



**Universidad de Valladolid**

**E.T.S Ingeniería Informática**

**TRABAJO FIN DE GRADO**

Grado en Ingeniería Informática

**APLICACIÓN MÓVIL DOCENTE PARA  
FACILITAR EL APRENDIZAJE DE CONCEPTOS  
DE CAMPO Y POTENCIAL ELÉCTRICO**

Autor:

**D. Antonio Pesquera Rodríguez**

Tutor:

**D. Manuel Ángel González Delgado**



Quiero dedicar este proyecto a mi abuelo Eugenio. Siempre te estaré agradecido por toda tu atención y tu cariño, por todo lo que me enseñaste y por esa alegría y orgullo que sentías por los éxitos de tus hijos y nietos.  
¡Gracias!

Quiero dar las gracias a mi familia por su cariño y su apoyo, especialmente en los momentos difíciles. Gracias por haberme acompañado en las diferentes etapas de mi vida y haberme ayudado en todo lo que he necesitado.

También tengo que agradecer su ayuda a mi tutor, pues ha sido una parte muy importante en la realización de este proyecto. Su colaboración ha sido indispensable.

Y también quiero agradecerles a mis amigos Diego Calvo, Diego Herrero y Adrián Herreras, su apoyo y todos los buenos momentos que hemos compartido juntos.



## RESUMEN.

El proyecto consiste en el desarrollo de una aplicación móvil que complemente la docencia de uno de los temas de la asignatura de Física impartida en el Grado en Ingeniería Informática, como es el de los Campos Eléctricos.

La aplicación ofrece al alumno la posibilidad de repasar los conceptos teóricos que le han sido impartidos en las clases de Física, realizar cuestionarios de autoevaluación de dichos conocimientos, realizar cuestionarios sobre los conceptos impartidos y que su profesor pueda conocer los resultados y un simulador de campos eléctricos, que, dado un conjunto de cargas, dibuja las líneas de campo y muestra el potencial eléctrico total en un punto del espacio.

La aplicación está disponible para la plataforma Android desde la API 18 (comercialmente Android 4.3 Jelly Bean). Su diseño está pensado para que pueda ser ejecutada tanto en teléfonos móviles como en "tablets".

Junto a esta aplicación se ha desarrollado una aplicación secundaria, esta vez de escritorio, que permite al personal docente (o a quien le competa la responsabilidad de actualizar los contenidos que se muestran en la aplicación) crear cuestionarios y páginas de teoría de una forma cómoda. Esta aplicación secundaria es compatible con los principales sistemas operativos de escritorio, pues requiere una Máquina Virtual Java para ser ejecutada.

# Contenido

Capítulo I: Introducción.....	15
Visión general. Contextualización del Proyecto.....	16
Objetivos.....	17
Fundamento teórico.....	18
Campo Eléctrico.....	18
Flujo Eléctrico.....	19
Potencial Eléctrico.....	20
Energía Potencial.....	20
Contexto de Desarrollo.....	21
Desarrollo de la aplicación principal.....	21
Desarrollo de la aplicación complementaria.....	21
Redacción de la memoria.....	21
Capítulo II: Planificación.....	23
Roles y responsabilidades.....	24
Roles relativos al equipo del proyecto.....	24
Roles relativos a stakeholders.....	24
Definición de Riesgos.....	25
Recursos utilizados. Estimación de costes.....	26
Product Backlog.....	28
Primer Sprint.....	28
Requisitos a realizar. Sprint Backlog.....	28
Calendario del sprint.....	29
Segundo sprint.....	29
Requisitos a realizar. Sprint Backlog.....	29
Calendario del sprint.....	30
Tercer sprint.....	30
Requisitos a realizar. Sprint Backlog.....	30
Calendario del sprint.....	31
Cuarto sprint.....	31
Requisitos a realizar. Sprint Backlog.....	31
Calendario del sprint.....	32
Quinto sprint.....	32
Requisitos a realizar. Sprint Backlog.....	32
Calendario del sprint.....	33
Capítulo III: Análisis.....	35

Objetivos del Proyecto. ....	36
Definición de Actores. ....	37
Requisitos. ....	37
Requisitos funcionales.....	37
Requisitos no funcionales.....	40
Casos de Uso.....	40
Diagrama de Casos de Uso. ....	41
Descripción de los casos de uso. ....	42
Modelo de dominio. ....	47
Descripción del modelo de dominio.....	48
Diagramas de Secuencia.....	49
Consultar fundamentos teóricos.....	49
Validar frente al servidor.....	49
Realizar cuestionario de autoevaluación. ....	50
Tomar fotografía. ....	50
Enviar resultados.....	51
Descargar contenidos.....	51
Realizar simulación.....	52
Calcular potencial eléctrico. ....	53
Editar carga.....	53
Borrar carga.....	54
Generar pregunta de cuestionario.....	54
Realizar cuestionario de examen. ....	55
Generar archivo de teoría. ....	56
Capítulo IV: Diseño. ....	57
Modelos de diseño. ....	58
Consultar fundamentos teóricos.....	58
Realizar simulación.....	59
Realizar cuestionario de autoevaluación. ....	60
Descargar contenidos.....	61
Generar cuestionario (Aplicación complementaria).....	62
Generar archivo de teoría (Aplicación complementaria).....	63
Explicación de las clases de diseño. ....	64
MainActivity. ....	64
DropboxConnection. ....	64
TeoriaActivity.....	65
TeoriaPagerAdapter. ....	65

FragmentoTeoria.....	65
PrevioCuestionario.....	66
UtilesConexion.....	66
ValidarUsuario.....	66
CuestionarioActivity.....	67
CuestionarioPagerAdapter.....	67
GestorCorreo.....	68
ProcesadorFoto.....	68
FragmentoCuestionario.....	69
GeneradorMathJax.....	70
SimulacionActivity.....	70
Carga.....	71
HiloDibujo.....	71
DibujaLineas.....	72
Líneas.....	72
Cuestionario (Aplicación complementaria).....	73
HiloPersistenciaCuestionario (Aplicación complementaria).....	73
Teoria (Aplicación complementaria).....	74
HiloPersistenciaTeoria (Aplicación complementaria).....	74
Diagramas de secuencia.....	75
Consultar fundamentos teóricos.....	75
Realizar cuestionario de autoevaluación.....	77
Realizar simulación.....	80
Generar cuestionario.....	85
Generar archivo de teoría.....	88
Realizar cuestionario de examen.....	92
Descargar contenidos.....	99
Capítulo V: Implementación.....	103
Mostrar la teoría.....	104
MathJax.....	104
Cuestionarios.....	104
Simulación.....	105
Algoritmo de cálculo de líneas de campo.....	106
Generación de contenidos teóricos y de evaluación.....	107
Conexión con el repositorio.....	107
Dropbox.....	107
Otras consideraciones.....	108



Capítulo VI: Pruebas. ....	109
Pruebas del dominio de la aplicación. ....	110
Pruebas de la interfaz gráfica. ....	121
Capítulo VII: Conclusiones. ....	123
Conclusiones. ....	124
Mejoras futuras. ....	124
Generación del software multiplataforma. ....	124
Paralelizar el algoritmo de cálculo de líneas de campo. ....	125
Capítulo VIII: Bibliografía. ....	127
Bibliografía sobre planificación de proyectos. ....	128
Bibliografía sobre la aplicación complementaria. ....	128
Bibliografía sobre Android. ....	128
Bibliografía sobre Ingeniería del software. ....	130
Bibliografía sobre Física. ....	130
Anexo I: Manuales. ....	131
Manual de instalación. ....	132
Manual de usuario. Campos Eléctricos. ....	134
Visualizar la teoría. ....	134
Realizar cuestionarios de autoevaluación. ....	135
Realizar cuestionario de examen. ....	137
Simulación. ....	139
Manual de Usuario. Gestor de Contenidos. ....	141
Generar un cuestionario. ....	142
Generar una página de teoría. ....	143
Anexo II: Contenido del CD. ....	145

## Índice de tablas

Tabla 1: Roles relativos al equipo del proyecto.....	24
Tabla 2: Roles relativos a interfaces externas .....	24
Tabla 3: Riesgo: Desconocimiento de las tecnologías a utilizar .....	25
Tabla 4: Riesgo: Pérdida total o parcial de los datos.....	25
Tabla 5: Riesgo: Avería de uno de los dispositivos utilizados para el desarrollo .....	25
Tabla 6: Riesgo: Incapacidad temporal.....	25
Tabla 7: Riesgo: Rechazo de un artefacto intermedio .....	26
Tabla 8: Riesgo: Modificación de los requisitos del proyecto .....	26
Tabla 9: Especificaciones Lenovo Y50-70 .....	26
Tabla 10: Especificaciones Samsung NP-R522.....	26
Tabla 11: Especificaciones: Huawei G6 Ascend.....	26
Tabla 12: Especificaciones: Sony Xperia M.....	27
Tabla 13: Especificaciones: BQ Aquaris M5.....	27
Tabla 14: Especificaciones: Acer B1-830 .....	27
Tabla 15: Objetivo 1: Aprendizaje de los conceptos relacionados con los campos eléctricos.....	36
Tabla 16: Objetivo 2 Evaluación de los conocimientos obtenidos .....	36
Tabla 17: Objetivo 3: Evaluación de los conocimientos obtenidos .....	36
Tabla 18: Objetivo 4: Interacción entre varias cargas .....	36
Tabla 19: Actor 1: Alumno .....	37
Tabla 20. Actor 2: Alumno .....	37
Tabla 21: Actor 3: Profesor.....	37
Tabla 22: R. Funcional 1: Mostrar fundamentos teóricos .....	37
Tabla 23: R. Funcional 2: Realizar cuestionarios de autoevaluación.....	38
Tabla 24: R. Funcional 3: Realizar cuestionarios de evaluación .....	38
Tabla 25: R. Funcional 4: Tomar foto en el cuestionario de examen .....	38
Tabla 26: R. Funcional 5: Validar la identidad antes del cuestionario de examen.....	38
Tabla 27: R. Funcional 6: Envió de resultados de un cuestionario de examen .....	39
Tabla 28: R. Funcional 7: Descargar contenidos.....	39
Tabla 29: R. Funcional 8: Generar Teoría .....	39
Tabla 30: R. Funcional 9: Generar cuestionarios.....	39
Tabla 31: R. Funcional 10: Simulación .....	40
Tabla 32: R. No Funcional 1: Servidor de correo .....	40
Tabla 33: R. No Funcional 2: Almacenamiento de datos.....	40
Tabla 34: Caso de Uso 1: Consultar fundamentos teóricos.....	42
Tabla 35: Caso de Uso 2: Realizar cuestionarios de autoevaluación.....	42
Tabla 36: Caso de Uso 3: Validar frente a servidor .....	42
Tabla 37: Caso de Uso 4: Tomar fotografía .....	43
Tabla 38: Caso de Uso 5: Enviar resultados.....	43
Tabla 39: Caso de Uso 6: Descargar contenidos.....	43
Tabla 40: Caso de Uso 7: Realizar cuestionarios de examen.....	44
Tabla 41: Caso de Uso 8: Realizar Simulación .....	44
Tabla 42: Caso de Uso 9: Calcular potencial eléctrico .....	45
Tabla 43: Caso de Uso 10: Generar teoría.....	45
Tabla 44: Caso de Uso 11: Generar pregunta de cuestionario.....	45
Tabla 45: Caso de Uso 12: Borrar carga.....	46
Tabla 46: Caso de Uso 13: Editar carga .....	46
Tabla 47: Prueba del dominio de la aplicación 1.....	110

Tabla 48: Prueba del dominio de la aplicación 2.....	111
Tabla 49: Prueba del dominio de la aplicación 3.....	111
Tabla 50: Prueba del dominio de la aplicación 4.....	112
Tabla 51: Prueba del dominio de la aplicación 5.....	112
Tabla 52: Prueba del dominio de la aplicación 6.....	113
Tabla 53: Prueba del dominio de la aplicación 7.....	113
Tabla 55: Prueba del dominio de la aplicación 9.....	114
Tabla 56: Prueba del dominio de la aplicación 10.....	114
Tabla 57: Prueba del dominio de la aplicación 11.....	115
Tabla 58: Prueba del dominio de la aplicación 12.....	115
Tabla 59: Prueba del dominio de la aplicación 13.....	116
Tabla 60: Prueba del dominio de la aplicación 14.....	116
Tabla 62: Prueba del dominio de la aplicación 16.....	117
Tabla 63: Prueba del dominio de la aplicación 17.....	118
Tabla 64: Prueba del dominio de la aplicación 18.....	119
Tabla 65: Prueba del dominio de la aplicación 19.....	120
Tabla 66: Prueba del dominio de la aplicación 20.....	120
Tabla 67: Prueba GUI 1.....	121
Tabla 68: Prueba GUI 2.....	121
Tabla 69: Prueba GUI 3.....	121
Tabla 70: Prueba GUI 4.....	121
Tabla 71: Prueba GUI 5.....	121
Tabla 72: Prueba GUI 6.....	121
Tabla 73: Prueba GUI 7.....	122
Tabla 74: Prueba GUI 8.....	122
Tabla 75 Prueba GUI 9.....	122
Tabla 76: Prueba GUI 10.....	122
Tabla 77: Prueba GUI 11.....	122
Tabla 78: Prueba GUI 12.....	122

## Índice de imágenes

Ilustración 1 Cuota de mercado de sistemas operativos en España. Imagen tomada de ADSLZONE. Fuente Kantar WorldPanel .....	16
Ilustración 2 Distribución de versiones Android activas en Google Play Store a nivel mundial .....	17
Ilustración 3: Product Backlog .....	28
Ilustración 4: Planificación del primer sprint.....	29
Ilustración 5: Planificación del segundo sprint.....	30
Ilustración 6: Planificación del tercer sprint.....	31
Ilustración 7: Planificación del cuarto sprint .....	32
Ilustración 8: Planificación del quinto sprint .....	33
Ilustración 9: Diagrama de casos de uso del análisis.....	41
Ilustración 10: Modelo de Dominio.....	47
Ilustración 11: Consultar fundamentos teóricos.....	49
Ilustración 12: Validar frente al servidor.....	49
Ilustración 13: Realizar cuestionario de autoevaluación.....	50
Ilustración 14: Tomar fotografía.....	50
Ilustración 15: Enviar resultados.....	51
Ilustración 16: Descargar contenidos.....	51
Ilustración 17: Realizar simulación.....	52
Ilustración 18: Calcular potencial eléctrico.....	53
Ilustración 19: Editar carga.....	53
Ilustración 20: Borrar carga.....	54
Ilustración 21: Generar pregunta de cuestionario.....	54
Ilustración 22: Realizar cuestionario de examen.....	55
Ilustración 23: Generar archivo de teoría.....	56
Ilustración 24: Modelo de diseño del caso de uso “Consultar fundamentos teóricos” .....	58
Ilustración 25: Modelo de diseño del caso de uso “Realizar simulación” .....	59
Ilustración 26: Modelo de diseño de los casos de uso “Realizar cuestionarios de examen” y “Realizar cuestionarios de autoevaluación” .....	60
Ilustración 27: Modelo de diseño del caso de uso “Descargar contenidos” .....	61
Ilustración 28: Modelo de diseño del caso de uso “Generar cuestionario” .....	62
Ilustración 29: Modelo de diseño del caso de uso “Generar archivo de teoría” .....	63
Ilustración 30: Clase MainActivity .....	64
Ilustración 31: Clase DropboxConnection .....	64
Ilustración 32: Clase TeoriaActivity .....	65
Ilustración 33: TeoriaPageAdapter .....	65
Ilustración 34: Clase FragmentoTeoria.....	65
Ilustración 35: Clase PrevioCuestionario .....	66
Ilustración 36: Clase UtilesConexion .....	66
Ilustración 37: Clases ValidarUsuarioActivity e Hilo .....	66
Ilustración 38: Clase CuestionarioActivity .....	67
Ilustración 39: Clase CuestionarioPageAdapter .....	67
Ilustración 40: Clase GestorCorreo.....	68
Ilustración 41: Clase ProcesadorFoto .....	68
Ilustración 42: Clase FragmentoCuestionario .....	69
Ilustración 43: Clase GeneradorMathJax.....	70
Ilustración 44: Clase SimulacionActivity .....	70
Ilustración 45: Clase Carga .....	71

Ilustración 46: Clase HiloDibujo.....	71
Ilustración 47: Clase DibujaLineas .....	72
Ilustración 48: Clase Lineas .....	72
Ilustración 49: Clase Cuestionario (Aplicación complementaria).....	73
Ilustración 50: HiloPersistenciaCuestionario (Aplicación complementaria) .....	73
Ilustración 51: Clase Teoria (Aplicación complementaria).....	74
Ilustración 52: HiloPersistenciaTeoria (Aplicación complementaria) .....	74
Ilustración 53: Diagrama de Secuencia: Consultar fundamentos teóricos I .....	75
Ilustración 54: Diagrama de Secuencia: Consultar fundamentos teóricos II .....	76
Ilustración 55: Diagrama de secuencia: Realizar cuestionario de autoevaluación I .....	77
Ilustración 56: Diagrama de secuencia: Realizar cuestionario de autoevaluación II .....	78
Ilustración 57: Diagrama de secuencia: Realizar cuestionario de autoevaluación III .....	79
Ilustración 58: Diagrama de secuencia: Realizar simulación.....	80
Ilustración 59: Diagrama de secuencia: Dibuja Líneas .....	81
Ilustración 60: Diagrama de Secuencia: Calcular potencial.....	82
Ilustración 61: Diagrama de secuencia: Reiniciar.....	82
Ilustración 62: Borrar carga .....	83
Ilustración 63: Diagrama de Secuencia: Mostrar datos de una carga.....	83
Ilustración 64: Diagrama de Secuencia: Editar Carga.....	84
Ilustración 65: Diagrama de secuencia: Generar cuestionario I .....	85
Ilustración 66: Diagrama de secuencia: Generar cuestionario II .....	86
Ilustración 67: Diagrama de secuencia: Generar cuestionario III .....	87
Ilustración 68: Diagrama de secuencia: Generar archivo de teoría I .....	88
Ilustración 69: Diagrama de secuencia: Generar archivo de teoría II .....	89
Ilustración 70: Diagrama de secuencia: Generar archivo de teoría III .....	89
Ilustración 71: Diagrama de secuencia: Generar archivo de teoría IV.....	90
Ilustración 72: Diagrama de secuencia: Generar archivo de teoría V.....	91
Ilustración 73: Diagrama de secuencia: Realizar cuestionario de examen I .....	92
Ilustración 74: Diagrama de secuencia: Realizar cuestionario de examen II .....	93
Ilustración 75: Diagrama de secuencia: Realizar cuestionario de examen III .....	94
Ilustración 76: Diagrama de secuencia: Realizar cuestionario de examen IV.....	95
Ilustración 77: Diagrama de secuencia: Validar usuario .....	96
Ilustración 78: Diagrama de secuencia: Enviar credenciales.....	97
Ilustración 79: Diagrama de secuencia: Tomar fotografía .....	97
Ilustración 80: Envío de los Datos .....	98
Ilustración 81: Descargar contenidos I.....	99
Ilustración 82: Descargar contenidos II .....	100
Ilustración 83: Descargar contenidos III .....	101
Ilustración 84: Descargar contenidos IV.....	102
Ilustración 85: Habilitar orígenes desconocidos .....	132
Ilustración 86: Pantalla del instalador .....	132
Ilustración 87: Pantalla durante la instalación .....	133
Ilustración 88: Pantalla de fin de instalación.....	133
Ilustración 89: Aspecto de la pantalla principal con el menú desplegado .....	134
Ilustración 90: Pantalla de teoría .....	135
Ilustración 91: Pantalla de selección de cuestionario .....	135
Ilustración 92: Pregunta sin responder y respondida .....	136
Ilustración 93: Resumen del intento .....	136
Ilustración 94: Pantalla de validación de usuarios .....	137
Ilustración 95: Pantalla de corrección .....	138

Ilustración 96: Pantalla de fin de examen .....	138
Ilustración 97: Opciones en un punto de la simulación .....	139
Ilustración 98: Diálogo de introducción de una carga.....	140
Ilustración 99: Opciones de una carga .....	140
Ilustración 100: Opción de reinicio de la simulación.....	141
Ilustración 101: Menú principal de la aplicación complementaria .....	141
Ilustración 102: Pantalla de creación de cuestionarios.....	142
Ilustración 103: Opciones del menú “archivo” en cuestionarios .....	142
Ilustración 104: Pantalla de creación de páginas de teoría.....	143

# Capítulo I: Introducción.

## Visión general. Contextualización del Proyecto.

La utilización de las tecnologías móviles ha experimentado un crecimiento notable en los últimos años. Es difícil pensar que un joven hoy en día no dispone de un Smartphone y más teniendo en cuenta nuestro ámbito académico, donde la mayoría (por no generalizar) tenemos interés sobre las nuevas tecnologías y hemos decidido encaminar nuestro futuro a su estudio o aprovechamiento. Este es el motivo fundamental para realizar un proyecto como este, utilizamos para afianzar conocimientos un aparato que siempre llevamos encima y conocemos perfectamente. Además de esto, una aplicación móvil es algo muy cómodo, en cualquier rato libre podemos sacar nuestro dispositivo del bolso o mochila y ponernos a trabajar, lo que supone poder aprovechar cualquier momento.

Ahora bien, de todas estas tecnologías móviles, ¿Cuál elegimos para desarrollar la aplicación? Pues bien, si hay una tecnología que puede abanderar el crecimiento de las tecnologías móviles, desde luego esa es Android, por lo menos en los que a nuestro país se refiere (aunque es extensible a nivel mundial).

Cuota de mercado de sistemas operativos en España

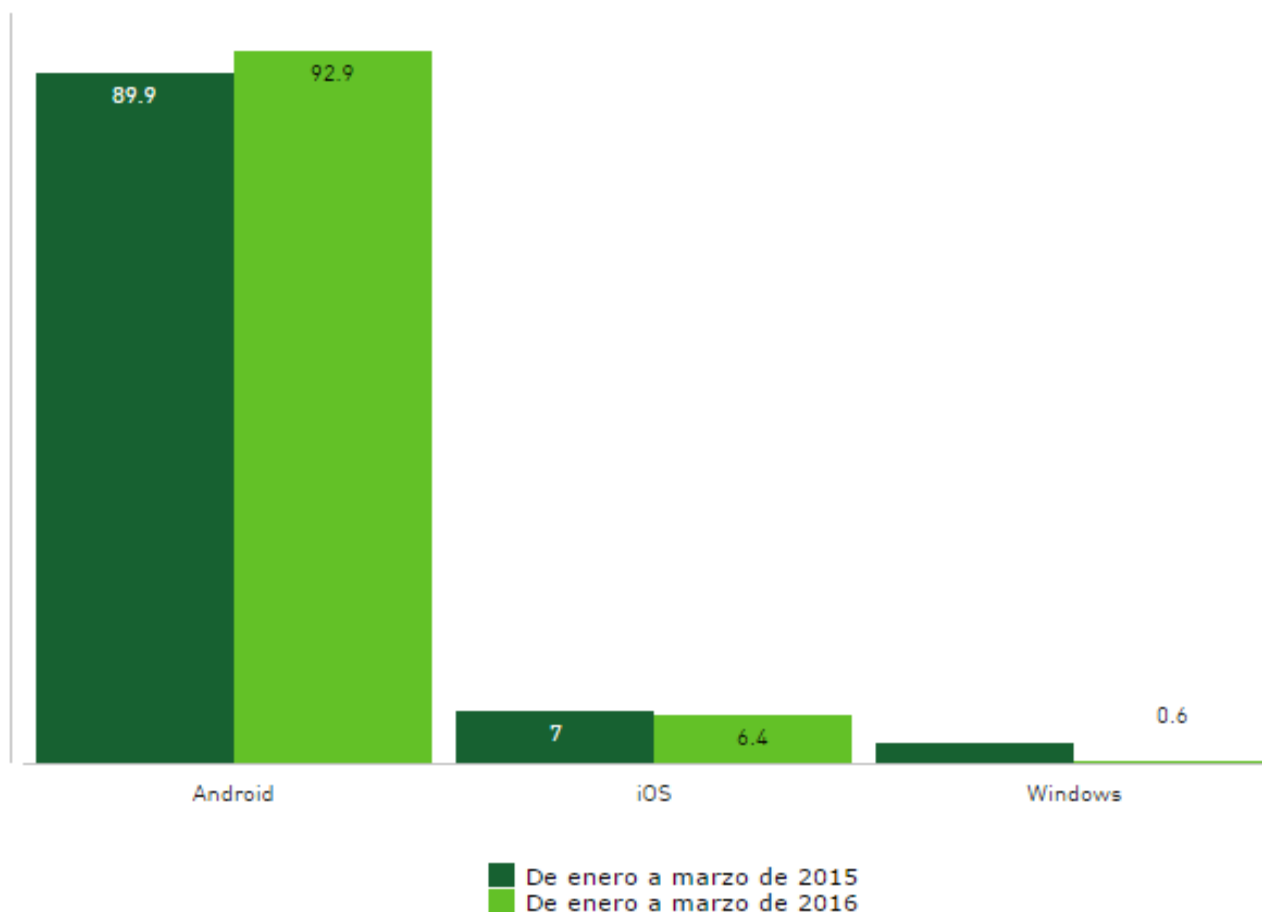


Ilustración 1 Cuota de mercado de sistemas operativos en España. Imagen tomada de ADSLZONE. Fuente Kantar WorldPanel

Como nos muestra la Ilustración 1, Android domina en nuestro país la venta de Smartphones. Si además lo comparamos con el año anterior, vemos que el número de ventas en el mismo trimestre se ha incrementado en tres puntos. Viendo estos datos, parece más que justificada la decisión de utilizar Android como plataforma sobre la que ejecutar nuestra aplicación móvil, si queremos garantizar que esta pueda ser utilizada por el mayor número de usuarios.



En cuanto a la selección de la versión mínima, debemos tener en cuenta la distribución de las versiones Android. Cuando se comenzó a trabajar en este proyecto en febrero de 2016, utilizar la API 18 como mínima permitía poder estar presentes en más de las tres cuartas partes de los teléfonos activos en Google Play Store.

2.3 Gingerbread	10	97,4%
4.0 Ice Cream Sandwich	15	95,2%
4.1 Jelly Bean	16	87,4%
4.2 Jelly Bean	17	76,9%
4.3 Jelly Bean	18	73,9%
4.4 KitKat	19	40,5%
5.0 Lollipop	21	24,1%
5.1 Lollipop	22	4,7%
6.0 Marshmallow	23	

*Ilustración 2 Distribución de versiones Android activas en Google Play Store a nivel mundial*

Los datos anteriores son ofrecidos por Android Studio en día 26 de mayo de 2016. Pese a ser un momento posterior, podemos ver que la cuota de terminales activos en Play Store desde la API 18 en adelante aún representa más de tres cuartas partes.

## Objetivos.

El objetivo fundamental de esta aplicación es servir como ayuda a la docencia de la asignatura de Física. Esta aplicación debe permitir que el alumno pueda obtener un compendio de la teoría sobre Campos Eléctricos que se imparte en la asignatura de Física, que pueda evaluar la asimilación de esta teoría y que pueda ver mediante el empleo de un simulador, cómo cargas puntuales interactúan entre sí. Para ello, la aplicación pinta en tiempo real las líneas de campo que esas cargas generan y permite conocer el potencial eléctrico en cualquier punto de la pantalla.

Aparte de los objetivos anteriores, existen otros objetivos derivados de ellos que la aplicación debe satisfacer. Estos objetivos son los siguientes:

- La aplicación debe ser capaz de actualizar el resumen de la teoría que contiene mediante una descarga del contenido de una nube.
- De la misma manera, la aplicación debe poder actualizar las preguntas de los cuestionarios.
- La aplicación debe contener un modo de evaluación que permita conocer al profesor los resultados de los alumnos mediante el envío de un correo electrónico.
- La aplicación debe tomar una foto del alumno que realiza un cuestionario en un momento aleatorio, si el modo elegido es el de cuestionario con posterior información al profesor, en adelante modo examen, para verificar su identidad.
- La aplicación debe validar que el alumno que realiza el cuestionario en modo examen es un alumno matriculado en la Universidad de Valladolid (en adelante Uva). Para ello, antes de acceder al cuestionario en modo examen, se realizará una verificación de identidad frente al SMTP de la de esta universidad (Cartero).

Debemos tener en cuenta que aparte de la aplicación móvil principal existe una secundaria que debe satisfacer los siguientes objetivos:

- Producir los contenidos teóricos que se muestran en la aplicación.
- Producir las preguntas que usa la aplicación para realizar.

Fundamento teórico.

Campo Eléctrico.

Según la Ley de Coulomb:

- $\vec{F}_{12} = k \frac{q_1 \cdot q_2}{r_{12}^2} \vec{u}_r = k \frac{q_1 \cdot q_2}{r_{12}^3} \vec{r}_{12}$
- $[q] = C (S.I.)$
- $k = \frac{1}{4\pi\epsilon_0}$
- $\epsilon_0 = 8,85 \cdot 10^{-12} C^2 N^{-1} m^{-2}$

Donde  $\epsilon_0$  es la permitividad eléctrica del vacío.

Los problemas de interpretación de las fuerzas a distancia hacen aparecer el concepto de campo.

Campo eléctrico  $\vec{E}(\vec{r}, t)$ :

- Una carga  $q$  altera el espacio que la rodea.
- Otra carga próxima  $q'$  nota esa alteración.
- La fuerza eléctrica es una medida de la interacción.
- El campo se define como la fuerza por unidad de carga

$$\vec{E} = \frac{\vec{F}}{q'} \text{ (N/C).}$$

Para una distribución continua de cargas, aplicar el principio de superposición:

1. En distribuciones discretas

$$-\vec{E}(r)_{total} = \sum_1^n k \frac{q_i}{r_i^2} \vec{u}_{r_i}$$

2. En distribuciones continuas

$$\vec{E}(r)_{total} = \int_{distribucion} d\vec{E}$$

$$d\vec{E} = k \frac{dq}{r^2} u_r$$

El campo se representa mediante líneas de fuerza. Estas líneas deben cumplir:

- Son tangentes a  $\vec{E}$  en todos los puntos.
- Su sentido es el de  $\vec{E}$
- No pueden cortarse ( $\vec{E}$  es univaluado).
- Se representan en función de la intensidad de  $\vec{E}$  en cada punto.

Flujo Eléctrico.

Si  $\vec{E}$  es un campo eléctrico y  $\vec{A} = A \vec{u}_a$  es una superficie plana, con área A y orientación dada por  $\vec{u}_a$ , el flujo de  $\vec{E}$  a través de  $\vec{A}$  es:

$$\phi = \vec{E} \cdot \vec{A}$$

En un caso general la superficie no es constante ( $\vec{A}$  no es constante), o el campo eléctrico no es uniforme, o ambos no forman un ángulo fijo en todos los puntos de la superficie. En estos casos el flujo se obtiene como:

$$\phi = \int_{superficie} \vec{E} \cdot d\vec{S}$$

$\phi = \vec{E} \cdot \vec{A}$  es proporcional al número de líneas de campo que atraviesan la superficie  $\vec{A}$ . Para una superficie cerrada, las líneas que salen de ella dan lugar a un flujo positivo y las que entran a uno negativo. Entonces:

- $\phi > 0$  si salen más líneas que las que entran.
- $\phi < 0$  si entran más líneas que las que salen.
- $\phi = 0$  si sale igual número de líneas que las que entran.

Según el **Teorema de Gauss**, Para una carga puntual q encerrada por una superficie esférica centrada en ella se tiene:

$$\phi = \frac{q}{\epsilon_0}$$

y el resultado es idéntico para cualquier superficie que encierre a la carga.

Para una distribución continua de cargas, aplicar el principio de superposición:

1. En distribuciones discretas

$$\phi = \frac{\sum q_{encerradas}}{\epsilon_0}$$

2. En distribuciones continuas

$$\phi = \int_{superficie} \vec{E} \cdot d\vec{S} = \frac{\sum q_{encerradas}}{\epsilon_0} = \frac{1}{\epsilon_0} \int_{distribucion} dq$$

El teorema de Gauss permite obtener el campo eléctrico cuando la simetría de cargas permite imaginar la propia simetría del campo.

Potencial Eléctrico.

Como el campo electrostático  $\vec{E}$  es conservativo, se puede definir, el potencial eléctrico asociado a  $\vec{E}$  como

$$\int_A^B \vec{E} \cdot d\vec{l} = V(A) - V(B)$$

$V(A) - V(B)$  representa el trabajo que realiza el campo eléctrico para desplazar la unidad de carga positiva desde A a B.

Esta definición sólo permite calcular diferencias de potencial entre dos puntos, no el potencial exacto en un punto dado. El origen de potenciales se puede definir arbitrariamente dependiendo de la distribución de cargas.

El potencial eléctrico se representa mediante líneas equipotenciales, perpendiculares a las líneas de campo. Las líneas equipotenciales se representan manteniendo diferencias constantes de V entre líneas consecutivas.

Energía Potencial.

La energía potencial de una carga en un campo  $\vec{E}$  es el trabajo que realiza el campo para trasladar esa carga desde el punto donde se encuentra hasta otro de potencial nulo. Entonces:

$$E_p(r) = qV(r)$$

y el trabajo para desplazar una carga desde un punto A hasta uno B sería entonces

$$W = -\Delta E_p = q(V(A) - V(B))$$

## Contexto de Desarrollo.

### Desarrollo de la aplicación principal.

Como dije en la “Contextualización del proyecto”, la aplicación principal va a ser desarrollada en Android y tendrá como API mínima la 18. La herramienta que se ha utilizado para escribir la aplicación es el entorno Android Studio en su versión 2.

Esta herramienta basada en IntelliJ ya lleva incorporada el Android SDK y permite acoplar distintos emuladores mediante su gestor de estos llamado Android Virtual Devices. Otra de sus ventajas es que permite conocer el estado del sistema de ficheros tanto de un emulador como de un teléfono e interactuar con los primeros, por ejemplo, simular una llamada o cambiar una ubicación. Para esto se utiliza el Android Device Monitor.

Para probar los avances de la aplicación, aparte de los emuladores, se ha dispuesto de distintos terminales físicos con distintas versiones de Android y tamaño de pantalla.

### Desarrollo de la aplicación complementaria.

Para desarrollar la aplicación complementaria se ha utilizado el lenguaje de programación Java. La versión elegida ha sido la 1.8 (comercialmente Java 8, que es la misma sobre la que corría el Android Studio). El hecho de elegir Java viene dado por la facilidad para generar software multiplataforma. Para ejecutar el programa solo es necesaria una Máquina Virtual Java 8 (Jre 8) que está disponible para Windows, Linux y Mac OS.

Con el objetivo de desarrollar más cómodamente, se ha utilizado el entorno de desarrollo integrado NetBeans en su versión 8.1. La razón para elegir esta herramienta es la experiencia ya adquirida con ella y el autocompletado de código, no solo sugerencias de código, sino también ciertas estructuras, por ejemplo, en este caso, el código de la interfaz gráfica.

### Redacción de la memoria.

Para desarrollar la memoria se ha utilizado el procesador de textos Microsoft Office Word en su versión 2016. La herramienta PowerPoint de esta familia de software ha sido la utilizada para crear la presentación de la exposición.

Para llevar a cabo la gestión del proyecto y obtener los informes que se presentaran en esta memoria, se ha utilizado Microsoft Office Project en su versión de 2013, debido a la experiencia previa que se tenía con esta herramienta.

Para generar todo el modelado (UML) que ha conllevado el proyecto se ha utilizado la herramienta Astah Professional en su versión 7.



# Capítulo II: Planificación.

La realización de este proyecto va a ser llevada a cabo por una persona, Antonio Pesquera Rodríguez, quien realizará todos los roles del equipo de desarrollo. El tutor del proyecto, Manuel Ángel González Delgado, representará el papel de cliente, esto es, indicará los requisitos de los sistemas a construir y decidirá si el trabajo realizado satisface estos requisitos. Además de esto, también resolverá las dudas que puedan surgir al equipo de trabajo durante el desarrollo.

La colaboración del cliente con el equipo de desarrollo, la posibilidad de modificar los requisitos durante la construcción de las soluciones y la independencia de los requisitos entre sí, lo que nos permite ir generando versiones ejecutables según se van satisfaciendo, hacen que la metodología de planificación del proceso de desarrollo sea una metodología ágil. Para realizar este proyecto, he adaptado Scrum al equipo de desarrollo (Un equipo de una persona).

## Roles y responsabilidades.

Roles relativos al equipo del proyecto.

<b>Rol</b>	<b>Responsabilidades</b>	<b>Persona encargada</b>
Analista	Analiza los requisitos de la aplicación, establece el modelo de dominio y los casos de uso.	Antonio Pesquera Rodríguez
Diseñador	Diseña el sistema a construir. Esto es, genera el modelo de diseño y realiza los casos de uso teniendo en cuenta la tecnología a emplear.	Antonio Pesquera Rodríguez
Diseñador de interfaces	Diseña la interfaz con la que interactuara el usuario.	Antonio Pesquera Rodríguez
Programador	Codifica los diseños realizados por el Diseñador y el Diseñador de interfaces. Es por tanto quien construye la aplicación.	Antonio Pesquera Rodríguez
Probador	Prueba que los desarrollos obtenidos no contienen errores y satisfacen los requisitos.	Antonio Pesquera Rodríguez

*Tabla 1: Roles relativos al equipo del proyecto*

Roles relativos a stakeholders.

<b>Rol</b>	<b>Responsabilidades</b>	<b>Persona encargada</b>
Dueño del producto	Establece los requisitos de la aplicación y aprueba los artefactos intermedios.	M. Ángel González Delgado

*Tabla 2: Roles relativos a interfaces externas*

Pese a que en Scrum las fechas de entrega de los artefactos y la priorización de requisitos las hace el propietario del producto, en este proyecto lo primero será de común acuerdo entre el propietario y el equipo, y lo segundo responsabilidad única del equipo de desarrollo.



## Definición de Riesgos.

A continuación, se definen los riesgos que afectan al desarrollo de este proyecto, algunos como el desconocimiento de la tecnología van viendo mermada su posibilidad de aparición según avanza el proyecto, otros como la pérdida de elementos del desarrollo pueden ocurrir en cualquier momento.

<b>Riesgo 01</b>	<b>Planificación incorrecta</b>
<b>Descripción</b>	Se planifica un tiempo para el desarrollo de las tareas inferior al necesario.
<b>Efecto</b>	La planificación no se cumple, pues se alarga la realización de alguna(s) tarea(s)
<b>Frecuencia</b>	Alta.
<b>Gravedad</b>	Alta.
<b>Detección</b>	Alta.
<b>Acción correctora</b>	Realizar la planificación atendiendo a la aparición de riesgos.
<b>Plan de contingencia</b>	Acelerar el ritmo de trabajo para poder volver a ceñirse a la planificación.

*Tabla 3: Riesgo: Desconocimiento de las tecnologías a utilizar*

<b>Riesgo 02</b>	<b>Pérdida total o parcial de los datos</b>
<b>Descripción</b>	Perdida de parte del material desarrollado para satisfacer los requisitos.
<b>Efecto</b>	Retrasa las tareas, pues se requiere tiempo para volver a desarrollarlo.
<b>Frecuencia</b>	Baja. Debido a las réplicas.
<b>Gravedad</b>	Alta.
<b>Detección</b>	Alta.
<b>Acción correctora</b>	Replicar el material generado diariamente utilizando nubes de almacenamiento.
<b>Plan de contingencia</b>	Acelerar el ritmo de trabajo para poder entregar artefactos a tiempo.

*Tabla 4: Riesgo: Pérdida total o parcial de los datos*

<b>Riesgo 03</b>	<b>Avería de uno de los dispositivos utilizados para el desarrollo</b>
<b>Descripción</b>	Fundamentalmente avería del ordenador de desarrollo o el teléfono de depuración.
<b>Efecto</b>	Retrasa las tareas, pues se requiere tiempo para repararlo o reemplazarlo.
<b>Frecuencia</b>	No calculable.
<b>Gravedad</b>	Alta.
<b>Detección</b>	Alta.
<b>Acción correctora</b>	Replicar el material generado diariamente utilizando nubes de almacenamiento, lo que nos permite poder instalarlo en un nuevo dispositivo (o en el actual reparado).
<b>Plan de contingencia</b>	Disponer de varios dispositivos del mismo tipo y acelerar el ritmo de trabajo para poder entregar artefactos a tiempo si es preciso.

*Tabla 5: Riesgo: Avería de uno de los dispositivos utilizados para el desarrollo*

<b>Riesgo 04</b>	<b>Incapacidad temporal</b>
<b>Descripción</b>	Se da una circunstancia que impide el desarrollo normal de la planificación.
<b>Efecto</b>	Retrasa la planificación.
<b>Frecuencia</b>	No calculable.
<b>Gravedad</b>	Alta.
<b>Detección</b>	Alta.
<b>Acción correctora</b>	Realizar la planificación atendiendo a la posibilidad de aparición de estas circunstancias.
<b>Plan de contingencia</b>	Acelerar el ritmo de trabajo para volver a poder ceñirse a la planificación.

*Tabla 6: Riesgo: Incapacidad temporal*

<b>Riesgo 05</b>	<b>Rechazo de un artefacto intermedio</b>
<b>Descripción</b>	El dueño del producto muestra su inconformidad con el artefacto presentado.
<b>Efecto</b>	Requiere tiempo para readaptar el artefacto. Retrasa la planificación.
<b>Frecuencia</b>	Baja.
<b>Gravedad</b>	Alta.
<b>Detección</b>	Alta.
<b>Acción correctora</b>	Contacto frecuente con el dueño del producto, lo que permite aclarar lo que se espera de cada entregable intermedio en caso de duda
<b>Plan de contingencia</b>	Acelerar el ritmo de trabajo para volver a poder ceñirse a la planificación.

*Tabla 7: Riesgo: Rechazo de un artefacto intermedio*

<b>Riesgo 06</b>	<b>Modificación de los requisitos del proyecto</b>
<b>Descripción</b>	El dueño del producto modifica los requisitos de la aplicación.
<b>Efecto</b>	Requiere modificar el análisis y el diseño del artefacto del sprint correspondiente.
<b>Frecuencia</b>	Alta.
<b>Gravedad</b>	Baja. Las metodologías ágiles permiten adaptarse a cambios de requisitos.
<b>Detección</b>	Alta.
<b>Acción correctora</b>	Contacto frecuente con el dueño del producto, lo que permite conocer lo antes posible los cambios.
<b>Plan de contingencia</b>	Acelerar el ritmo de trabajo para volver a poder ceñirse a la planificación.

*Tabla 8: Riesgo: Modificación de los requisitos del proyecto*

Recursos utilizados. Estimación de costes.

<b>Dispositivo</b>	<b>Lenovo Y50-70</b>
<b>Tipo de dispositivo</b>	Computadora
<b>Sistema Operativo</b>	Microsoft Windows 10 Pro
<b>Procesador</b>	Intel i7 4710-HQ
<b>Memoria</b>	8 GB DDR3 1600MHz
<b>Almacenamiento</b>	1 TB
<b>Coste</b>	0 € (Adquirido antes del comienzo del proyecto)

*Tabla 9: Especificaciones Lenovo Y50-70*

<b>Dispositivo</b>	<b>Samsung NP-R522</b>
<b>Tipo de dispositivo</b>	Computadora
<b>Sistema Operativo</b>	Microsoft Windows 10 Pro
<b>Procesador</b>	Intel dual core T4200 2,0 GHz
<b>Memoria</b>	4 GB DDR2 1600MHz
<b>Almacenamiento</b>	512 GB
<b>Coste</b>	0 € (Adquirido antes del comienzo del proyecto)

*Tabla 10: Especificaciones Samsung NP-R522*

<b>Dispositivo</b>	<b>Huawei G6 Ascend</b>
<b>Tipo de dispositivo</b>	Teléfono
<b>Sistema Operativo</b>	Android 4.3 (API 18)
<b>Procesador</b>	Quad Core 1,2GHz
<b>Memoria</b>	1 GB
<b>Almacenamiento</b>	8 GB
<b>Coste</b>	0 € (Adquirido antes del comienzo del proyecto)

*Tabla 11: Especificaciones: Huawei G6 Ascend*

<b>Dispositivo</b>	<b>Sony Xperia M</b>
<b>Tipo de dispositivo</b>	Teléfono
<b>Sistema Operativo</b>	Android 4.3 (API 18)
<b>Procesador</b>	Procesador dual core Qualcomm de 1 GHz
<b>Memoria</b>	1 GB
<b>Almacenamiento</b>	4 GB
<b>Coste</b>	0 € (Adquirido antes del comienzo del proyecto)

*Tabla 12: Especificaciones: Sony Xperia M*

<b>Dispositivo</b>	<b>BQ Aquaris M5</b>
<b>Tipo de dispositivo</b>	Teléfono
<b>Sistema Operativo</b>	Android 5.1.1 (API 22)
<b>Procesador</b>	Qualcomm® Snapdragon™ 615 Octa Core A53 hasta 1,5 GHz
<b>Memoria</b>	2 GB
<b>Almacenamiento</b>	16 GB
<b>Coste</b>	0 € (Adquirido antes del comienzo del proyecto)

*Tabla 13: Especificaciones: BQ Aquaris M5*

<b>Dispositivo</b>	<b>Acer B1-830</b>
<b>Tipo de dispositivo</b>	Tablet
<b>Sistema Operativo</b>	Android 5.1 (API 22)
<b>Procesador</b>	MediaTel MT8151V Octa-core 1,70 GHz
<b>Memoria</b>	1 GB
<b>Almacenamiento</b>	32 GB
<b>Coste</b>	159,99 €

*Tabla 14: Especificaciones: Acer B1-830*

## Product Backlog.

A continuación, se muestran los requisitos del proyecto ordenados por prioridad. Como dijimos al comienzo de esta sección, el propietario del producto ha permitido que sea el equipo quien establezca la prioridad.

Identificador	Nombre	Prioridad	Notas
1	Generar archivos de teoría	1	Herramienta de escritorio que permita generar archivos de teoría.
2	Generar cuestionarios	1	Herramienta de escritorio que permita generar cuestionarios.
3	Realizar cuestionarios de autoevaluación.	2	Tiene la complejidad de seleccionar la forma de representar el cuestionario, como se corrige, como se muestra la corrección.
4	Realizar cuestionarios de examen.	2	Adaptar lo anterior a la especificación de examen.
5	Validar la identidad antes del cuestionario de examen	2	Añadir la validación de usuarios ante el servidor de correo de la Uva en el cuestionario de examen.
6	Envío de resultados de un cuestionario de examen	2	Añadir el envío de resultados y la foto en los cuestionarios de examen.
7	Tomar foto.	2	Añadir la función de toma de foto al cuestionario de examen.
8	Mostrar fundamentos teóricos	3	La complejidad estará en mostrar y acceder a las páginas de teoría. La activity será una TabActivity como la de los cuestionarios.
9	Descargar contenidos	3	Descarga de los cuestionarios y la teoría del repositorio
10	Realizar simulación	4	Simulación de campos eléctricos generados por cargas puntuales.
11	Generar informe final	5	Memoria final a entregar. Pese a que no es una parte de la aplicación, si lo es del proyecto y por tanto debe ser planificada.

Ilustración 3: Product Backlog

## Primer Sprint.

Requisitos a realizar. Sprint Backlog.

- Generar archivos de teoría.
- Generar cuestionarios.

Calendario del sprint.

