

Universidad Politécnica de Madrid

Escuela Técnica Superior de Ingeniería de
Sistemas Informáticos

Control de asistencia de alumnos automatizado vía Bluetooth Smart

Tutor

Departamento: **Inteligencia Artificial**

Tutor de Proyecto: **Francisco Serradilla García**

Alumno: **Miguel Ángel Sánchez Fernández**

Curso académico: **2016/2017**

*A mi familia.
Por su apoyo incondicional en todo momento.*

Stay hungry. Stay foolish.

STEVE JOBS

*Ever tried. Ever failed. No matter.
Try again. Fail again. Fail better.*

SAMUEL BECKETT

En este documento se describe un Proyecto Fin de Grado de la titulación Graduado en Ingeniería del Software en el que se presenta una herramienta software denominada Tutor, pensada y desarrollada para facilitar a los profesores la gestión de asistencias a clase de alumnos.

El objetivo principal de este proyecto es modernizar y optimizar el proceso de control de asistencia habitual que se lleva a cabo en las asignaturas, reduciendo los tiempos de gestión y centralizando toda la información en una herramienta dedicada.

Esta herramienta se ha pensado para ser ejecutada en dispositivos móviles inteligentes (smartphones) con sistema operativo iOS 7 o superior y requiere de un dispositivo hardware adicional para aprovechar al máximo todas sus características. Este hardware adicional es un dispositivo de baliza activa (active beacon), que transmite una señal Bluetooth LTE de manera omnidireccional con un radio de hasta 100 metros, con el que determinaremos de manera unívoca una región en el espacio que estará asociada a una asignatura.

El proceso de control involucrado en este sistema está basado en la detección de alumnos dentro o fuera de la región del espacio que hayamos configurado con la ayuda del active beacon. Una vez detectada la ubicación del alumno, la herramienta se encarga automáticamente de almacenar sus entradas y salidas de la región.

Además de este sistema de control de asistencias automático, se ha desarrollado un sistema de control de asistencias manual para poder realizar la gestión de asistencias en aquellos casos en los que la ausencia de algún requisito tecnológico, como el active beacon por parte del profesor o dispositivo iOS 7 en caso del alumno, suponga un impedimento para realizar la tarea principal de esta herramienta software.

Dependiendo del perfil del usuario, la herramienta permite:

- En el caso del profesor, gestión de asignaturas, grupos y alumnos, consulta en tiempo real de la asistencia de alumnos, gestión manual de asistencias y consulta de estadísticas detalladas de asistencia.
- En el caso del alumno, envío automático de presencia en asignaturas y consulta de estadísticas detalladas de asistencia.

On this paper is presented a Final Degree Project on a Software Engineering Degree which has developed a software tool called Tutor, designed to facilitate teachers the class student attendances management.

The main objective of this project is modernize and optimize the common process routine that takes place in the subjects attendances control, reducing management time and centralizing all information in a dedicated tool.

This tool has been designed to be executed on smart mobile devices (smartphones) with iOS 7 or higher and requires an additional hardware device to take full advantage of its features. This additional hardware is an active beacon device which transmits a 100 meters radius Bluetooth LTE omni-directionally signal. The signal can determine uniquely a region in space that will be associated with a subject.

The control process involved in this system is based on detection of students within or outside the region of space which have been configured with the help of the active beacon. Once detected the location of the student, the tool automatically stores student inputs and outputs through the region.

In addition to the automatic attendance control system, it has been developed a manual attendance control system to perform management in cases where the absence of any technological requirement, as the active beacon by the teacher or as device iOS 7 in the case of the student, are an obstacle for the main task of this software tool.

Depending on your profile you log in, the tool allows to you:

- For teacher profile: subjects, groups and students management, manual attendances management, check out real-time student attendances and detailed attendance statistics.
- In case of student profile: send automatic inputs and outputs through the region and check out detailed attendance statistics.

1. INTRODUCCIÓN.....	17
1.1. Bases para el desarrollo y descripción.....	17
1.2. Objetivos generales.....	18
1.3. Objetivos específicos.....	19
2. HERRAMIENTAS UTILIZADAS.....	21
2.1. Software, Frameworks y bibliotecas.....	21
2.1.1. CodeIgniter.....	21
2.1.2. jQuery.....	21
2.1.3. Xcode.....	22
2.1.4. NetBeans y IntelliJ.....	22
2.1.5. Apache.....	22
2.1.6. MySQL.....	23
2.1.7. Creately y Astah.....	23
2.1.8. Bitbucket.....	23
2.2. Dispositivos hardware.....	24
2.2.1. Servidor.....	24
2.2.2. Dispositivos móviles.....	24
2.2.3. Active beacons.....	25
3. CICLO DE VIDA SOFTWARE.....	27
4. ANÁLISIS.....	29
4.1. Modelo de negocio.....	29
4.2. Requisitos.....	30
4.2.1. Requisitos funcionales.....	31
4.2.2. Requisitos no funcionales.....	33
4.3. Casos de Uso.....	34
4.4. Diagrama de Entidad-Relación Extendido.....	52
4.5. Diagrama de Clases.....	54
5. DISEÑO.....	55
5.1. Modelo de datos.....	55
5.2. Descripción de tablas.....	56
5.3. Diagramas de secuencia.....	61
5.3.1. Log In.....	62
5.3.2. Sign up.....	63
5.3.3. Email verification.....	64

5.3.4. Log out	64
5.3.5. Resend verification email	65
5.3.6. Forgot password	66
5.3.7. In-app purchase	67
5.3.8. Join group	68
5.3.9. Search subject groups	69
5.3.10. See all attendances records	69
5.3.11. See attendances graph	70
5.3.12. See automatic attendances log	70
5.3.13. Send failed automatic attendance requests	71
5.3.14. Set enter automatic attendance	72
5.3.15. Set exit automatic attendance	73
5.3.16. Unjoin group	74
5.3.17. Create and register student account	75
5.3.18. Register student into group	76
5.3.19. Search and register student	77
5.3.20. See all student attendance records	78
5.3.21. See student attendance graph	78
5.3.22. Set/unset attendance	79
5.3.23. Unregister student from group	80
5.3.24. Create group	81
5.3.25. Create subject	82
5.3.26. Delete group	83
5.3.27. Delete subject	84
5.3.28. Set/unset free day	85
5.3.29. Update group	86
5.3.30. Update subject	87

6. PRUEBAS89

6.1. Casos de prueba	89
6.1.1. Log in	89
6.1.2. Sign Up	90
6.1.3. Email verification	91
6.1.4. Resend email verification	91
6.1.5. Forgot password	92
6.1.6. Logout	93
6.1.7. Join group	93
6.1.8. Search subject groups	94
6.1.9. Unjoin group	94

6.1.10. Set enter automatic attendance	94
6.1.11. Set exit automatic attendance.....	95
6.1.12. See all attendance records.....	95
6.1.13. See attendances graph.....	95
6.1.14. See automatic attendances log.....	96
6.1.15. Send failed automatic attendances requests	96
6.1.16. See all student attendances records	96
6.1.17. Create and register student account	97
6.1.18. Search and register student.....	97
6.1.19. Register student into group.....	97
6.1.20. Unregister student from group	98
6.1.21. See student attendance graph.....	98
6.1.22. Set/unset attendance.....	98
6.1.23. Create subject	99
6.1.24. Update subject	99
6.1.25. Delete subject.....	99
6.1.26. Create group	100
6.1.27. Update group	100
6.1.28. Delete group.....	100
6.1.29. Set/unset free day.....	101
7. CONCLUSIONES	103
7.1. Herramienta software Tutor	103
7.2. Tecnología iBeacon y Bluetooth Low Energy	104
7.3. Futuras versiones del sistema	105
8. VÍAS DE CONTINUACIÓN DE ESTE PROYECTO.	107
8.1. Integración con plataforma de enseñanza Moodle.	107
8.2. Gestión académica integral.....	107
9. REFERENCIAS BIBLIOGRÁFICAS	109
10.GLOSARIO	111

1. Esquema de funcionamiento de iBeacons.....	17
2. Estimote beacons	25
3. Etapas del ciclo de vida en cascada.....	27
4. Casos de uso 1/4	35
5. Casos de uso 2/4	39
6. Casos de uso 3/4	44
7. Casos de uso 4/4	48
8. Diagrama Entidad-Relación Extendido	53
9. Diagrama de clases.....	54
10. Modelo de datos.....	55
11. Modelo local de Core Data.....	59
12. Diagrama de secuencia - Log In.....	62
13. Diagrama de secuencia - Sign up.....	63
14. Diagrama de secuencia - Email verification.....	64
15. Diagrama de secuencia - Log out	64
16. Diagrama de secuencia - Resend verification email	65
17. Diagrama de secuencia - Forgot password	66
18. Diagrama de secuencia - In-app purchase	67
18. Diagrama de secuencia - Join group.....	68
19. Diagrama de secuencia - Search subject groups.....	69
20. Diagrama de secuencia - See all attendances records	69
21. Diagrama de secuencia - See attendances graph	70
22. Diagrama de secuencia - See automatic attendances log	70
23. Diagrama de secuencia - Send failed automatic attendance requests	71
24. Diagrama de secuencia - Set enter automatic attendance.....	72
25. Diagrama de secuencia - Set exit automatic attendance	73
26. Diagrama de secuencia - Unjoin group.....	74
27. Diagrama de secuencia - Create and register student account.....	75
28. Diagrama de secuencia - Register student into group	76
29. Diagrama de secuencia - Search and register student.....	77
30. Diagrama de secuencia - See all student attendance records	78
31. Diagrama de secuencia - See student attendance graph	78
32. Diagrama de secuencia - Set/unset attendance	79
33. Diagrama de secuencia - Unregister student from group	80
34. Diagrama de secuencia - Create group.....	81
35. Diagrama de secuencia - Create subject.....	82
36. Diagrama de secuencia - Delete group.....	83
37. Diagrama de secuencia - Delete subject	84
38. Diagrama de secuencia - Set/unset free day	85
39. Diagrama de secuencia - Update group.....	86
40. Diagrama de secuencia - Update subject.....	87

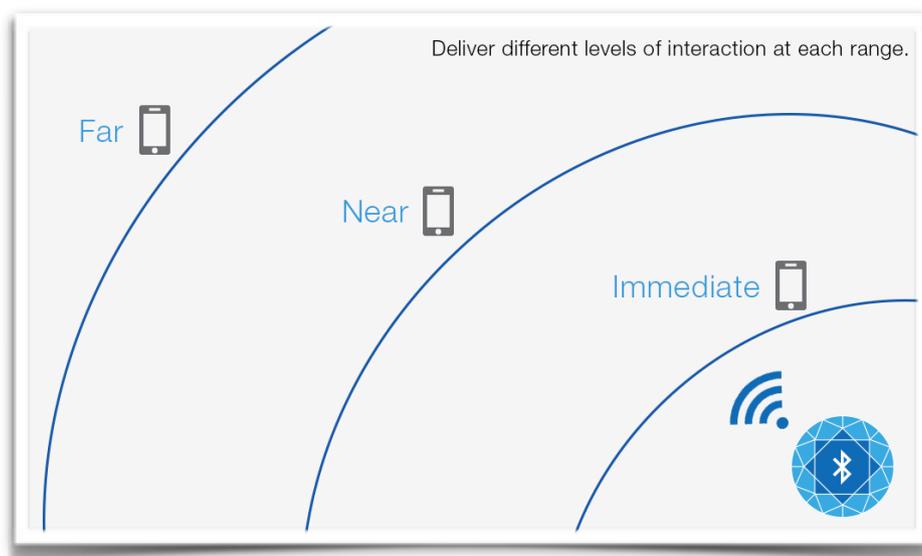
1. INTRODUCCIÓN.

1.1. Bases para el desarrollo y descripción.

Con la aparición de los smartphones y la enorme acogida por parte de los usuarios finales, el desarrollo de aplicaciones para dispositivos móviles ha llegado a convertirse en uno de los pilares fundamentales para la industria del desarrollo de software a nivel mundial en los días en los que se escriben estas líneas.

El auge producido por la llegada de los smartphones junto a la aparición de la tecnología iBeacons, desarrollada por Apple en su sistema operativo iOS, han proporcionado las bases idóneas para el desarrollo de este sistema o herramienta software que se presenta en este documento, Tutor.

iBeacons es una tecnología desarrollada para la localización de dispositivos a través de la señal Bluetooth LE ó 4.0 e intenta explotar las deficiencias que tiene el actual Sistema de posicionamiento Global (GPS) en interiores o áreas donde la señal recibida de los satélites GPS no es válida para una localización precisa. Esta tecnología se apoya en un pequeños dispositivos hardware denominados active beacons, encargados de emitir una señal omnidireccional constante a través de Bluetooth (de hasta 100 m. de radio), con los que posteriormente seremos capaces de determinar la posición relativa de los dispositivos móviles. La precisión de iBeacons es de menos de 10 cm (en óptimas condiciones).



1. Esquema de funcionamiento de iBeacons

El sistema Tutor, moderniza y optimiza el proceso de tradicional de control de asistencias de alumnos reduciendo los tiempos de gestión y centralizando toda la información en una herramienta dedicada. El sistema utiliza la señal de los active beacons para crear una región virtual en el espacio que asociaremos a un aula o un área física que queramos monitorizar. Una vez realizada la asociación región virtual-aula física, podremos conocer la posición relativa de todos los usuarios que se encuentren dentro de esta región.

Por último y para completar el proceso, la herramienta se encarga de gestionar todas las notificaciones de entrada y salida del aula que realiza cada alumno, y de esta manera, tener un historial completo disponible tanto para el profesor como para el alumno.

1.2. Objetivos generales.

El objetivo principal de este proyecto es el desarrollo de una herramienta software acorde a las necesidades actuales en educación, que permita por una lado a los profesores la gestión automatizada del control de asistencias de los alumnos, y por otra parte que permita a los alumnos de manera autónoma enviar al sistema toda la información de asistencia de las asignaturas que está cursando.

Esta herramienta adquiere un carácter relevante a la hora de gestionar las asistencias de los alumnos ya que, tanto en los diferentes bloques formativos obligatorios anteriores a la universidad como en la propia universidad desde 2009 con la introducción del Plan Bolonia, es obligatoria la asistencia a clase para conseguir ser evaluado por el conocimiento adquirido.

Esta herramienta ofrecerá a los profesores la posibilidad de olvidarse completamente de tener que ir nombrando alumno tras alumno para comprobar si ha acudido a clase o no, evitando de esta manera la consecuente pérdida de tiempo que conlleva este proceso. De esta manera, la herramienta se ofrece como una alternativa más eficiente al método tradicional de gestión de asistencias.

Alumnos y profesores en todo momento tendrán a su disposición un historial detallado de asistencias así como un gráfico global en el que se aprecia de manera rápida toda la información necesaria.

Esta herramienta software además de facilitar la gestión automatizada de asistencias, permitirá la veracidad de la asistencia del alumno ya que será necesaria la presencia física del alumno dentro de la región del espacio asociada a la asignatura.

1.3. Objetivos específicos.

- Explorar capacidades y posibilidades de un sistema de control de asistencia basado en BLE active beacons.
- Identificar ventajas y problemas del uso de la tecnología Bluetooth Low Energy para el control de asistencia en interiores.
- Desarrollo de una aplicación de control de asistencia específica para el control de asistencia de alumnos con las siguientes capacidades:
 - ▶ Gestión de asignaturas
 - ▶ Gestión grupos dentro de asignaturas
 - ▶ Gestión alumnos en grupos
 - ▶ Envío automático de información de asistencia a una asignatura
 - ▶ Proveer un mecanismo manual que permita realizar la tarea de gestión asistencias para los casos en los que no se dispongan de los requisitos mínimos de tecnología (el active beacon en caso del profesor o un dispositivo con iOS 7 o superior en caso del alumno) o para el caso de que el profesor prefiera utilizarlo en vez del automático.
 - ▶ Consulta del histórico de asistencias
- Documentar y realizar todas las fases del ciclo de vida de la herramienta software Tutor

2. HERRAMIENTAS UTILIZADAS.

En este apartado se describen las herramientas utilizadas en el desarrollo del proyecto.

Este apartado no pretende ser una descripción pormenorizada de cada herramienta, sino más bien una introducción necesaria para entender las bases tecnológicas en las que se apoya el proyecto y el motivo de su elección. Así mismo, para cada herramienta se incluirán los enlaces que amplían la información aquí recogida.

2.1. Software, Frameworks y bibliotecas.

2.1.1. CodeIgniter.

CodeIgniter es un framework para aplicaciones web de código abierto para crear sitios web dinámicos con PHP. Su objetivo es permitir que los desarrolladores puedan realizar proyectos mucho más rápido que creando toda la estructura desde cero, brindando un conjunto de bibliotecas para tareas comunes, así como una interfaz simple y una estructura lógica para acceder esas bibliotecas.

Se ha decidido utilizar este framework debido a la sencillez de uso, su rápido despliegue, el alto desempeño ofrecido y por los resultados muy positivos obtenidos en otros proyectos los que se utilizó dicho framework.

<http://www.codeigniter.com>

2.1.2. jQuery.

jQuery es una biblioteca de JavaScript que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web. jQuery es software libre y de código abierto, y al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

Se ha utilizado esta biblioteca en el desarrollo de la página web informativa de la herramienta debido a su sencillez de uso y a las funcionalidades que ofrece.

<http://jquery.com>

2.1.3. Xcode.

Xcode es el IDE de Apple y se suministra gratuitamente junto con Mac OS X. Xcode trabaja conjuntamente con Interface Builder, una herencia de NeXT, una herramienta gráfica para la creación de interfaces de usuario.

Xcode incluye la colección de compiladores del proyecto GNU (GCC), y puede compilar código C, C++, Objective-C, Objective-C++, Java, AppleScript y Swift mediante una amplia gama de modelos de programación, incluyendo, pero no limitado a Cocoa, Carbón y Java. Otras compañías han añadido soporte para GNU Pascal, Free Pascal, Ada y Perl.

<https://developer.apple.com/xcode/>

2.1.4. NetBeans y IntelliJ.

NetBeans e IntelliJ son IDE que permiten el desarrollo en diferentes lenguajes como son Java, PHP, C,C++, HTML y Groovy.

Se han utilizado estos IDE para el desarrollo de la API escrita en PHP y para la web informativa. La elección de estos IDE se debe a los conocimientos previos que poseo en ambos, la facilidad de uso y en general, debido a las capacidades adicionales que ofrecen como puedan ser cliente FTP, gestor de base de datos, servidores HTTP integrados, etc.

<https://netbeans.org>

2.1.5. Apache.

Apache es un servidor HTTP de código abierto. En la actualidad es uno de los servidores HTTP más utilizado en estos momentos y se puede ejecutar local y remotamente en plataformas Unix (BSD, GNU/Linux, etc.), Macintosh y Microsoft . Este servidor cuenta entre otras características altamente configurables, bases de datos de autenticación y negociado de contenido.

El servidor donde se ha implementado la API de Tutor cuenta con una instalación bien configurada y probada del servidor Apache, por lo que se ha decidido utilizar dicha instalación.

<http://httpd.apache.org>

2.1.6. MySQL.

MySQL es el sistema gestor de base de datos relacional elegido para el desarrollo de la herramienta. La facilidad de uso y el desempeño ofrecido por este sistema han sido las principales razones para su elección.

El cliente utilizado ha sido MySQLWorkbench para la gestión de la base de datos, scripts y modelado de datos.

<http://www.mysql.com>

<http://www.mysql.com/products/workbench/>

2.1.7. Creately y Astah.

Creately y Astah son las herramientas elegidas para el modelado de los diagramas del proyecto.

Astah se ha utilizado principalmente para el modelado de diagramas de secuencia y Creately para el resto de diagramas y figuras.

<http://creately.com>

<http://astah.net>

2.1.8. Bitbucket

Se han utilizado diferentes repositorios GIT para el almacenamiento del código. Se ha utilizado la plataforma Bitbucket de Atlassian ya que es gratuita y permite repositorios privados.

2.2. Dispositivos hardware.

2.2.1. Servidor.

La máquina encargada de gestionar el backend del proyecto, es un servidor potente, robusto y seguro contratado a la empresa de alojamiento Hostingular. Este servidor está ubicado en Amsterdam y se encarga de la gestión de todos los servicios del backend, Apache, MySQL e interprete PHP. Además, en dicho servidor está alojada la página informativa del proyecto.

Todas las peticiones realizadas al servidor a través de la API, se realizan a través de protocolos seguros SSL (HTTPS).

La principales características del servidor son:

- Sistema operativo CentOS 6
- 8 procesadores Intel(R) Xeon(R) CPU E5620 @ 2.40GHz con 12.288 KB de caché
- 32 Gb de memoria RAM

2.2.2. Dispositivos móviles.

Para la realización de las pruebas del proyecto, además de los diferentes simuladores software que proporciona Xcode, se han utilizado diferentes dispositivos iPhone. Se han realizado numerosas pruebas en cada uno de ellos y en las diferentes iteraciones del proyecto se ha ido solventando los errores encontrados.

- iPhone 5, 5S y 7 Plus
- iPad 3ª generación

2.2.3. Active beacons.

Para el desarrollo del proyecto, previamente se comentó la utilización de pequeños dispositivos bluetooth para la creación de regiones virtuales y su posterior asociación con asignaturas. Estos dispositivos son active beacon.

En el mercado actual se pueden encontrar numerosos fabricantes de estos dispositivos. Para la herramienta Tutor se han elegido los active beacon fabricados por Estimote.



2. Estimote beacons

Las principales características de los active beacon de Estimote son:

- 3 años de duración de batería.
- ARM M0 Cortex, Bluetooth Low Energy, sensores de temperatura.
- UUID configurable y rango de emisión de hasta 70 metros.
- Actualizaciones de firmware y de seguridad OTA.
- Compatibilidad con iOS y Android.
- SDK usado por más de 50.000 desarrolladores.

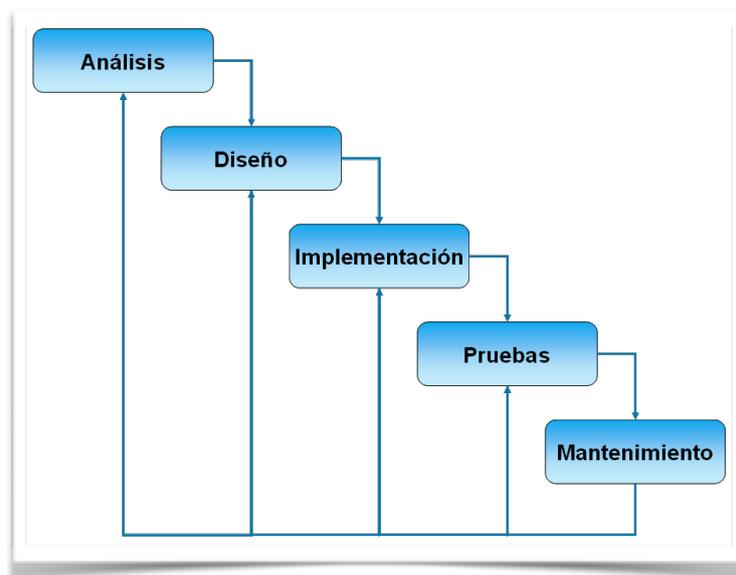
<http://estimote.com>

3. CICLO DE VIDA SOFTWARE.

En la actualidad, existen numerosos ciclos de vida para el desarrollo de un proyecto software, por lo que la tarea de elegir un ciclo u otro, podría afectar en gran medida al éxito del proyecto, ya que cada uno presenta ventajas y desventajas que se verán reflejadas en el producto final. En general, tenemos ciclos de vida en cascada, iterativos, en espiral, en V, orientado a objetos, etc.

Para el desarrollo de este proyecto se ha decidido utilizar un ciclo de vida en cascada y en este apartado enunciarán las razones de dicha elección y las principales ventajas e inconvenientes encontrados en el desarrollo de con este tipo de ciclo.

El ciclo de vida en cascada es el enfoque metodológico que ordena las etapas del proceso para el desarrollo de software, de tal forma que el inicio de cada etapa debe esperar a la finalización de la etapa anterior.



3. Etapas del ciclo de vida en cascada

De esta forma, cualquier error de diseño detectado en la etapa de prueba conduce necesariamente a una nueva iteración o rediseño y la programación del código afectado, aumentando los costos o tiempo del desarrollo.

Si bien ha sido ampliamente criticado desde el ámbito académico y la industria, sigue siendo uno de los paradigmas más seguido al día de hoy.

La principal razón que ha llevado a la elección del ciclo de vida en cascada para el desarrollo de este proyecto, es el conocimiento de todos los requisitos al comienzo del mismo. El objetivo del proyecto se conoce a la perfección así como todas las características que se desean implementar. Otra razón de peso para la elección de este ciclo es la sencillez de aplicación, ya que es un ciclo sencillo de comprender y aplicar. Por último, el tamaño del proyecto, que aun siendo un proyecto que cuenta con unas 18.000-20.000 líneas de código, es perfectamente viable con este tipo de ciclo de vida.

Las principales ventajas encontradas al ciclo de vida en cascada tras la finalización del proyecto son la sencillez de aplicación y de organización desde el comienzo del proyecto. Es un ciclo sencillo de aprender y debido a su popularidad existe numerosa documentación para contrastar. Otra ventaja que ofrece este ciclo es la detección de errores en las primeras etapas y su posible corrección sin aumentar en exceso el tiempo de desarrollo.

Algunos inconvenientes encontrados son que los resultados no los veremos hasta que no estemos en las etapas finales del ciclo. Además, es un ciclo de vida poco flexible para el cambio o adición de requisitos.

Aún teniendo los requisitos muy claros desde el comienzo del proyecto, a lo largo del desarrollo fueron surgiendo nuevos requisitos e ideas que mejoraban la herramienta dotándola de mayor funcionalidad. Por este motivo, se han desarrollado y publicado en el App Store diferentes versiones hasta llegar a la versión final 1.3 que es la que se presenta en este documento.

4. ANÁLISIS

En este apartado del documento se tratará en detalle la especificación formal del modelo de negocio, requisitos y casos de uso.

Esta información será representada en algunos casos mediante el lenguaje UML, por ejemplo para los casos de uso, o bien textualmente como para la definición del modelo de negocio o los requisitos.

4.1. Modelo de negocio

Tras un exhaustivo análisis del proceso de control de asistencia de alumnos, se ha obtenido suficiente información para elaborar la siguiente documentación que define el modelo de negocio que se desea desarrollar.

El sistema Tutor será una aplicación para dispositivos móviles con sistema operativo iOS 7 (o superior) en el que los profesores podrán gestionar las asistencias de sus alumnos de manera rápida y efectiva. Para tal propósito, una vez completado el registro en el sistema, el profesor desde su perfil podrá gestionar las asignaturas, grupos y alumnos. Los alumnos, una vez completado el registro en el sistema, podrán enviar notificaciones al sistema para indicar su asistencia a los grupos de las asignaturas en las que están matriculados.

La novedad planteada por este sistema, frente al método tradicional, es la inclusión de nuevas herramientas software y hardware que permiten la automatización de este proceso evitando pérdidas de tiempo innecesario y la centralización de la gestión en una herramienta dedicada.

El proceso de gestión de asistencias de este sistema se ha pensado para que puedan convivir el proceso de gestión tradicional y el proceso automático desarrollado en este proyecto. Este planteamiento es fundamental, ya que permite la integración desde el primer momento de todos los implicados en el sistema, posean los requisitos necesarios de software/hardware o no.

El primer paso que se debe realizar para comenzar la gestión de asignaturas es establecer la estructura del sistema. Este primer paso requiere que el profesor complete la información solicitada para la creación de asignaturas asociadas a active beacons y posteriormente de grupos para dichas asignaturas. Una vez completada dicha información, el profesor podrá matricular a sus alumnos en los diferentes grupos o bien ser los propios alumnos que se matriculen por medio de una contraseña que les proporcione el profesor.

