos involucrados:	RF5, RNF15, RNF1, RNF3 y RNF7.
Flujos secundarios:	2a. En el paso 2, si UsuarioNoRegistrado: 1. Sistema representa el menú de inicio y UsuarioNoRegistrado selecciona la opción de filtrar eventos. 2. El sistema vuelve al paso 3. 2b. En el paso 2, si UsuarioRegistrado: 1. UsuarioRegistrado inicia sesión y selecciona la opción de filtrar usuarios. 2. El sistema vuelve al paso 3. 5a. En el paso 5, si no se encontraron eventos: 3. El sistema muestra a UsuarioRegistrado o UsuarioNoRegistrado que no se han podido buscar eventos con esos campos de búsqueda. 4. El sistema regresa al menú principal o al de inicio.
<<Includes>>:	2. Iniciar sesión.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

Identificador caso de uso:	UC-2.2.4
Nombre caso de uso:	Seguir eventos
Actores:	<i>Iniciador 1:</i> UsuarioRegistrado
Descripción:	Tras poder ver la lista de eventos como usuario registrado, se pueden seguir los eventos que el usuario estime.
Precondiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado dispone de una cuenta de usuario en la aplicación. 2. Existen eventos en la lista de eventos.
Post-condiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado se inscribe en un evento.
Flujo básico:	<ol style="list-style-type: none"> 1. UsuarioRegistrado ve la lista de eventos con el filtro que aplicó en su búsqueda. 2. UsuarioRegistrado selecciona un evento con el que quiere interactuar. 3. Sistema comprueba que el evento existe. 4. Sistema representa los datos del evento seleccionado. 5. UsuarioRegistrado selecciona la opción de inscribirse en el evento. 6. Sistema comprueba que actualmente UsuarioRegistrado no está siguiendo el evento.
Requisitos involucrados:	RF5, RF14, RNF1, RNF3, RNF7.
Flujos secundarios:	<ol style="list-style-type: none"> 3a. En el paso 3, el evento seleccionado no existe: <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado que el evento no existe. 2. El sistema vuelve al paso 1. 6a. En el paso 6, si UsuarioRegistrado ya está inscrito en el evento: <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado que ya está inscrito en el evento. 2. El sistema vuelve al paso 4 6b. En el paso 6, UsuarioRegistrado no está inscrito en el evento: <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado que ya está inscrito en el evento. 2. El sistema vuelve al menú principal.
<<Includes>>:	<ol style="list-style-type: none"> 1. Iniciar sesión. 2. Ver eventos.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

Identificador caso de uso:	UC-2.2.5
Nombre caso de uso:	Comentar eventos
Actores:	<i>Iniciador 1:</i> UsuarioRegistrado
Descripción:	Tras poder ver la lista de eventos como usuario registrado, se pueden comentar los eventos que el usuario estime.
Precondiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado dispone de una cuenta de usuario en la aplicación. 2. Existen eventos en la lista de eventos.
Post-condiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado comenta en un evento.
Flujo básico:	<ol style="list-style-type: none"> 1. UsuarioRegistrado ve la lista de eventos con el filtro que aplicó en su búsqueda. 2. UsuarioRegistrado selecciona un evento en el que quiere comentar. 3. Sistema comprueba que el evento existe. 4. Sistema representa los datos del evento seleccionado. 5. UsuarioRegistrado selecciona la opción comentar en el evento. 6. UsuarioRegistrado redacta y guarda el comentario. 7. Se registra el comentario de UsuarioRegistrado.
Requisitos involucrados:	RF5, RF19, RNF1, RNF3, RNF7.
Flujos secundarios:	<ol style="list-style-type: none"> 3a. En el paso 3, el evento seleccionado no existe: <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado que el evento no existe. 2. El sistema vuelve al paso 1.
<<Includes>>:	<ol style="list-style-type: none"> 1. Iniciar sesión. 2. Ver eventos.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

Identificador caso de uso:	UC-2.2.6
Nombre caso de uso:	Eliminar eventos
Actores:	<i>Iniciador 1:</i> UsuarioRegistrado
Descripción:	Tras iniciar sesión en la aplicación, se pueden eliminar eventos previamente creados por ese mismo usuario.
Precondiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado dispone de una cuenta de usuario en la aplicación. 2. UsuarioRegistrado ha creado el evento que desea eliminar.
Post-condiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado elimina un evento.
Flujo básico:	<ol style="list-style-type: none"> 1. UsuarioRegistrado inicia sesión y en el menú principal y ha filtrado los eventos que quiere visualizar (entre ellos el que desea eliminar). 2. Sistema representa los eventos filtrados. 3. UsuarioRegistrado selecciona el evento que desea eliminar. 4. Sistema muestra la información del evento. 5. UsuarioRegistrado selecciona la opción de eliminar evento. 6. Sistema pregunta si está seguro de que quiere eliminar ese evento. 7. UsuarioRegistrado confirma la eliminación del evento. 8. Sistema elimina el evento.
Requisitos involucrados:	RF5, RF13, RNF1, RNF3 y RNF7.
Flujos secundarios:	<p>3a. En el paso 3, si el evento no existe:</p> <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado que el evento seleccionado no existe. 2. El sistema vuelve al paso 2. <p>5a. En el paso 5, si el evento no fue creado por UsuarioRegistrado:</p> <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado que no puede eliminar ese evento ya que no es el creador del mismo. 2. El sistema vuelve al paso 4. <p>7a. En el paso 7, si la respuesta es errónea:</p> <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado que la respuesta es errónea. 2. El sistema vuelve al paso 6.
<<Includes>>:	<ol style="list-style-type: none"> 1. Iniciar sesión. 2. Crear eventos.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

