



UNIVERSIDAD DE JAÉN  
Escuela Politécnica Superior de Jaén

## Trabajo Fin de Grado

# **APLICACIÓN MÓVIL PARA LA CORRECCIÓN AUTOMÁTICA DE EXÁMENES TIPO TEST DE RESPUESTA SIMPLE**

**Alumno: Iván Ibáñez Torres**

Tutor: Prof. D. Carlos Javier Ogayar Anguita  
Dpto: Informática

**Junio, 2017**



Universidad de Jaén  
Escuela Politécnica Superior de Jaén  
Departamento de Informática

Don Carlos Javier Ogayar Anguita, tutor del Proyecto Fin de Carrera titulado: APP móvil para la corrección automática de exámenes tipo test de respuesta simple, que presenta Iván Ibáñez Torres, autoriza su presentación para defensa y evaluación en la Escuela Politécnica Superior de Jaén.

Jaén, JUNIO de 2017

El alumno:

Iván Ibáñez Torres

Los tutores:

Carlos Javier Ogayar Anguita

## Índice

Agradecimientos.....	5
1. INTRODUCCIÓN AL PROBLEMA.....	6
1.1. Descripción del problema .....	8
1.2. Motivación.....	10
1.3. Soluciones disponibles .....	10
1.4. Objetivos.....	13
1.5. Estructura del documento.....	14
2. Gestión y Planificación.....	15
2.1. Planificación del proyecto .....	15
2.1.1. Diagrama de Gantt.....	18
2.1.2. Diagrama Pert.....	19
2.2. Análisis de costes .....	21
3. Análisis.....	24
3.1. Requisitos .....	24
3.1.1. Requisitos funcionales .....	24
3.1.2. Requisitos no funcionales .....	24
3.1.3. Requisitos del dispositivo del usuario final .....	25
3.2. OCR (Reconocimiento óptico de caracteres) .....	25
3.2.1. Conclusión .....	28
3.3. Procesamiento imágenes .....	28
3.3.1. Conclusión .....	31
4. Diseño .....	32
4.1. Diagrama de clases .....	32
4.2. Diagrama de casos de uso .....	34
4.3. Diagrama de actividad .....	34
4.4. Patrón Modelo-Vista-Controlador .....	36
4.5. Diagramas de secuencia .....	37
4.5.1. Nuevo examen.....	37
4.5.2. Eliminar examen .....	38
4.5.3. Editar examen.....	39
4.5.4. Corregir examen .....	40
4.5.5. Eliminar calificación .....	42
4.5.6. Exportar calificaciones .....	43

4.6.	StoryBoards .....	44
5.	Implementación .....	<b>52</b>
5.1.	Entorno de desarrollo .....	52
5.2.	Lenguajes de programación .....	53
5.2.1.	Lenguaje programación C.....	53
5.2.2.	C++ .....	53
5.2.3.	Objective-C .....	53
5.2.4.	Swift .....	54
5.3.	Formato plantilla examen .....	55
5.3.1.	Plantillas de prueba .....	55
5.3.2.	Conclusión .....	57
5.3.3.	Tipos de respuesta válidos para la autocorrección.....	58
5.4.	Detección de respuestas .....	59
5.4.1.	Cámara .....	62
5.4.2.	Procesamiento visual y reconocimiento de texto.....	65
5.4.3.	Detección y clasificación de las respuestas.....	66
5.4.4.	Exportación y envío de los resultados .....	70
6.	Pruebas aplicación.....	<b>72</b>
6.1.	Vista principal .....	74
6.2.	Tomar foto examen.....	74
6.3.	Configuración inicial.....	76
6.4.	Procesamiento imagen .....	77
6.5.	Establecer la plantilla de corrección .....	81
6.6.	Calcular calificación .....	82
6.7.	Almacenar calificación .....	83
6.8.	Consultar calificaciones .....	83
6.9.	Exportar calificaciones .....	84
7.	Conclusiones y posibles mejoras .....	<b>86</b>
	Bibliografía .....	<b>90</b>

## Agradecimientos

Con la finalización de este trabajo termina una etapa muy importante de mi vida, pero a su vez se comienza otra mucho más importante en la que seguir aprendiendo.

Me gustaría dedicar este Trabajo Fin de Grado a todo aquel que ha aportado algo a lo largo de todo este camino, en este grupo incluyo familiares, amigos, profesores y conocidos que, aun no sabiéndolo, entre todos han dado la fuerza y el apoyo necesario para conseguir llegar hasta el final.

En especial a mis padres, ya que sin su apoyo y sacrificio no hubiese podido llegar donde hoy me encuentro.

Y por supuesto a mis compañeros y amigos, que, sin ellos, no hubiera sido posible hacer de estudiar algo divertido y ameno.

Muchísimas gracias a todos.

## 1. INTRODUCCIÓN AL PROBLEMA

El examen es la prueba que una persona debe superar para demostrar que ha asimilado los conocimientos del tema que se proponga. En la educación, los profesores se valen de estas pruebas para confirmar que los alumnos están preparados para superar la materia.

Alumnos examinándose:



Ilustración 1 – Alumnos examinándose

Tomando la definición de evolución como calcular el valor de algo, se puede decir, que el término examen se encuentra directamente relacionado con este.

Según ha ido surgiendo la necesidad, a lo largo del tiempo se han definido varios tipos de exámenes, caracterizándose unos por un número de preguntas no muy alto y siendo estas de ámbito general respecto al tema que trate y con respuestas extensas. En este tipo de examen el docente, en cada pregunta, pide al alumno un punto del tema y este debe contestar con todo lo que conoce acerca de ese tema intentando ser lo más explicativo posible.

Otro tipo de examen son los que constan de un mayor número de preguntas, pero con unas respuestas mucho más breves y rápidas de contestar, con lo que se intenta que los alumnos demuestren conocimientos más específicos acerca de una cuestión en concreto. Suelen ser preguntas directas en las que se exige brevedad en la respuesta, sin exceder más de dos o tres líneas.

Por otro lado, tenemos otro tipo de examen que se conoce como examen multirespuesta en los que a cada pregunta se le adjuntan varias posibles respuestas y los alumnos deben elegir entre ellas la respuesta correcta o la considerada más correcta ya que se puede dar el caso en el que varias o todas las respuestas son correctas, pero, una de ellas prevalece sobre todas las demás, siendo esta la correcta y las demás incorrectas.

### Examen multirespuesta:

Diplomatura Estadística e I.T.I.G CURSO 3º. EXAMEN BBDD 2 18/01/13 Usuario: GR	
APELLIDOS _____ NOMBRE: _____ DNI: _____	
<b>Antes de comenzar el examen se ruega leer las siguientes notas:</b>	
1.- Al finalizar el examen, el alumno entregará este gráfico y sin doblar (incluidas las hojas de encabezado). Si el alumno decide no entregar el examen, deberá igualmente entregar este examen con los datos personales cumplimentados. 2.- Completar la caja de respuesta de acuerdo a las indicaciones que se detallan a continuación. 3.- La lista de respuestas se incluye para facilitar el proceso de corrección. No escribir nada en ella. 4.- Colocar la respuesta correcta en la casilla correspondiente. Una respuesta incorrecta no tiene valor. 5.- Marca las respuestas que consideres correctas. Cada pregunta puede tener entre 0 y 4 respuestas correctas.	
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35	
1. (7 puntos) Marca las respuestas que consideres correctas. Cada pregunta puede tener entre 0 y 4 respuestas correctas. Para una pregunta se dí por correcta deben marcarse todas las respuestas correctas. Las preguntas mal respondidas no restan nota. Cada pregunta correcta suma 0.20 puntos.	
i. La independencia de datos implica: a. Si modificamos el modelo físico de datos entonces debemos modificar el modelo lógico de datos b. Si modificamos el modelo lógico de datos no es necesario modificar el modelo físico de datos c. Cambiar modificando en una BBDD conllevará la necesaria modificación de las asociaciones que la accedan d. La independencia de datos se establece sólo entre la capa física y lógica de la BBDD	
ii. Respecto a las variables en PL/SQL: a. Pueden ser de tipo base y de tipo ancla b. Los de tipo ancla pueden, entre otras cosas, ser del mismo tipo que una columna de una tabla c. Las de tipo ancla pueden, entre otras cosas, almacenar varias filas de una tabla d. Las de tipo ancla pueden, entre otras cosas, almacenar una única fila de un cursor	
iii. Respecto a los procedimientos almacenados: a. Es un procedimiento que se ejecuta dentro de un DBMS b. Es un procedimiento que se ejecuta en el DBMS pero se ejecuta en el cliente que lo invoca c. Es un procedimiento que se ejecuta en el DBMS pero se almacena en el cliente que lo invoca d. Permite compartir código entre distintas sesiones	
iv. iv. Respecto a la sentencia SQL UPDATE X SET X.a=X.a+1. Un disparador declarado "CREATE TRIGGER TRG_UPD_X_A AFTER INSERT"	
a. Se ejecutará una única vez después de realizarse ninguna actualización b. Se ejecutará una vez antes de actualizarse cada una de las filas de la tabla X c. No tendrá acceso a las variables :new ni :old d. Tendrá acceso de lectura a la tabla X	
v. Una tabla radioactiva: a. No puede ser leída por un disparador de tupla b. No puede ser modificada por un disparador de tupla c. Puede ser leída por un disparador de sentencia d. Es una tabla a la que le ha picado una araña radioactiva	
vi. El diccionario de datos: a. Viene predefinido por el DBMS, que a su vez es el encargado de mantenerlo actualizado b. Un usuario solo puede realizar las operaciones de lectura sobre este c. Dependiendo entre otras cosas, el modelo lógico de datos de la BBDD d. Describe, entre otras cosas, el modelo físico de datos de la BBDD	
vii. Tareas del DBA son: a. Evaluar el hardware disponible para el DBMS b. Instalar el DBMS c. Planificar la Base de Datos d. Realizar copias de seguridad de la Base de Datos completa	
viii. En Oracle, los grupos de ficheros <i>redolog</i> están multiplexados si y solo si: a. Hay varios grupos de ficheros <i>redolog</i> online b. Hay más de un fichero <i>redolog</i> grupo c. El modo ARCHIVELOG está activado d. No hay ficheros <i>redolog</i> offline	
ix. En Oracle, los espacios de tablas o <i>tablespace</i> : a. No son parte de ninguna instancia b. Se organizan en segmentos, extensiones y bloques de datos c. Son manipulados por el proceso database writer (dbwr) d. Se almacenan en uno o más archivos	

**Ilustración 2 - Examen multirespuesta**

Por último, tenemos el examen de respuesta simple, el cual se centra este Trabajo Fin de Grado, este tipo de exámenes suele caracterizarse por tener un número elevado de preguntas y en el que solo se dan las opciones de verdadero o falso. En ocasiones la persona examinada debe decidir entre una de las dos opciones que se le dan a elegir o dejar la pregunta en blanco si no está seguro de cuál es la respuesta correcta, ya que una respuesta mal contestada puede considerarse como puntuación negativa en el momento de realizar el cálculo global del resultado del examen.

## Examen respuesta simple Verdadero/Falso:

Criptografía y Seguridad en Computadores 5 de septiembre de 2005. Examen																																	
Apellidos y Nombre: _____	D.N.I.: _____																																
<p>A. Indique si cada uno de los siguientes enunciados es verdadero o falso, colocando una V o una F en la casilla correspondiente. Tenga en cuenta que las respuestas erróneas restan la mitad del valor de una respuesta correcta (5 puntos).</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; padding: 2px;">Pregunta</th> <th style="width: 15px;"></th> </tr> </thead> <tbody> <tr><td style="padding: 2px;">1. Una curva elíptica está compuesta por una serie de puntos que cumplen cierta condición, más un punto <math>\mathcal{O}</math> situado en el infinito.</td><td style="width: 15px; text-align: center; vertical-align: middle;"><input type="checkbox"/></td></tr> <tr><td style="padding: 2px;">2. La criptografía de curva elíptica permite definir versiones sobre este formalismo de los algoritmos asimétricos más comunes.</td><td style="width: 15px; text-align: center; vertical-align: middle;"><input type="checkbox"/></td></tr> <tr><td style="padding: 2px;">3. Existen algoritmos eficientes para saber si un número es o no primo, pero no para factorizarlo.</td><td style="width: 15px; text-align: center; vertical-align: middle;"><input type="checkbox"/></td></tr> <tr><td style="padding: 2px;">4. Los registros de desplazamiento son el mecanismo más seguro para construir cifrados de flujo, pero son muy lentos.</td><td style="width: 15px; text-align: center; vertical-align: middle;"><input type="checkbox"/></td></tr> <tr><td style="padding: 2px;">5. Los criptosistemas simétricos emplean la misma clave para cifrar y descifrar.</td><td style="width: 15px; text-align: center; vertical-align: middle;"><input type="checkbox"/></td></tr> <tr><td style="padding: 2px;">6. El algoritmo DES no se considera seguro porque se le han encontrado fallos de diseño.</td><td style="width: 15px; text-align: center; vertical-align: middle;"><input type="checkbox"/></td></tr> <tr><td style="padding: 2px;">7. La confusión está asociada al cambio de posición (transposición) de los símbolos.</td><td style="width: 15px; text-align: center; vertical-align: middle;"><input type="checkbox"/></td></tr> <tr><td style="padding: 2px;">8. La criptografía asimétrica puede usarse tanto para proteger información, como para firmarla digitalmente.</td><td style="width: 15px; text-align: center; vertical-align: middle;"><input type="checkbox"/></td></tr> <tr><td style="padding: 2px;">9. Una S-Caja es la unidad básica de transposición en algoritmos de cifrado por bloques.</td><td style="width: 15px; text-align: center; vertical-align: middle;"><input type="checkbox"/></td></tr> <tr><td style="padding: 2px;">10. Los criptosistemas asimétricos emplean la misma clave para cifrar y descifrar.</td><td style="width: 15px; text-align: center; vertical-align: middle;"><input type="checkbox"/></td></tr> <tr><td style="padding: 2px;">11. Una red de Feistel tiene la propiedad de que puede usarse tanto para cifrar como para descifrar, empleando las inversas módulo <math>n</math> de las sub-claves en lugar de éstas.</td><td style="width: 15px; text-align: center; vertical-align: middle;"><input type="checkbox"/></td></tr> <tr><td style="padding: 2px;">12. Un MD5 es una función de autenticación que necesita una clave para poder ser calculada.</td><td style="width: 15px; text-align: center; vertical-align: middle;"><input type="checkbox"/></td></tr> <tr><td style="padding: 2px;">13. El modo de operación ECB no permite detectar sustituciones e inserciones de bloques en el mensaje.</td><td style="width: 15px; text-align: center; vertical-align: middle;"><input type="checkbox"/></td></tr> <tr><td style="padding: 2px;">14. El problema de los logaritmos discretos puede resolverse de forma eficiente empleando computación paralela.</td><td style="width: 15px; text-align: center; vertical-align: middle;"><input type="checkbox"/></td></tr> <tr><td style="padding: 2px;">15. Cuando se genera un par de llaves PGP, conviene obtener un certificado de revocación y guardarlo para su posterior uso.</td><td style="width: 15px; text-align: center; vertical-align: middle;"><input type="checkbox"/></td></tr> </tbody> </table>		Pregunta		1. Una curva elíptica está compuesta por una serie de puntos que cumplen cierta condición, más un punto $\mathcal{O}$ situado en el infinito.	<input type="checkbox"/>	2. La criptografía de curva elíptica permite definir versiones sobre este formalismo de los algoritmos asimétricos más comunes.	<input type="checkbox"/>	3. Existen algoritmos eficientes para saber si un número es o no primo, pero no para factorizarlo.	<input type="checkbox"/>	4. Los registros de desplazamiento son el mecanismo más seguro para construir cifrados de flujo, pero son muy lentos.	<input type="checkbox"/>	5. Los criptosistemas simétricos emplean la misma clave para cifrar y descifrar.	<input type="checkbox"/>	6. El algoritmo DES no se considera seguro porque se le han encontrado fallos de diseño.	<input type="checkbox"/>	7. La confusión está asociada al cambio de posición (transposición) de los símbolos.	<input type="checkbox"/>	8. La criptografía asimétrica puede usarse tanto para proteger información, como para firmarla digitalmente.	<input type="checkbox"/>	9. Una S-Caja es la unidad básica de transposición en algoritmos de cifrado por bloques.	<input type="checkbox"/>	10. Los criptosistemas asimétricos emplean la misma clave para cifrar y descifrar.	<input type="checkbox"/>	11. Una red de Feistel tiene la propiedad de que puede usarse tanto para cifrar como para descifrar, empleando las inversas módulo $n$ de las sub-claves en lugar de éstas.	<input type="checkbox"/>	12. Un MD5 es una función de autenticación que necesita una clave para poder ser calculada.	<input type="checkbox"/>	13. El modo de operación ECB no permite detectar sustituciones e inserciones de bloques en el mensaje.	<input type="checkbox"/>	14. El problema de los logaritmos discretos puede resolverse de forma eficiente empleando computación paralela.	<input type="checkbox"/>	15. Cuando se genera un par de llaves PGP, conviene obtener un certificado de revocación y guardarlo para su posterior uso.	<input type="checkbox"/>
Pregunta																																	
1. Una curva elíptica está compuesta por una serie de puntos que cumplen cierta condición, más un punto $\mathcal{O}$ situado en el infinito.	<input type="checkbox"/>																																
2. La criptografía de curva elíptica permite definir versiones sobre este formalismo de los algoritmos asimétricos más comunes.	<input type="checkbox"/>																																
3. Existen algoritmos eficientes para saber si un número es o no primo, pero no para factorizarlo.	<input type="checkbox"/>																																
4. Los registros de desplazamiento son el mecanismo más seguro para construir cifrados de flujo, pero son muy lentos.	<input type="checkbox"/>																																
5. Los criptosistemas simétricos emplean la misma clave para cifrar y descifrar.	<input type="checkbox"/>																																
6. El algoritmo DES no se considera seguro porque se le han encontrado fallos de diseño.	<input type="checkbox"/>																																
7. La confusión está asociada al cambio de posición (transposición) de los símbolos.	<input type="checkbox"/>																																
8. La criptografía asimétrica puede usarse tanto para proteger información, como para firmarla digitalmente.	<input type="checkbox"/>																																
9. Una S-Caja es la unidad básica de transposición en algoritmos de cifrado por bloques.	<input type="checkbox"/>																																
10. Los criptosistemas asimétricos emplean la misma clave para cifrar y descifrar.	<input type="checkbox"/>																																
11. Una red de Feistel tiene la propiedad de que puede usarse tanto para cifrar como para descifrar, empleando las inversas módulo $n$ de las sub-claves en lugar de éstas.	<input type="checkbox"/>																																
12. Un MD5 es una función de autenticación que necesita una clave para poder ser calculada.	<input type="checkbox"/>																																
13. El modo de operación ECB no permite detectar sustituciones e inserciones de bloques en el mensaje.	<input type="checkbox"/>																																
14. El problema de los logaritmos discretos puede resolverse de forma eficiente empleando computación paralela.	<input type="checkbox"/>																																
15. Cuando se genera un par de llaves PGP, conviene obtener un certificado de revocación y guardarlo para su posterior uso.	<input type="checkbox"/>																																
<i>Sigue en la página siguiente...</i> 1																																	

Ilustración 3 - Examen respuesta simple

### 1.1. Descripción del problema

Este trabajo fin de grado, de ahora en adelante TFG, consiste en crear una aplicación móvil para la corrección automática de exámenes tipo test de respuesta simple mediante la cámara del dispositivo.

En este tipo de examen, normalmente se le indica al alumno donde debe escribir la respuesta, la localización de estas casi siempre se encuentran en uno de los márgenes laterales del examen, haciendo coincidir en línea horizontal el lugar de la respuesta con el texto de dicha pregunta. Hay ocasiones en las que las respuestas son anotadas en un cuadrante, ya sea en la misma hoja o en otra a parte en la que se indica el número de la pregunta y la respuesta.

Hay distintos tipos de respuestas que se pueden dar en este tipo de examen como puede ser redondear o tachar la opción correcta, escribir las iniciales de V y F o incluso llenar algún tipo forma geométrica como puede ser un cuadrado o círculo.

Hoja de respuesta de un examen tipo test de respuesta simple:

V	F	V	F	V	F	V	F	V	F
1		31		61		91		121	
2		32		62		92		122	
3		33		63		93		123	
4		34		64		94		124	
5		35		65		95		125	
6		36		66		96		126	
7		37		67		97		127	
8		38		68		98		128	
9		39		69		99		129	
10		40		70		100		130	
11		41		71		101		131	
12		42		72		102		132	
13		43		73		103		133	
14		44		74		104		134	
15		45		75		105		135	
16		46		76		106		136	
17		47		77		107		137	
18		48		78		108		138	
19		49		79		109		139	
20		50		80		110		140	
21		51		81		111		141	
22		52		82		112		142	
23		53		83		113		143	
24		54		84		114		144	
25		55		85		115		145	
26		56		86		116		146	
27		57		87		117		147	
28		58		88		118		148	
29		59		89		119		149	
30		60		90		120		150	

Nombre \_\_\_\_\_ / Cargo \_\_\_\_\_ / Fecha \_\_\_\_\_

PB     PP  
 IP     In  
 C     IA  
 Media     D.S.