Posteriormente, cuando el profesor vaya a comenzar la lección, deberá portar consigo el active beacon asociado a la asignatura para que el sistema detecte la entrada y salida de alumnos en el aula. Este proceso se realiza de manera automática por el sistema cuando se detecta que el alumno está dentro o fuera del rango de señal del active beacon.

El profesor dispondrá de un listado de alumnos por cada grupo en el que aparece el estado actual de los alumnos (presente o ausente). Estos estados se pueden actualizar en tiempo real por los alumnos, según vayan enviando las notificaciones al sistema al entrar en clase o bien de manera manual por el profesor en el caso de que el alumno no disponga de un dispositivo móvil para enviar la notificación.

Ambos perfiles, profesor y alumno, pueden conocer el estado de presencia en el aula en tiempo real ya que se realiza de manera automática y sin intervención de los usuarios. No obstante, y llegado a este punto, el profesor podría revisar el listado de manera tradicional para añadir a aquellos alumnos que no dispongan del sistema en su dispositivo móvil.

4.2. Requisitos

En este apartado se presentan los requisitos que deben ser satisfechos por la herramienta de control de asistencias Tutor. Los requisitos aquí expuestos son esenciales para el correcto funcionamiento del sistema.

Para la denominación de los requisitos utilizaremos la siguiente nomenclatura:

RFXXX-X

El estándar seguido para la especificación del identificador de cada requisito funcional será de la siguiente manera:

- **R** = Requisito
- **F** = Funcional
- **XXX-X** = secuencia de dígitos que servirá para la enumeración de cada requisito.

4.2.1. Requisitos funcionales

Requisitos generales de usuario

RF001	Creación de cuentas de usuario.
Descripción:	Este requisito hace referencia a la posibilidad que tendrán todos los usuarios del sistema para crear una nueva cuenta. El usuario deberá completar todos los campos requeridos. Cada cuenta de usuario debe de estar relacionada con un tipo de perfil, bien profesor o bien alumno, siendo posible que una cuenta de usuario posea ambos perfiles asociados, es decir, un usuario puede ser profesor y alumno.

RF002	Restablecimiento de contraseña.
Descripción:	Este requisito hace referencia a la posibilidad que tendrá un usuario registrado para restablecer su contraseña. Para dotar al sistema de mayor seguridad, el restablecimiento de contraseña se realizará a través de un enlace enviado por correo electrónico.

RF003	Verificación del email de usuario.
Descripción:	Con este requisito se pretende la verificación del email de cada usuario. Con esta medida la herramienta adquiere un grado de seguridad adicional evitando en la medida de lo posible la suplantación de identidad.

RF004	Acceso al sistema seguro.
Descripción:	Este requisito hace referencia al acceso de los perfiles de usuario en la herramienta. Cualquier usuario que quiera acceder a su perfil, deberá introducir el email y contraseña correctamente.

Requisitos orientados al perfil de profesor

RF005	Gestión de asignaturas
Descripción:	Este requisito hace referencia a la gestión de las asignaturas. Para una correcta gestión, se deberán implementar los requisitos adicionales que permiten todas las operaciones CRUD.
RF005-1	Agregar asignatura
RF005-2	Consultar asignatura
RF005-3	Editar asignatura
RF005-4	Eliminar asignatura

RF006	Gestión de grupos
Descripción:	Este requisito hace referencia a la gestión de los grupos. Para una correcta gestión, se deberán implementar los requisitos adicionales que permiten todas las operaciones CRUD y los requisitos adicionales especificados en la siguiente tabla.
RF006-1	Agregar grupo
RF006-2	Consultar grupo
RF006-3	Editar grupo
RF006-4	Eliminar grupo
RF006-5	El profesor podrá consultar el listado de alumnos y su estado (presente o ausente) para cada grupo.
RF006-6	El profesor podrá añadir y eliminar alumnos ya existentes a los grupos
RF006-7	Modificar días festivos

RF007	Gestión de alumnos
Descripción:	Este requisito hace referencia a la gestión de alumnos. Para una correcta gestión, se deberán implementar los requisitos adicionales especificados en la siguiente tabla.
RF007-1	Creación de alumnos
RF007-2	El profesor podrá editar el estado (presente o ausente) de los alumnos de cada grupo manualmente
RF007-3	El profesor podrá consultar estadísticas detalladas y globales de asistencia para cada alumno

Requisitos orientados al perfil de alumno

RF008	Automatricula
Descripción:	El alumno podrá automatricularse en los grupos de las asignaturas que los profesores hayan creado introduciendo el código de acceso

RF009	Envío de notificaciones
Descripción:	El alumno podrá enviar notificaciones automáticas de presencia en una asignatura

RF010	Listado de asistencias
Descripción:	El alumno podrá consultar estadísticas detalladas y globales de asistencia para cada asignatura

RF011	Listado de notificaciones enviadas
Descripción:	El alumno podrá consultar estadísticas detalladas de envíos automáticos de asistencia

4.2.2. Requisitos no funcionales

Requisitos de rendimiento

El sistema ofrecerá un sistema óptimo de navegación por las diferentes secciones, evitando la ambigüedad y ofreciendo todas las opciones disponibles para cada elemento.

El rendimiento del sistema será fluido optimizando las operaciones internas para evitar tiempos de espera innecesarios.

Las operaciones en las que intervenga una comunicación directa con el servidor, serán optimizadas para evitar el envío de información redundante o innecesaria a través de la red, consiguiendo de esta manera una comunicación eficiente con el servidor. Además, en óptimas condiciones de conexión WI-FI, no se superará los 15 segundos en las operaciones CRUD. Para el resto de conexiones dependerá del tipo de conexión y cantidad de datos a transferir.

Requisitos de seguridad

El sistema será accesible solamente por usuarios registrados y verificados.

La información almacenada en el sistema estará protegida contra accesos no autorizados.

La información que se transmite en las comunicaciones realizadas entre los diferentes terminales y el servidor deberá ser cifrada con protocolos seguros SSL/TLS.

Requisitos de interfaz

El sistema deberá seguir los estilos de interfaz recomendados por Apple en el documento iOS Human Interface Guidelines. Estos estilos garantizan una increíble experiencia de usuario combinando consistencia, fácil aprendizaje, legibilidad, óptimas metáforas y productividad.

Nota: Para acceder al documento iOS Human Interface Guidelines consultar el enlace proporcionado en el capítulo Bibliografía de este documento.

Requisitos de hardware

Para la explotación del sistema en su máxima capacidad, es decir, control de asistencias automatizado a través de active beacon, los requisitos hardware mínimos serán:

- Todo usuario del sistema (profesor y alumno) necesitará un dispositivo móvil con sistema iOS 7 (o superior) que disponga de Bluetooth 4.0 y conectividad a internet.
- El profesor necesitará al menos un active beacon.

Para la explotación del sistema de forma manual o tradicional, los requisitos hardware mínimos serán:

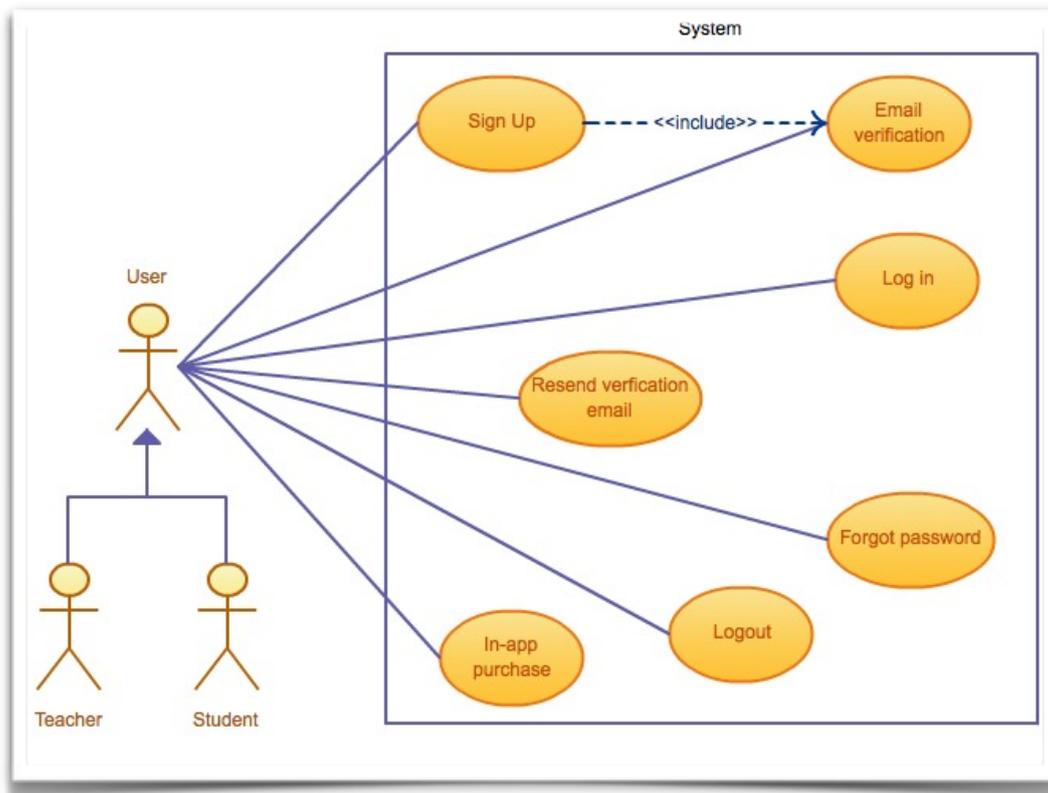
- El profesor necesitará un dispositivo móvil con sistema iOS 7 (o superior) y conectividad a internet.

4.3. Casos de Uso

En este apartado se muestra el diagrama de casos de uso. Este diagrama se ha separado en 4 subdiagramas para la correcta visualización en este documento.

El diagrama de casos de uso se ha representado de manera formal utilizando UML. Así mismo, para cada caso de uso del diagrama se realizará la definición textual indicando el flujo de eventos de alto nivel. Para los casos de uso más complejos, se ha decidido adjuntar los diagramas de actividad para tener una visión completa y pormenorizada del proceso.

Para el primer subdiagrama, se ha decidido agrupar los casos de uso comunes para los perfiles de profesor y alumno, señalando la relación de especialización entre los diferentes perfiles. Para el segundo y sucesivos subdiagramas, se especifican los casos de uso para perfiles concretos de usuario.



4. Casos de uso 1/4

Sign Up		
Actores:	User	
Descripción:	El acceso al sistema está restringido a usuarios registrados. Si un usuario desea interactuar con el sistema debe realizar este proceso de registro o sign up.	
Precondición:	N/A	
Postcondición:	Caso de uso: Email verification	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Elegir opción: Nueva cuenta > Perfil 3. Insertar datos en el formulario 6. Enviar datos	2. Mostrar formulario de registro 4. Validar datos del formulario 5. Activar botón enviar 7. Verificar que el email está bien escrito 8. Procesar creación de cuenta 9. Mensaje al usuario
Excepciones:	<ul style="list-style-type: none"> - Si los datos del formulario no son válidos, no se activará el botón de enviar - Si la verificación del email no es válida, no se continuará con el flujo - Si el email ya está registrado para ese perfil, se muestra mensaje al usuario - Si ocurre algún error en el proceso de creación, se muestra mensaje al usuario 	

Email verification		
Actores:	User	
Descripción:	Mecanismo de verificación del email del usuario. El email se verifica una sola vez, es decir, si el usuario ya posee un email verificado y crea otro perfil diferente, no será necesaria la verificación.	
Precondición:	El usuario debe de estar registrado en el sistema y su email no se ha verificado todavía.	
Postcondición:	Habilita el acceso al sistema o log in	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Acceder al enlace proporcionado en el correo de bienvenida 3. Insertar datos en el formulario 4. Enviar datos	2. Mostrar formulario para insertar contraseña 5. Validar datos del formulario 6. Procesar verificación 7. Mostrar mensaje al usuario
Excepciones:	<ul style="list-style-type: none"> - Si los datos del formulario no son válidos, no se continua con el flujo - Si ocurre algún error en el proceso de verificación, se muestra mensaje al usuario 	

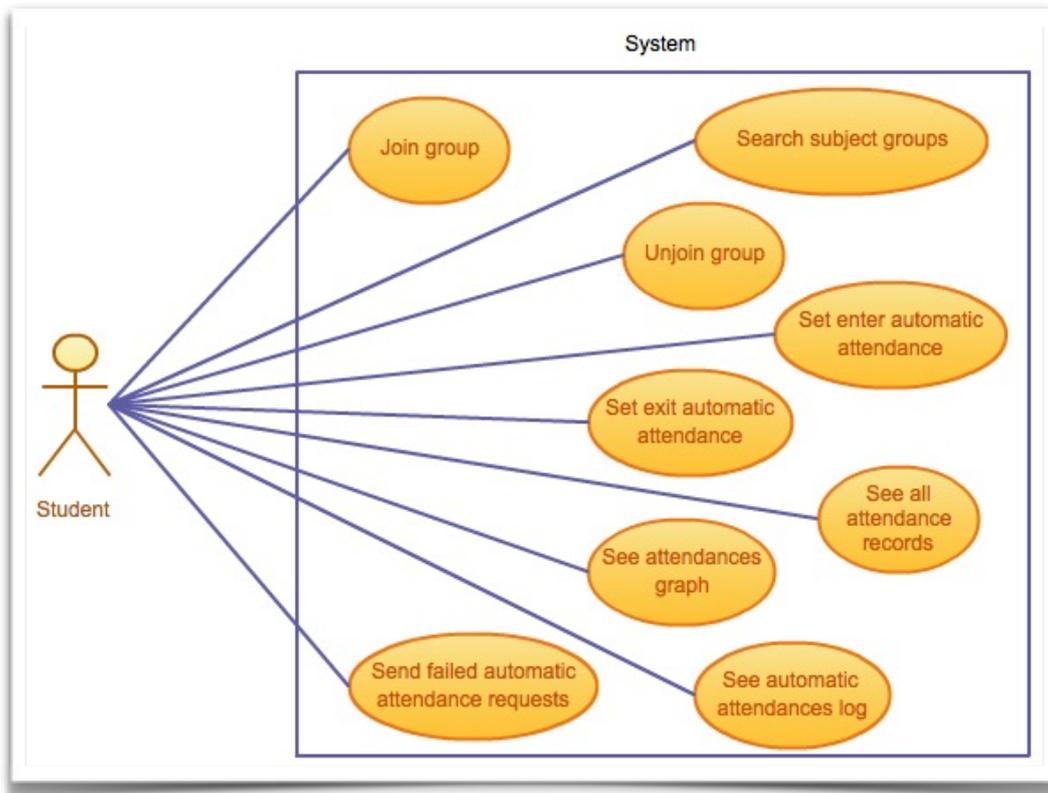
Log in		
Actores:	User	
Descripción:	Acceso al sistema mediante las credenciales de usuario.	
Precondición:	Para completar acceso es necesario ser usuario registrado previamente	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Insertar datos en el formulario de acceso 4. Enviar datos	2. Validar datos del formulario 3. Activar botón de acceso 5. Procesar acceso 6. Mostrar mensaje al usuario 7. Mostrar perfil usuario
Excepciones:	<ul style="list-style-type: none"> - Si los datos del formulario no son válidos, no se continua con el flujo - Si el usuario no está verificado, no podrá acceder al sistema y se mostrará un mensaje de información - Si el usuario no se ha registrado previamente, se mostrará un mensaje de información - Si ocurre algún error en el proceso de acceso, se muestra mensaje al usuario 	

Resend verification email		
Actores:	User	
Descripción:	El usuario quiere verificar su identidad (email) pero no dispone del email que el sistema envía automáticamente en el registro de usuarios. Al realizar esta operación, el usuario recibirá de nuevo dicho email con el link de verificación.	
Precondición:	Para recibir el email es necesario haber realizado el registro previo en el sistema.	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Elegir opción: Reenviar email de verificación 2. Insertar email 5. Enviar datos	3. Validar email 4. Activar botón enviar 6. Procesar envío 5. Mostrar mensaje al usuario
Excepciones:	<ul style="list-style-type: none"> - Si el email no es válido, no se activará el botón de enviar - Si el usuario no está registrado o ya está verificado, se muestra mensaje al usuario - Si ocurre algún error en el proceso de reenvío, se muestra mensaje al usuario 	

Forgot password		
Actores:	User	
Descripción:	El usuario quiere acceder al sistema pero no recuerda su contraseña o simplemente desea modificarla. Al realizar esta operación, el usuario recibirá un email con las instrucciones para recuperar su contraseña.	
Precondición:	Para restablecer la contraseña es necesario ser usuario registrado	
Postcondición:	Es necesario seguir el enlace del correo y completar las instrucciones	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Elegir opción: Contraseña olvidada 2. Insertar email 5. Enviar datos	3. Validar email 4. Activar botón enviar 6. Procesar envío 5. Mostrar mensaje al usuario
Excepciones:	<ul style="list-style-type: none"> - Si el email no es válido, no se activará el botón de enviar - Si el usuario no está registrado, se muestra mensaje al usuario - Si el usuario no está verificado, se muestra mensaje al usuario - Si ocurre algún error en el proceso de reenvío, se muestra mensaje al usuario 	

Log out		
Actores:	User	
Descripción:	El usuario quiere terminar la sesión de su perfil actual. Al realizar esta operación, si el usuario quiere volver a gestionar su perfil tendrá que acceder de nuevo al sistema.	
Precondición:	El usuario ha accedido a su perfil del sistema	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Elegir opción: Salir	2. Mostrar formulario de acceso
Excepciones:	N/A	

In-app purchase		
Actores:	User	
Descripción:	El usuario desea realizar una compra in-app para desbloquear las restricciones de usuario impuestas o eliminar la publicidad.	
Precondición:	El usuario ha accedido a su perfil del sistema	
Postcondición:	Será necesario repetir el proceso de acceso al sistema para aplicar los cambios de algunos ítems	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Elegir opción: Compras 3. Seleccionar un ítem para comprar	2. Mostrar formulario de compras 4. Validación de usuario e ítem 5. Procesar pedido 6. Mostrar mensaje al usuario 7. Desbloquear ítem comprado 8. Mostrar mensaje al usuario
Excepciones:	<ul style="list-style-type: none"> - Si no se completa la validación de usuario e ítem, no se continua con el flujo - Si ocurre algún error en el proceso compra, se muestra mensaje al usuario 	



5. Casos de uso 2/4

Join group		
Actores:	Student	
Descripción:	El usuario, en este caso el alumno, desea automatricularse en un grupo de una asignatura. El usuario inicia sesión en su perfil e inicia el proceso.	
Precondición:	El usuario ha ingresado en la herramienta y además, ha completado la búsqueda de asignaturas satisfactoriamente (caso de uso Search subject groups). La herramienta muestra un listado de los grupos disponibles de la asignatura.	
Postcondición:	Al seleccionar una asignatura no se mostrará ningún grupo si el profesor no ha creado algún grupo para dicha asignatura.	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Seleccionar grupo 2. Insertar datos (código acceso a grupo) 3. Enviar datos	4. Validar datos 5. Procesar matrícula 6. Mostrar mensaje 7. Mostrar asignaturas matriculadas
Excepciones:	<ul style="list-style-type: none"> - Si el código de acceso al grupo no coincide con el que ha establecido el profesor se detendrá el flujo. - Si el alumno está matriculado en un grupo de una asignatura e intenta matricularse en otro grupo diferente de la misma asignatura, se muestra un mensaje informativo. - Si ocurre algún error en el proceso de matrícula, se muestra mensaje al usuario 	

Search subject groups		
Actores:	Student	
Descripción:	El usuario, en este caso el alumno, desea realizar una búsqueda de asignaturas disponibles. El usuario inicia sesión en su perfil e inicia el proceso.	
Precondición:	<ul style="list-style-type: none"> - La asignatura debe haber sido creada por el profesor. - El alumno debe conocer el código de asignatura. 	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Elegir opción: Nueva asignatura 3. Insertar datos (código de asignatura) 4. Enviar datos 6. Seleccionar asignatura	2. Mostrar buscador de asignaturas 5. Mostrar resultados de asignaturas 7. Mostrar grupos de la asignatura
Excepciones:	<ul style="list-style-type: none"> - Al buscar asignaturas no se mostrará ningún resultado si el código de asignatura no coincide con algún registro en el sistema. - Si la asignatura no tiene ningún grupo, no se muestran resultados - Si ocurre algún error en el proceso de búsqueda, se muestra mensaje al usuario 	

Unjoin group		
Actores:	Student	
Descripción:	El usuario, en este caso el alumno, desea eliminar la matricula de una asignatura. El usuario inicia sesión en su perfil e inicia el proceso.	
Precondición:	El usuario debe de estar matriculado en al menos una asignatura para poder realizar este proceso.	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Seleccionar opción eliminar 2. Enviar datos	3. Procesar petición 4. Mostrar mensaje
Excepciones:	<ul style="list-style-type: none"> - Si ocurre algún error en el proceso de desmatriculación, se muestra mensaje al usuario 	

Set enter automatic attendance		
Actores:	Student	
Descripción:	El alumno entra a un aula de una asignatura asignada a un active beacon y el sistema se encarga automáticamente de almacenar la hora de llegada y la notificación al profesor.	
Precondición:	<ul style="list-style-type: none"> - El usuario debe de estar matriculado en al menos una asignatura controlada con active beacon para poder realizar este proceso. - El alumno debe portar consigo el dispositivo móvil con bluetooth activado para que se detecte la entrada. 	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	Nota: El alumno debe de entrar en un asignatura monitorizada para que se inicie el proceso	<ol style="list-style-type: none"> 1. Detectar la entrada a la asignatura 2. Obtener la hora real en internet 3. Procesar entrada 4. Mostrar notificación
Excepciones:	<ul style="list-style-type: none"> - Debido a la velocidad de la conexión del dispositivo, es posible que la notificación de entrada no se realice correctamente. En este caso, la notificación quedará almacenada y con estado "pendiente", para que el alumno posteriormente cuando tenga una adecuada conexión a internet pueda validarla. - Si el dispositivo no tiene bluetooth encendido no se detectará la entrada a la asignatura y no comenzará el proceso. - Si el dispositivo no tiene conexión a internet, se mostrará una notificación al usuario y no se realizará ninguna acción. 	

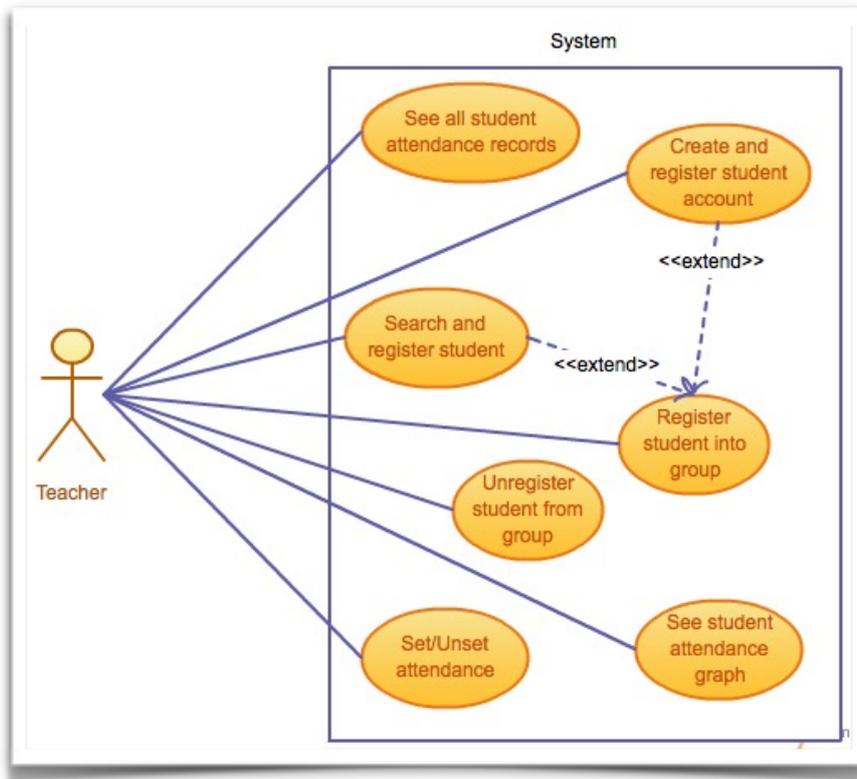
See all attendance records		
Actores:	Student	
Descripción:	El alumno quiere consultar el histórico de asistencias que lleva completadas para una asignatura concreta	
Precondición:	El usuario ha accedido a su perfil del sistema	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	<ol style="list-style-type: none"> 1. Seleccionar asignatura 3. Seleccionar todos los registros de asistencia 	<ol style="list-style-type: none"> 2. Mostrar opciones 4. Mostrar los registros de asistencia almacenados en el servidor
Excepciones:	<ul style="list-style-type: none"> - Si ocurre algún error a la hora de mostrar los registros, se muestra mensaje al usuario 	

Set exit automatic attendance		
Actores:	Student	
Descripción:	El alumno sale de un aula de una asignatura asignada a un active beacon y el sistema se encarga automáticamente de almacenar la hora de salida y la notificación al profesor.	
Precondición:	<ul style="list-style-type: none"> - El usuario debe de estar matriculado en al menos una asignatura controlada con active beacon para poder realizar este proceso. - El alumno debe portar consigo el dispositivo móvil con bluetooth activado. - El dispositivo debe de tener conexión a internet. - El alumno debe haber entrado previamente al aula y tener un registro de entrada. 	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	Nota: El alumno debe de salir del área monitorizada para que se inicie el proceso.	<ol style="list-style-type: none"> 1. Detectar la salida de la asignatura 2. Comprobar registro de entrada 3. Obtener la hora real en internet 4. Procesar salida 5. Mostrar mensaje
Excepciones:	<ul style="list-style-type: none"> - Debido a la velocidad de la conexión del dispositivo, es posible que la notificación de entrada no se realice correctamente. En este caso, la notificación quedará almacenada y con estado "pendiente", para que el alumno posteriormente cuando tenga una adecuada conexión a internet pueda validarla. - Si el dispositivo no tiene bluetooth encendido no se detectará la entrada a la asignatura y no comenzará el proceso. - Si el dispositivo no tiene conexión a internet, se mostrará una notificación al usuario. 	