DIAGRAMA DE GANTT

	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
1	▲ <b>Sprint 1</b>	<b>24 días</b>	<b>vie 12/02/16</b>	<b>dom 06/03/16</b>		
2	▲ <b>Planificacion del Sprint</b>	<b>1 día</b>	<b>vie 12/02/16</b>	<b>vie 12/02/16</b>		
3	Planificacion del Sprint	1 día	vie 12/02/16	vie 12/02/16		Antonio Pesquera Rodríguez
4	▲ <b>Analisis</b>	<b>6 días</b>	<b>sáb 13/02/16</b>	<b>jue 18/02/16</b>		
5	Generacion y descripcion de los casos de uso	2 días	sáb 13/02/16	dom 14/02/16	3	Antonio Pesquera Rodríguez
6	Modelado de dominio	2 días	lun 15/02/16	mar 16/02/16	5	Antonio Pesquera Rodríguez
7	Diagrama de secuencia de analisis	2 días	mié 17/02/16	jue 18/02/16	6	Antonio Pesquera Rodríguez
8	▲ <b>Diseño</b>	<b>8 días</b>	<b>vie 19/02/16</b>	<b>vie 26/02/16</b>		
9	Modelo de diseño	2 días	vie 19/02/16	sáb 20/02/16	7	Antonio Pesquera Rodríguez
10	Diseño de la interfaz grafica	1 día	dom 21/02/16	dom 21/02/16	9	Antonio Pesquera Rodríguez
11	Diagramas de secuencia	5 días	lun 22/02/16	vie 26/02/16	10	Antonio Pesquera Rodríguez
12	▲ <b>Implementacion</b>	<b>6 días</b>	<b>sáb 27/02/16</b>	<b>jue 03/03/16</b>		
13	Desarrollo de la aplicación	6 días	sáb 27/02/16	jue 03/03/16	11	Antonio Pesquera Rodríguez
14	▲ <b>Pruebas</b>	<b>3 días</b>	<b>vie 04/03/16</b>	<b>dom 06/03/16</b>		
15	Pruebas del dominio de la	2 días	vie 04/03/16	sáb 05/03/16	13	Antonio Pesquera Rodríguez
16	Pruebas de la interfaz	1 día	dom 06/03/16	dom 06/03/16	15	Antonio Pesquera Rodríguez

Ilustración 4: Planificación del primer sprint

Segundo sprint.

Requisitos a realizar. Sprint Backlog.

- Realizar cuestionarios de autoevaluación.
- Realizar cuestionarios de examen.
- Validar la identidad antes del cuestionario de examen.
- Envío de resultados de un cuestionario de examen.
- Tomar foto.

Calendario del sprint.

DIAGRAMA DE GANTT

	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
1	<b>▲ Sprint 2</b>	<b>35 días</b>	<b>lun 07/03/16</b>	<b>dom 10/04/16</b>		
2	<b>▲ Planificación del Sprint</b>	<b>1 día</b>	<b>lun 07/03/16</b>	<b>lun 07/03/16</b>		
3	Planificación del Sprint	1 día	lun 07/03/16	lun 07/03/16		Antonio Pesquera Rodríguez
4	<b>▲ Analisis</b>	<b>7 días</b>	<b>mar 08/03/16</b>	<b>lun 14/03/16</b>		
5	Generación y descripción de los casos de uso	2 días	mar 08/03/16	mié 09/03/16	3	Antonio Pesquera Rodríguez
6	Modelado de dominio	2 días	jue 10/03/16	vie 11/03/16	5	Antonio Pesquera Rodríguez
7	Diagrama de secuencia de análisis	3 días	sáb 12/03/16	lun 14/03/16	6	Antonio Pesquera Rodríguez
8	<b>▲ Diseño</b>	<b>13 días</b>	<b>mar 15/03/16</b>	<b>dom 27/03/16</b>		
9	Modelo de diseño	5 días	mar 15/03/16	sáb 19/03/16	7	Antonio Pesquera Rodríguez
10	Diseño de la interfaz gráfica	2 días	dom 20/03/16	lun 21/03/16	9	Antonio Pesquera Rodríguez
11	Diagramas de secuencia	6 días	mar 22/03/16	dom 27/03/16	10	Antonio Pesquera Rodríguez
12	<b>▲ Implementación</b>	<b>11 días</b>	<b>lun 28/03/16</b>	<b>jue 07/04/16</b>		
13	Desarrollo de la aplicación	11 días	lun 28/03/16	jue 07/04/16	11	Antonio Pesquera Rodríguez
14	<b>▲ Pruebas</b>	<b>3 días</b>	<b>vie 08/04/16</b>	<b>dom 10/04/16</b>		
15	Pruebas del dominio de la	2 días	vie 08/04/16	sáb 09/04/16	13	Antonio Pesquera Rodríguez
16	Pruebas de la interfaz	1 día	dom 10/04/16	dom 10/04/16	15	Antonio Pesquera Rodríguez

Ilustración 5: Planificación del segundo sprint

Tercer sprint.

Requisitos a realizar. Sprint Backlog.

- Mostrar fundamentos teóricos.
- Descargar contenidos.

Calendario del sprint.

DIAGRAMA DE GANTT

	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
1	▸ <b>Sprint 3</b>	<b>29 días</b>	<b>lun 11/04/16</b>	<b>lun 09/05/16</b>		
2	▸ <b>Planificación del Sprint</b>	<b>1 día</b>	<b>lun 11/04/16</b>	<b>lun 11/04/16</b>		
3	Planificación del Sprint	1 día	lun 11/04/16	lun 11/04/16		Antonio Pesquera Rodríguez
4	▸ <b>Análisis</b>	<b>7 días</b>	<b>mar 12/04/16</b>	<b>lun 18/04/16</b>		
5	Generación y descripción de los casos de uso	2 días	mar 12/04/16	mié 13/04/16	3	Antonio Pesquera Rodríguez
6	Modelado de dominio	3 días	jue 14/04/16	sáb 16/04/16	5	Antonio Pesquera Rodríguez
7	Diagrama de secuencia de análisis	2 días	dom 17/04/16	lun 18/04/16	6	Antonio Pesquera Rodríguez
8	▸ <b>Diseño</b>	<b>10 días</b>	<b>mar 19/04/16</b>	<b>jue 28/04/16</b>		
9	Modelo de diseño	3 días	mar 19/04/16	jue 21/04/16	7	Antonio Pesquera Rodríguez
10	Diseño de la interfaz gráfica	3 días	vie 22/04/16	dom 24/04/16	9	Antonio Pesquera Rodríguez
11	Diagramas de secuencia	4 días	lun 25/04/16	jue 28/04/16	10	Antonio Pesquera Rodríguez
12	▸ <b>Implementación</b>	<b>8 días</b>	<b>vie 29/04/16</b>	<b>vie 06/05/16</b>		
13	Desarrollo de la aplicación	8 días	vie 29/04/16	vie 06/05/16	11	Antonio Pesquera Rodríguez
14	▸ <b>Pruebas</b>	<b>3 días</b>	<b>sáb 07/05/16</b>	<b>lun 09/05/16</b>		
15	Pruebas del dominio de la	2 días	sáb 07/05/16	dom 08/05/16	13	Antonio Pesquera Rodríguez
16	Pruebas de la interfaz	1 día	lun 09/05/16	lun 09/05/16	15	Antonio Pesquera Rodríguez

Ilustración 6: Planificación del tercer sprint

Cuarto sprint.

Requisitos a realizar. Sprint Backlog.

- Realizar simulación

Calendario del sprint.

DIAGRAMA DE GANTT

	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos	Agregar nueva
1	<b> Sprint 4</b>	<b>34 días</b>	<b>mar 10/05/16</b>	<b>dom 12/06/16</b>			
2	<b> Planificación del Sprint</b>	<b>1 día</b>	<b>mar 10/05/16</b>	<b>mar 10/05/16</b>			
3	Planificación del Sprint	1 día	mar 10/05/16	mar 10/05/16		Antonio Pesquera Rodríguez	
4	<b> Análisis</b>	<b>7 días</b>	<b>mié 11/05/16</b>	<b>mar 17/05/16</b>			
5	Generación y descripción de los casos de uso	2 días	mié 11/05/16	jue 12/05/16	3	Antonio Pesquera Rodríguez	
6	Modelado de dominio	2 días	vie 13/05/16	sáb 14/05/16	5	Antonio Pesquera Rodríguez	
7	Diagrama de secuencia de análisis	3 días	dom 15/05/16	mar 17/05/16	6	Antonio Pesquera Rodríguez	
8	<b> Diseño</b>	<b>9 días</b>	<b>mié 18/05/16</b>	<b>jue 26/05/16</b>			
9	Modelo de diseño	3 días	mié 18/05/16	vie 20/05/16	7	Antonio Pesquera Rodríguez	
10	Diseño de la interfaz gráfica	2 días	sáb 21/05/16	dom 22/05/16	9	Antonio Pesquera Rodríguez	
11	Diagramas de secuencia	4 días	lun 23/05/16	jue 26/05/16	10	Antonio Pesquera Rodríguez	
12	<b> Implementación</b>	<b>12 días</b>	<b>vie 27/05/16</b>	<b>mar 07/06/16</b>			
13	Desarrollo de la aplicación	12 días	vie 27/05/16	mar 07/06/16	11	Antonio Pesquera Rodríguez	
14	<b> Pruebas</b>	<b>5 días</b>	<b>mié 08/06/16</b>	<b>dom 12/06/16</b>			
15	Pruebas del dominio de la	3 días	mié 08/06/16	vie 10/06/16	13	Antonio Pesquera Rodríguez	
16	Pruebas de la interfaz	2 días	sáb 11/06/16	dom 12/06/16	15	Antonio Pesquera Rodríguez	

Ilustración 7: Planificación del cuarto sprint

Quinto sprint.

Requisitos a realizar. Sprint Backlog.

- Generar informe final



## Calendario del sprint.

DIAGRAMA DE GANTT


























		Modo de	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
1			<b> Sprint 4</b>	<b>25 días</b>	<b>dom 12/06/16</b>	<b>vie 08/07/16</b>		
2			<b> Planificacion del Sprint</b>	<b>1 día</b>	<b>dom 12/06/16</b>	<b>dom 12/06/16</b>		
3			Planificacion del Sprint	1 día	dom 12/06/16	dom 12/06/16		Antonio Pesquera Rodríguez
4			<b> Introduccion</b>	<b>4 días</b>	<b>mié 15/06/16</b>	<b>sáb 18/06/16</b>		
5			Toma de datos	1 día	mié 15/06/16	mié 15/06/16	3	Antonio Pesquera Rodríguez
6			Redaccion	3 días	jue 16/06/16	sáb 18/06/16	5	Antonio Pesquera Rodríguez
7			<b> Planificacion</b>	<b>2 días</b>	<b>dom 19/06/16</b>	<b>lun 20/06/16</b>		
8			Redaccion	2 días	dom 19/06/16	lun 20/06/16	6	Antonio Pesquera Rodríguez
9			<b> Analisis</b>	<b>4 días</b>	<b>mar 21/06/16</b>	<b>vie 24/06/16</b>		
10			Modelo de dominio final	1 día	mar 21/06/16	mar 21/06/16	8	Antonio Pesquera Rodríguez
11			Modelo de casos de uso final	1 día	mar 21/06/16	mar 21/06/16	8	Antonio Pesquera Rodríguez
12			Redaccion	3 días	mié 22/06/16	vie 24/06/16	11	Antonio Pesquera Rodríguez
13			<b> Diseño</b>	<b>2 días</b>	<b>sáb 25/06/16</b>	<b>dom 26/06/16</b>		
14			Redaccion	2 días	sáb 25/06/16	dom 26/06/16	12	Antonio Pesquera Rodríguez
15			<b> Implementacion</b>	<b>3 días</b>	<b>lun 27/06/16</b>	<b>mié 29/06/16</b>		
16			Estructuracion del contenido	1 día	lun 27/06/16	lun 27/06/16	14	Antonio Pesquera Rodríguez
17			Redaccion	2 días	mar 28/06/16	mié 29/06/16	16	Antonio Pesquera Rodríguez
18			<b> Pruebas</b>	<b>2 días</b>	<b>jue 30/06/16</b>	<b>vie 01/07/16</b>		
19			Redaccion	2 días	jue 30/06/16	vie 01/07/16	17	Antonio Pesquera Rodríguez
20			<b> Conclusiones</b>	<b>3 días</b>	<b>sáb 02/07/16</b>	<b>lun 04/07/16</b>		
21			Estructuracion del contenido	1 día	sáb 02/07/16	sáb 02/07/16	19	Antonio Pesquera Rodríguez
22			Redaccion	2 días	dom 03/07/16	lun 04/07/16	21	Antonio Pesquera Rodríguez
23			<b> Revision del documento</b>	<b>4 días</b>	<b>mar 05/07/16</b>	<b>vie 08/07/16</b>		
24			Revision y correccion	4 días	mar 05/07/16	vie 08/07/16	22	Antonio Pesquera Rodríguez

Ilustración 8: Planificación del quinto sprint



# Capítulo III: Análisis.

En este capítulo analizamos los objetivos que debe satisfacer la aplicación a construir, los requisitos de funcionalidad y los no funcionales que, pese a no describir funciones a realizar, si nos indican características de funcionamiento.

Posterior a esto, generaremos el modelo de dominio y mostraremos las interacciones de las clases de este modelo mediante los diagramas de secuencia.

## Objetivos del Proyecto.

Las tablas contenidas en esta sección (de la 15 a la 18 ambas incluidas) nos describen los objetivos del proyecto, esto es, todo lo que está pensado que la aplicación resultado de este trabajo debe satisfacer.

<b>Objetivo 1</b>	<b>Aprendizaje de los conceptos relacionados con los campos eléctricos</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Fuente</b>	Manuel Ángel González Delgado.
<b>Descripción</b>	El sistema deberá permitir que el Usuario aprenda los conceptos básicos sobre los campos eléctricos.
<b>Importancia</b>	Vital.
<b>Urgencia</b>	Inmediatamente.

*Tabla 15: Objetivo 1: Aprendizaje de los conceptos relacionados con los campos eléctricos*

<b>Objetivo 2</b>	<b>Evaluación de los conocimientos obtenidos</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Fuente</b>	Manuel Ángel González Delgado.
<b>Descripción</b>	El sistema deberá permitir que el Usuario puede evaluar el nivel de conocimiento que tiene sobre campos eléctricos.
<b>Importancia</b>	Vital.
<b>Urgencia</b>	Inmediatamente.

*Tabla 16: Objetivo 2 Evaluación de los conocimientos obtenidos*

<b>Objetivo 3</b>	<b>Evaluación de los conocimientos obtenidos frente al profesor</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Fuente</b>	Manuel Ángel González Delgado.
<b>Descripción</b>	El sistema deberá permitir que el Alumno evalúe sus conocimientos y el profesor sepa los resultados de esa prueba.
<b>Importancia</b>	Vital.
<b>Urgencia</b>	Inmediatamente.

*Tabla 17: Objetivo 3: Evaluación de los conocimientos obtenidos*

<b>Objetivo 4</b>	<b>Mostrar la Interacción entre varias cargas</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Fuente</b>	Manuel Ángel González Delgado.
<b>Descripción</b>	El sistema deberá permitir que el Usuario observe como varias cargas interactúan sobre un plano, tanto a nivel de líneas de campo como al de potenciales acumulados en un punto.
<b>Importancia</b>	Vital.
<b>Urgencia</b>	Inmediatamente.

*Tabla 18: Objetivo 4: Interacción entre varias cargas*

## Definición de Actores.

La siguiente sección, describe los actores que interactuarán con la aplicación a construir. El alumno puede interactuar de igual manera que el usuario con el sistema con la diferencia de que al tener cuenta de correo de la Uva puede realizar cuestionarios de examen.

El profesor interactúa con el sistema de la misma manera que el alumno, la diferencia radica en que él podrá generar y publicar contenidos con la aplicación de escritorio.

<b>Actor 1</b>	<b>Usuario</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Descripción</b>	Cualquier usuario de la aplicación.

*Tabla 19: Actor 1: Alumno*

<b>Actor 2</b>	<b>Alumno</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Descripción</b>	Alumno de la Universidad de Valladolid.

*Tabla 20. Actor 2: Alumno*

<b>Actor 3</b>	<b>Profesor</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Descripción</b>	Quien modifica los contenidos y obtiene los resultados del cuestionario de evaluación.

*Tabla 21: Actor 3: Profesor*

## Requisitos.

Son funcionalidades que el sistema debe poseer para satisfacer los objetivos o restricciones sobre estas funcionalidades.

### Requisitos funcionales.

Indican todas las funcionalidades que el sistema debe tener para poder satisfacer los objetivos para los que se ha construido. En las tablas de esta sección se describen los requisitos y se muestran las dependencias entre ellos.

<b>R. Funcional 1</b>	<b>Mostrar fundamentos teóricos</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Fuente</b>	Manuel Ángel González Delgado.
<b>Descripción</b>	El sistema deberá mostrar al alumno un resumen de la teoría sobre campos eléctricos que se imparte en la asignatura.
<b>Dependencias</b>	R. Funcional 7.
<b>Importancia</b>	Vital.
<b>Urgencia</b>	Inmediatamente.

*Tabla 22: R. Funcional 1: Mostrar fundamentos teóricos*

<b>R. Funcional 2</b>	<b>Realizar cuestionarios de autoevaluación</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Fuente</b>	Manuel Ángel González Delgado.
<b>Descripción</b>	El sistema deberá permitir que el alumno evalúe sus conocimientos sobre campos eléctricos.
<b>Dependencias</b>	R. Funcional 7.
<b>Importancia</b>	Vital.
<b>Urgencia</b>	Inmediatamente.

*Tabla 23: R. Funcional 2: Realizar cuestionarios de autoevaluación*

<b>R. Funcional 3</b>	<b>Realizar cuestionarios de examen</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Fuente</b>	Manuel Ángel González Delgado.
<b>Descripción</b>	El sistema deberá permitir que el alumno evalúe y el profesor obtenga los resultados.
<b>Dependencias</b>	R. Funcional 4, R. Funcional 5, R. Funcional 6, R. Funcional 7.
<b>Importancia</b>	Vital.
<b>Urgencia</b>	Inmediatamente.

*Tabla 24: R. Funcional 3: Realizar cuestionarios de evaluación*

<b>R. Funcional 4</b>	<b>Tomar foto en el cuestionario de examen</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Fuente</b>	Manuel Ángel González Delgado.
<b>Descripción</b>	El sistema deberá tomar una foto del alumno en un momento aleatorio si realiza un cuestionario de examen para probar que lo hace él.
<b>Dependencias</b>	Ninguna.
<b>Importancia</b>	Vital.
<b>Urgencia</b>	Inmediatamente.

*Tabla 25: R. Funcional 4: Tomar foto en el cuestionario de examen*

<b>R. Funcional 5</b>	<b>Validar la identidad antes del cuestionario de examen</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Fuente</b>	Manuel Ángel González Delgado.
<b>Descripción</b>	El sistema deberá permitir que el alumno verifique su identidad frente al servidor SMTP de la UVa (cartero).
<b>Dependencias</b>	Ninguna.
<b>Importancia</b>	Vital.
<b>Urgencia</b>	Inmediatamente.

*Tabla 26: R. Funcional 5: Validar la identidad antes del cuestionario de examen*

<b>R. Funcional 6</b>	<b>Envío de resultados de un cuestionario de examen</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Fuente</b>	Manuel Ángel González Delgado.
<b>Descripción</b>	El sistema deberá enviar al profesor los resultados del cuestionario de examen.
<b>Dependencias</b>	Ninguna.
<b>Importancia</b>	Vital.
<b>Urgencia</b>	Inmediatamente.

*Tabla 27: R. Funcional 6: Envío de resultados de un cuestionario de examen*

<b>R. Funcional 7</b>	<b>Descargar contenidos</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Fuente</b>	Manuel Ángel González Delgado.
<b>Descripción</b>	El sistema deberá permitir descargar los contenidos más actualizados (cuestionarios y teoría).
<b>Dependencias</b>	Ninguna.
<b>Importancia</b>	Vital.
<b>Urgencia</b>	Inmediatamente.

*Tabla 28: R. Funcional 7: Descargar contenidos*

<b>R. Funcional 8</b>	<b>Generar teoría</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Fuente</b>	Manuel Ángel González Delgado.
<b>Descripción</b>	El sistema deberá permitir actualizar los contenidos teóricos que muestra la aplicación.
<b>Dependencias</b>	Ninguna.
<b>Importancia</b>	Vital.
<b>Urgencia</b>	Inmediatamente.

*Tabla 29: R. Funcional 8: Generar Teoría*

<b>R. Funcional 9</b>	<b>Generar cuestionarios</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Fuente</b>	Manuel Ángel González Delgado.
<b>Descripción</b>	El sistema deberá permitir actualizar los cuestionarios que utiliza la aplicación para la evaluación.
<b>Dependencias</b>	Ninguna.
<b>Importancia</b>	Vital.
<b>Urgencia</b>	Inmediatamente.

*Tabla 30: R. Funcional 9: Generar cuestionarios*

<b>R. Funcional 10</b>	<b>Simulación</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Fuente</b>	Manuel Ángel González Delgado.
<b>Descripción</b>	El sistema deberá mostrar cómo interaccionan varias cargas ubicadas en un punto del espacio. Mostrará las líneas de campo y el potencial eléctrico en cada punto.
<b>Dependencias</b>	Ninguna.
<b>Importancia</b>	Vital.
<b>Urgencia</b>	Inmediatamente.

*Tabla 31: R. Funcional 10: Simulación*

Requisitos no funcionales.

Estos requisitos no describen funcionalidades que debemos incorporar al sistema. Lo que indican son restricciones sobre la funcionalidad del sistema a construir. Las tablas de esta sección describen los dos requisitos no funcionales que nos afectan: comunicarnos con el servidor de correo de la Uva y almacenar la teoría y los cuestionarios en Dropbox.

<b>R. No Funcional 1</b>	<b>Servidor de correo</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Fuente</b>	Manuel Ángel González Delgado.
<b>Descripción</b>	El sistema deberá utilizar el servidor de correo de la Uva para enviar los resultados del test.
<b>Dependencias</b>	Ninguna.
<b>Importancia</b>	Vital.
<b>Urgencia</b>	Inmediatamente.

*Tabla 32: R. No Funcional 1: Servidor de correo*

<b>R. No Funcional 2</b>	<b>Almacenamiento de datos</b>
<b>Autor</b>	Antonio Pesquera Rodríguez.
<b>Fuente</b>	Manuel Ángel González Delgado.
<b>Descripción</b>	El sistema deberá usar un almacenamiento en la nube de Dropbox para guardar los cuestionarios y la teoría y poder descargarlos desde allí.
<b>Dependencias</b>	Ninguna.
<b>Importancia</b>	Vital.
<b>Urgencia</b>	Inmediatamente.

*Tabla 33: R. No Funcional 2: Almacenamiento de datos*

Casos de Uso.

A continuación, se enumeran y describen los casos de uso del sistema. En primer lugar, un diagrama de casos de uso muestra cuáles son, cómo se relacionan, y que rol realiza la secuencia de pasos de los casos de uso enumerados. Posteriormente, se describe la secuencia de interacción de cada uno de ellos.



## Diagrama de Casos de Uso.

Este diagrama muestra los casos de uso asociados a sus roles y las interacciones entre ellos. Como podemos ver, tenemos dos casos de uso que se relacionan con varios. El primero de ellos es “Realizar cuestionario de examen” que requiere siempre de la funcionalidad de otros (incluye) para satisfacer su requisito asociado. El otro caso es “Realizar simulación” que dependiendo de la secuencia de acciones depende de uno u otro caso de uso asociado (extends).

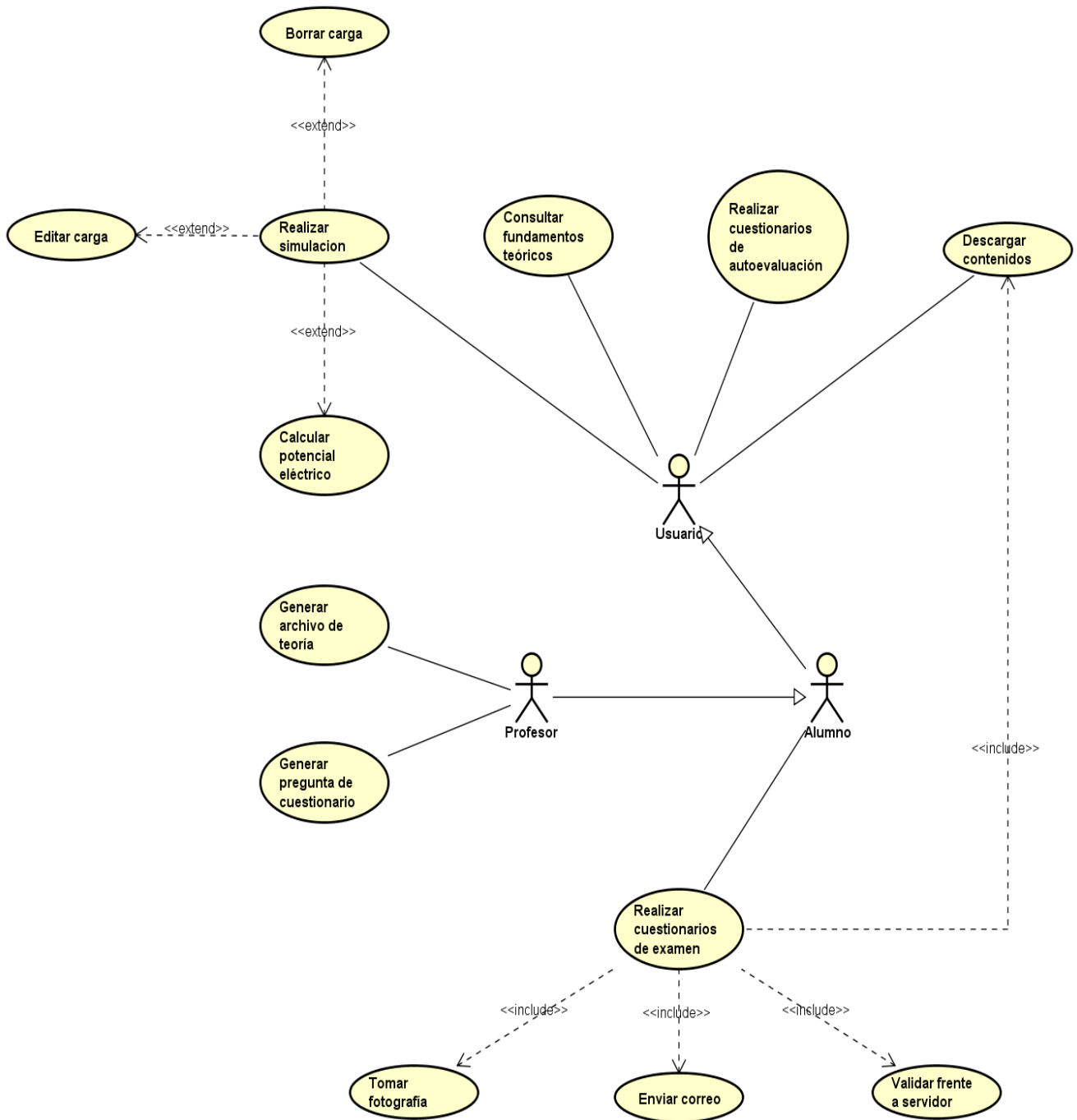


Ilustración 9: Diagrama de casos de uso del análisis

Descripción de los casos de uso.

Las tablas contenidas en esta sección (de la 34 a la 46 ambas incluidas) describen la secuencia de interacciones entre el sistema y el actor para realizar el caso de uso. A parte de la secuencia, se indican también precondiciones y postcondiciones y secuencias alternativas.

<b>Caso de Uso 1</b>	<b>Consultar fundamentos teóricos</b>
<b>Resumen</b>	El Sistema deberá comportarse de esta manera cuando el Usuario consulta la teoría.
<b>Actor</b>	Usuario.
<b>Precondición</b>	Ninguna.
<b>Postcondición</b>	Ninguna.
<b>Secuencia base</b>	1. El Usuario pide al sistema consultar la teoría. 2. El Sistema carga la teoría y la muestra por pantalla.
<b>Secuencia alternativa</b>	Ninguna.
<b>Notas</b>	

*Tabla 34: Caso de Uso 1: Consultar fundamentos teóricos*

<b>Caso de Uso 2</b>	<b>Realizar cuestionarios de autoevaluación</b>
<b>Resumen</b>	El Sistema deberá comportarse de esta manera cuando el Usuario realiza un cuestionario de autoevaluación.
<b>Actor</b>	Usuario.
<b>Precondición</b>	Ninguna.
<b>Postcondición</b>	Ninguna.
<b>Secuencia base</b>	1. El Usuario pide al Sistema realizar un cuestionario de autoevaluación. 2. El Sistema carga las preguntas y las muestra por pantalla. 3. El Usuario responde una pregunta. 4. El Sistema comprueba la respuesta y la muestra por pantalla. 5. El Sistema le muestra al final los resultados del cuestionario.
<b>Secuencia alternativa</b>	3,4,5 Si el Usuario sale antes de finalizar el cuestionario el caso de uso queda sin efecto.
<b>Notas</b>	Antes de que el Sistema muestre los resultados finales, el Usuario ha respondido a todas las preguntas.

*Tabla 35: Caso de Uso 2: Realizar cuestionarios de autoevaluación*

<b>Caso de Uso 3</b>	<b>Validar usuario frente a servidor</b>
<b>Resumen</b>	El Sistema deberá comportarse de esta manera cuando el Usuario se valida.
<b>Actor</b>	Alumno.
<b>Precondición</b>	El Alumno ha entrado en un cuestionario de examen.
<b>Postcondición</b>	Ninguna.
<b>Secuencia base</b>	1. El Alumno introduce sus credenciales pide al Sistema que las compruebe. 2. El Sistema comprueba las credenciales y muestra por pantalla el resultado.
<b>Secuencia alternativa</b>	2. Si los datos son incorrectos, el Sistema se lo comunica al Alumno.
<b>Notas</b>	

*Tabla 36: Caso de Uso 3: Validar frente a servidor*

<b>Caso de Uso 4</b>	<b>Tomar fotografía</b>
<b>Resumen</b>	El Sistema deberá comportarse de esta manera cuando debe tomar una fotografía.
<b>Actor</b>	Alumno.
<b>Precondición</b>	El Alumno ha entrado en un cuestionario de examen.
<b>Postcondición</b>	Ninguna.
<b>Secuencia base</b>	<ol style="list-style-type: none"> <li>1. El Sistema calcula el momento en que debe tomar una foto.</li> <li>2. El Usuario interactúa con el Sistema hasta que llega el momento que el Sistema calculó.</li> <li>3. El Sistema toma una foto y la almacena.</li> </ol>
<b>Secuencia alternativa</b>	Ninguna.
<b>Notas</b>	

*Tabla 37: Caso de Uso 4: Tomar fotografía*

<b>Caso de Uso 5</b>	<b>Enviar resultados</b>
<b>Resumen</b>	El Sistema deberá comportarse de esta manera cuando debe enviar los resultados de un cuestionario.
<b>Actor</b>	Alumno.
<b>Precondición</b>	El Alumno ha finalizado un cuestionario de examen.
<b>Postcondición</b>	Ninguna.
<b>Secuencia base</b>	1. El Sistema envía los resultados del examen al profesor
<b>Secuencia alternativa</b>	Ninguna.
<b>Notas</b>	

*Tabla 38: Caso de Uso 5: Enviar resultados*

<b>Caso de Uso 6</b>	<b>Descargar contenidos</b>
<b>Resumen</b>	El Sistema deberá comportarse de esta manera cuando el Usuario quiere descargar los contenidos más actualizados.
<b>Actor</b>	Usuario.
<b>Precondición</b>	Ninguna.
<b>Postcondición</b>	Ninguna.
<b>Secuencia base</b>	<ol style="list-style-type: none"> <li>1. El Usuario pulsa en descargar contenidos</li> <li>2. El Sistema descarga y almacena los contenidos</li> </ol>
<b>Secuencia alternativa</b>	2. Si no existen contenidos el Sistema informa al Usuario y el caso de uso queda sin efecto.
<b>Notas</b>	

*Tabla 39: Caso de Uso 6: Descargar contenidos*

<b>Caso de Uso 7</b>	<b>Realizar cuestionarios de examen</b>
<b>Resumen</b>	El Sistema deberá comportarse de esta manera cuando el Usuario realiza un cuestionario de examen.
<b>Actor</b>	Alumno.
<b>Precondición</b>	El Alumno ha elegido cuestionario de examen. El Alumno se ha validado.
<b>Postcondición</b>	Ninguna.
<b>Secuencia base</b>	<ol style="list-style-type: none"> <li>1. El Sistema carga las preguntas, genera el momento de tirar la foto muestra las preguntas y enciende el temporizador.</li> <li>2. El Alumno responde una pregunta.</li> <li>3. El Sistema almacena la respuesta.</li> <li>4. El Alumno solicita corregir.</li> <li>5. El Sistema para el temporizador y corrige las preguntas.</li> <li>6. Comienza el caso de uso Enviar Email</li> <li>7. El Sistema muestra los resultados.</li> </ol>
<b>Secuencia alternativa</b>	<ol style="list-style-type: none"> <li>2. Si el Alumno abandona el cuestionario antes de corregirlo el caso de uso queda sin efecto.</li> <li>2. Si es el momento de tomar la foto comienza el caso de uso "Tomar Fotografía", a continuación, el caso de uso continua por el paso 3.</li> </ol>
<b>Notas</b>	El paso 2 se repite para todas las preguntas

*Tabla 40: Caso de Uso 7: Realizar cuestionarios de examen*

<b>Caso de Uso 8</b>	<b>Realizar simulación</b>
<b>Resumen</b>	El Sistema deberá comportarse de esta manera cuando el Usuario realiza una simulación.
<b>Actor</b>	Usuario.
<b>Precondición</b>	Ninguna.
<b>Postcondición</b>	Ninguna.
<b>Secuencia base</b>	<ol style="list-style-type: none"> <li>1. El Usuario pide al Sistema realizar una simulación.</li> <li>2. El Sistema le lleva al simulador.</li> <li>3. El Usuario pide añadir una carga.</li> <li>4. El Sistema solicita los datos de la carga (valor, signo y posición).</li> <li>5. El Usuario introduce los datos de la carga.</li> <li>6. El Sistema genera la carga y recalcula las líneas de campo.</li> </ol>
<b>Secuencia alternativa</b>	<ol style="list-style-type: none"> <li>3. Si el Usuario quiere borrar una carga se sigue por el punto de extensión "borrar carga", a continuación, el caso de uso continua en el paso 6.</li> <li>3. Si el Usuario quiere editar una carga se sigue por el punto de extensión "editar carga", a continuación, el caso de uso continua en el paso 6.</li> <li>3. Si el Usuario conocer el potencial en un punto se sigue por el punto de extensión "calcular potencial", a continuación, el caso de uso continua en el paso 6.</li> </ol>
<b>Notas</b>	Los pasos del 3 al 6 ambos inclusive pueden repetirse varias veces

*Tabla 41: Caso de Uso 8: Realizar Simulación*

<b>Caso de Uso 9</b>	<b>Calcular potencial eléctrico</b>
<b>Resumen</b>	El Sistema deberá comportarse de esta manera cuando el Usuario le pide calcular el potencial eléctrico generado por las cargas que hay en la simulación.
<b>Actor</b>	Usuario.
<b>Precondición</b>	El Usuario está dentro de la simulación
<b>Postcondición</b>	Ninguna.
<b>Secuencia base</b>	<ol style="list-style-type: none"> <li>1. El Usuario solicita calcular el potencial eléctrico.</li> <li>2. El Sistema le pide el punto donde calcularlo.</li> <li>3. El Usuario introduce el punto.</li> <li>4. El Sistema calcula el potencial eléctrico y se lo muestra al Usuario.</li> </ol>
<b>Secuencia alternativa</b>	
<b>Notas</b>	

Tabla 42: Caso de Uso 9: Calcular potencial eléctrico

<b>Caso de Uso 10</b>	<b>Generar archivo de teoría</b>
<b>Resumen</b>	El Sistema deberá comportarse de esta manera cuando el Profesor le pide generar un archivo de teoría.
<b>Actor</b>	Profesor.
<b>Precondición</b>	Ninguna.
<b>Postcondición</b>	Se genera un archivo con la teoría.
<b>Secuencia base</b>	<ol style="list-style-type: none"> <li>1. El Profesor le pide al Sistema escribir un fichero de teoría.</li> <li>2. El Sistema le pide que introduzca los contenidos.</li> <li>3. El Profesor añade los contenidos.</li> <li>4. El Sistema toma esos contenidos y exporta un fichero.</li> </ol>
<b>Secuencia alternativa</b>	2,3 Si el Profesor cancela la introducción el caso de uso queda sin efecto.
<b>Notas</b>	

Tabla 43: Caso de Uso 10: Generar teoría

<b>Caso de Uso 11</b>	<b>Generar pregunta de cuestionario</b>
<b>Resumen</b>	El Sistema deberá comportarse de esta manera cuando el Profesor le pide generar una pregunta.
<b>Actor</b>	Profesor.
<b>Precondición</b>	Ninguna.
<b>Postcondición</b>	Se genera un archivo con la pregunta.
<b>Secuencia base</b>	<ol style="list-style-type: none"> <li>1. El Profesor le pide al Sistema escribir una pregunta de cuestionario.</li> <li>2. El Sistema le solicita que introduzca la pregunta.</li> <li>3. El Profesor introduce la pregunta.</li> <li>4. El Sistema la almacena y solicita una respuesta.</li> <li>5. El Usuario introduce el texto de la respuesta.</li> <li>6. El Sistema solicita al Usuario que indique la respuesta correcta</li> <li>7. El Profesor introduce la respuesta correcta</li> <li>8. El Sistema genera un fichero con la respuesta</li> </ol>
<b>Secuencia alternativa</b>	2,3,4,5,6,7 Si el Profesor cancela la introducción el caso de uso queda sin efecto.
<b>Notas</b>	Los pasos 4 y 5 se repiten tantas veces como preguntas tiene el cuestionario.

Tabla 44: Caso de Uso 11: Generar pregunta de cuestionario

<b>Caso de Uso 12</b>	<b>Borrar carga</b>
<b>Resumen</b>	El Sistema deberá comportarse de esta manera cuando el Usuario elimina una carga
<b>Actor</b>	Usuario.
<b>Precondición</b>	Hay cargas añadidas en la simulación.
<b>Postcondición</b>	La carga elegida es eliminada.
<b>Secuencia base</b>	<ol style="list-style-type: none"> <li>1. El Usuario pide al Sistema eliminar una carga.</li> <li>2. El Sistema pregunta que carga</li> <li>3. El Usuario elige la carga</li> <li>4. El Sistema elimina esa carga</li> </ol>
<b>Secuencia alternativa</b>	Ninguna
<b>Notas</b>	

*Tabla 45: Caso de Uso 12: Borrar carga*

<b>Caso de Uso 13</b>	<b>Editar carga</b>
<b>Resumen</b>	El Sistema deberá comportarse de esta manera cuando el Usuario edita una carga
<b>Actor</b>	Usuario.
<b>Precondición</b>	Hay cargas añadidas en la simulación.
<b>Postcondición</b>	La carga elegida es editada.
<b>Secuencia base</b>	<ol style="list-style-type: none"> <li>1. El Usuario pide al Sistema editar una carga.</li> <li>2. El Sistema pide el nuevo valor, el nuevo signo y la carga a editar</li> <li>3. El Usuario introduce los datos</li> <li>4. El Sistema edita esa carga</li> </ol>
<b>Secuencia alternativa</b>	Ninguna
<b>Notas</b>	

*Tabla 46: Caso de Uso 13: Editar carga*

## Modelo de dominio.

El modelo de dominio contiene las clases de obtenidas en el análisis, como estas clases están relacionadas, sus atributos y las funciones que deben poseer. En un rápido vistazo podemos ver que las clases involucradas en los requisitos relacionados con los cuestionarios, las clases relacionadas con la presentación de la teoría y las relacionadas con la simulación no tienen ninguna relación, lo que nos lleva a pensar la aplicación principal puede ser desarrollada como varios subsistemas independientes. Visto de otra forma, durante el avance del desarrollo del proyecto podemos tener ejecutables intermedios (lo que no refuerza la idea de usar metodologías ágiles de desarrollo).

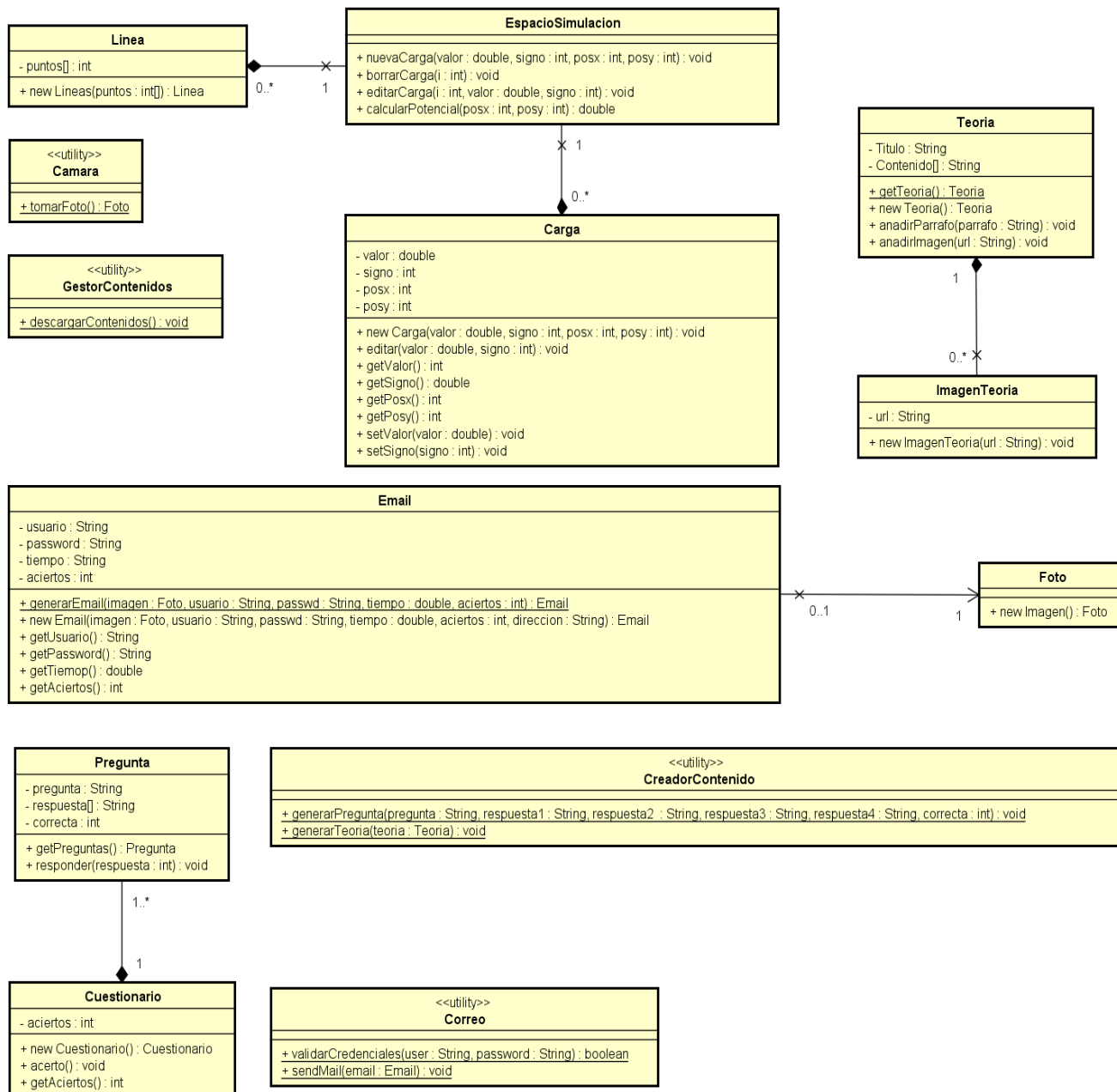


Ilustración 10: Modelo de Dominio.

## Descripción del modelo de dominio.

- **Línea:** Representa una línea de campo eléctrico. Está formada por un vector de puntos.
- **EspacioSimulacion:** Lugar donde se ubican las cargas y donde se pintan las líneas de campo. Permite agregar, eliminar o editar las cargas que contiene. Es un agregado de elementos Carga y elementos Línea.
- **Carga:** Poseen un valor, un signo y una posición y funciones que permiten editar o conocer el valor de estos atributos.
- **Teoría:** Representa la teoría a mostrar. Contiene un conjunto de párrafos representado por un vector de "String" y puede que un conjunto de imágenes (clase ImagenTeoria).
- **ImagenTeoria:** Representa cada una de las imágenes que se muestran en la teoría. Contiene la dirección de la imagen que debe mostrar.
- **Camara:** Utilidad que nos permite tomar una fotografía.
- **Foto:** Representa la fotografía que es devuelta por la función de captura (tomarFoto) de la clase Camara. La clase Email contiene un elemento de este tipo.
- **Email:** Contiene el identificador de un usuario, su clave, el número de aciertos en un cuestionario de examen y el tiempo que necesito el usuario para resolver este cuestionario. La utilidad Correo envía objetos de este tipo.
- **Correo:** Utilidad que permite validar las credenciales de un usuario y enviar los correos con los resúmenes de los intentos al realizar cuestionarios de examen. Dicho de otro modo, interacciona con el servidor de correo.
- **GestorContenidos:** Utilidad encargada de descargar los archivos de teoría y los cuestionarios de la nube y colocarlos en el directorio correcto para poderlos mostrar.
- **CreadorContenido:** Utilidad que genera elementos de teoría o cuestionarios para mostrar en la aplicación.
- **Cuestionario:** Esta clase es un agregado de preguntas que además almacena el número de estas que se han respondido correctamente.
- **Pregunta:** Contienen una pregunta, varias respuestas y un valor que indica cuál de ellas es la correcta.



## Diagramas de Secuencia.

Los diagramas de secuencia nos muestran como el actor interacciona con las clases del modelo de dominio y como estas interaccionan entre sí mediante paso de mensajes.

Consultar fundamentos teóricos.

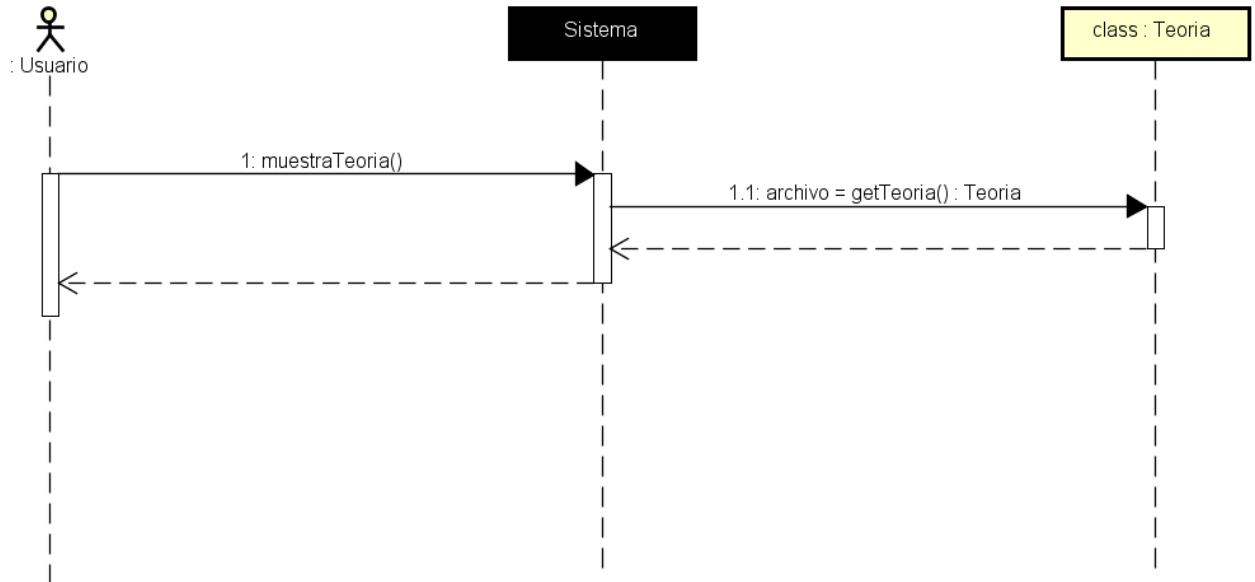


Ilustración 11: Consultar fundamentos teóricos.

Validar frente al servidor.