Ilustración 4 - Hoja respuestas simples

El principal problema a la hora de plantear esta aplicación se da en el momento de diseñar una plantilla de examen por defecto pensando en los alumnos que posteriormente tendrán que rellenarlo, pensando que disponen de un tiempo limitado y debe ser algo relativamente fácil de entender y de contestar.

Pensando en los profesores que son los que deben editar el examen y completarlo con las preguntas correspondientes al tema, se busca un formato que pueda ser modificado con cualquier editor de textos en cualquier momento sin necesidad de marcas especiales.

A la hora de corregir el examen, la aplicación debe simplificar el máximo posible, el proceso de comparar las respuestas del alumno con la plantilla de soluciones. Por lo que el formato del examen debe ser analizado por la aplicación mediante procesamiento visual de la imagen y que este proceso no ralentice la aplicación y suponga un elevado tiempo de respuesta de esta.

En definitiva, el principal problema es la creación de un examen tipo, atractivo visualmente, similar a los que se realizan normalmente en papel, sin que suponga un gran esfuerzo por parte del profesor a la hora de crear uno nuevo, ni un problema para el alumno a la hora de responder las cuestiones que se le plantean y por supuesto analizable por parte del dispositivo en un tiempo razonable.

## 1.2. Motivación

Este TFG está dentro de la rama Tecnologías de la Información, ya que se utilizan diferentes herramientas y servicios para el proceso de corrección de exámenes y posterior envío de resultados, además está dentro de la mención de Sistemas Gráficos ya que se utiliza procesamiento de imágenes para la captura de los exámenes y mejoras en la calidad para el posterior análisis para la extracción de las respuestas del alumno.

El realizar una aplicación para el móvil me llamaba mucho la atención ya que a día de hoy las tecnologías móviles están en alza y tenía mucho interés por aprender y hacer algo en alguna de las plataformas móvil como es Swift, totalmente desconocido para mí y un nuevo reto.

Otro de los motivos por el cual he escogido este TFG, es porque creo que puede ser una aplicación que puliendo algunos detalles puede llegar ayudar y ser bastante útil tanto a los profesores ahorrando tiempo a la hora de corregir los exámenes como a los alumnos para disminuir la espera de publicación de las calificaciones.

## 1.3. Soluciones disponibles

Para entender un poco mejor los objetivos de esta aplicación, se hará un breve recopilatorio con el software disponible actualmente con el mismo fin que este TFG.

- Gexcat:

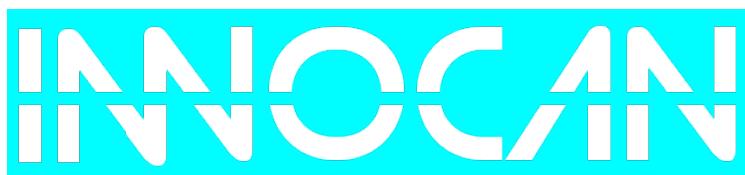


Ilustración 5 - Innocan

Software propiedad de INNOCAN SISTEMAS S.L., empresa TIC dedicada al desarrollo de aplicaciones basadas en el tratamiento de datos e imágenes.

Oferta una aplicación de escritorio con una fácil de generación de exámenes, la cual recibe estos mediante unas imágenes obtenidas con un escáner.

## Plantilla examen de ejemplo Gexcat:

ULL | Universidad de La Laguna

# ASIGNATURA DE PRUEBA

## EXAMEN DE PRUEBA (17/02/2014)

Mod.	::	Identificador	::	::	::	::	::	::	::	::	::	::
Escriba los números uniendo los puntos como en el ejemplo												
	□	.:	2	3	4	5	6	7	8	9		

1º Apellido

2º Apellido

Nombre

Escriba sólo letras en mayúscula y sin tilde

Correcto:

Incorrecto:

x	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
a	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
b	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
c	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
d	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
e	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
x	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
a	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
b	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
c	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
d	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
e	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
x	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75
a	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
b	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
c	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
d	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
e	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
x	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
a	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
b	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
c	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
d	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
e	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.

Rellene una de las dos mitades para seleccionar la opción

Para anular, rellene las dos mitades

Procure rellenar sin salirse del recuadro

Utilice rotulador o bolígrafo negro

Ejemplo:

GE/CAT

## Ilustración 6 - Ejemplo examen Gexcat

- EXAMIO:



Ilustración 7 - ExamIO

ExamIO es un programa basado en la tecnología OMR (Reconocimiento Óptico de Marcas) que sirve para diseñar, corregir y analizar resultados de encuestas y pruebas tipo test realizados en papel, de manera automática.

Al igual que la anterior, es una aplicación de escritorio, requiere un escáner como entrada de los exámenes y recomienda uno con alimentador de hojas para facilitar la tarea de escaneo al usuario.

## Plantilla examen ejemplo ExamIO:

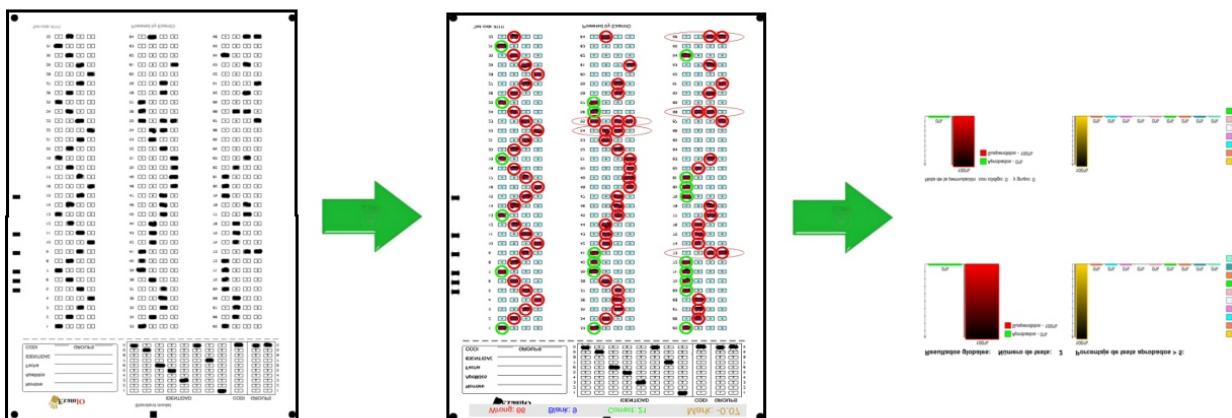


Ilustración 8 - Plantilla ejemplo ExamIO

- COTEST:

Software desarrollado en Java, como aplicación multiplataforma para escritorio, por Darío Gutiérrez, Lucas Díaz Sanzo, del Departamento de Informática de la Universidad de Oviedo.

Permite generar las hojas para realizar el examen, que se imprimen en papel normal. El alumno marca su identificador, el modelo y las respuestas en la hoja. Después únicamente se necesita un escáner (mejor automático), para digitalizar las hojas y llevarlas al ordenador.

#### Plantilla examen de ejemplo Gexcat:

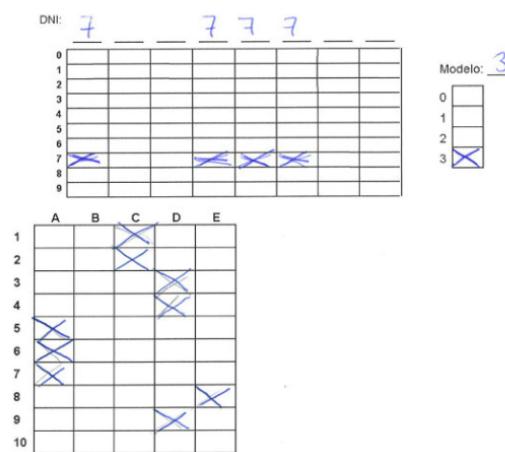


Ilustración 9 - Hoja respuestas Gexcat

Como se ha podido observar en las opciones listadas, las soluciones software para abordar el problema descrito son pocas, de reducida movilidad y con necesidad de un desembolso si no se dispone de un escáner. Otra característica común en las tres aplicaciones es una hoja de respuestas independiente de las preguntas. Esto puede dificultar la realización de la prueba y provocar fallos en la respuesta.

#### **1.4. Objetivos**

El objetivo principal de este trabajo es crear una aplicación móvil que facilite a los docentes la tarea de corregir los exámenes tipo test de respuesta simple de manera cómoda, rápida y eficaz a través de una aplicación simple e intuitiva.

Para llevar a cabo este desarrollo será necesario analizar los requisitos e implementar una solución que los satisfaga.

Para desarrollar este proyecto se necesitará:

- Investigar acerca de las aplicaciones existentes con el mismo fin o parecido para analizar tanto las posibilidades que ofrecen como los puntos donde falla e intentar mejorarlos.
- Realización de la guía oficial de Apple para desarrolladores principiantes en Swift
- Analizar distintos formatos de exámenes tipo test de respuesta simple utilizados por otras herramientas similares.
- Diseñar una plantilla en la que las preguntas y respuestas se encuentren en la misma hoja para facilitar la realización de este al alumno.
- Investigar acerca de las librerías de procesamiento visual disponibles para Swift.
- Investigar acerca de las librerías OCR disponibles para Swift.
- Implementación de la aplicación.

## 1.5. Estructura del documento

La documentación de este trabajo se divide en varias partes:

Una primera parte presenta la gestión y planificación del proyecto, en este apartado se tratará de documentar cómo se organizará el desarrollo de este proyecto, así como los costes de este.

En la parte del análisis hablaremos de los diferentes requisitos de nuestro proyecto, además estudiaremos los distintos tipos de librerías que podemos utilizar y las herramientas que se utilizarán para el desarrollo de este.

En la parte del diseño se desarrollarán los diferentes diagramas como de clases, casos de uso, actividad o de secuencia.

En la parte de la implementación se tratará de explicar las diferentes herramientas utilizadas, y una explicación más concisa del desarrollo del proyecto. En este apartado se explicarán los códigos más relevantes.

Por último, se presentarán unas conclusiones y mejoras acerca de este proyecto.

## 2. Gestión y Planificación

En este capítulo se tratará de planificar, localizar en el tiempo y documentar las diferentes tareas en las que se basa este proyecto, además se calculará el coste de este.

### 2.1. Planificación del proyecto

Este proyecto se comenzó el día 20 de febrero de 2017 y se ha terminado el día 25 de junio de 2017, por lo que se ha estado desarrollando durante 125 días, durante los cuales se han trabajado unas 2'5 horas diarias, alcanzando así y superando las 300 horas obligatorias de este Trabajo Fin de Grado.

El ciclo de vida del desarrollo Software (SDLC en sus siglas inglesas), es una secuencia estructurada y bien definida de las etapas en Ingeniería de software para desarrollar el producto software deseado.

Para este proyecto se ha utilizado una modelo de ciclo de vida incremental, de este modo se han dividido los problemas en sub-problemas más sencillos y en los que cada uno forma parte de un incremento. Cada incremento es indispensable para poder continuar con el siguiente, una vez terminado un incremento se comprueba que todo funciona correctamente y se pasa al siguiente.

Algunos de los objetivos de esta metodología son:

- Poder enseñar un producto operativo pero incompleto al cliente en períodos cortos de tiempo.
- Los requisitos más importantes de la aplicación son incluidos en los primeros incrementos del proyecto.
- Cada incremento tiene un tiempo determinado y breve.
- No hay marcha atrás, cualquier modificación necesaria a un incremento anterior, será incluido en un incremento nuevo y posterior.

Las tareas quedan divididas de la siguiente forma:

Sub-problemas	Duración (Horas)
Interfaz gráfica básica	15
Captura foto examen	20.6
Mejora calidad imagen	37.4
OCR y procesamiento texto	31.4
Gestión y almacenamiento de exámenes	12.6
Gestión y almacenamientos de calificaciones	12.6
Mejora visual general de la app	25.4
Exportación de calificaciones para posterior envío	2.6
<b>Total</b>	<b>157.6</b>

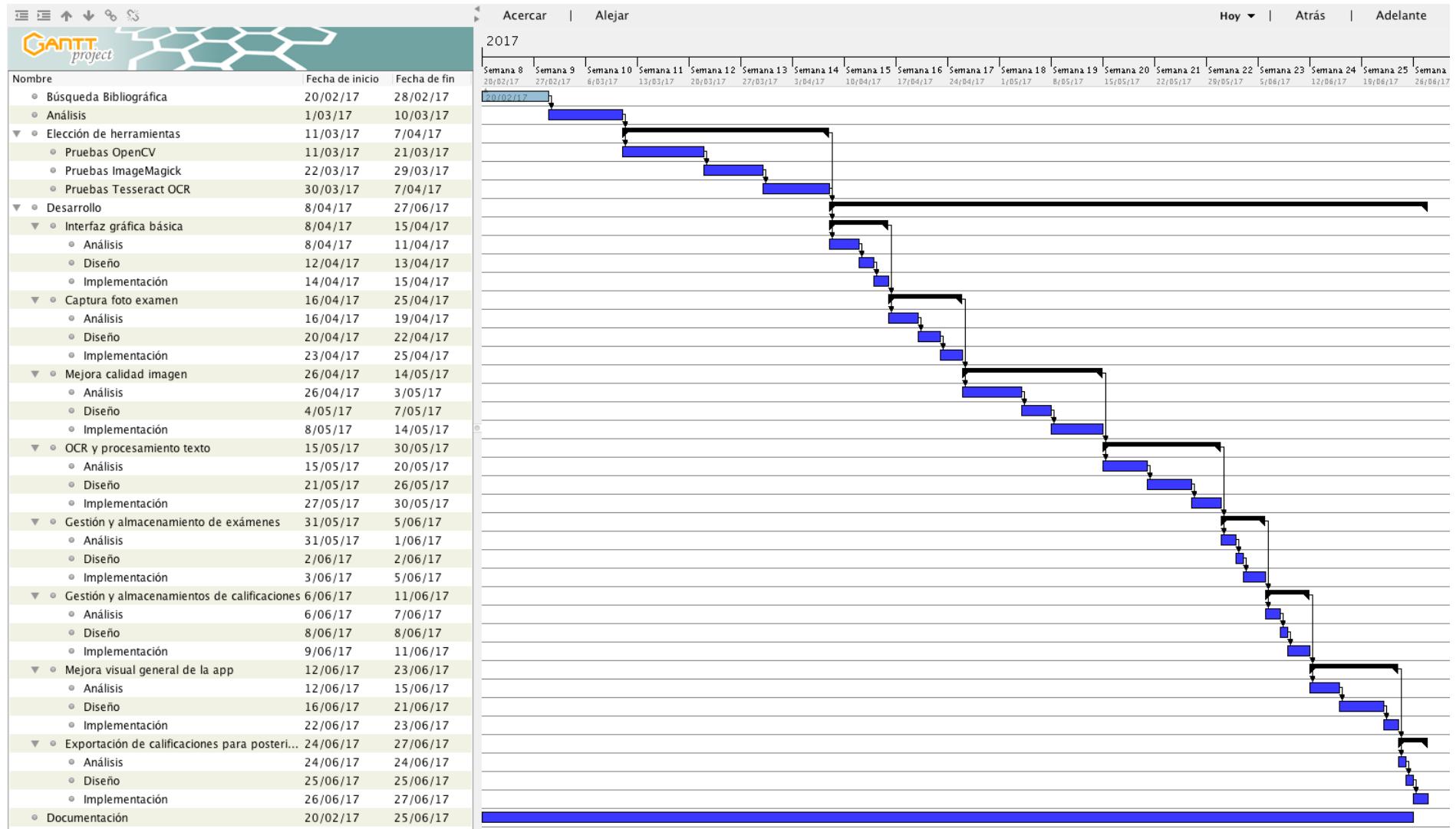
Tabla 1 – División de tareas

En la siguiente tabla se detallan cada una de las actividades y el tiempo dedicado a cada una de ellas:

Tarea	Duración (Días)	Horas diarias	Duración (Horas)	Fecha Inicio	Fecha Fin
<b>Búsqueda Bibliográfica</b>	<b>8</b>	<b>2</b>	<b>16</b>	<b>20/02/17</b>	<b>28/02/17</b>
<b>Análisis</b>	<b>10</b>	<b>2</b>	<b>20</b>	<b>01/03/17</b>	<b>10/03/17</b>
<b>Elección de herramientas</b>	<b>28</b>	<b>2</b>	<b>56</b>	<b>11/03/17</b>	<b>07/04/17</b>
Pruebas OpenCV	11	2	22	11/03/17	21/03/17
Pruebas ImageMagick	8	2	16	22/03/17	29/03/17
Pruebas Tesseract OCR	9	2	18	30/03/17	07/04/17
<b>Desarrollo</b>	<b>78</b>	<b>2</b>	<b>157.6</b>	<b>08/04/17</b>	<b>25/06/17</b>
Interfaz gráfica básica	7.6	2	15	08/04/17	15/04/17
Análisis	4	2	8	08/04/17	11/04/17
Diseño	2	2	4	12/04/17	13/04/17
Implementación	1.7	2	3.4	14/04/17	15/04/17
Captura foto examen	10.3	2	20.6	16/04/17	25/04/17
Análisis	4.3	2	8.6	16/04/17	19/04/17
Diseño	2.7	2	5.4	20/04/17	22/04/17
Implementación	3.3	2	6.6	23/04/17	25/04/17
Mejora calidad imagen	18.7	2	37.4	26/04/17	14/05/17
Análisis	8.3	2	16.6	26/04/17	03/05/17
Diseño	4.3	2	8.6	04/05/17	07/05/17
Implementación	6	2	12	08/05/17	14/05/17
OCR y procesamiento texto	15.7	2	31.4	15/05/17	30/05/17
Análisis	6.3	2	12.6	15/05/17	20/05/17
Diseño	5.3	2	10.6	21/05/17	26/05/17
Implementación	4	2	8	27/05/17	30/05/17
Gestión y almacenamiento de exámenes	6.3	2	12.6	31/05/17	05/06/17
Análisis	2	2	4	31/06/17	01/06/17
Diseño	1.3	2	2.6	02/06/17	02/06/17
Implementación	3	2	6	03/06/17	05/06/17
Gestión y almacenamientos de calificaciones	6.3	2	12.6	06/06/17	11/06/17
Análisis	2	2	4	06/06/17	07/06/17
Diseño	1.3	2	2.6	08/06/17	08/06/17
Implementación	3	2	6	09/06/17	11/06/17
Mejora visual general de la app	11.7	2	25.4	12/06/17	23/06/17
Análisis	4	2	8	12/06/17	15/06/17
Diseño	5.7	2	11.4	16/06/17	21/06/17
Implementación	2	2	4	22/06/17	23/06/17
Exportación de calificaciones para posterior envío	1.3	2	2.6	24/06/17	25/06/17
Análisis	0.7	2	1.4	24/06/17	24/06/17
Diseño	0.3	2	0.6	24/06/17	24/06/17
Implementación	0.3	2	0.6	24/06/17	25/06/17
<b>Documentación</b>	<b>125</b>	<b>0'5</b>	<b>62'5</b>	<b>20/02/17</b>	<b>25/06/17</b>
<b>Total</b>	<b>125</b>	<b>2'5</b>	<b>312'5</b>	<b>20/02/17</b>	<b>25/06/17</b>

Tabla 2 - Planificación horas

### 2.1.1. Diagrama de Gantt



## Ilustración 10 - Diagrama Gantt

### 2.1.2. Diagrama Pert

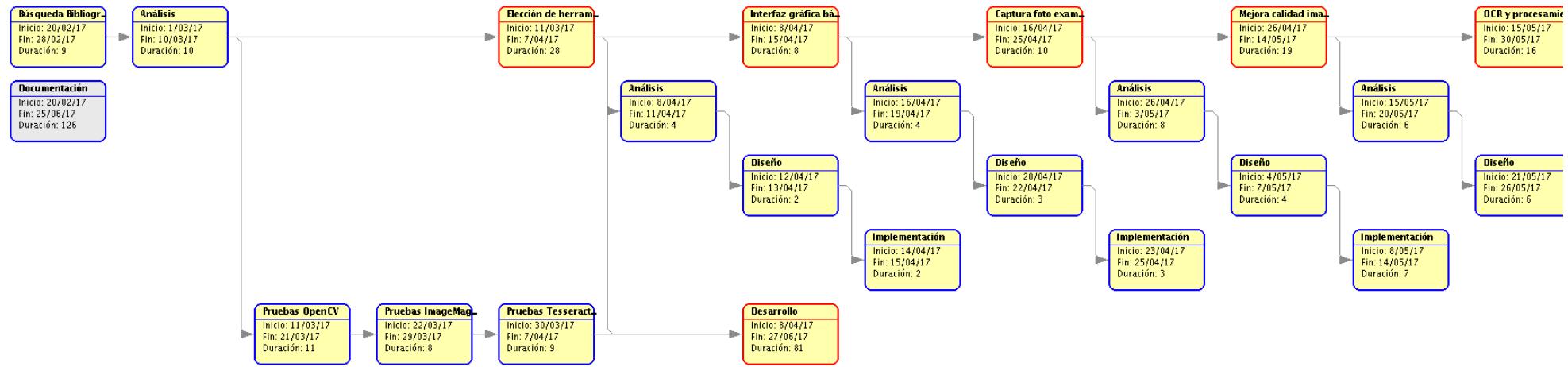
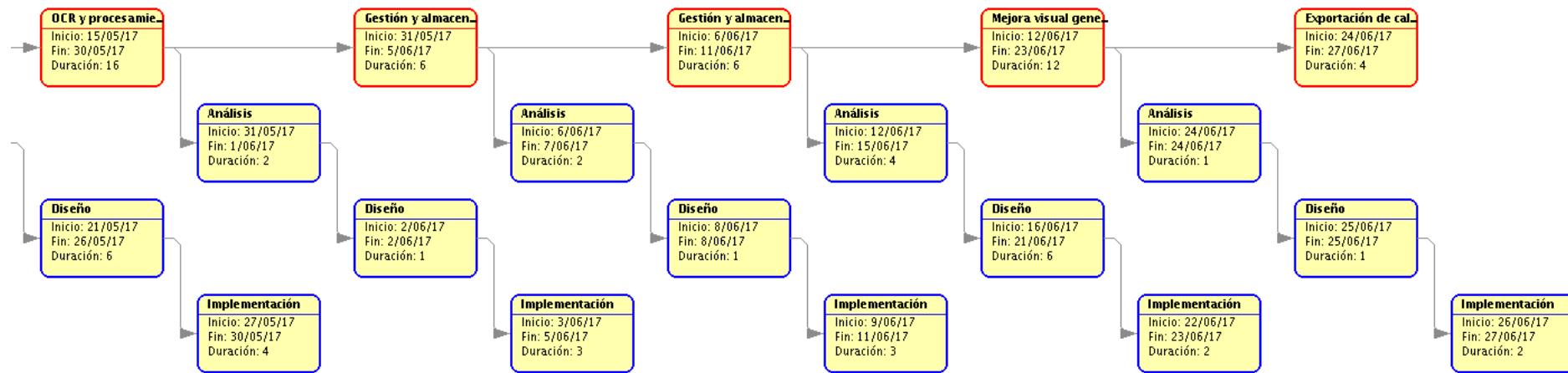


Ilustración 11 - Diagrama de Pert



## 2.2. Análisis de costes

En este apartado se hará una estimación de los costes que conlleva la realización de este proyecto.