See attendances graph		
Actores:	Student	
Descripción:	El alumno quiere consultar el gráfico global de asistencias que lleva completadas para una asignatura concreta	
Precondición:	El usuario ha accedido a su perfil del sistema	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	<ol style="list-style-type: none"> 1. Seleccionar asignatura 3. Seleccionar gráfico de asistencias 	<ol style="list-style-type: none"> 2. Mostrar opciones 4. Procesar petición en servidor 5. Mostrar gráfico de asistencias
Excepciones:	<ul style="list-style-type: none"> - Si ocurre algún error a la hora de mostrar el gráfico, se muestra mensaje al usuario 	

See automatic attendances log		
Actores:	Student	
Descripción:	El alumno quiere consultar el registro de asistencias enviadas automáticamente al sistema.	
Precondición:	El usuario ha accedido a su perfil del sistema	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Seleccionar asignatura 3. Seleccionar registros de asistencia automáticos	2. Mostrar opciones 4. Mostrar registros automáticos
Excepciones:	- Si ocurre algún error a la hora de mostrar los registros, se muestra mensaje al usuario	

Send failed automatic attendance requests		
Actores:	Student	
Descripción:	El alumno tiene un registro pendiente de envío en el log de asistencias automáticas y desea enviarlo. Solo se podrán enviar los registro pendientes.	
Precondición:	<ul style="list-style-type: none"> - El usuario ha accedido a su perfil del sistema - Existe un registro de asistencia pendiente de envío al sistema en el log. - El dispositivo tiene conexión a internet - El usuario a accedido al log de asistencias automáticas (caso de uso See automatic attendances log) 	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Seleccionar un registro pendiente 3. Seleccionar opción: enviar	2. Mostrar opciones 4. Procesar petición en API 5. Actualizar registro de asistencia a confirmado 6. Mostrar listado
Excepciones:	- Si ocurre algún error a la hora de mostrar los registros, se muestra mensaje al usuario	



6. Casos de uso 3/4

See all student attendance records		
Actores:	Teacher	
Descripción:	El profesor desea consultar el histórico de asistencias de uno de sus alumnos.	
Precondición:	<ul style="list-style-type: none"> - El profesor ha accedido a su perfil del sistema - El profesor debe tener al menos un alumno matriculado en un grupo de una asignatura. Se muestra el listado de alumnos del grupo 	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	<ol style="list-style-type: none"> 1. Seleccionar alumno (Mantener presionado durante un segundo) 3. Seleccionar 'Ver asistencias' 	<ol style="list-style-type: none"> 2. Mostrar opciones de alumno 4. Procesar petición en API 5. Mostrar listado de asistencias
Excepciones:	<ul style="list-style-type: none"> - Si ocurre algún error a la hora de mostrar los registros de asistencia, se muestra mensaje al usuario 	

Search and register student		
Actores:	Teacher	
Descripción:	El profesor desea buscar un alumno para incluirlo en uno de los grupos disponibles. En general, serán los alumnos los que se matriculen en los grupos sin la intervención del profesor. Se ha creado este caso de uso para facilitar al profesor la posibilidad de matricular a un alumno manualmente en caso de el alumno no posea un dispositivo móvil compatible o en caso de que no pueda realizar la matriculación por el mismo.	
Precondición:	<ul style="list-style-type: none"> - El profesor ha accedido a su perfil del sistema - El profesor tiene creado al menos un grupo y se muestra el listado de alumnos 	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Seleccionar opción 'añadir' 3. Seleccionar opción 'buscar' 5. Introducir código alumno y presionar buscar 7. Seleccionar alumno y enviar	2. Mostrar opciones para añadir 4. Mostrar formulario de búsqueda 6. Mostrar resultados de la búsqueda 8. Procesar matrícula 9. Mensaje de información
Excepciones:	<ul style="list-style-type: none"> - Si ocurre algún error a la hora de mostrar los resultados de la búsqueda, se muestra mensaje al usuario - Si el alumno ya está matriculado en el grupo o en otro grupo diferente de la asignatura, no se realizará la nueva matrícula 	

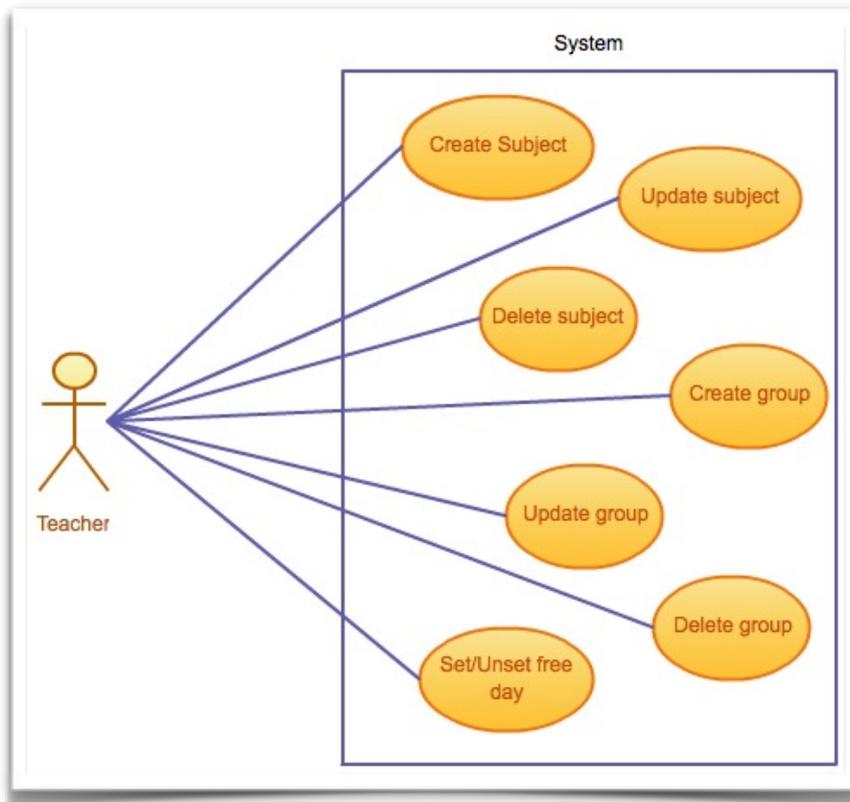
Create and register student account		
Actores:	Teacher	
Descripción:	El profesor desea crear un alumno para incluirlo en uno de los grupos disponibles. En general, serán los alumnos los que se matriculen en los grupos sin la intervención del profesor. Se ha creado este caso de uso para facilitar al profesor la posibilidad de matricular a un alumno manualmente en caso de el alumno no posea un dispositivo móvil compatible o en caso de que no pueda realizar la matriculación por el mismo.	
Precondición:	<ul style="list-style-type: none"> - El profesor ha accedido a su perfil del sistema - El profesor tiene creado al menos un grupo y se muestra el listado de alumnos 	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Seleccionar opción 'añadir' 2. Seleccionar opción 'Nuevo' 5. Introducir datos del formulario 8. Enviar datos	2. Mostrar opciones para añadir 4. Mostrar formulario de creación 6. Validar datos del formulario 7. Activar botón enviar 9. Procesar creación de cuenta y matriculación en el grupo 10. Mensaje al usuario
Excepciones:	<ul style="list-style-type: none"> - Si ocurre algún error procesar la creación de la cuenta, se muestra mensaje al usuario - Mientras no se validen los datos del formulario no se podrá continuar el flujo - Tras el envío de datos habrá una segunda verificación del email del alumno creado, y éste debe de coincidir con el introducido anteriormente para continuar el flujo - Si el email ya esta registrado en el sistema, se muestra mensaje informativo 	

Register student into group		
Actores:	Teacher	
Descripción:	El profesor desea matricular un alumno en uno de los grupos disponibles. En general, serán los alumnos los que se matriculen sin la intervención del profesor. Se ha creado este caso de uso para facilitar al profesor la posibilidad de matricular a un alumno manualmente en caso de el alumno no posea un dispositivo móvil compatible o no pueda realizar la auto matrícula.	
Precondición:	<ul style="list-style-type: none"> - El profesor ha accedido a su perfil del sistema - El profesor tiene creado al menos un grupo y éste está seleccionado 	
Postcondición:	Según la opción elegida por el profesor para matricular un alumno en un grupo, se procederá a la ejecución del CU de búsqueda o CU de creación.	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Seleccionar opción 'añadir' 3. Seleccionar opción: 'busqueda'	2. Mostrar opciones de matriculación 4. Ejecutar caso de uso Search and register student 5. Mostrar vista de búsqueda
Flujo alternativo:	3. Seleccionar opción: 'nuevo'	4. Ejecutar caso de uso Create and register student account 5. Mostrar vista de creación
Excepciones:		

Unregister student from group		
Actores:	Teacher	
Descripción:	El profesor desea eliminar la matrícula de un alumno en un grupo.	
Precondición:	<ul style="list-style-type: none"> - El profesor ha accedido a su perfil del sistema - El profesor tiene creado al menos un grupo con un alumno matriculado. El grupo está seleccionado y se muestra el listado de alumnos. 	
Postcondición:	- Se eliminarán todos los registros de asistencia asociados a dicha matrícula	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Seleccionar opción 'eliminar' (deslizando la celda hacia la izquierda o presionando botón 'editar')	2. Procesar la eliminación de matrícula 3. Mostrar mensaje al usuario
Excepciones:	<ul style="list-style-type: none"> - Si ocurre algún error a la hora de procesar la eliminación de matrícula, se muestra mensaje al usuario - Si ocurre algún error al recuperar el listado de alumnos del servidor, se muestra mensaje al usuario 	

Set/Unset attendance		
Actores:	Teacher	
Descripción:	El profesor desea marcar la asistencia de un alumno (o varios) manualmente. Este caso de uso se ha creado para facilitar la gestión de las asistencias de los alumnos que no dispongan de un dispositivo móvil compatible con el sistema. Este caso de uso es el que permite al profesor realizar el proceso de control de manera tradicional, en caso que lo desee.	
Precondición:	<ul style="list-style-type: none"> - El profesor ha accedido a su perfil del sistema - El profesor tiene creado al menos un grupo con un alumno matriculado. El grupo está seleccionado y se muestra el listado de alumnos. 	
Postcondición:	- El alumno cambia su estado de asistencia de tal manera que, al terminar el proceso, si el alumno estaba 'ausente' su nuevo estado será 'presente' y viceversa.	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	<ol style="list-style-type: none"> 1. Seleccionar alumno (o alumnos) 3. Presionar botón 'Marcar asistencia' 	<ol style="list-style-type: none"> 2. Activar botón 'Marcar asistencia' 4. Procesar marcado de asistencia 5. Mostrar mensaje al usuario
Excepciones:	<ul style="list-style-type: none"> - Si ocurre algún error a la hora de procesar el marcado de asistencia, se muestra mensaje al usuario - Si ocurre algún error al recuperar el listado de alumnos del servidor, se muestra mensaje al usuario - Solo alumnos con el mismo estado (presente o ausente) pueden ser seleccionados. 	

See student attendance graph		
Actores:	Teacher	
Descripción:	El profesor desea consultar el gráfico de asistencias para conocer el estado global de asistencias del alumno.	
Precondición:	<ul style="list-style-type: none"> - El profesor ha accedido a su perfil del sistema - El profesor tiene creado al menos un grupo con un alumno matriculado. El grupo está seleccionado y se muestra el listado de alumnos. 	
Postcondición:		
Flujo de eventos:	Actividades del actor	Respuesta sistema
	<ol style="list-style-type: none"> 1. Seleccionar alumno (Mantener presionado durante un segundo) 3. Seleccionar 'Ver gráfico' 	<ol style="list-style-type: none"> 2. Mostrar opciones de alumno 4. Procesar petición en API 5. Mostrar gráfico de asistencias
Excepciones:	- Si ocurre algún error a la hora de mostrar el gráfico, se muestra un mensaje al usuario	



7. Casos de uso 4/4

Create Subject		
Actores:	Teacher	
Descripción:	El profesor desea crear una nueva asignatura.	
Precondición:	- El profesor ha accedido a su perfil del sistema	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Elegir opción 'Nueva asignatura' 3. Insertar datos en el formulario 6. Seleccionar opción: Guardar	2. Mostrar formulario de nueva asignatura 4. Validar datos del formulario 5. Activar botón 'Guardar' 7. Procesar nueva asignatura en local y en API 8. Mensaje al usuario
Excepciones:	<ul style="list-style-type: none"> - Si los datos del formulario no son válidos, no se activará el botón de guardar - Si ocurre algún error en el proceso de la asignatura, se muestra mensaje al usuario 	

Update subject		
Actores:	Teacher	
Descripción:	El profesor desea modificar una asignatura.	
Precondición:	<ul style="list-style-type: none"> - El profesor ha accedido a su perfil del sistema - El profesor debe de tener una asignatura creada. La vista actual muestra el listado de asignaturas. 	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Seleccionar opción: Editar 3. Modificar datos deseados en el formulario 6. Seleccionar opción: Guardar	2. Mostrar formulario con datos de asignatura 4. Validar datos del formulario 5. Activar botón 'Guardar' 6. Procesar modificación asignatura en local y en API 8. Mensaje al usuario
Excepciones:	<ul style="list-style-type: none"> - Si los datos del formulario no son válidos, no se activará el botón de enviar - Si ocurre algún error en el proceso de la asignatura, se muestra mensaje al usuario 	

Delete subject		
Actores:	Teacher	
Descripción:	El profesor desea eliminar una asignatura.	
Precondición:	<ul style="list-style-type: none"> - El profesor ha accedido a su perfil del sistema - El profesor debe de tener una asignatura creada 	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Elegir opción: 'eliminar' (deslizandó la celda hacia la izquierda o presionando botón 'editar')	2. Procesar borrado de asignatura 3. Mensaje al usuario
Flujo de eventos alternativo:	1. Seleccionar asignatura (Presionar asignatura al menos durante 5 segundos) 3. Seleccionar 'borrar'	2. Mostrar opciones de borrado 4. Procesar borrado de asignatura 5. Mensaje al usuario
Excepciones:	<ul style="list-style-type: none"> - Si ocurre algún error en el proceso de borrado, se muestra mensaje al usuario - En el caso de error de borrado, se ha habilitado un modo de borrado seguro que da lugar al flujo alternativo de eventos 	

Create group		
Actores:	Teacher	
Descripción:	El profesor desea crear un nuevo grupo	
Precondición:	<ul style="list-style-type: none"> - El profesor ha accedido a su perfil del sistema - El profesor debe tener al menos una asignatura creada. La vista actual del sistema muestra el listado de grupos pertenecientes a una asignatura. 	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Elegir opción: Nuevo grupo 3. Insertar datos de grupo en el formulario 6. Guardar datos	2. Mostrar formulario de nuevo grupo 4. Validar datos del formulario 5. Activar botón 'Guardar' 7. Procesar creación nuevo grupo en local y API 10. Mensaje al usuario
Excepciones:	<ul style="list-style-type: none"> - Si los datos del formulario no son válidos, no se activará el botón de enviar - Si ocurre algún error en el proceso de la asignatura, se muestra mensaje al usuario 	

Update group		
Actores:	Teacher	
Descripción:	El profesor desea modificar la información de un grupo	
Precondición:	<ul style="list-style-type: none"> - El profesor ha accedido a su perfil del sistema - El profesor debe de tener un grupo creado. La vista actual del sistema muestra el listado de grupos pertenecientes a una asignatura. 	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Elegir opción 'Editar' 3. Modificar datos deseados en el formulario 6. Guardar datos	2. Mostrar formulario con datos de grupo 4. Validar datos del formulario 5. Activar botón 'Guardar' 7. Procesar modificación grupo en local y API 10. Mensaje al usuario
Excepciones:	<ul style="list-style-type: none"> - Si los datos del formulario no son válidos, no se activará el botón de enviar - Si ocurre algún error en el proceso de de modificación, se muestra mensaje al usuario 	

Delete group		
Actores:	Teacher	
Descripción:	El profesor desea eliminar un grupo.	
Precondición:	<ul style="list-style-type: none"> - El profesor ha accedido a su perfil del sistema - El profesor debe de tener un grupo creado. La vista actual del sistema muestra el listado de grupos pertenecientes a una asignatura. 	
Postcondición:	N/A	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Elegir opción: 'eliminar' (deslizand o la celda hacia la izquierda o presionando botón 'editar')	2. Procesar borrado de grupo 3. Mensaje al usuario
Excepciones:	<ul style="list-style-type: none"> - Si ocurre algún error en el proceso de borrado, se muestra mensaje al usuario 	

Set/Unset free day		
Actores:	Teacher	
Descripción:	El profesor desea modificar el estado de un día lectivo a festivo o viceversa. El cambio de estado para un día concreto, dependerá del estado actual del día, es decir, al aplicar este proceso, el día pasará a festivo si su estado actual es lectivo y viceversa.	
Precondición:	<ul style="list-style-type: none"> - El profesor ha accedido a su perfil del sistema - El profesor debe de tener un grupo creado. La vista actual del sistema muestra horario de un grupo creado 	
Postcondición:	- El día cambia su estado de tal manera que, al terminar el proceso, si el día tenía estado 'lectivo' su nuevo estado será 'festivo' y viceversa.	
Flujo de eventos:	Actividades del actor	Respuesta sistema
	1. Seleccionar día para modificar 3. Elegir opción 'Modificar'	2. Mostrar opciones de día 4. Procesar cambio de estado local y API 5. Mostrar listado con cambios 6. Mensaje al usuario
Excepciones:	<ul style="list-style-type: none"> - Si ocurre algún error en el proceso del cambio, se muestra mensaje al usuario 	

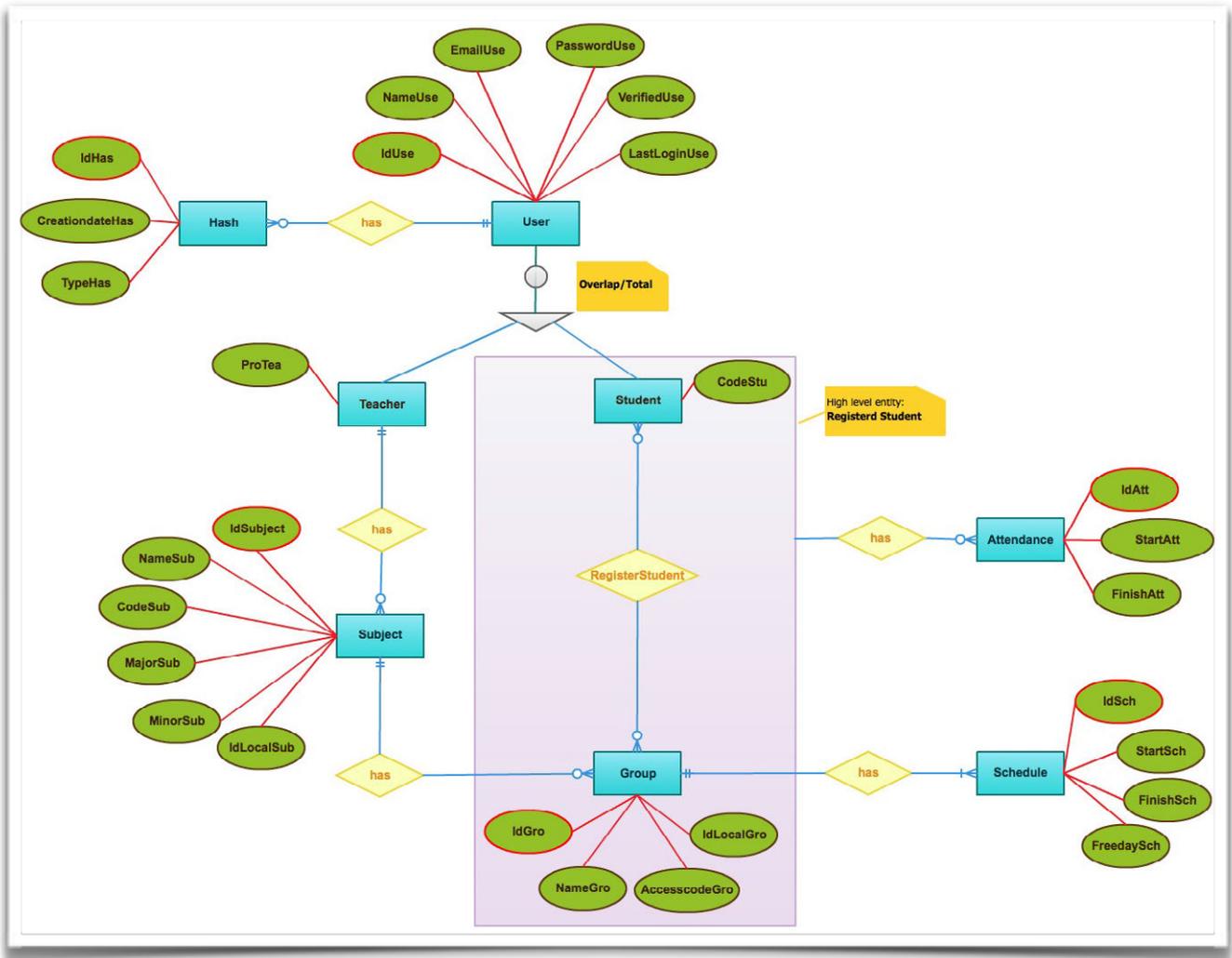
4.4. Diagrama de Entidad-Relación Extendido

En este apartado se representa el diagrama de Entidad-Relación Extendido, incluyendo las entidades más relevantes del sistema y la relaciones existentes entre ellas. La notación empleada para la representación de cardinalidades es Crow's foot, además para una mejor visualización de las relaciones se ha añadido la notación de Chen.

Uno de los aspectos más interesantes de este diagrama es la relación de generalización total solapada existente entre las entidades Usuario, Profesor y Alumno. Dicha relación se entiende de la siguiente manera:

- Solapada (overlap): La generalización es solapada si una ocurrencia del supertipo puede ser a la vez de varios subtipos, es decir, un Usuario puede ser Profesor y Alumno
- Total (total): La generalización es total si no hay ocurrencias en el supertipo que no pertenezcan a ninguno de sus subtipos, es decir, un Usuario es Profesor o es Alumno, no hay de otro tipo.

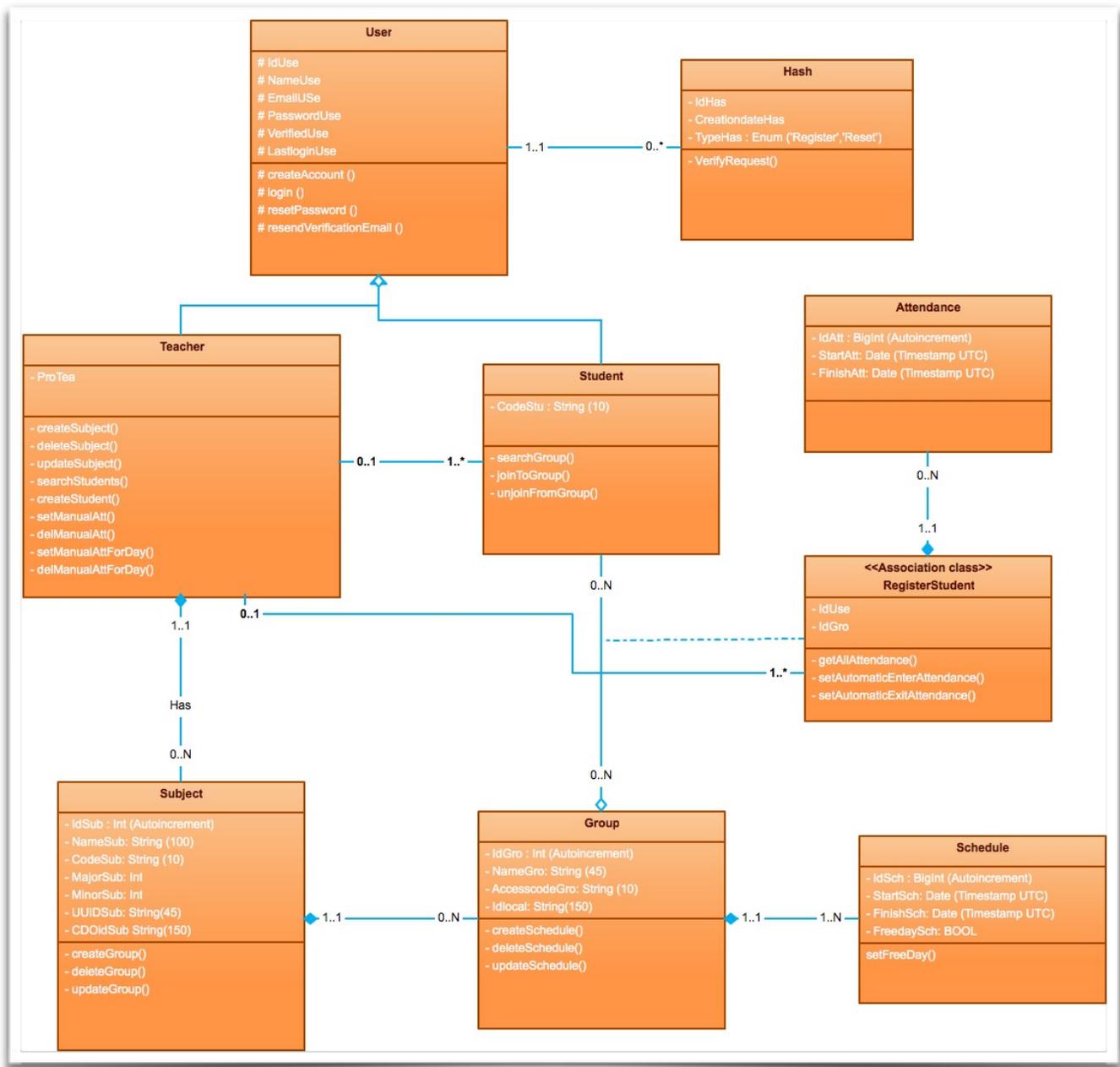
Otro aspecto relevante de este diagrama, es la creación de la entidad de alto nivel Alumno Registrado. Dicha entidad surge debido a las limitaciones existentes en los diagramas de entidad-relación, ya que no es posible representar "relaciones entre relaciones".



8. Diagrama Entidad-Relación Extendido

4.5. Diagrama de Clases

En este apartado se muestra un diagrama de clases de alto nivel en el que se muestran las relaciones existentes entre las diferentes clases y las principales operaciones que se deben de cumplir en la herramienta.



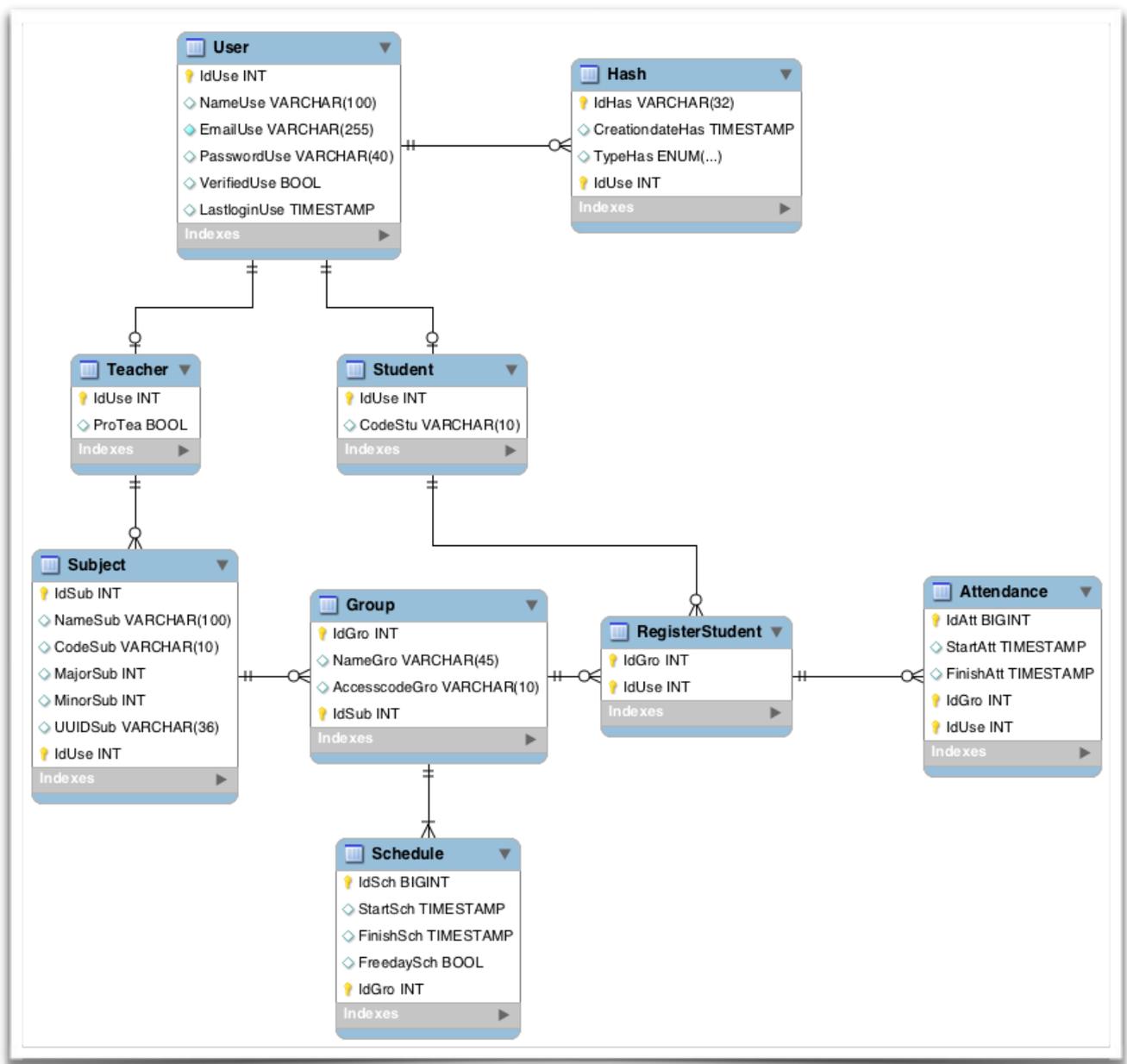
9. Diagrama de clases

5. DISEÑO

5.1. Modelo de datos

En este apartado se presenta el modelo de datos que almacena toda la información del sistema. Para la creación del modelo de datos se han utilizado MySQLWorkbench y CoreData.