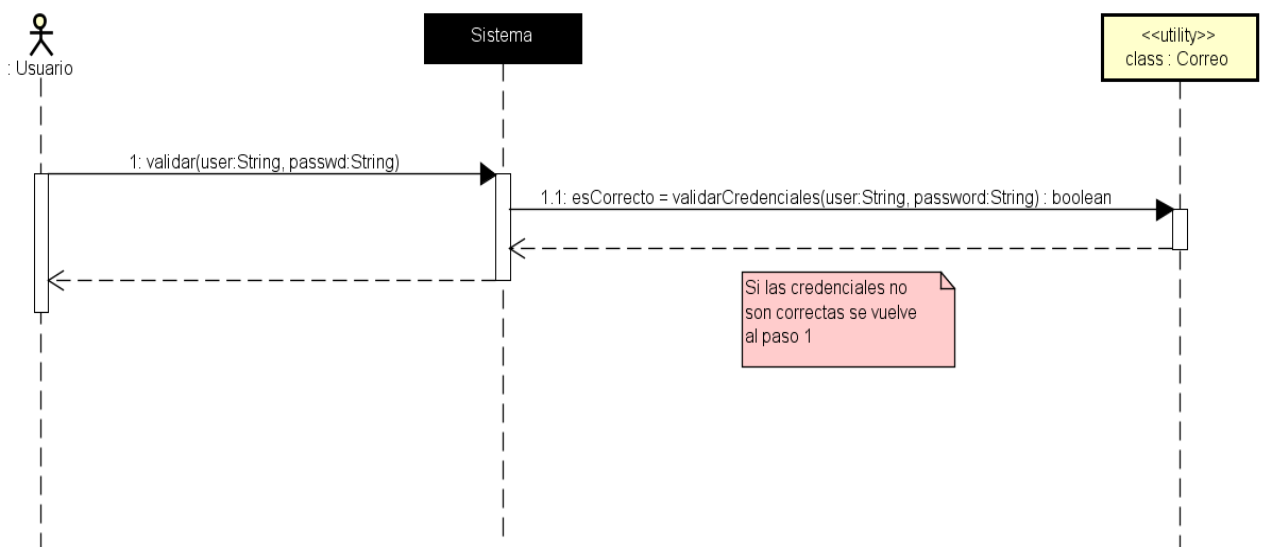


Ilustración 12: Validar frente al servidor.

### Realizar cuestionario de autoevaluación.

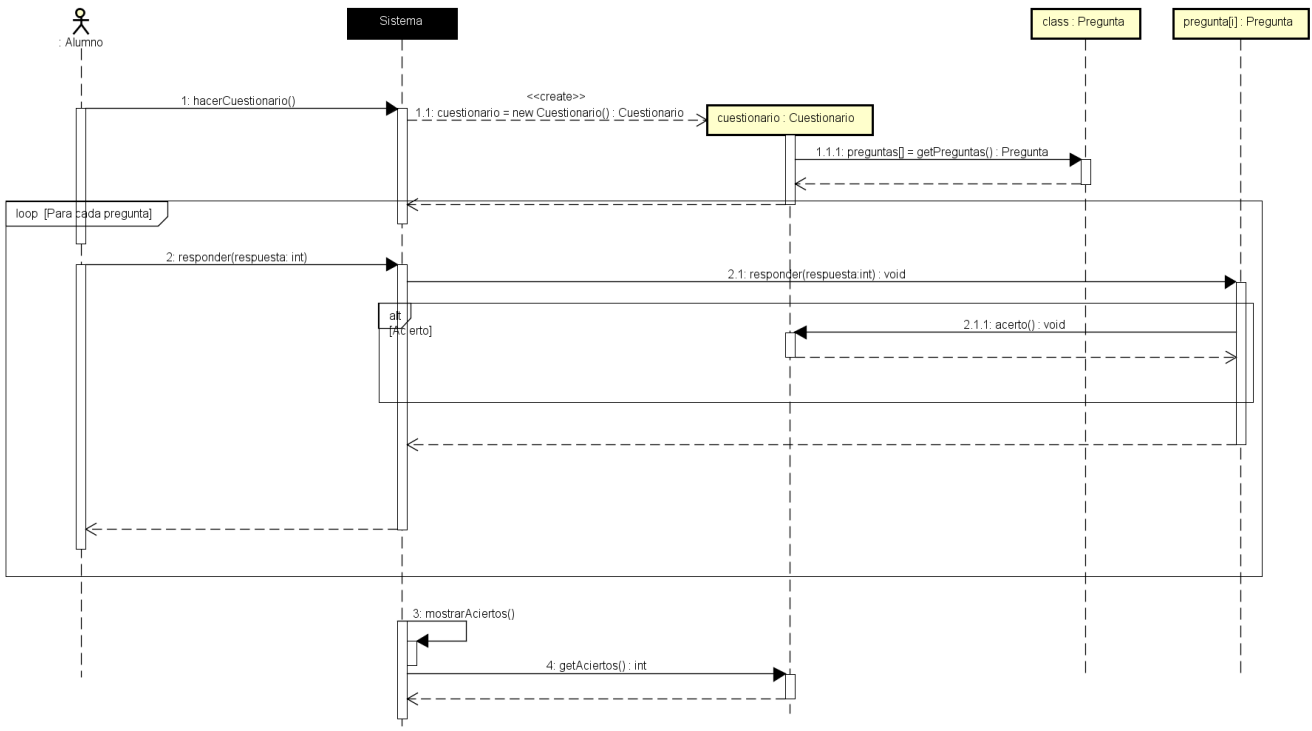


Ilustración 13: Realizar cuestionario de autoevaluación.

### Tomar fotografía.

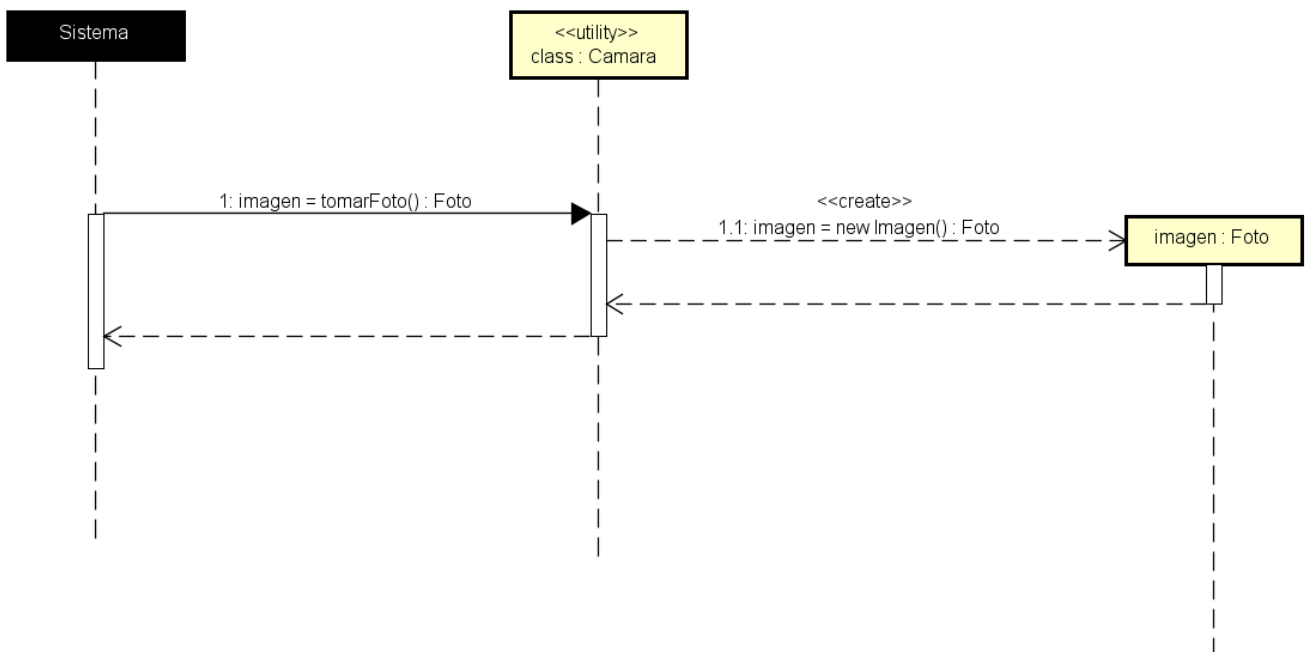


Ilustración 14: Tomar fotografía.

Enviar resultados.

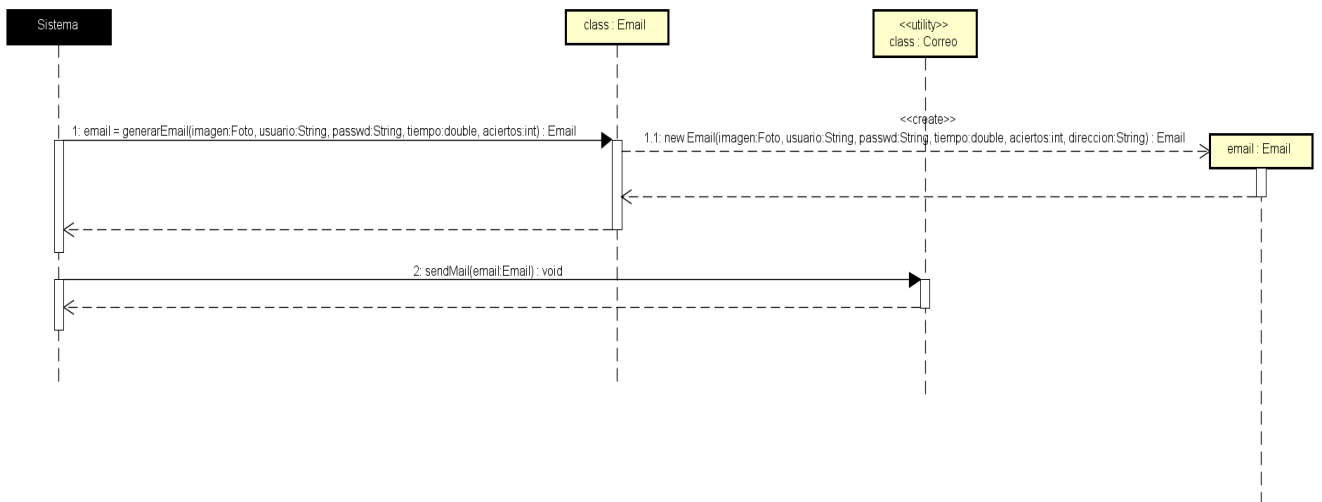


Ilustración 15: Enviar resultados.

Descargar contenidos.

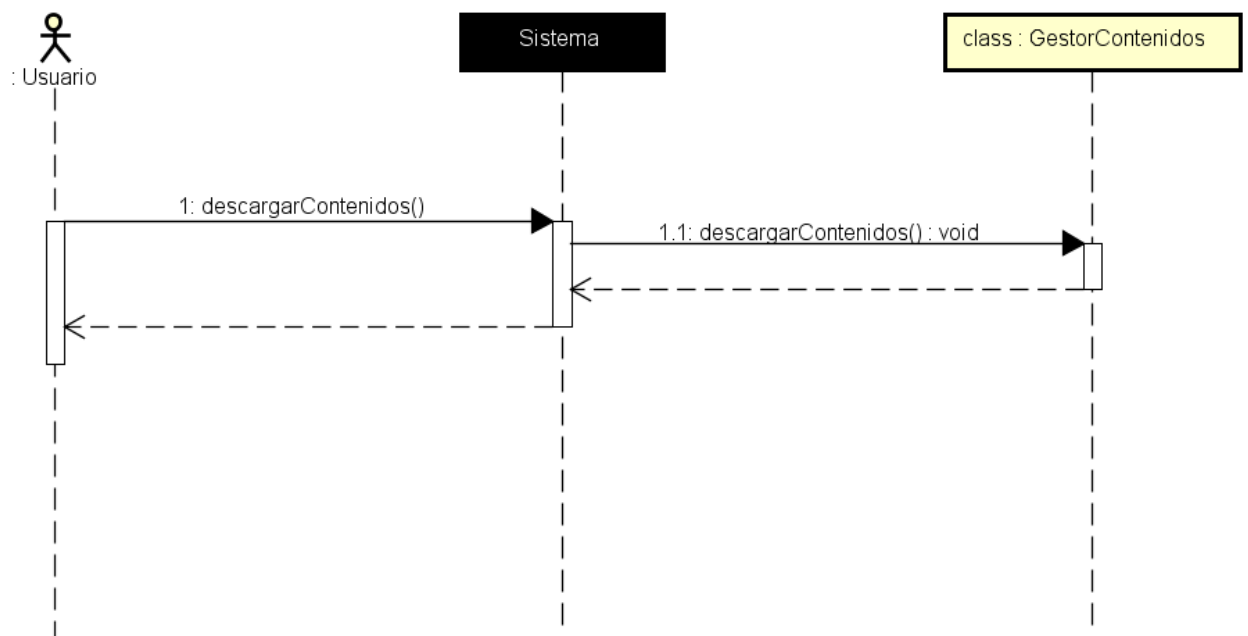


Ilustración 16: Descargar contenidos.

# Realizar simulación.

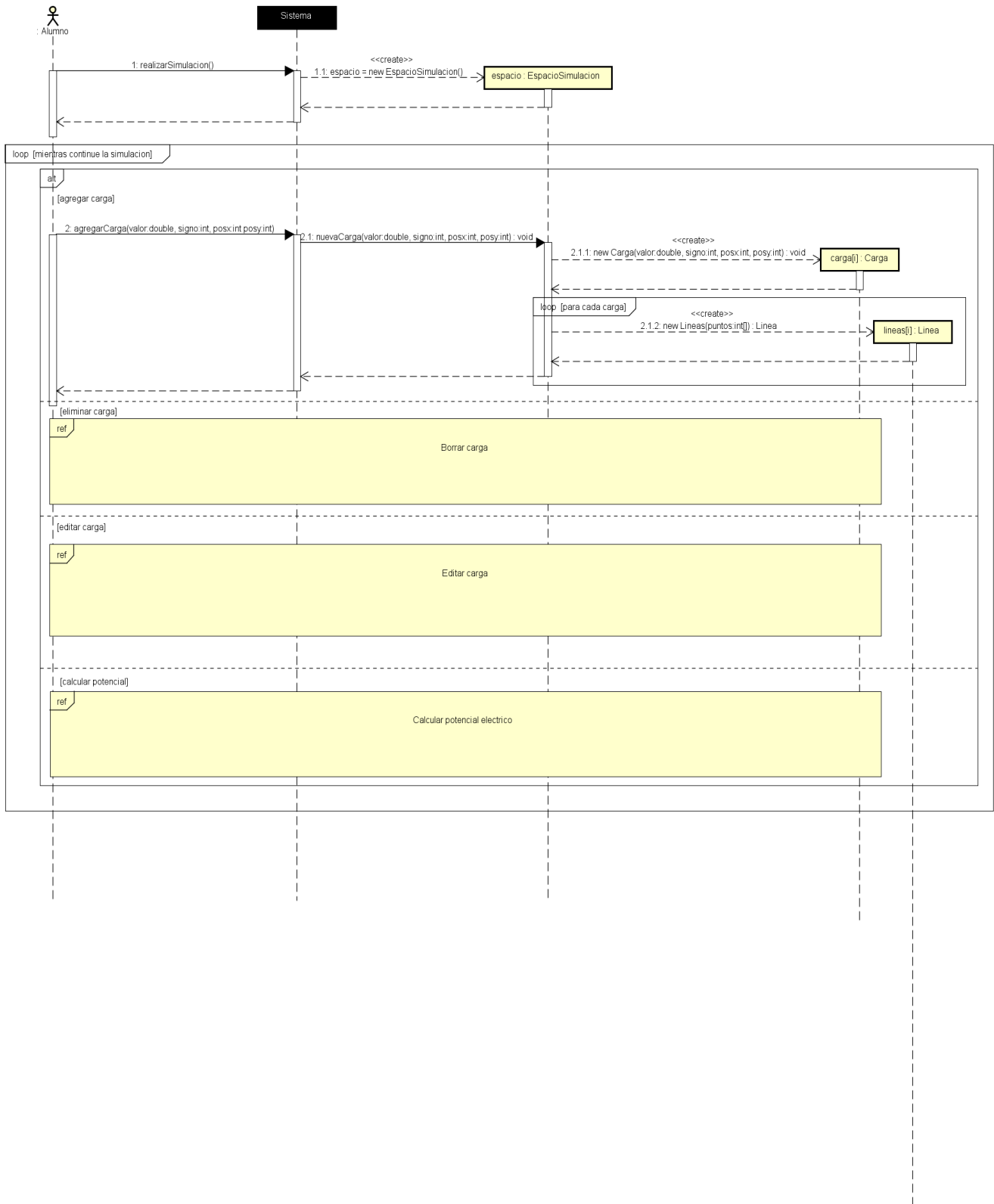


Ilustración 17: Realizar simulación.

Calcular potencial eléctrico.

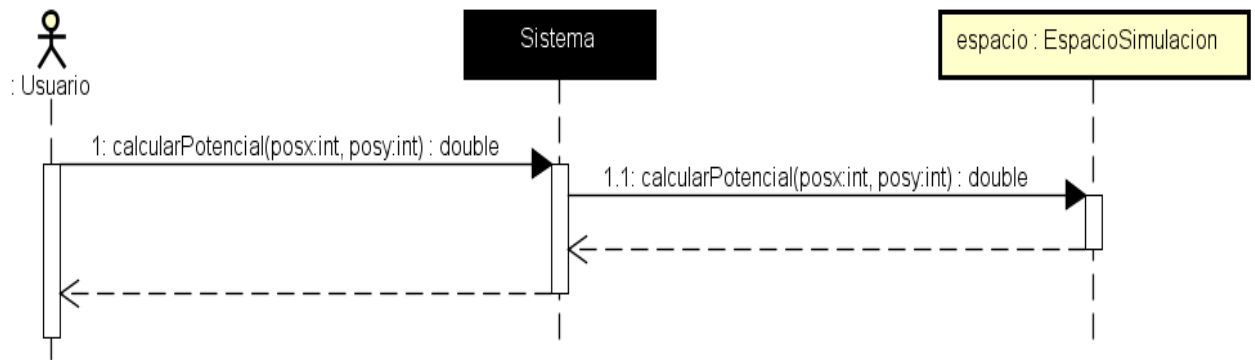


Ilustración 18: Calcular potencial eléctrico.

Editar carga.

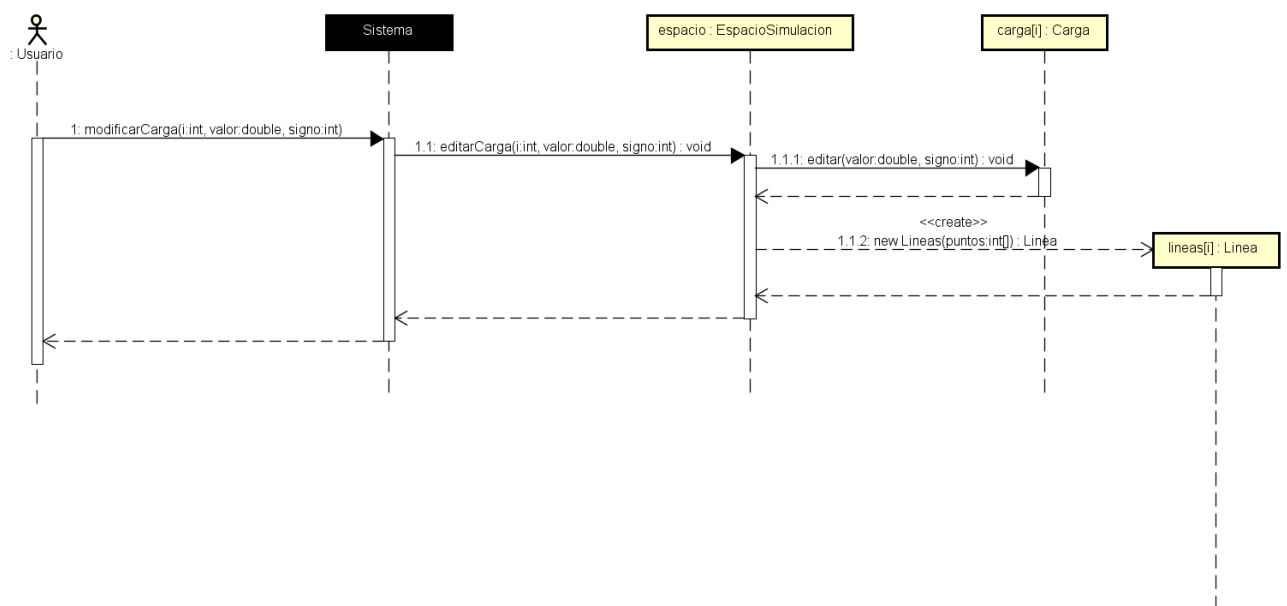


Ilustración 19: Editar carga.

Borrar carga.

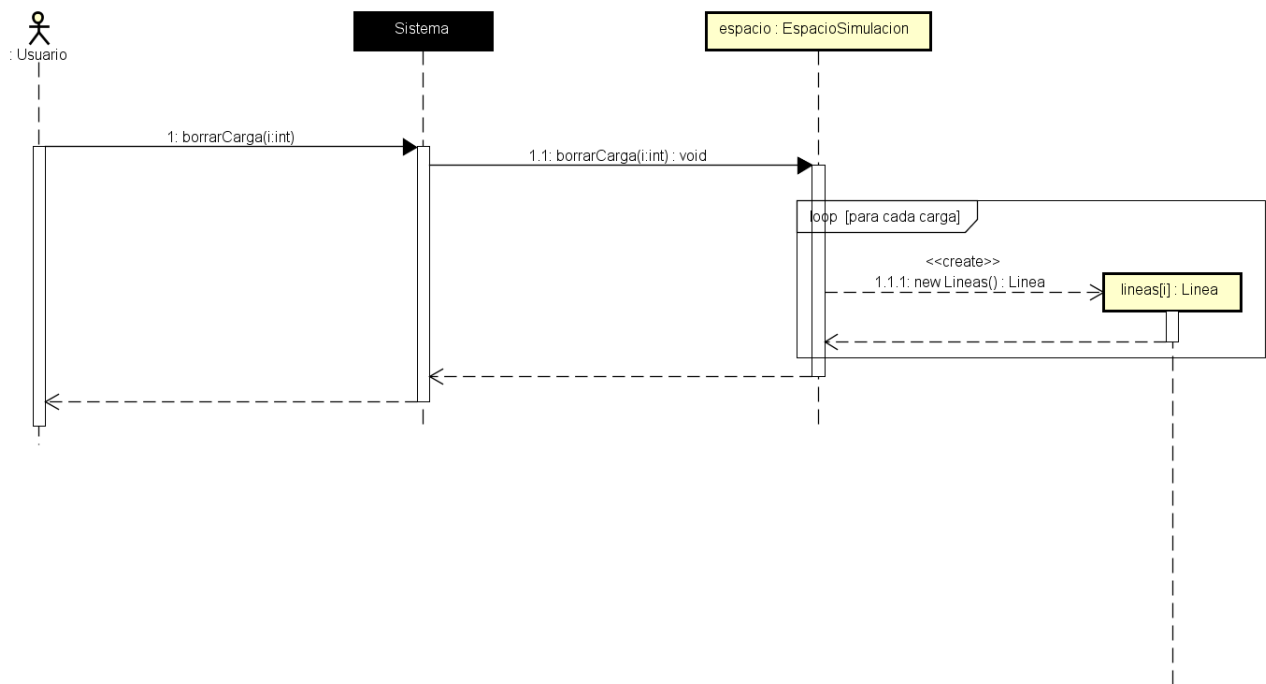


Ilustración 20: Borrar carga.

Generar pregunta de cuestionario.

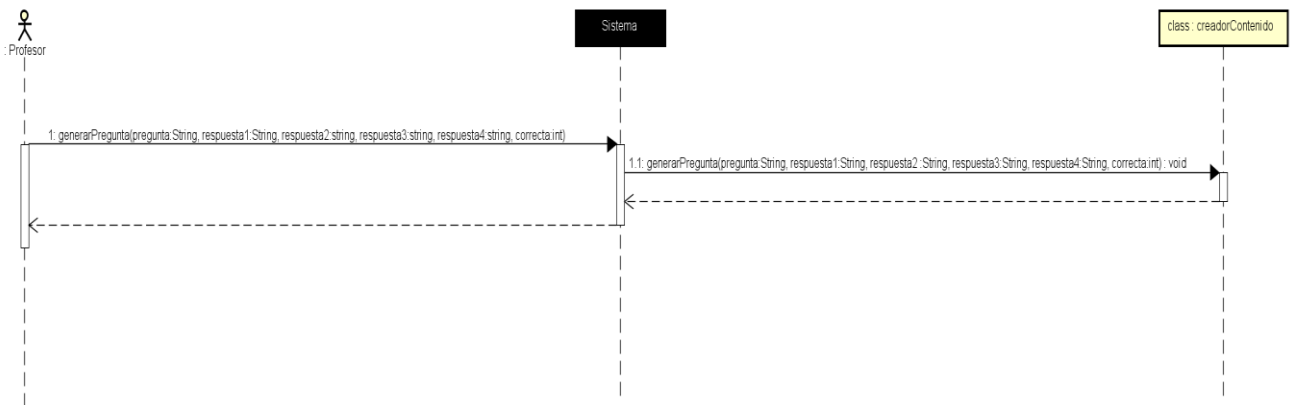


Ilustración 21: Generar pregunta de cuestionario.

# Realizar cuestionario de examen.

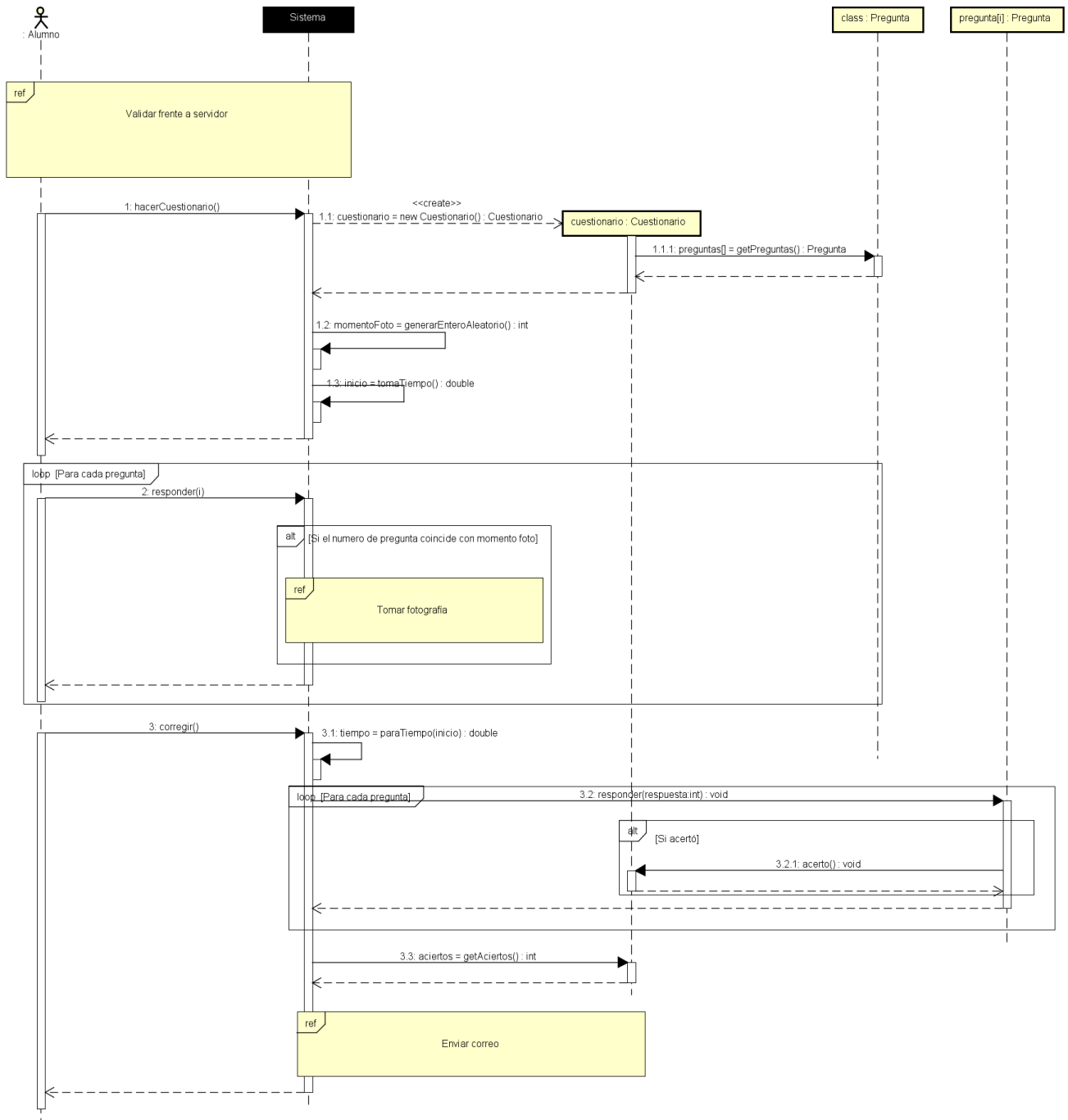


Ilustración 22: Realizar cuestionario de examen.

# Generar archivo de teoría.

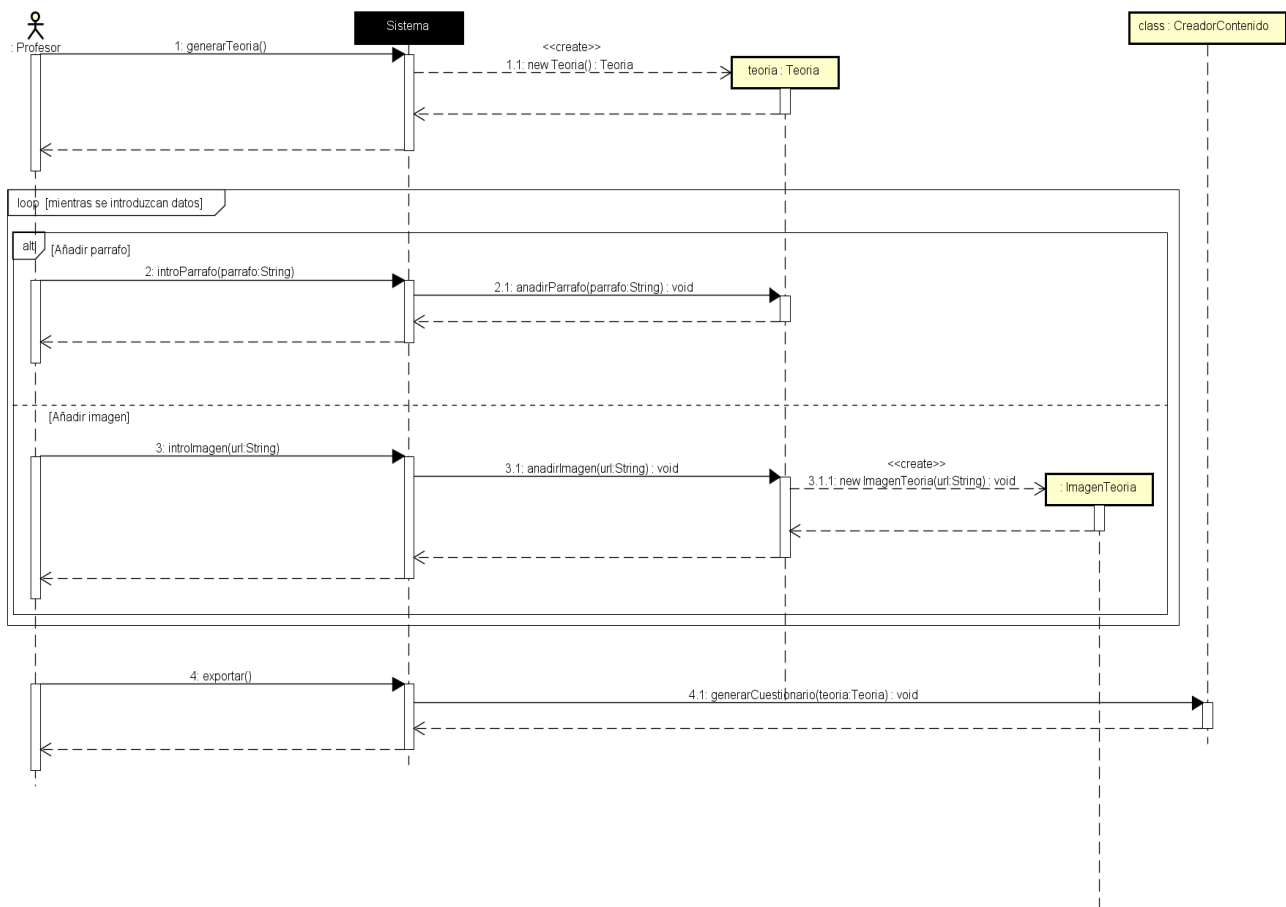


Ilustración 23: Generar archivo de teoría.



# Capítulo IV: Diseño.

En este capítulo veremos que clases de diseño están contenidas en cada caso de uso, como se relacionan para poder llevar a cabo la funcionalidad del caso de uso, la estructura de cada una de las clases de diseño y finalmente los diagramas de secuencia de cada caso de uso en diseño.

Estos diagramas de secuencia pueden visualizarse perfectamente en la versión digital de la memoria, por el contrario, en la versión en papel puede que esta visualización sea más costosa.

## Modelos de diseño.

En los siguientes diagramas veremos cómo se relacionan las clases de diseño que componen cada una de las funcionalidades de la aplicación.

Consultar fundamentos teóricos.

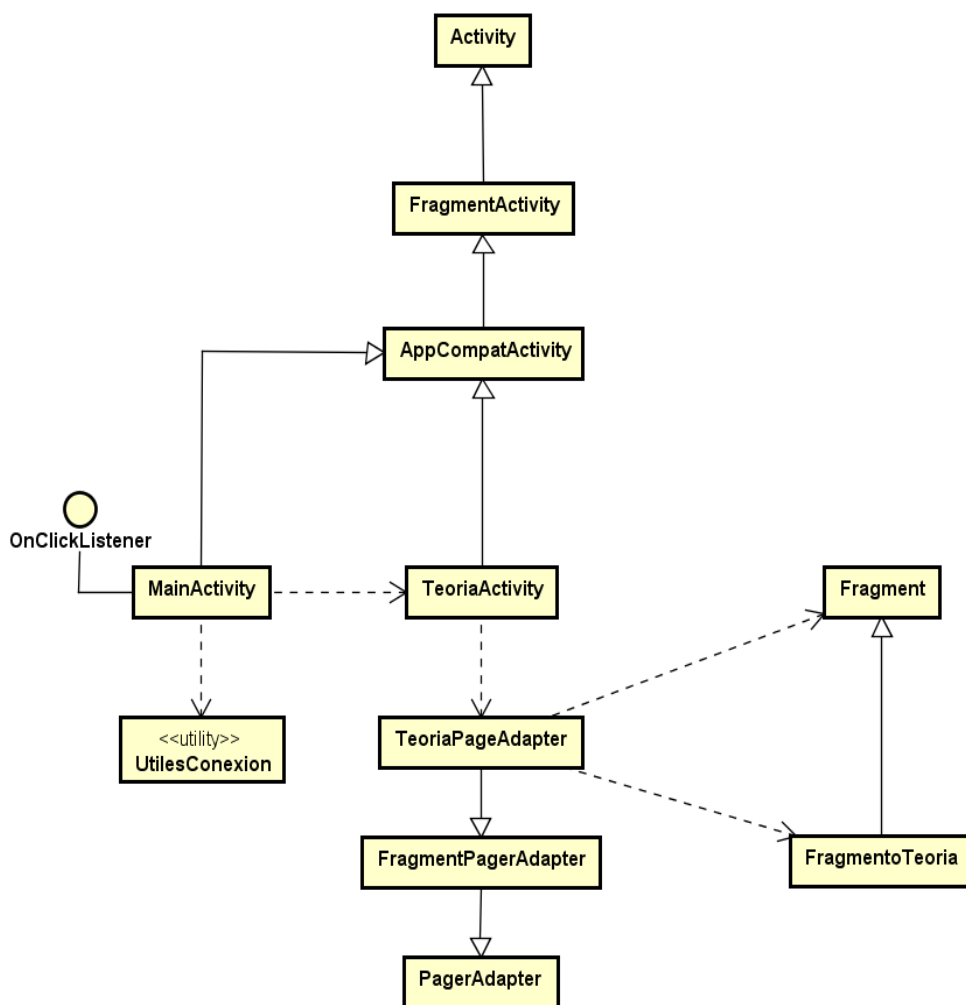


Ilustración 24: Modelo de diseño del caso de uso "Consultar fundamentos teóricos"

La ilustración 24 muestra las clases de diseño implicadas en la presentación de los fundamentos teóricos. La actividad TeoriaActivity genera un TeoriaPagerAdapter que crea y referencia las páginas de teoría. Estas páginas son FragmentoTeoria.



Realizar cuestionario de autoevaluación.

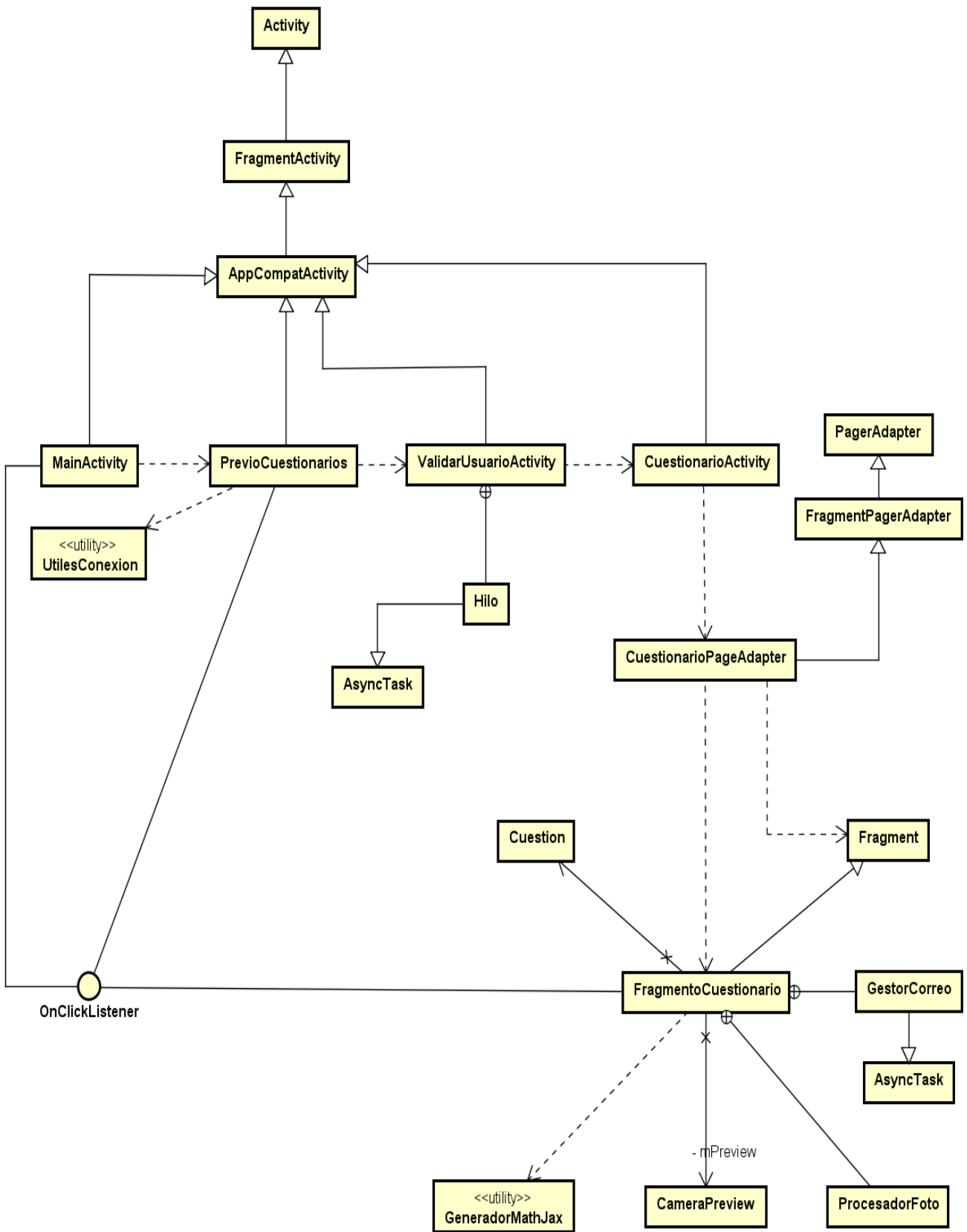


Ilustración 26: Modelo de diseño de los casos de uso "Realizar cuestionarios de examen" y "Realizar cuestionarios de autoevaluación"

La ilustración 26 muestra las clases de diseño del caso de uso “Realizar cuestionarios de examen”. En este diagrama están contenidas clases para mostrar las preguntas y corregirlas (CuestionarioPageAdapter, FragmentoCuestionario y Cuestión), clases para tomar y procesar las fotografías (CameraPreview y ProcesadorFoto), la clase GestorCorreo para enviar los resultados, y ValidarUsuarioActivity para que el usuario valide su identidad entre otras.

El caso de uso “Realizar cuestionario de Autoevaluación” utiliza las mismas clases para realizar el cuestionario, pero no incluye clases para tomar fotografías o comunicarse con un servidor de correo.

Descargar contenidos.

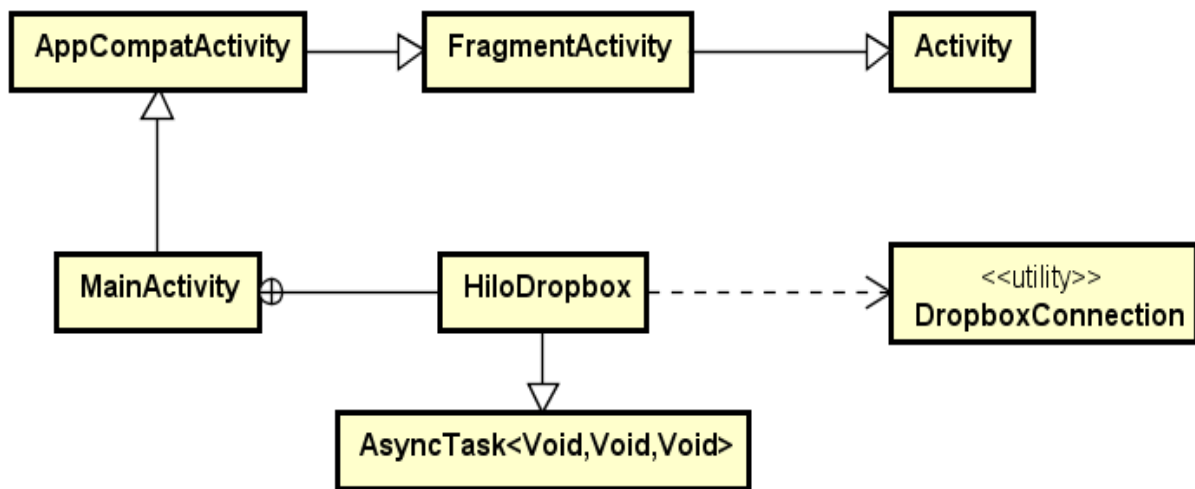


Ilustración 27: Modelo de diseño del caso de uso “Descargar contenidos”

La figura 27 muestra las clases encargadas de borrar los contenidos actuales y descargar los nuevos. Dicho de otra forma, las clases que intervienen en el caso de uso “Descargar contenidos”.

La utilidad DropboxConnection es la encargada de eliminar todos los contenidos anteriores que se encuentren en la ubicación de la teoría y en la de los cuestionarios. Esta clase también es la que descarga los contenidos del repositorio.

La clase HiloDropbox, que es quien llama al método de actualización de DropboxConnection, es un descendiente de AsyncTask, en definitiva, un hilo. Como ya hemos dicho, todas las operaciones de red deben hacerse en un segundo hilo para no bloquear la interfaz.

Generar cuestionario (Aplicación complementaria)

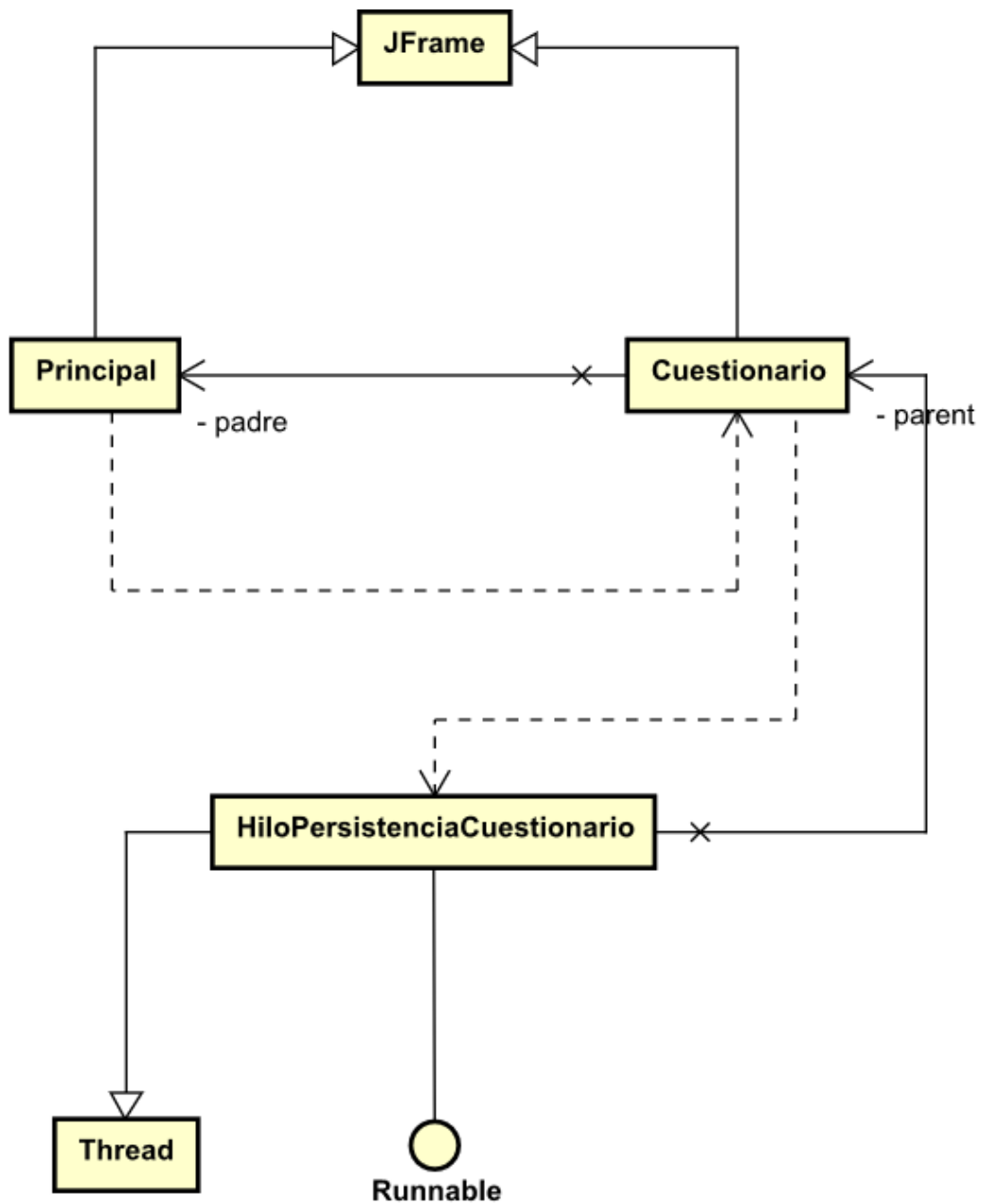


Ilustración 28: Modelo de diseño del caso de uso "Generar cuestionario"

La ilustración 28 nos muestra las clases de diseño implicadas en el caso de uso "Generar cuestionario". La clase Cuestionario (independiente de la clase Cuestionario de los casos de uso de realización de cuestionarios) soporta las operaciones de creación, edición y guardado de los cuestionarios. La clase HiloPersistenciaCuestionario resuelve la ubicación de guardado.

Generar archivo de teoría (Aplicación complementaria).