Para ello, hay que tener en cuenta todos los elementos que intervienen en el proyecto como son software y hardware utilizado, conexión a internet para la consulta de información, licencia de desarrollador de Apple para la publicación de la App en la App Store.

Comenzando por el software, para la implementación de la aplicación se ha utilizado el entorno de desarrollo integrado (IDE) Xcode corriendo bajo el sistema operativo macOS Sierra. Para los retoques realizados en imágenes se utilizará el programa de edición GIMP, la edición de esta documentación y los distintos modelos de exámenes utilizados para las pruebas han sido llevados a cabo con la aplicación Microsoft Word incluida en el paquete Microsoft Office 2016, para los diagramas de clase, casos de uso y demás, se utilizará la herramienta Visual Paradigm. Los diagramas utilizados para la representación de la planificación del tiempo han sido editados con la aplicación GanttProject.

Para hacer un análisis de costes lo más real posible, es necesario calcular la parte proporcional del coste del paquete Office, estimando un tiempo de vida útil de 2 años. El coste del paquete se corresponde a 0,00850457€/hora.

Coste software del proyecto:

Software	Precio
Xcode	0
macOS Sierra	0
GIMP	0
Microsoft Office 2016	3,18
Visual Paradigm	19,00
GanttProject	0
<b>Total</b>	<b>22,18</b>

Tabla 3 - Coste software proyecto

Por la parte hardware, se ha utilizado un MacBook Pro Retina 15", mediados 2014 con un procesador Intel Core i7, 16GB de memoria RAM DDR3 y como dispositivo de almacenamiento un disco SSD de 256GB, por un precio de 1.888,37€. Se supone que el equipo será utilizado para más trabajos y no ha sido adquirido expresamente para el proyecto, por lo que hay que calcular la parte proporcional. La vida útil del dicho equipo suele ser 5 años por lo que es un total de 43800 horas. El coste de uso por hora del equipo corresponde a 0,0443€.

Como dispositivo real para las pruebas de la app se ha utilizado un iPhone SE con un procesador A9 dual-core de 64bits, memoria RAM de 2GB y 16GB de almacenamiento. El dispositivo tiene un precio de 299€. Al igual que el equipo anteriormente especificado, hay que calcular la parte proporcional del coste asociado a este proyecto. La vida útil de este dispositivo suele estar entre los 3-4 años, lo que da unas 35040 horas. Este dispositivo ha estado implicado en un 70% del desarrollo de este proyecto, por lo que en el cálculo también hay tenerlo en cuenta. El coste de utilización de este dispositivo es de 0,0085€/h.

En el proceso visual de las imágenes y el reconocimiento del texto se realizaron una cantidad importante de impresiones de distintos formatos de exámenes con una impresora All In One HP OfficeJet 3833, con un coste 52,99€. Realizando el mismo calculo con una vida útil de 2-3 años el coste por hora equivale a 0,0020€. La implicación de la impresora en el proyecto corresponde a un 25% del total de este.

Coste hardware del proyecto:

Hardware	Precio Unidad(€)	Unidades	Total
Equipo desarrollo	0,04	312,5	13,47
Dispositivo pruebas	0,009	306,25	2,61
Impresora	0,002	109,38	0,22
<b>Total</b>			12,26€

Tabla 4 - Coste hardware proyecto

Hay que incluir el coste de personal que tiene el proyecto. Para ello se hacen los cálculos basándose en “XVI Convenio colectivo estatal de empresas de consultoría y estudios de mercados de la opinión pública” publicado en el BOE el 4 de abril de 2009. El sueldo anual de un titulado es de 20954,36 anuales. Al realizar el cálculo del coste por hora del personal se obtiene 9,922€. El total de horas de este proyecto es de 437,5 horas por lo que el coste de personal asciende a 4340,875€. Teniendo en cuenta el gasto de seguridad social que suele rondar el 30% del sueldo, este sube a 5643,1375€.

Toda la información y ayuda necesita acerca del tema trabajado ha sido extraída íntegramente en internet, por lo que se ha necesitado una conexión a internet contratada con ONO de 30Mb simétricos de fibra óptica con un precio mensual de 39,45€. Realizando el cálculo para los 4 meses que dura el proyecto el coste de la conexión a internet asciende a 157,8€.

Por último, para la distribución de la App se necesita una licencia de desarrollador para poder publicarla en la App Store. Esta tiene un coste anual de 99€.

Coste del desarrollo del proyecto:

Actividad	Precio (€)
Software	22,19
Hardware	12,26
Personal	5643,14
Conexión Internet	157,8
Licencia desarrollador	99
<b>Total</b>	<b>5934,39</b>

Tabla 5 - Coste del proyecto

### 3. Análisis

Esta fase es la más importante en el desarrollo de una aplicación, es imprescindible un buen análisis para crear una buena base para el proyecto. En ella se analiza el problema detalladamente, los requisitos que debe cumplir la aplicación para un correcto funcionamiento y la solución escogida.

#### 3.1. Requisitos

Los requisitos son las funcionalidades o restricciones que la aplicación debe cumplir. Este apartado es dedicado al análisis de estos requisitos, ya sean requisitos funcionales o requisitos no funcionales.

##### 3.1.1. Requisitos funcionales

En los requisitos funcionales se incluyen las posibilidades que al App debe brindar al usuario final a la hora de corregir exámenes.

Los requisitos funcionales de esta App son:

- Corregir exámenes tipo test de respuesta simple mediante la ayuda de la cámara del dispositivo móvil.
- Posibilidad de corregir exámenes con más de una pagina
- Gestión de exámenes (añadir, editar, eliminar).
- Gestión de calificaciones asociadas alumnos.
- Exportación y posterior envío de calificaciones.

##### 3.1.2. Requisitos no funcionales

Los requisitos no funcionales describen aspectos del sistema que están relacionados con el grado de cumplimiento de los requisitos funcionales.

Los requisitos no funcionales de esta aplicación son:

- Un formato de examen simple, para una fácil edición a la hora de crearlo.
- Preguntas y repuestas juntas, sin necesidad de hojas independientes.

- Utilización de caracteres en el reconocimiento de respuestas que no influya en la redacción de las preguntas.
- Un formato de examen atractivo visualmente para el alumno que realiza la prueba.
- Debe permitir corregir exámenes en distintas sesiones, almacenado el proceso y reanudándolo donde se dejó.
- Interfaz simple, que permita la corrección de exámenes fácilmente.

### 3.1.3. Requisitos del dispositivo del usuario final

Como requisito hardware se recomienda utilizar la App en un dispositivo iPhone 5s o cualquiera de los modelos posteriores a este. Por debajo de este modelo no se puede obtener un gran rendimiento de procesamiento.

La aplicación esta compilada para la versiona 10 de iOS por lo que el dispositivo utilizado debe estar actualizado como mínimo a dicha versión o posterior.

## 3.2. OCR (Reconocimiento óptico de caracteres)

Para el desarrollo de este trabajo, un requisito fundamental es la obtención de las respuestas elegidas por los alumnos para poder calcular la calificación de este en la prueba. Para ello se utilizará el proceso de reconocimiento óptico de caracteres. A continuación, veremos las diferentes alternativas disponibles para iOS, investigadas para este TFG:

- **BlinkOCR SDK:**



Ilustración 12 - microblink

BlinkOCR SDK es un módulo de OCR desarrollado por MicroBlink. Su tecnología OCR está optimizada específicamente para la arquitectura de dispositivos móviles. Ofrecen una API Key de prueba al registrarse en su web con 100 ejecuciones.

- **Anyline OCR Document:**



Ilustración 13 - ANYLINE

Desarrollado por Anyline, SDK multiplataforma que encapsula todo el proceso de reconocimiento mediante la configuración de parámetros como son el tamaño de la letra, diseño de fuente, color, etc. dependiendo del documento a escanear. Ofertan un mes gratuito por registrarte en la web. Después la suscripción es anual con packs de escaneos.

- **SwiftOCR:**



Ilustración 14 - SwiftOCR

Librería OCR escrita por Nicolas Camenisch en Swift bajo licencia Apache 2.0 disponible para iOS y OS X. Solo esta optimizada para el reconocimiento de códigos alfanuméricos cortos de una línea y no para documentos extensos.

- **ABBY Fine Reader Engine:**



Ilustración 15 - ABBYY

Propiedad de ABBY, basado en su plataforma de reconocimiento, aseguran buen rendimiento sin un gran coste de recursos. Ofrece reconocimiento inteligente de caracteres (ICR) y código de barras. Licencia gratuita para prueba de 30 días.

- **Scansbot SDK:**



# Scansbot

Ilustración 16 - Scansbot

SDK de fácil integración desarrollado por PSPPDFKit. Disponible para Android e iOS en Java y Objective-C respectivamente. El núcleo esta implementado de forma independiente en C++. Dispone de detección de documentos, *cropping* y optimización de imagen. Puede obtenerse una licencia de prueba poniéndose en contacto con ellos, después el pago se realiza mediante suscripción.

- **Orbit SDK:**



# Orbit SDK

Ilustración 17 - Orbit SDK

Desarrollado por OCR Labs, el motor de este SDK combina una alta tasa de captura de imágenes con una conversión óptica de alto rendimiento. Administra inteligentemente los caracteres, la iluminación del entorno e incluso los movimientos de la mano para ofrecer mejores resultados con alta precisión, aseguran un promedio del 98% y menor tiempo. Para obtener una licencia de prueba es necesario contactar mediante correo electrónico.

- **Tesseract OCR iOS:**

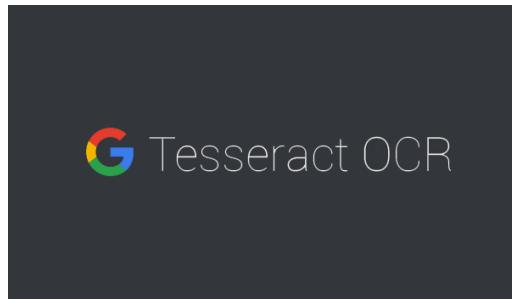


Ilustración 18 - Tesseract OCR

Tesseract es un motor OCR Open Source, desarrollado originalmente como software propietario por Hewlett Packard en 1995, liberado posteriormente y actualmente desarrollado por Google y distribuido bajo licencia Apache 2.0. Considerado de los mejores motores libres disponibles en la actualidad. Con reconocimiento de más de 60 idiomas.

### 3.2.1. Conclusión

Se han realizado pruebas tanto con las distintas versiones de prueba como con la librería Open Source, comparando tiempo de reconocimiento, recursos consumidos y resultados obtenidos no se ha apreciado una gran diferencia entre el software de pago y el libre, por lo que para este proyecto la librería que será utilizada será Tesseract que no supone ningún coste.

## 3.3. Procesamiento imágenes

El rendimiento del motor OCR a la hora de detectar los datos en el examen está directamente relacionado con la calidad de la imagen que se le pase. Por lo que la elección de una buena librería para el procesamiento visual de imágenes, es otro de los puntos importantes de este TFG. A continuación, se listan las diferentes alternativas disponibles para iOS:

- **GPUImage2:**



Ilustración 19 - GPUImage2

Esta librería para iOS está distribuida bajo licencia BSD, permite aplicar filtros personalizados a imágenes, captura en tiempo real de cámara y videos. Para agilizar el procesamiento de las imágenes se ayuda de la GPU para realizar los cálculos.

- **Toucan:**



Ilustración 20 - Toucan

Toucan es una librería para Swift que provee de una API limpia y sencilla para el procesamiento de imágenes. Funciones de redimensionamiento de imágenes, recorte y estilizado de imágenes.

- **ImageMagick:**



Ilustración 21 - ImageMagick

Es un paquete software Open Source para la creación, composición y conversión de imágenes de mapa de bits. Compatible con más de 200 formatos. ImageMagick se puede usar para redimensionar, reflejar, recortar. También incluye funcionalidades para realizar ajustes de color, escribir texto o dibujar líneas, polígonos, etc.

- **OpenCV:**



Ilustración 22 - OpenCV

OpenCV (Open Source Computer Vision Library) distribuida bajo licencia BSD. Con interfaz en C++, C, Python y Java y soportado en Windows, Linux, Mac OS, iOS and Android. OpenCV fue diseñado pensando en la eficiencia computacional y centrándose en las aplicaciones de tiempo real. Implementado para aprovechar el procesamiento multi-core.

Esta librería es conocida y utilizada en todo el mundo, con una comunidad de 47000 usuarios.

### 3.3.1. Conclusión

Analizando las opciones disponibles y realizando algunas pruebas se ha optado por utilizar la librería OpenCV, principalmente por los buenos resultados obtenidos en ocasiones anteriores en otros trabajos realizados y por la gran comunidad de usuarios de la que dispone en la que poder realizar cualquier consulta en caso de duda.

## 4. Diseño

En este capítulo se analizará más a fondo cada una de las características que tendrá la aplicación, lo que nos permitirá implementarlo de forma efectiva. Sigue un patrón Modelo-Vista-Controlador. Se estudiarán los diferentes diagramas y se mostrará una primera vista de lo que será esta aplicación.

### 4.1. Diagrama de clases

En este apartado se muestra el diagrama de clases que compone este proyecto.

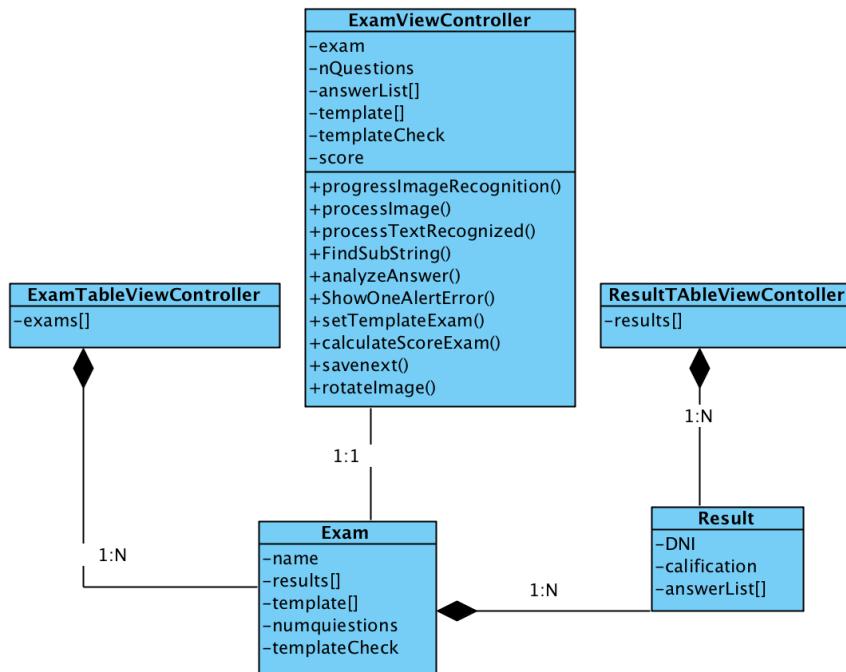


Ilustración 23 - Diagrama de clases

La clase Exam es la base de esta aplicación, es la entidad real que se va a tratar y la que se intenta abstraer. Un examen está compuesto por un nombre para la prueba, un listado con los resultados, una plantilla de corrección y el número de preguntas que consta el examen.

Un resultado está asociado a un alumno identificado por su D.N.I., una calificación obtenida para la prueba y un listado con todas las opciones escogidas por el alumno.

Como se puede observar en el diagrama, la clase con más funciones de la aplicación es ExamViewController, es la encargada de gestionar el procesamiento del examen. En ella están implementados los principales métodos que se utilizan para dicho fin. A continuación, se describe la funcionalidad de cada uno de ellos:

- progressImageRecognition: Encargado de informar acerca del estado del procesamiento y reconocimiento del examen.
- processImage: En este método se realiza una parte del procesamiento visual de la imagen, detección del texto del examen.
- processTextRecognized: El objetivo de esta función es localizar en el texto reconocido el número de la pregunta y con la ayuda de FinSubString la posición de esta en el texto para extraerla.
- analizeAnswer: Una vez extraída la respuesta, este método la evalúa y la codifica en la lista d respuestas según sea verdadera, falsa o sin respuesta.
- ShowOneAlertError: Comprueba de una lista de preguntas que no han sido evaluadas correctamente y muestra una ventana de alerta que permite introducir la respuesta elegida por el alumno.
- setTemplateExam: Cuando todas las preguntas del examen tienen un valor correcto, se da la opción de establecer ese examen como plantilla. Este es el método encargado de hacerlo.
- calculateScoreExam: Calcula la nota obtenida por el alumno en la prueba, una vez que todas las preguntas tienen un valor correcto y ya hay una plantilla de corrección establecida.
- Savenext: Permite almacenar la nota del alumno en corrección y reiniciar la interfaz para continuar con el siguiente.
- rotatetImage: Si la imagen es tomada de lado con esta función se tiene la opción de rotarla.

Las clases `ResultTableViewController` y `ExamTableViewController` son las encargadas de controlar las vistas donde se muestran los listados de exámenes y resultados almacenados dentro de estos.

#### 4.2. Diagrama de casos de uso

A continuación, se muestra el diagrama de casos de uso:

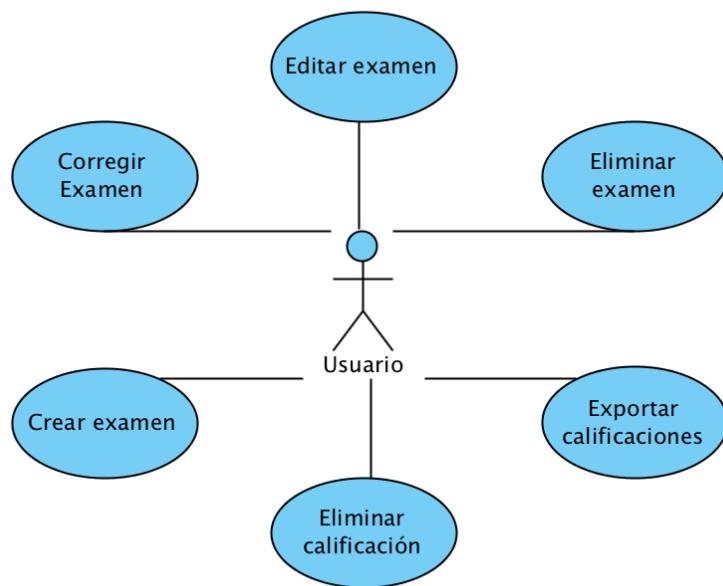


Ilustración 24 - Diagrama de casos de uso

La aplicación permite al usuario:

- Crear un nuevo examen
- Editar examen
- Corregir el examen de un alumno
- Eliminar un examen
- Eliminar la calificación de un alumno
- Exportar los resultados para enviarlos

#### 4.3. Diagrama de actividad

En la siguiente imagen se muestra el diagrama de actividad que sigue la aplicación. A continuación, se detalla cada uno de los pasos.