Obsérvese en el diagrama la relajación de las cardinalidades mínimas en el diagrama para el estado inicial del sistema.



10. Modelo de datos

5.2. Descripción de tablas

En este apartado se presenta la descripción textual completa de las tablas del modelo de datos.

USER		
Tabla que almacena la información de cada usuario del sistema.		
Campo	Tipo	Descripción
IdUse	INT	Primary Key. Campo que almacena el número que identifica cada usuario del sistema.
NameUse	VARCHAR(100)	Campo que almacena el nombre de cada usuario.
EmailUse	VARCHAR(255)	Campo que almacena el email de cada usuario. Este campo además, se utilizará como información de acceso al sistema.
PasswordUse	VARCHAR(40)	Campo que almacena la contraseña de acceso al sistema. Esta contraseña será codificada con un algoritmo SHA1 antes de ser almacenada.
VerifiedUse	BOOL	Campo que almacena si el usuario ha sido verificado por el sistema.
LastloginUse	TIMESTAMP	Campo que almacena la fecha de la última sesión del usuario.

HASH		
Tabla que almacena las solicitudes de registro en el sistema y cambio de contraseña creadas por los usuarios. La existencia de esta tabla facilita un sistema de seguridad adicional al verificar el email de cada usuario. Además, nos permite gestionar la duración de cada petición de registro y cambio de contraseña.		
Campo	Tipo	Descripción
IdHas	VARCHAR(32)	Primary Key. Campo que almacena una cadena de texto generada aleatoriamente para identificar y verificar las operaciones de registro y cambio de contraseña.
CreationdateHas	TIMESTAMP	Campo que almacena la fecha en la que se creó el registro hash. Utilizaremos este campo para determinar la duración de validez de las operaciones solicitadas.
TypeHas	ENUM	Campo que determina el tipo de operación solicitado. Este campo puede contener los valores (Register o Reset).
IdUse	INT	Foreign Key. Campo que almacena el usuario que ha realizado la operación de registro o cambio de contraseña (referencia de la tabla User).

TEACHER

Tabla que almacena la información específica del profesor. Dicha tabla junto al registro asociado en la tabla User, nos proporciona toda la información para un profesor concreto.

Campo	Tipo	Descripción
IdUse	INT	Foreign Key. Campo que almacena el identificador de usuario para la relación entre usuario y profesor (referencia de la tabla User)
Protea	BOOL	Campo que almacena si el profesor ha realizado la compra para desbloquear las restricciones de la aplicación.

STUDENT

Tabla que almacena la información específica del alumno. Dicha tabla junto al registro asociado en la tabla User, nos proporciona toda la información para un alumno concreto.

Campo	Tipo	Descripción
IdUse	INT	Foreign Key. Campo que almacena el identificador de usuario para la relación entre usuario y alumno (referencia de la tabla User)
CodeStu	VARCHAR(10)	Campo que almacena el código del alumno. Este campo se utilizará además para localizar alumnos en el buscador.

SUBJECT

Tabla que almacena la información de cada asignatura del sistema.

Campo	Tipo	Descripción
IdSub	INT	Primary Key. Campo que almacena el número que identifica cada asignatura del sistema.
NameSub	VARCHAR(100)	Campo que almacena el nombre de la asignatura.
CodeSub	VARCHAR(10)	Campo que almacena el código de la asignatura. Este campo se utilizará para localizar las asignaturas en el buscador del sistema.
MajorSub	INT	Campo que almacena el número mayor de la región asociado a un active beacon.
MinorSub	INT	Campo que almacena el número menor de la región asociado a un active beacon.
UUIDSub	VARCHAR(36)	Campo que almacena el identificador de la región asociado a un active beacon.
IdUse	INT	Foreign Key. Campo que almacena el usuario creador de la asignatura (referencia de la tabla User)

GROUP

Tabla que almacena la información de cada grupo del sistema.

Campo	Tipo	Descripción
IdGro	INT	Primary Key. Campo que almacena el número que identifica cada grupo del sistema.
NameGro	VARCHAR(45)	Campo que almacena el nombre del grupo.
AccesscodeGro	VARCHAR(10)	Campo que almacena el código de acceso al grupo. Este campo se solicitará al alumno para realizar la automatrícula en dicho grupo.
IdSub	INT	Foreing Key. Campo que almacena la referencia de la asignatura a la que pertenece el grupo (referencia de la tabla Subject)

REGISTEREDSTUDENT

Tabla utilizada para almacenar los registros de alumnos matriculados en los diferentes grupos. Esta tabla nos permite llevar una gestión de todas las matrículas realizadas en el sistema.

Campo	Tipo	Descripción
IdGro	INT	Foreing Key. Campo que almacena el grupo de la matrícula (referencia de la tabla Group).
IdUse	INT	Foreing Key. Campo que almacena el usuario matriculado (referencia de la tabla User).

ATTENDANCE

Tabla utilizada para almacenar la información de las asistencias de los alumnos matriculados.

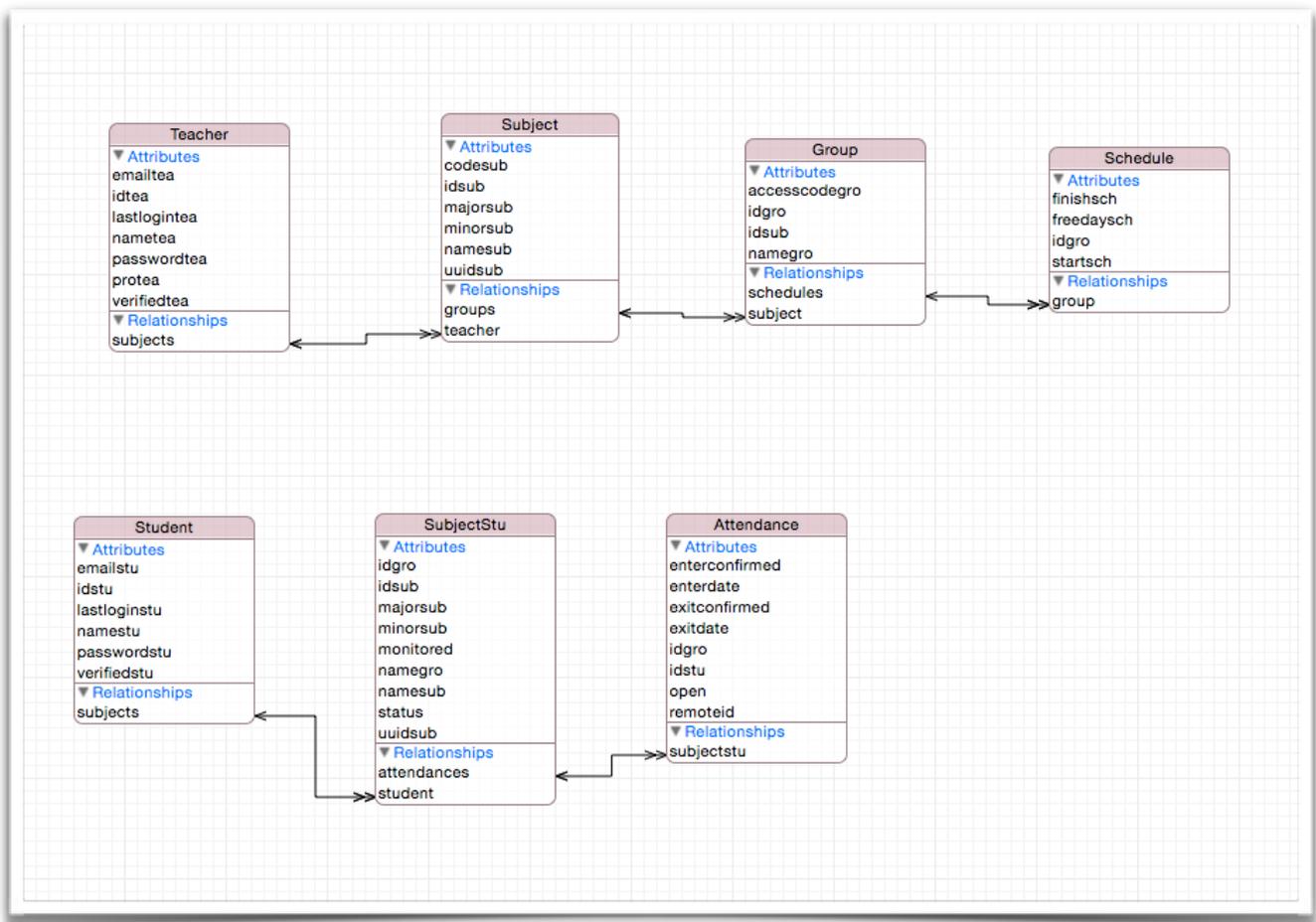
Campo	Tipo	Descripción
IdAtt	BIGINT	Primary Key. Campo que almacena el número que identifica cada asistencia dentro del sistema.
StartAtt	TIMESTAMP	Campo que almacena la fecha de comienzo de la asistencia.
FinishAtt	TIMESTAMP	Campo que almacena la fecha de finalización de la asistencia.
IdGro	INT	Foreing Key. Campo que almacena el grupo al que se realiza la asistencia (referencia de la tabla Group).
IdUse	INT	Foreing Key. Campo que almacena el usuario que realiza la asistencia (referencia de la tabla User).

SCHEDULE

Tabla utilizada para almacenar la información del horario lectivo para cada grupo.

Campo	Tipo	Descripción
IdSch	BIGINT	Primary Key. Campo que almacena el número que identifica cada día del horario de clases.
StartSch	TIMESTAMP	Campo que almacena la fecha de comienzo del horario.
FinishSch	TIMESTAMP	Campo que almacena la fecha de finalización del horario.
FreedaySch	BOOL	Campo que almacena si el día del horario especificado es festivo.
IdGro	INT	Foreing Key. Campo que almacena el grupo al que pertenece el horario (referencia de la tabla Group).

El siguiente diagrama muestra el modelo local de Core Data.



11. Modelo local de Core Data

La mayoría de las tablas creadas en el dispositivo cliente mantienen la misma información que las tablas creadas en el servidor por lo que no se va a detallar la información de nuevo y solo se especificará la información de las tablas relevantes en este nuevo modelo.

La creación de este modelo local se ha realizado básicamente para la sincronización de los datos estáticos del sistema, como puedan ser asignaturas, grupos y horarios, de esta manera se evitan peticiones innecesarias al servidor con el consecuente ahorro en el consumo de datos y energía en el dispositivo móvil.

Las tablas que aportan información adicional en el sistema son:

ATTENDANCE		
Tabla creada en los dispositivos de los alumnos para almacenar la información de los envíos automáticos de asistencia a clase.		
Campo	Tipo (CoreData)	Descripción
enterconfirmed	Boolean	Campo que almacena si el envío automático de entrada a una asignatura ha sido recibido en el servidor.
enterdate	Date	Campo que almacena la fecha de entrada a la asignatura.
exitconfirmed	Boolean	Campo que almacena si el envío automático de salida de una asignatura ha sido recibido en el servidor.
exitdate	Date	Campo que almacena la fecha de salida de la asignatura.
idgro	Integer 32	Campo que almacena el identificador del grupo para el que se está realizando la asistencia.
idstu	Integer 32	Campo que almacena el identificador del alumno que está realizando la asistencia.
open	Boolean	Campo que nos indica si el registro de asistencia está pendiente de envío de salida de asignatura.
remoteid	Integer 64	Campo que almacena el identificador remoto de la asistencia enviada.

SUBJECTSTU

Tabla creada en los dispositivos de los alumnos para almacenar la información de las asignaturas en las que está matriculado.

Campo	Tipo	Descripción
idgro	Integer 32	Campo que almacena el identificador del grupo.
idsub	Integer 32	Campo que almacena el identificador de la asignatura.
majorsub	Integer 32	Campo que almacena el número mayor del beacon asociado a la asignatura.
minorsub	Integer 32	Campo que almacena el número menor del beacon asociado a la asignatura.
monitored	String	Campo que almacena si la asignatura está siendo monitoreada por tecnología iBeacon.
namegro	String (45)	Campo que almacena el nombre del grupo.
namesub	String (100)	Campo que almacena el nombre de la asignatura.
status	String	Campo que almacena el estado del alumno. Presente o ausente.
uuidsub	String (36)	Campo que almacena el UUID del beacon asociado a la asignatura.

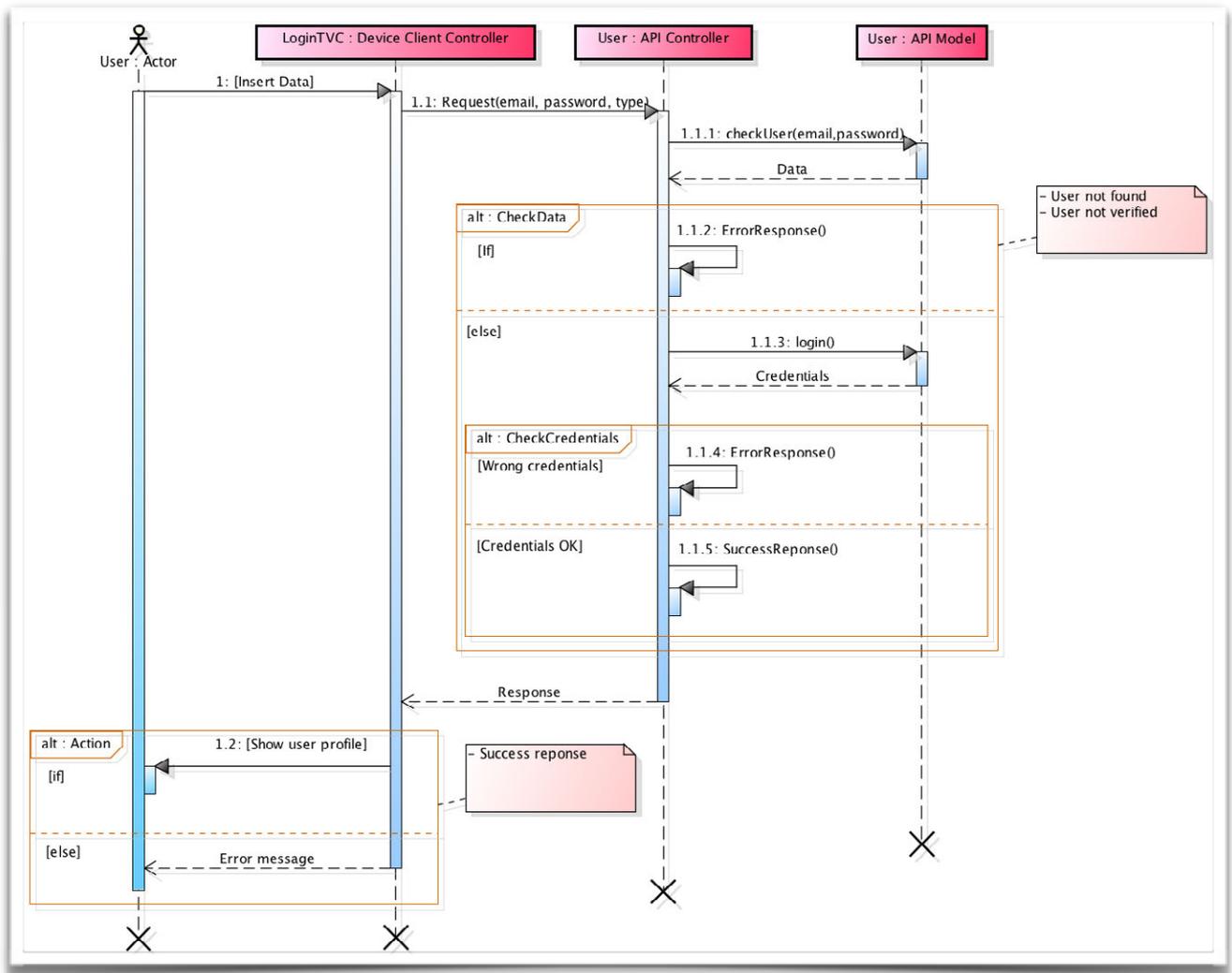
5.3. Diagramas de secuencia

En este apartado se presentan los diagramas de secuencia asociados al comportamiento de la herramienta. Algunos de los casos de uso descritos de forma textual son de suma sencillez y se ha descartado el diseño del diagrama de secuencia.

En algunos casos se ha simplificado el desarrollo de los diagramas para su mejor legibilidad en este documento, es decir se representa la secuencia desde un punto de vista de alto nivel sin profundizar al máximo detalle o incluso omitiendo algunas operaciones básicas.

5.3.1. Log In

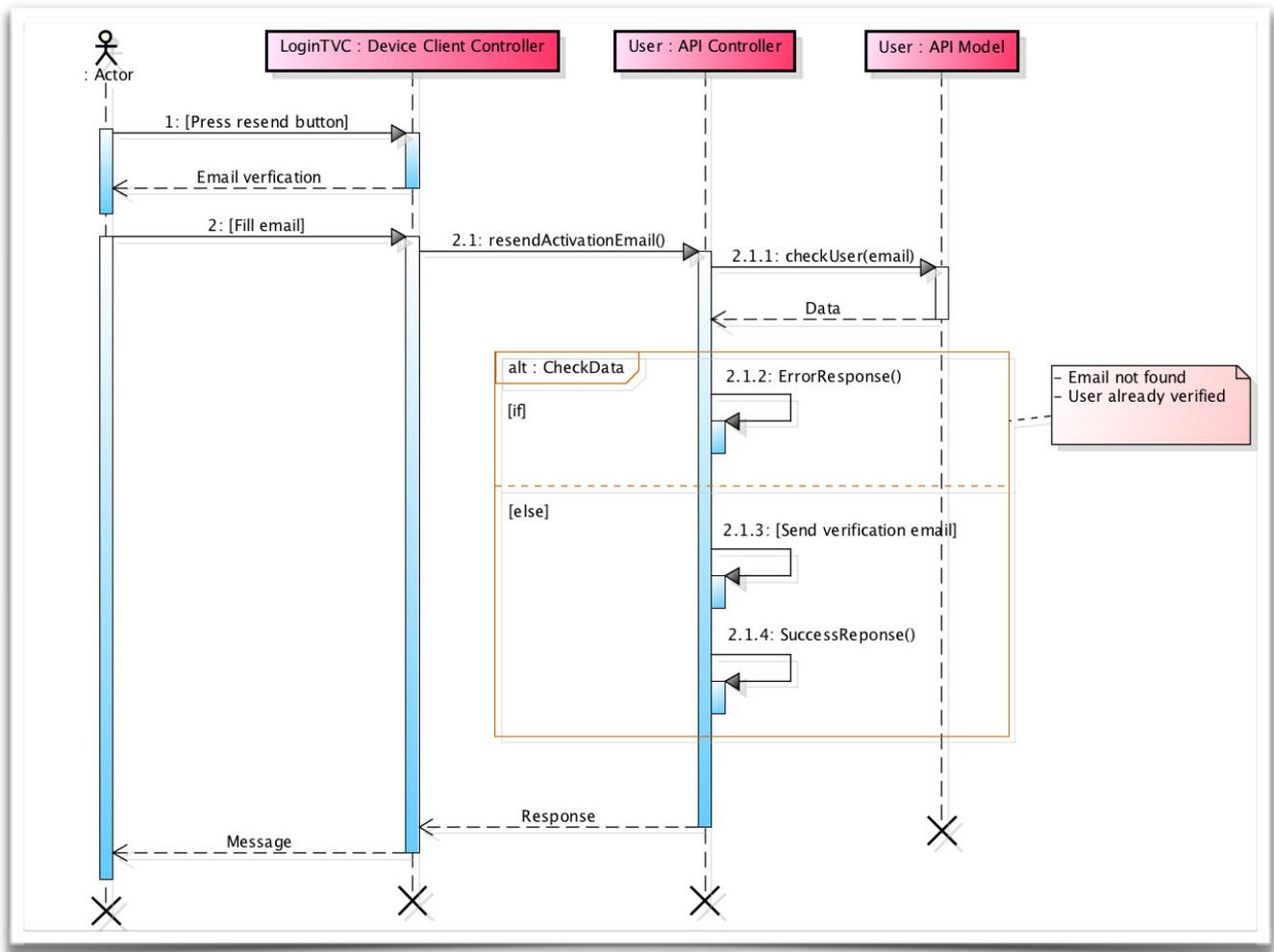
Este diagrama de secuencia muestra el proceso a seguir para que un usuario (profesor o alumno) acceda a su perfil del sistema.



12. Diagrama de secuencia - Log In

5.3.2. Sign up

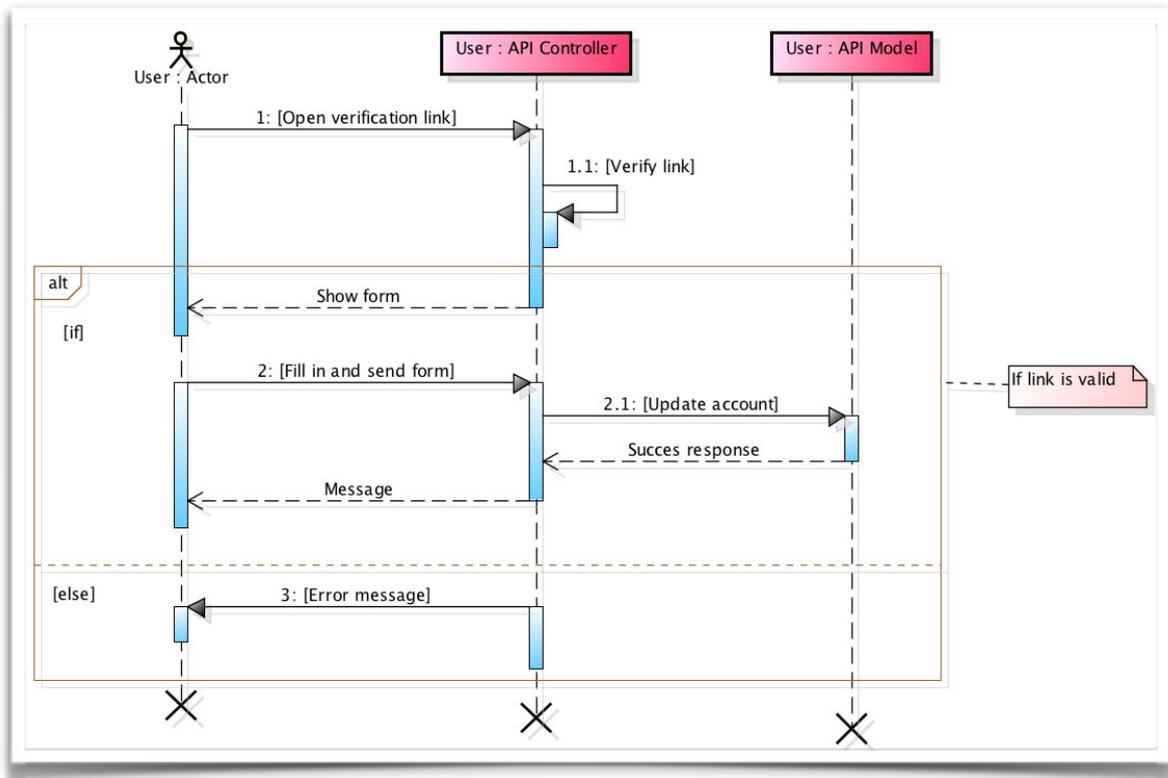
Este diagrama de secuencia muestra el proceso a seguir para que un usuario (profesor o alumno) cree una cuenta en el sistema.



13. Diagrama de secuencia - Sign up

5.3.3. Email verification

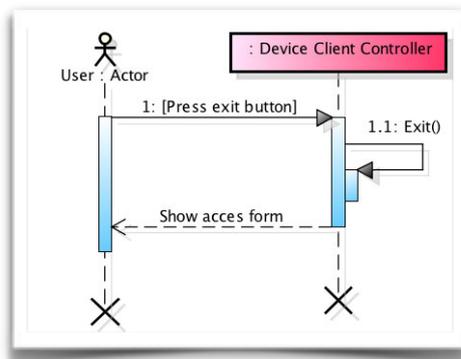
Este diagrama de secuencia muestra el proceso a seguir para que un usuario (profesor o alumno) verifique su cuenta. El proceso se inicia siguiendo el enlace que se envía en el correo de bienvenida y se puede completar desde cualquier navegador.



14. Diagrama de secuencia - Email verification

5.3.4. Log out

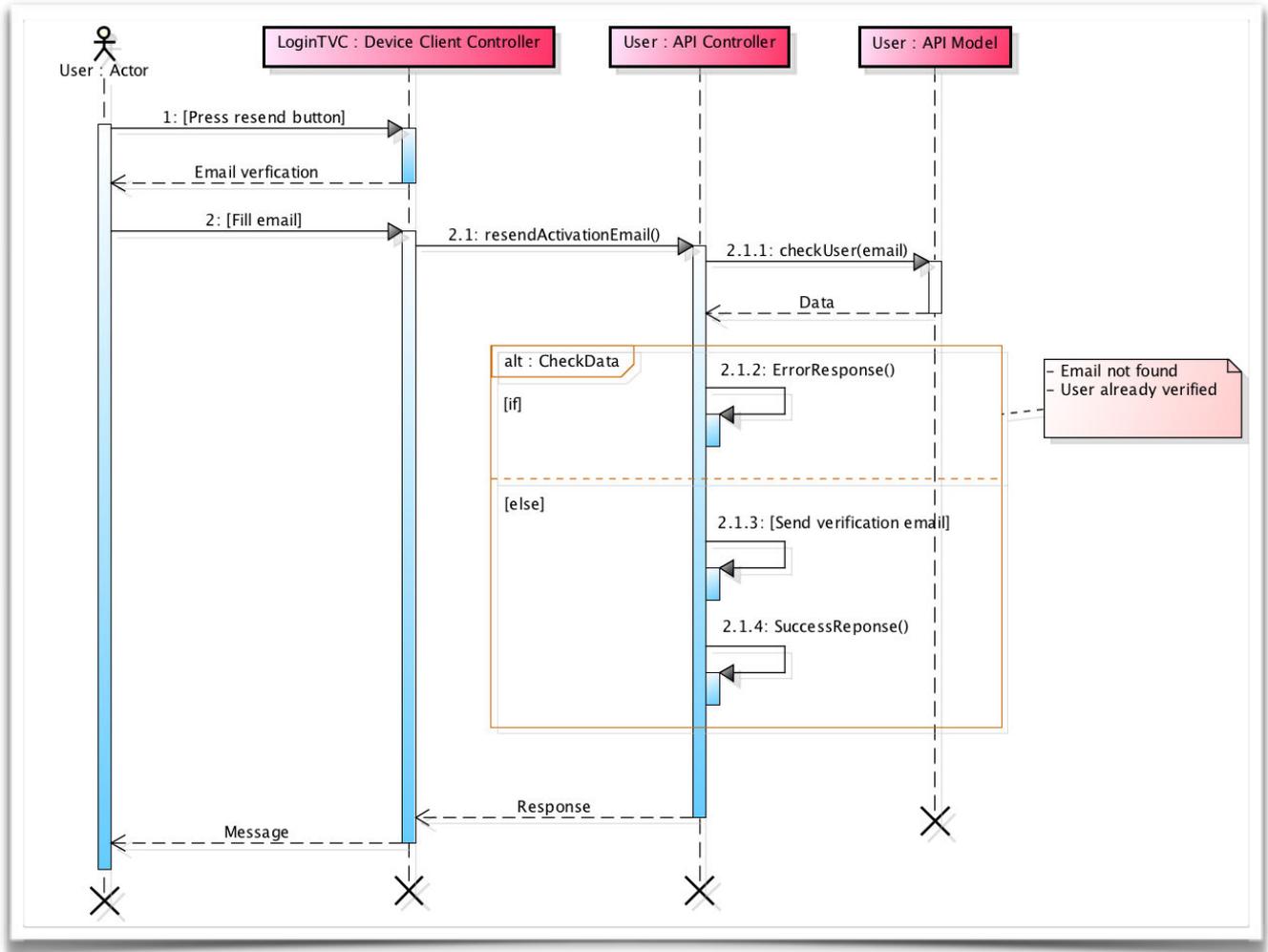
Este diagrama de secuencia muestra el proceso a seguir para que un usuario (profesor o alumno) se desconecte de su perfil del sistema.