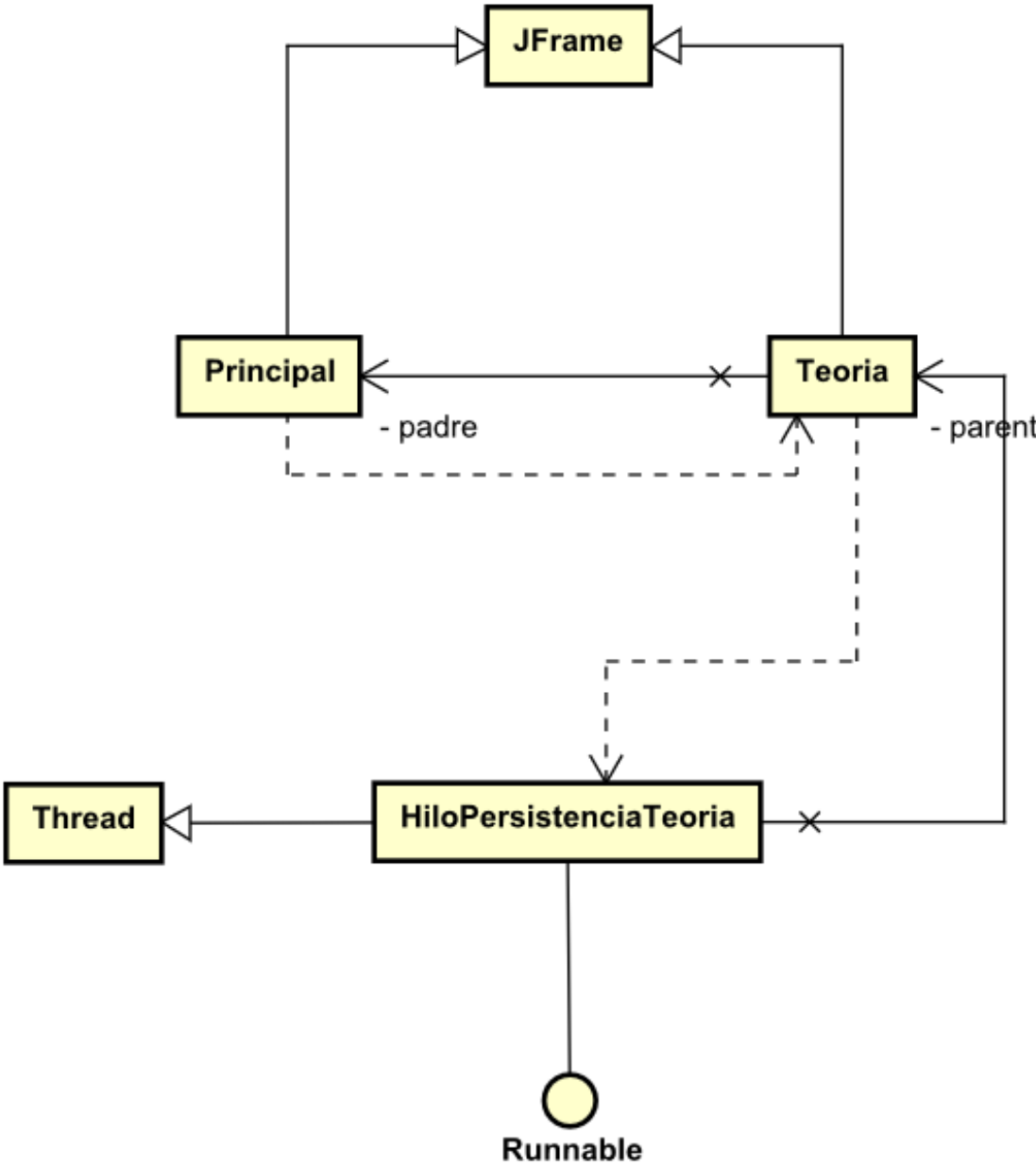


Ilustración 29: Modelo de diseño del caso de uso "Generar archivo de teoría"

De la misma manera que ocurre en el caso de uso anterior, la clase Teoria de la aplicación complementaria no es la misma que la del caso de uso "Mostrar fundamentos teóricos". Esta clase es la que soporta las operaciones de edición, creación y guardado de ficheros de teoría. La clase HiloPersistenciaTeoria es la encargada de resolver la ubicación de guardado.

## Explicación de las clases de diseño.

En esta sección se muestran en profundidad las clases de diseño. En los siguientes diagramas veremos sus atributos y operaciones y una breve descripción de su estructura

### MainActivity.

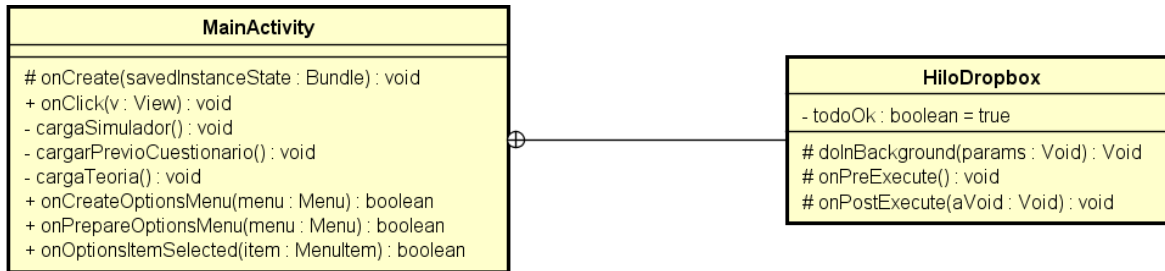


Ilustración 30: Clase MainActivity

La clase MainActivity descende de AppCompatActivity e implementa View.OnClickListener. Es la actividad que se lanza al abrir la aplicación. Contiene los métodos necesarios para lanzar las actividades de cuestionarios, teoría y simulación y responder a las opciones del menú que contiene.

Contenida en MainActivity está la clase HiloDropbox, que es una clase descendiente de AsyncTask. Esta clase permite generar un hilo secundario para realizar operaciones de red.

### DropboxConnection.

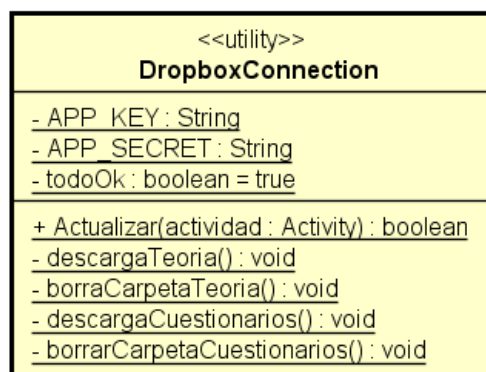


Ilustración 31: Clase DropboxConnection

La utilidad DropboxConnection es la encargada de realizar la actualización de contenidos, tanto páginas de teoría como preguntas de cuestionarios.

Esta clase primero borra los contenidos actuales y después realiza la descarga de contenidos y los coloca en sus respectivas carpetas. En caso de no existir la carpeta “teoría” o la carpeta “cuestionarios” las genera.



TeoriaActivity.



Ilustración 32: Clase TeoriaActivity

Clase que desciende de AppCompatActivity. Su función es la de crear el TeoriaPagerAdapter (podemos pensar en el como el libro que contiene la teoría) indicándole las páginas que debe contener.

TeoriaPagerAdapter.

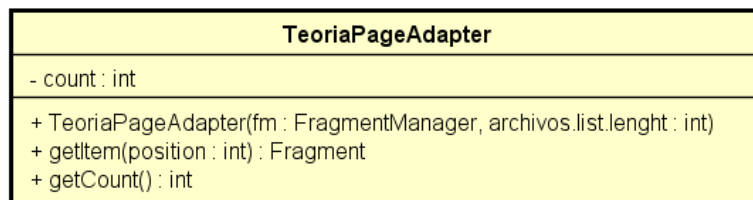


Ilustración 33: TeoriaPagerAdapter

Clase de descendiente de FragmentPagerAdapter. Es quien genera cada uno de los FragmentoTeoria y quien soporta los desplazamientos entre las distintas páginas. La variable count indica el número de Fragmentos que referencia.

FragmentoTeoria.

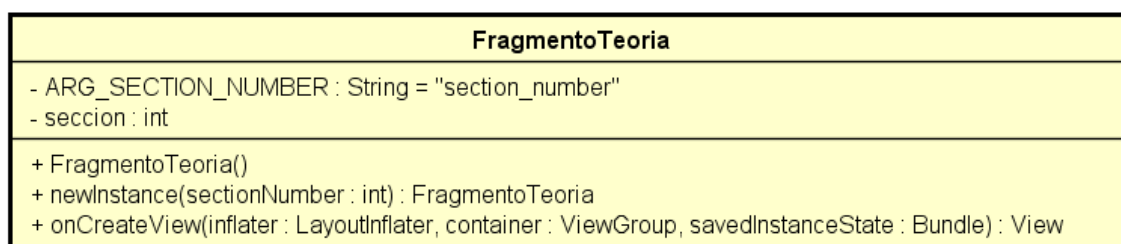


Ilustración 34: Clase FragmentoTeoria

La clase FragmentoTeoria desciende de Fragment. Una instancia de esta clase corresponde con una página de teoría a mostrar. En la creación de su vista (onCreateView) accede a la memoria externa, toma la dirección de la página de teoría que le corresponde mostrar utilizando el atributo "sección" y se la pasa al WebView con el que se mostrará el contenido. El argumento ARG\_SECTION\_NUMBER permite almacenar el número de sección en el Bundle para que permanezca si se ejecuta un onDestroy (por ejemplo, al girar el dispositivo).

PrevioCuestionario.

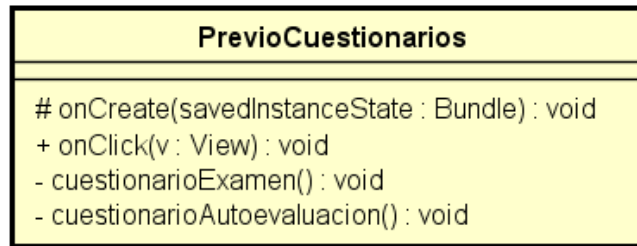


Ilustración 35: Clase PrevioCuestionario

Clase que desciende AppCompatActivity e implementa View.OnClickListener. Su función es permitir elegir entre cuestionarios de examen o cuestionarios de autoevaluación. Antes de lanzar cualquier tipo de cuestionario comprueba que existan contenidos para mostrar.

UtilesConexion.

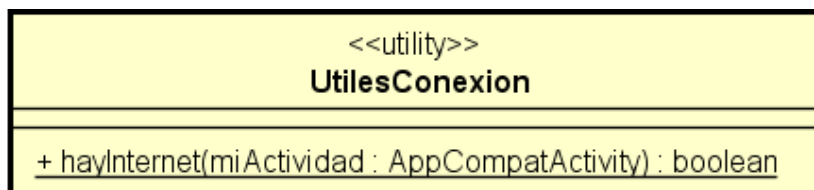


Ilustración 36: Clase UtilesConexion

Clase utilidad que comprueba la disponibilidad de una conexión a internet. Comprueba tanto si hay Wifi como si se disponen de datos móviles.

ValidarUsuario.

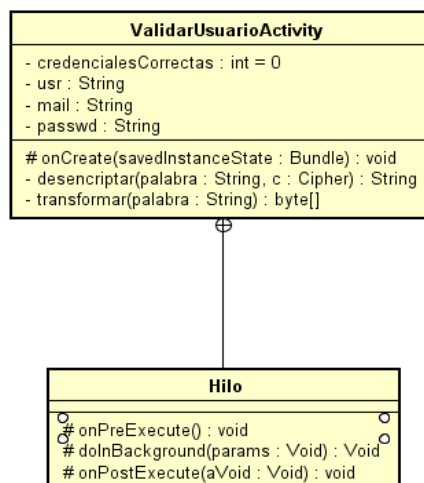


Ilustración 37: Clases ValidarUsuarioActivity e Hilo

Son las clases encargadas de comprobar que el usuario que realiza un cuestionario de examen es alumno de la Universidad de Valladolid.

La clase ValidarUsuarioActivity es la actividad que gestiona la interacción del usuario con la interfaz de usuario de la actividad. Una vez toma los datos se los pasa a la clase Hilo que es quien realiza la comprobación.

Hilo es una clase descendiente de AsyncTask que es quien se comunica con el servidor de correo. La razón de hacer esta comunicación mediante un hilo vuelve a ser la imposibilidad de realizar operaciones de red en el hilo principal. Si la comunicación es correcta y se selecciona la opción de almacenar claves, estas son encriptadas y almacenadas.

CuestionarioActivity.

<b>CuestionarioActivity</b>
- NUMERO_PREGUNTAS : int = 10 - mViewPager : ViewPager
# onCreate(savedInstanceState : Bundle) : void - cuestionariosElegidos(cuestionarios : int) : int[] + onKeyDown(keyCode : int, event : KeyEvent) : boolean - getCameraInstance() : Camera - esconderBarraNotificacion() : void

*Ilustración 38: Clase CuestionarioActivity*

Esta clase desciende de AppCompatActivity e implementa View.OnClickListener. Se encarga de crear un CuestionarioPagerAdapter e indicarle el identificador de cada una de las preguntas que debe mostrar, el tipo de comportamiento que tiene que tener (examen o autoevaluación), y en caso de ser un examen le da una instancia de la cámara. Además, esconde la barra de notificación para que no interfiera con el desarrollo del test.

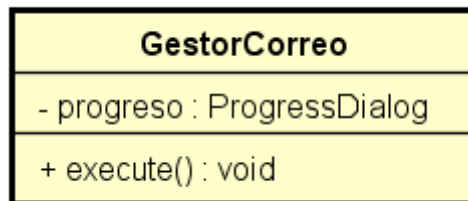
CuestionarioPagerAdapter.

<b>CuestionarioPageAdapter</b>
- numPaginas : int - cuestionariosElegidos : int[] - tipoPrueba : int - inicio : long - user : String - password : String - camara : Camera
+ CuestionarioPageAdapter(fm : FragmentManager, elegidos : int[], tipo : int, user : String, password : String, cameraInstance : Camera) + getItem(position : int) : Fragment + getItemPosition(object : Object) : int + getCount() : int

*Ilustración 39: Clase CuestionarioPageAdapter*

La clase CuestionarioPageAdapter descende de FragmentPagerAdapter. Se encarga de crear cada una de las páginas (fragmentos) que contienen los cuestionarios. A cada una de las páginas que genera, les entrega el identificador de la pregunta que debe contener y la instancia de la cámara. En caso de ser un examen, a la última página se le pasan los datos de acceso para generar el correo, en vez de lo anterior. También calcula el momento en que comienza el cuestionario.

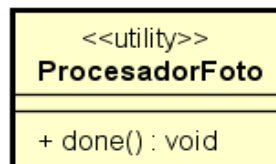
GestorCorreo.



*Ilustración 40: Clase GestorCorreo*

Es una clase descendiente de AsyncTask, su función es la de enviar el correo electrónico con los resultados del examen al profesor. La razón de usar otro hilo es la misma que el caso de Hilo de ValidarUsuarioActivity, no realizar operaciones de red en el hilo principal.

ProcesadorFoto.



*Ilustración 41: Clase ProcesadorFoto*

Una vez se ha tomado la fotografía, esta clase utilidad es la responsable de generar el mapa de bits y realizar posteriormente la rotación.

## FragmentoCuestionario.

FragmentoCuestionario
<pre>- ARG_SECTION_NUMBER : String = "section_number" - EXAMEN : int = 2 - PREGUNTA_PENDIENTE : int = 1 - PREGUNTA_ACERTADA : int = 2 - PREGUNTA_FALLADA : int = 3 - PANTALLA_CORRECCION : int = 11 - PREGUNTA : int = 1 - tempdata : byte[] - seccion : int - miCuestionario : int - elegida : int = 1 - aciertos : int[] - tipoPrueba : int - corregido : boolean - numAciertos : int - inicio : long - elegidos : int[] - tiempo : long - tomada : boolean = false - seccionFoto : int - cargaCamara : boolean = false - user : String - pwd : String - urlFoto : String</pre>
<pre>+ FragmentoCuestionario() + newInstance(sectionNumber : int, elegidos : int[], seleccionado : int, aciertos : int[], tipo : int, inicio : long, foto : int, user : String, password : String, camera : Camera) : FragmentoCuestionario + onCreateView(inflater : LayoutInflater, container : ViewGroup, savedInstanceState : Bundle) : View - generalInterfazCuestionario(rootView : View) : void - funcionRadio() : void - generaFuncionesRadio() : void - preparaPregunta(rootView : View) : void - preparaElementosRespuesta(rootView : View) : void - adaptaElementosRespuesta() : void - cargaVistasRespuestas(rootView : View) : void - botonSalir(rootView : View) : void - muestraTiempo(rootView : View) : void - calculaTiempoTotal() : void - preparaRespuesta() : void - generaResultadoSeccionActual() : void - comprobarFinCuestionario() : boolean - bloqueoCuestion() : void - preparaCamara(rootView : View) : void - cargarCamara(rootView : View) : void - done() : void - cargaCuestionario() : void - muestraFin(acierto : int) : AlertDialog - generadorMathJax(texto : String, campo : int) : String + onClick(v : View) : void</pre>

Ilustración 42: Clase FragmentoCuestionario

Esta clase es descendiente de Fragment e implementa View.OnClickListener. Si es una página que contiene una pregunta, recibirá la instancia de la cámara y el identificador del cuestionario que debe descargar del CuestionarioPagerAdapter que la crea, si es la página final de un examen, recibe los datos de contacto para generar el correo.

Cada una de la paginas que muestran una pregunta utilizan el identificador recibido para acceder a la memoria externa y tomar la información necesaria para generar un objeto Cuestión. Esta información contiene, las respuestas, la pregunta y la respuesta correcta.

En el momento de responder, la funcionRadio() calcula si la respuesta es correcta o no y modifica el campo correspondiente en el vector aciertos. Si estamos en una autoevaluación se mostrará el resultado y al responder las diez obtendremos un resumen del intento.

Si el cuestionario es de examen el comportamiento es algo distinto. Al responder, si el número de página (sección) corresponde con el de seccionFoto, la página llama a la cámara para que tome la fotografía y después corrige la pregunta, aunque no muestra el resultado. Una vez se responden todas, se va a la última página que tiene un botón de corrección. Este botón activa la función comprobarFinCuestionario que comprueba que todo este respondido y toma los datos del vector aciertos para mostrar los resultados. Una vez hecho esto, se muestra la foto, los resultados y se ejecuta el hilo GestorCorreo para que envíe los resultados.

GeneradorMathJax.

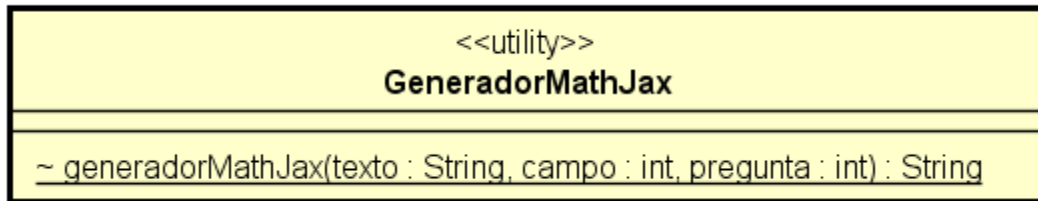


Ilustración 43: Clase GeneradorMathJax

Clase utilidad cuya función es resolver el texto en TEX de un WebView. El valor “pregunta” le permite distinguir si debe generar texto para una pregunta o una respuesta. En caso de que la utilidad no encuentre texto para resolver (no hay texto en TEX encerrado entre “\$”) la función simplemente retorna.

SimulacionActivity.

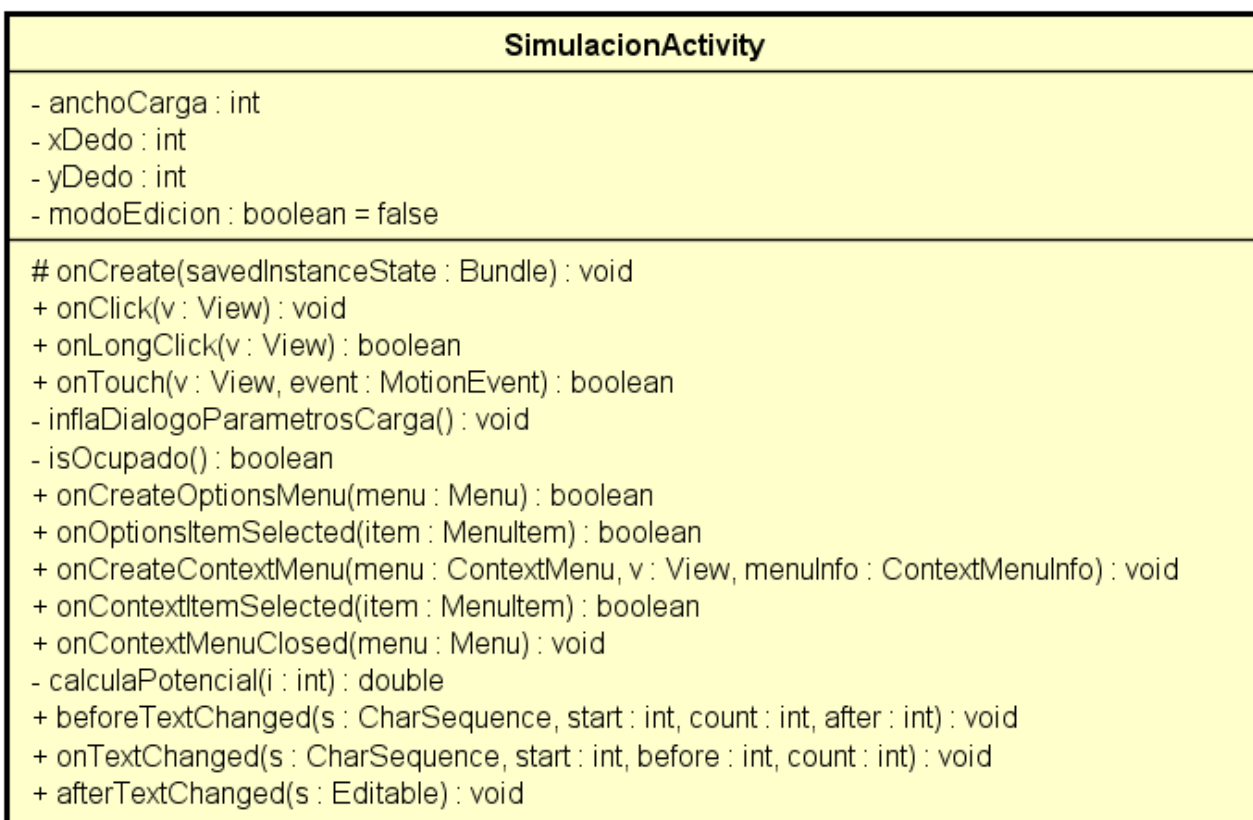


Ilustración 44: Clase SimulacionActivity

Esta clase descende de AppCompatActivity e implementa View.OnClickListener, View.OnLongClickListener, View.OnTouchListener y TextWatcher. Se encarga de la simulación, es decir, crear, modificar y eliminar las cargas, llamar a la utilidad que dibuja líneas y calcular el potencial en un punto.

Cuando se pulsa sobre un punto de la pantalla, el método `OnClick` ofrece la posibilidad de calcular el potencial (función `calculaPotencial`) o añadir una carga (función `inflaParametrosCarga`), al hacer esto segundo, se llama a la utilidad `DibujaLineas` para que muestre las líneas de campo. Para evitar que una carga tome un valor incorrecto se implementa `TextWatcher` que añade eventos que se ejecutan antes de introducir el texto, en el momento de hacerlo y en el instante posterior, lo que nos permite controlar la introducción.

Si se deja una carga pulsada, el método `OnLongClickListener` hace que se muestre un menú que permite elegir entre editar la carga, borrarla o conocer sus parámetros. Para realizar esto y lo descrito en el párrafo anterior, es necesario conocer que parte de la pantalla se ha pulsado, de ahí la necesidad de implementar `OnTouchListener`.

Carga.

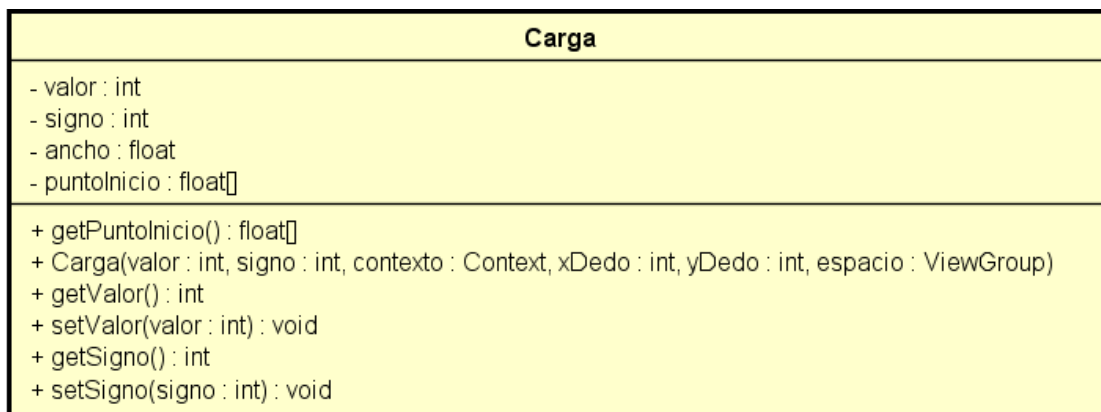


Ilustración 45: Clase Carga

Esta clase desciende de `ImageView` (elemento de interfaz gráfica) define las cargas de la simulación. Los objetos cargas almacenan el valor de la carga como un entero positivo y el signo como un entero que puede ser -1 o 1 lo que facilita operar. La lista `puntolnicio` almacena los puntos en los que deben comenzarse a dibujar las líneas de campo.

HiloDibujo

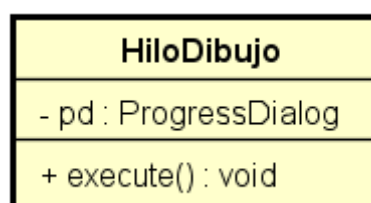


Ilustración 46: Clase HiloDibujo

La clase `HiloDibujo`, descendiente de `AsyncTask`, es la responsable de llamar a las utilidades que dibujan las líneas de campo. El cálculo de las líneas de campo puede requerir algunos segundos hasta completarse. Durante este tiempo no podemos tener bloqueada la pantalla, por lo que recurrimos a un segundo hilo y mostramos un dialogo de carga.

Dibujalineas.

<b>&lt;&lt;utility&gt;&gt; Dibujalineas</b>
- D : float = (float)5 - inicio : float[] - puntos : ArrayList<Float>
~ dibujar(cargas : ArrayList<Carga>, lineas : Lineas, ancho : int) : void - calculaOrigenes(q : Carga) : int - calculaDeltaY(campoTotalX : double, campoTotalY : double) : float - calculaDeltaX(campoTotalX : double, campoTotalY : double) : float - sumaCampoY(q : Carga, campo : float, angulo : float, inicio : float[], ancho : int) : float - sumaCampoX(q : Carga, campo : float, angulo : float, inicio : float[], ancho : int) : double - calculaCampo(carga : Carga, inicio : float[], ancho : int) : float - calculaAngulo(carga : Carga, inicio : float[], ancho : int) : float

Ilustración 47: Clase Dibujalineas

Esta clase utilidad es la encargada de calcular los puntos que componen cada una de las líneas de campo que deben mostrarse en la simulación. Cuando el método dibujar es invocado, se almacena en inicio los puntos de inicio de la carga de la que se saldrán las líneas, posteriormente, se calculan las coordenadas del punto siguiente en función de los anteriores, el algoritmo lo explicare en detalle en el capítulo “Implementación”. El valor constante D define la distancia entre un punto y el anterior.

Líneas.

<b>Lineas</b>
- lineas : ArrayList<float[]> - borrado : boolean = false
+ Lineas(context : Context) ~ agregaLinea(puntos : float[]) : void # onDraw(canvas : Canvas) : void + reinicio() : void

Ilustración 48: Clase Lineas

Esta clase es la encargada de realizar la creación y el borrado de las líneas de campo, no la representación de la línea en sí. El atributo lineas contiene vectores de puntos que componen cada línea de campo, un vector por línea. Cada vez que se ejecuta agregaLinea, se invalida la vista, lo que hace esta se recargue, en esta recarga, el canvas dibuja las líneas.



Cuestionario (Aplicación complementaria).

<b>Cuestionario</b>
- ruta : String = null
+ Cuestionario() + Cuestionario(aThis : Principal) - initComponents() : void - jMenuItem3ActionPerformed(evt : ActionEvent) : void - jMenuItem4ActionPerformed(evt : ActionEvent) : void - jMenuItem2ActionPerformed(evt : ActionEvent) : void - jMenuItem1ActionPerformed(evt : ActionEvent) : void - jMenuItem7ActionPerformed(evt : ActionEvent) : void - jMenuItem6ActionPerformed(evt : ActionEvent) : void - jMenuItem5ActionPerformed(evt : ActionEvent) : void - jButton1ActionPerformed(evt : ActionEvent) : void - jButton2ActionPerformed(evt : ActionEvent) : void - jButton3ActionPerformed(evt : ActionEvent) : void - jButton4ActionPerformed(evt : ActionEvent) : void - jButton5ActionPerformed(evt : ActionEvent) : void - guardar(file : File) : void - compruebaVacios() : boolean - comprobacion(text : String) : boolean - abrir(file : File) : boolean - previsualizacion(text : String) : void

*Ilustración 49: Clase Cuestionario (Aplicación complementaria)*

Clase que hereda de JFrame. Esta clase contiene todas las funciones necesarias para poder crear y editar un cuestionario. Para crear un cuestionario, escribiremos en la interfaz la pregunta, las respuestas y el número de respuestas correctas. En el momento de guardar los cambios genera un fichero.

HiloPersistenciaCuestionario (Aplicación complementaria).

<b>HiloPersistenciaCuestionario</b>
~ HiloPersistenciaCuestionario(jFileChooser1 : JFileChooser, aThis : Cuestionario) + start() : void ~ getFile() : File

*Ilustración 50: HiloPersistenciaCuestionario (Aplicación complementaria)*

Este hilo permite obtener la ruta del archivo seleccionado o del archivo a guardar que se ha indicado mediante un jFileChooser.

Teoria (Aplicación complementaria).

Teoria
- ruta : String = null
+ Teoria(parent : Principal) - initComponents() : void - jMenuItem7ActionPerformed(evt : ActionEvent) : void - jMenuItem6ActionPerformed(evt : ActionEvent) : void - abrir(file : File) : boolean - jMenuItem1ActionPerformed(evt : ActionEvent) : void - jMenuItem2ActionPerformed(evt : ActionEvent) : void - jMenuItem3ActionPerformed(evt : ActionEvent) : void - jMenuItem5ActionPerformed(evt : ActionEvent) : void - jButton1ActionPerformed(evt : ActionEvent) : void - jButton2ActionPerformed(evt : ActionEvent) : void - jButton3ActionPerformed(evt : ActionEvent) : void - jButton4ActionPerformed(evt : ActionEvent) : void - jButton5ActionPerformed(evt : ActionEvent) : void - jButtonActionPerformed(evt : ActionEvent) : void - comprobarVacios() : boolean - comprobacion(text : String) : boolean - guardar(file : File) : void - previsualizar() : void + escribeTeoria() : void + copiaAdjuntos(parent : File) : void + eliminaNoDependientes(carpeta : File) : void

Ilustración 51: Clase Teoria (Aplicación complementaria)

Clase que hereda de JFrame. Esta clase contiene todas las funciones necesarias para poder crear y editar una página de teoría. Para crear una página de teoría, agregaremos todas las dependencias, escribiremos la pagina de teoría en HTML y lo exportaremos. En el momento de exportar, se ejecutan la función copiaAdjuntos para que almacene las dependencias en el directorio de la página de teoría y la función eliminaNoDependencias que elimina del directorio de la página todo lo que no sea una dependencia.

HiloPersistenciaTeoria (Aplicación complementaria).

HiloPersistenciaTeoria
~ HiloPersistenciaTeoria(jFileChooser1 : JFileChooser, aThis : Teoria) + start() : void ~ getFile() : File

Ilustración 52: HiloPersistenciaTeoria (Aplicación complementaria)

Este hilo permite obtener la ruta del archivo seleccionado o del archivo a guardar que se ha indicado mediante un jFileChooser.

## Diagramas de secuencia.

En esta sección se muestra como interaccionan las clases de diseño entre ellas para realizar las funciones requeridas a la aplicación. Además de esto, se muestran también las relaciones con los objetos del Framework Android y con las clases contenidas en el JDK de Java.

Consultar fundamentos teóricos.

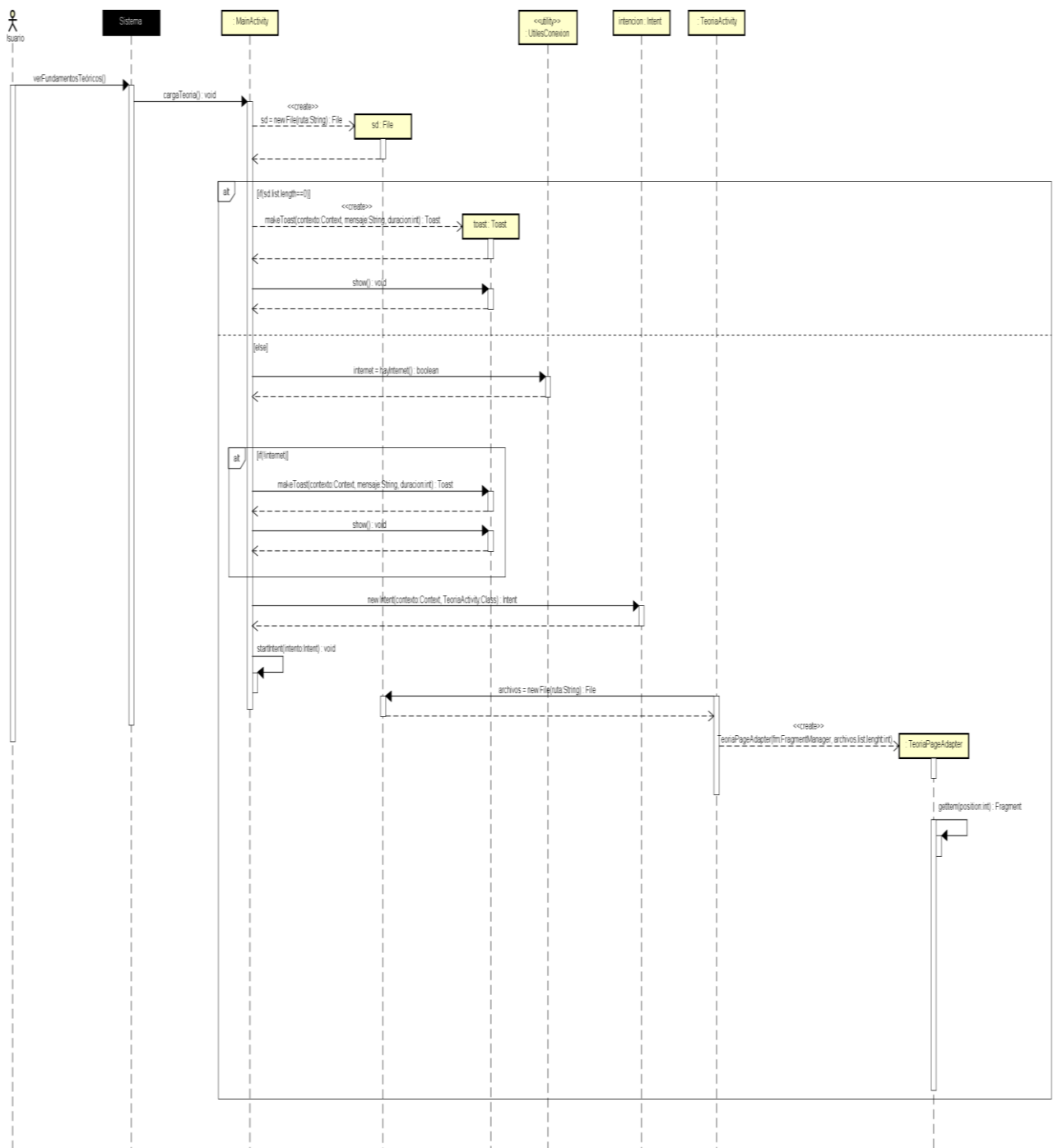


Ilustración 53: Diagrama de Secuencia: Consultar fundamentos teóricos I



# Realizar cuestionario de autoevaluación.

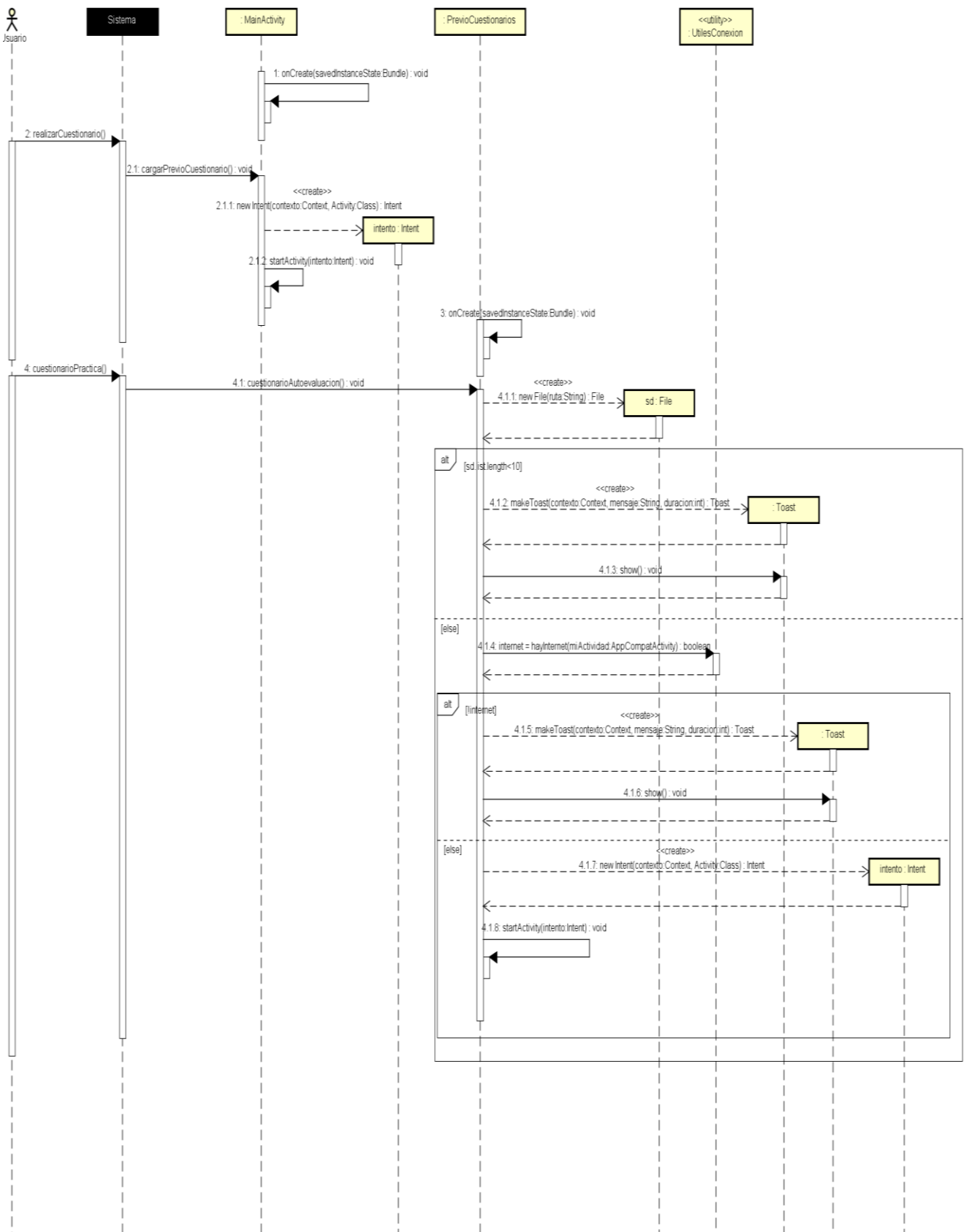


Ilustración 55: Diagrama de secuencia: Realizar cuestionario de autoevaluación I

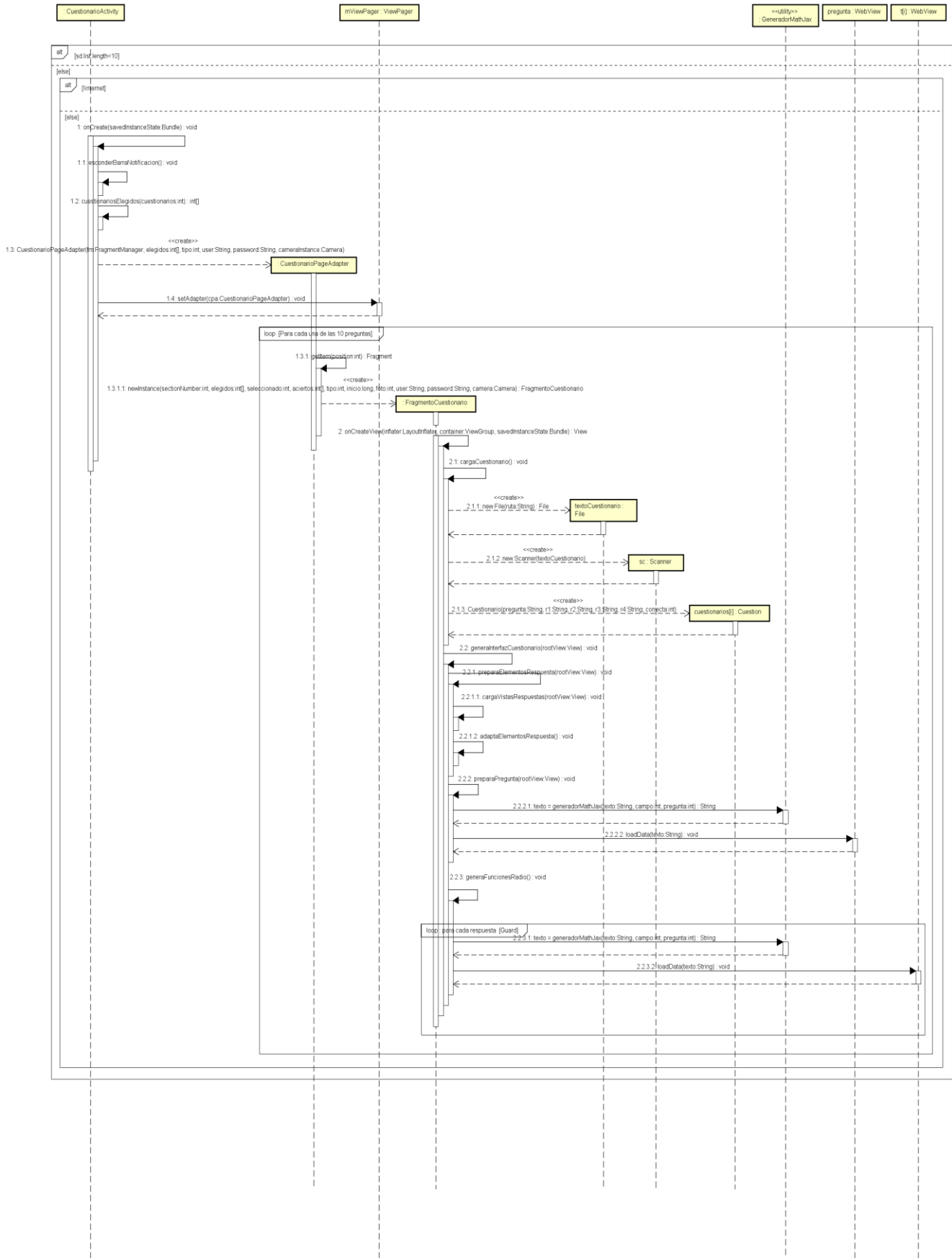


Ilustración 56: Diagrama de secuencia: Realizar cuestionario de autoevaluación II

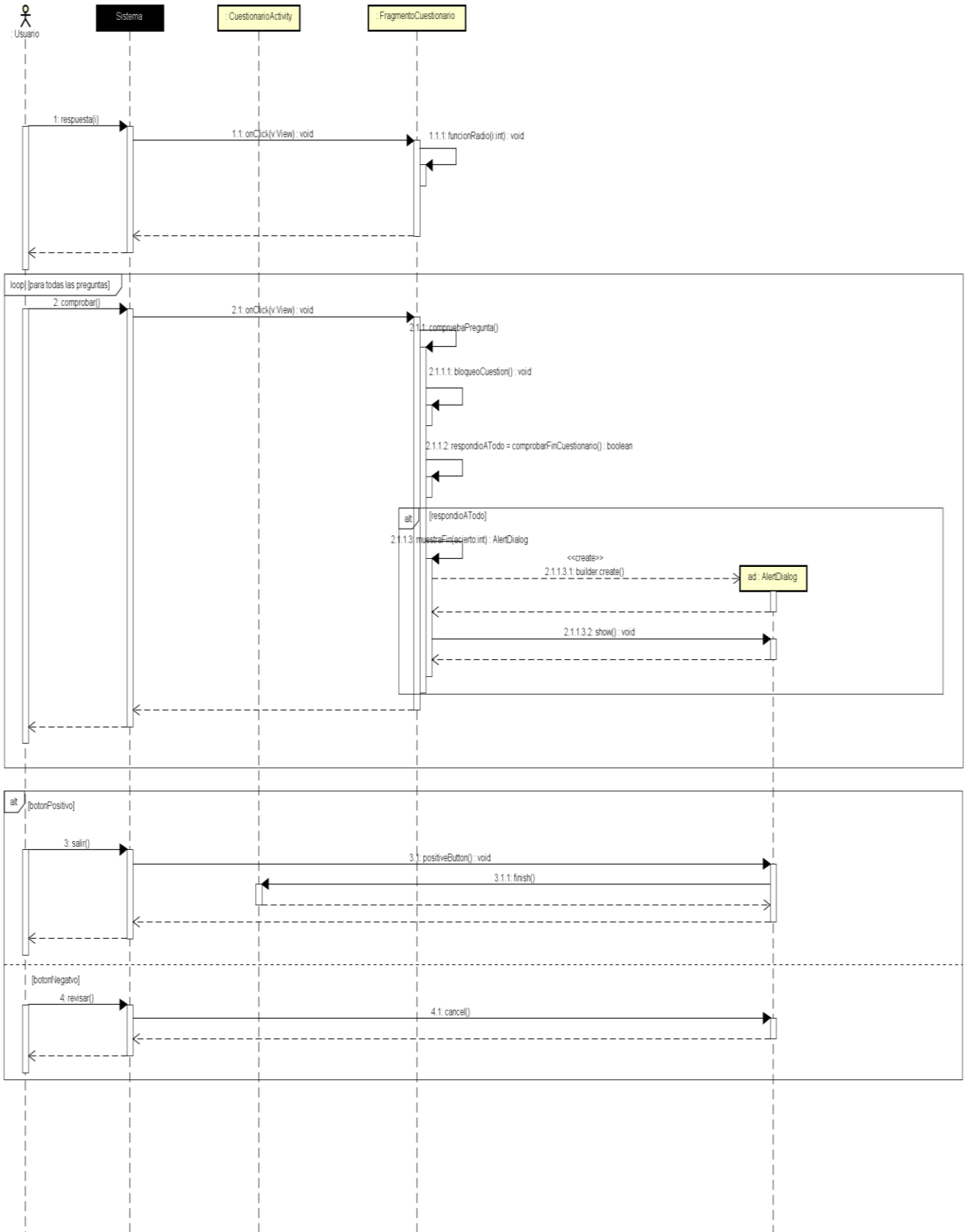


Ilustración 57: Diagrama de secuencia: Realizar cuestionario de autoevaluación III