Identificador caso de uso:	UC-2.3.1
Nombre caso de uso:	Seguir usuarios
Actores:	<i>Iniciador 1:</i> UsuarioRegistrado
Descripción:	Tras poder ver la lista de usuarios como usuario registrado, se pueden seguir a los usuarios que el usuario estime.
Precondiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado dispone de una cuenta de usuario en la aplicación. 2. Existen usuarios en la lista de usuarios.
Post-condiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado sigue a un usuario.
Flujo básico:	<ol style="list-style-type: none"> 1. UsuarioRegistrado ve la lista de usuarios con el filtro que aplicó en su búsqueda. 2. UsuarioRegistrado selecciona un usuario con el que quiere interactuar. 3. Sistema comprueba que el usuario existe. 4. Sistema representa los datos del usuario seleccionado. 5. UsuarioRegistrado selecciona la opción seguir al usuario. 6. Sistema comprueba que actualmente UsuarioRegistrado no está siguiendo al usuario.
Requisitos involucrados:	RF5, RF10, RNF1, RNF3, RNF7.
Flujos secundarios:	<p>3a. En el paso 3, el usuario seleccionado no existe:</p> <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado que el usuario no existe. 2. El sistema vuelve al paso 1. <p>6a. En el paso 6, si UsuarioRegistrado ya está siguiendo al usuario:</p> <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado que ya está siguiendo al usuario. 2. El sistema vuelve al paso 4 <p>6b. En el paso 6, UsuarioRegistrado no está siguiendo al usuario:</p> <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado que ya está siguiendo al usuario. 2. El sistema vuelve al menú principal.
<<Includes>>:	<ol style="list-style-type: none"> 1. Iniciar sesión. 2. Ver usuarios.

MiPandilla	Versión: <1.0>
Especificación de Requerimientos de Software	Fecha: <08-10-2015>

Identificador caso de uso:	UC-2.3.2
Nombre caso de uso:	Comentar en perfiles de usuario.
Actores:	<i>Iniciador 1:</i> UsuarioRegistrado
Descripción:	Tras poder ver la lista de usuarios como usuario registrado, se pueden comentar en los perfiles de usuario.
Precondiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado dispone de una cuenta de usuario en la aplicación. 2. Existen usuarios en la lista de usuarios.
Post-condiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado comenta en un perfil de usuario.
Flujo básico:	<ol style="list-style-type: none"> 1. UsuarioRegistrado ve la lista de usuarios con el filtro que aplicó en su búsqueda. 2. UsuarioRegistrado selecciona un usuario con el que quiere interactuar. 3. Sistema comprueba que el usuario existe. 4. Sistema representa los datos del usuario seleccionado. 5. UsuarioRegistrado selecciona la opción comentar en el tablón. 6. UsuarioRegistrado redacta y guarda el comentario. 7. Se registra el comentario de UsuarioRegistrado.
Requisitos involucrados:	RF5, RF20, RNF1, RNF3, RNF7.
Flujos secundarios:	<ol style="list-style-type: none"> 3a. En el paso 3, el usuario seleccionado no existe: <ol style="list-style-type: none"> 1. El sistema muestra a UsuarioRegistrado que el usuario no existe. 2. El sistema vuelve al paso 1.
<<Includes>>:	<ol style="list-style-type: none"> 1. Iniciar sesión. 2. Ver usuarios.

Identificador caso de uso:	UC-3
Nombre caso de uso:	Cerrar sesión.
Actores:	<i>Iniciador 1:</i> UsuarioRegistrado
Descripción:	Proceso tras del cual UsuarioRegistrado sale de la aplicación como usuario registrado.
Precondiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado dispone de una cuenta de usuario en la aplicación. 2. UsuarioRegistrado inició sesión en la aplicación.
Post-condiciones:	<ol style="list-style-type: none"> 1. UsuarioRegistrado cierra sesión en la aplicación.
Flujo básico:	<ol style="list-style-type: none"> 1. UsuarioRegistrado está en el menú principal de la aplicación como usuario que ha iniciado sesión. 2. UsuarioRegistrado selecciona la opción de cerrar sesión. 3. El sistema muestra al usuario la pantalla de inicio (se verá como UsuarioRegistrado no está en la aplicación como usuario que ha iniciado sesión).
Requisitos involucrados:	RNF1 y RNF7.