15. Diagrama de secuencia - Log out

5.3.5. Resend verification email

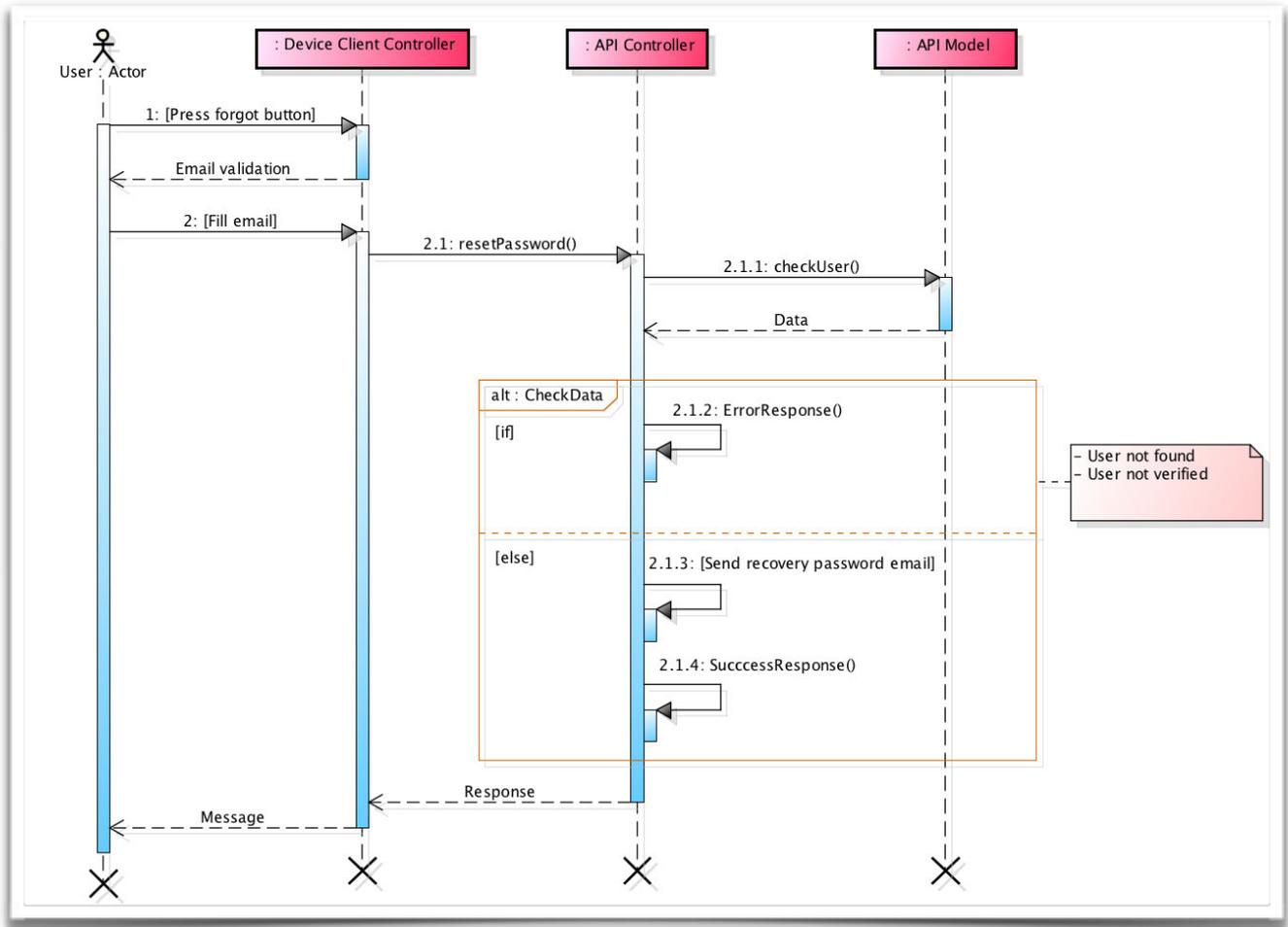
Este diagrama de secuencia muestra el proceso a seguir para que un usuario (profesor o alumno) solicite al sistema un nuevo email de verificación.



16. Diagrama de secuencia - Resend verification email

5.3.6. Forgot password

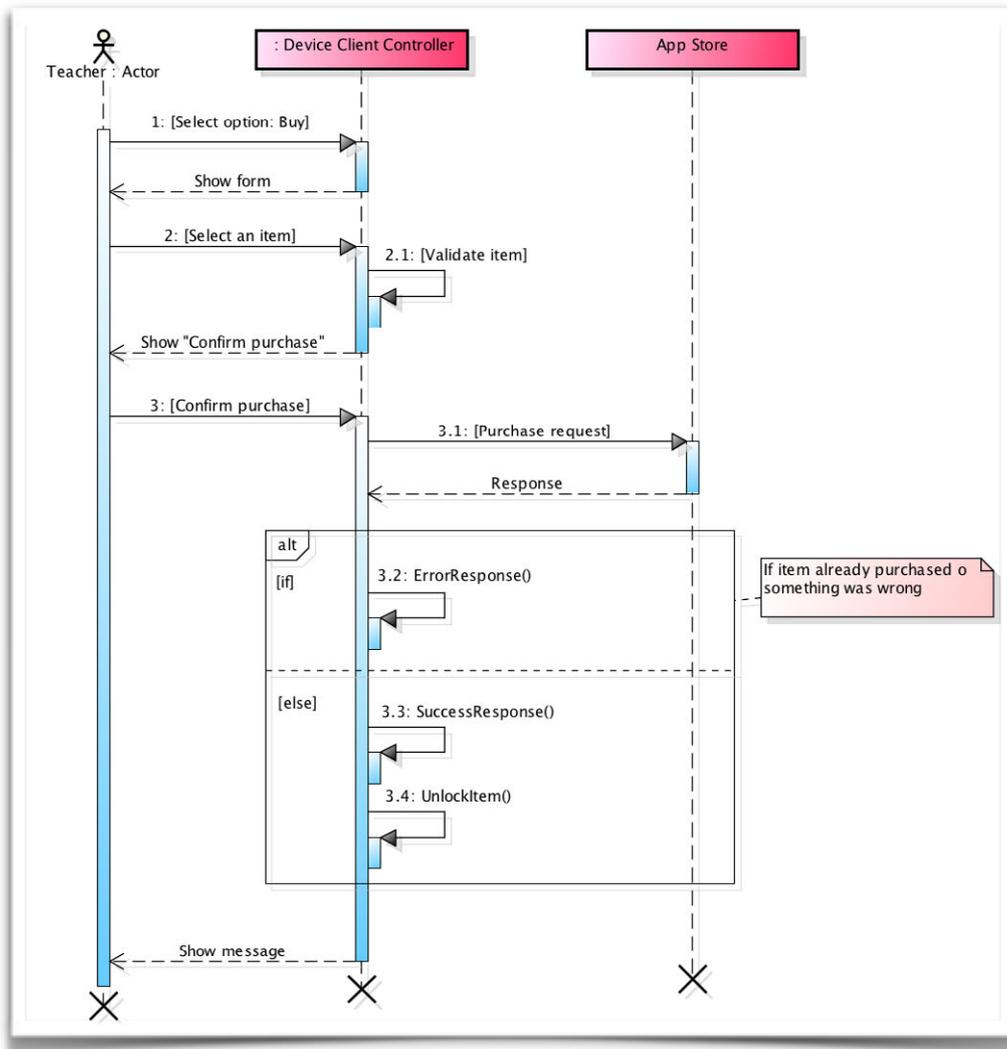
Este diagrama de secuencia permite a un usuario (profesor o alumno) iniciar el proceso para modificar/restablecer la contraseña de acceso al sistema. Una vez completado el usuario recibirá un email con instrucciones para modificar o restablecer la contraseña.



17. Diagrama de secuencia - Forgot password

5.3.7. In-app purchase

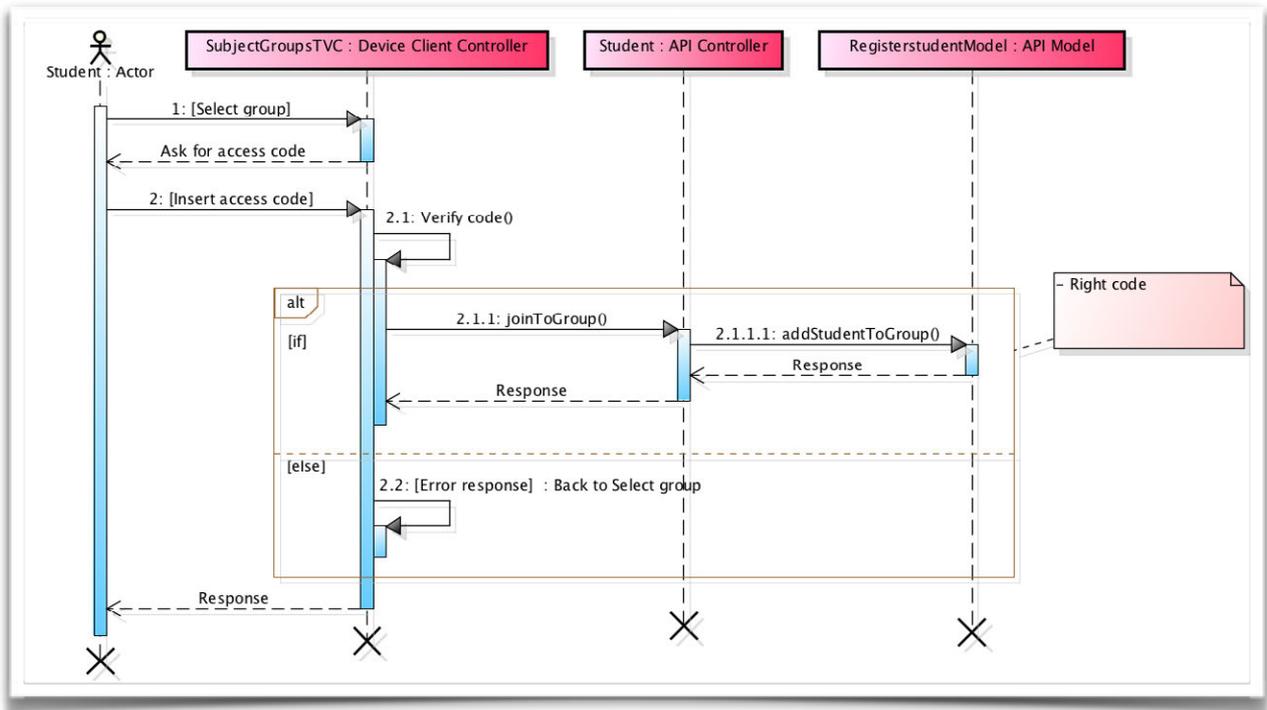
Este diagrama de secuencia muestra el proceso a seguir para que un usuario realice una compra in-app para desbloquear las restricciones de publicidad y gestión (asignaturas, grupo, y alumnos).



18. Diagrama de secuencia - In-app purchase

5.3.8. Join group

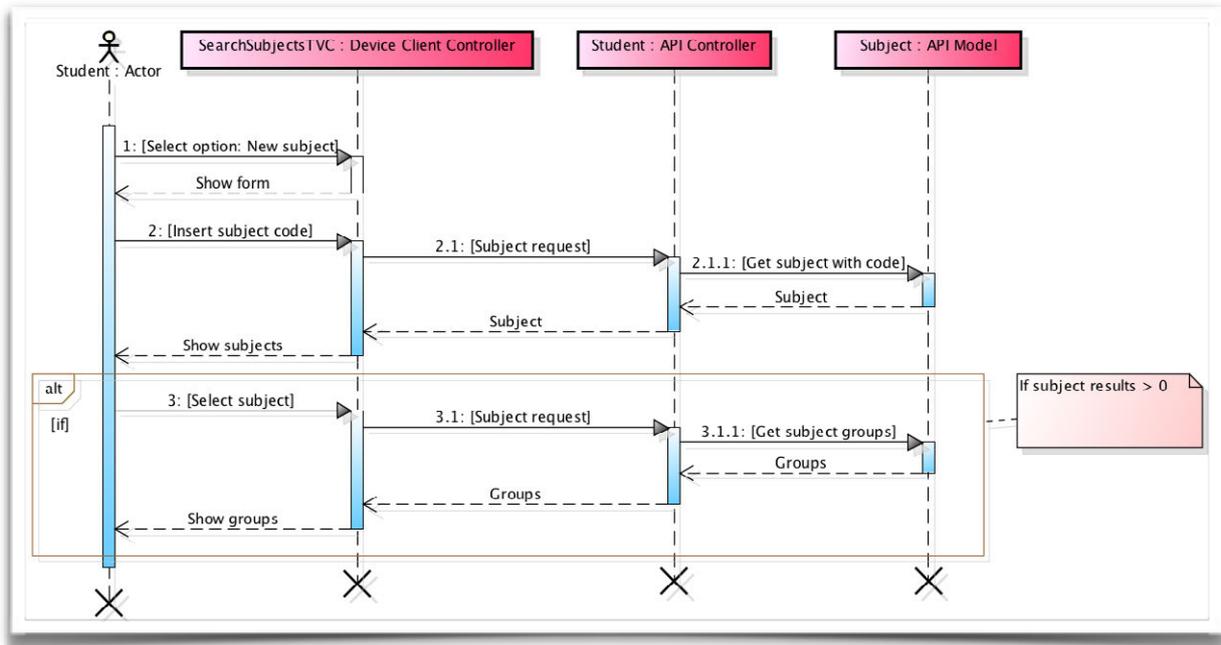
Este diagrama de secuencia muestra el proceso a seguir para que un alumno realice la automatrícula en un grupo de una asignatura. Para completar el proceso satisfactoriamente el alumno debe de conocer el código de acceso al grupo (proporcionado por el profesor).



18. Diagrama de secuencia - Join group

5.3.9. Search subject groups

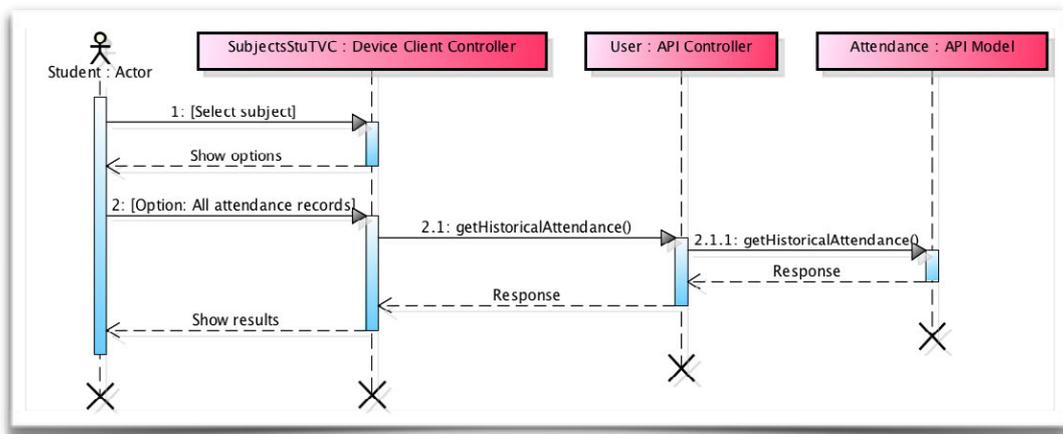
Este diagrama de secuencia muestra el proceso que un alumno debe de seguir para buscar los grupos de las asignaturas.



19. Diagrama de secuencia - Search subject groups

5.3.10. See all attendances records

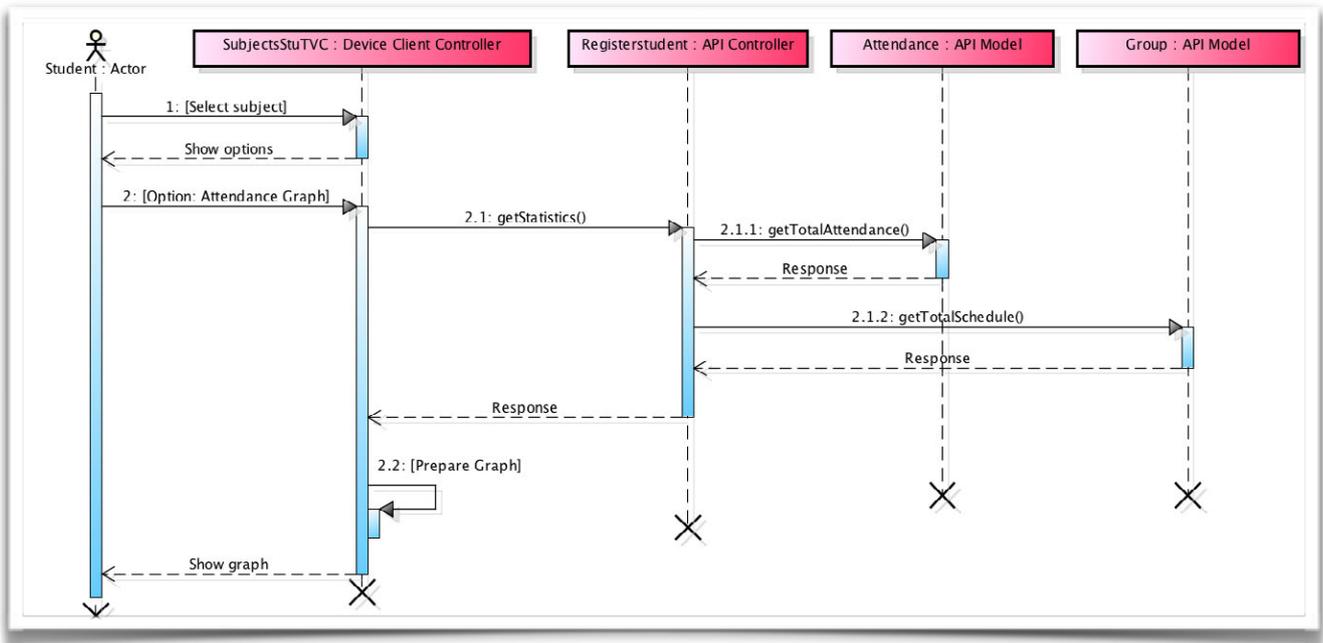
Este diagrama de secuencia muestra el proceso que un alumno debe de seguir para consultar el histórico detallado de asistencias.



20. Diagrama de secuencia - See all attendances records

5.3.11. See attendances graph

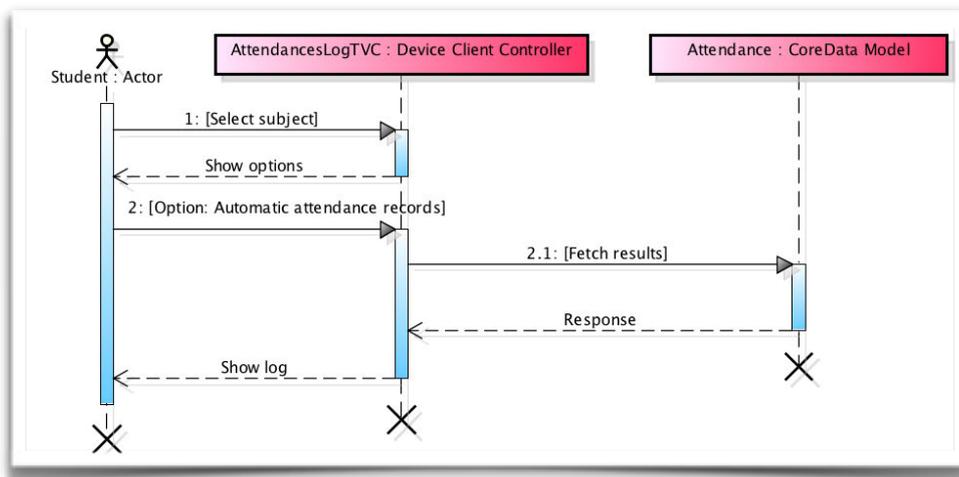
Este diagrama de secuencia muestra el proceso que un alumno debe de seguir para consultar el gráfico general de asistencias.



21. Diagrama de secuencia - See attendances graph

5.3.12. See automatic attendances log

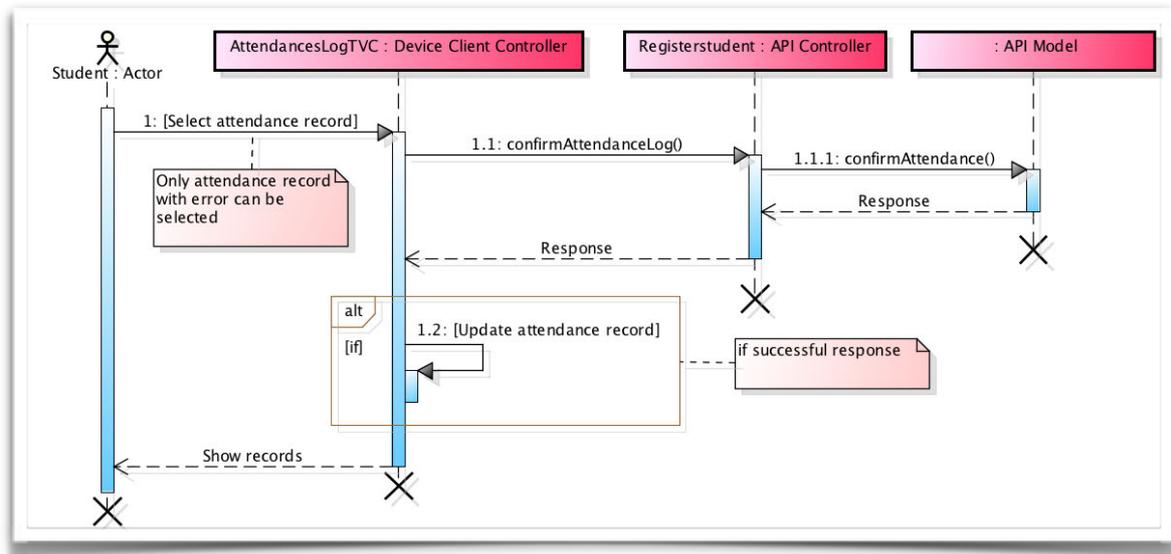
Este diagrama de secuencia muestra el proceso que un alumno debe de seguir para consultar el histórico detallado de envío de asistencias automáticos que ha realizado el sistema.



22. Diagrama de secuencia - See automatic attendances log

5.3.13. Send failed automatic attendance requests

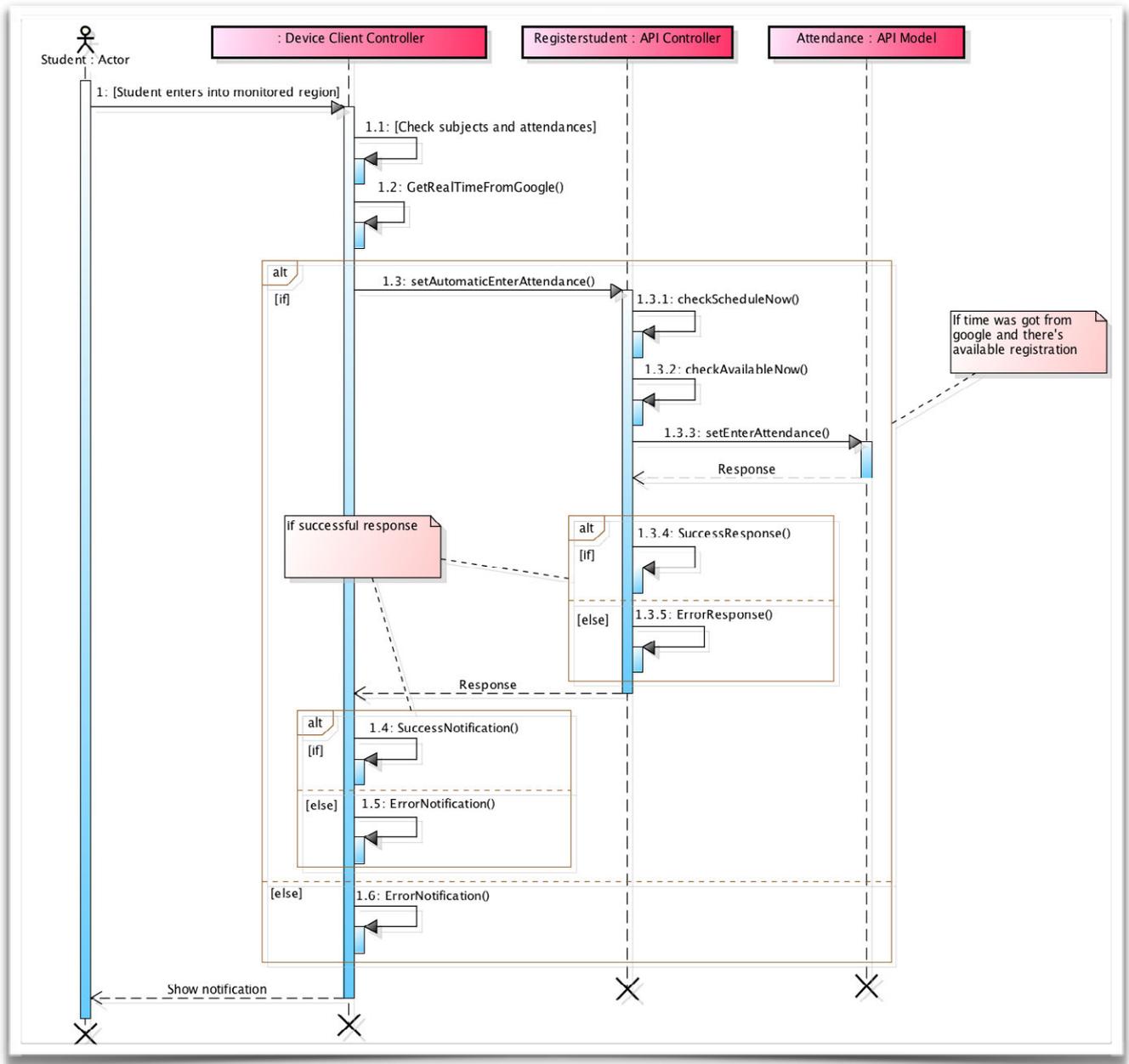
Este diagrama de secuencia muestra el proceso que un alumno debe de seguir para completar el envío de las asistencias automáticas fallidas.