# Realizar simulación.

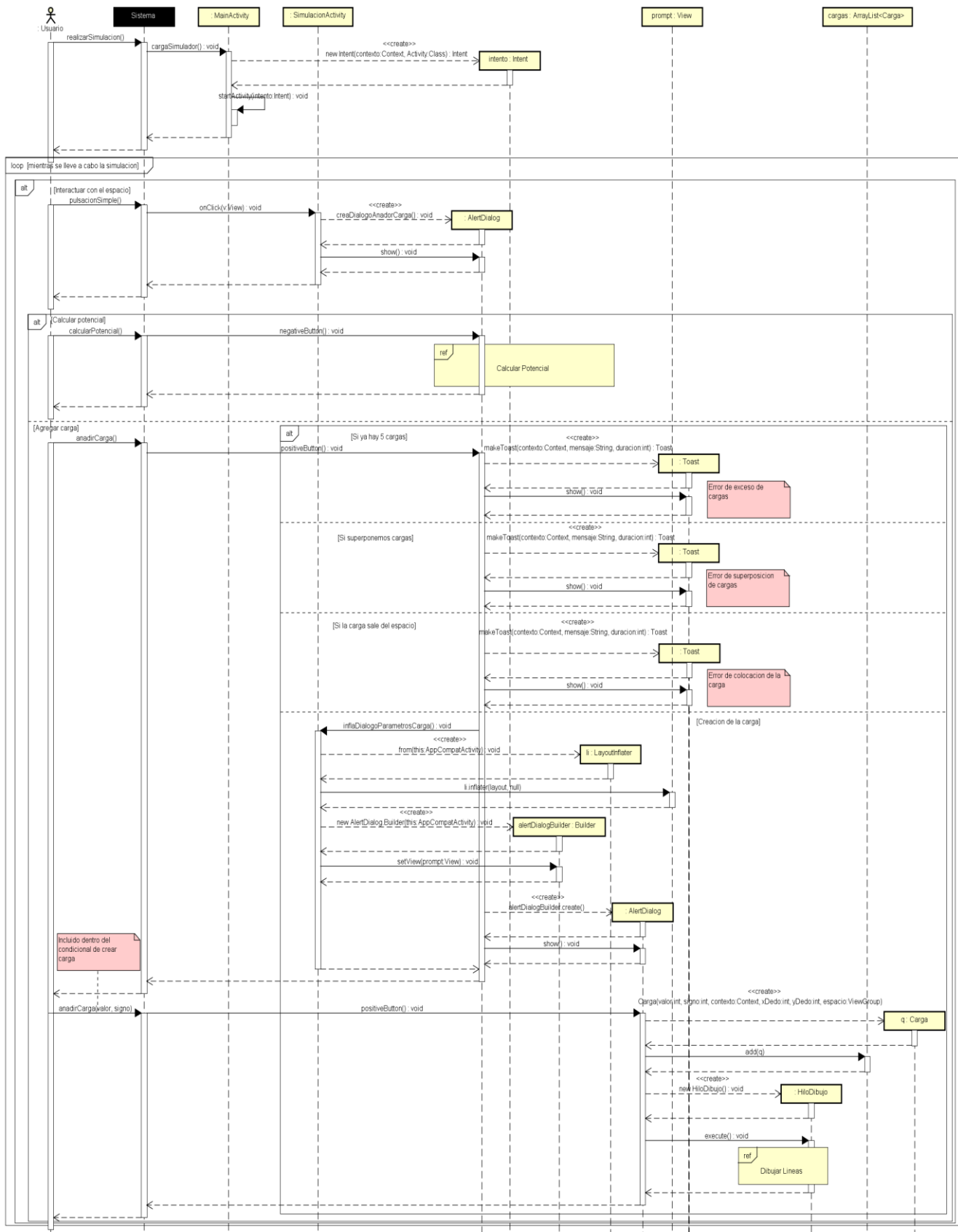


Ilustración 58: Diagrama de secuencia: Realizar simulación



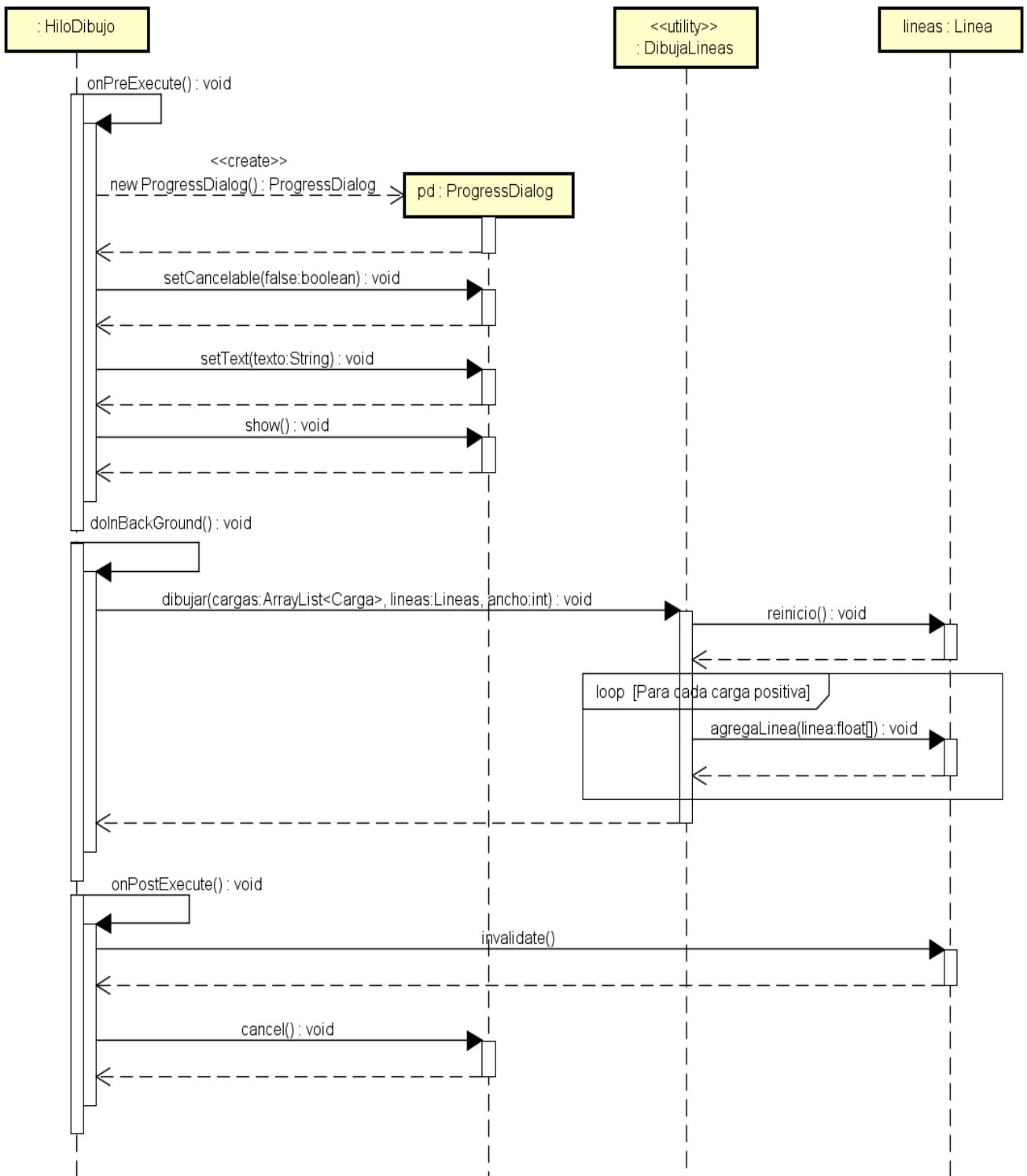


Ilustración 59: Diagrama de secuencia: Dibuja Líneas

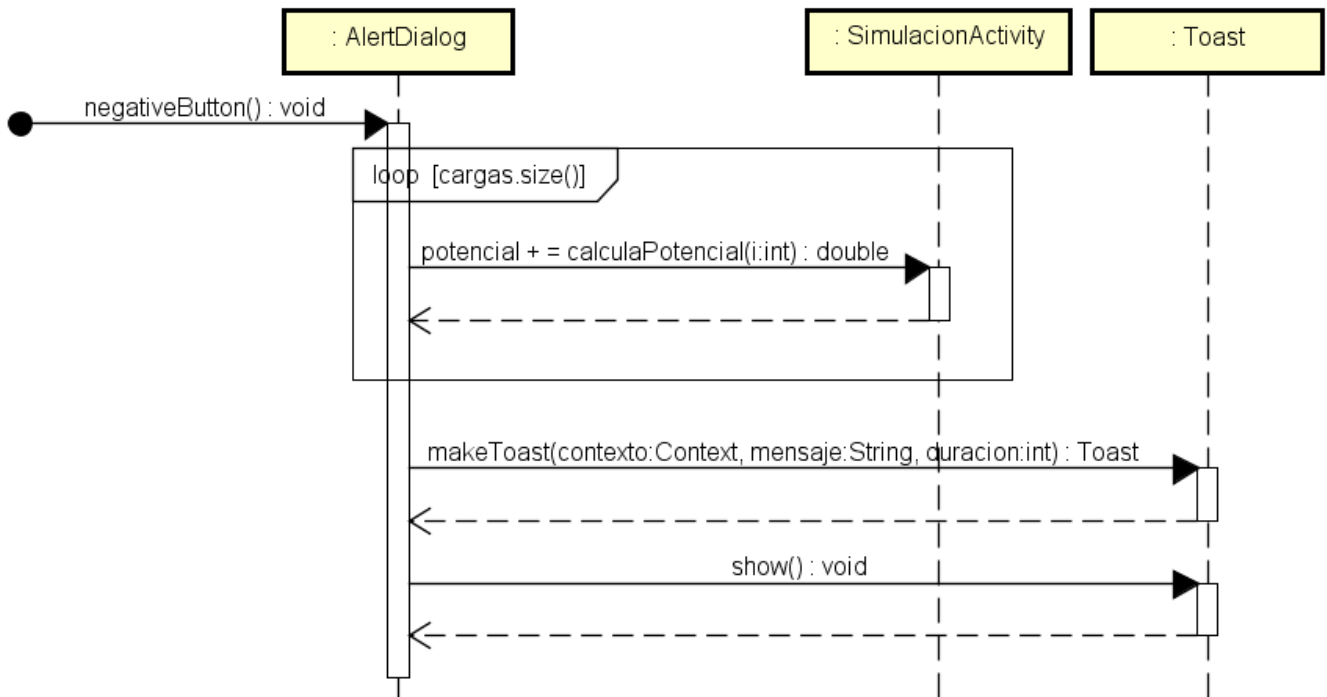


Ilustración 60: Diagrama de Secuencia: Calcular potencial

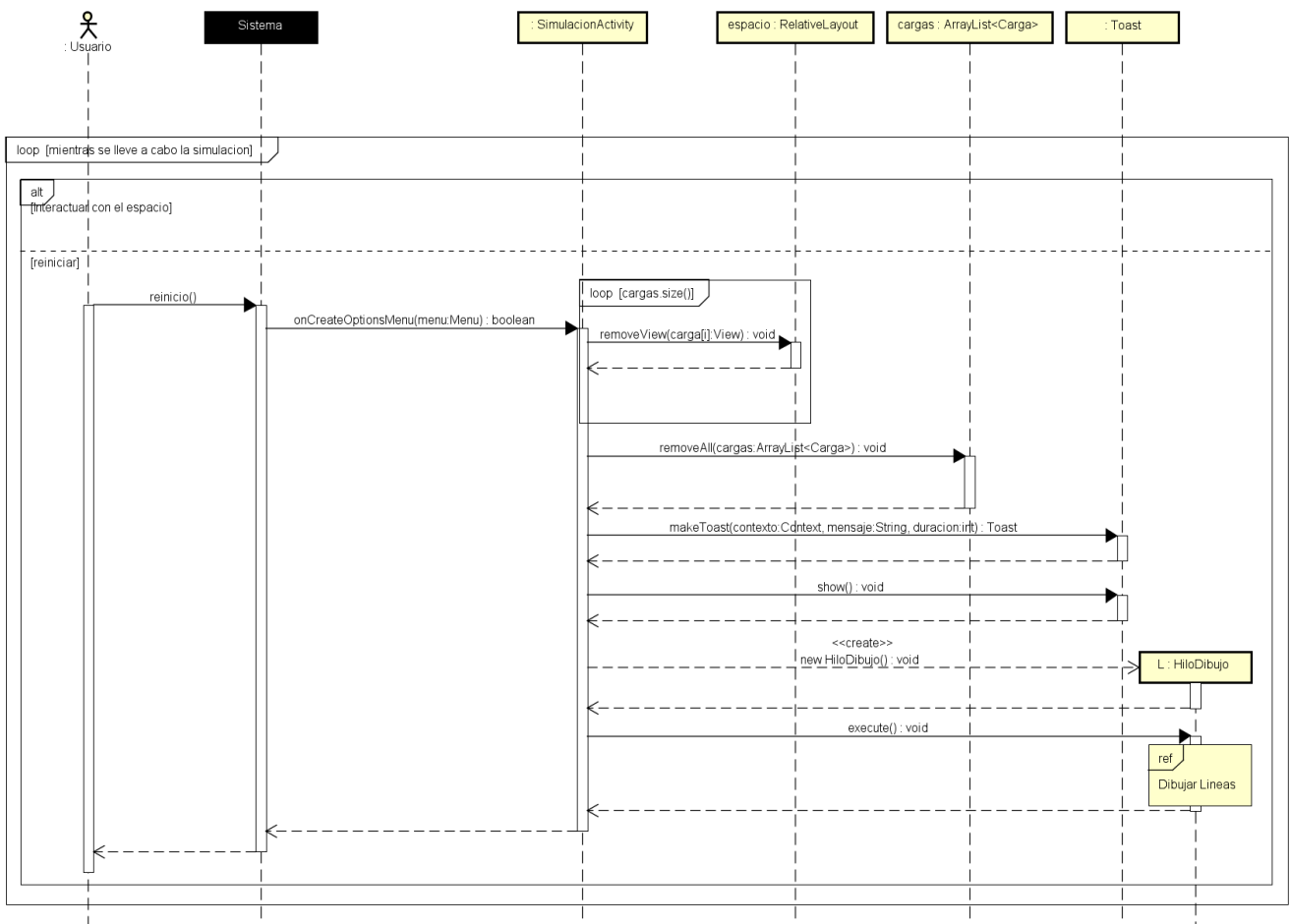


Ilustración 61: Diagrama de secuencia: Reiniciar

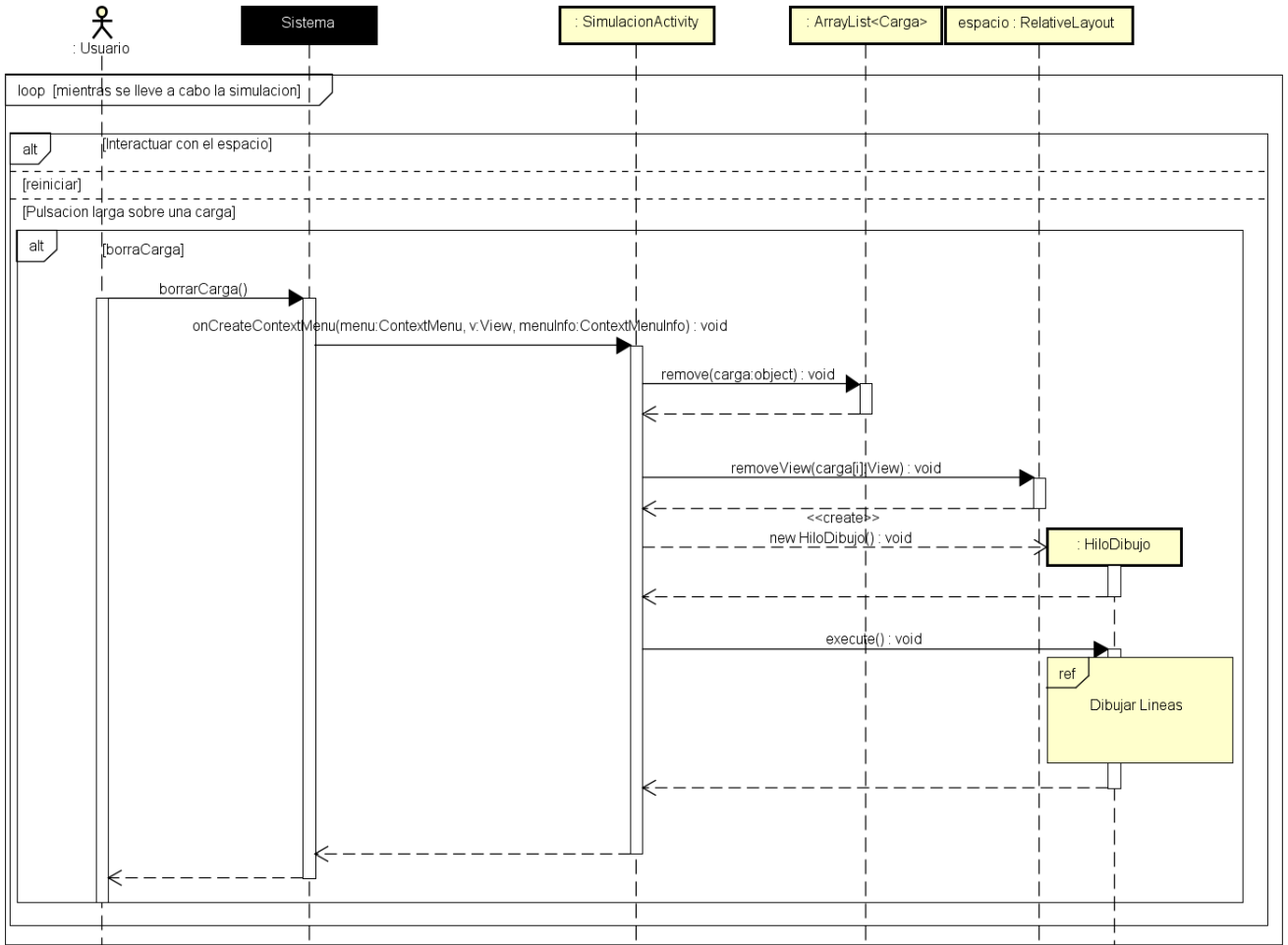


Ilustración 62: Borrar carga

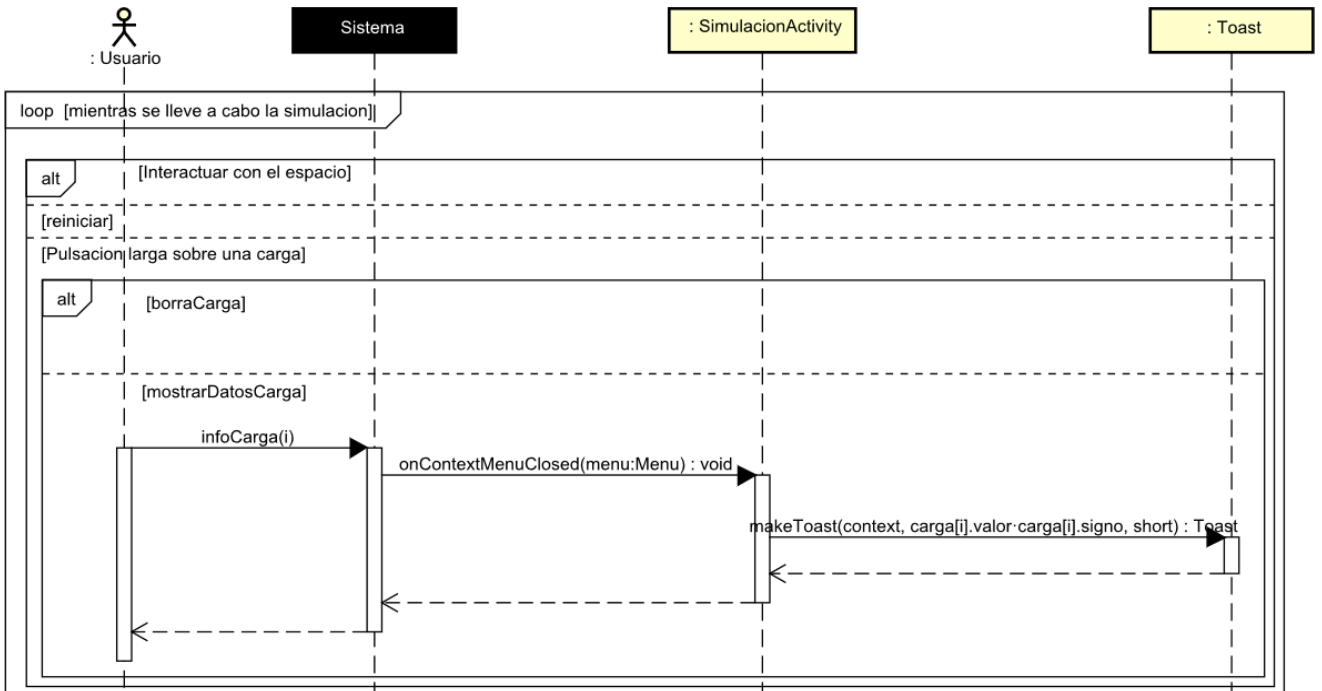


Ilustración 63: Diagrama de Secuencia: Mostrar datos de una carga

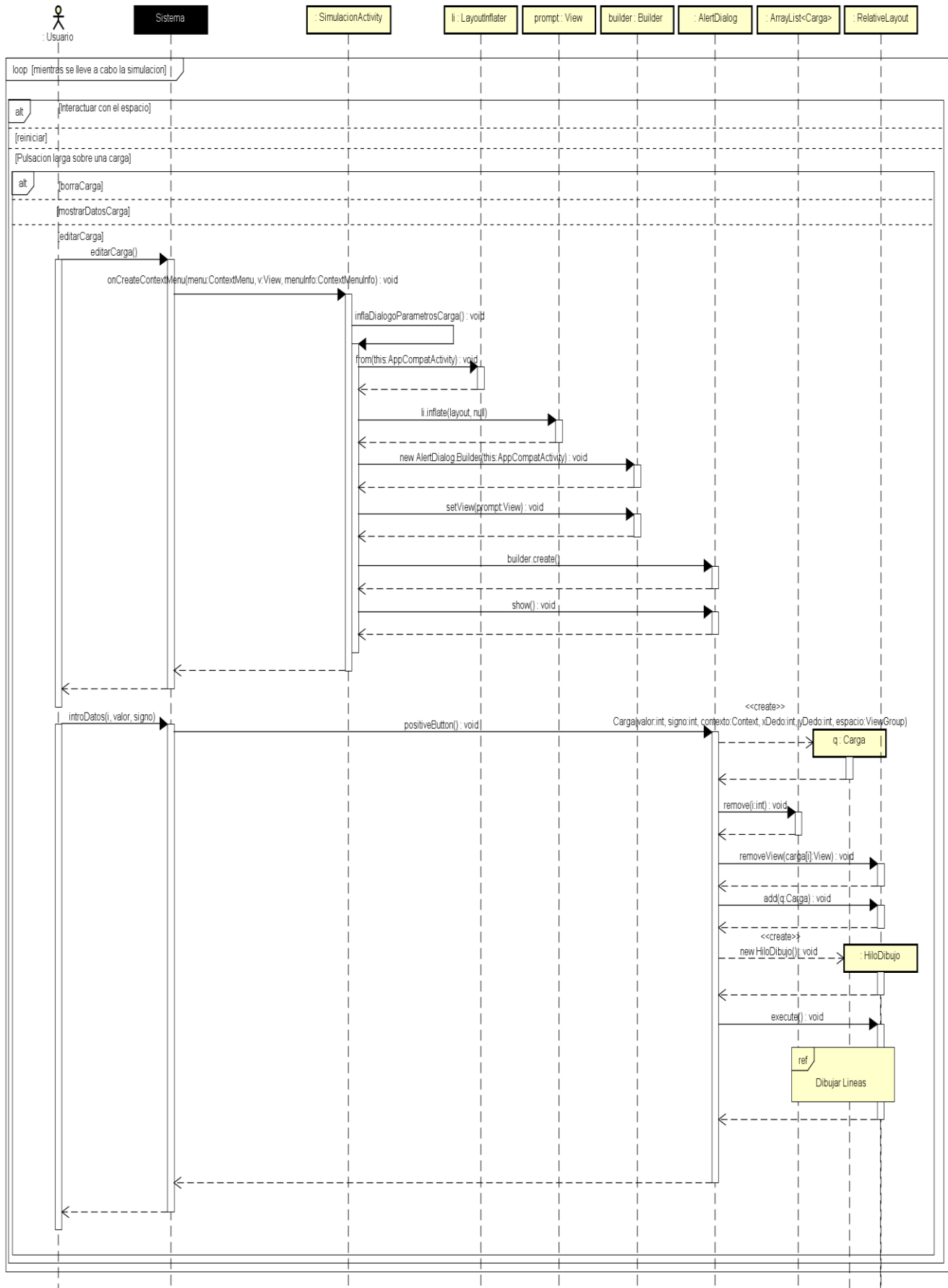


Ilustración 64: Diagrama de Secuencia: Editar Carga

# Generar cuestionario.

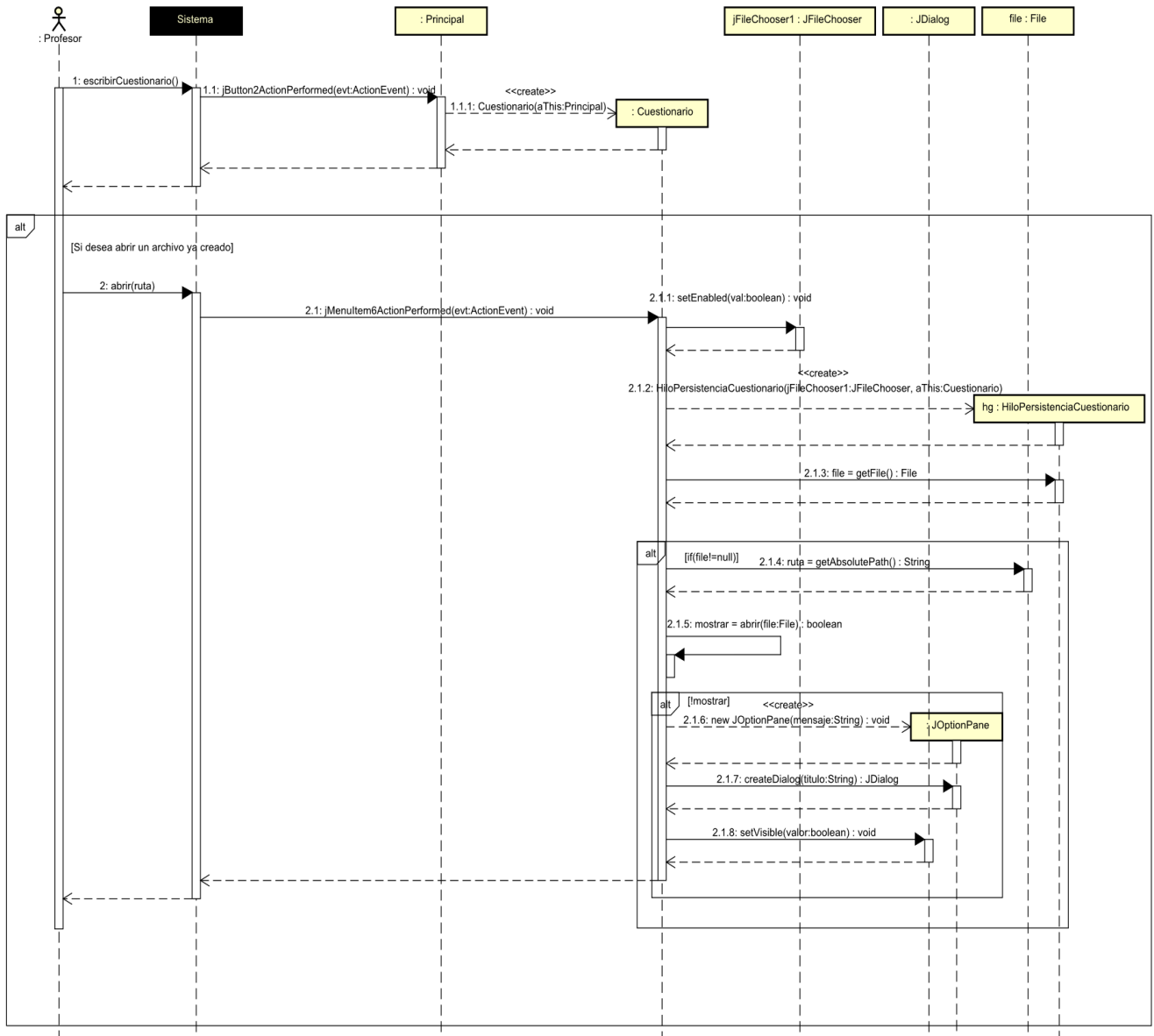


Ilustración 65: Diagrama de secuencia: Generar cuestionario I

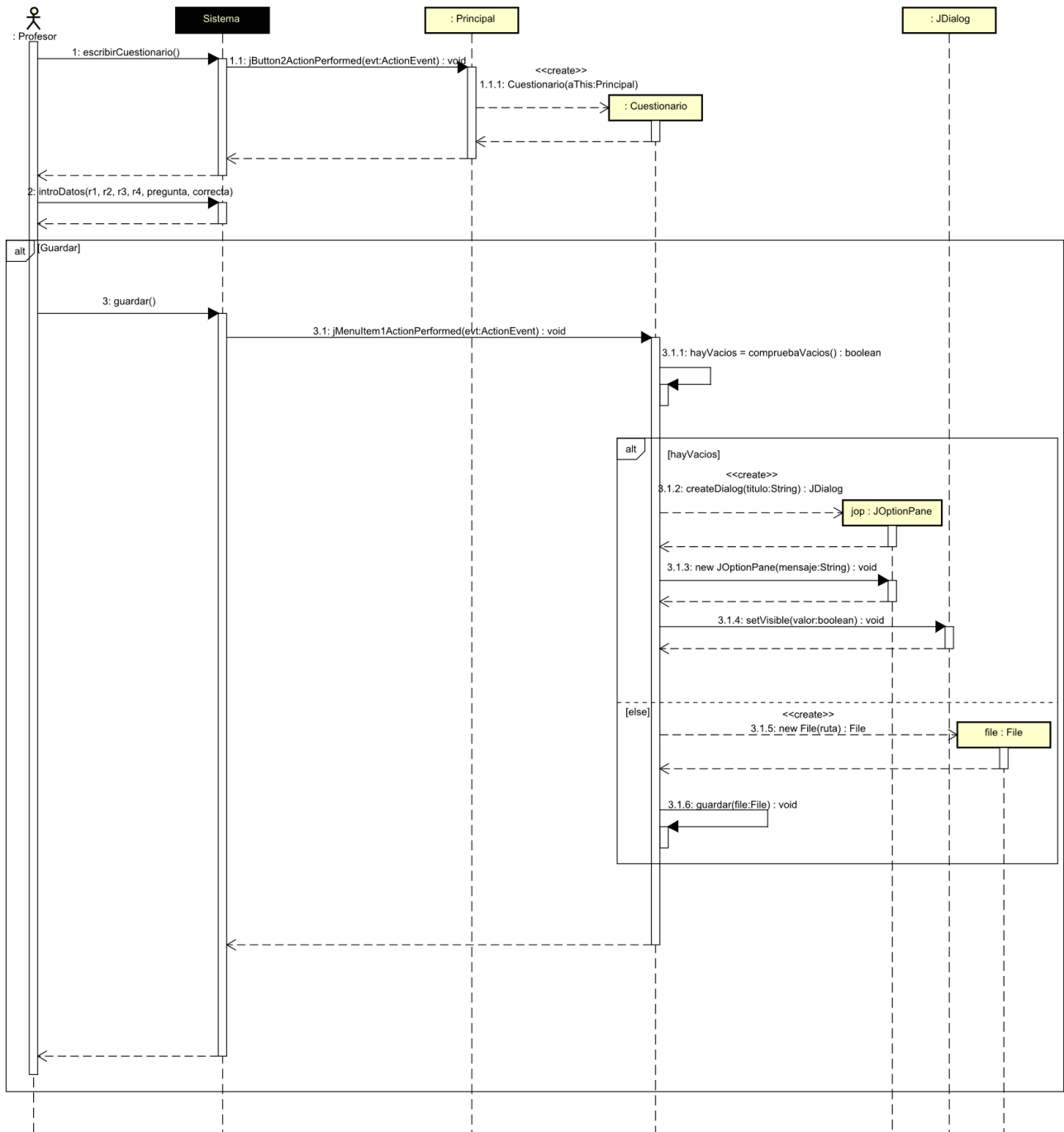


Ilustración 66: Diagrama de secuencia: Generar cuestionario II

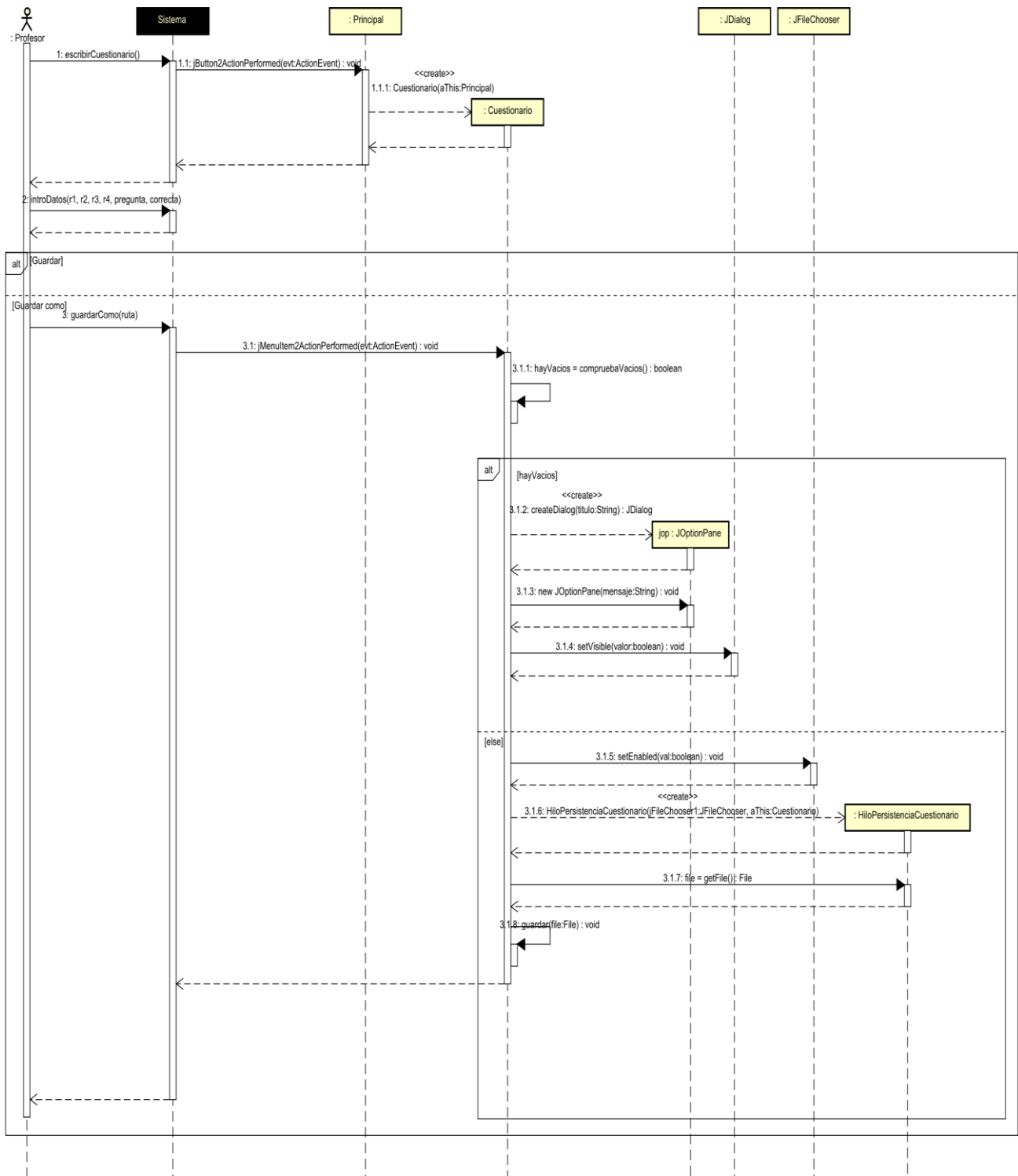


Ilustración 67: Diagrama de secuencia: Generar cuestionario III

# Generar archivo de teoría.

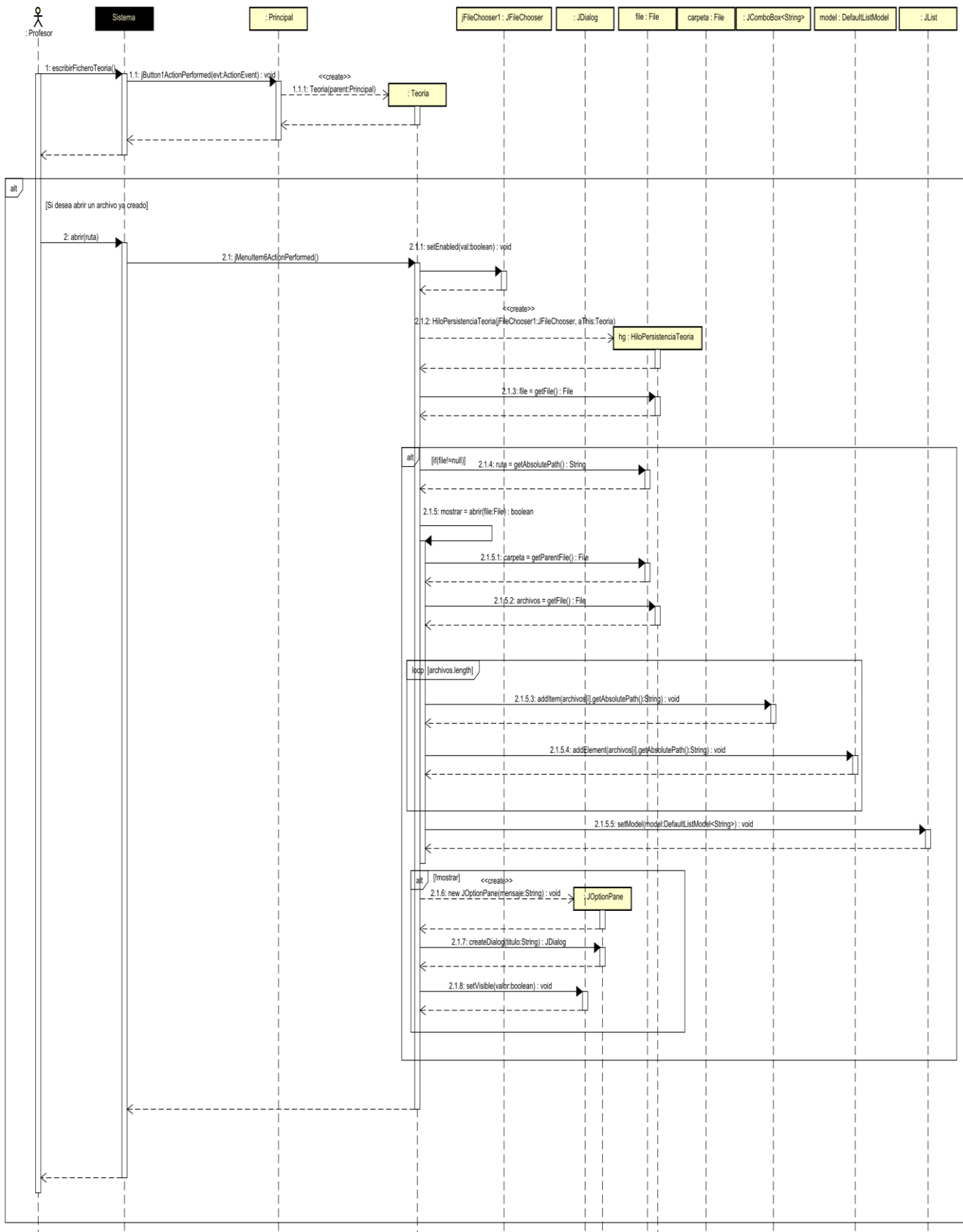


Ilustración 68: Diagrama de secuencia: Generar archivo de teoría I



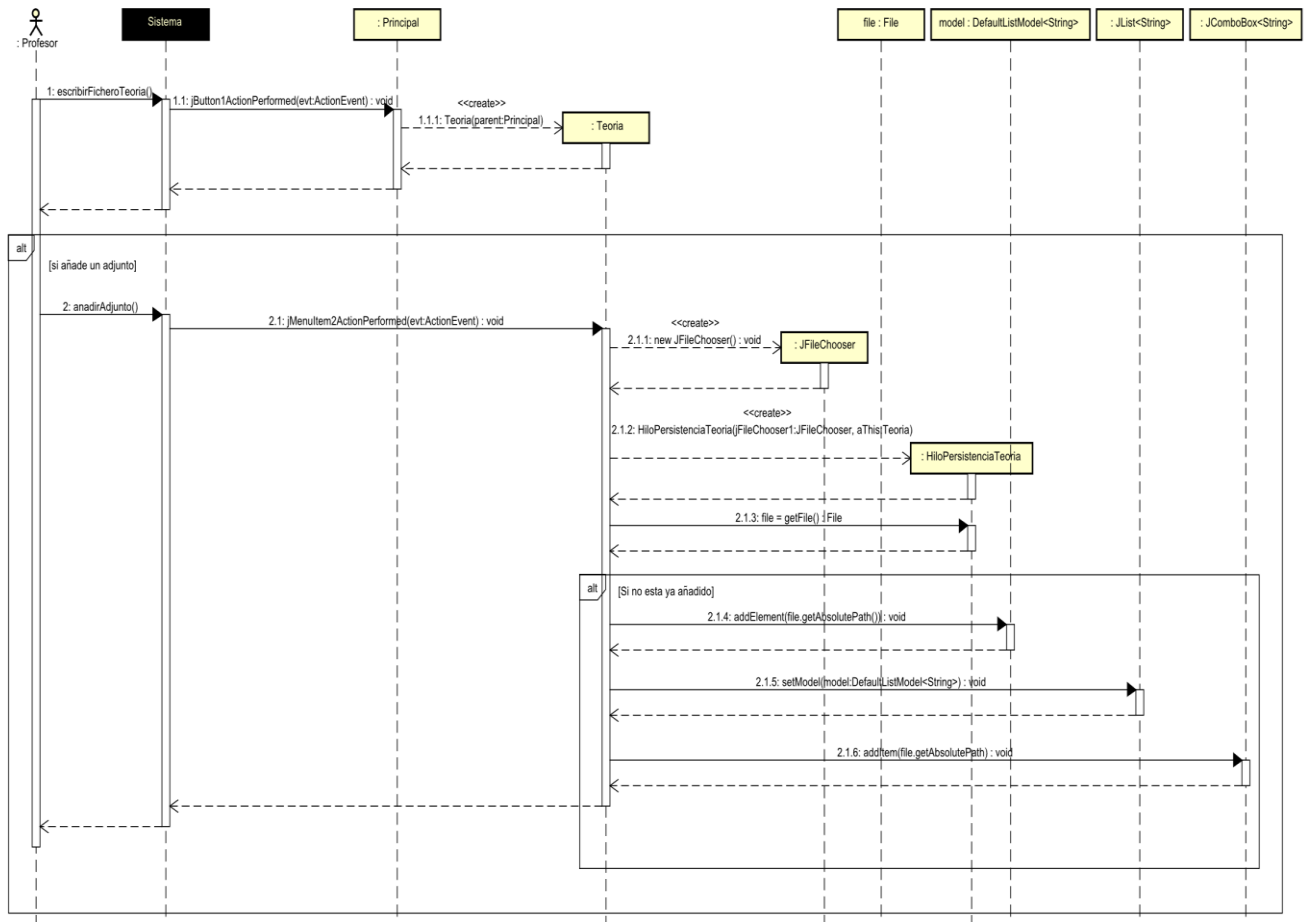


Ilustración 69: Diagrama de secuencia: Generar archivo de teoría II

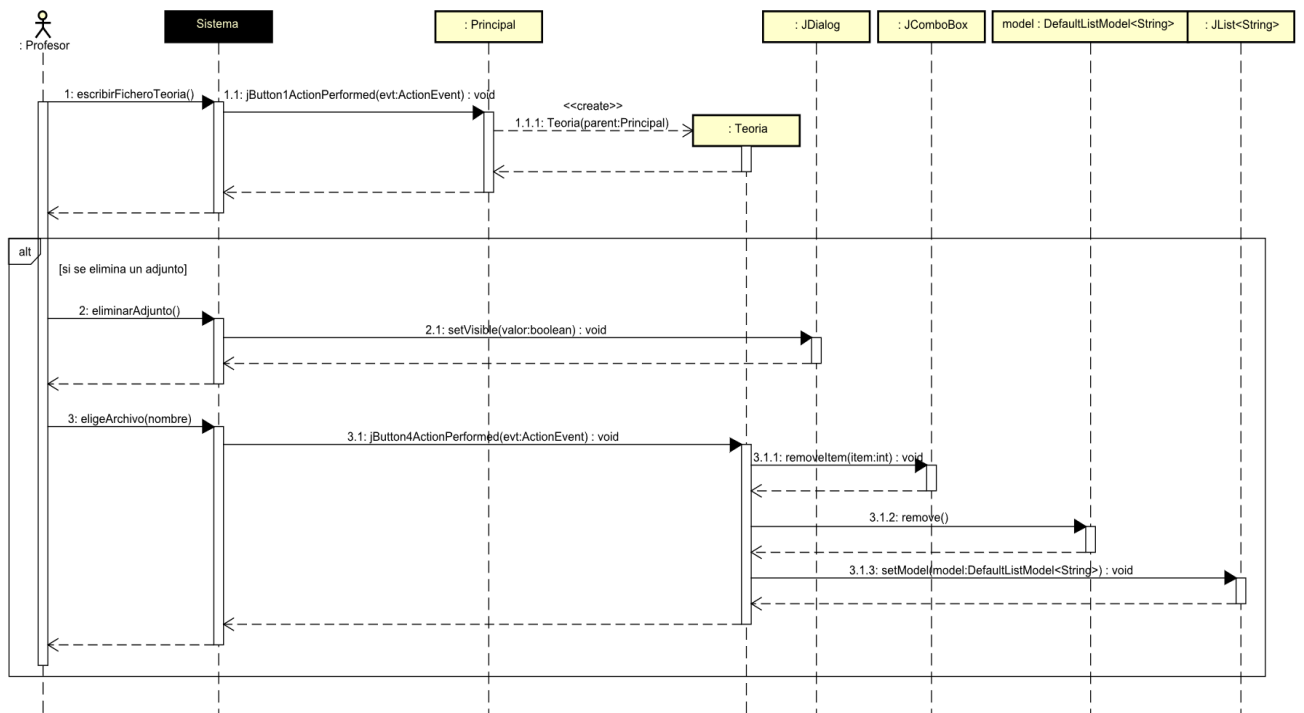


Ilustración 70: Diagrama de secuencia: Generar archivo de teoría III

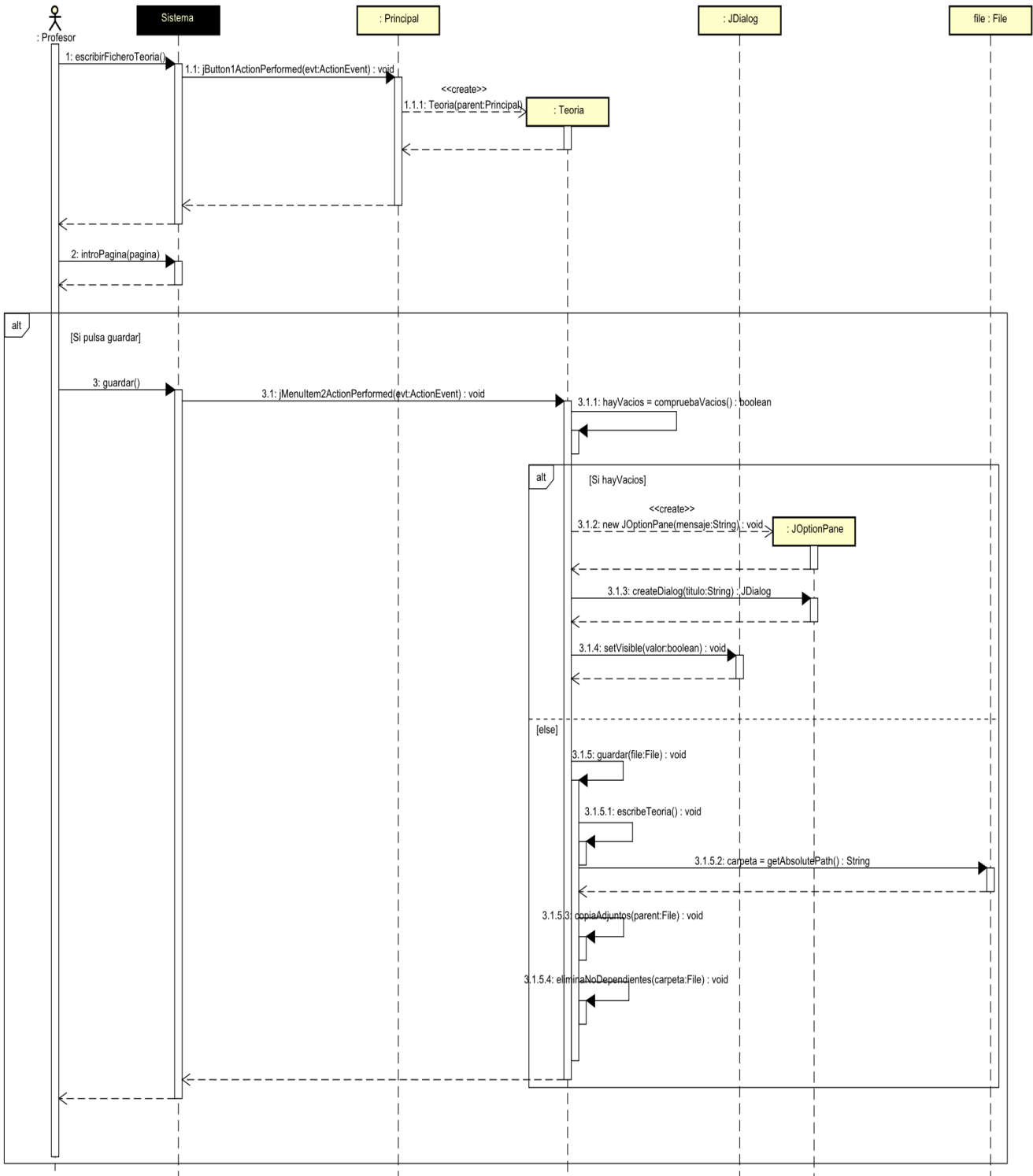


Ilustración 71: Diagrama de secuencia: Generar archivo de teoría IV

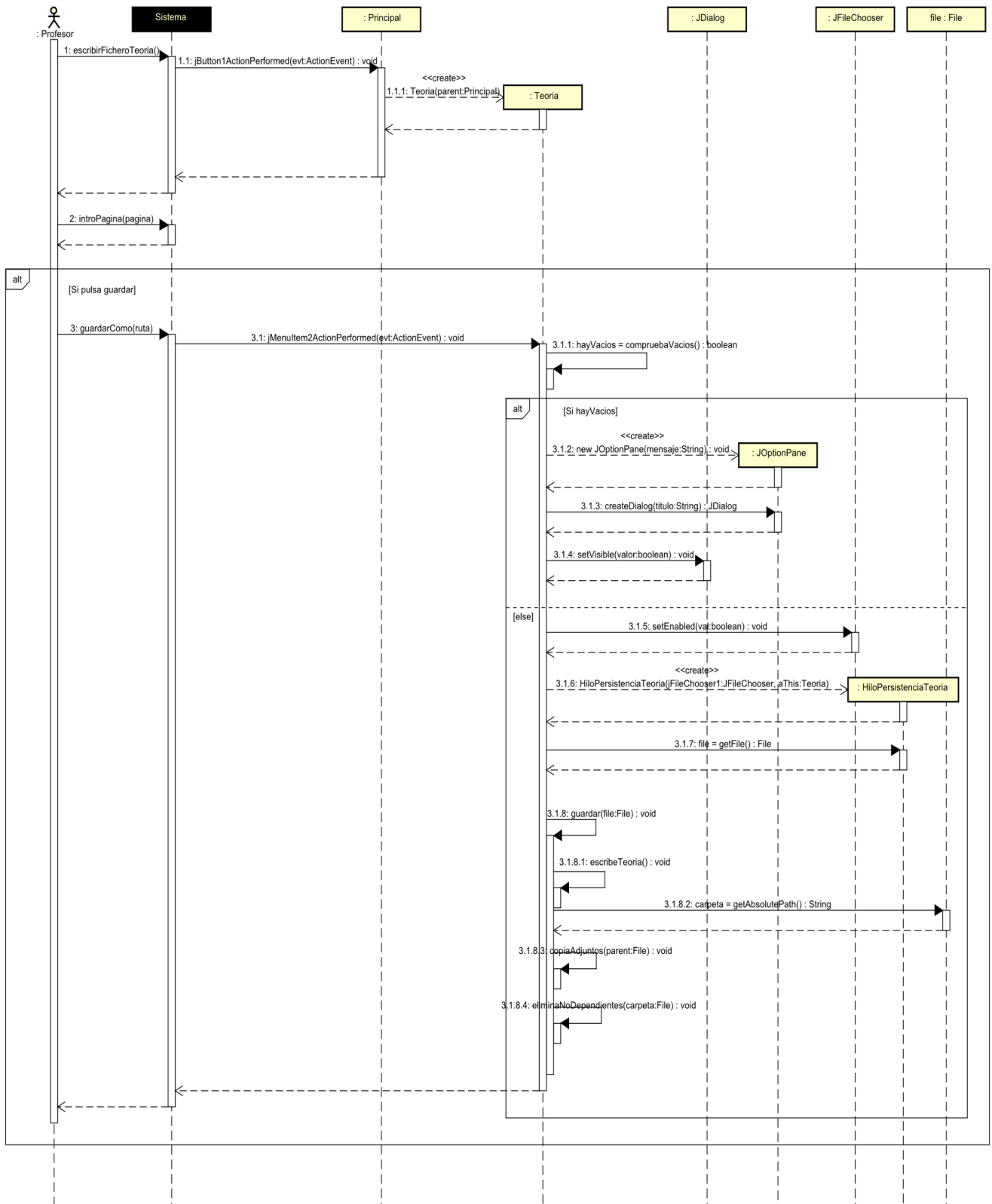


Ilustración 72: Diagrama de secuencia: Generar archivo de teoría V

Realizar cuestionario de examen.