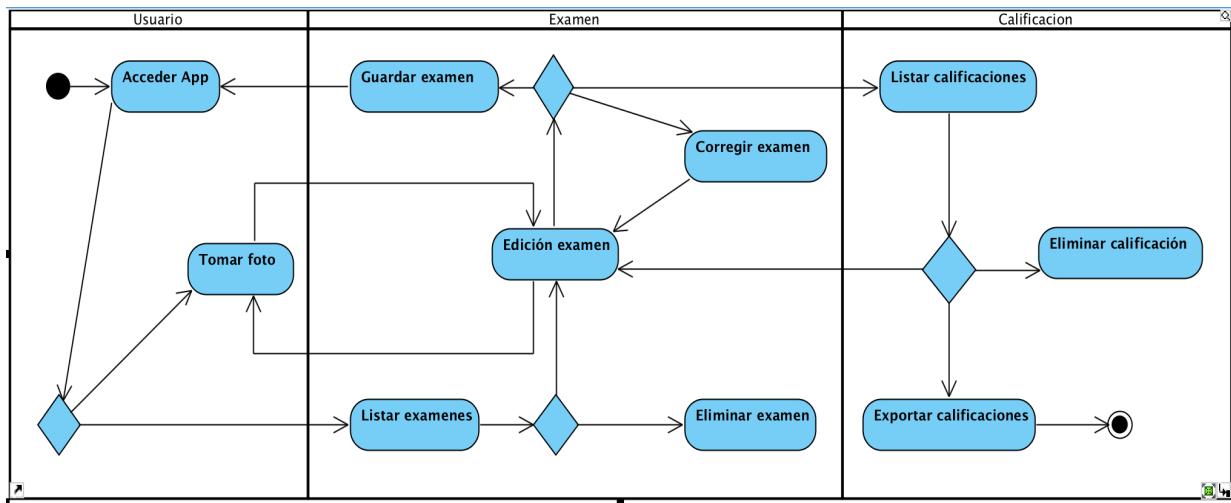


Ilustración 25 - Diagrama de actividad

Al acceder a la aplicación, el usuario se encuentra con dos opciones, puede fotografiar un examen directamente o acceder al listado de exámenes guardados. Desde el listado tiene la opción de crear un nuevo examen, donde tendrá que realizar la configuración inicial con el número de preguntas que tiene y empezar a tomar imágenes para su procesamiento. Otra opción de listado es seleccionar alguno de los exámenes guardados y acceder a la vista de edición donde podrá continuar con la corrección donde se quedó la última vez. Desde el listado también se puede eliminar exámenes deslizando el dedo sobre su fila de derecha a izquierda.

Al seleccionar la opción de fotografiar directamente en la vista principal, se activará la cámara del dispositivo y una vez tomada la foto, el usuario es redirigido a la vista de creación de exámenes para la configuración inicial.

En la vista de edición, una vez fotografiado el examen, el usuario debe procesar la imagen y corregir los errores detectados. Si hay una plantilla establecida para el examen, el usuario podrá obtener la nota del alumno, almacenarla y continuar corrigiendo otro examen. Antes de calcular la nota, el examen debe tener una plantilla de corrección establecida, esta puede ser modificada en cualquier momento.

En esta misma ventana, se encuentra la opción de listar las calificaciones siempre y cuando la lista no esté vacía, en este listado es posible eliminar cualquier calificación si hubiera ocurrido algún error en el proceso de corrección o, si se ha

terminado de corregir todos los exámenes, exportar el listado a un fichero csv. Este fichero se podrá enviar por correo electrónico, WhatsApp, DropBox, etc.

#### 4.4. Patrón Modelo-Vista-Controlador

El patrón modelo-vista-controlador (MVC) separa la representación de las vistas de la aplicación de la interacción con estas. Es el patrón más extendido en el desarrollo software. Consta de tres capas, como su nombre indica, la capa del modelo, la capa de la vista y la capa del controlador.

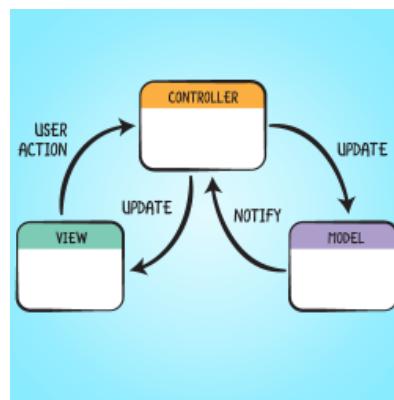


Ilustración 26 - Patrón MVC

La principal ventaja de este patrón es la flexibilidad que ofrece a la hora de poder hacer cualquier modificación a cada capa independientemente. El objetivo es separar la interfaz que se le muestra al usuario de los datos de la aplicación y de la lógica que controla estos datos.

- El modelo es el encargado de mantener la gestión y estructura de los datos de la aplicación. Es el que permite el procesamiento como la creación, eliminación o modificación. Normalmente suele corresponder con un Sistema Gestor de Base de Datos.
- El controlador es el encargado de cargar la información en la vista que el usuario ve. También actúa de intermediario entre la vista y el modelo avisando al modelo de alteración de los datos desde la interfaz.
- La vista es la encargada de dar formato a la información que el controlador le carga proveniente del modelo en forma de interfaz.

## 4.5. Diagramas de secuencia

Estos diagramas representan la interacción entre los objetos en una línea temporal. Al utilizar un patrón modelo-vista-controlador, el diagrama muestra la interacción entre estas capas.

### 4.5.1. Nuevo examen

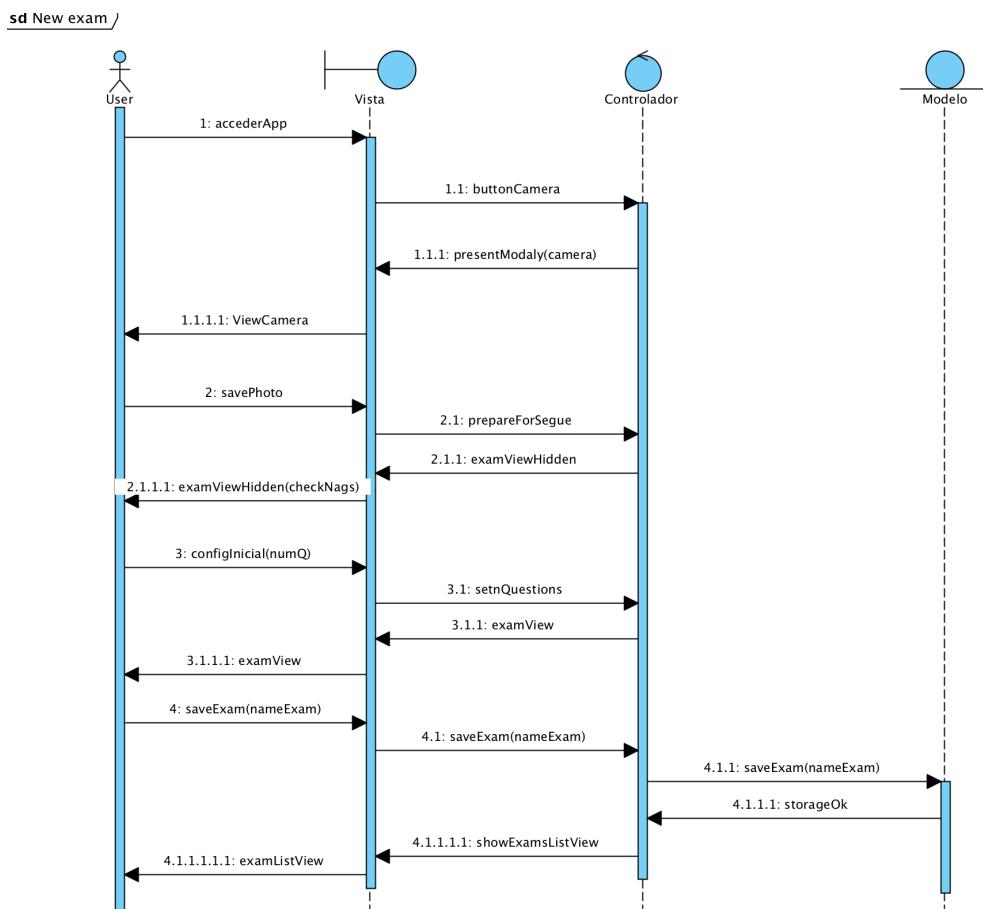


Ilustración 27 - Diagrama secuencia nuevo examen

Nada más abrir la App, el usuario tiene la opción de activar directamente la cámara del dispositivo o listar el histórico de exámenes guardados. En esta secuencia el usuario selecciona la opción de abrir la cámara y capturar la foto del examen.

Una vez realizada la foto del examen, la App es redirigida a la vista de configuración del examen donde el usuario tendrá como primer paso obligatorio introducir el número de preguntas de este. Ya configurado el número de preguntas,

el usuario tiene que introducir como mínimo el nombre del examen para que el botón de almacenar examen se active.

#### 4.5.2. Eliminar examen

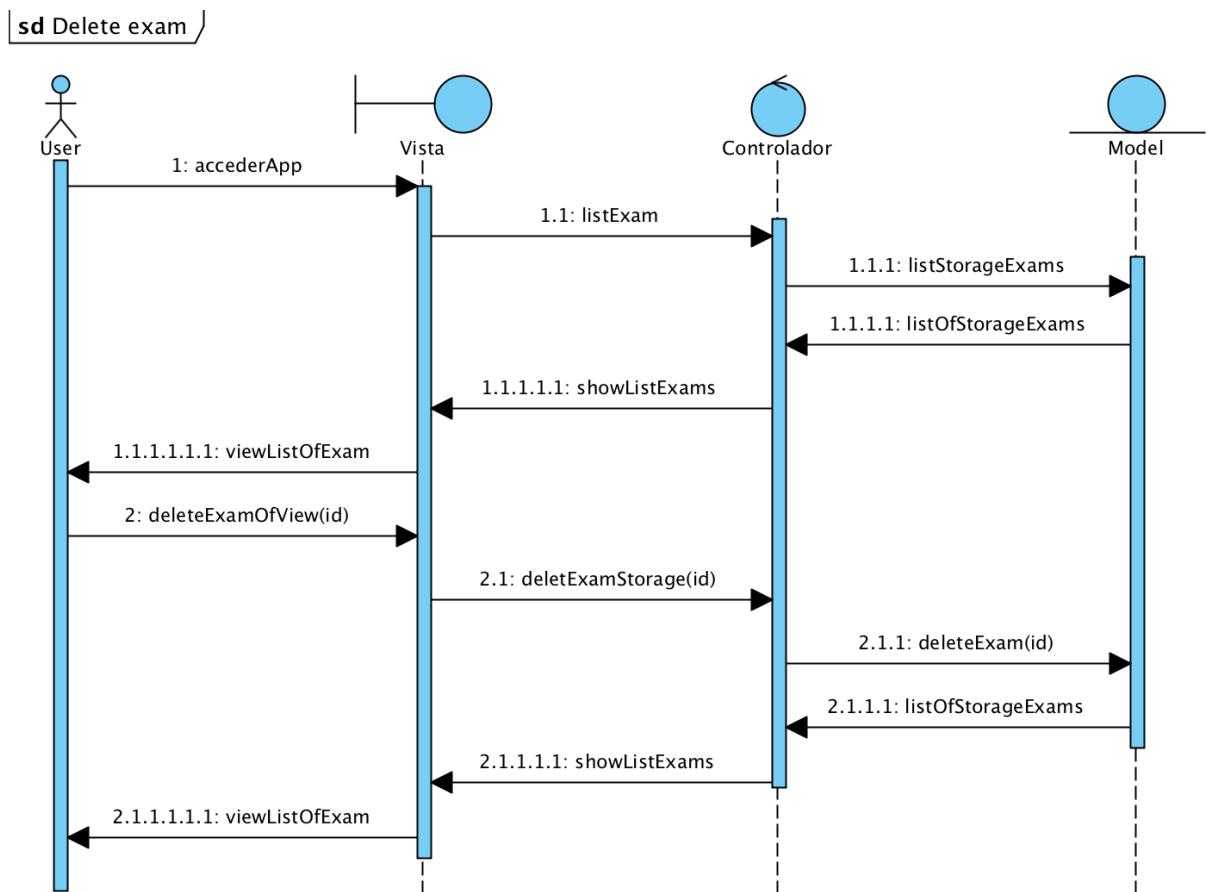


Ilustración 28 - Diagrama secuencia eliminar examen

En esta secuencia, al abrir la App, el usuario selecciona la opción de listar el histórico de exámenes almacenados. Desde el listado, el usuario puede deslizar el dedo de derecha a izquierda sobre la celda del examen que se quiere borrar, a la derecha de dicha celda aparece un botón rojo que hay que pulsar para eliminar el examen.

La vista comunica al controlador que se ha eliminado un elemento de la tabla, este se lo comunica al modelo y este le devuelve una lista actualizada sin el examen eliminado en la vista. El controlador le devuelve este la lista a la vista y ya el usuario la recibe formateada gráficamente.

#### 4.5.3. Editar examen

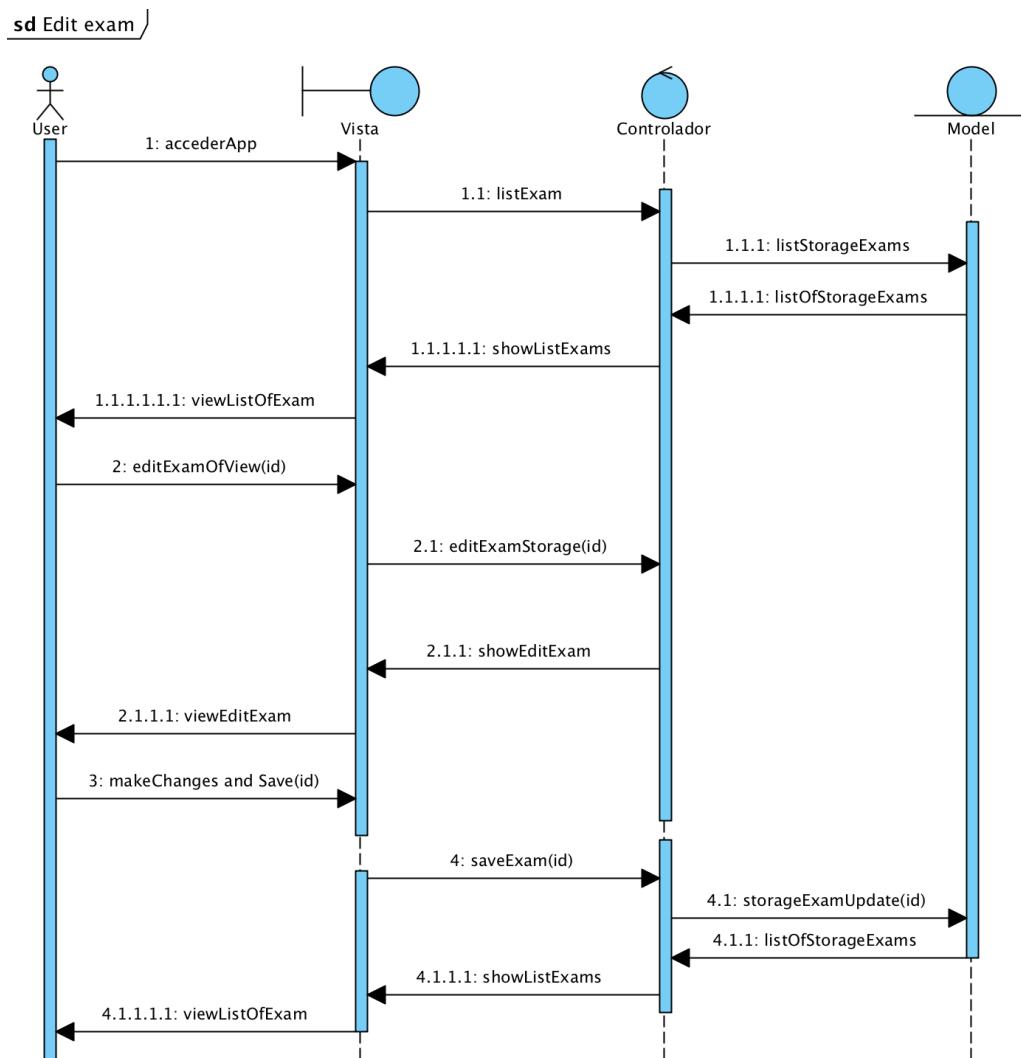


Ilustración 29 - Diagrama secuencia editar examen

En esta secuencia, al abrir la App, el usuario selecciona la opción de listar el histórico de exámenes almacenados. Desde el listado, el usuario puede seleccionar la celda del examen que quiere editar, la vista será redirigida a la de edición de exámenes y en ella el usuario podrá realizar los cambios que desee.

La vista comunica al controlador que se ha editado un examen de forma correcta, este se lo comunicará al modelo y este le devolverá una lista actualizada con los cambios realizados por el usuario. El controlador, le devolverá este la lista a la vista y ya el usuario la recibirá formateada gráficamente.

#### 4.5.4. Corregir examen

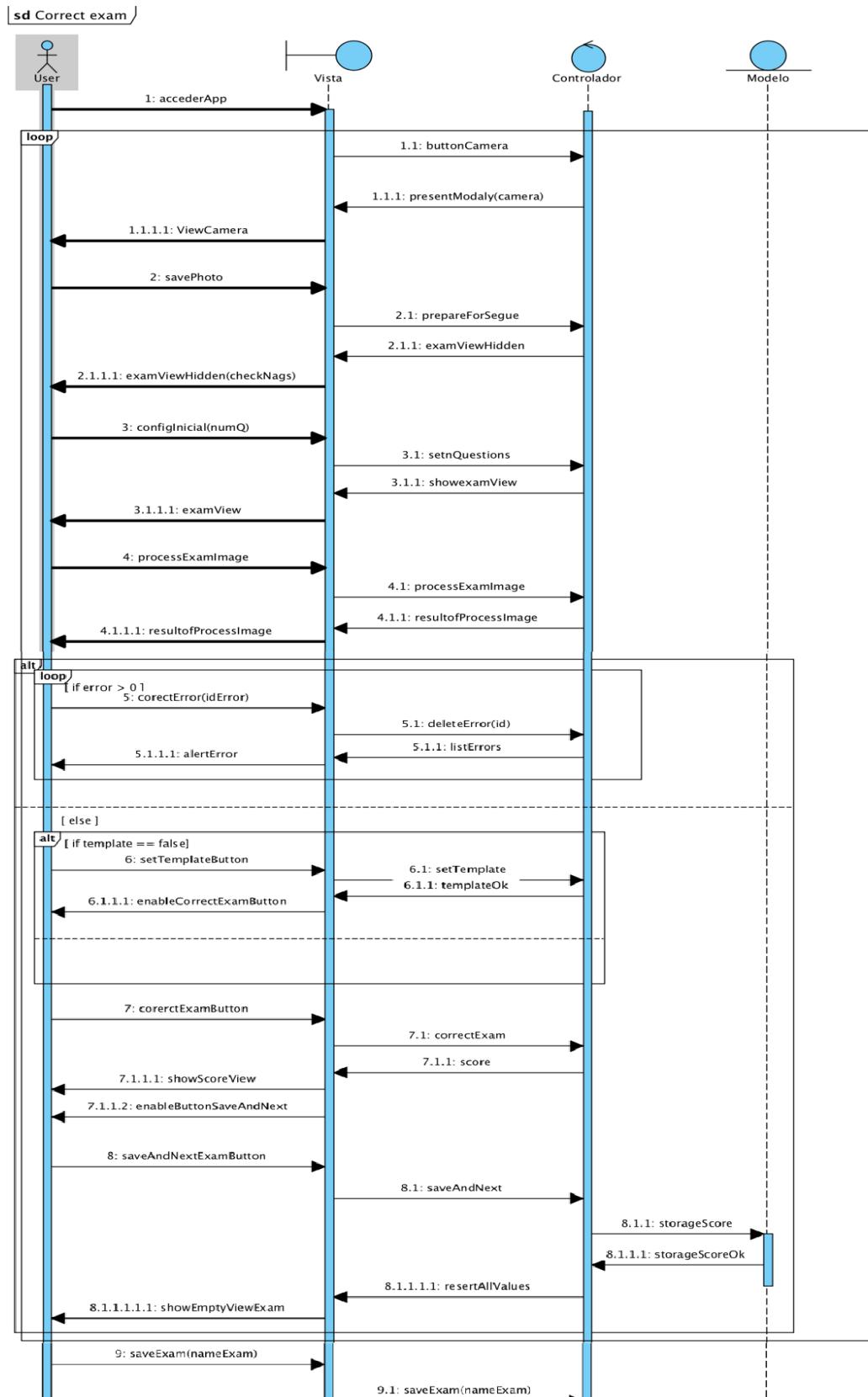


Ilustración 30 - Diagrama secuencia corregir examen

El usuario abre la aplicación y selecciona la activación de la cámara para realizar la foto al examen directamente, una vez tomada la foto deseada, el usuario es redirigido a la configuración inicial del examen.