23. Diagrama de secuencia - Send failed automatic attendance requests

5.3.14. Set enter automatic attendance

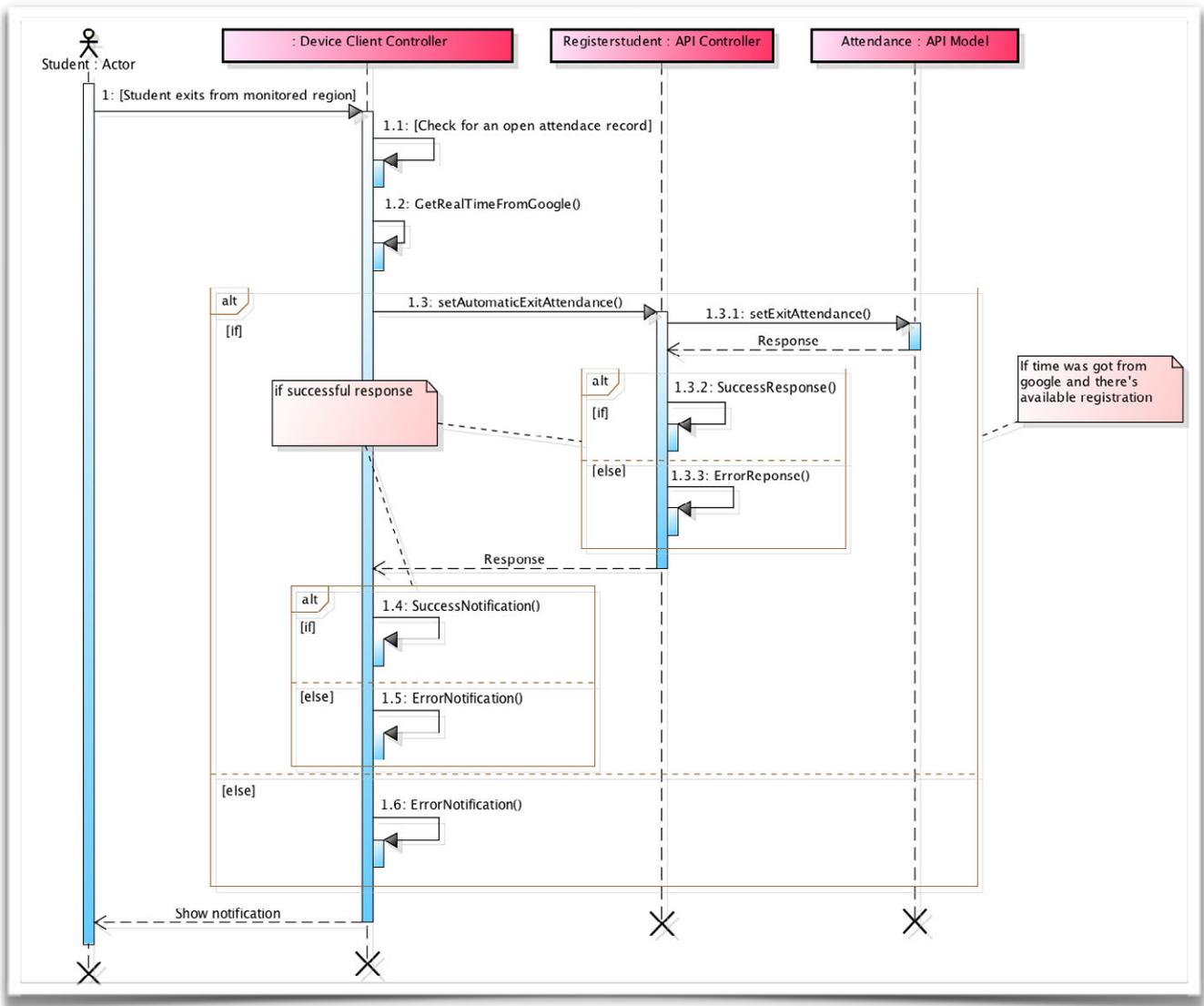
Este diagrama de secuencia muestra el proceso que un alumno debe de seguir para completar el envío de asistencias automáticas al entrar en una región monitorizada (asignatura). El proceso se inicia de manera automática por el sistema cuando detecta que se ha cruzado la región del beacon.



24. Diagrama de secuencia - Set enter automatic attendance

5.3.15. Set exit automatic attendance

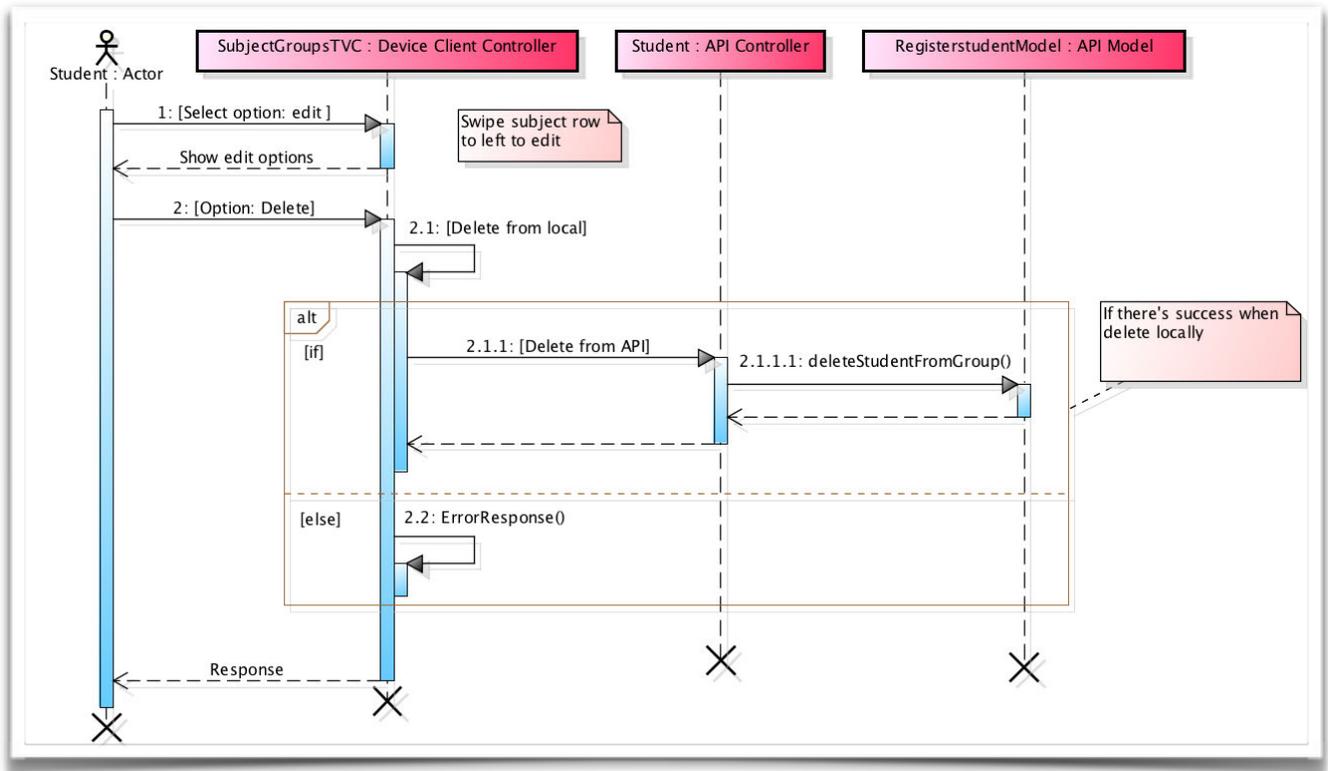
Este diagrama de secuencia muestra el proceso que un alumno debe de seguir para completar el envío de asistencias automáticas al salir en una región monitorizada (asignatura). El proceso se inicia de manera automática por el sistema cuando detecta que se ha cruzado la región del beacon.



25. Diagrama de secuencia - Set exit automatic attendance

5.3.16. Unjoin group

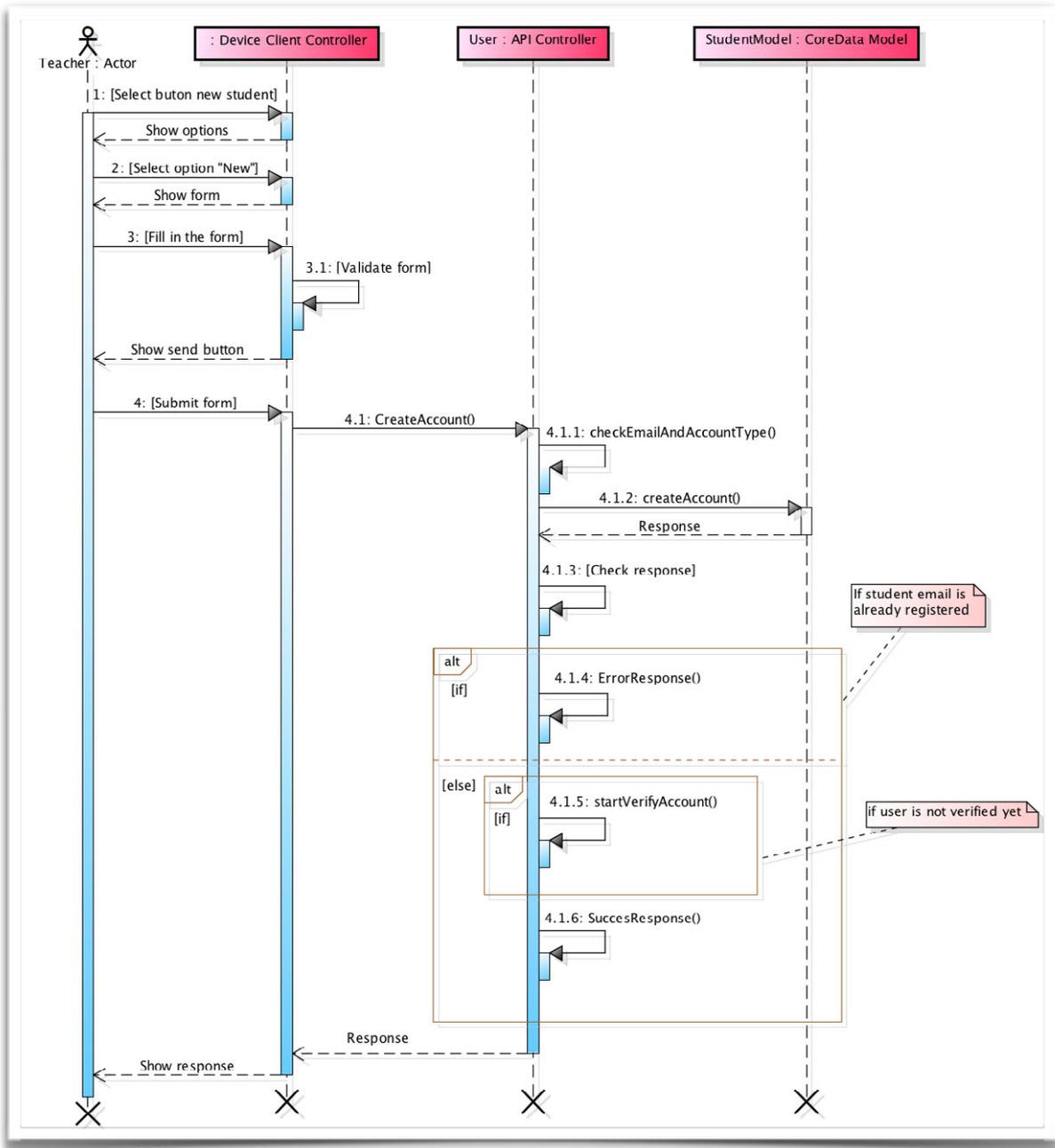
Este diagrama de secuencia muestra el proceso para que un alumno pueda eliminar la matrícula que tiene en un grupo. Al eliminar dicha matrícula, se perderán todos los datos de asistencia asociados.



26. Diagrama de secuencia - Unjoin group

5.3.17. Create and register student account

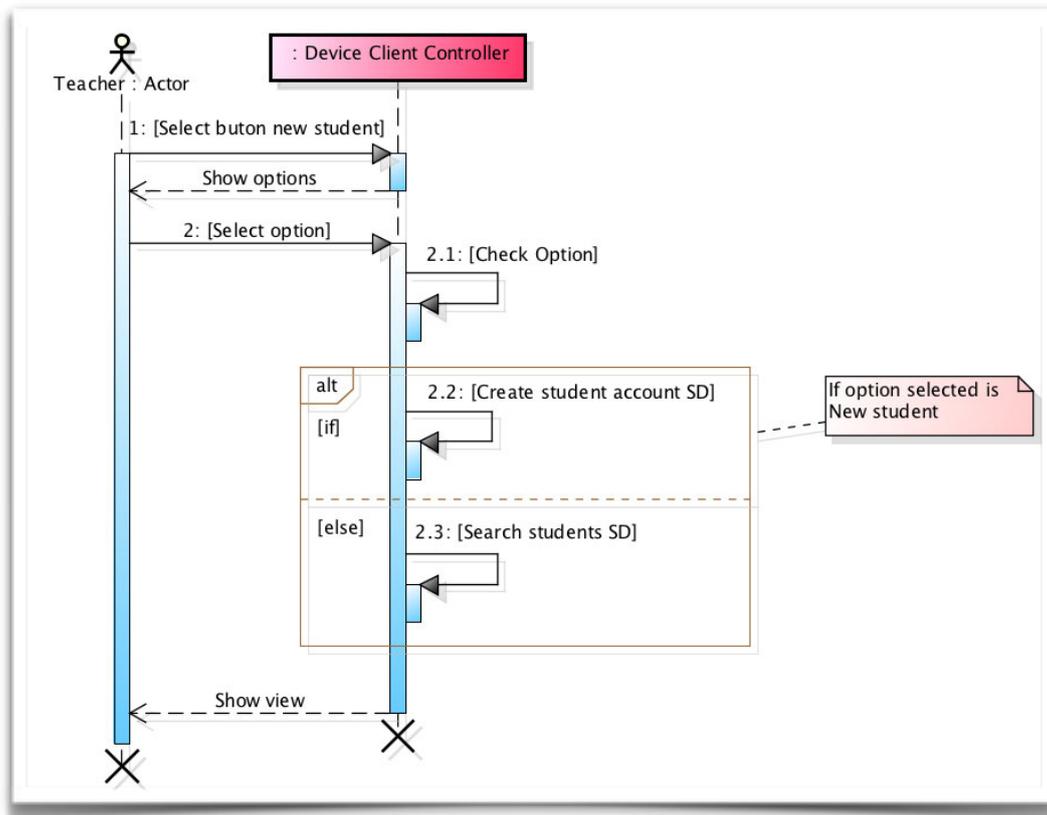
Este diagrama de secuencia muestra el proceso para que un profesor pueda crear una nueva cuenta de alumno. Se realizará una matrícula en el grupo seleccionado para este nuevo alumno.



27. Diagrama de secuencia - Create and register student account

5.3.18. Register student into group

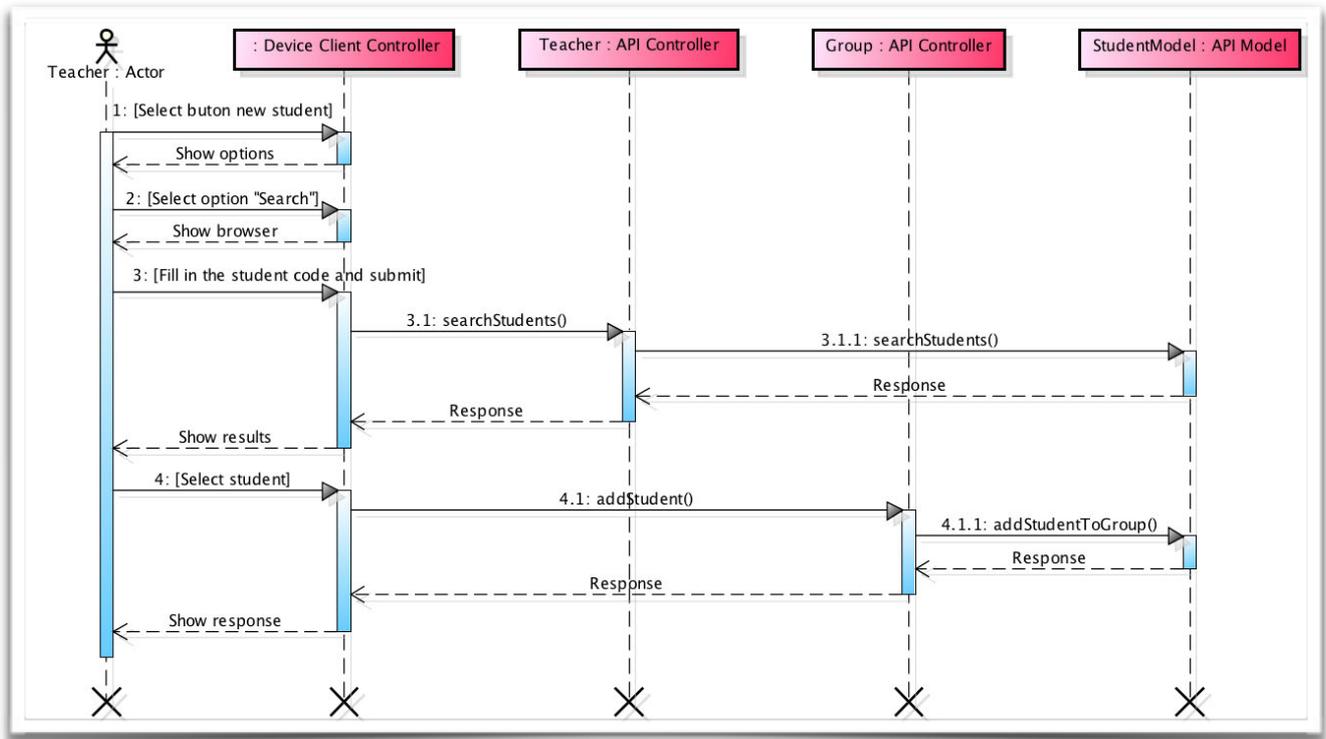
Este diagrama de secuencia muestra el inicio del proceso para que un profesor pueda matricular un alumno en un grupo. El proceso se completa bien creando una nueva cuenta de alumno o bien buscando uno ya existente en el sistema.



28. Diagrama de secuencia - Register student into group

5.3.19. Search and register student

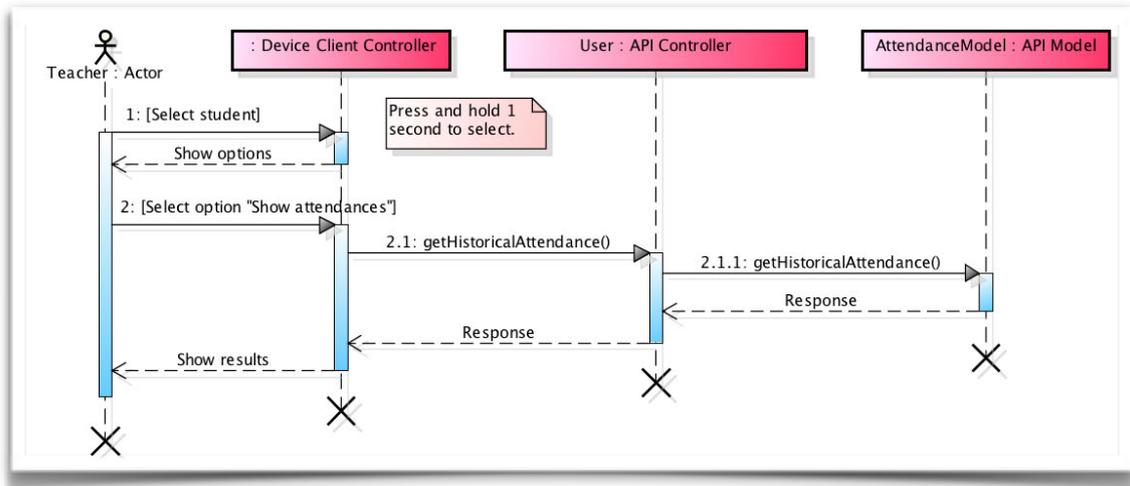
Este diagrama de secuencia muestra el proceso para que un profesor pueda buscar una cuenta de alumno y matricularlo en el grupo seleccionado.



29. Diagrama de secuencia - Search and register student

5.3.20. See all student attendance records

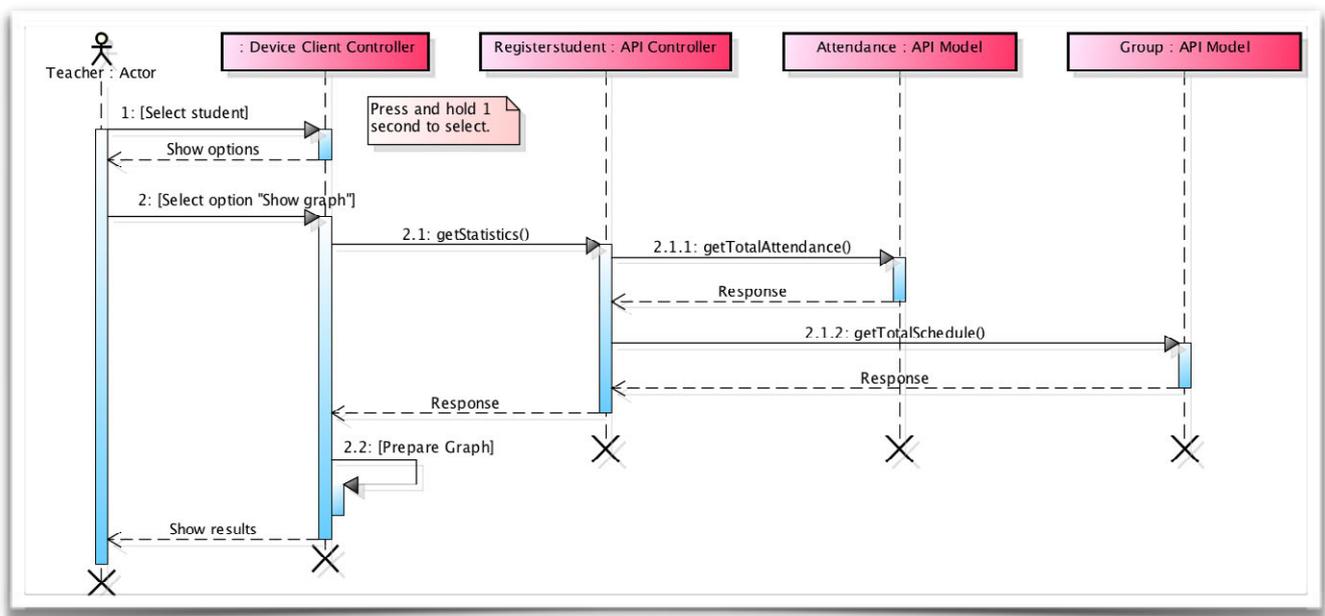
En este diagrama de secuencia se representa el proceso que un profesor debe de realizar para consultar el histórico de asistencias de un alumno.



30. Diagrama de secuencia - See all student attendance records

5.3.21. See student attendance graph

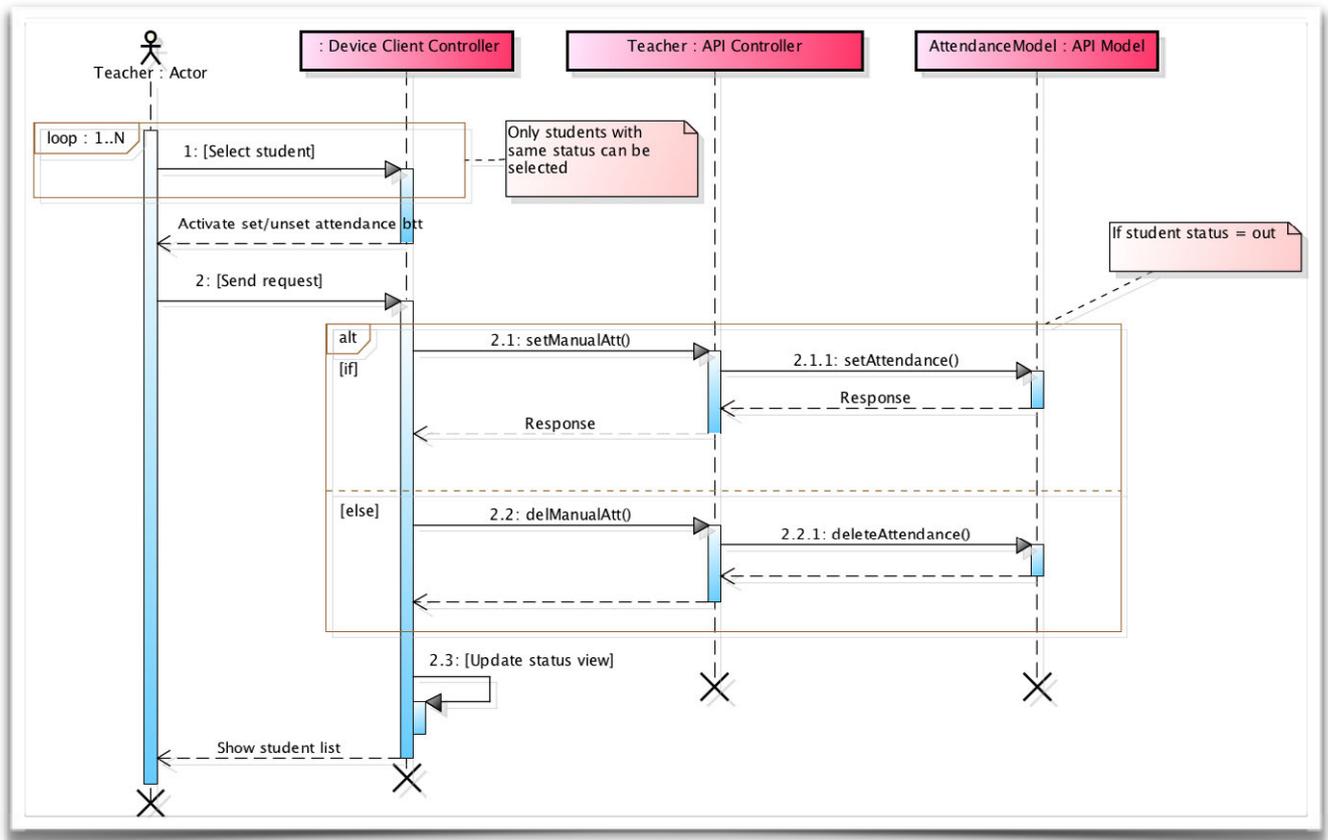
En este diagrama de secuencia se representa el proceso que un profesor debe de realizar para consultar el gráfico de asistencias de un alumno.