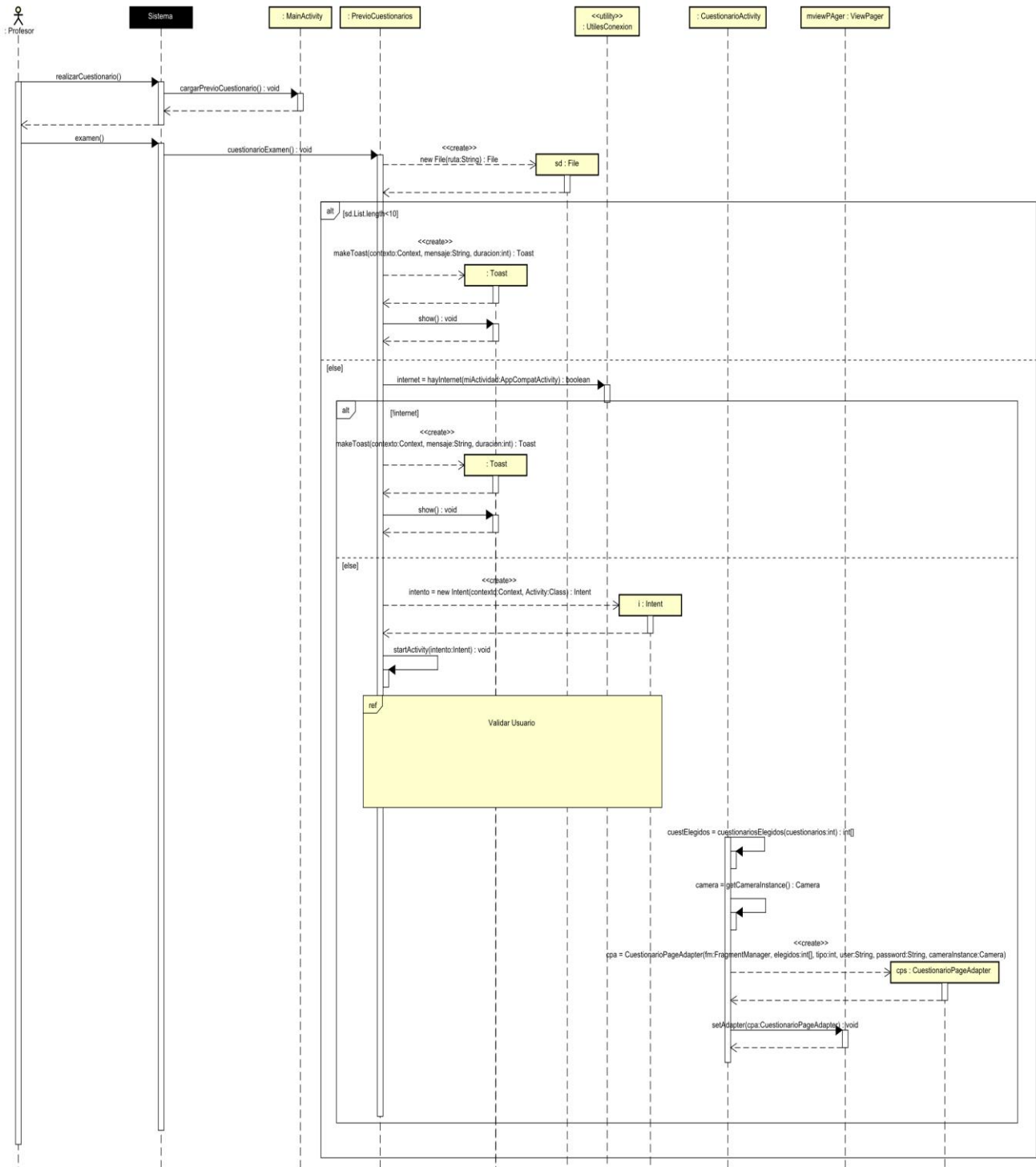


Ilustración 73: Diagrama de secuencia: Realizar cuestionario de examen I

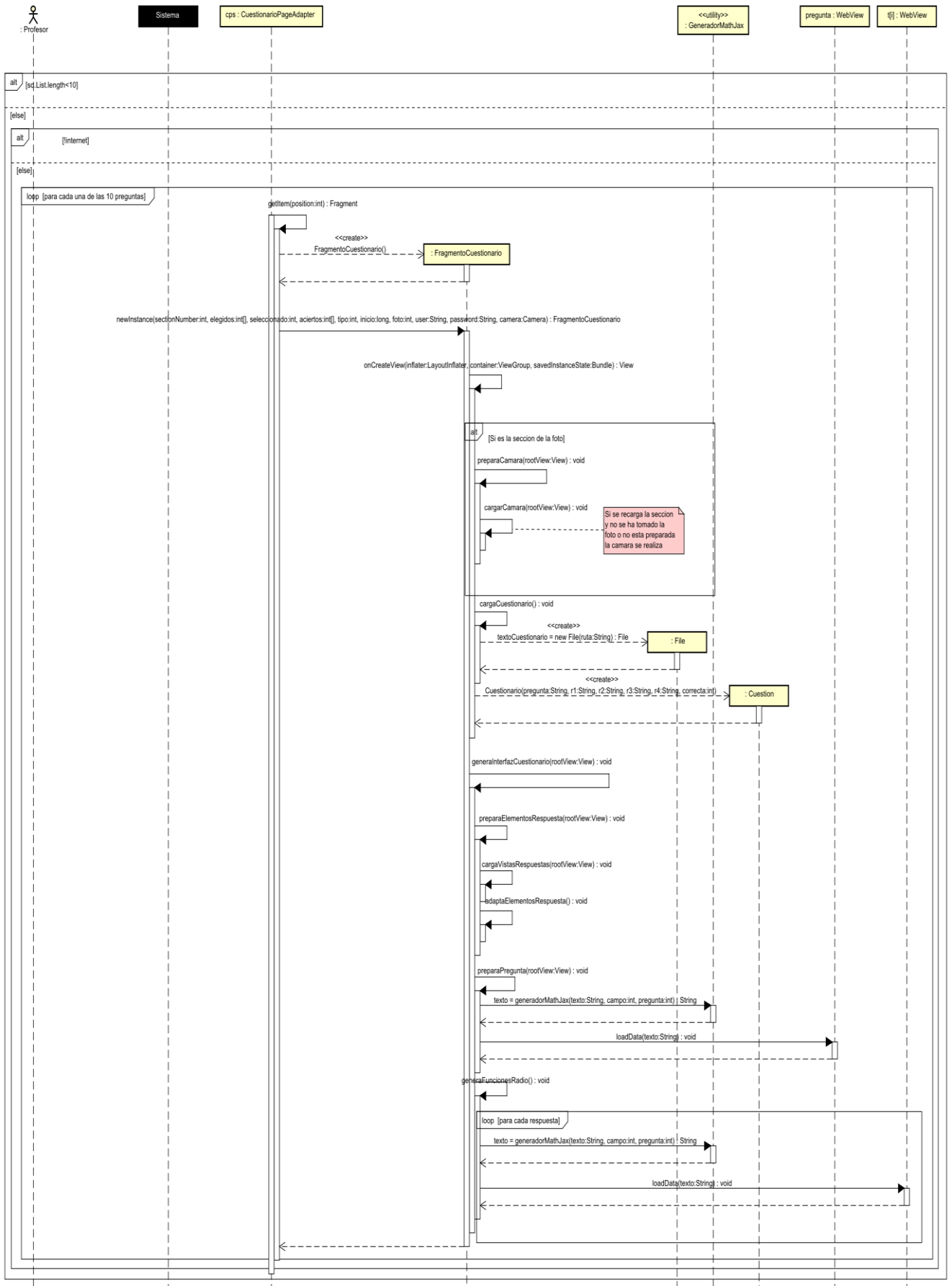


Ilustración 74: Diagrama de secuencia: Realizar cuestionario de examen II

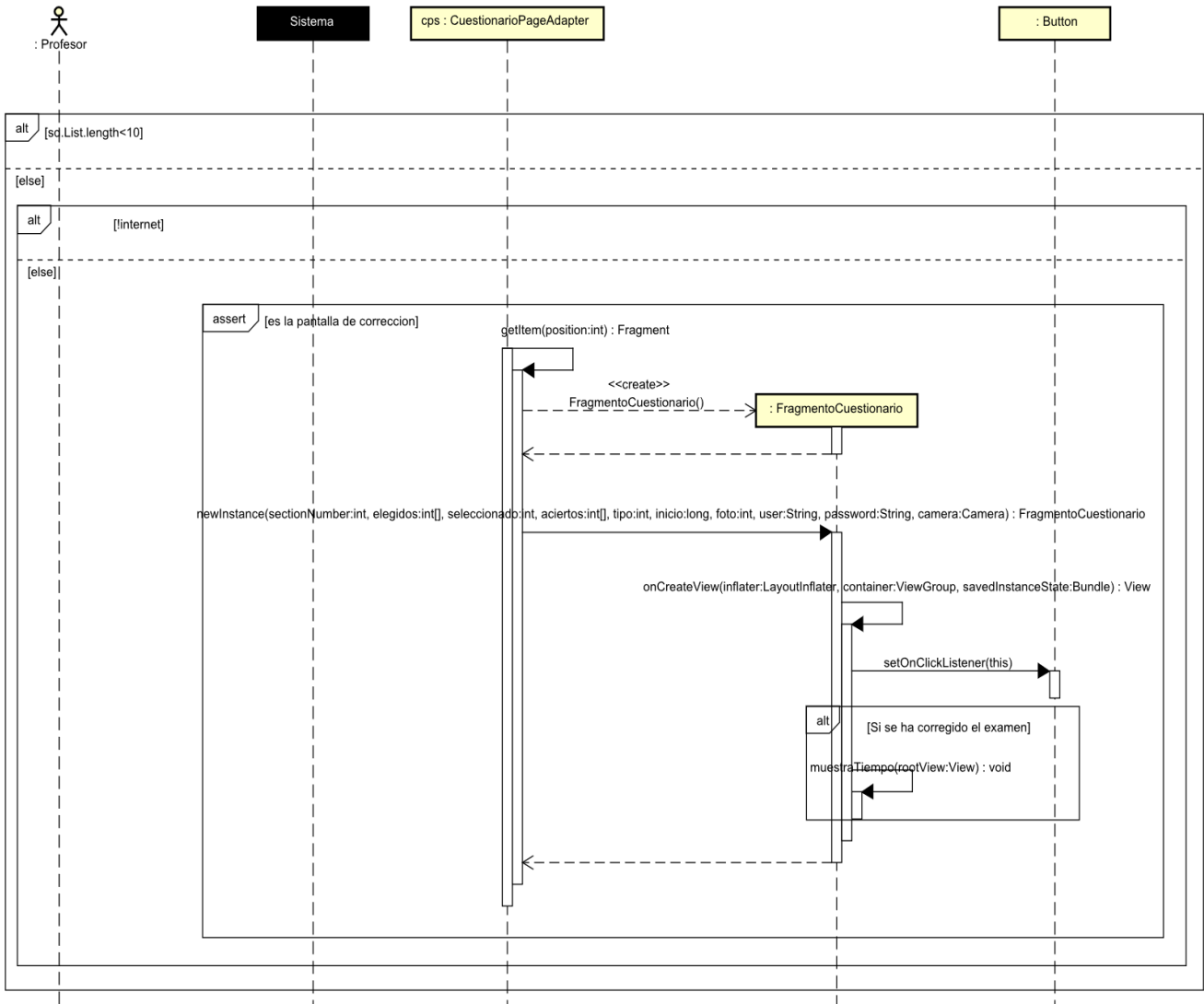


Ilustración 75: Diagrama de secuencia: Realizar cuestionario de examen III

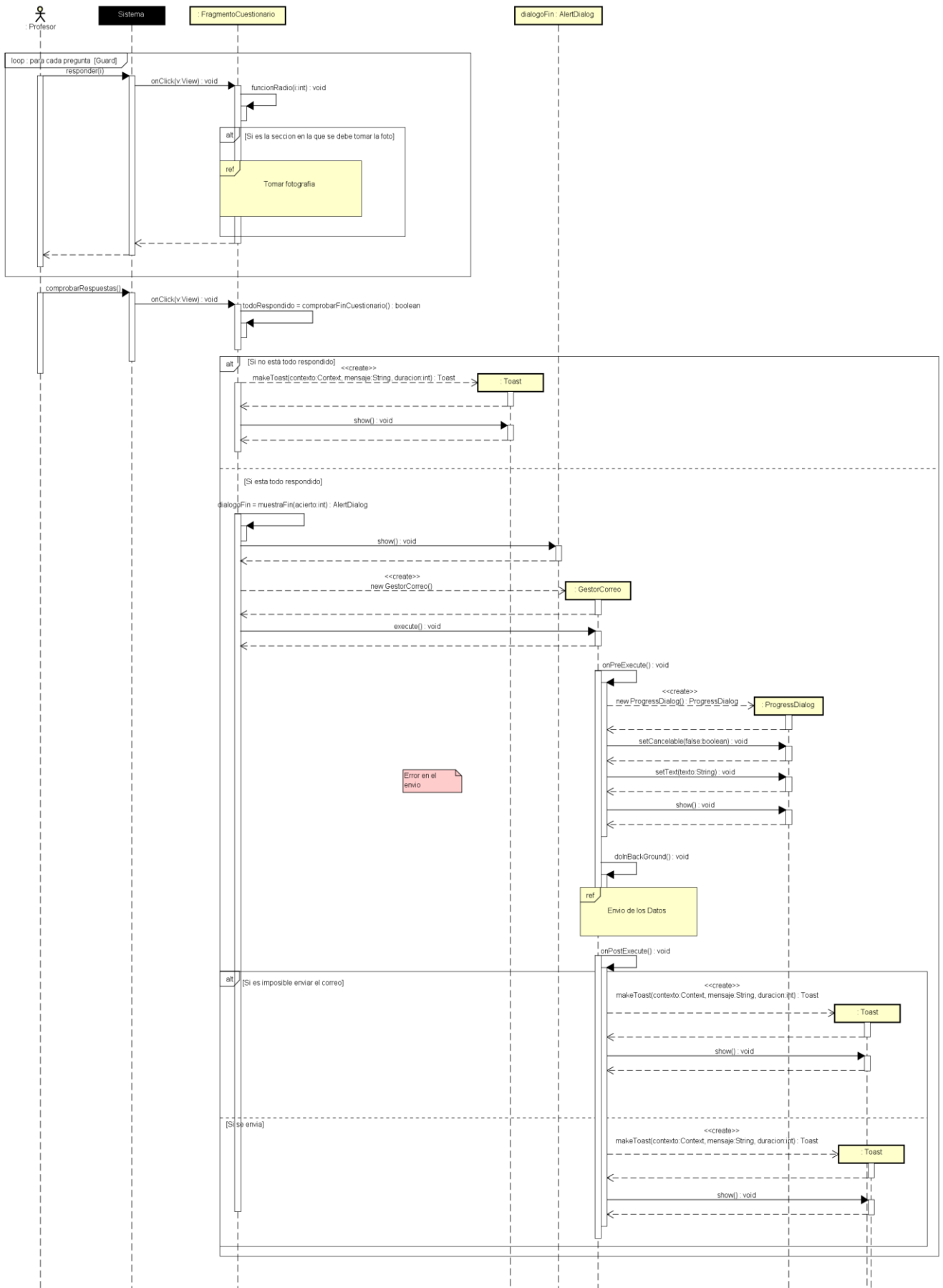


Ilustración 76: Diagrama de secuencia: Realizar cuestionario de examen IV

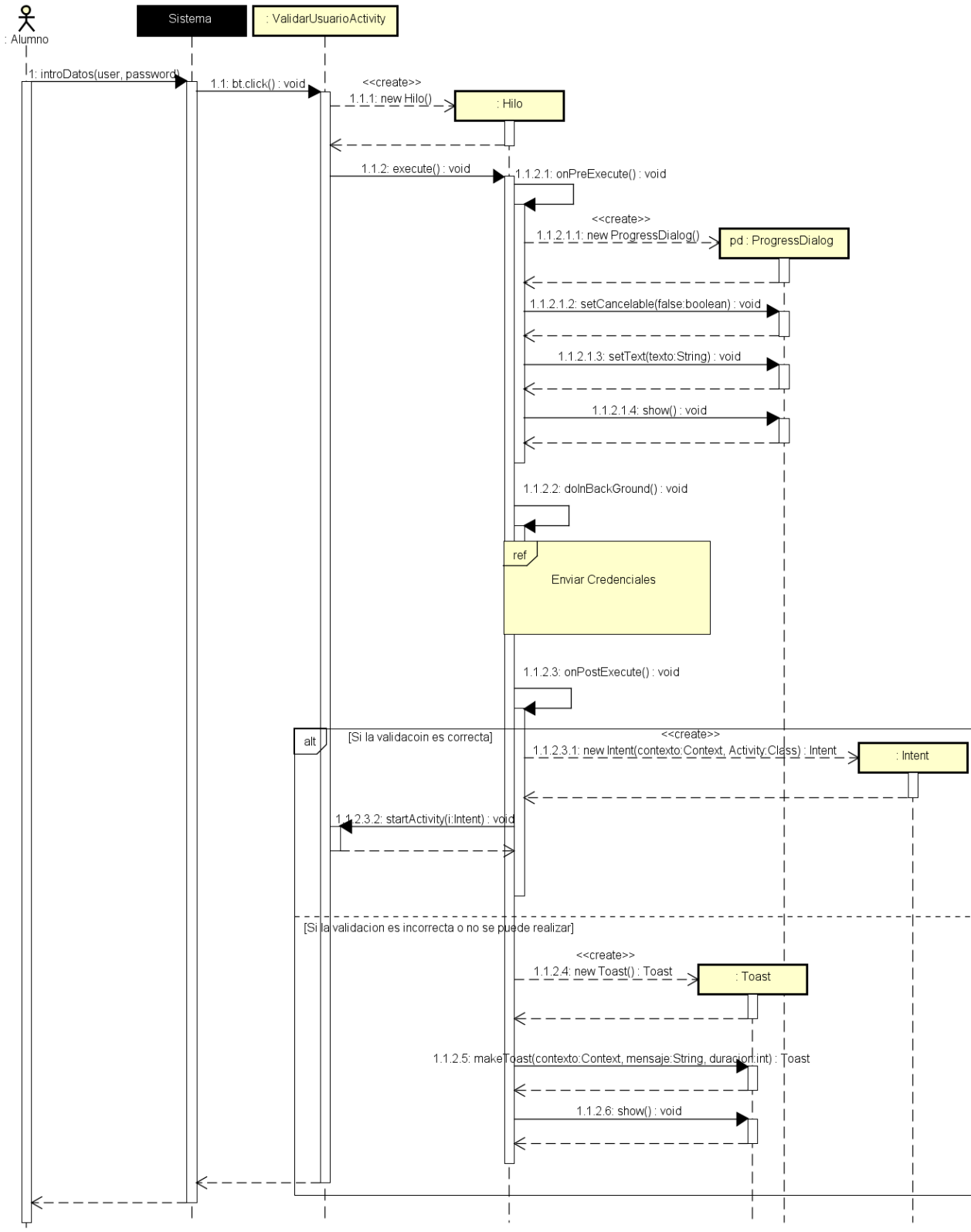


Ilustración 77: Diagrama de secuencia: Validar usuario



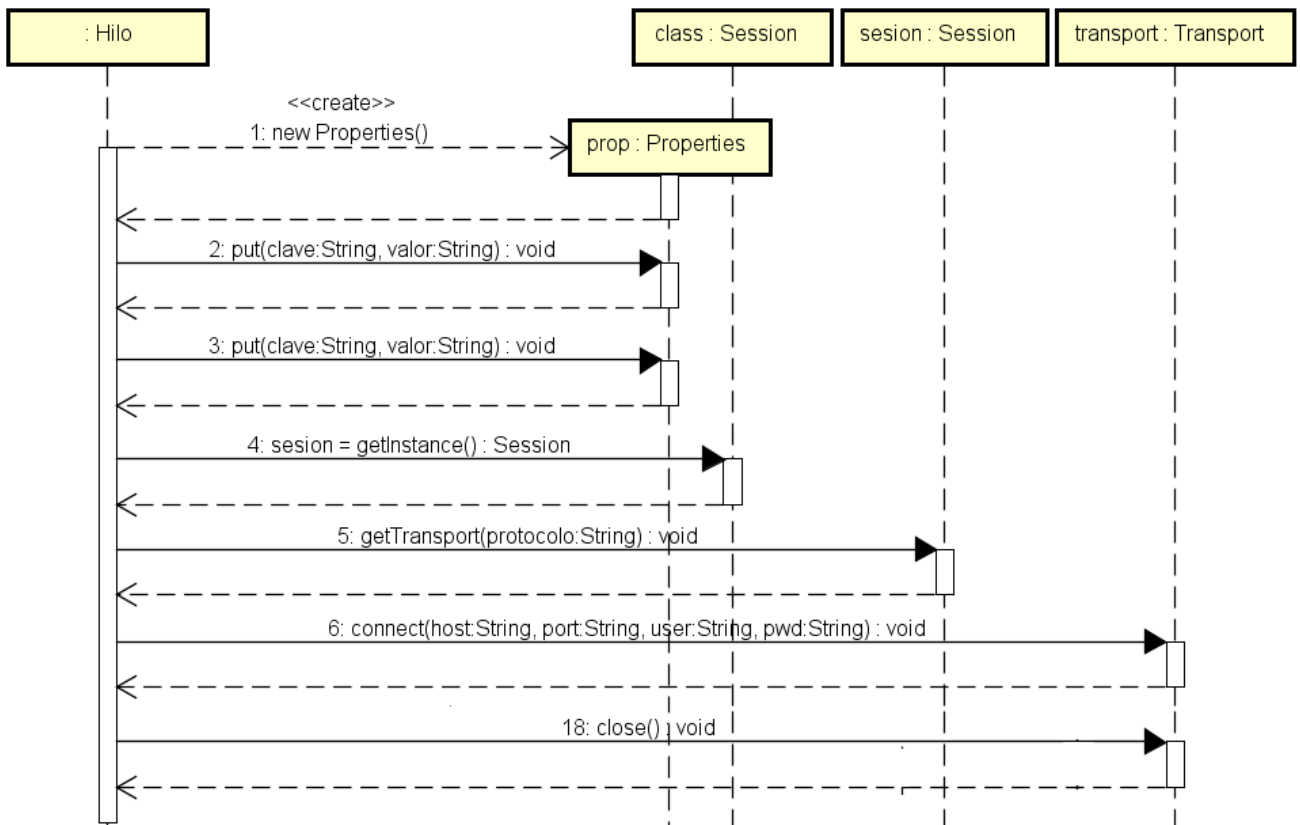


Ilustración 78: Diagrama de secuencia: Enviar credenciales

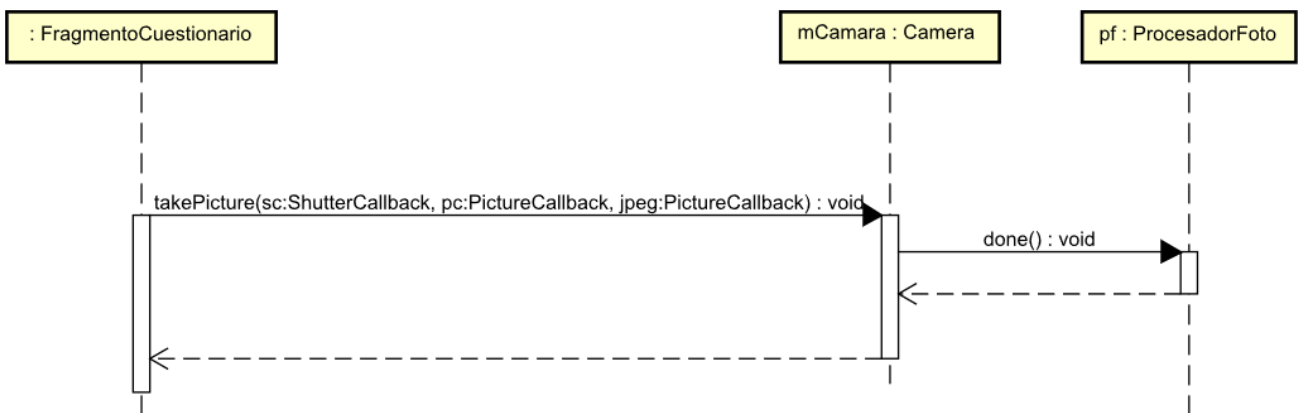


Ilustración 79: Diagrama de secuencia: Tomar fotografía

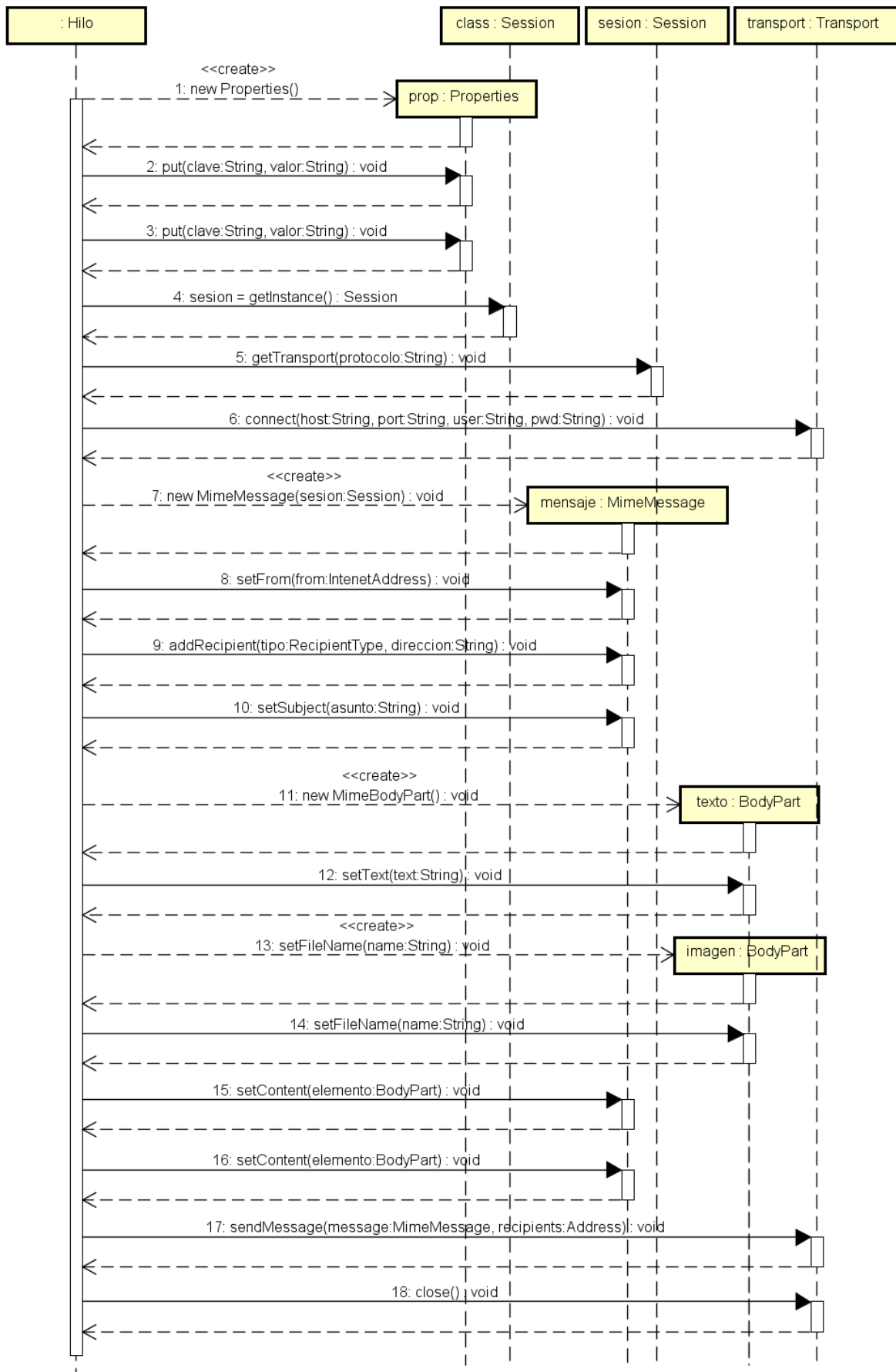


Ilustración 80: Envío de los Datos

# Descargar contenidos.

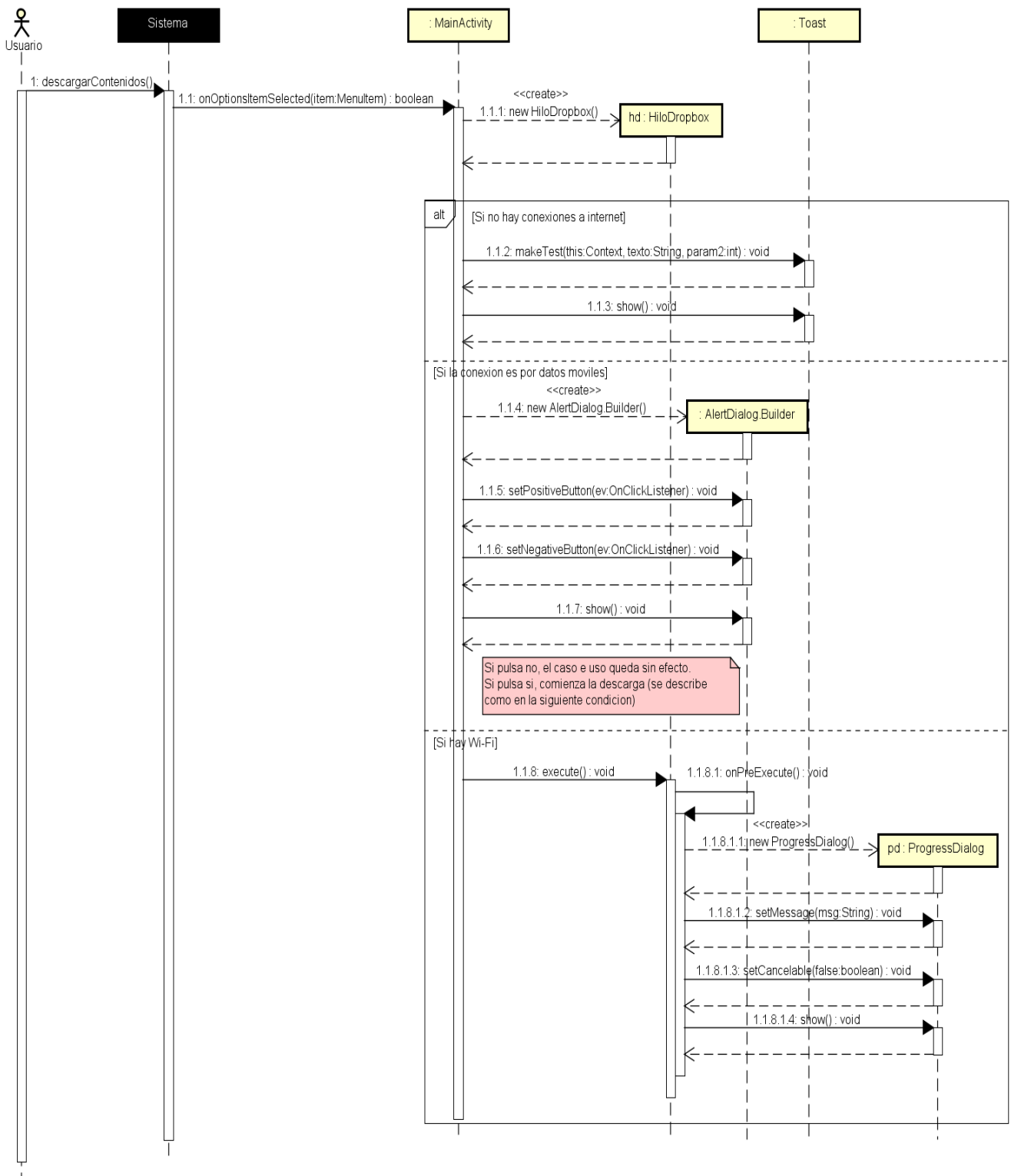


Ilustración 81: Descargar contenidos I

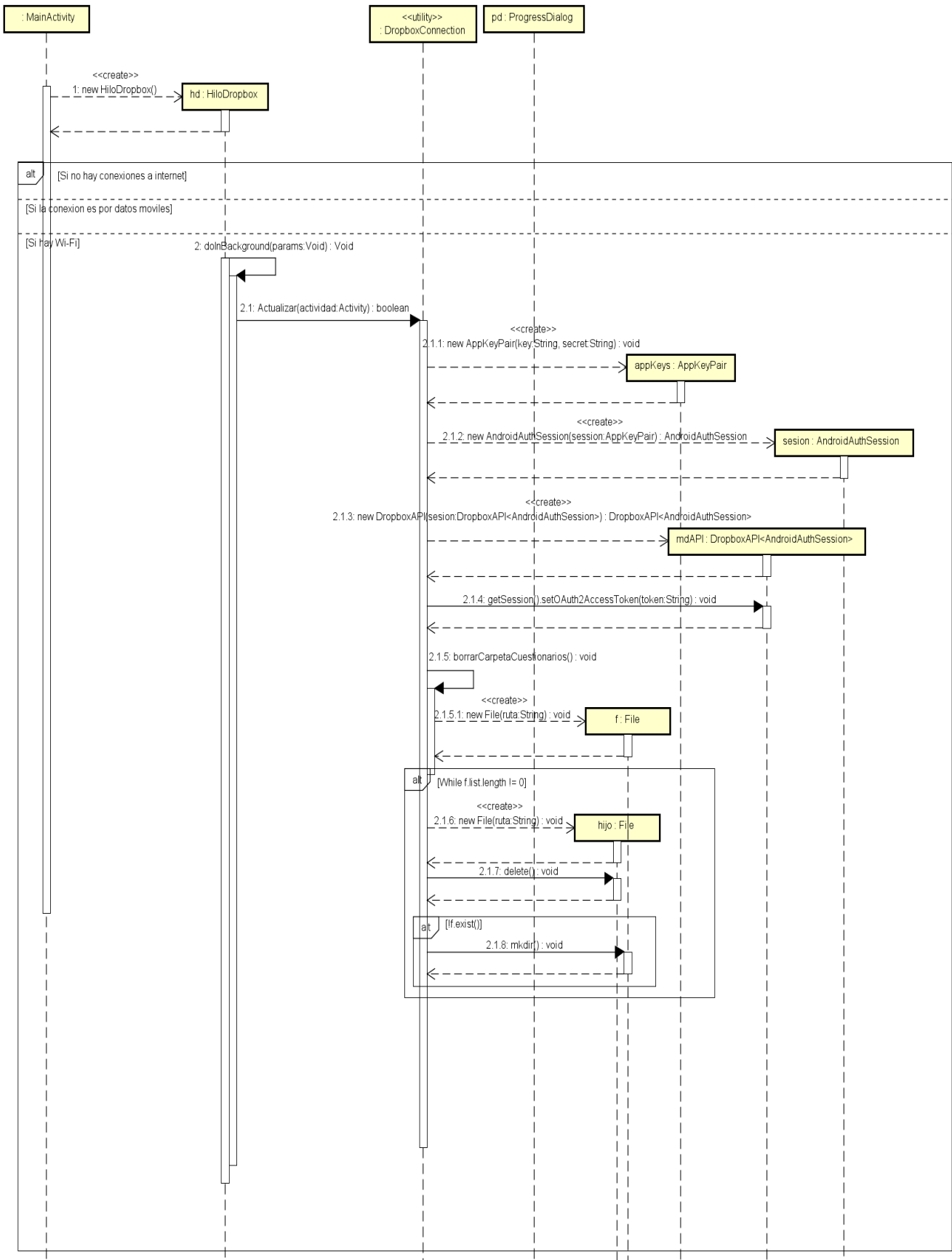


Ilustración 82: Descargar contenidos II

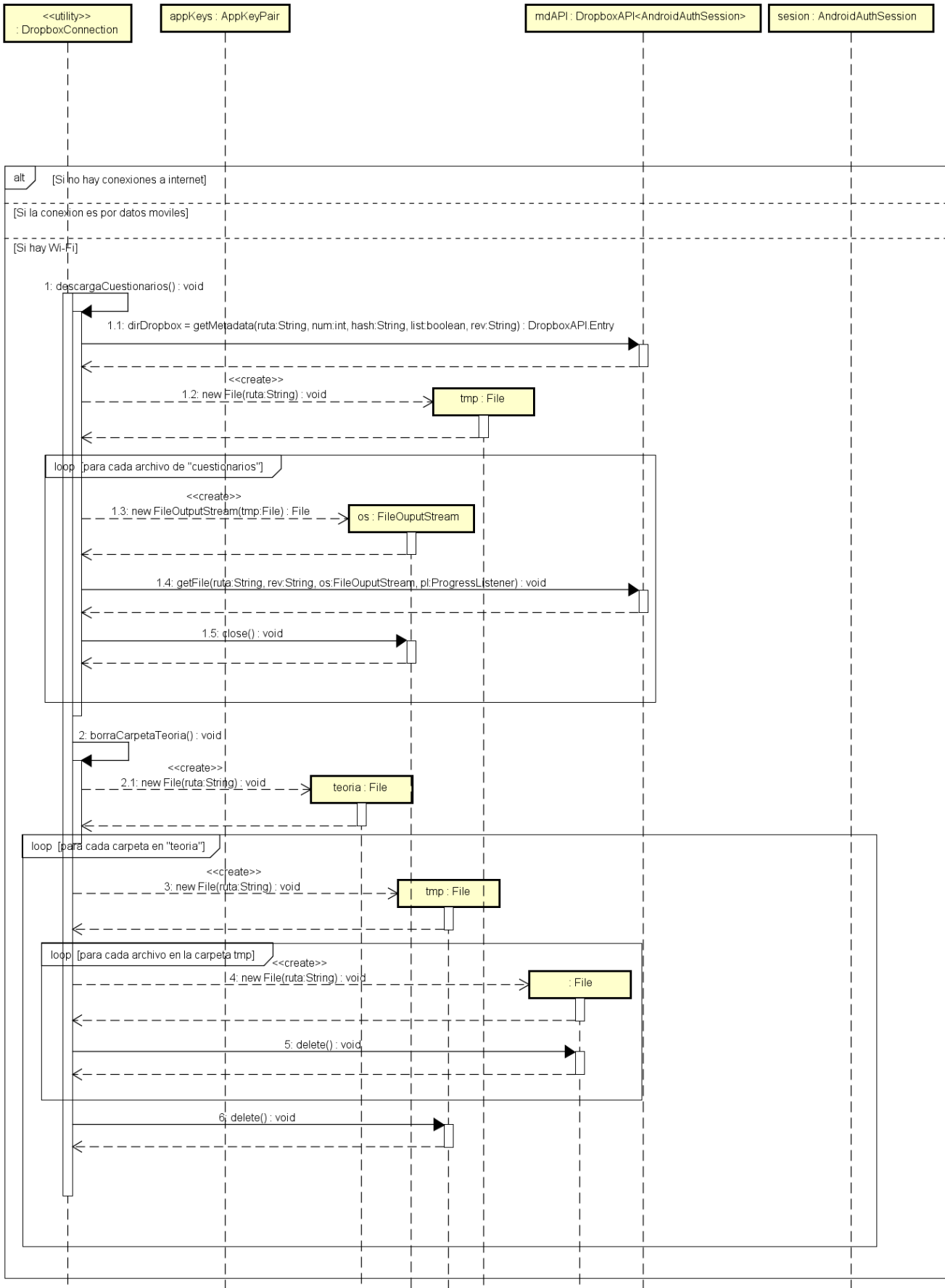


Ilustración 83: Descargar contenidos III

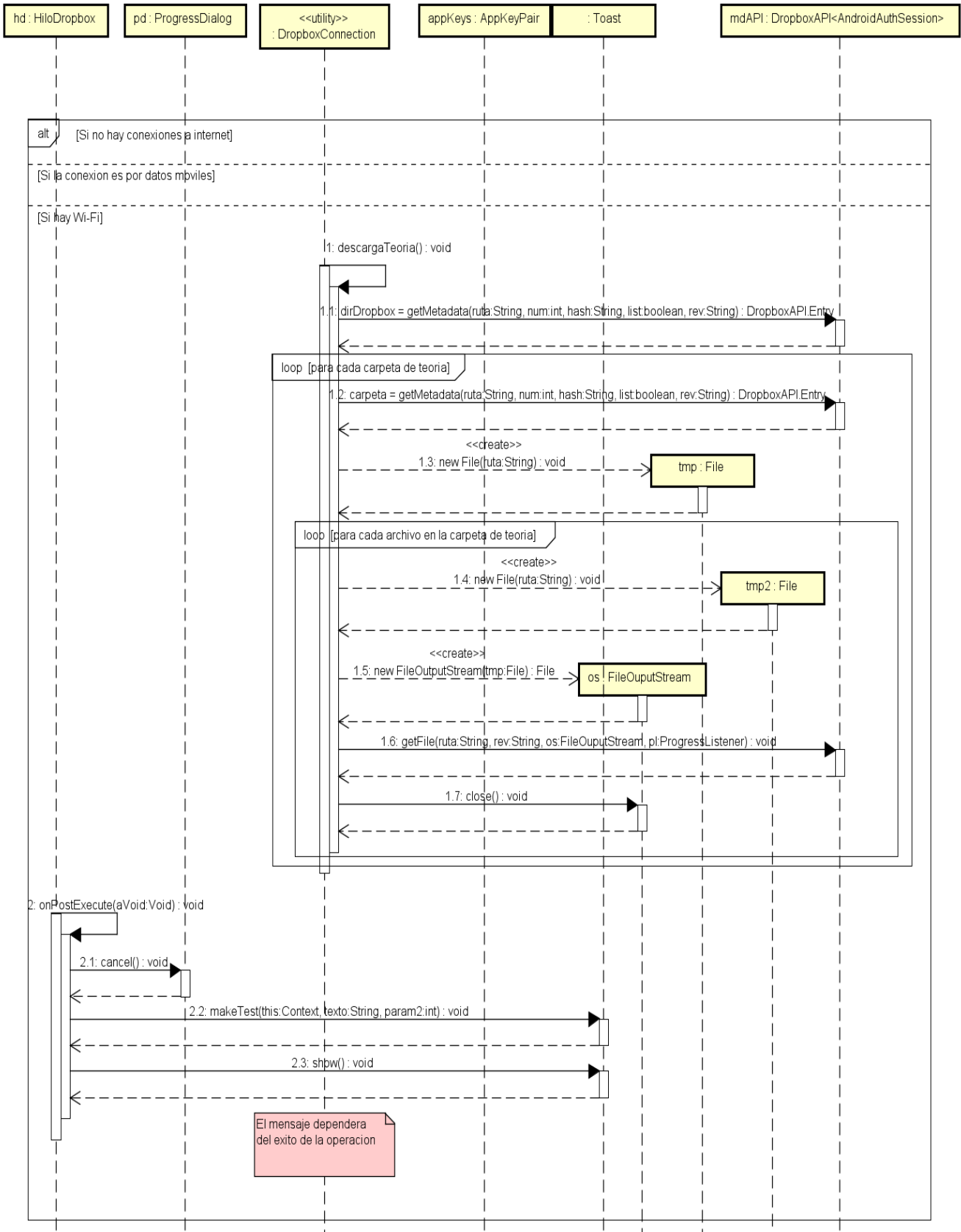


Ilustración 84: Descargar contenidos IV

# Capítulo V: Implementación.

En esta sección se explican cómo han sido implementadas todas las funcionalidades de la aplicación principal y de la aplicación complementaria.