Procesara la imagen tomada y si hubiera algún error, el usuario puede corregirlos manualmente o capturar otra fotografía para intentar tomar de nuevo las repuestas erróneas. El profesor no podrá continuar el proceso de corrección sin corregir los errores.

Una vez solventados todos los errores, el controlador comprueba si hay una plantilla de corrección establecida para el examen. Si no hay ninguna plantilla establecida, el usuario solo tendrá la opción de establecer ese examen como plantilla de corrección. Si, por el contrario, el usuario ya ha establecido una plantilla de corrección el controlador comunicara a la vista la activación del botón de obtener la nota del alumno.

Cuando el profesor obtiene la nota, el proceso de corrección para ese alumno ha concluido. Al calcular la nota, el controlador vuelve a comunicar a la vista la activación del botón que permite al usuario almacenar la nota y limpiar la vista para continuar con el examen del siguiente alumno.

#### 4.5.5. Eliminar calificación

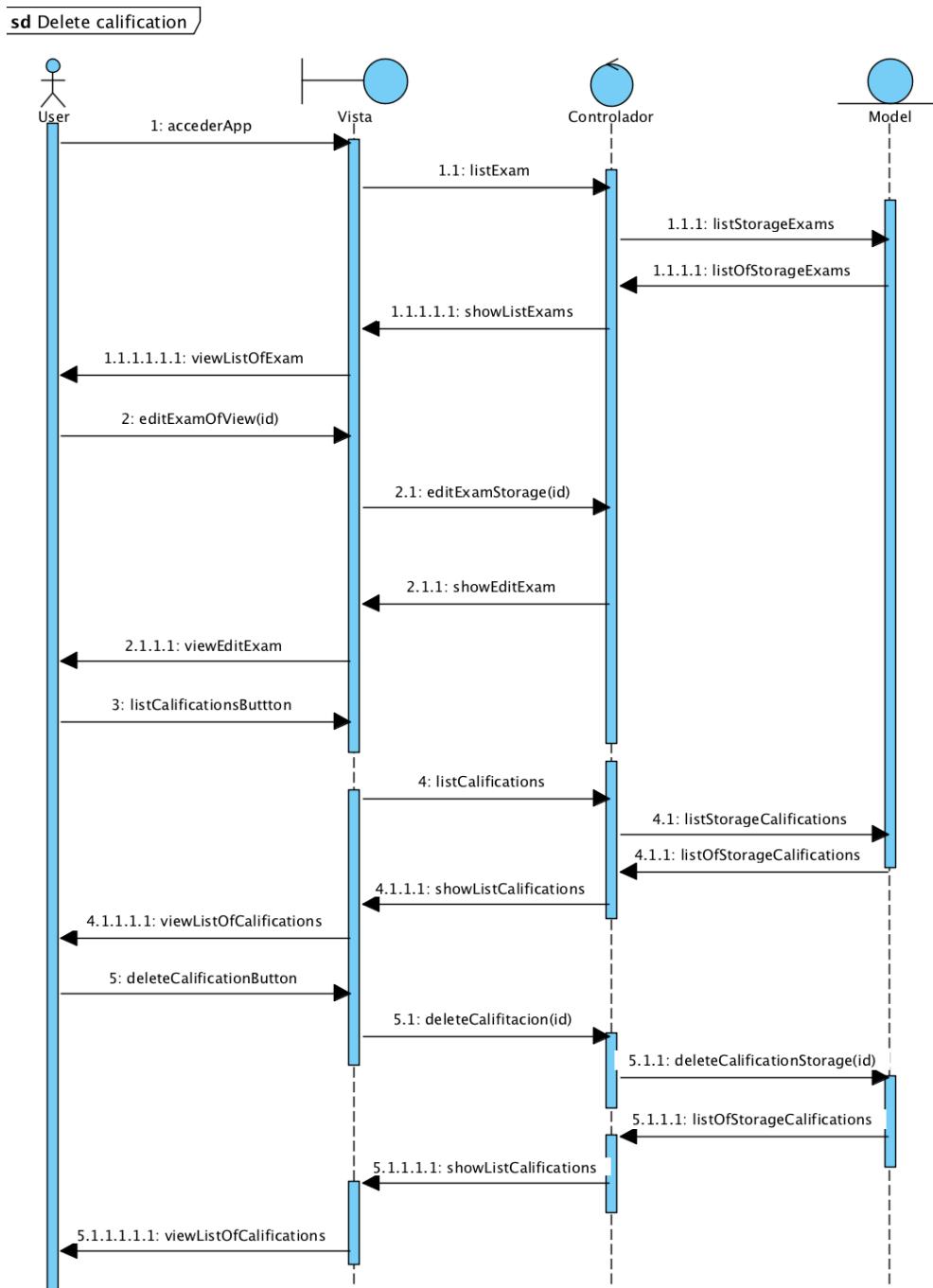


Ilustración 31 - Diagrama secuencia eliminar calificación

En esta secuencia, el usuario, al abrir la App, accede al listado de exámenes almacenados, pulsando sobre alguno de ellos se le presenta vista de edición del examen. Si el examen seleccionado tiene almacenados resultados, el usuario podrá acceder al listado de estos.

En este listado, al igual que en listado de exámenes, el usuario podrá deslizar el dedo de derecha a izquierda sobre la celda del resultado que desea eliminar y aparecerá botón para eliminarla.

#### 4.5.6. Exportar calificaciones

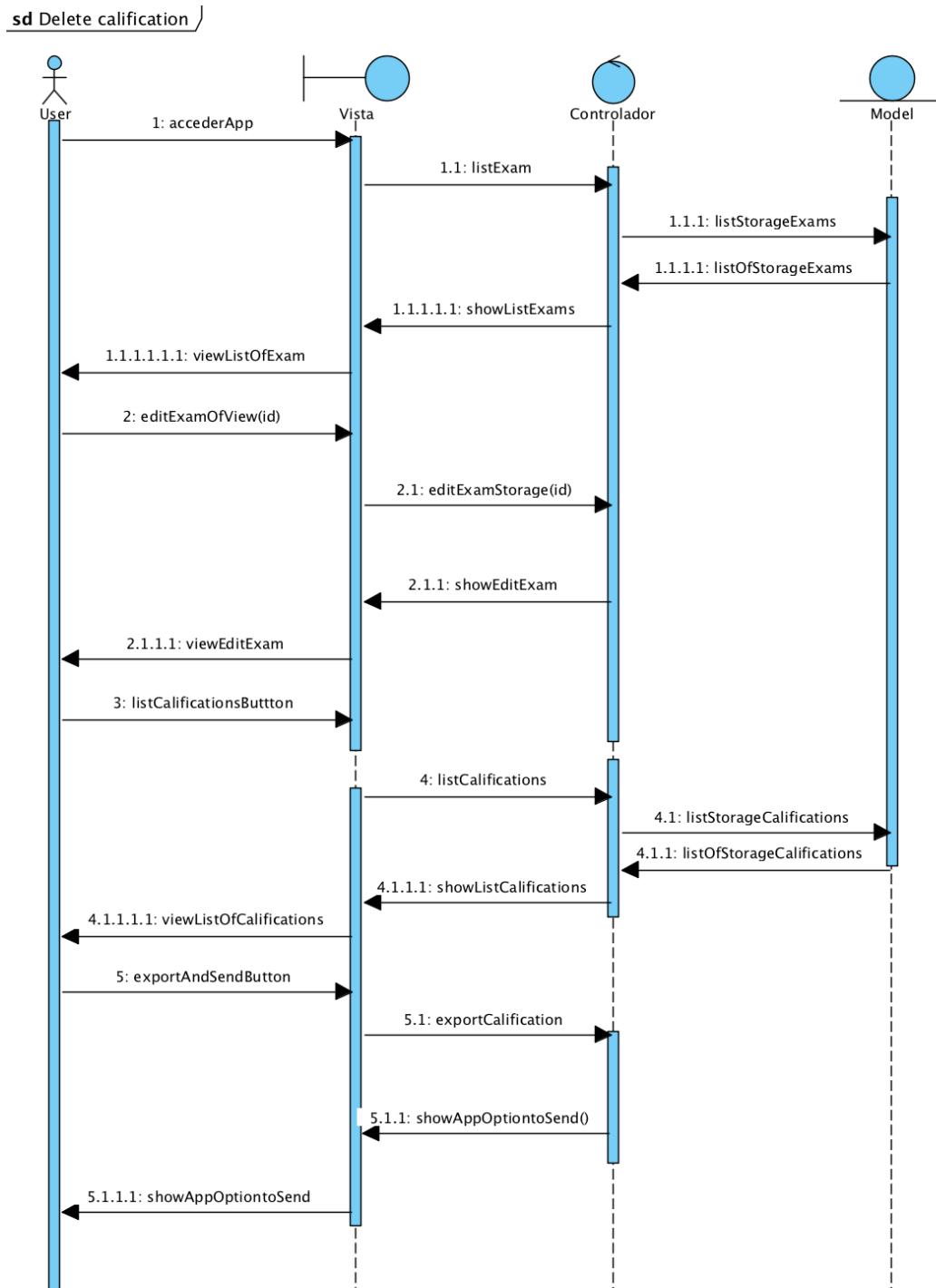


Ilustración 32 - Diagrama secuencia exportar calificaciones

Desde el listado de calificaciones, al que hemos explicado en la secuencia anterior la manera de llegar a él, se puede realizar la exportación de los resultados a un fichero ‘csv’ y mostrar al usuario una lista de aplicaciones disponibles para enviar o compartir este tipo de fichero como por ejemplo son las aplicaciones de correo o mensajería.

#### 4.6. StoryBoards

Este apartado está dedicado al diseño de un prototipo para la interfaz de usuario que tendrá la aplicación. También se ilustrará las distintas opciones que tendrá el usuario y sus resultados.

El uso de StoryBoard, ayuda a tomar una primera visión de cómo la aplicación final, debería llegar a mostrarse al usuario.

A continuación, se muestran las imágenes cómo será la interfaz.

La pantalla principal, muestra las dos opciones que tiene el usuario, realizar una foto para empezar la corrección directamente o consultar el listado de exámenes almacenados.

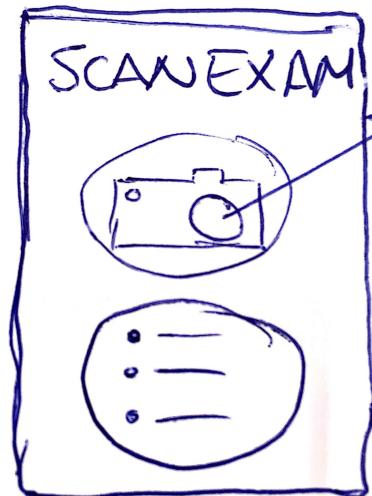


Ilustración 33 - Storyboard principal

La primera vez que el usuario ejecute la App, es muy probable que la primera opción que elija sea la de tomar la foto del examen, por lo que se seguirá ese itinerario y después se continuara por el listado.



Ilustración 34 - Storyboard cámara

La vista de la cámara se compone de tres botones al iniciarla:

- Home: Arriba a la izquierda, este botón permite al usuario volver a la pantalla principal de la aplicación. Este ícono estará presente en casi todo momento.
- Flash: Abajo a la izquierda, este botón es utilizado para entornos con poca iluminación, activa el flash del dispositivo.
- Filter: A la derecha del botón anterior, permite cambiar el filtro en tiempo real para capturar la imagen a color o blanco y negro.

Falta mencionar la funcionalidad más importante de esta vista, el botón central redondo, utilizado para capturar la foto del examen. La siguiente imagen muestra la respuesta de la App al presionar el botón para capturar la imagen.

Ofrece una detección del examen mientras se está enfocando con la cámara, superponiendo un cuadro a modo de indicación para tomar la foto en el momento correcto.

Al capturar la imagen, el cuadrante rojo detectado es tomado como la imagen principal al que se le corrige la perspectiva y de esta forma permitir un mayor ángulo para hacer la fotografía.

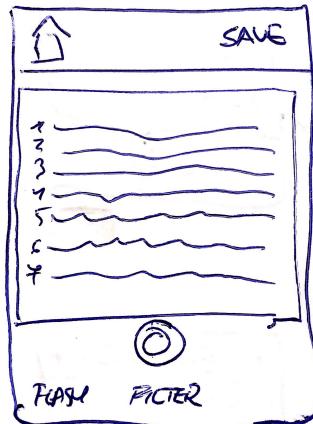


Ilustración 35 - Storyboard captura foto

Al capturar la imagen, esta es presentada ya con la corrección de perspectiva aplicada, para que el usuario decida si la selecciona o la descarta, si decide utilizarla, presionara en el botón Save que aparece al presentar la foto tomada. Si la imagen tomada no le parece de calidad, solo pulsando la imagen tomada podrá volver a tomar otra foto.

La siguiente vista es el resultado de la confirmación de la fotografía tomada al examen. El usuario es redirigido a la vista de creación de un nuevo examen.

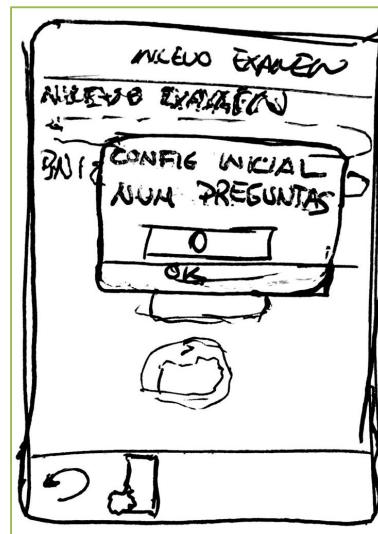


Ilustración 36 - Storyboard configuración inicial nuevo examen

Nada más presentar la vista, se muestra al usuario una alerta para la configuración inicial y obligatoria del examen. En este paso, el usuario no tiene más opción que introducir el número de preguntas.

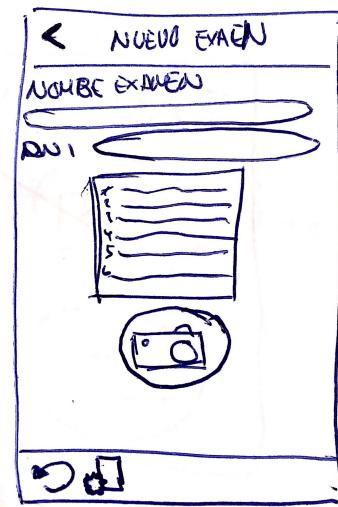


Ilustración 37 - Storyboard vista nuevo examen

Una vez completada la configuración inicial, se obtiene la presentación completa de la vista de creación. Cuenta con una pequeña vista de la imagen del examen y un menú inferior con la opción de rotarla, si es necesario.

El otro botón disponible en el menú inferior es el de procesar la imagen para detectar las respuestas del alumno.

A continuación, se muestra una imagen con las actualizaciones que afectan a la vista según avanza el proceso de corrección.

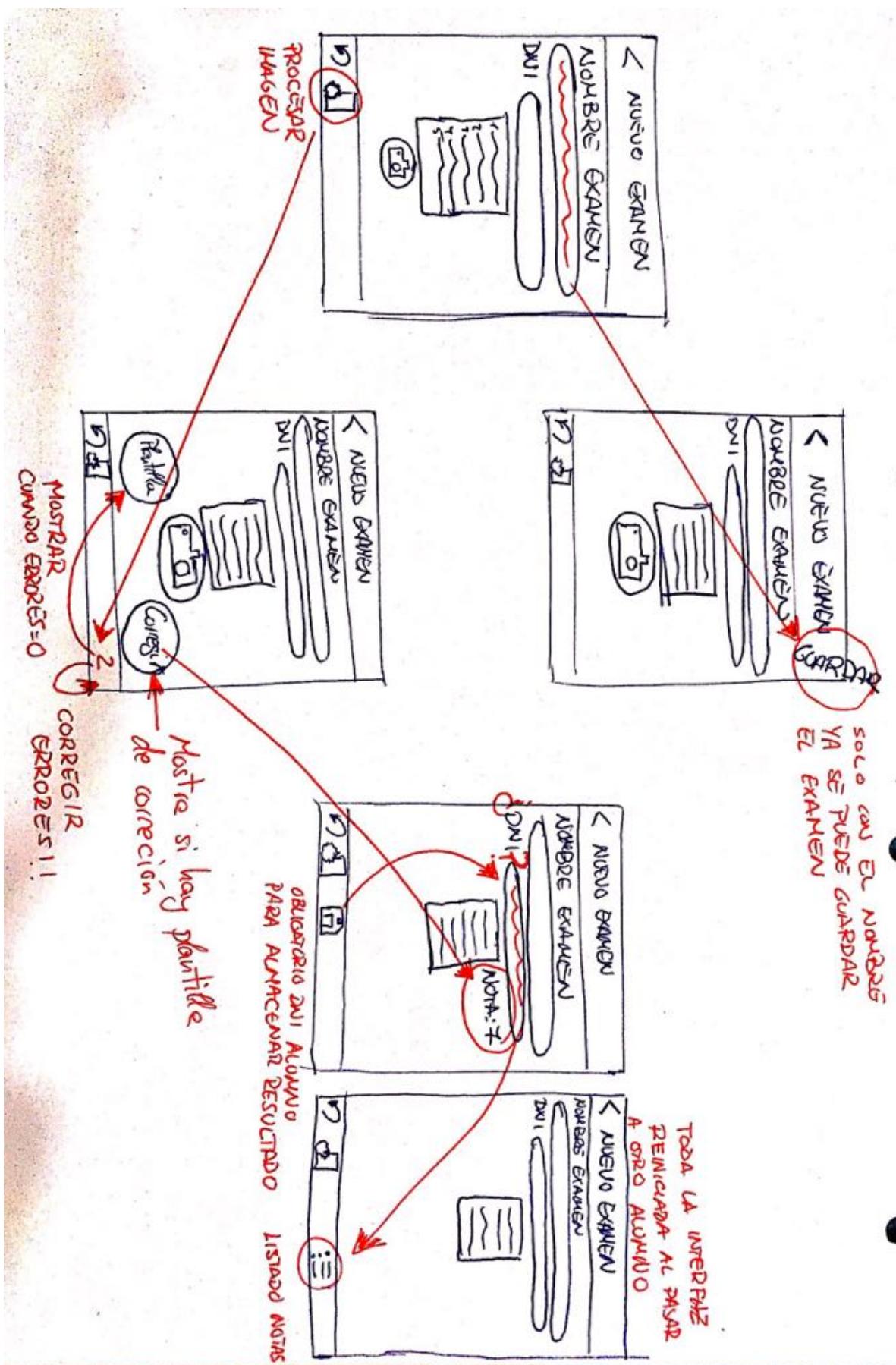


Ilustración 38 - Storyboard actualizaciones vista nuevo examen

La primera flecha hacia arriba indica que un examen se podrá guardar nada más que introduciendo un nombre para este.

La segunda flecha indica el procesamiento de la imagen y detección de las respuestas del alumno, en la segunda vista por abajo aparece una alerta si existen errores en la detección de las respuestas y el botón para establecer la plantilla en el momento que todas las respuestas están correctamente detectadas.

Si la plantilla ya está establecida, en el momento que todas las respuestas estén correctamente detectadas, el botón para obtener la nota se activa en el mismo momento.

Una vez calculada la nota, ya puede ser almacenada y pasar a la corrección del siguiente alumno, siempre y cuando el identificador del alumno este completado.

Finalizado el primer itinerario, se continuará por el listado de exámenes. La apariencia del listado será algo similar a la siguiente imagen.



Ilustración 39 - Storyboard listado exámenes

Desde esta vista, se puede acceder a la creación de un nuevo examen con el botón situado a la derecha del menú superior. Un examen puede ser seleccionado para ser modificado o continuar la corrección por donde se quedó. El gesto de deslizar el dedo de derecha a izquierda por encima de alguna de las celdas, dará la opción de eliminarla de la tabla y consigo, el examen que contiene.