31. Diagrama de secuencia - See student attendance graph

5.3.22. Set/unset attendance

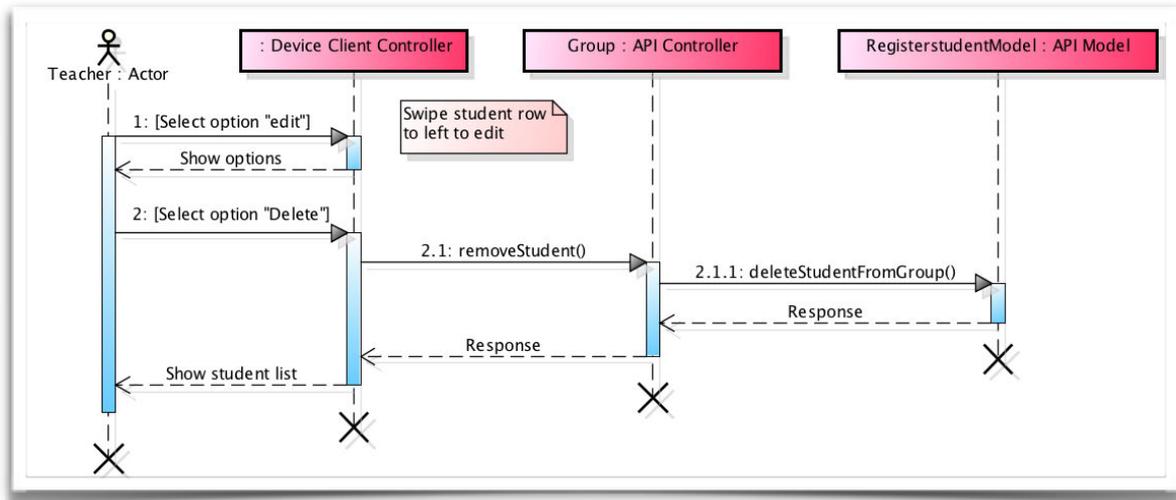
En este diagrama de secuencia se representa el proceso que un profesor debe de realizar para modificar el estado de asistencia de un alumno.



32. Diagrama de secuencia - Set/unset attendance

5.3.23. Unregister student from group

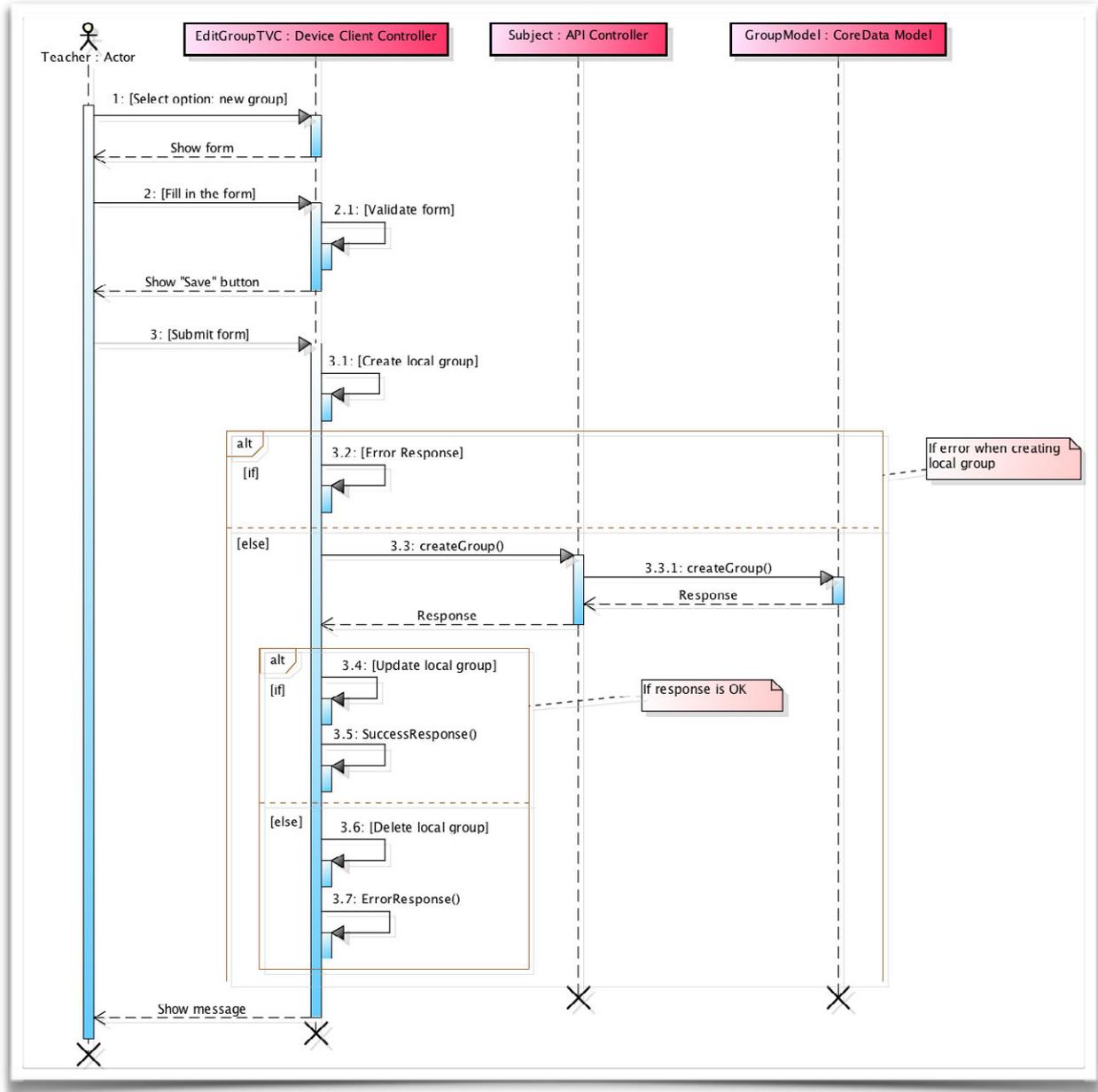
Este diagrama de secuencia muestra el proceso para que un profesor pueda eliminar la matricula de un alumno.



33. Diagrama de secuencia - Unregister student from group

5.3.24. Create group

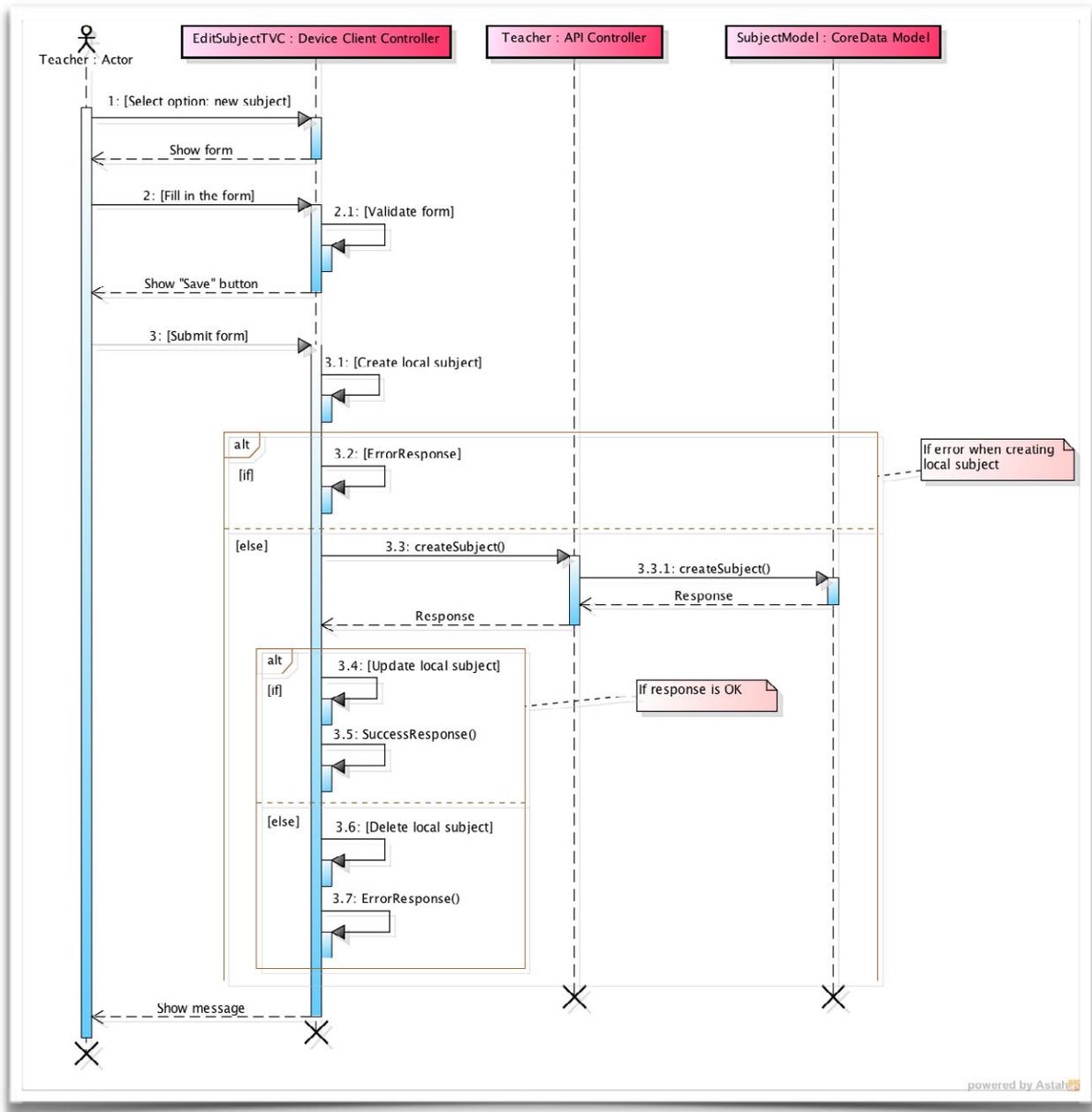
En este diagrama de secuencia se representa el proceso que un profesor debe de realizar para crear un nuevo grupo en una asignatura.



34. Diagrama de secuencia - Create group

5.3.25. Create subject

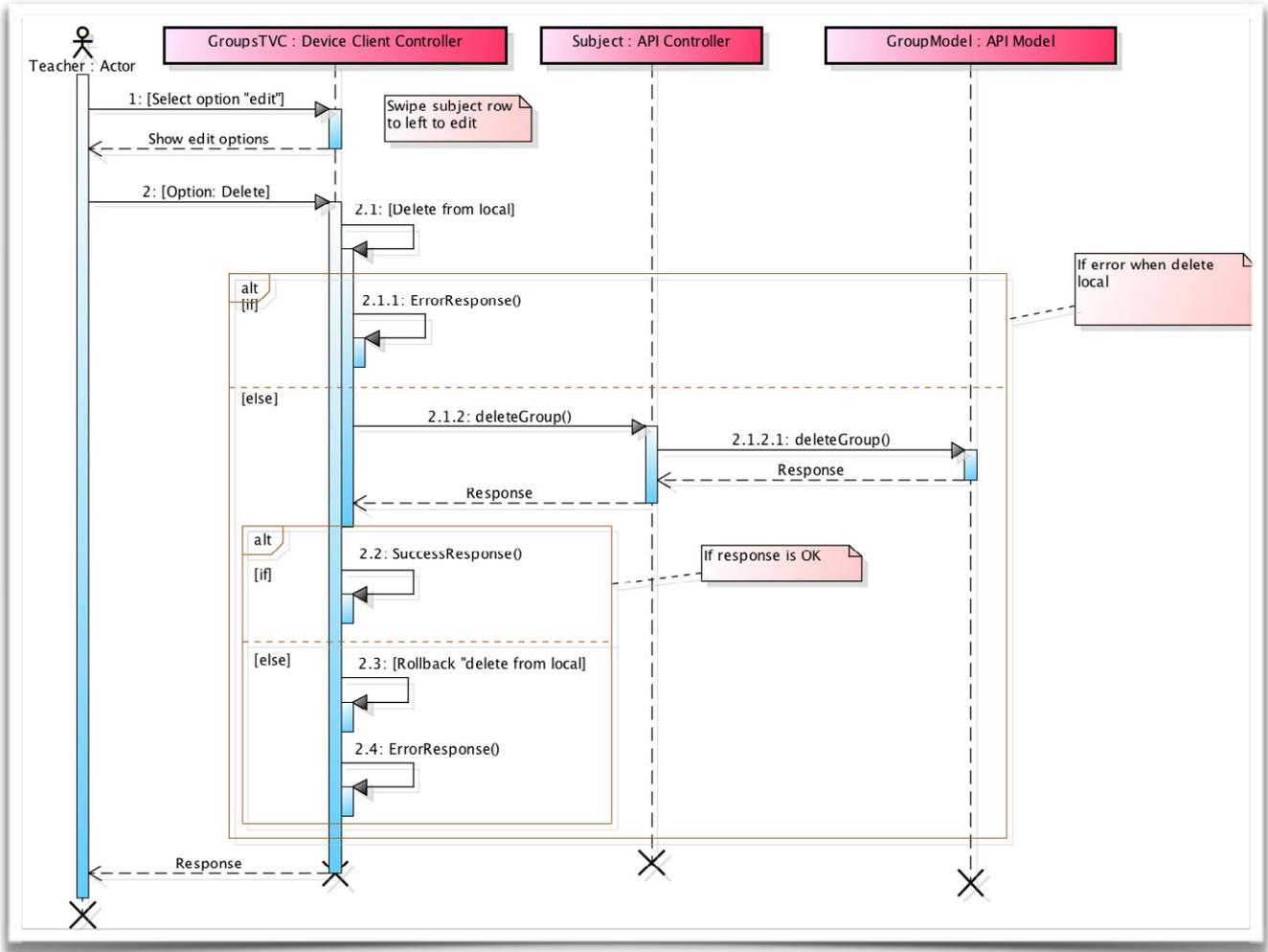
En este diagrama de secuencia se representa el proceso que un profesor debe de realizar para crear una nueva asignatura.



35. Diagrama de secuencia - Create subject

5.3.26. Delete group

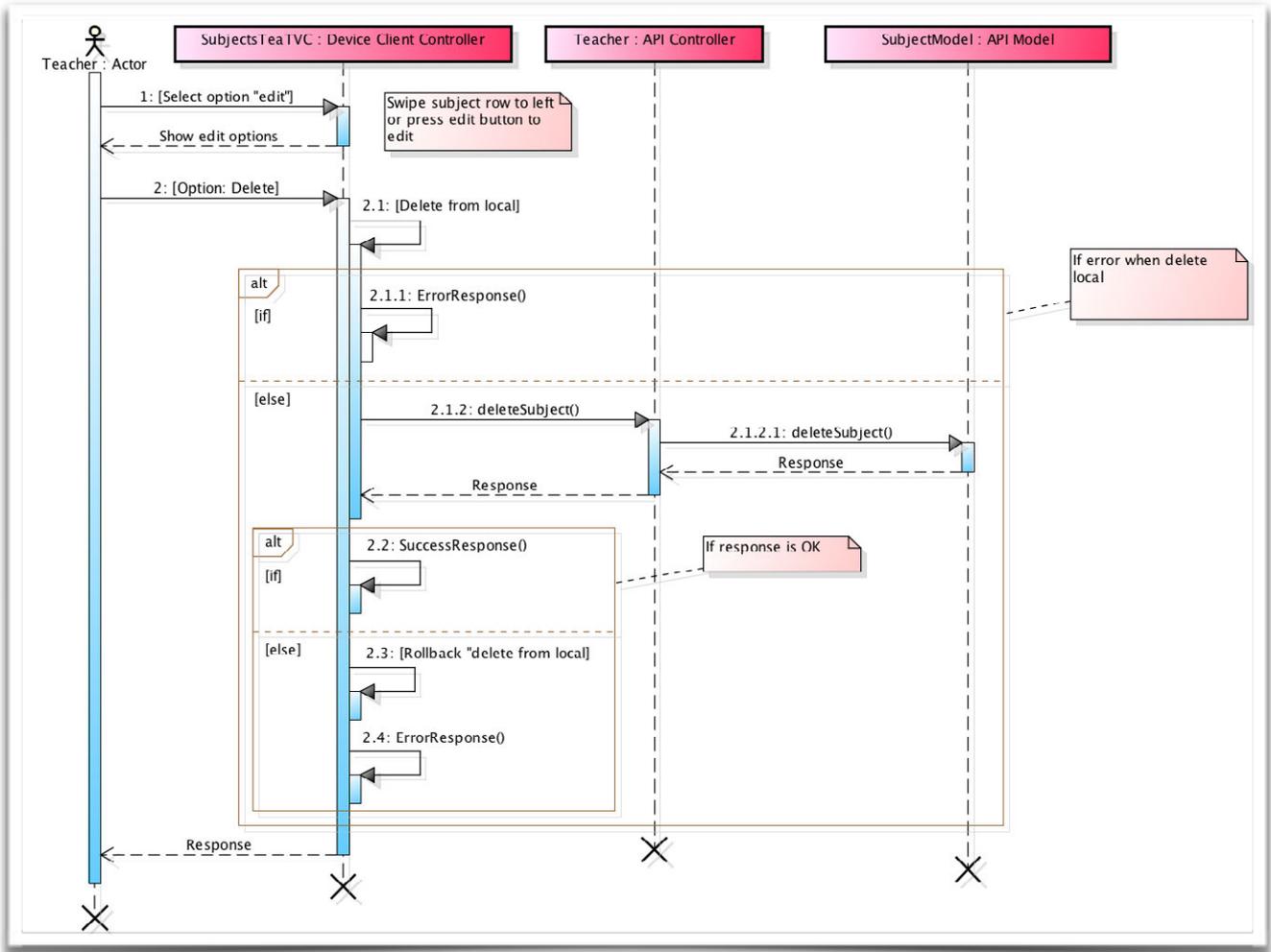
En este diagrama de secuencia se representa el proceso que un profesor debe de realizar para eliminar un grupo de una asignatura.



36. Diagrama de secuencia - Delete group

5.3.27. Delete subject

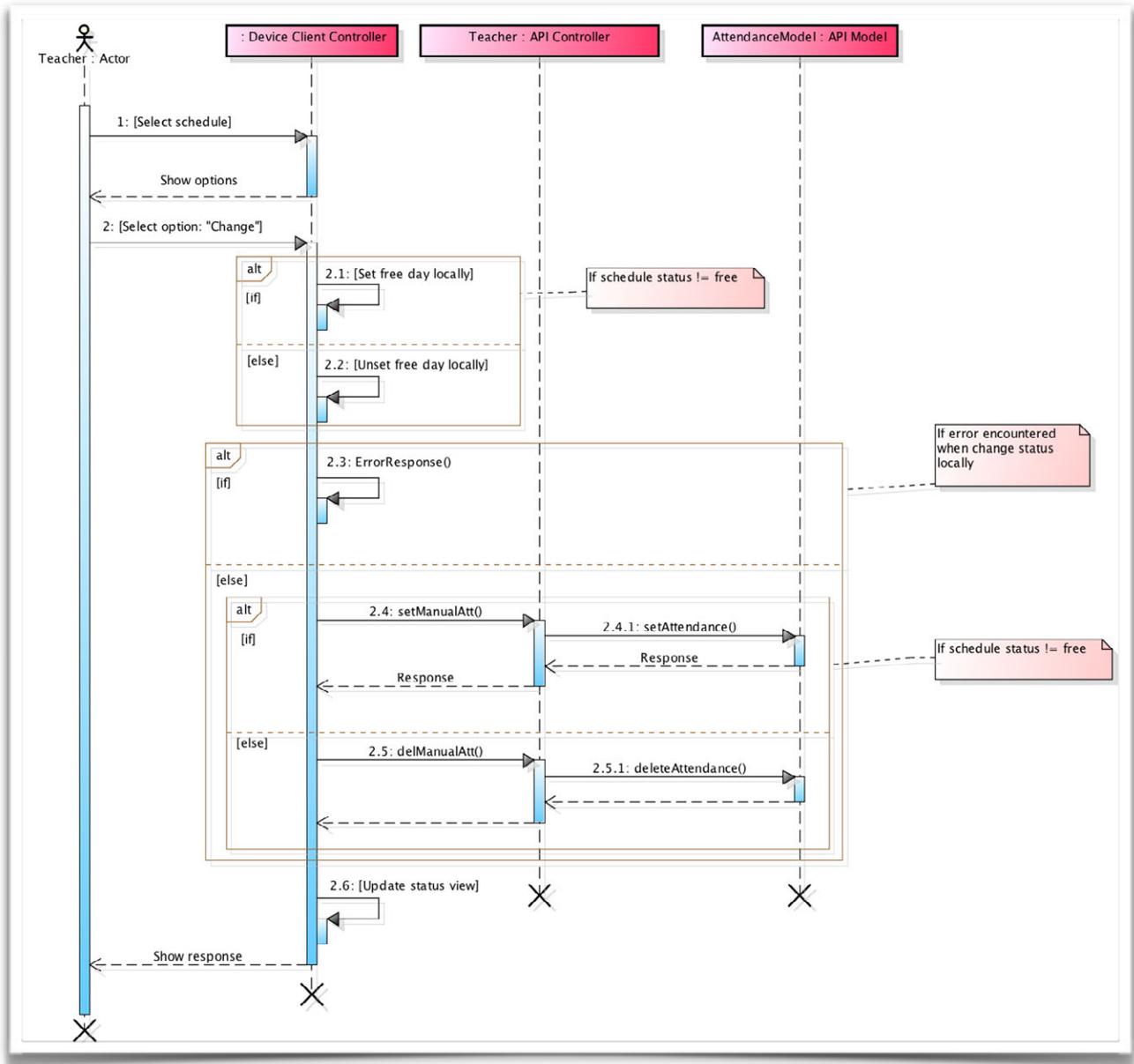
En este diagrama de secuencia se representa el proceso que un profesor debe de realizar para eliminar una asignatura.



37. Diagrama de secuencia - Delete subject

5.3.28. Set/unset free day

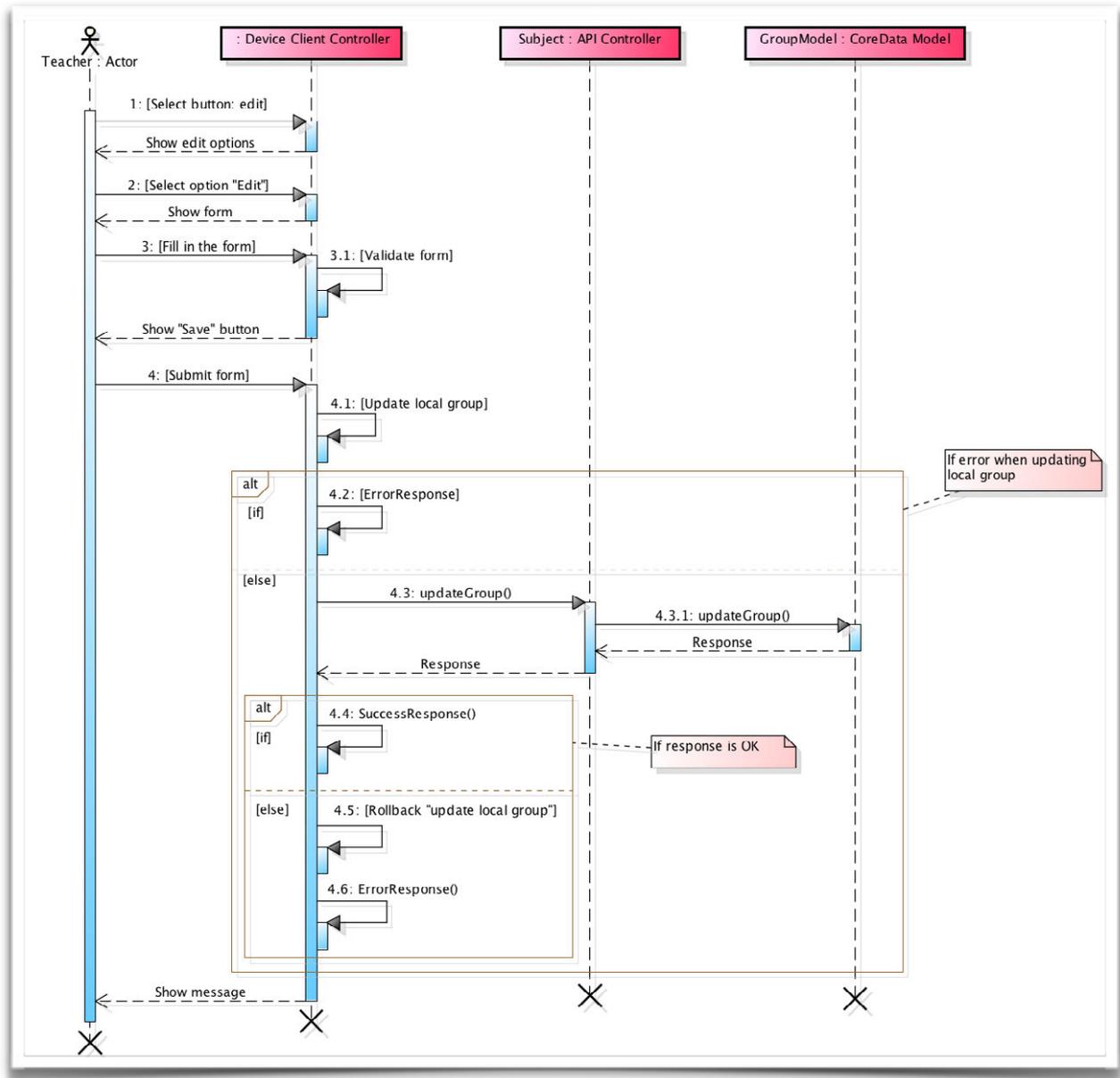
En este diagrama de secuencia se representa el proceso que un profesor debe de realizar para fijar un día del horario como festivo u obligatorio.



38. Diagrama de secuencia - Set/unset free day

5.3.29. Update group

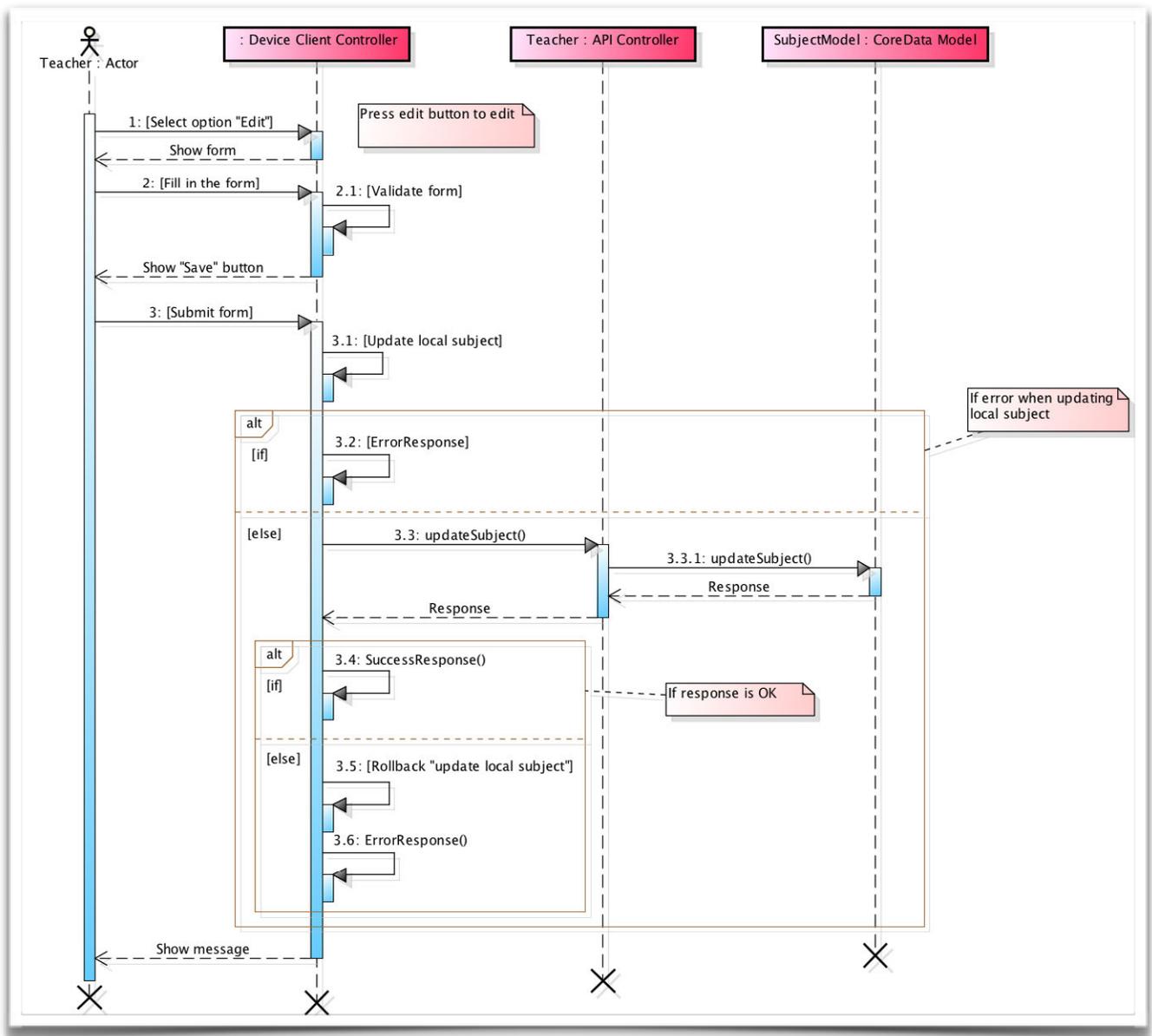
En este diagrama de secuencia se representa el proceso que un profesor debe de realizar para modificar la información de un grupo.