## Mostrar la teoría.

Cuando se solicita a la aplicación mostrar los fundamentos teóricos, la actividad principal comprueba que existan contenidos para mostrar, si no existen, informa mediante un Toast de que no hay contenidos que mostrar. Si por el contrario si hay contenido que mostrar, previamente a lanzar TeoriaActivity comprueba que haya conexión a internet. Independientemente de si hay o no conexión se lanza TeoriaActivity pero en el caso negativo informa de la conveniencia de conectarse a la red (MathJax lo necesita para resolver TEX).

Una vez se lanza la actividad TeoriaActivity, esta crea un objeto TeoriaPagerAdapter. Este objeto genera un FragmentoTeoria (descendiente de Fragment), por cada página de teoría a mostrar y soportará los desplazamientos entre las páginas.

Cada FragmentoTeoria contiene un WebView, que es la vista que muestra el contenido, al que indica la ruta de la página web a mostrar (las paginas se almacenan en el teléfono). La idea de mostrar cada página de teoría como una página web se debe a la facilidad de mostrar imágenes, texto y otros contenidos multimedia con HTML, a la resolución de fórmulas con MathJax y a la funcionalidad que incluye WebView, que nos permite ampliar, compartir el contenido o buscar en internet un texto resaltado.

MathJax.

MathJax es una utilidad Javascript que permite mostrar formulas y otro contenido matemático escrito en la notación de TEX. Incluyendo un pequeño script en una página web, MathJax evaluará y mostrará todo el contenido encerrado entre '\$' o '\$\$' si lo que queremos es enfatizar ese contenido.

La implementación utilizada resuelve vía internet el contenido indicado, por lo tanto, es recomendable acceder con una conexión a internet a esta funcionalidad, de lo contrario, el contenido matemático no se resolverá.

Si quiere conocer más sobre esta utilidad, en la bibliografía podrá encontrar un enlace a la página web de los autores [41].

## Cuestionarios.

Cuando accedemos a los cuestionarios, se lanza una actividad que nos permite elegir entre realizar un cuestionario de examen o uno de autoevaluación. Una vez hemos elegido el tipo de cuestionario, la actividad de selección previa comprueba que haya preguntas en memoria y que haya conexión a internet. Si es un cuestionario de examen no permite acceder si no existe conexión, pues no se puede validar la identidad del alumno, si es uno de autoevaluación si permite acceder, pero insiste en la necesidad de tener una conexión para poder utilizar la resolución de fórmulas de MathJax.

Una vez hemos elegido un tipo de cuestionario se ejecuta un Intent que lanza la actividad CuestionarioActivity en caso de que sea una autoevaluación o la actividad ValidarUsuarioActivity para comprobar las credenciales de usuario en caso de examen. En ese Intent se almacena el tipo de cuestionario a realizar para informar a CuestionarioActivity del comportamiento que debe tener (En caso de exámenes se llamara a esta actividad cuando se validen los usuarios).



Si hemos elegido examen deberemos introducir nuestras credenciales proporcionadas UVa para acceder al test (identificador, email y contraseña). Un hilo comprobaba frente a Cartero la veracidad de esos datos, si son incorrectos o no es posible resolverlos, nos informara mediante un Toast, si son correctos nos dejara acceder al examen. Los datos de acceso pueden almacenarse en la aplicación para evitar introducirlos varias veces, estos datos se encriptan mediante AES, un algoritmo de encriptación simétrico. Puede encontrar una pequeña referencia a AES en la bibliografía [42].

Una vez se ha iniciado la actividad CuestionarioActivity, el CuestionarioPagerAdapter genera diez números aleatorios distintos entre 0 y el número de preguntas almacenadas en memoria (siempre 10 o más) para elegir las preguntas que se incluyen en el cuestionario. Después genera los FragmentoCuestionario que muestran las preguntas.

Cada FragmentoCuestionario accederá a un fichero de pregunta utilizando la ruta que le otorga el CuestionarioPagerAdapter y la muestra por pantalla. Para desplazarnos por las preguntas deslizaremos nuestro dedo hacia la derecha o hacia la izquierda si queremos retroceder o avanzar respectivamente.

Si prestamos atención a la vista de una pregunta, veremos que los elementos autoexcluyentes de respuesta no son RadioButton sino CheckBox, la razón de esto es que el texto que acompaña a un RadioButton solo puede ser texto plano. Junto a cada CheckBox tenemos una página web con la posible respuesta, esta decisión nos permite poder emplear contenido matemático en la respuesta utilizando MathJax. La pregunta también es una página web. Para evitar que en un examen un alumno utilice las características de un WebView (como por ejemplo buscar en internet un texto remarcado) se ha bloqueado en LongClick en este tipo de prueba.

Una vez que pulsamos un elemento de respuesta este es marcado y es imposible desmarcarlo, salvo seleccionando otro. En un cuestionario de autoevaluación podemos corregir al momento pulsando el botón “comprobar” en la parte de abajo de la pantalla, si lo pulsamos sin haber respondido, un Toast nos avisa de que debemos responder. Cuando hemos respondido y corregido todas las preguntas, un Dialog nos muestra el resultado y nos permite ofrece revisar el cuestionario o salir.

En un cuestionario de examen es algo distinto. En primer lugar, no hay botón responder en cada pregunta, solo hay uno en la última página y requiere haber respondido todas las preguntas, en segundo lugar, al responder una pregunta, si el número de página coincide con un número indicado por el CuestionarioPagerAdapter que contiene las preguntas, el sistema tomará una fotografía para verificar que es el alumno que introdujo sus datos de acceso quien realiza el examen y en tercer lugar, al pulsar el botón “comprobar”, antes de mostrar los resultados se envía un correo al profesor con el número de aciertos, el tiempo que se ha tardado en realizar el examen y la fotografía.

## Simulación.

En la simulación, lo primero que se puede apreciar simplemente girando nuestro dispositivo, es que la pantalla no rota. La razón de esto es muy sencilla, para calcular las líneas de campo y el potencial, es necesario conocer la posición de las cargas y esta posición no es la misma con el teléfono vertical y con el teléfono horizontal.

En cuanto a la representación de los elementos, representaremos las cargas en la interfaz mediante elementos Carga que son descendientes de ImageView. Estos elementos Carga se dispondrán sobre un RelativeLayout al igual que las líneas de campo y tendrán un diámetro de una sexta vez el ancho de la pantalla, la razón de esto es evitar cargas muy pequeñas en pantallas grandes o cargas muy grandes en pantallas pequeñas.

Para pintar las líneas de campo en el RelativeLayout utilizamos el objeto Canvas, al que le indicamos los puntos que debe pintar mediante una lista de vectores de números flotantes de simple precisión. Cada vector contiene los puntos que forman una línea (el elemento 0 es la coordenada X del primer punto, el 1 es la coordenada Y, el 2 la coordenada X del segundo punto y el 3 la Y y así sucesivamente). Cuando el usuario introduce una carga, la utilidad DibujaLineas genera un vector para cada una de las líneas con los puntos que la forman y se los pasa a la utilidad Lineas que es quien dibuja.

El cálculo de las líneas de campo se realiza mediante un algoritmo publicado en 1985 en "The Institute of Physics". Este algoritmo toma el último punto que ha calculado (en el caso de ser el primer punto, el origen es el uno de los puntos del vector puntosInicio de la carga), calcula la dirección del campo en ese punto y después calcula un segmento de un tamaño dado que comienza en el punto origen y tiene la dirección de ese vector de dirección.

A continuación, se expone el algoritmo de cálculo de líneas en más profundidad.

Algoritmo de cálculo de líneas de campo.

Como hemos dicho antes, este algoritmo utiliza el punto anterior para calcular el siguiente. Para asegurarnos que las líneas de campo no se pisan, no tomaremos el centro de la carga como inicio sino varios puntos de su superficie, uno por cada línea. El número de puntos de inicio (líneas) por cada carga depende del valor de la carga. Para cargas positivas de hasta 12,5 culombios son 4 puntos, de 12,5 a 25 son 8, de 25 a 37,5 son 12 y de 37,5 culombios a 50 son 16 puntos, de cargas negativas no salen líneas. En cuanto al tamaño del segmento dibujado en cada iteración son 5 píxeles independientes de la densidad (en adelante D).

Una vez tenemos estos datos el algoritmo nos permite saber cuánto debemos incrementar (decrementar) los puntos actuales para alcanzar el punto siguiente. Esos incrementos se calculan del siguiente modo:

$$\Delta x = \frac{D \cdot E_x}{\sqrt{E_x^2 + E_y^2}}$$

$$\Delta y = \frac{D \cdot E_y}{\sqrt{E_x^2 + E_y^2}}$$

donde  $E_x$  y  $E_y$  son la componente X e Y del campo total.

Repetiremos este cálculo 3000 veces por cada línea. Por lo tanto, si una línea de campo es totalmente recta tendrá una longitud de 15000 píxeles independientes de la densidad. Obviamente, rara vez será necesario realizar 3000 iteraciones por lo que hay que deducir un criterio de parada. El criterio de parada utilizado lo deduje realizando pruebas con el depurador de Android Studio y se trata de lo siguiente.

Cuando la línea de campo llega a una posición muy cercana al centro de una carga de negativa, los puntos calculados posteriormente se repiten cada dos iteraciones, es decir, los puntos calculados se mueven entre dos separados una distancia D. Si esta explicación no le resulta clara, pondré un ejemplo:

Si la línea sale de la carga positiva y llega a una posición cercana (muy cercana) al centro de la negativa en 2000 iteraciones, las 1000 siguientes se moverán entre dos valores separados una distancia D fija, es decir, si suponemos que la primera posición que se va a repetir es x, la iteración 2000 devolverá x, la 2001 devolverá  $x + D$ , la 2002 devolverá X, la 2003 devolverá  $x+D$  y así sucesivamente.

El criterio se basa en comprobar si se da esta situación.

## Generación de contenidos teóricos y de evaluación.

Para generar los contenidos teóricos y de evaluación se ha decidido implementar una herramienta que facilite estas tareas. Esta herramienta no solo permite crear contenidos sino también editarlos y previsualizarlos mediante el navegador predeterminado del sistema. El desarrollo de esta herramienta ha sido en java con el objetivo de ser utilizable en la mayoría de los sistemas operativos.

Para generar un cuestionario, introduciremos la pregunta, las respuestas y el número de la respuesta correcta. El archivo resultante será un fichero de tipo test (\*.test) con una estructura definida. La primera línea será un número mágico para evitar cargar a la aplicación archivos indebidos, la segunda es la pregunta, las 4 siguientes las respuestas y la última el número de la respuesta correcta.

En el caso de los ficheros de teoría, la estructura será la siguiente. La primera línea del fichero de tipo teoría generado (\*.teoria) volverá a ser un número mágico para evitar que se abran ficheros indebidos con la aplicación. El resto de las líneas serán una página HTML (solo el body) que, en caso de enlazar con los ficheros adjuntos, lo hará mediante rutas relativas.

Tanto en el caso de ficheros de teoría como en el de cuestionarios es importante como guardarlos en el repositorio. En la raíz del repositorio, debemos colocar dos carpetas: "cuestionarios" y "teoría". Dentro de la primera, guardaremos los archivos de cuestionarios (.test) y dentro de la segunda, los directorios que contienen las cada una de las páginas de teoría y sus adjuntos. Esto es necesario para el correcto funcionamiento de la aplicación.

## Conexión con el repositorio.

Para actualizar los datos archivos de teoría y cuestionarios utilizaremos las clases DropboxConnection e HiloDropbox que nos permiten comunicarnos con el repositorio.

La clase HiloDropbox, descendiente de AsyncTask, nos permite generar un hilo secundario para realizar la actualización, pues recordemos que no está permitido realizar operaciones de red en el hilo principal. Además de esto, ese hilo también se encarga de mostrar un ProgressDialog que se mantiene durante toda la actualización.

Para realizar la actualización, el hilo llama a al método actualizar de DropboxConnection. Este método, primero elimina los cuestionarios que actualmente están en el teléfono y después realiza la descarga de cuestionarios. Una vez ha finalizado realiza el mismo procedimiento con los archivos de teoría.

## Dropbox.

Para utilizar Dropbox como repositorio, lo primero que se requiere es dar de alta la aplicación en la Consola de Aplicaciones de Dropbox, para ello indicamos el nombre de la aplicación, si utilizara toda la cuenta o una

carpeta (Campos Eléctricos solo puede acceder a una carpeta) y si utilizaremos o no el API empresarial. Como resultado de este registro obtendremos dos valores, el secreto y la llave, estos valores son necesarios para poder conectarnos al repositorio.

Una vez finalizado este proceso, descargamos es SDK Java de Dropbox y lo agregamos a nuestro proyecto (Esta aplicación utiliza el SDK 1.6.3). De este SDK destacamos las clases `AndroidAuthSession`, y `DropboxAPI` que son las que nos permiten conectarnos y los métodos `metadata` y `getFile` que nos permiten conocer el contenido de una carpeta y descargar un archivo respectivamente.

## Otras consideraciones.

El estilo que se ha aplicado a toda la aplicación es de elaboración propia. En este diseño, se ha utilizado como estilo base `Theme.AppCompat.Light.DarkActionBar` y sobre él se han realizado diversas modificaciones. Para el estilo de los botones el estilo base utilizado ha sido `android:Widget.Holo.Button.Inset` sobre el que también se han modificado atributos. En `res/values/styles.xml` esta toda la definición de los estilos.

La aplicación está escrita en inglés como idioma principal y también tiene versión en castellano que es la que se muestra si tenemos el terminal móvil configurado en español. En cuanto a los contenidos de teoría y examen mostrados, el idioma de estos será el que decida la persona que los genere.

A parte de las clases contenidas en Android Studio y las clases de diseño, se han tenido que utilizar otras contenidas en las siguientes librerías para poder realizar todas las funciones requeridas:

- `activation.jar` (Para conexión con SMTP)
- `aditional.jar` (Para conexión con SMTP)
- `mail.jar` (Para conexión con SMTP)
- `dropbox-android-sdk-1.6.3.jar` (Para conexión con Dropbox)
- `json_simple-1.1.jar` (Para conexión con Dropbox)

# Capítulo VI: Pruebas.

En esta sección se muestran todas las pruebas que se han realizado en los diferentes sprint realizados en el proyecto. Cada tabla muestra la descripción de la prueba, la salida obtenida y la esperada.

Pruebas del dominio de la aplicación.

**Prueba 01**


<b>Descripción</b>	Se intenta cargar páginas de teoría sin existir contenidos.
<b>Salida esperada</b>	Mensaje de error.
<b>Salida obtenida</b>	
<b>Resultado</b>	Correcto.

Tabla 47: Prueba del dominio de la aplicación 1

**Prueba 02**

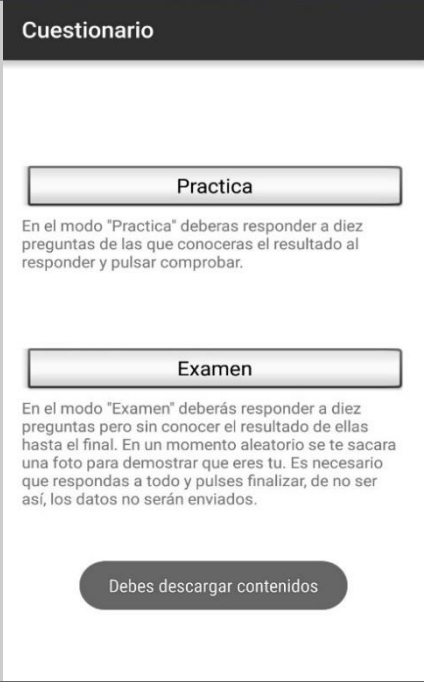
<b>Descripción</b>	Se intenta cargar cuestionarios sin existir contenidos.	
<b>Salida esperada</b>	Mensaje de error.	
<b>Salida obtenida</b>		
<b>Resultado</b>	Correcto.	

Tabla 48: Prueba del dominio de la aplicación 2

**Prueba 03**

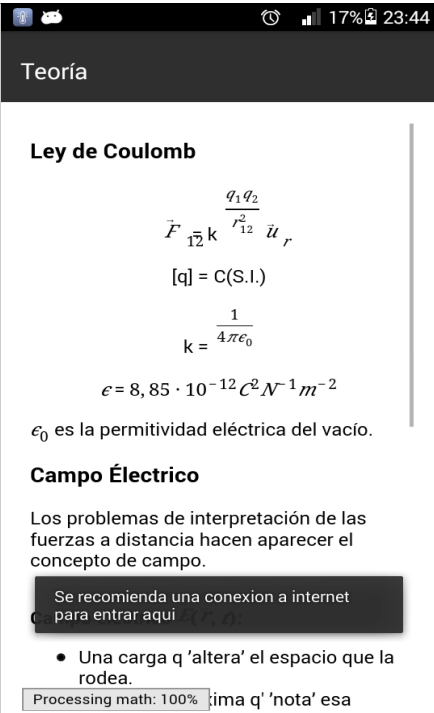
<b>Descripción</b>	Se intenta cargar páginas de teoría sin conexión a internet.	
<b>Salida esperada</b>	Aviso.	
<b>Salida obtenida</b>		
<b>Resultado</b>	Correcto.	

Tabla 49: Prueba del dominio de la aplicación 3

**Prueba 04**

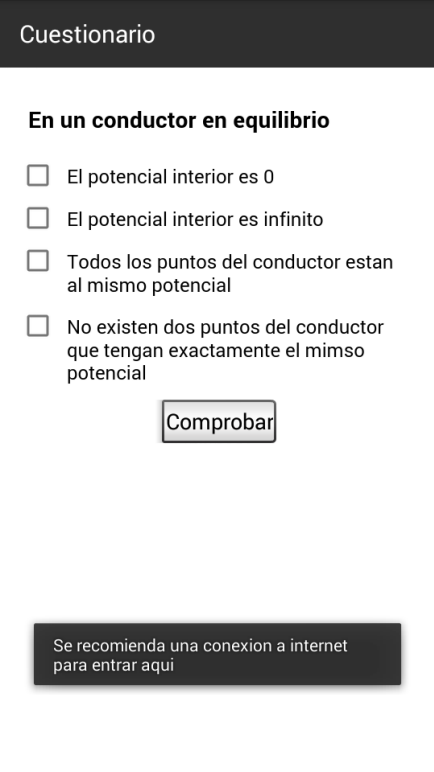
<b>Descripción</b>	Se intenta cargar un cuestionario de autoevaluación sin conexión a internet.
<b>Salida esperada</b>	Aviso.
<b>Salida obtenida</b>	 <p>The screenshot shows a questionnaire titled "Cuestionario" with the heading "En un conductor en equilibrio". It contains four multiple-choice questions:         <ul style="list-style-type: none"> <li><input type="checkbox"/> El potencial interior es 0</li> <li><input type="checkbox"/> El potencial interior es infinito</li> <li><input type="checkbox"/> Todos los puntos del conductor estan al mismo potencial</li> <li><input type="checkbox"/> No existen dos puntos del conductor que tengan exactamente el mismo potencial</li> </ul>         Below the questions is a "Comprobar" button. At the bottom, a dark box contains the text: "Se recomienda una conexión a internet para entrar aquí".       </p>
<b>Resultado</b>	Correcto.

Tabla 50: Prueba del dominio de la aplicación 4

**Prueba 05**

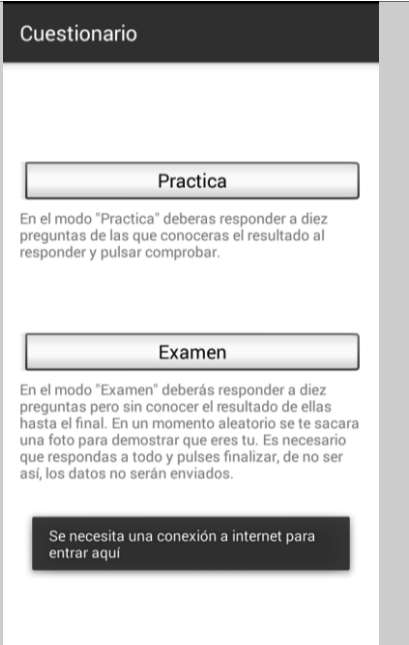
<b>Descripción</b>	Se intenta cargar un cuestionario de examen sin conexión a internet.
<b>Salida esperada</b>	Error.
<b>Salida obtenida</b>	 <p>The screenshot shows a questionnaire titled "Cuestionario" with two main options: "Practica" and "Examen".         <ul style="list-style-type: none"> <li>The "Practica" button is followed by the text: "En el modo 'Practica' deberas responder a diez preguntas de las que conoceras el resultado al responder y pulsar comprobar."</li> <li>The "Examen" button is followed by the text: "En el modo 'Examen' deberás responder a diez preguntas pero sin conocer el resultado de ellas hasta el final. En un momento aleatorio se te sacara una foto para demostrar que eres tu. Es necesario que respondas a todo y pulses finalizar, de no ser así, los datos no serán enviados."</li> </ul>         At the bottom, a dark box contains the text: "Se necesita una conexión a internet para entrar aquí".       </p>
<b>Resultado</b>	Correcto.

Tabla 51: Prueba del dominio de la aplicación 5



**Prueba 06**

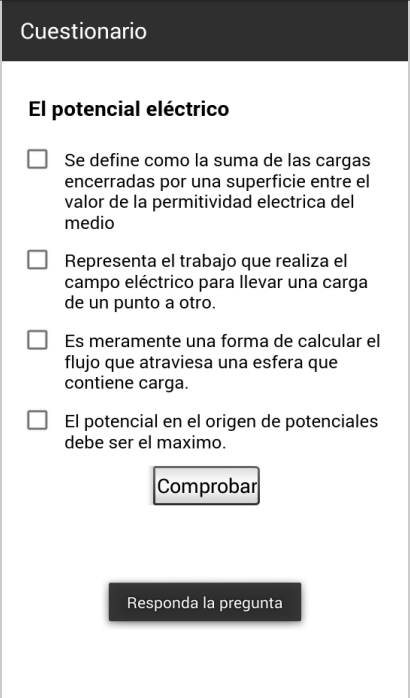
<b>Descripción</b>	Se intenta corregir una pregunta sin responderla.
<b>Salida esperada</b>	Error.
<b>Salida obtenida</b>	 <p>The screenshot shows a questionnaire titled 'Cuestionario' with the sub-heading 'El potencial eléctrico'. It contains four multiple-choice questions, each with an unchecked checkbox. The questions are:         <ul style="list-style-type: none"> <li>Se define como la suma de las cargas encerradas por una superficie entre el valor de la permitividad electrica del medio</li> <li>Representa el trabajo que realiza el campo eléctrico para llevar una carga de un punto a otro.</li> <li>Es meramente una forma de calcular el flujo que atraviesa una esfera que contiene carga.</li> <li>El potencial en el origen de potenciales debe ser el maximo.</li> </ul>         Below the questions is a 'Comprobar' button and a 'Responda la pregunta' button.       </p>
<b>Resultado</b>	Correcto.

Tabla 52: Prueba del dominio de la aplicación 6

**Prueba 07**

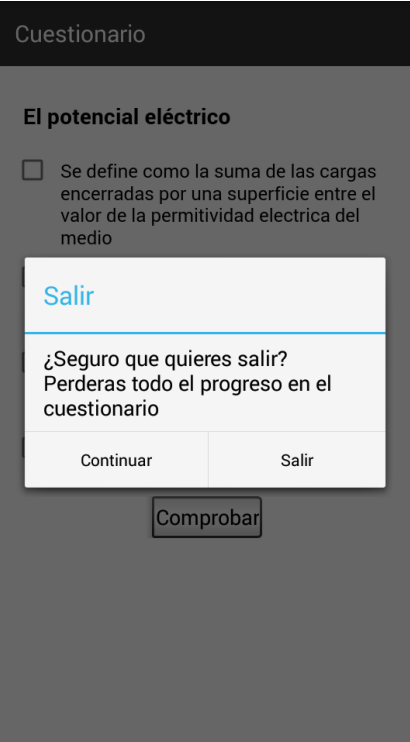
<b>Descripción</b>	Se intenta abandonar un cuestionario sin resolverlo
<b>Salida esperada</b>	Aviso.
<b>Salida obtenida</b>	 <p>The screenshot shows the same questionnaire as in Table 52. A confirmation dialog box is overlaid on the screen, titled 'Salir'. The dialog asks: '¿Seguro que quieres salir? Perderas todo el progreso en el cuestionario'. It has two buttons: 'Continuar' and 'Salir'. Below the dialog is a 'Comprobar' button.       </p>
<b>Resultado</b>	Correcto.

Tabla 53: Prueba del dominio de la aplicación 7

### Prueba 09

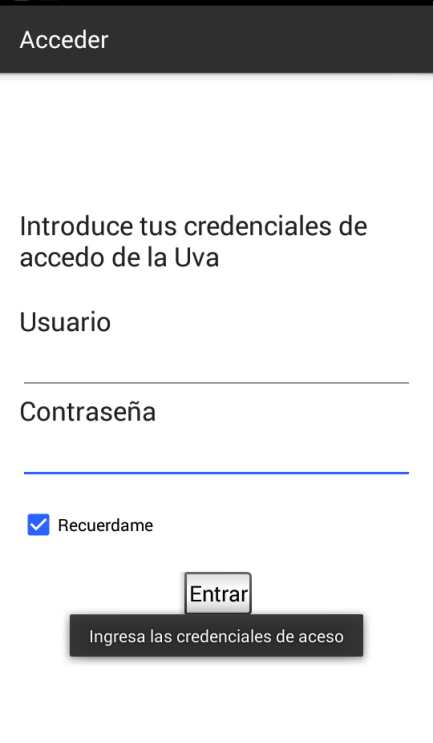
<b>Descripción</b>	Se intenta acceder al cuestionario de examen sin introducir datos de acceso
<b>Salida esperada</b>	Error.
<b>Salida obtenida</b>	
<b>Resultado</b>	Correcto.

Tabla 54: Prueba del dominio de la aplicación 9

### Prueba 10

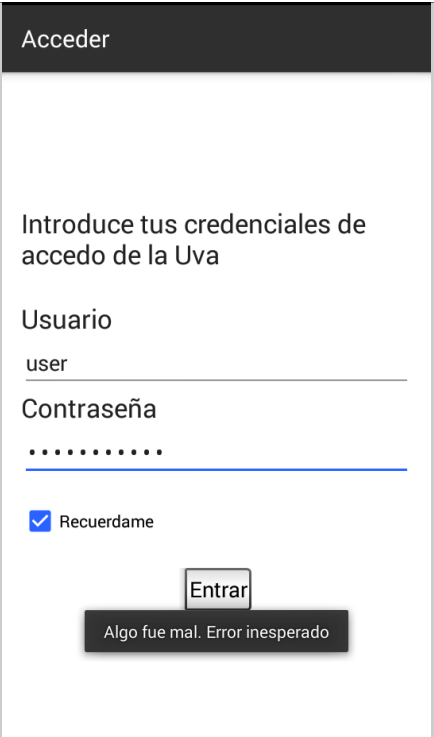
<b>Descripción</b>	Es imposible realizar la validación de los datos.
<b>Salida esperada</b>	Error.
<b>Salida obtenida</b>	
<b>Resultado</b>	Correcto.

Tabla 55: Prueba del dominio de la aplicación 10

**Prueba 11**

<b>Descripción</b>	La pregunta ha sido respondida incorrectamente.	
<b>Salida esperada</b>	Respuesta correcta en verde, falsa en rojo.	
<b>Salida obtenida</b>		<div style="background-color: #333; color: white; padding: 5px; text-align: center;">Cuestionario</div> <p><b>Las líneas de campo:</b></p> <ul style="list-style-type: none"> <li><input type="checkbox"/> No dependen de la intensidad del campo eléctrico solo muestran el sentido de este</li> <li><input type="checkbox"/> Pueden cortarse en el punto medio de la recta que une las dos cargas</li> <li><input checked="" type="checkbox"/> Su sentido es inverso al del campo eléctrico</li> <li><input type="checkbox"/> Son tangentes al campo eléctrico en todos los puntos.</li> </ul>
<b>Resultado</b>	Correcto.	

Tabla 56: Prueba del dominio de la aplicación 11

**Prueba 12**

<b>Descripción</b>	La pregunta ha sido respondida correctamente.	
<b>Salida esperada</b>	Respuesta correcta en verde.	
<b>Salida obtenida</b>		<div style="background-color: #333; color: white; padding: 5px; text-align: center;">Cuestionario</div> <p><b>El potencial eléctrico</b></p> <ul style="list-style-type: none"> <li><input type="checkbox"/> Se define como la suma de las cargas encerradas por una superficie entre el valor de la permitividad eléctrica del medio</li> <li><input checked="" type="checkbox"/> Representa el trabajo que realiza el campo eléctrico para llevar una carga de un punto a otro.</li> <li><input type="checkbox"/> Es meramente una forma de calcular el flujo que atraviesa una esfera que contiene carga.</li> <li><input type="checkbox"/> El potencial en el origen de potenciales debe ser el máximo.</li> </ul>
<b>Resultado</b>	Correcto.	

Tabla 57: Prueba del dominio de la aplicación 12

### Prueba 13

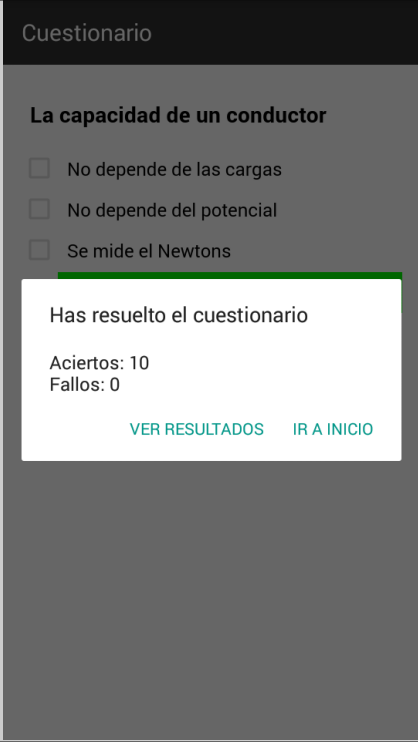
<b>Descripción</b>	El cuestionario de autoevaluación ha sido respondido.
<b>Salida esperada</b>	Dialogo con el resumen del intento
<b>Salida obtenida</b>	
<b>Resultado</b>	Correcto.

Tabla 58: Prueba del dominio de la aplicación 13

### Prueba 14

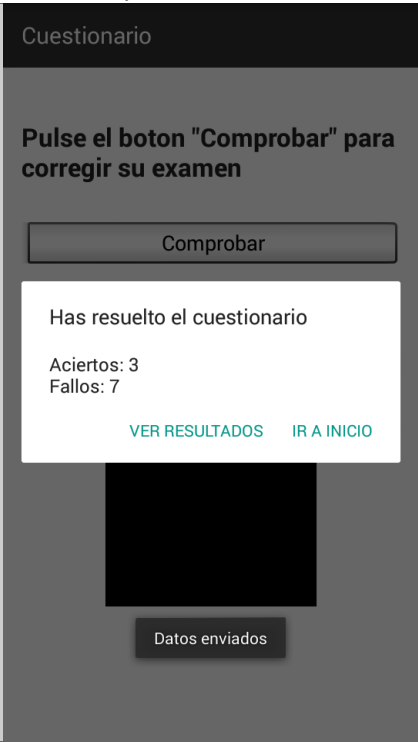
<b>Descripción</b>	El cuestionario de examen ha sido respondido.
<b>Salida esperada</b>	Dialogo con el resumen del intento y aviso de envío de datos
<b>Salida obtenida</b>	
<b>Resultado</b>	Correcto. Al principio fallo la conexión con el servidor de correo

Tabla 59: Prueba del dominio de la aplicación 14

**Prueba 16**

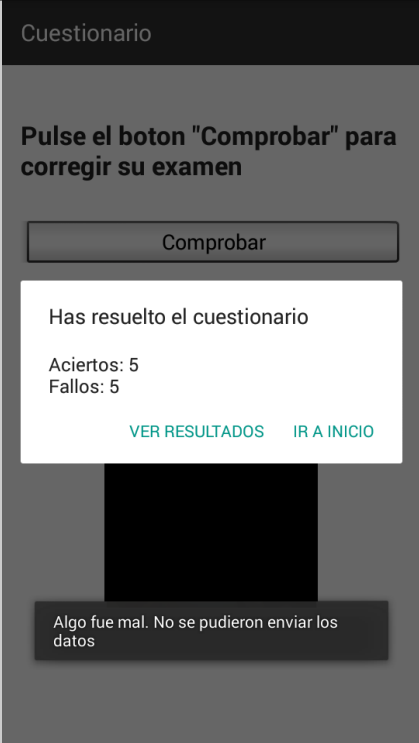
<b>Descripción</b>	El cuestionario de examen ha sido respondido, pero no se puede enviar los datos.
<b>Salida esperada</b>	Dialogo con el resumen del intento y aviso de error
<b>Salida obtenida</b>	
<b>Resultado</b>	Correcto.

Tabla 60: Prueba del dominio de la aplicación 16

Prueba 17

Descripción

Se añaden dos cargas positivas la de la derecha con mayor carga

Salida esperada

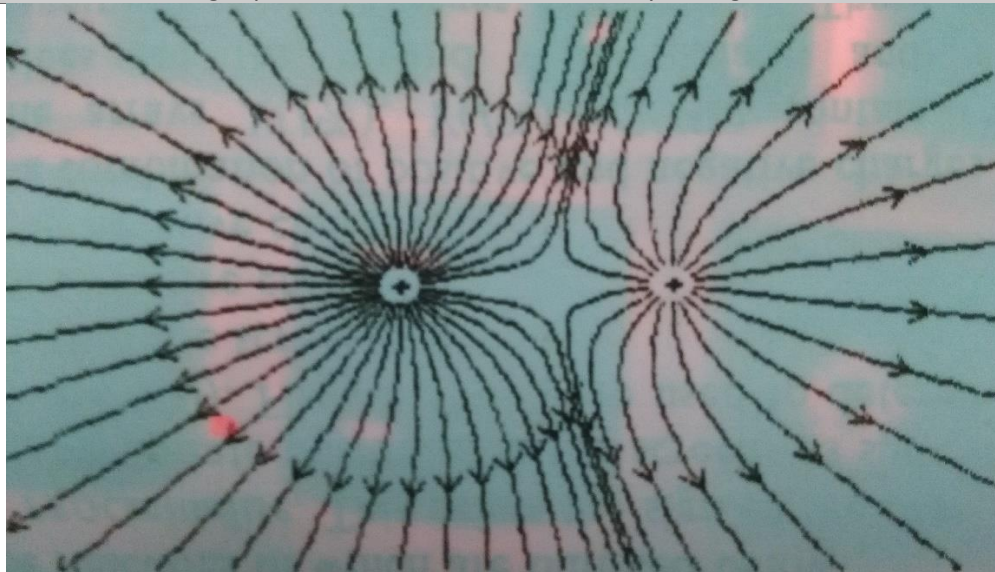
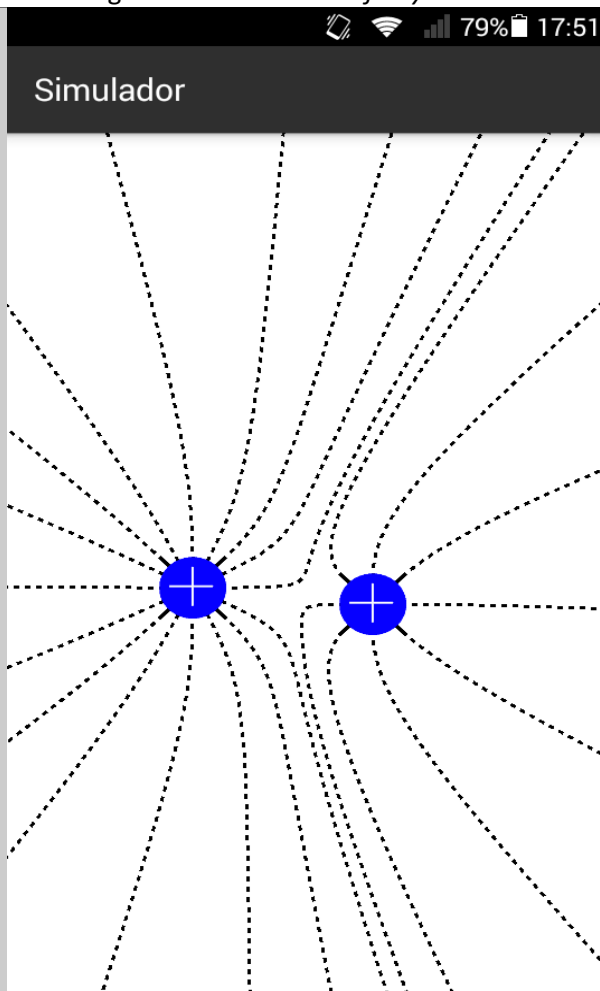


Imagen de *The Institute of Physics* ©1985

Salida obtenida



Resultado

Correcto.

Tabla 61: Prueba del dominio de la aplicación 17

**Prueba 18**

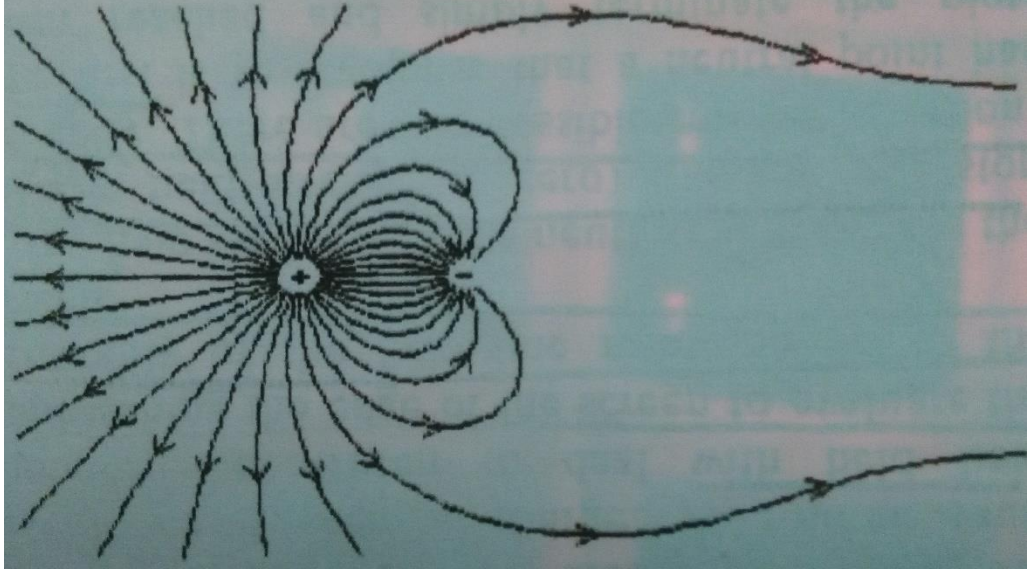
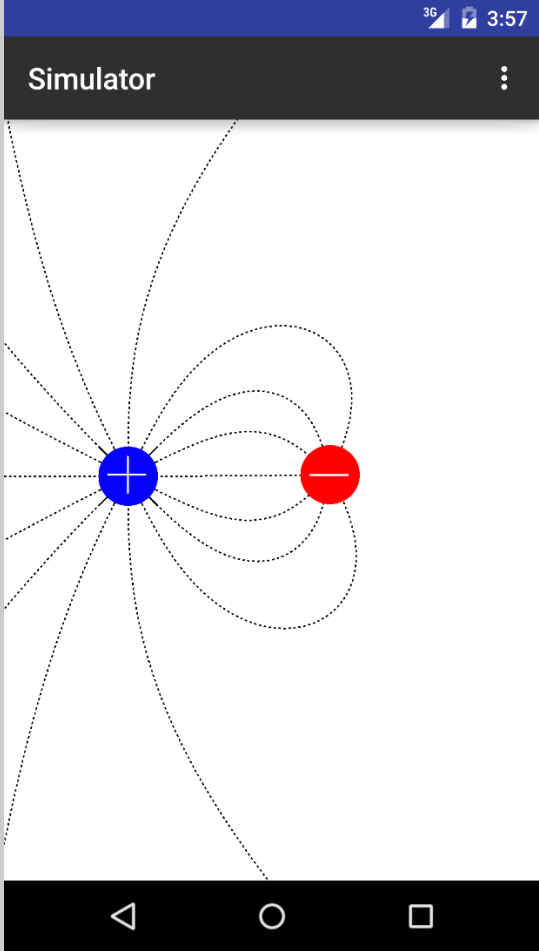
<b>Descripción</b>	Se añaden dos una carga positiva y una negativa
<b>Salida esperada</b>	 <p>Imagen de <i>The Institute of Physics</i> ©1985</p>
<b>Salida obtenida</b>	 <p>Simulador</p>
<b>Resultado</b>	Correcto.

Tabla 62: Prueba del dominio de la aplicación 18

### Prueba 19

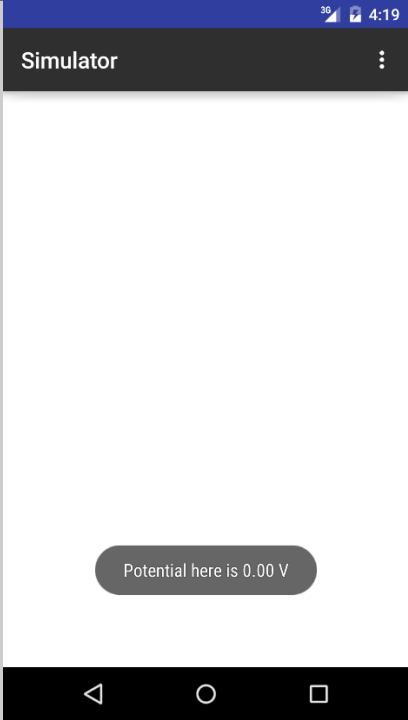
<b>Descripción</b>	Calcular potencial en un entorno sin cargas.
<b>Salida esperada</b>	0,00 V
<b>Salida obtenida</b>	 A screenshot of a mobile application titled "Simulator". The interface is white with a dark blue header. At the bottom, there is a dark grey button with the text "Potential here is 0.00 V". The status bar at the top shows signal strength, 3G, battery, and the time 4:19. The Android navigation bar is visible at the bottom.
<b>Resultado</b>	Correcto.

Tabla 63: Prueba del dominio de la aplicación 19

### Prueba 20

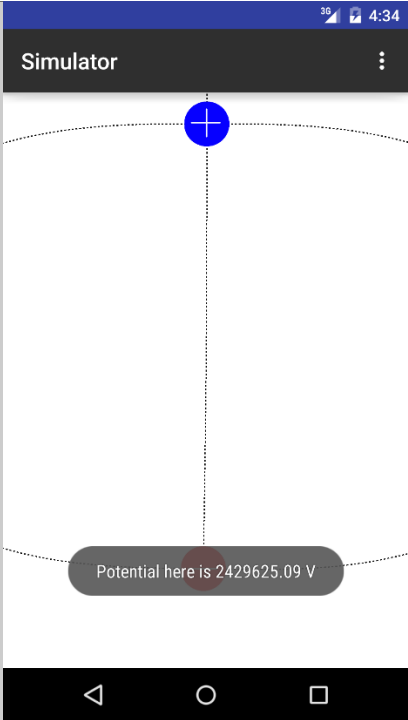
<b>Descripción</b>	Calcular potencial en un entorno con una carga positiva en (542,84) y una negativa en (533,1264) cargadas con un culombio.
<b>Salida esperada</b>	2429612.641 V
<b>Salida obtenida</b>	 A screenshot of the "Simulator" app. It shows a blue circle with a white plus sign at the top, representing a positive charge. A vertical dotted line extends downwards from this charge to a dark grey button at the bottom that displays "Potential here is 2429625.09 V". The status bar shows 3G, battery, and the time 4:34. The Android navigation bar is at the bottom.
<b>Resultado</b>	Correcto (Yo solo he tomado dos cifras significativas en los cálculos).

Tabla 64: Prueba del dominio de la aplicación 20



## Pruebas de la interfaz gráfica.

### Prueba GUI 1

<b>Descripción</b>	En el menú principal se pulsa el botón teoría.
<b>Evento</b>	Pulsar un botón.
<b>Salida esperada</b>	Lanzar la actividad que muestra la teoría (TeoriaActivity).
<b>Resultado</b>	Correcto.

Tabla 65: Prueba GUI 1

### Prueba GUI 2

<b>Descripción</b>	En el menú principal se pulsa el botón cuestionario.
<b>Evento</b>	Pulsar un botón.
<b>Salida esperada</b>	Lanzar la actividad que permite elegir el tipo de cuestionario (PrevioCuestionarioActivity).
<b>Resultado</b>	Correcto.

Tabla 66: Prueba GUI 2

### Prueba GUI 3

<b>Descripción</b>	En el menú de cuestionarios se pulsa autoevaluación
<b>Evento</b>	Pulsar un botón.
<b>Salida esperada</b>	Lanzar la actividad de cuestionarios (CuestionarioActivity) con la configuración de autoevaluación (es distinguible porque cada pregunta tiene un botón responder).
<b>Resultado</b>	Correcto.

Tabla 67: Prueba GUI 3

### Prueba GUI 4

<b>Descripción</b>	En el menú de cuestionarios se pulsa examen.
<b>Evento</b>	Pulsar un botón.
<b>Salida esperada</b>	Lanzar la actividad que en la que se valida el usuario (ValidarUsuarioActivity).
<b>Resultado</b>	Correcto.

Tabla 68: Prueba GUI 4

### Prueba GUI 5

<b>Descripción</b>	En la pantalla de validación se pulsa acceder con los datos correctos.
<b>Evento</b>	Pulsar un botón.
<b>Salida esperada</b>	Lanzar la actividad de cuestionarios (CuestionarioActivity) con la configuración de examen (es distinguible porque cada pregunta no tiene un botón responder).
<b>Resultado</b>	Correcto.

Tabla 69: Prueba GUI 5

### Prueba GUI 6

<b>Descripción</b>	En la pantalla de validación se pulsa acceder con los datos correctos.
<b>Evento</b>	Pulsar un botón.
<b>Salida esperada</b>	Lanzar la actividad de cuestionarios (CuestionarioActivity) con la configuración de examen (es distinguible porque cada pregunta no tiene un botón responder).
<b>Resultado</b>	Correcto.

Tabla 70: Prueba GUI 6

### Prueba GUI 7

<b>Descripción</b>	Durante la resolución de un cuestionario se pulsa atrás.
<b>Evento</b>	Pulsar back.
<b>Salida esperada</b>	Solicitar confirmación de salida.
<b>Resultado</b>	Correcto.

*Tabla 71: Prueba GUI 7*

### Prueba GUI 8

<b>Descripción</b>	Al terminar de responder un examen se pulsa comprobar.
<b>Evento</b>	Pulsar un botón.
<b>Salida esperada</b>	Mostrar resultados del cuestionario y el resultado del envío.
<b>Resultado</b>	Correcto.

*Tabla 72: Prueba GUI 8*

### Prueba GUI 9

<b>Descripción</b>	En un cuestionario de examen finalizado se pulsa salir.
<b>Evento</b>	Pulsar un botón.
<b>Salida esperada</b>	Ir a la pantalla de validación.
<b>Resultado</b>	Correcto.

*Tabla 73 Prueba GUI 9*

### Prueba GUI 10

<b>Descripción</b>	En el layout de la simulación se pulsa la pantalla.
<b>Evento</b>	Pulsar un layout.
<b>Salida esperada</b>	Mostrar opciones para crear una carga o calcular potencial en ese punto.
<b>Resultado</b>	Correcto.