A continuación, se muestra una imagen donde se representa las distintas interacciones con esta vista.

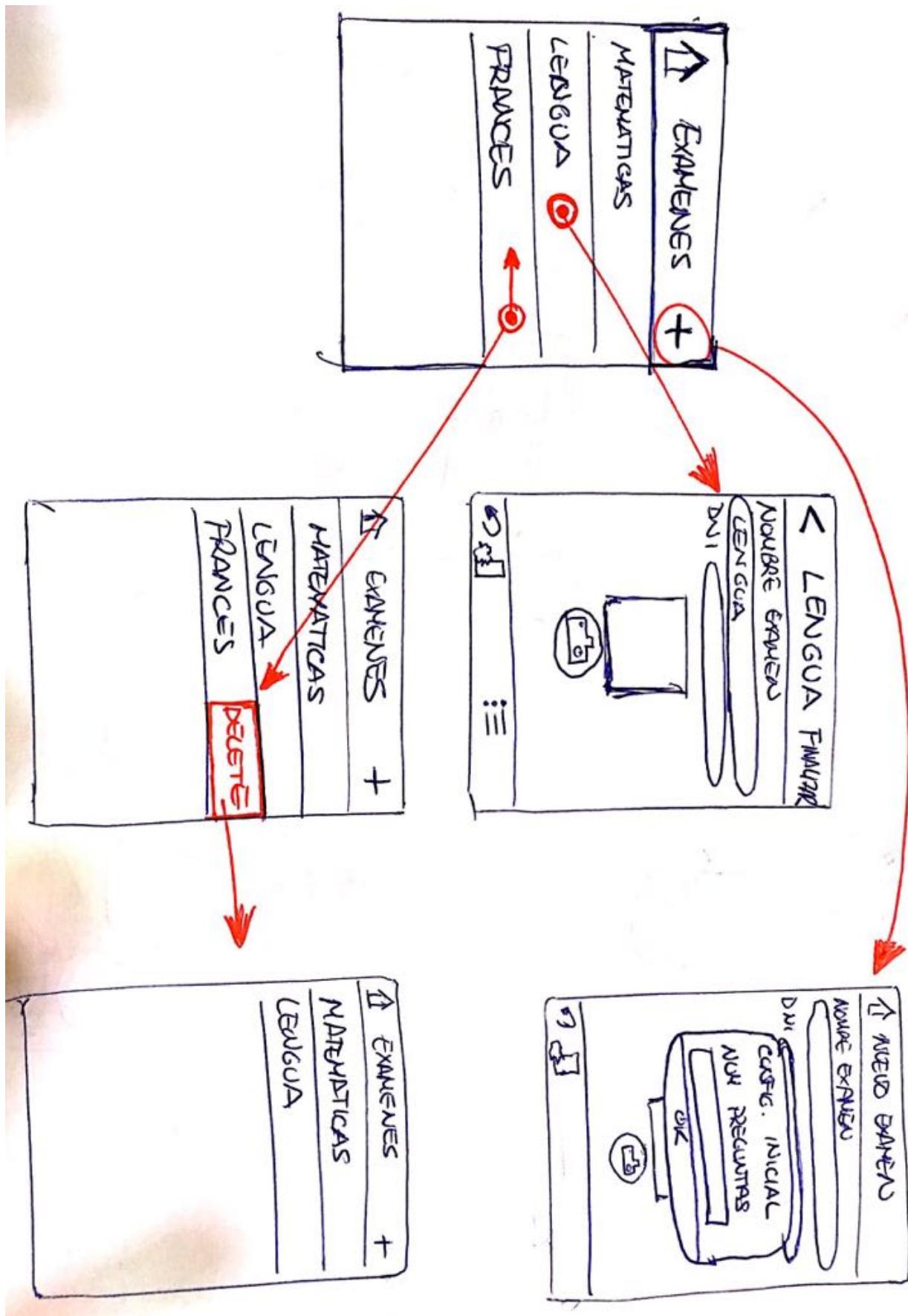


Ilustración 40 - Storyboard actualizaciones vista listado exámenes

Para finalizar este apartado de Storyboard se muestra la vista de lo que será los listados de calificaciones.

Como se puede observar es muy parecida al listado de exámenes, con la misma funcionalidad de arrastrar el dedo sobre la celda de la calificación que por algún motivo debe ser eliminada.

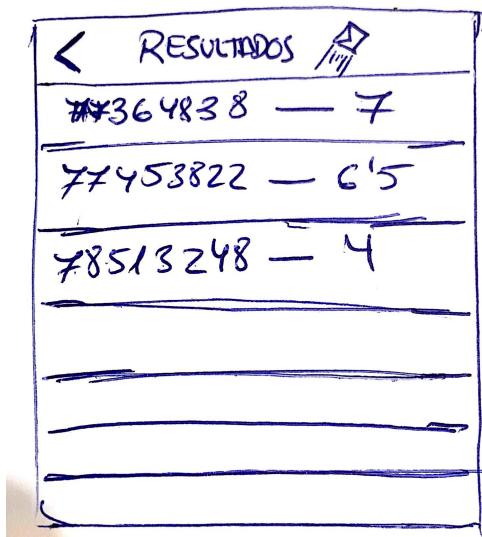


Ilustración 41 - Storyboard listado calificaciones

En el menú superior, a la derecha, hay un ícono de un sobre, este será utilizado para realizar la exportación de las calificaciones a un formato csv y compartirlas mediante las aplicaciones que lo permitan Finalizado el primer itinerario, se continuará por el listado de exámenes. La apariencia del listado será algo similar a la siguiente imagen.

## 5. Implementación

En este capítulo hablaremos sobre los componentes utilizados en la implementación, explicando detalladamente el uso de cada uno, así como de su función en este proyecto.

### 5.1. Entorno de desarrollo

El entorno de desarrollo utilizado en la implementación de la aplicación es Xcode en su versión 8.3, desarrollado y propiedad de Apple. Con total integración con Cocoa y Cocoa Touch frameworks. Es el entorno para el desarrollo de apps para Mac, iPhone, iPad, Apple Watch, and AppleTV más productivo que hay actualmente.



Ilustración 42 - XCode

Las ventanas dentro de XCode funcionan como espacios de trabajo o workspaces, y lo bueno es que podemos “personalizarlo” para que se adapte a la forma en la que queremos trabajar: puede contener un simple editor de texto, así como un editor gráfico, una librería multimedia, y un debugger, entre otras cosas.

El hecho de poder usar workspaces también permite que el usuario tenga la oportunidad de trabajar en varios proyectos a la vez, como sucede en la realidad. Gracias al uso de los espacios de trabajo podemos subdividir las partes de una aplicación o de lo que sea en que estemos trabajando de forma limpia y organizada, viendo todos los archivos disponibles en todos los proyectos, pero limitándonos a trabajar en uno solo.

## 5.2. Lenguajes de programación

### 5.2.1. Lenguaje programación C



Ilustración 43 - Lenguaje C

Lenguaje desarrollado en el año 1972 por Dennis Ritchie para UNIX. Es del tipo lenguaje estructurado. Incluye sentencias como if,else,for, do y while. Se caracteriza por tener una gran facilidad para escribir código compacto y sencillos a su vez.

### 5.2.2. C++



Ilustración 44 - C++

Desarrollado en el año 1983 por Bjarne Stroustrup, en los laboratorios AT&T. Es un derivado del lenguaje C. C++ intenta llevar a C hacia un paradigma de clases y objetos, con lo que pretende ayudar al programador en la tarea de abstracción de objetos de la realidad y gestionarlos más fácilmente en el código.

### 5.2.3. Objective-C



Ilustración 45 - Objective C

Es un lenguaje basado en C, desarrollado originalmente en 1980 y usado por Apple en sus sistemas operativos. Con soporte para la programación orientada a objetos. Hereda la sintaxis, primitivas y el control de flujo de las instrucciones de C y añade la sintaxis para la definición de clases y métodos.

#### 5.2.4. Swift



Ilustración 46 - Swift

Es un lenguaje de programación potente e intuitivo para el desarrollo software en cualquiera de las plataformas de Apple.

Es el resultado del análisis de los distintos lenguajes de programación disponibles, combinado con décadas de experiencia en el desarrollo de software por parte de Apple.

Actualmente se encuentra en su versión 4 desarrollado bajo licencia Apache 2.0. Toda la información necesaria, acerca del desarrollo y código fuente se encuentra disponible en Swift.org. Dispone de una comunidad muy activa de desarrolladores en la que poder consultar dudas.

Las características resaltadas en su documentación son:

- Seguro: Las variables utilizadas son siempre inicializadas antes de su uso, se comprueba la posibilidad de desbordamiento para arrays y enteros y la memoria es manejada automáticamente.
- Rápido: Utiliza el compilador LLVM de alto rendimiento, transformando el código Swift en código nativo optimizado para el máximo aprovechamiento del hardware.
- Expresivo: Se beneficia de décadas de avances informáticos para ofrecer una sintaxis fácil de entender.

### 5.3. Formato plantilla examen

Uno de los requisitos de este proyecto, es poder utilizar una plantilla de examen que sea fácil de editar para los docentes y cómodo de responder para los alumnos.

#### 5.3.1. Plantillas de prueba

A continuación, se muestra varios de los modelos con lo que se ha trabajado y testeado la detección de respuestas.

Seguridad en Tecnologías de la Información	V	F
10 de febrero de 2012		
Apellidos y Nombre:		
D.N.I.:		
1. Las copias de seguridad no deben estar físicamente en el mismo lugar que el sistema.	.1.-	[ ] [ ]
2. Un sistema se considera seguro si la probabilidad de que se produzca un robo de datos, una manipulación, o una interrupción del servicio, está por debajo de un límite tolerable.	.2.-	[ ] [ ]
3. En Criptografía, hay que suponer que un atacante lo sabe todo sobre nuestro sistema, excepto los valores concretos de las claves que emplea.	.3.-	[ ] [ ]
4. El control de acceso por contraseñas es un método que deposita la responsabilidad en el usuario.	.4.-	[ ] [ ]
5. Una Honeynet captura habitualmente poca información, pero de mucho valor.	.5.-	[ ] [ ]
6. Lo más razonable es tener una copia de seguridad, que se irá sobrescribiendo cada vez que se genere otra.	.6.-	[ ] [ ]
7. Una condición de carrera se produce cuando dos recursos de red compiten por acceder a un mismo proceso.	.7.-	[ ] [ ]
8. El elemento más característico de una honeynet de segunda generación (Gen1) es el honeypot.	.8.-	[ ] [ ]
9. Existen programas maliciosos que "nadan expresamente para eliminar vulnerabilidades de los sistemas.	.9.-	[ ] [ ]
10. Un bugtrap es una lista de correo electrónico sobre vulnerabilidades y temas de seguridad.	.10.-	[ ] [ ]
11. Es inadmisible que Internet sea una red insegura: deberían prohibirlo mientras puedan cometerse en ella crímenes tales como descargar discos de Chenoa, Ramoncín o Jarabe de Palo.	.11.-	[ ] [ ]
12. AES (Rijndael) es uno de los algoritmos criptográficos más modernos, y está considerado como muy seguro.	.12.-	[ ] [ ]
13. Es conveniente tener la misma contraseña para los servicios esenciales, ya que así sería más fácil comprometerlos.	.13.-	[ ] [ ]
14. Una asociación de seguridad en IPSec es una relación unidireccional entre un emisor y un receptor que ofrece servicios de seguridad al tráfico que transporta.	.14.-	[ ] [ ]
15. Las vulnerabilidades de un sistema pueden ser debidas al diseño - no, al uso o a la implementación.	.15.-	[ ] [ ]
16. Podemos considerar que todos los sistemas informáticos tienen vulnerabilidades.	.16.-	[ ] [ ]
17. Un sistema de archivos emplea el control de accesos para garantizar la confidencialidad y disponibilidad de los datos.	.17.-	[ ] [ ]
18. La ingeniería social consiste en aprovechar vulnerabilidades en redes como Facebook o Twitter.	.18.-	[ ] [ ]
19. El algoritmo de Diffie-Hellman es un refinamiento sobre el algoritmo RSA.	.19.-	[ ] [ ]
20. La inversión en seguridad en una empresa debe ser la necesaria para hacerla totalmente segura.	.20.-	[ ] [ ]
21. Las expectativas de seguridad representan el funcionamiento ideal de un sistema.	.21.-	[ ] [ ]
22. El algoritmo DES no se considera seguro porque puede romperse por la fuerza bruta.	.22.-	[ ] [ ]
23. Los cifrados monoalfabéticos son aquellos en los que una letra del texto claro se convierte siempre en la misma letra en el texto cifrado.	.23.-	[ ] [ ]
24. El algoritmo RC4 emplea 256 S-Cajas.	.24.-	[ ] [ ]
25. Una auditoría de seguridad es el proceso de generar, almacenar y revisar eventos de un sistema de forma cronológica.	.25.-	[ ] [ ]
26. La criptografía clásica permite resolver el problema de la distribución de claves.	.26.-	[ ] [ ]
27. Los distintos usuarios de un sistema operativo poseen cuentas, con una serie de privilegios asociados.	.27.-	[ ] [ ]
28. Los ataques a un sistema informático solo pueden ser llevados a cabo por personas con muchos conocimientos y gran motivación.	.28.-	[ ] [ ]
29. Se dice que un Sistema de Detección de Intrusos da respuestas activas cuando está programado para contraatacar a los intrusos que detecta.	.29.-	[ ] [ ]
30. Las copias de seguridad o backups deben realizarse con frecuencia.	.30.-	[ ] [ ]

Ilustración 47 - Plantilla pruebas con corchetes a la derecha

Se ha realizado el diseño de distintos tipos de plantilla, probado con distintos caracteres que delimiten las respuestas del alumno y posicionarlas en distintas localizaciones dentro documento.

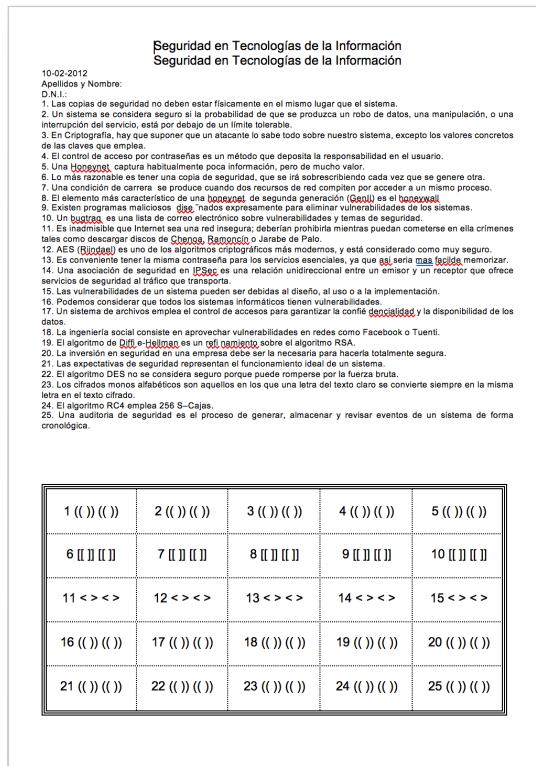


Ilustración 48 - Plantilla pruebas cuadrante respuestas inferior

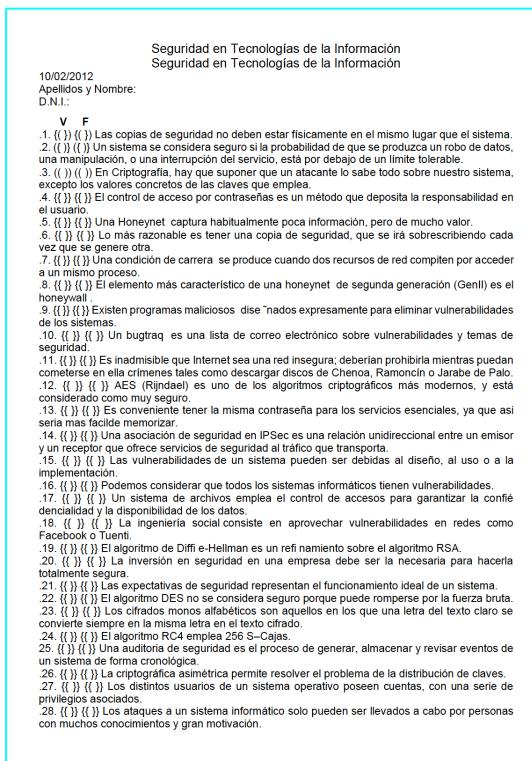


Ilustración 49 - Plantilla prueba corchetes izquierda

### 5.3.2. Conclusión

Tras analizar los resultados de reconocimiento de texto obtenidos y observando en que situaciones trabaja mejor y tiene mayor rendimiento. Se ha definido la plantilla por defecto debe cumplir los siguientes requisitos:

- La fuente debe ser todo el documento tiene que ser Arial
- El tamaño de la fuente de delimitadores de las respuestas, el número de pregunta y delimitador del D.N.I. al 13.
- Los delimitadores de respuestas, seguirán el siguiente formato, respetando espacios en blanco entre caracteres: ((nº pregunta)) (( )) (( ))
- El delimitador para identificar el DNI del alumno es el siguiente: ((DNI)). Este es opcional, pero el profesor tendría que introducirlo a mano en el momento de almacenar la calificación.
- El examen está organizado en una tabla con dos columnas, a la izquierda se encuentra el lugar donde responder y a la derecha la redacción de las preguntas.
- El ancho de las filas tiene que ser de 0,8cm.

La plantilla descrita en los puntos anteriores queda de la siguiente forma:

Título de la asignatura	
Apellidos, Nombre: ((DNI)) 77364731E V F	
((1)) (( )) (( ))	Las copias de seguridad no deben estar físicamente en el mismo lugar que el sistema.
((2)) (( )) (( ))	Un sistema se considera seguro si la probabilidad de que se produzca un robo de datos, una manipulación, o una interrupción del servicio, está por debajo de un límite tolerable.
((3)) (( )) (( ))	En Criptografía, hay que suponer que un atacante lo sabe todo sobre nuestro sistema, excepto los valores concretos de las claves que emplea.
((4)) (( )) (( ))	El control de acceso por contraseñas es un método que deposita la responsabilidad en el usuario.
((5)) (( )) (( ))	Una HoneyNet captura habitualmente poca información, pero de mucho valor.
((6)) (( )) (( ))	Lo más razonable es tener una copia de seguridad, que se irá sobrescribiendo cada vez que se genere otra.
((7)) (( )) (( ))	Una condición de carrera se produce cuando dos recursos de red compiten por el mismo recurso.
((8)) (( )) (( ))	El elemento más característico de una honeynet de segunda generación (GenII) es el honeypot.
((9)) (( )) (( ))	Existen programas maliciosos diseñados expresamente para eliminar vulnerabilidades de los sistemas.
((10)) (( )) (( ))	Un bugtraq es una lista de correo electrónico sobre vulnerabilidades y temas de seguridad.
((11)) (( )) (( ))	Es inadmisible que Internet sea una red insegura; deberían prohibirlo mientras puedan cometerse en ella crímenes tales como descargar discos de Chino, Ramoncín o Jarabe de Palo.
((12)) (( )) (( ))	Algoritmo Rijndael es uno de los algoritmos criptográficos más modernos, y está siendo usado como muy seguro.
((13)) (( )) (( ))	Es conveniente tener la misma contraseña para los servicios esenciales, ya que así sería más fácil de memorizar.
((14)) (( )) (( ))	Una asociación de seguridad en IPsec es una relación unidireccional entre un emisor y un receptor que ofrece servicios de seguridad al tráfico que transporta.
((15)) (( )) (( ))	Las vulnerabilidades de un sistema pueden ser debidas al diseño, al uso o a la implementación.
((16)) (( )) (( ))	Podemos considerar que todos los sistemas informáticos tienen vulnerabilidades.
((17)) (( )) (( ))	Un sistema de archivos emplea el control de acceso para garantizar la confidencialidad y disponibilidad de los datos.
((18)) (( )) (( ))	La ingeniería social consiste en aprovechar vulnerabilidades en redes como Facebook o Twitter.
((19)) (( )) (( ))	El algoritmo de Diffi e-Hellman es un rellamamiento sobre el algoritmo RSA.
((20)) (( )) (( ))	La inversión en seguridad en una empresa debe ser la necesaria para hacerla totalmente segura.
((21)) (( )) (( ))	Las expectativas de seguridad representan el funcionamiento ideal de un sistema.
((22)) (( )) (( ))	El algoritmo DES no se considera seguro porque puede romperse por la fuerza bruta.
((23)) (( )) (( ))	Los cifrados monos alfabeticos son aquellos en los que una letra del texto claro se convierte siempre en la misma letra en el texto cifrado.
((24)) (( )) (( ))	El algoritmo RC4 emplea 256 S-Cajas.
((25)) (( )) (( ))	Una auditoría de seguridad es el proceso de generar, almacenar y revisar eventos de un sistema de forma cronológica.
((26)) (( )) (( ))	La criptografía asimétrica permite resolver el problema de la distribución de claves.