39. Diagrama de secuencia - Update group

5.3.30. Update subject

En este diagrama de secuencia se representa el proceso que un profesor debe de realizar para modificar la información de una asignatura



40. Diagrama de secuencia - Update subject

6. PRUEBAS

En este apartado se documentan las pruebas realizadas al sistema para su validación. Debido a la extensión de las numerosas pruebas realizadas al sistema, solo se han incluido en esta documentación las más importantes o aquellas que validan las principales características.

6.1. Casos de prueba

6.1.1. Log in

CP Login - 1	
Caso de uso:	Log in
Objetivo:	El objetivo de este caso de prueba es comprobar el funcionamiento del acceso de los usuarios a la herramienta.
Entrada:	Email y contraseña válidos
Salida:	Acceso a la herramienta
Condiciones:	Usuario registrado en la herramienta.
Resultado esperado:	Acceso a la herramienta

CP Login - 2	
Caso de uso:	Log in
Objetivo:	El objetivo de este caso de prueba es comprobar el funcionamiento en caso de intento de acceso no autorizado de un usuario a la herramienta.
Entrada:	Email y/o contraseña incorrectos. Email y/o contraseña generados aleatoriamente.
Salida:	Error de validación y mensaje al usuario
Condiciones:	Usuario no registrado en el sistema o usuario registrado en el sistema pero con datos de acceso distintos a los introducidos en la entrada del caso de prueba.
Resultado esperado:	Error de validación y mensaje al usuario

CP Login - 3	
Caso de uso:	Log in
Objetivo:	El objetivo de este caso de prueba es comprobar el funcionamiento en caso de intento de acceso de un usuario sin verificar a la herramienta.
Entrada:	Email y contraseña válidos
Salida:	Error de verificación de usuario y mensaje al usuario
Condiciones:	Usuario registrado en la herramienta.
Resultado esperado:	Error de verificación de usuario y mensaje al usuario

6.1.2. Sign Up

CP Signup - 1	
Caso de uso:	Sign up
Objetivo:	El objetivo de este caso de prueba es comprobar el funcionamiento en caso de nuevo registro en el sistema
Entrada:	Nombre, email (y código en caso de alumno) válidos
Salida:	Registro completado correctamente
Condiciones:	El usuario es un nuevo usuario que desea crear una nueva cuenta en la herramienta. Los datos deben de satisfacer los requisitos del formulario (longitud, email válido)
Resultado esperado:	Registro completado correctamente

CP Signup - 2	
Caso de uso:	Sign up
Objetivo:	El objetivo de este caso de prueba es comprobar el comportamiento en caso de que un usuario ya registrado en la herramienta intente de nuevo el registro.
Entrada:	Nombre, email (y código en caso de alumno) válidos
Salida:	Error de usuario ya registrado y mensaje al usuario.
Condiciones:	El usuario ya está dado de alta en la herramienta. Los datos deben de satisfacer los requisitos del formulario (longitud, email válido)
Resultado esperado:	Error de usuario ya registrado y mensaje al usuario.

6.1.3. Email verification

CP Email verification - 1	
Caso de uso:	Email verification
Objetivo:	El objetivo de este caso de prueba es comprobar el comportamiento que verifica a un usuario en el sistema.
Entrada:	Abrir enlace enviado por el sistema para la verificación.
Salida:	Formulario de contraseña y mensaje de confirmación.
Condiciones:	El usuario ha completado el registro en la herramienta y dispone de un enlace válido
Resultado esperado:	Formulario de contraseña y mensaje de confirmación.

CP Email verification - 2	
Caso de uso:	Email verification
Objetivo:	El objetivo de este caso de prueba es comprobar el comportamiento al intentar verificar un usuario con un link de verificación incorrecto.
Entrada:	Abrir enlace de verificación incorrecto.
Salida:	Error de enlace y mensaje al usuario
Condiciones:	El usuario ha completado el registro en la herramienta pero el enlace de verificación es incorrecto o ha caducado
Resultado esperado:	Error de enlace y mensaje al usuario

6.1.4. Resend email verification

CP Resend email verification - 1	
Caso de uso:	Resend email verification
Objetivo:	Comprobar el comportamiento de la herramienta al solicitar un nuevo email de verificación.
Entrada:	Envío de solicitud de nuevo email de verificación. El email asociado al usuario se incluye en la solicitud.
Salida:	Email con link de verificación.
Condiciones:	El usuario está registrado en la herramienta y no ha verificado su cuenta.
Resultado esperado:	Email con link de verificación.

CP Resend email verification - 2	
Caso de uso:	Resend email verification
Objetivo:	Comprobar el comportamiento de la herramienta al solicitar un nuevo email de verificación.
Entrada:	Envío de solicitud de nuevo email de verificación. El email asociado al usuario se incluye en la solicitud.
Salida:	No se envía el email con el link de verificación y se notifica al usuario del error.
Condiciones:	El usuario no está registrado en la herramienta o su cuenta ya está verificada
Resultado esperado:	No se envía el email con el link de verificación y se notifica al usuario del error.

6.1.5. Forgot password

CP Forgot password - 1	
Caso de uso:	Forgot password
Objetivo:	Comprobar el comportamiento de la herramienta cuando un usuario no recuerda su contraseña o desea restablecerla.
Entrada:	Envío de solicitud para recuperar contraseña. El email asociado al usuario se incluye en la solicitud.
Salida:	Email con link de verificación.
Condiciones:	El usuario está registrado en la herramienta y ha verificado su cuenta.
Resultado esperado:	Email con link de verificación.

CP Forgot password - 1	
Caso de uso:	Forgot password
Objetivo:	Comprobar el comportamiento de la herramienta cuando un usuario no recuerda su contraseña o desea restablecerla.
Entrada:	Envío de solicitud para recuperar contraseña. El email asociado al usuario se incluye en la solicitud.
Salida:	No se inicia el proceso para restablecer la contraseña y se notifica al usuario del error.
Condiciones:	El usuario no está registrado en la herramienta o no ha verificado su cuenta.
Resultado esperado:	No se inicia el proceso para restablecer la contraseña y se notifica al usuario del error.

6.1.6. Logout

CP Logout - 1	
Caso de uso:	Logout
Objetivo:	Comprobar el comportamiento de la herramienta cuando un usuario desea desconectarse del sistema.
Entrada:	Solicitud de desconexión.
Salida:	Desconexión del usuario y ventana de login.
Condiciones:	El usuario debe haber ingresado previamente al sistema.
Resultado esperado:	Desconexión del usuario y ventana de login.

6.1.7. Join group

CP Join group - 1	
Caso de uso:	Join group
Objetivo:	Comprobar el comportamiento de la herramienta cuando un alumno desea inscribirse en un grupo de una asignatura.
Entrada:	Envío del grupo seleccionado y su código de acceso.
Salida:	Se realiza la matrícula del alumno en el grupo solicitado.
Condiciones:	Código de acceso válido.
Resultado esperado:	Se realiza la matrícula del alumno en el grupo solicitado.

CP Join group - 2	
Caso de uso:	Join group
Objetivo:	Comprobar el comportamiento de la herramienta cuando un alumno desea inscribirse en un grupo de una asignatura.
Entrada:	Envío del grupo seleccionado y su código de acceso.
Salida:	No se realiza la matrícula del alumno y se muestra mensaje de error.
Condiciones:	Código de acceso incorrecto.
Resultado esperado:	No se realiza la matrícula del alumno y se muestra mensaje de error.

6.1.8. Search subject groups

CP Search subject groups - 1	
Caso de uso:	Search subject groups
Objetivo:	Comprobar el comportamiento de la herramienta cuando un alumno realiza una búsqueda de grupos
Entrada:	Envío de código de búsqueda.
Salida:	Se muestran todos los grupos relacionados con el código insertado.
Condiciones:	El término de búsqueda está asociado a una asignatura y debe ser el proporcionado por el profesor.
Resultado esperado:	Se muestran todos los grupos relacionados con el código insertado.

6.1.9. Unjoin group

CP Unjoin group - 1	
Caso de uso:	Unjoin group
Objetivo:	Comprobar el comportamiento de la herramienta cuando un alumno elimina la matrícula existente en un grupo de una asignatura
Entrada:	Enviar petición de borrado del grupo seleccionado
Salida:	Se elimina la matrícula existente para dicho grupo
Condiciones:	El alumno debe de estar matriculado de al menos un grupo
Resultado esperado:	Se elimina la matrícula existente para dicho grupo

6.1.10. Set enter automatic attendance

CP Set enter automatic attendance - 1	
Caso de uso:	Set enter automatic attendance
Objetivo:	Comprobar el comportamiento de la herramienta cuando un alumno entra en una asignatura que está siendo monitorizada a través de un active beacon
Entrada:	El alumno entra en el radio de alcance del active beacon
Salida:	Se almacena la hora de entrada del alumno a la asignatura
Condiciones:	El alumno debe de estar matriculado en la asignatura, tener conexión a internet y el bluetooth activado en el dispositivo móvil
Resultado esperado:	Se almacena la hora de entrada del alumno a la asignatura

6.1.11. Set exit automatic attendance

CP Set exit automatic attendance - 1	
Caso de uso:	Set exit automatic attendance
Objetivo:	Comprobar el comportamiento de la herramienta cuando un alumno sale de una asignatura que está siendo monitorizada a través de un active beacon
Entrada:	El alumno sale del radio de alcance del active beacon
Salida:	Se almacena la hora de salida del alumno a la asignatura
Condiciones:	El alumno debe de estar matriculado en la asignatura, tener conexión a internet y el bluetooth activado en el dispositivo móvil. Es necesario haber marcado la entrada en la asignatura previamente.
Resultado esperado:	Se almacena la hora de salida del alumno a la asignatura

6.1.12. See all attendance records

CP See all attendance records - 1	
Caso de uso:	See all attendance records
Objetivo:	Comprobar el comportamiento de la herramienta cuando un alumno desea consultar el histórico completo de asistencias de una asignatura.
Entrada:	Envío de asignatura para consultar
Salida:	Histórico detallado de asistencias de una asignatura
Condiciones:	El alumno debe estar matriculado en la asignatura
Resultado esperado:	Histórico detallado de asistencias de una asignatura

6.1.13. See attendances graph

CP See attendances graph - 1	
Caso de uso:	See attendances graph
Objetivo:	Comprobar el comportamiento de la herramienta cuando un alumno desea consultar el gráfico de asistencias de una asignatura.
Entrada:	Envío de asignatura para consultar
Salida:	Gráfico de asistencias de una asignatura
Condiciones:	El alumno debe estar matriculado en la asignatura
Resultado esperado:	Gráfico de asistencias de una asignatura

6.1.14. See automatic attendances log

CP See automatic attendances log - 1	
Caso de uso:	See automatic attendances log
Objetivo:	Comprobar el comportamiento de la herramienta cuando un alumno desea consultar el histórico de asistencias automáticas enviadas al sistema.
Entrada:	Envío de asignatura para consultar
Salida:	Histórico detallado de asistencias automáticas enviadas
Condiciones:	El alumno debe estar matriculado en la asignatura
Resultado esperado:	Histórico detallado de asistencias automáticas enviadas

6.1.15. Send failed automatic attendances requests

CP Send failed automatic attendances requests - 1	
Caso de uso:	Send failed automatic attendances requests
Objetivo:	Comprobar el comportamiento de la herramienta cuando un alumno desea enviar las notificaciones erróneas de asistencia automática
Entrada:	Envío de la asistencia errónea que se desea almacenar
Salida:	Se almacena la hora de entrada y salida del alumno para la asignatura
Condiciones:	El alumno debe estar matriculado en la asignatura y existe al menos una asistencia errónea en el log de envíos automáticos.
Resultado esperado:	Se almacena la hora de entrada y salida del alumno para la asignatura

6.1.16. See all student attendances records

CP See all student attendances records - 1	
Caso de uso:	See all student attendances records
Objetivo:	Comprobar el comportamiento de la herramienta cuando un profesor desea consultar el histórico de asistencias de un alumno
Entrada:	Envío del alumno que se desea consultar
Salida:	Histórico detallado de asistencias de una asignatura
Condiciones:	El alumno debe estar matriculado en la asignatura
Resultado esperado:	Histórico detallado de asistencias de una asignatura

6.1.17. Create and register student account

CP Create and register student account - 1	
Caso de uso:	Create and register student account
Objetivo:	Comprobar el comportamiento de la herramienta cuando un profesor crea una nueva cuenta de alumno en un grupo
Entrada:	La información del alumno (nombre, email y código)
Salida:	Alumno creado y matriculado en el grupo
Condiciones:	El profesor debe tener al menos un grupo creado
Resultado esperado:	Alumno creado y matriculado en el grupo

6.1.18. Search and register student

CP Search and register student - 1	
Caso de uso:	Search and register student
Objetivo:	Comprobar el comportamiento de la herramienta cuando un profesor crea una nueva cuenta de alumno en un grupo
Entrada:	Código del alumno a buscar
Salida:	Alumnos resultantes de la búsqueda
Condiciones:	N/A
Resultado esperado:	Alumnos resultantes de la búsqueda

6.1.19. Register student into group

CP Register student into group - 1	
Caso de uso:	Register student into group
Objetivo:	Comprobar el comportamiento de la herramienta cuando un profesor inicia el proceso de matriculación del alumno. Para finalizar este proceso se debe crear un alumno nuevo o buscar uno existente en el sistema.
Entrada:	Selección del grupo en el que se va añadir el nuevo alumno
Salida:	Mostrar las opciones que existen para matricular un nuevo alumno
Condiciones:	El profesor debe tener al menos un grupo creado
Resultado esperado:	Mostrar las opciones que existen para matricular un nuevo alumno

6.1.20. Unregister student from group

CP Unregister student from group - 1	
Caso de uso:	Unregister student from group
Objetivo:	Comprobar el comportamiento de la herramienta cuando un profesor desea eliminar la matrícula existente de un alumno para un grupo concreto
Entrada:	Alumno y grupo
Salida:	Se elimina la matrícula del alumno para dicho grupo
Condiciones:	Debe existir al menos un alumno matriculado en el grupo
Resultado esperado:	Se elimina la matrícula del alumno para dicho grupo

6.1.21. See student attendance graph

CP See student attendance graph - 1	
Caso de uso:	See student attendance graph
Objetivo:	Comprobar el comportamiento de la herramienta cuando un profesor desea consultar el gráfico de asistencias de un alumno
Entrada:	Envío de alumno y asignatura para consultar
Salida:	Gráfico de asistencias del alumno
Condiciones:	El alumno debe estar matriculado en la asignatura
Resultado esperado:	Gráfico de asistencias del alumno

6.1.22. Set/unset attendance

CP Set/unset attendance - 1	
Caso de uso:	Set/unset attendance
Objetivo:	Comprobar el comportamiento de la herramienta cuando un profesor desea modificar el estado de asistencia de un alumno
Entrada:	Alumno (o alumnos) de los que se desea cambiar el estado
Salida:	Listado actualizado de estados de los alumnos
Condiciones:	El alumno (o alumnos) debe estar matriculado en la asignatura
Resultado esperado:	Listado actualizado de estados de los alumnos

6.1.23. Create subject

CP Create subject - 1	
Caso de uso:	Create subject
Objetivo:	Comprobar el comportamiento de la herramienta cuando un profesor desea crear una nueva asignatura
Entrada:	Información de la asignatura (nombre, código) y en caso de usar monitorización los datos del active beacon (UUID, major, minor)
Salida:	Nueva asignatura creada
Condiciones:	N/A
Resultado esperado:	Nueva asignatura creada

6.1.24. Update subject

CP Update subject - 1	
Caso de uso:	Update subject
Objetivo:	Comprobar el comportamiento de la herramienta cuando un profesor desea actualizar la información de una asignatura
Entrada:	Información de la asignatura (nombre, código) y en caso de usar monitorización los datos del active beacon (UUID, major, minor)
Salida:	Asignatura actualizada
Condiciones:	El profesor debe tener al menos una asignatura
Resultado esperado:	Asignatura actualizada

6.1.25. Delete subject

CP Delete subject	
Caso de uso:	Delete subject
Objetivo:	Comprobar el comportamiento de la herramienta cuando un profesor desea borrar una asignatura
Entrada:	Asignatura que se quiere borrar
Salida:	Asignatura borrada
Condiciones:	El profesor debe tener al menos una asignatura
Resultado esperado:	Asignatura borrada

6.1.26. Create group

CP Create group - 1	
Caso de uso:	Create group
Objetivo:	Comprobar el comportamiento de la herramienta cuando un profesor desea crear un nuevo grupo
Entrada:	Información del grupo (nombre, código y horario)
Salida:	Nuevo grupo creado
Condiciones:	El profesor debe tener al menos una asignatura creada previamente
Resultado esperado:	Nuevo grupo creado

6.1.27. Update group

CP Update group - 1	
Caso de uso:	Update group
Objetivo:	Comprobar el comportamiento de la herramienta cuando un profesor desea actualizar la información de un grupo
Entrada:	Información del grupo (nombre, código y horario)
Salida:	Grupo actualizado
Condiciones:	El profesor debe tener al menos un grupo creado previamente
Resultado esperado:	Grupo actualizado

6.1.28. Delete group

CP Delete group - 1	
Caso de uso:	Delete group
Objetivo:	Comprobar el comportamiento de la herramienta cuando un profesor desea borrar un grupo
Entrada:	Grupo que se quiere borrar
Salida:	Grupo borrado
Condiciones:	El profesor debe tener al menos un grupo creado previamente
Resultado esperado:	Grupo borrado

6.1.29. Set/unset free day

CP Set/unset free day - 1	
Caso de uso:	Set/unset free day
Objetivo:	Comprobar el comportamiento de la herramienta cuando un profesor desea cambiar es estado de un día del horario. Los estados posibles son festivo o lectivo
Entrada:	Día que se desea modificar
Salida:	Cambio de estado para el día seleccionado
Condiciones:	El profesor debe tener un grupo con horario creado. El estado cambia según el estado actual del día, es decir, si el día es lectivo y se aplica el cambio pasará a ser festivo y viceversa
Resultado esperado:	Cambio de estado para el día seleccionado

7. CONCLUSIONES

Este TFG ha sido desarrollado con un objetivo claro y bien definido: realizar el control de asistencias de los alumnos. Tras el desarrollo y las posteriores pruebas en diferentes entornos, el trabajo cumple favorablemente con dicho objetivo.

Se han realizado tres iteraciones del proceso evolutivo estudiando y mejorando en cada de ellas la funcionalidad del software, así como solventando los problemas encontrados. De dichas iteraciones, he decidido documentar solo la tercera y última que es la se puede encontrar en este documento.

En esta última versión del proyecto, nos encontramos con un prototipo maduro y muy cercano a un producto final listo para su explotación.

Al margen de este proyecto, se ha desarrollado una web informativa en la que se presenta el producto. Esta web, servirá además como punto de encuentro entre desarrollador y usuario y se pretende añadir diferentes secciones de soporte y preguntas frecuentes con el fin de ofrecer un producto completo. La web esta disponible desde el siguiente enlace:

<http://tutor.mafernandez.com>

7.1. Herramienta software Tutor

En primer lugar, el desarrollo de esta herramienta me ha servido para consolidar los conocimientos adquiridos a lo largo de mi experiencia universitaria.

Además, el completo desarrollo de un proyecto de estas características, me ha servido para comprender la importancia del estudio y análisis del proyecto desde sus orígenes, para evitar en la medida de lo posible problemas futuros en el sistema. Esta tarea es una tarea compleja, ya que resulta complicado obtener todos los requerimientos desde etapas tempranas y por regla general, en las últimas etapas se descubren también algunos problemas y requerimientos adicionales en el software.

7.2. Tecnología iBeacon y Bluetooth Low Energy

Tras las investigaciones y pruebas realizadas con iBeacon, es obligatorio decir que es una tecnología que se puede aplicar en numerosos entornos comerciales, laborales, sanitarios y evidentemente educativos con resultados bastante aceptables. Estamos ante una tecnología novedosa y con muchas posibilidades de éxito en cuanto a posicionamiento en interiores se refiere.

Las primeras aproximaciones de este proyecto no fueron dedicadas al control de asistencia de alumnos, sino que se intentó desarrollar un sistema de posicionamiento que se pudiera equiparar al ya conocido sistema GPS pero en entornos indoor. Lamentablemente, estas investigaciones no dieron los resultados de precisión esperados y se decidió abandonar dicha línea de investigación.

Bluetooth LE (low energy) o Bluetooth Smart fue introducido en el estándar bluetooth en 2010 con la especificación 4.0. Las mejoras que presenta frente su antecesor se pueden resumir en la siguiente tabla:

Especificación	Bluetooth clásico	Bluetooth LE
Distancia (teórica)	100 m	< 100 m
Transferencia	1-3 Mbits/s	1 Mbits/s
Rendimiento	0.7-2.1 Mbits/s	0.27 Mbits/s
Seguridad	56/128-bit	128-bit
Latencia	100 ms	6 ms
Tiempo mínimo de envío	100 ms	3 ms
Soporte de voz	Si	No
Consumo	< 30 mA	< 15 mA
Topología de red	Scatternet	Scatternet

7.3. Futuras versiones del sistema

El ámbito de este proyecto finaliza con el desarrollo de la aplicación para el sistema operativo iOS y la elaboración de esta documentación, no obstante y como meta personal a corto plazo, me gustaría seguir desarrollando la herramienta para otros sistemas operativos que existen en la actualidad como puedan ser Android o Windows Phone. Estos futuros desarrollos completarán mi formación como Graduado en Ingeniería de Software y me aportarán la base de conocimientos necesaria para emprender mi carrera profesional como tal.

Algunas ideas para las futuras versiones del sistema podrían incluir:

- Sincronización de datos cliente-servidor en todos los dispositivos asociados del usuario.
- Login a través de TouchID
- Notificaciones push
- Mejoras de seguridad en API
- Aplicación de estándares REST en la parte de API

8. VÍAS DE CONTINUACIÓN DE ESTE PROYECTO.

En el transcurso del desarrollo de este proyecto, no han sido pocas las ideas surgidas para la evolución de la herramienta Tutor, sin embargo se detallarán solo las más relevantes desde mi punto de vista y siempre pensando en el aporte que puedan ofrecer mejorando nuestra comunidad educativa.

8.1. Integración con plataforma de enseñanza Moodle.

Moodle es un sistema de gestión de cursos de libre distribución que ayuda a los profesores a crear comunidades de aprendizaje en línea. Este tipo de plataformas se conocen también como LCMS (Learning Content Management System).

La plataforma Moodle es de sobra conocida por toda la comunidad educativa formada por la Universidad Politécnica de Madrid, siendo unos de los puntos de encuentro principales entre profesor y alumno. En dicha plataforma los usuarios dependiendo del perfil de acceso, pueden encontrar listados de alumnos, el material utilizado en clase, las calificaciones obtenidas por las tareas realizadas y un largo etc.

Por todo esto y siendo Moodle un software abierto, se podría añadir sin ninguna dificultad un módulo que permita la comunicación entre la herramienta Tutor y dicha plataforma. De esta manera ambos sistemas se aprovecharían de las funcionalidades que ofrecen completando el sistema de gestión integral de educación.

8.2. Gestión académica integral.

El sistema Tutor desde su origen ha sido enfocado a la gestión académica de alumnos y las asistencias a clase de los mismos, pero ofrece un amplio abanico de posibilidades para aumentar sus prestaciones básicas en futuras versiones.

El sistema utiliza los listados de alumnos matriculados por lo que facilita una comunicación directa entre profesores y alumnos. Debido a dicho enfoque, se presenta interesante la posibilidad de implementar numerosas prestaciones que faciliten la gestión académica en su totalidad.

Algunas de estas prestaciones adicionales podrían ser:

- Notificaciones de próximos exámenes.
- Gestión de calificaciones.
- Reserva de tutorías y seguimiento de formación continua.

9. REFERENCIAS BIBLIOGRÁFICAS

En este apartado se incluye la bibliografía consultada para la realización del proyecto y para la redacción de la documentación.

- SCHMULLER, JOSEPH. *Aprendiendo UML*. <<http://www.ingenieriasystems.com/2013/10/Aprendiendo-UML-en-24-Horas-Autor--Joseph-Schmuller.html>> [Consulta: 30 de octubre 2014]
- APPLE. *Human Interface Guidelines*. <<https://developer.apple.com/library/ios/documentation/userexperience/conceptual/mobilehig/>> [Consulta: 30 de octubre 2014]
- [sin autoría reconocida]. *Requisitos no funcionales*. http://www.ecured.cu/index.php/Requisitos_no_funcionales. [Consulta: 30 de octubre 2014]
- L. MEDINA, JULIO. LOPEZ, PATRICIA. MEDIAVILLA, ELENA. Facultad de Ciencias, Universidad de Cantabria. *Programación Orientada a Objetos: Lenguajes, metodología y herramientas. Master de Computación*. <http://www.ctr.unican.es/asignaturas/MC_OO/Doc/Casos_de_uso.pdf> [Consulta: 30 de noviembre 2014]
- GUTIERREZ, JAVIER. *Jerarquías: Generalizaciones y Especializaciones*. <http://dryvalleycomputer.com/index.php/bases-de-datos/el-modelo-entidadrelacion/55-jerarquias-generalizaciones-y-especializaciones> [Consulta: 7 de noviembre de 2014]
- SPARX SYSTEMS ARGENTINA. *Diagrama de Secuencia UML*. http://www.sparxsystems.com.ar/resources/tutorial/uml2_sequencediagram.html. [Consulta: 10 de noviembre 2014]
- WIKIPEDIA. *Caso de prueba*. http://es.wikipedia.org/wiki/Caso_de_prueba. [Consulta: 13 de noviembre]

10.GLOSARIO

iOS: Sistema operativo de la empresa Apple Inc.

Active Beacon: Baliza activa. Pequeño dispositivo hardware encargado de la emisión de señal bluetooth para determinar presencia.

iOS Human Interface Guidelines: Guía de estilos para las aplicaciones desarrolladas en la plataforma iOS.

App Store: Servicio creado por Apple Inc. para la distribución de aplicaciones, que permite a los usuarios buscar y descargar dichas aplicaciones.

CRUD: Acrónimo de las operaciones Crear, Obtener, Actualizar y Borrar (del original Create, Read, Update and Delete)

UML: Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG (Object Management Group).

Crow's foot: Notación utilizada para representar entidades y relaciones en diferentes diagramas. Las cardinalidades se representan al final de la línea de la relación con diferentes formas.

Backend: Capa software que provee acceso simplificado a los datos persistentes del sistema.

IDE: Entorno de desarrollo integrado (siglas en inglés de integrated development environment), es un programa informático compuesto por un conjunto de herramientas de programación.

N/A: No aplica

Scatternet: Es un tipo de red informática ad hoc formada por dos o más dispositivos bluetooth.