*Tabla 74: Prueba GUI 10*

### Prueba GUI 11

<b>Descripción</b>	En la simulación, se hace una pulsación larga sobre la carga.
<b>Evento</b>	Pulsar sobre un ImageView.
<b>Salida esperada</b>	Mostrar opciones para esa carga.
<b>Resultado</b>	Correcto.

*Tabla 75: Prueba GUI 11*

### Prueba GUI 12

<b>Descripción</b>	En la simulación, pulsa reiniciar.
<b>Evento</b>	Pulsar una opción del menú.
<b>Salida esperada</b>	Se vacía el layout
<b>Resultado</b>	Correcto.

*Tabla 76: Prueba GUI 12*

# Capítulo VII: Conclusiones.

## Conclusiones.

Una vez finalizado el desarrollo del proyecto, puedo afirmar que estoy muy satisfecho con el resultado del mismo, pues cumple con todos los requisitos que se han indicado y además ha sido desarrollado en el periodo de un cuatrimestre lectivo.

Este proyecto me ha servido para ampliar mis conocimientos sobre tecnologías que ya conocía como Java o Android y descubrir otras nuevas como el API de Dropbox, MathJax o AES, tecnologías que en el caso del API de Dropbox o AES nunca había empleado y en el caso de MathJax directamente lo desconocía.

Para poder adquirir estos conocimientos a la velocidad suficiente para realizar este proyecto, cabe destacar la importancia de lo aprendido durante el estudio del Grado en Ingeniería Informática, pues ha servido como una base sólida sobre la que asentar todo este aprendizaje.

Pero no solo a nivel de desarrollo ha sido importante la formación de estos últimos cinco años. Los conocimientos sobre ingeniería de software y planificación de proyectos son fundamentales para reducir la incertidumbre que genera un problema nuevo, pues sabemos que lanzarse a programar sin haber analizado el problema y planificado las acciones a realizar, nos va a llevar seguramente al fracaso.

Precisamente en la planificación considero que ha estado el mayor desafío para mí. Pese a tener que usar tecnologías para mí nuevas como MathJax, el API de Dropbox o la encriptación usando AES, esto solo me requirió tener que extender mis conocimientos en el lenguaje Java, conocido por mí desde hace bastante tiempo. Por el contrario, yo nunca había empleado una metodología ágil como proceso de desarrollo de software, pero la idoneidad de este proyecto para aplicar un proceso así me hizo tomar esta decisión. Tener disponible al cliente (tutor), unos requisitos que pudieran cambiar y una aplicación que permitiera generar ejecutables intermedios, son los principios del desarrollo ágil y de ahí mi decisión.

Las decisiones tomadas, las tecnologías aprendidas, la experiencia adquirida y los resultados obtenidos son los motivos de la satisfacción con el proyecto que indicaba en el primer párrafo. Este proyecto me parece que ha sido la forma idónea de poner un punto (desde luego no un punto y final) a mi formación académica.

## Mejoras futuras.

Como dije en la sección anterior, el resultado de este proyecto satisface todos los requisitos que se habían descrito inicialmente. Pese a eso, debo reconocer que hay dos cosas que me hubieran gustado poder realizar, estas cosas son la generación de este software para las otras principales plataformas móviles y la paralelización del algoritmo de cálculos de líneas.

### Generación del software multiplataforma.

Este software ha sido construido para Android utilizando la herramienta de desarrollo Android Studio. Una mejora futura sería construir esta aplicación para las principales plataformas móviles. Para ello, podemos utilizar herramientas de desarrollo pensadas para cada plataforma y realizar desarrollos independientes o utilizar alguna plataforma que, una vez generado el código en un lenguaje específico permita obtener versiones para distintos sistemas operativos.

Una de las principales trabas que puede encontrarse al hacer esto, es que solo el análisis es reutilizable en esta nueva construcción, pues el diseño y la implementación dependen en gran medida de la tecnología a utilizar, lo que implica tener que repetirlos.

Paralelizar el algoritmo de cálculo de líneas de campo.

Como dijimos en la implementación, el algoritmo de cálculo de líneas se basa en tomar una circunferencia de un radio concreto tomando como centro un punto dado (el anterior al que se va a calcular) y calcular el vector de campo eléctrico para saber qué punto de esta circunferencia es el siguiente.

Dicho de otra forma, para calcular una línea de campo necesitamos conocer el valor y la posición de cada carga, no existe ninguna dependencia entre la línea a calcular y las calculadas anteriormente. Estos valores se mantienen inmutables durante el cálculo de todas las líneas, por lo tanto, podríamos calcular las líneas en distintos procesadores y al acabar el cálculo, el hilo principal pintaría dichas líneas (En Android solo el hilo principal puede usar canvas).



# Capítulo VIII: Bibliografía.

## Bibliografía sobre planificación de proyectos.

[1] Henrik Kniberg. Scrum y XP desde las trincheras. Cómo hacemos Scrum. 2007 C4Media Inc

[2] <https://proyectosagiles.org/>

[3] Gobierno de España. Inteco. Curso de Desarrollo Ágil. Junio del 2009.

## Bibliografía sobre la aplicación complementaria.

[4] <http://stackoverflow.com/questions/9119481/how-to-present-a-simple-alert-message-in-java>. Fecha de la última consulta: 10/02/2016

[5] <http://webdiis.unizar.es/~ezpeleta/lib/exe/fetch.php?media=misdatos:pc:introjava.pdf>. Fecha de la última consulta: 10/02/2016

[6] <http://stackoverflow.com/questions/15880844/how-to-limit-jspinner>. Fecha de la última consulta: 10/02/2016

[7] <http://www.iteramos.com/pregunta/1928/estandar-de-manera-concisa-para-copiar-un-archivo-en-java>. Fecha de la última consulta: 13/03/2016

[8] <http://lineadecodigo.com/java/conocer-el-sistema-operativo-con-java>. Fecha de la última consulta: 19/03/2016

[9] <http://stackoverflow.com/questions/16214480/adding-elements-to-a-jlist>. Fecha de la última consulta: 19/03/2016

[10] <http://stackoverflow.com/questions/5212983/jlist-add-remove-item>. Fecha de la última consulta: 22/03/2016

[11] [http://www.javacode.tech/2014/07/uso-basico-de-elementos-swing-con\\_17.html](http://www.javacode.tech/2014/07/uso-basico-de-elementos-swing-con_17.html). Fecha de la última consulta: 22/03/2016

## Bibliografía sobre Android.

[12] <http://www.iteramos.com/pregunta/23004/de-android-webview-y-loaddata>. Fecha de la última consulta: 15/03/2016

[13] <http://stackoverflow.com/questions/3687065/textview-settextsize-behaves-abnormally-how-to-set-text-size-of-textview-dynam>. Fecha de la última consulta: 15/03/2016

[14] <http://developer.android.com/intl/es/guide/topics/ui/notifiers/toasts.html>. Fecha de la última consulta: 15/03/2016

[15] <http://www.forsdelweb.com/f45/rellenar-array-con-numeros-aleatorios-sin-repetir-498606>. Fecha de la última consulta: 16/03/2016

[16] <http://lineadecodigo.com/java/numero-aleatorio-en-java>. Fecha de la última consulta: 16/04/2016



- [17] <http://developer.android.com/intl/es/guide/topics/ui/controls/radiobutton.html>. Fecha de la última consulta: 16/02/2016
- [18] <http://www.sgoliver.net/blog/notificaciones-en-android-iii-dialogos>. Fecha de la última consulta: 16/03/2016
- [19] <http://stackoverflow.com/questions/3060837/validate-smtp-server-credentials-using-java-without-actually-sending-mail>. Fecha de la última consulta: 18/03/2016
- [20] <http://stackoverflow.com/questions/3961589/android-webview-and-loaddata>. Fecha de la última consulta: 20/3/2016
- [21] <http://stackoverflow.com/questions/5474089/how-to-check-currently-internet-connection-is-available-or-not-in-android>. Fecha de la última consulta: 22/05/2016
- [22] <http://stackoverflow.com/questions/14398778/android-front-camera-api>. Fecha de la última consulta: 24/03/2016
- [23] <http://androideity.com/2011/10/08/hello-camera>. Fecha de la última consulta: 24/03/2016
- [24] <https://developer.android.com/training/basics/data-storage/shared-preferences.html#ReadSharedPreference>. Fecha de la última consulta: 24/05/2016
- [25] <http://www.chuidiang.com/java/herramientas/javamail/enviar-adjuntos-javamail.php>. Fecha de la última consulta: 25/05/2016
- [26] <http://stackoverflow.com/questions/7263291/viewpager-pageradapter-not-updating-the-view>. Fecha de la última consulta: 29/03/2016
- [27] <http://www.iteramos.com/pregunta/4074/como-ocultar-la-barra-de-titulo-para-una-actividad-en-xml-con-tema-personalizado-existente>. Fecha de la última consulta: 1/04/2016
- [28] <http://stackoverflow.com/questions/16848058/oncontextmenuclosed-not-implemented-in-fragment>. Fecha de la última consulta: 3/04/2016
- [29] <http://www.comocreartuweb.com/curso-de-html/curso-html-introduccion/tipo-de-codificacion.html>. Fecha de la última consulta: 4/04/2016
- [30] <http://stackoverflow.com/questions/9606046/android-rotate-picture-before-saving?q=1>. Fecha de la última consulta: 4/04/2016
- [31] <http://www.iteramos.com/pregunta/12434/mostrar-imagen-vista-desde-la-ruta-del-archivo-en-android>. Fecha de la última consulta: 19/04/2016
- [32] <http://dropbox.github.io/dropbox-sdk-java/api-docs/v1.7.x>. Fecha de la última consulta: 20/04/2016
- [33] <https://www.dropbox.com/developers-v1/sync/sdks/android>. Fecha de la última consulta: 29/04/2016
- [35] <http://stackoverflow.com/questions/28942377/how-to-use-the-dropbox-android-core-api-on-android-studio>, Fecha de la última consulta: 2/05/2016
- [36] <https://www.dropbox.com/developers/documentation/java#tutorial>. Fecha de la última consulta: 2/05/2016
- [37] <http://javapapers.com/java/dropbox-java-api-tutorial>. Fecha de la última consulta: 3/05/2016
- [38] <http://stackoverflow.com/questions/15171654/set-imageview-width-and-height-programmatically-on-a-widget>. Fecha de la última consulta: 27/05/2016

[39] <http://www.sgoliver.net/blog/menus-en-android-ii-menus-contextuales>. Fecha de la última consulta: 29/05/2016

[40] <http://stackoverflow.com/questions/4784336/two-decimal-places-using-printf>. Fecha de la última consulta: 29/05/2016

[41] <https://www.mathjax.org/>. Fecha de la última consulta: 07/07/2016

[42] [https://es.wikipedia.org/wiki/Advanced\\_Encryption\\_Standard](https://es.wikipedia.org/wiki/Advanced_Encryption_Standard). Fecha de la última consulta: 07/07/2016

[42] Tomás Gironés, Jesús. El gran libro de Android. 2016. Macrocombo

[43] Tomás Gironés, Jesús. El gran libro de Android. 2014. Macrocombo

[44] Tomás Gironés, Jesús. El gran libro de Android Avanzado. 2014. Macrocombo

### Bibliografía sobre Ingeniería del software.

[45] Larman, Craig. UML y patrones: introducción al análisis y diseño orientado a objetos y al proceso unificado. 2006. Prentice-Hall

[46] Sommerville, Ian. Ingeniería del software. 2006. Pearson

[47] <http://ocw.usal.es/enseanzas-tecnicas/ingenieria-del-software/contenidos/Tema5-Principiosdeldisenodelsoftware-1pp.pdf> Fecha de la última consulta: 25/05/2016

### Bibliografía sobre Física

[48] Tipler, Paul Allen. Física para la ciencia y la tecnología 5ª Edición. 2004. Reverte

[49] González Delgado, Manuel Ángel. Tema 1, Campo eléctrico en el vacío. Universidad de Valladolid

[50] González Delgado, Manuel Ángel. Tema 2, Campo eléctrico en la materia. Universidad de Valladolid

# Anexo I: Manuales

## Manual de instalación.

Antes de proceder a la instalación de la aplicación, debemos tener en cuenta que esta no se va a realizar desde una tienda, por ejemplo, Google Play (al menos de momento) y por lo tanto debemos indicar a nuestro dispositivo que debe instalar aplicaciones de “orígenes desconocidos”. Para ello vaya a Menú → Ajustes → Seguridad y active la pestaña “orígenes desconocidos”. El resultado debe ser algo así:

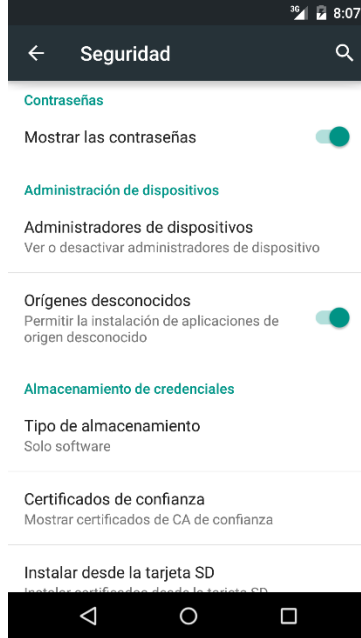


Ilustración 85: Habilitar orígenes desconocidos

Una vez hecho esto, vaya a la ubicación donde guardo CamposElectricos.apk y pulse sobre ella. El teléfono le ofrecerá un instalador, donde lo primero que se muestran son los permisos que se deben conceder a la aplicación para ser instalada. Si está de acuerdo con la solicitud de permisos pulse instalar. Su pantalla será similar a esta:

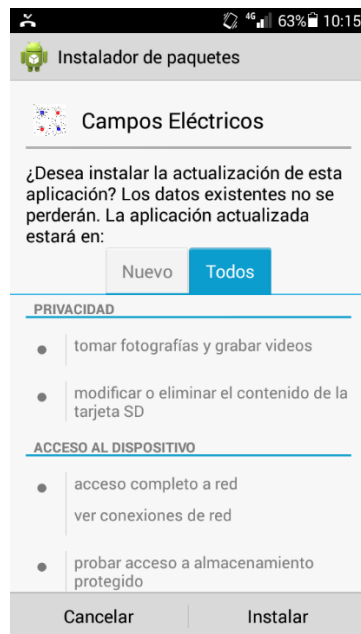


Ilustración 86: Pantalla del instalador

Si decide pulsar instalar, comenzará el proceso de instalación, espere unos segundos mientras el teléfono instala la aplicación. Su pantalla mostrara lo siguiente:

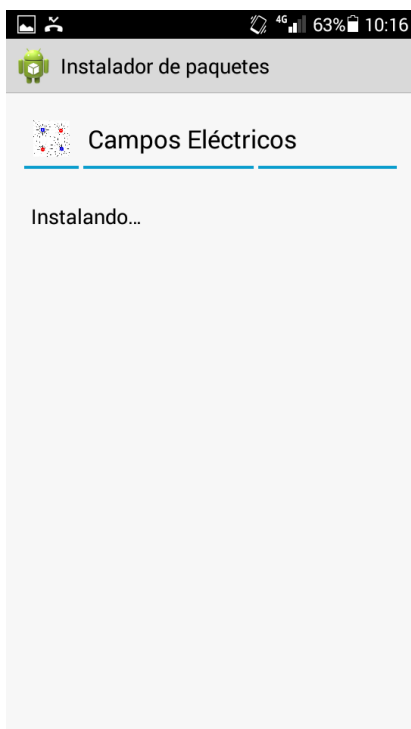


Ilustración 87: Pantalla durante la instalación

Cuando la aplicación este instalada en su dispositivo, se le mostrara una pantalla donde puede salir del asistente de instalación o abrir la aplicación directamente con un mensaje que le informe de que la instalación se ha completado. Será algo así:

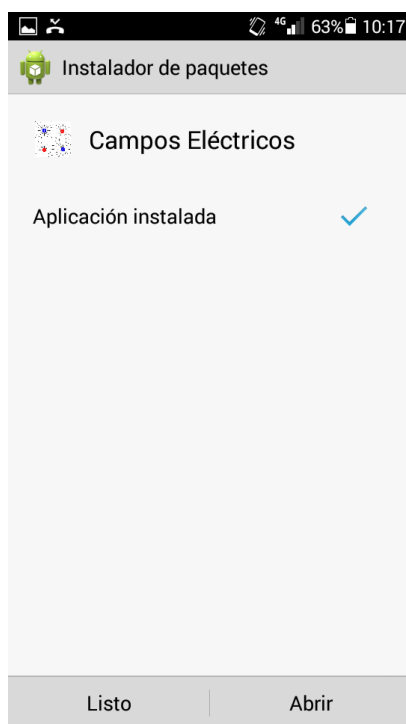


Ilustración 88: Pantalla de fin de instalación

Al abrir la aplicación, lo primero que vemos es el menú principal. Este menú nos permite acceder a la teoría, realizar los cuestionarios o utilizar la simulación. Aparte de esto, en esta pantalla también podemos actualizar los contenidos o consultar los datos de la aplicación. Para ello hay que pulsar sobre los tres puntitos (o botón menú, depende del teléfono) y seleccionar una de estas opciones. La pantalla principal se distribuye de esta forma:

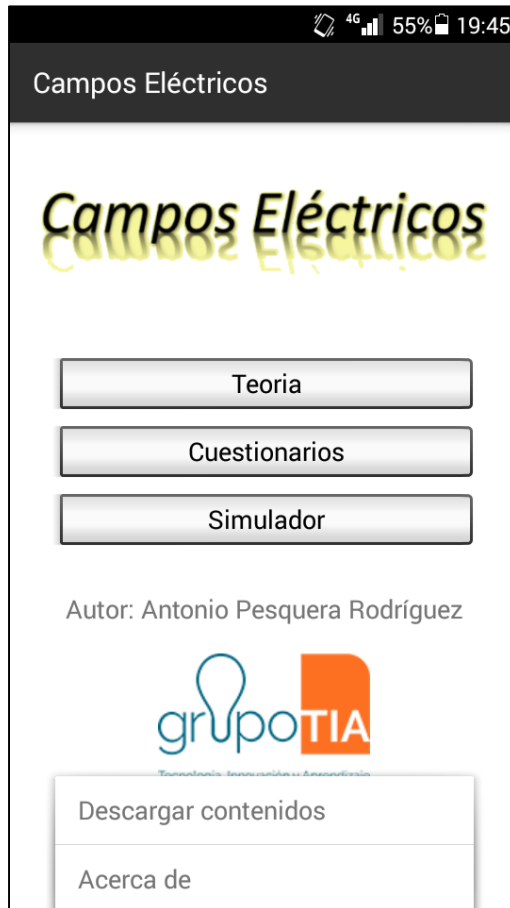


Ilustración 89: Aspecto de la pantalla principal con el menú desplegado

Visualizar la teoría.

Para visualizar la teoría, pulse en el botón “teoría”. A continuación, el sistema le mostrará la primera página. Para moverse entre todas las paginas, desplace su dedo hacia la izquierda para avanzar y hacia la derecha para retroceder. Nada más cargar una página puede que vea un mensaje similar al marcado en rojo en la ilustración 90. Se trata de la resolución de las formulas mediante MathJax. Esta resolución precisa de comunicación con un servicio externo, de ahí que se recomiende acceder a la teoría con el terminal conectado a internet.

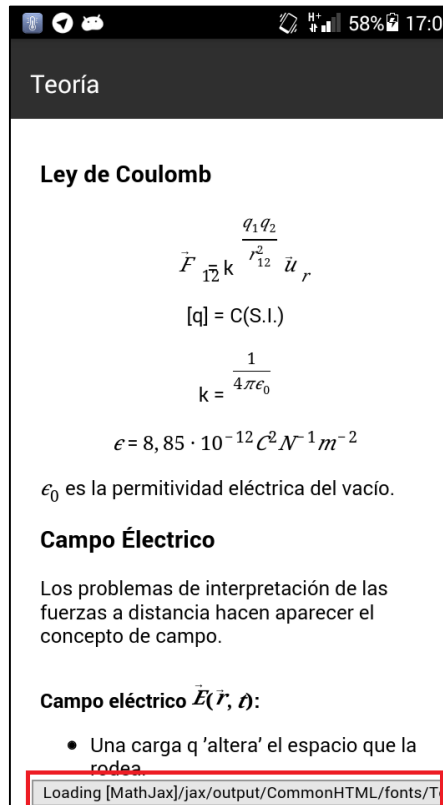


Ilustración 90: Pantalla de teoría

Realizar cuestionarios de autoevaluación.

Para acceder a las funciones de evaluación, seleccione la opción “cuestionarios” del menú principal (el de la ilustración 89). A continuación, se le mostrara un menú como el siguiente.

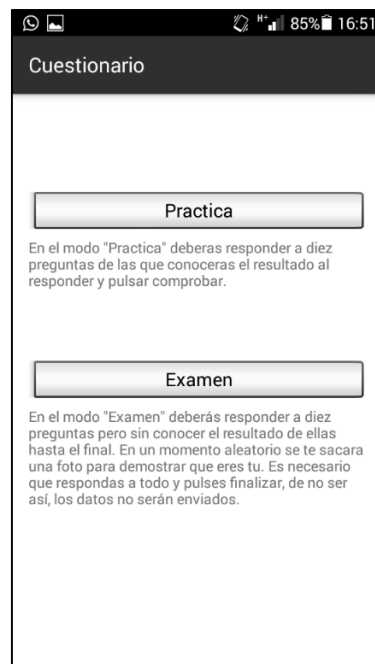


Ilustración 91: Pantalla de selección de cuestionario

En este menú, seleccione “autoevaluación”, a continuación, el sistema le mostrará la primera pregunta del cuestionario. Desplace su dedo hacia la izquierda para ver la siguiente pregunta y hacia la derecha para ver la anterior. El orden de respuesta es indiferente.

Para responder una pregunta seleccione una respuesta, a continuación, pulse “comprobar” para conocer el resultado de su intento. La respuesta correcta aparecerá en verde y la equivocada, si procede, en rojo. Si pulsa “comprobar” sin seleccionar una respuesta obtendrá un mensaje de error.

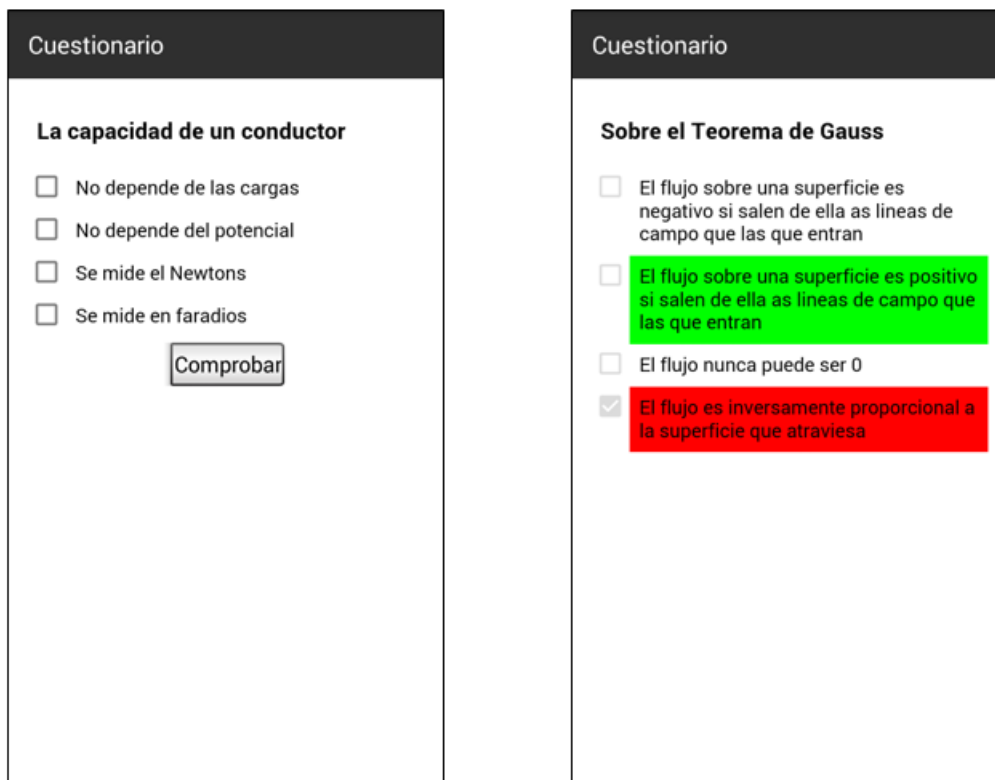


Ilustración 92: Pregunta sin responder y respondida

Una vez haya respondido y corregido las diez preguntas que componen el cuestionario, se le mostrará un dialogo con el resumen del intento. Este dialogo le permitirá además revisar el intento o salir.

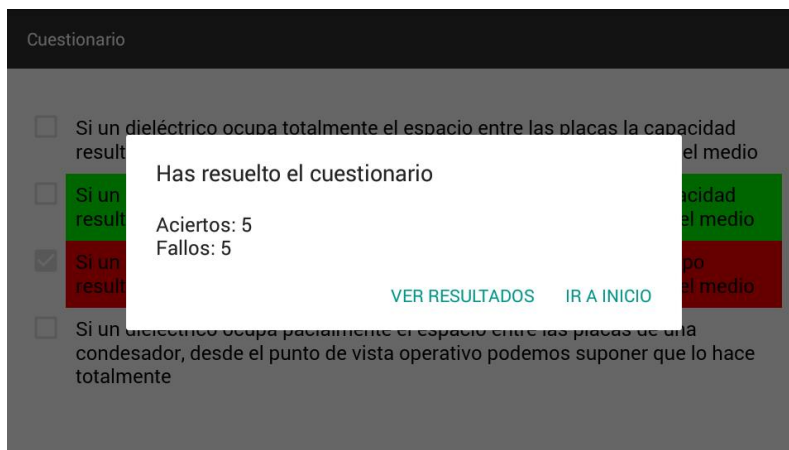


Ilustración 93: Resumen del intento



Tal y como ocurre en la teoría, es posible que aparezca un mensaje de carga de MathJax.

Realizar cuestionario de examen.

Para realizar un cuestionario de examen, pulse “Cuestionarios” en el menú principal (ilustración 89), a continuación, seleccione “Examen” como tipo de cuestionario en el submenú de cuestionarios (ilustración 91).

La siguiente pantalla que se muestra es la de la validación del usuario. Para acceder al cuestionario de examen introduzca sus credenciales de acceso, estas coinciden con las de su correo proporcionado por la Universidad de Valladolid y pulse “Entrar”, tras unos segundos, accederá al cuestionario u obtendrá una información de la imposibilidad de acceder.

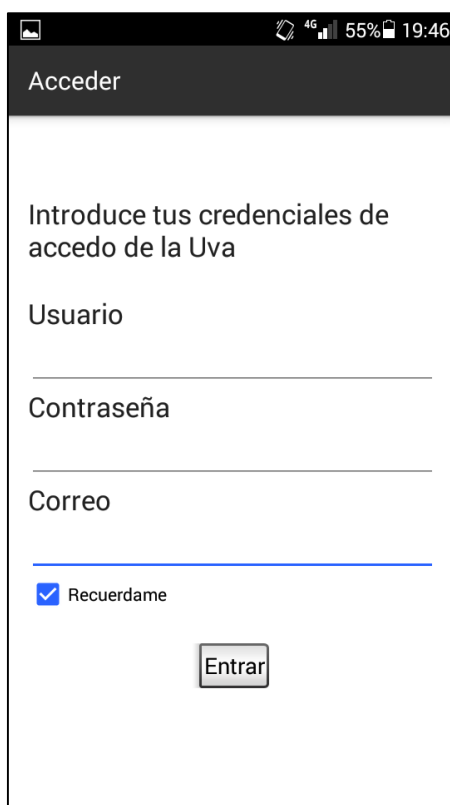
The image shows a mobile application screen for user login. At the top, there is a status bar with icons for signal strength, 4G connectivity, 55% battery, and the time 19:46. Below the status bar is a dark header with the word "Acceder" in white. The main content area is white and contains the text "Introduce tus credenciales de acceso de la Uva". There are three input fields: "Usuario", "Contraseña", and "Correo", each with a horizontal line below it. Below the "Correo" field is a blue checkmark icon followed by the text "Recuérdame". At the bottom center, there is a button labeled "Entrar".

Ilustración 94: Pantalla de validación de usuarios

Una vez ha accedido al cuestionario verá que las pantallas con las preguntas son similares a las de ilustración 93 salvo que no hay botón “comprobar”. Para realizar el cuestionario, responda las diez preguntas (tenga en cuenta que en una de ellas se le realizara una foto, no tape el objetivo) y avance a la última página, a continuación, pulse “Comprobar”. Si no ha respondido a todas las preguntas se le mostrara un mensaje de error.

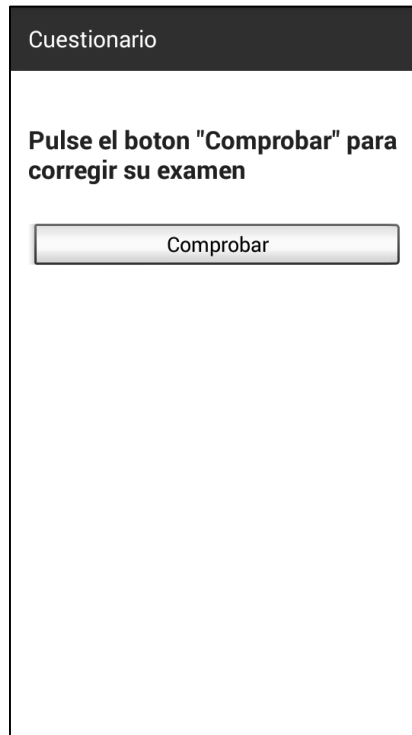


Ilustración 95: Pantalla de corrección

Una vez ha pulsado comprobar, espere unos segundos mientras se envían los resultados al profesor, a continuación, obtendrá una pantalla similar a la de la ilustración 96 con el resumen del intento que le permitirá revisar el intento o salir. Si revisa el intento podrá ver la foto que se le ha tomado y que ha sido enviada a su profesor.

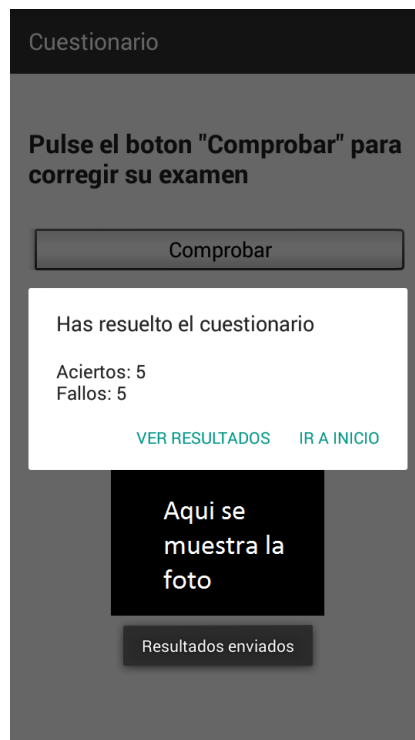


Ilustración 96: Pantalla de fin de examen

Simulación.

Para realizar un cuestionario de examen, pulse “Simulación” en el menú principal (ilustración 89), a continuación, se le mostrara un espacio vacío. Este espacio reacciona en tiempo real a sus decisiones, por lo que no hay botón de activación, en el momento en que interactúe con el comenzara a responder.

Si pulsa en el espacio, el sistema le ofrecerá calcular el potencial allí o colocar una carga. En caso de solicitar lo primero, el sistema le ofrecerá un mensaje con el potencial eléctrico en voltios en ese punto. En caso de querer crear una carga en ese punto el sistema le mostrara un dialogo para recoger los datos.

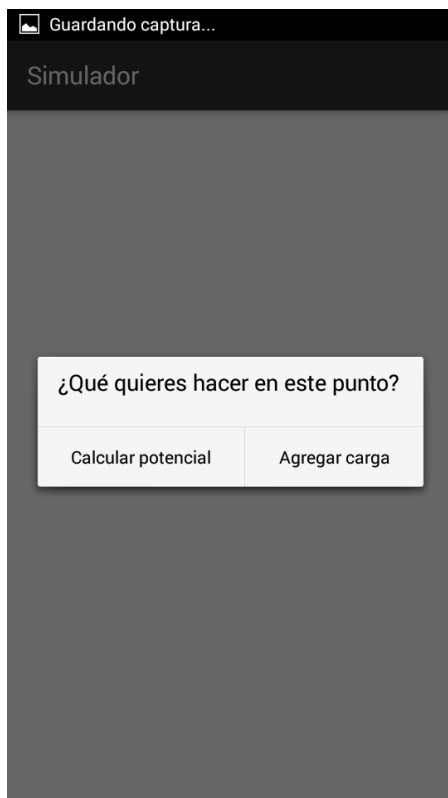


Ilustración 97: Opciones en un punto de la simulación

Tenga en cuenta que si crear una en el punto donde usted pulsa implica dejar parte de una carga fuera, el sistema se lo impedirá, exactamente igual que si intenta solapar dos cargas.

Una vez ha pulsado “Agregar carga”, el sistema le mostrará un dialogo que le permitirá introducir el valor de la carga y el signo. El valor es un número entero contenido entre 1 y 50, los dos contenidos en el rango. Si selecciona algo mayor que 50, el sistema fijara este valor automáticamente. Si selecciona 0 el sistema fijara 1 como valor. A continuación, se muestra la pantalla de creación de carga.

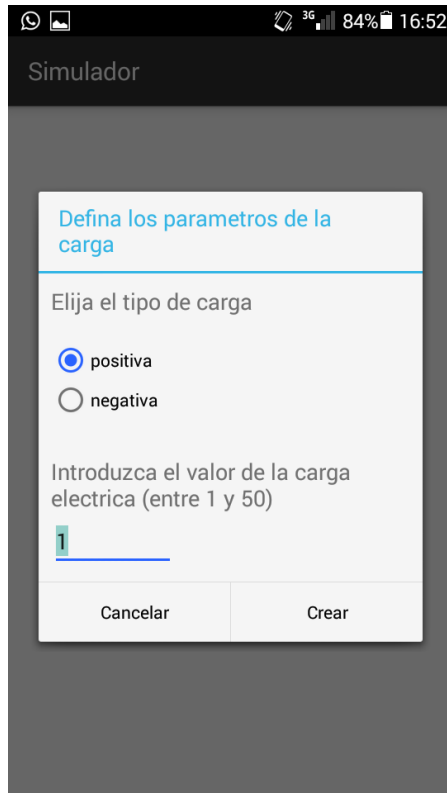


Ilustración 98: Diálogo de introducción de una carga

Si desea eliminar, editar o conocer las características de una carga en concreto, realice una pulsación larga sobre la carga (más de un segundo) y a continuación, se le mostrará un menú con estas opciones.



Ilustración 99: Opciones de una carga

Si desea volver a la situación inicial no es necesario que salga y entre de nuevo a la simulación, sencillamente pulse menú (botón menú o los tres puntitos en la parte superior derecha) y después pulse “reiniciar”.

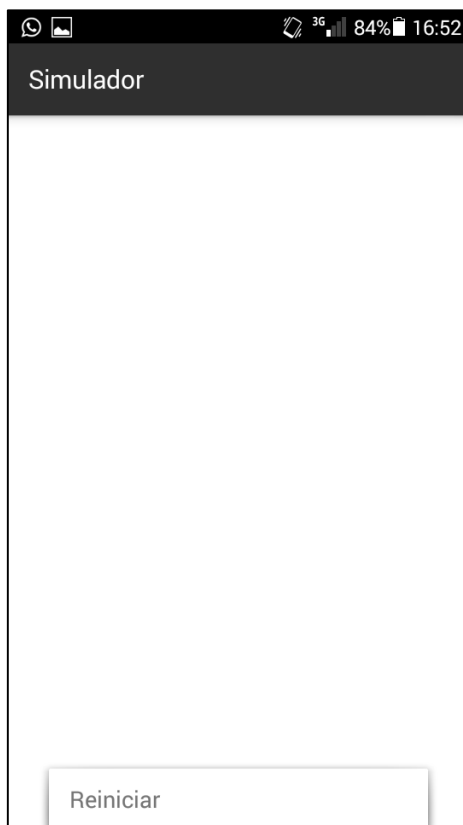


Ilustración 100: Opción de reinicio de la simulación

## Manual de Usuario. Gestor de Contenidos.

Gestor de Contenidos es una aplicación complementaria, a Campos Eléctricos que nos permite generar cuestionarios y páginas de teoría. Este programa está escrito en java, lo que nos permite ejecutarlo en cualquier computador que tenga instalada la Máquina Virtual Java.

Al iniciar el programa, se mostrará un menú como el de la Ilustración 101. Este menú nos permitirá acceder a la generación de páginas de teoría o cuestionarios.

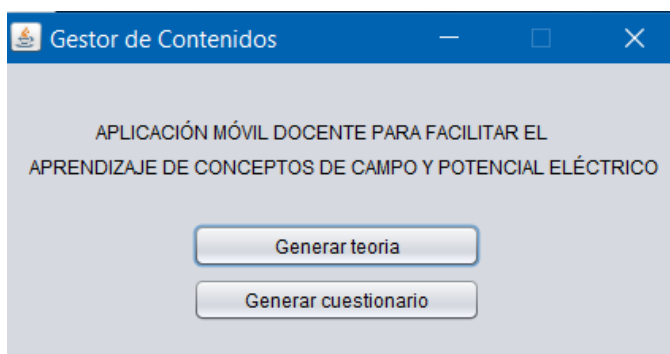


Ilustración 101: Menú principal de la aplicación complementaria

Generar un cuestionario.

Para crear un cuestionario, pulse “generar cuestionario” en el menú principal, a continuación, será dirigido a la siguiente pantalla:

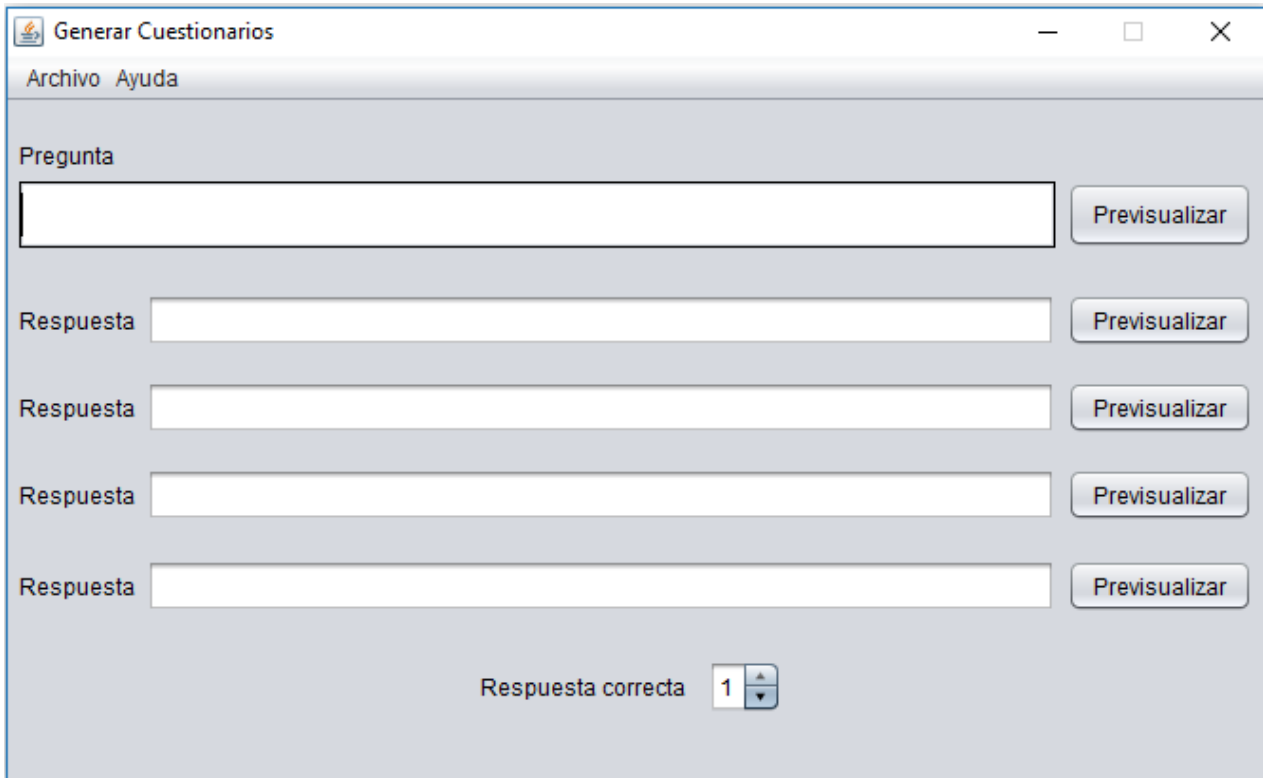


Ilustración 102: Pantalla de creación de cuestionarios

Una vez aquí, introduzca la pregunta, las respuestas e indique la respuesta correcta. Si desea introducir código TEX, enciérreelo entre “\$”. Para previsualizar como quedarán las respuestas o la pregunta, pulse en el correspondiente botón “Previsualizar”, esto le mostrará la apariencia del elemento elegido en su navegador predeterminado.

Para guardar, pulse archivo → guardar como, a continuación, podrá indicar el nombre del archivo y donde guardarlo mediante un explorador del sistema de archivos.

Si lo que desea es editar un archivo existente, puede abrirlo mediante el menú “archivo”, pulsando abrir. El menú “archivo” le ofrece varias opciones que a continuación se detallan.

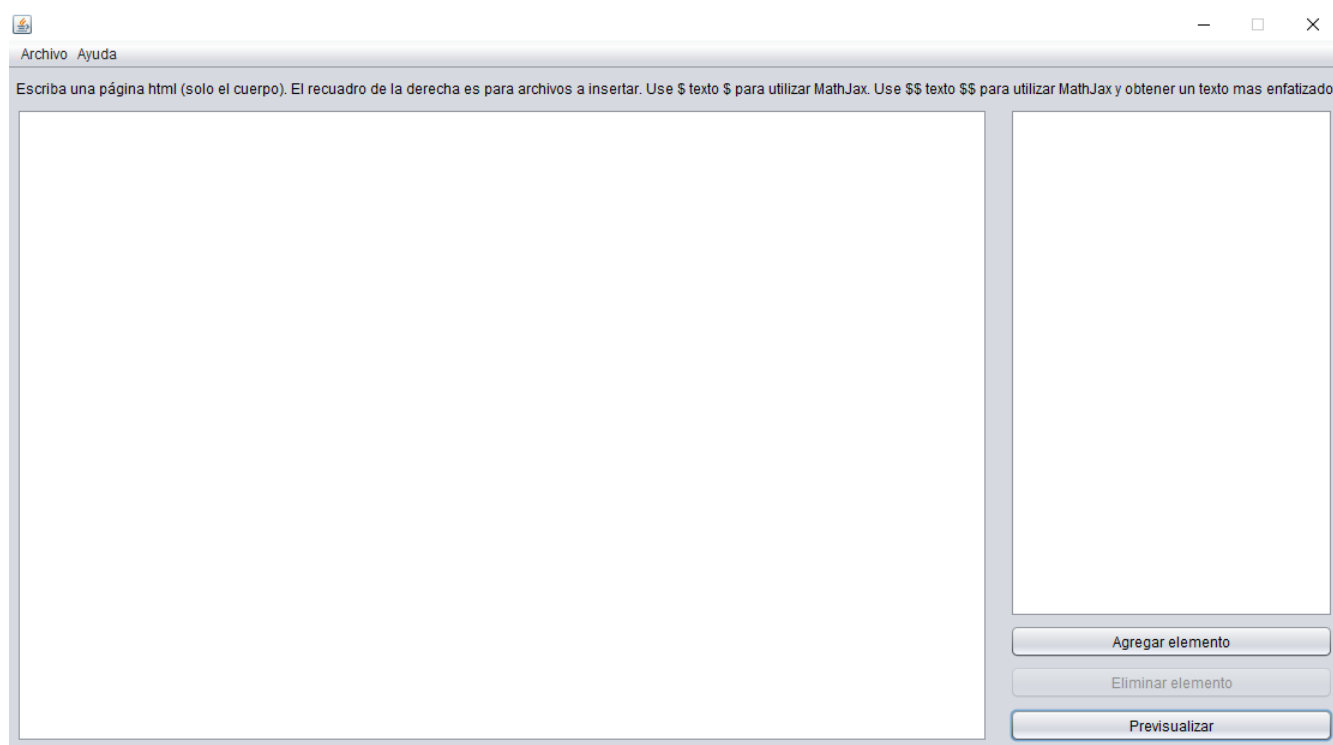
Nuevo	Ctrl+N
Abrir...	Ctrl+A
Guardar	Ctrl+S
Guardar como...	Ctrl+Mayús+S
Salir	Ctrl+Q

Ilustración 103: Opciones del menú “archivo” en cuestionarios

- Nuevo: Vacía todos los elementos de introducción de contenidos. “Guardar” se bloquea, pues aún no se le ha indicado que archivo debe actualizar.
- Abrir: Permite editar un cuestionario, la selección se hace mediante un explorador del sistema de archivos. El botón “guardar” se habilita, pues ya se ha indicado que fichero actualizar.
- Guardar: Actualiza el fichero actual.
- Guardar como: Permite elegir donde guardar el cuestionario generado, tras esto se habilita guardar.
- Salir: Se vuelve al menú principal. Se diferencia del botón cerrar en que este para a la aplicación.

Generar una página de teoría.

Para generar una página de teoría pulse “generar teoría” en el menú principal (ilustración 101), a continuación, se le mostrara la siguiente pantalla:



*Ilustración 104: Pantalla de creación de páginas de teoría*

Introduzca el cuerpo de una página HTML en el campo de introducción de texto, si desea referenciar un fichero, adjunto hágalo **siempre** mediante una ruta relativa.

Para añadir un adjunto, pulse “agregar elemento” y selecciónelo mediante un explorador del sistema de archivos. Si desea eliminar un fichero adjunto pulse “eliminar elemento” y seleccione el nombre del elemento a eliminar. **Nunca** adjunte una página web.

Para escribir código TEX, enciérrelo entre “\$”, si desea enfatizar ese texto, utilice dos símbolos “\$” a cada lado.

Puede previsualizar la página que usted ha escrito en su navegador predeterminado mediante “Previsualizar”, pero tenga en cuenta que al necesitar mostrar los ficheros adjuntos requerirá primero que guarde la página.

Para guardar una página seleccione “guardar como” si no está actualizando un fichero ya creado. El programa considerara el directorio de la página el lugar donde almacene el \*.teoria y ahí guardara los adjuntos. En el momento de guardar, seleccione una carpeta que tenga solo el cometido de guarda esa página o cree una, pues **si el directorio contiene algo que no está declarado en la lista de adjuntos, el programa lo eliminará sin preguntar**. El comportamiento de “guardar” es exactamente el mismo.

En el menú archivo (ilustración 103) se muestran otras opciones que a continuación se detallan.

- Nuevo: Vacía el cuerpo de la página y la lista de adjuntos. “Guardar” se bloquea, pues aún no se le ha indicado que archivo debe actualizar.
- Abrir: Permite editar una página de teoría, la selección se hace mediante un explorador del sistema de archivos. Esta opción carga el texto y la lista de archivos adjuntos. El botón “guardar” se habilita, pues ya se ha indicado que fichero actualizar.
- Guardar: Actualiza la página de teoría actual.
- Guardar como: Permite elegir donde guardar la página de teoría y los ficheros adjuntos, tras esto se habilita guardar.
- Salir: Se vuelve al menú principal. Se diferencia del botón cerrar en que este para a la aplicación.



# Anexo II: Contenido del CD.

El CD entregado para la evaluación de este trabajo se estructura de la siguiente manera:

- Aplicación Complementaria:
  - Código fuente: Contiene el código de la aplicación complementaria
  - Contenidos.jar: Ejecutable de la aplicación complementaria. No precisa de instalación.
- Código fuente: Contiene todo el código y ficheros adjuntos utilizados para construir la aplicación principal.
- Instalable:
  - Campos Eléctricos.apk: Instalable de la aplicación Campos Eléctricos.
- Memoria.pdf: Versión digital de esta memoria.
- Informacion.pdf: Información del proyecto (título, autor y convocatoria).
- Leeme.pdf: Esta información en formato electrónico.