Ilustración 50 - Plantilla examen con bordes

En la imagen anterior se muestra la plantilla con los bordes de la tabla visibles, en el momento de realizar la impresión en papel es importante anular la visibilidad de estos, quedando de la siguiente forma.

Título de la asignatura		
Apellidos, Nombre:		
((DNI))	77364731E	
V	F	
((1))	(( )) (( ))	Las copias de seguridad no deben estar físicamente en el mismo lugar que el sistema.
((2))	(( )) (( ))	Un sistema se considera seguro si la probabilidad de que se produzca un robo de datos, una manipulación, o una interrupción del servicio, está por debajo de un límite tolerable.
((3))	(( )) (( ))	En Criptografía, hay que suponer que un atacante lo sabe todo sobre nuestro sistema, excepto los valores concretos de las claves que emplea.
((4))	(( )) (( ))	El control de acceso por contraseñas es un método que deposita la responsabilidad en el usuario.
((5))	(( )) (( ))	Una HoneyNet captura habitualmente poca información, pero de mucho valor.
((6))	(( )) (( ))	Lo más razonable es tener una copia de seguridad, que se irá sobrescribiendo cada día.
((7))	(( )) (( ))	Una condición de carrera se produce cuando dos recursos de red compiten por acceder a un mismo proceso.
((8))	(( )) (( ))	El elemento más característico de una honeyNet de segunda generación (GenII) es el honeywall .
((9))	(( )) (( ))	Existen programas maliciosos diseñados expresamente para eliminar vulnerabilidades de los sistemas.
((10))	(( )) (( ))	Un bugtrap es una lista de correo electrónico sobre vulnerabilidades y temas de investigación.
((11))	(( )) (( ))	Es inadmisible que Internet sea una red insegura; deberían prohibirlo mientras puedan cometerse en ella crímenes tales como descargar discos de Cheney, Ramoncín o Jarabe de Palo.
((12))	(( )) (( ))	AES (Rijndael) es uno de los algoritmos criptográficos más modernos, y está considerado como muy seguro.
((13))	(( )) (( ))	Es conveniente tener la misma contraseña para los servicios esenciales, ya que así se minimiza la posibilidad de que se pierda.
((14))	(( )) (( ))	Una asociación de seguridad en IPsec es una relación unidireccional entre un emisor y un receptor que ofrece servicios de seguridad al tráfico que transporta.
((15))	(( )) (( ))	Las vulnerabilidades de un sistema pueden ser debidas al diseño, al uso o a la implementación.
((16))	(( )) (( ))	Podemos considerar que todos los sistemas informáticos tienen vulnerabilidades.
((17))	(( )) (( ))	Un sistema de archivos emplea el control de accesos para garantizar la confidencialidad y la disponibilidad de los datos.
((18))	(( )) (( ))	La ingeniería social consiste en aprovechar vulnerabilidades en redes como Facebook o Twitter.
((19))	(( )) (( ))	El algoritmo de Diffie-Hellman es un refinamiento sobre el algoritmo RSA.
((20))	(( )) (( ))	La inversión en seguridad en una empresa debe ser la necesaria para hacerla totalmente segura.
((21))	(( )) (( ))	Las expectativas de seguridad representan el funcionamiento ideal de un sistema.
((22))	(( )) (( ))	El algoritmo DES no se considera seguro porque puede romperse por la fuerza bruta.
((23))	(( )) (( ))	Los cifrados monos alfabeticos son aquéllos en los que una letra del texto claro se convierte siempre en la misma letra en el texto cifrado.
((24))	(( )) (( ))	El algoritmo RC4 emplea 256 S-Cajas.
((25))	(( )) (( ))	Una auditoría de seguridad es el proceso de generar, almacenar y revisar eventos de un sistema de forma cronológica.
((26))	(( )) (( ))	La criptografía asimétrica permite resolver el problema de la distribución de claves.

**Ilustración 51 - Plantilla examen final**

Siguiendo estas especificaciones, el trabajo del profesor a la hora de editar un nuevo examen es redactar las preguntas y añadir el número de esta. El examen puede constar de tantas preguntas y páginas como se necesite.

### 5.3.3. Tipos de respuesta válidos para la autocorrección

Para una correcta detección de las respuestas, es necesario que el alumno, siga unas normas:

- Para seleccionar una respuesta, simplemente tiene que pintar una barra invertida a la derecha entre los dos paréntesis, sin llegar a tocarlos.
- Si se quiere rectificar la opción escogida, se tachará la errónea dentro de los paréntesis y se elegirá la correcta.
- Todo lo que salga de entre los dos paréntesis puede provocar errores.

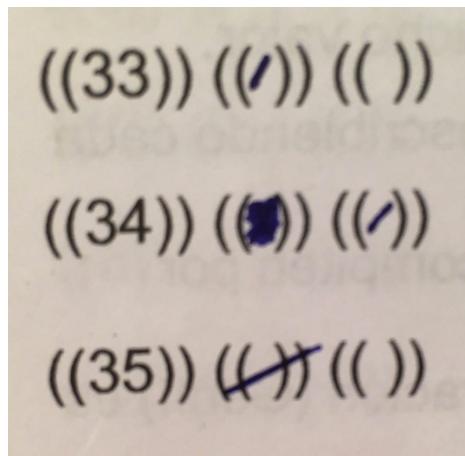


Ilustración 52 - Formato respuesta alumno

Como puede observarse, la numero 33 sería un ejemplo de una pregunta contestada correctamente. La línea no llega a tocar los paréntesis.

La número 34 se trata de una respuesta rectificada y la 33 un ejemplo de una palabra que posiblemente no sea posible detectar y clasificar correctamente.

#### 5.4. Detección de respuestas

En este apartado se explicará la implementación del proceso de reconocimiento de texto de la imagen y la detección de respuestas. Como se ha especificado en el análisis de este TFG, la librería utilizada para el reconcomiendo de texto ha sido Tesseract OCR y la de procesamiento visual OpenCV.

Consultando la documentación de Tesseract, disponible en el repositorio de GitHub, se recomienda hacer una serie de retoques a las imágenes antes de procesarlas con el motor OCR. A continuación, se explican algunas las más importantes:

- Reescalar: Aumento o reducción del tamaño físico de la imagen, alterando el número de pixeles que la componen.

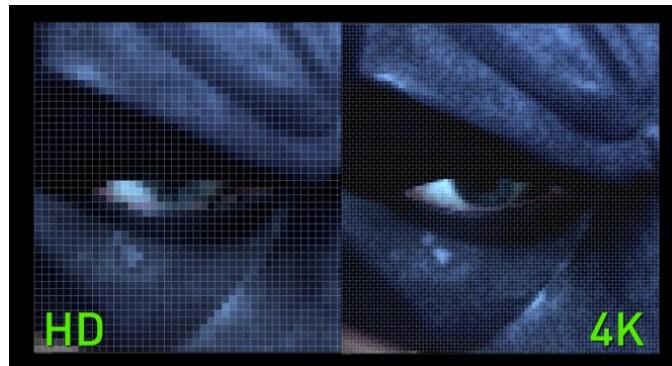


Ilustración 53 - Reescalado HD a 4K

- **Binarizar:** Consiste en un proceso de reducción de la información de una imagen digital a dos valores: 0 (negro) y 255(blanco).

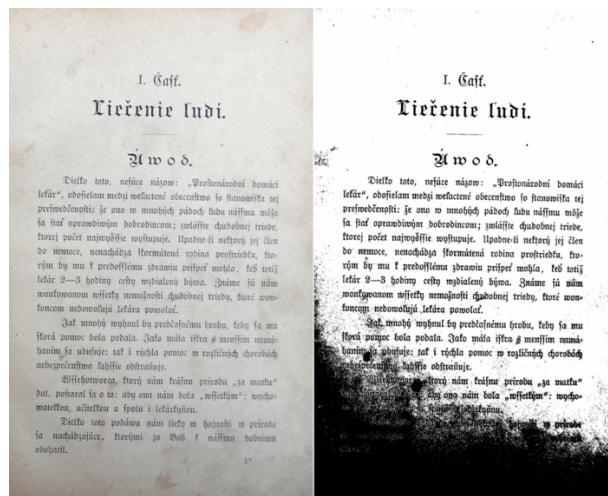


Ilustración 54 – Imagen blanco/negro

- **Eliminación del ruido:** Las cámaras producen una variación aleatoria del brillo o color de la imagen, dificultando el reconocimiento de texto.

Θεού τὸν πλάνον διέφερεν· αὐτορρυθμόν γε τούτους ἔργην  
οὐκ ἡ θεῖα ἀνθετεῖ· αἰτιώσας μήτε θεός μήτε εργάσας διέ-  
ποντας εἶναι, ἀλλὰ τὸν φεύγοντα διέσπασαν· καὶ πονηρούς  
τε πάτερας· τούτος δὲ Πλάτων ἦταν· τὸν Τιμίαν οὐδὲ εἴδεντος  
εὔποντας γράψας· τὸν τὴν πονηρήν εἰρημένα πέρι αὐτοῦ εἰ-  
λέγει· «ἀνθεταὶ μὲν οὖν τοτὲ αὐτὸν τὸ πάτεραν  
οὐντι· μὲν δὲ ἀπόδισθες, τὸς ἴητος βούλοντος, τυγχά-  
νοις· γε Οὐρανὸς ταντές δοκεῖ αἰδοντας γάρ αὐτοὺς  
πατεράγονοισιν· τοιούς γε τὸν αὐτὸν· φράσας, ἔδοντο,  
οὐ πάντας αἴθοντας τούτους αἰνάριστος εἰπεν· καὶ εἴδεντος  
καλύπτον·».<sup>1</sup>

71 Τοπερ πάρα τοῖς πράταις καὶ φιλοσόφοις πειρῶν  
οὐδὲν μὲν, μάλισταν γένοντα στιθεῖαν διαμετέρη, τοπειραῖς καὶ  
τερψις· διδούσαντα καὶ βασινόντα πορεοποιήσασαν καὶ οὐδαεις  
εἴησαντας καὶ θῆται καὶ εἰπαντας· εἰ δύνανται καὶ λέπτοι  
καὶ τοῦ πλάνου οὐλῶν διεγράφανται· Θεός τον προστρέψαντα  
τὴν πειραστὴν αἴθοντα καὶ τὴν Φεύδον καὶ Πολικαρπεῖον  
καὶ Πολεύστειον τέρπην ἐργάσαντα τὴν δέλταν προσπορεύειν·<sup>2</sup>

72 τούτον· τούτον δὲ τὸν πλάνον κατηρρέειν Σεπούριον· ὁ  
Κολορώνιος τούτου φράγιν· μᾶλλον οὐ ποτε δοκούσιν γενε-  
θεῖσις θρόνος καὶ λόρην· εἰσιθοῦν θεῖν φαντοὶ τε δύμας τε·  
καὶ πάτην· μᾶλλον εἰ τοις λέγονταί εἰρηνοῖς βόες δὲ λόγιοις  
γράφειν τελεῖν καὶ λόρη τελεῖν μόνη εὑρόντες· μάτιον μὲν  
εἰπούσας, βέσι δὲ τοντούς οὐρασας καὶ θεῖν λόρας· ἔργασσον  
καὶ σώματι εἰπούντοις τοιαῦτα, οἴστροι κατοῖν δέμας εἴρην·<sup>3</sup>

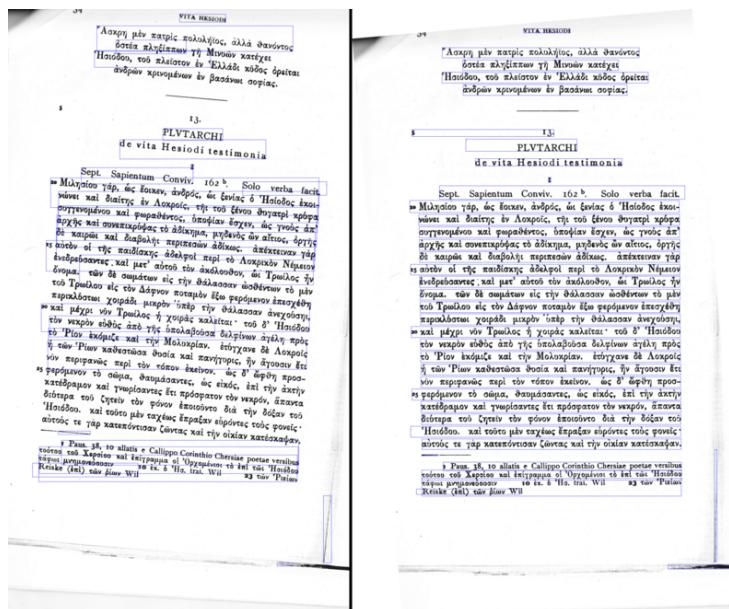
— 6 — 1. Eust. Ex. XII. 22, a. XIII. 18, 10 (Plat. Tim. p. 41 B).

9—11. Hom.—E. 344—342—340—10—9, 1. Clem. Strom. Ex. XII. 18, 100  
— Eust. Pr. XIII. 13, a. (Xenophor. ap. Clem. Strom. 14—14).

1. Καρπούς· Εἰπειν ἐπέλθειν εοτεῖς corr. S., 7. εὐτ.: μήτ BL-S.  
οὐτὸν Β. ἐπιθετεῖται M. corr. Μηρ.: ἐπιθετεῖται L. 1. ως γε οπ-  
BLMCV γρ. πανταχός K.: παντοχός L. 1. δέσμων corr. 1. εὐτ.:  
εὐτερον, loco. οὐδὲ BLMCV. 10 πίστουν corr. 1. αὐτ.:  
Καὶ 14 προτεταγμένοις Β. προτεταγμένοις K. καὶ θεούς  
εἰπούσας· οὐδὲν δέ τι προτεταγμένοις Β. προτεταγμένοις K. προ-  
θεῖται. 17 γραφούσας MCV 20 παντεῖται BL. περιοῖται M. 21. εὐ-  
πλαστοῦς; τελεῖται K. 22. εὐτ.: Λ. ε. corr. 1. εὐτ.: V. εὐτ.:  
καὶ πάτην: ή διεργάτες MSCV, 23 γράφειν MS. εἴρηται M.  
οὐδὲ ΜSC 24 οὐτε V. εἰσεῖται BLCD, sed corr. S.

## Ilustración 55 - Imagen con ruido digital

- Eliminar ángulo rotación texto: En muchas ocasiones al tomar una foto de un texto, este es capturado con algún ángulo de rotación y no totalmente en horizontal. Es necesario rectificar este desvío.



## Ilustración 56 - Rectificación ángulo rotación texto

Después de estudiar los requerimientos del motor OCR. Se ha tomado la decisión de descomponer este proceso e implementarlo en 3 partes:

- La implementación de una cámara con reconocimiento de formas que realice el cambio de perspectiva al examen.
- Implementación del procesamiento visual y reconocimiento de texto.
- Detección y clasificación de respuestas para el cálculo de la calificación.

#### 5.4.1. Cámara

Para la implementación de la cámara se ha utilizado IPDCameraViewController, una subclase de UIView implementada en Objective-C con vista previa de la cámara, detección de bordes en tiempo real y corrección de perspectiva para el examen fotografiado.

En la siguiente imagen, se realiza la detección del rectángulo y posteriormente la corrección de perspectiva de la imagen.

```
CIRectangleFeature *rectangleFeature = [self biggestRectangleInRectangles:[[self highAccuracyRectangleDetector] featuresInImage]];  
if (rectangleFeature)  
{  
    enhancedImage = [self correctPerspectiveForImage:enhancedImage, withFeatures:rectangleFeature];  
}
```

Ilustración 57 - Detección rectángulo y cambio perspectiva

El primer método realiza la detección de la forma rectangular del examen con la ayuda de la clase CoreImage.

El segundo método procesa los rectángulos devueltos por el primer método para escoger el más grande.

```
- (CIDetector *)highAccuracyRectangleDetector
{
    static CIDetector *detector = nil;
    static dispatch_once_t onceToken;
    dispatch_once(&onceToken, ^{
        detector = [CIDetector detectorOfType:CIDetectorTypeRectangle context:nil options:@{CIDetectorAccuracy : CIDetectorAccuracyHigh}];
    });
    return detector;
}

- (CIRectangleFeature *)_biggestRectangleInRectangles:(NSArray *)rectangles
{
    if (![rectangles count]) return nil;

    float halfPerimiterValue = 0;
    CIRectangleFeature *biggestRectangle = [rectangles firstObject];
    for (CIRectangleFeature *rect in rectangles)
    {
        CGPoint p1 = rect.topLeft;
        CGPoint p2 = rect.topRight;
        CGFloat width = hypotf(p1.x - p2.x, p1.y - p2.y);

        CGPoint p3 = rect.topLeft;
        CGPoint p4 = rect.bottomLeft;
        CGFloat height = hypotf(p3.x - p4.x, p3.y - p4.y);

        CGFloat currentHalfPerimiterValue = height + width;
        if (halfPerimiterValue < currentHalfPerimiterValue)
        {
            halfPerimiterValue = currentHalfPerimiterValue;
            biggestRectangle = rect;
        }
    }
    return biggestRectangle;
}
```

#### Ilustración 58 - Detección rectángulo

Este método recibe el rectángulo seleccionado por el método anterior, calcula los puntos de los extremos de la diagonal, el centro del rectángulo e identifica las coordenadas de las cuatro esquinas del rectángulo para después utilizarla en la corrección de la perspectiva.

```

- (CIRectangleFeature *)biggestRectangleInRectangles:(NSArray *)rectangles
{
    CIRectangleFeature *rectangleFeature = [self _biggestRectangleInRectangles:rectangles];
    if (!rectangleFeature) return nil;

    NSArray *points = @[[NSValue valueWithCGPoint:rectangleFeature.topLeft],[NSValue valueWithCGPoint:rectangleFeature.topRight]
        ,[NSValue valueWithCGPoint:rectangleFeature.bottomLeft],[NSValue valueWithCGPoint:rectangleFeature.bottomRight]];

    CGPoint min = [points[0] CGPointValue];
    CGPoint max = min;
    for (NSValue *value in points)
    {
        CGPoint point = [value CGPointValue];
        min.x = fminf(point.x, min.x);
        min.y = fminf(point.y, min.y);
        max.x = fmaxf(point.x, max.x);
        max.y = fmaxf(point.y, max.y);
    }

    CGPoint center =
    {
        0.5f * (min.x + max.x),
        0.5f * (min.y + max.y),
    };

    NSNumber *(^angleFromPoint)(id) = ^(NSValue *value)
    {
        CGPoint point = [value CGPointValue];
        CGFloat theta = atan2f(point.y - center.y, point.x - center.x);
        CGFloat angle = fmodf(M_PI - M_PI_4 + theta, 2 * M_PI);
        return @angle;
    };

    NSArray *sortedPoints = [points sortedArrayUsingComparator:^NSComparisonResult(id a, id b)
    {
        return [angleFromPoint(a) compare:angleFromPoint(b)];
    }];

    IPDFRectangleFeature *rectangleFeatureMutable = [IPDFRectangleFeature new];
    rectangleFeatureMutable.topLeft = [sortedPoints[3] CGPointValue];
    rectangleFeatureMutable.topRight = [sortedPoints[2] CGPointValue];
    rectangleFeatureMutable.bottomRight = [sortedPoints[1] CGPointValue];
    rectangleFeatureMutable.bottomLeft = [sortedPoints[0] CGPointValue];

    return (id)rectangleFeatureMutable;
}

```

#### Ilustración 59 - Identificación cuatro puntos

Por último, queda la corrección de perspectiva de la foto, implementada en la siguiente imagen. El método recibe las coordenadas del rectángulo y crea una imagen con ellas. Con este método se asegura una dirección del texto aceptable para el reconocimiento de texto.

```

- (CIImage *)correctPerspectiveForImage:(CIImage *)image withFeatures:(CIRectangleFeature *)rectangleFeature
{
    NSMutableDictionary *rectangleCoordinates = [NSMutableDictionary new];
    rectangleCoordinates[@"inputTopLeft"] = [CIVector vectorWithCGPoint:rectangleFeature.topLeft];
    rectangleCoordinates[@"inputTopRight"] = [CIVector vectorWithCGPoint:rectangleFeature.topRight];
    rectangleCoordinates[@"inputBottomLeft"] = [CIVector vectorWithCGPoint:rectangleFeature.bottomLeft];
    rectangleCoordinates[@"inputBottomRight"] = [CIVector vectorWithCGPoint:rectangleFeature.bottomRight];
    return [image imageByApplyingFilter:@"CIPerspectiveCorrection" withInputParameters:rectangleCoordinates];
}

```

#### Ilustración 60 - Corrección perspectiva

El ultimo método que comentar en este subapartado es el encargado de marcar el rectángulo detectado al usuario a tiempo real en la previa de la cámara. Este método, se ejecuta cíclicamente comprobando si hay coordenadas válidas para el rectángulo.

```

- (CIIImage *)drawHighlightOverlayForPoints:(CIIImage *)image topLeft:(CGPoint)topLeft topRight:(CGPoint)topRight bottomLeft:(CGPoint)bottomLeft bottomRight:(CGPoint)bottomRight
{
    CIIImage *overlay = [CIIImage imageWithColor:[CIColor colorWithRed:0.2 green:0 blue:0.9 alpha:0.5]];
    overlay = [overlay imageByCroppingToRect:image.extent];
    overlay = [overlay imageByApplyingFilter:@"CIPerspectiveTransformWithExtent" withInputParameters:@{@"inputExtent": [CIVector vectorWithCGRect:image.extent], @"inputTopLeft": [CIVector vectorWithCGPoint:topLeft], @"inputTopRight": [CIVector vectorWithCGPoint:topRight], @"inputBottomLeft": [CIVector vectorWithCGPoint:bottomLeft], @"inputBottomRight": [CIVector vectorWithCGPoint:bottomRight]}];
}

return [overlay imageByCompositingOverImage:image];
}

```

Ilustración 61 - Indicador detección rectángula en cámara

#### 5.4.2. Procesamiento visual y reconocimiento de texto

Para este proceso, se eligió como mejor opción, utilizar la librería OpenCV. El siguiente método recibe la imagen tomada en el paso anterior, la binariza, selecciona los diccionarios y es procesada con el motor OCR.

```

@IBAction func processImage(_ sender: UIBarButtonItem) {
    if let tesseract = G8Tesseract(language: "eng") {
        tesseract.delegate = self
        print("Comenzando el procesamiento de la imagen...")
        OpenCVWrapper.show(withStatus: "Procesando la imagen...")
        DispatchQueue.global().async() {
            // Hilo en segundo plano
            self.image.image = OpenCVWrapper.scale2gray(self.image.image)
            //self.image.image = OpenCVWrapper.changeScaleGrayAndCleanNoise(self.image.image)
            print("Binarización y limpieza de ruido finalizada...")

            tesseract.image = self.image.image
            tesseract.recognize()

            DispatchQueue.main.async {
                self.image.image = tesseract.image
                self.textFieldRecognized.text = tesseract.recognizedText

                //Análisis del texto reconocido por OCR
                self.processTextRecognized()
                var errorPercentage = ( self.errors.count * 100 ) / self.nPreguntas
                if(errorPercentage == 0) {
                    OpenCVWrapper.dismissProgressRing("Escaneo finalizado sin errores!")
                } else if(errorPercentage < 35) {
                    OpenCVWrapper.showError(withStatus: "Detectados algunos errores, puedes volver a captura una fotografía e intentare solucionarlos o corregirlos desde la barra inferior de la pantalla, el icono rojo de la derecha.")
                    OpenCVWrapper.dismissWithDelay()
                } else if(errorPercentage > 35) {
                    OpenCVWrapper.showError(withStatus: "Demasiados errores, ¿Quedan mas hojas?, si no es asi, deberias volver a repetir la captura del examen.")
                    OpenCVWrapper.dismissWithDelay()
                }
            }
        }
    }
}

```

Ilustración 62 - Función processImage()

La función anterior se ayuda de métodos implementados en C para la binarización de la imagen. A continuación, se muestra el código que lo realiza.

```
+ (UIImage*)scale2gray:(UIImage *)inputImage {
    //UIImage a cv::mat
    cv::Mat input = [inputImage CVMat];
    cv::cvtColor(input, input, CV_BGR2GRAY);

    //Procesamiento imagen...
    cv::Mat result;
    cv::Size size = input.size();
    cv::bitwise_not(input, input);

    return [UIImage imageWithCVMat:input];
}
```

Ilustración 63 - Función imagen escala grises

En este mismo proceso se incluyó la limpieza de ruido de la imagen. Después de varias pruebas en distintos escenarios, se comprobó que el tiempo requerido para dicha operación y el resultado obtenido, no merecían suficientemente la pena como para incluirlo. La implementación de este método es la siguiente:

```
+ (UIImage*)changeScaleGrayAndCleanNoise:(UIImage *)inputImage {
    //UIImage a cv::mat
    cv::Mat input = [inputImage CVMat];
    cv::cvtColor(input, input, CV_BGR2GRAY);
    cv::Mat result;
    cv::bitwise_not(input, result);
    cv::fastNlMeansDenoising(result, result);
    return [UIImage imageWithCVMat:result];
}
```

Ilustración 64 - Función imagen escala de grises con limpieza de ruido

Una vez realizada la modificación de la imagen a escala de grises, se procesa con el motor OCR y se extrae el texto del examen, finalizando esta parte de la implementación.

#### 5.4.3. Detección y clasificación de las respuestas

En esta última parte de la implementación se aborda la detección de las respuestas escogidas por el alumno y clasificadas según su elección.

Estas respuestas son codificadas en una lista de enteros para calcular la nota después. Una respuesta que el alumno deja en blanco es codificada en la lista como 0, la elección de respuesta Verdadera se codifica como 1 y Falsa como un 2.

Las respuestas que no son posible identificar, son incluidas en una lista de errores para que el usuario posteriormente los corrija. Estos errores pueden ser por una mala detección del número de la pregunta o una rectificación de respuesta por parte del alumno.

Esta es la primera parte de la función que analiza el texto reconocido por el motor OCR. En este fragmento de código se intenta localizar el D.N.I. del alumno mediante la marca ((DNI)). Al identificar el D.N.I., será introducido en el cuadro de texto textDNI por si el usuario necesita hacer alguna modificación antes de almacenar la nota identificada por este campo.

```
@IBAction func processTextRecognized() {
    if(self.textFieldRecognized != nil) {
        if(textFieldRecognized.text.contains("((DNI))")) {
            //Identificar la posición para la marca de comienzo del DNI
            let fullString = textFieldRecognized.text
            let searchStrs : Array<String>? = ["(DNI)"]
            let resultSearchPosition : Array<(String, Int, Int)>? = FindSubString(inputStr: fullString!, subStrings:
                searchStrs)
            //Establecer inicio y fin para dni
            let posAnswer : Int = resultSearchPosition![0].1+8
            let endAnswer : Int = (resultSearchPosition![0].1)+18
            //Creación de un rango para la extracción del texto deseado
            let start = self.textFieldRecognized.text.index((self.textFieldRecognized.text.startIndex), offsetBy: posAnswer)
            let end = self.textFieldRecognized.text.index((self.textFieldRecognized.text.startIndex), offsetBy: endAnswer)
            let myRange: Range = start..
```

Ilustración 65 - Método procesamiento texto 1

La segunda parte de la implementación de este método es la encargada de localizar el número de las preguntas con su correspondiente respuesta y enviarla al método que las analiza para clasificarlas.

Primero, localiza el número de la pregunta que se está buscando en ese momento, si no lo consigue, insertara dicha pregunta en la lista de errores. En caso afirmativo extrae su posición en el texto y a partir de esta posición, extrae un numero determinados de caracteres que agrupa el número de esta y sus dos opciones.

Una vez extraída la cadena que compone la respuesta completa, hace la llamada al método que la clasifica. Este será explicado a continuación.

Con todas las preguntas analizadas y categorizadas, comprueba si existe algún error para comunicarse al usuario y poder corregirlo. De no haber ningún error, muestra el listado completo de respuestas a un lado de la imagen

```

for numAnswer in 1...self.nPreguntas {
    if(textFieldRecognized.text.contains("(\"+numAnswer.description+\"))" & self.answersList[numAnswer] == -1) {

        //Identificar la posición para el numero de las pregunta
        let fullString = textFieldRecognized.text
        let searchStrs : Array<String>? = [\"(\"+numAnswer.description+\")\")"]
        let resultSearchPosition : Array<(String, Int, Int)>? = FindSubString(inputStr: fullString!, subStrings:
            searchStrs)
        //Establecer el inicio y fin de la respuesta
        let posAnswer : Int = resultSearchPosition![0].1
        let endAnswer : Int = (resultSearchPosition![0].1)+18
        //Creación de un rango para la extracción del texto deseado
        let start = self.textFieldRecognized.text.index((self.textFieldRecognized.text.startIndex), offsetBy:
            posAnswer)
        let end = self.textFieldRecognized.text.index((self.textFieldRecognized.text.startIndex), offsetBy:
            endAnswer)
        let myRange: Range = start..

```

---

#### Ilustración 66 - Método procesamiento texto 2

Para concluir la implementación se explicará el funcionamiento del método clasificador de respuestas según la elección del alumno. Este método recibe una cadena con la cadena extraída en el método anterior, compuesta por el número de pregunta y las dos opciones de respuesta.

Analiza la cadena y detecta las dos opciones de respuesta posible por separado. Las dos opciones son evaluadas por independiente dando un único resultado: Blanco, Verdadero, Falso o Error/Corrección.

```

// NS/NC => 0
// TRUE => 1
// FALSE => 2
func analyzeAnswer2(answer : String, numAnswer : Int) -> Bool {
    if(answer.contains("(("+numAnswer.description+")) (( )) (( )))") {
        //Respuesta en blanco
        self.answersList[numAnswer] = 0
        print(numAnswer.description+" - White")

        //Comprobar si el numero de pregunta esta en el listado de errores y borrarlo
        if(self.errors.index(of: numAnswer) != nil) {
            self.errors.remove(at: self.errors.index(of: numAnswer)!)
        }
    } else {
        var trueWhite = false
        var falseWhite = false

        //Identificación de la opción Verdadero
        var startTrue = answer.index((answer.startIndex), offsetBy: 5)
        var endTrue = answer.index((answer.startIndex), offsetBy: 11)
        if(numAnswer > 9) {
            startTrue = answer.index((answer.startIndex), offsetBy: 6)
            endTrue = answer.index((answer.startIndex), offsetBy: 12)
        }
        let myRangeTrue: Range = startTrue..

```

#### Ilustración 67 – Método clasificación de respuestas 1

En la segunda parte de este método se evalúa el resultado independiente de las opciones disponibles para clasificarla. Se realiza la comprobación de buscarlo en la lista de errores y eliminarlo o incluirlo si se trata de un error/corrección.

Gracias a estas comprobaciones se permite tomar tantas fotos como páginas tenga el examen o repetir la foto de la misma página actualizando el listado de errores en cada procesamiento de imagen.

```

if(!trueWhite && falseWhite) {
    //Opción elegida: Verdadero
    self.answersList[numAnswer] = 1
    print(numAnswer.description+" - True")

    //Check if the array have the error element and drop
    //Comprobar si esta pregunta esta en la lista de errores y borrarla
    if(self.errors.index(of: numAnswer) != nil) {
        self.errors.remove(at: self.errors.index(of: numAnswer)!)
    }

} else if(trueWhite && !falseWhite) {
    //Opción elegida: Falso
    //self.answersList.insert(2, at: numAnswer)
    self.answersList[numAnswer] = 2
    print(numAnswer.description+" - False")

    //Comprobar si esta pregunta esta en la lista de errores y borrarla
    if(self.errors.index(of: numAnswer) != nil) {
        self.errors.remove(at: self.errors.index(of: numAnswer)!)
    }

} else if(trueWhite && falseWhite) {
    //Opción elegida: NS/NC
    self.answersList[numAnswer] = 0
    print(numAnswer.description+" - White")

    //Comprobar si esta pregunta esta en la lista de errores y borrarla
    if(self.errors.index(of: numAnswer) != nil) {
        self.errors.remove(at: self.errors.index(of: numAnswer)!)
    }

} else if(!trueWhite && !falseWhite) {
    if(self.errors.index(of: numAnswer) == nil) {
        //Respuesta no reconocida correctamente
        print(numAnswer.description+" - Error, i don't know if is true or false")
        errors.append(numAnswer)
    }

}

}

return false
}

```

Ilustración 68 - Método clasificación de respuestas 2

#### 5.4.4. Exportación y envío de los resultados

Una vez clasificadas todas las respuestas, el usuario podrá calcular la calificación que corresponde al alumno y posteriormente enviarlas mediante una lista de aplicaciones que aceptan el formato csv.

Este es un formato muy conocido y bastante sencillo para el manejo de listados.

A continuación, se muestra la implementación del método encargado de las exportaciones de estas calificaciones a un fichero separado por comas.

```
@IBAction func exportAndShare(_ sender: UIBarButtonItem) {
    let fileName = "Results.csv"
    let path = NSURL(fileURLWithPath: NSTemporaryDirectory()).appendingPathComponent(fileName)
    var csvText = "DNI,Nota\n"

    for result in results {
        let newLine = "\(result.DNI),\(result.calification)\n"
        csvText.append(newLine)
    }

    do {
        try csvText.write(to: path!, atomically: true, encoding: String.Encoding.utf8)
    } catch {
        print("Failed to create file")
        print("\(error)")
    }
}

let vc = UIActivityViewController(activityItems: [path], applicationActivities: [])
present(vc, animated: true, completion: nil)

vc.excludedActivityTypes = [
    UIActivityType.assignToContact,
    UIActivityType.saveToCameraRoll,
    UIActivityType.postToFlickr,
    UIActivityType.postToVimeo,
    UIActivityType.postToTencentWeibo,
    UIActivityType.postToTwitter,
    UIActivityType.postToFacebook,
    UIActivityType.openInIBooks
]
}
```

Ilustración 69 - Método exportación y envío de resultados

## 6. Pruebas aplicación

En este apartado se mostrará mediante imágenes, la aplicación en funcionamiento y como sería el proceso de corrección y consulta de calificaciones.

El siguiente examen es el utilizado para la prueba, consta de 40 preguntas.

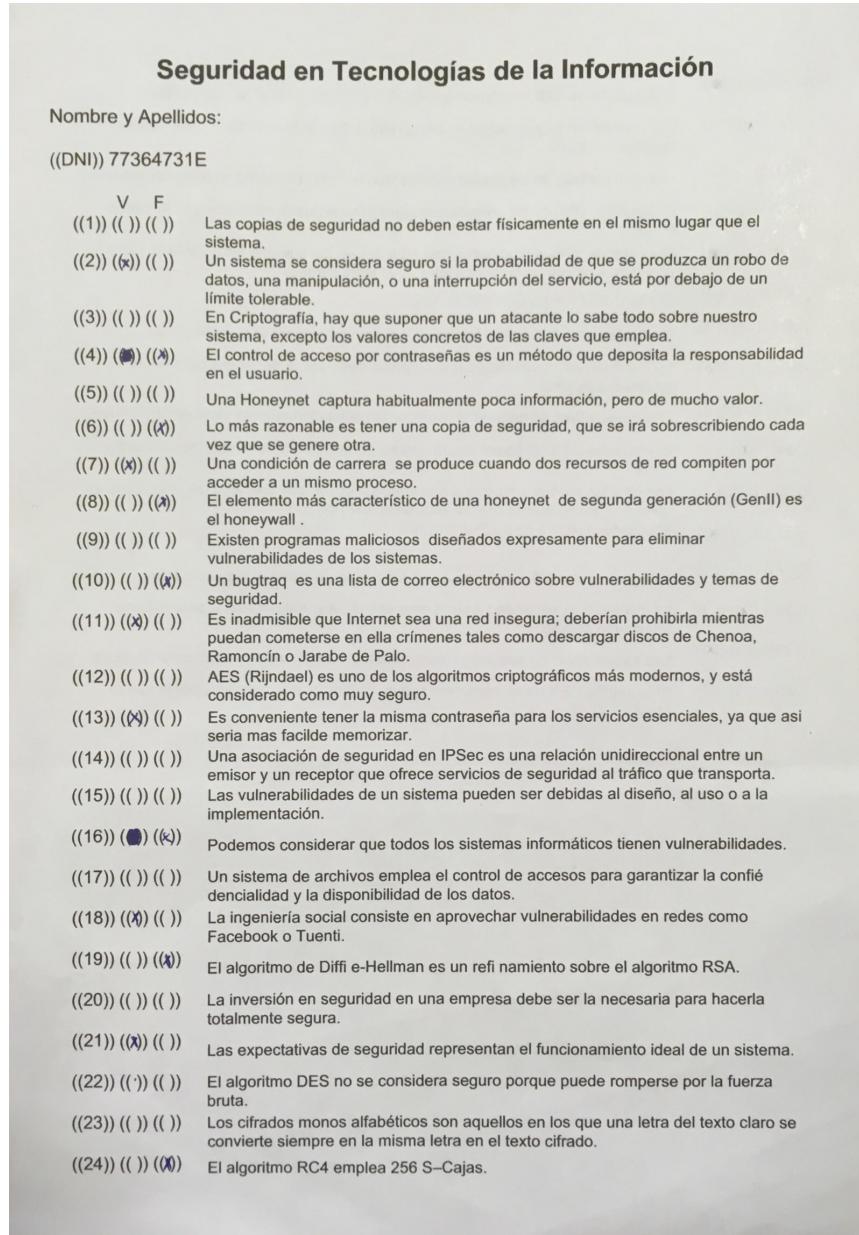


Ilustración 70 - Examen pruebas 1

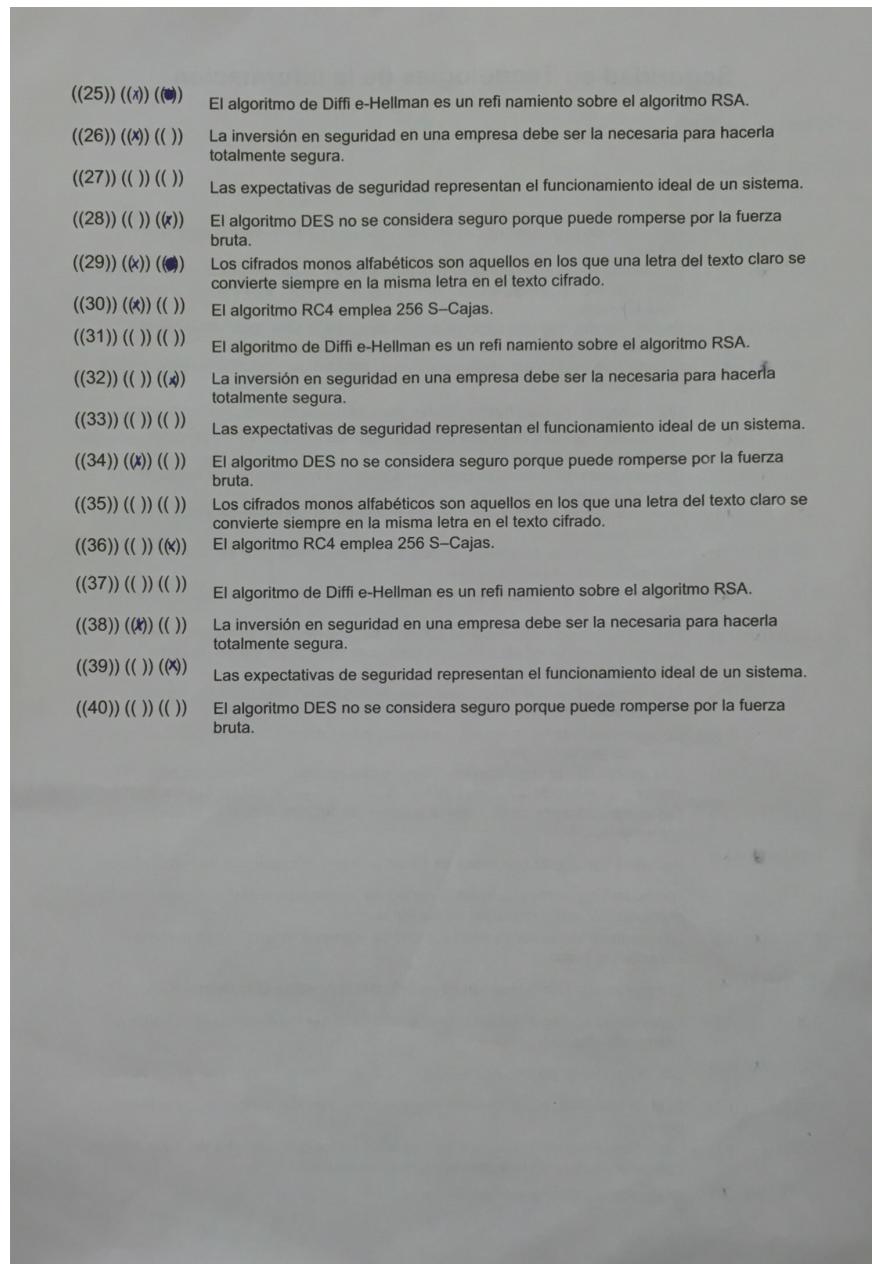


Ilustración 71 - Examen pruebas 2

## 6.1. Vista principal

Nada más entrar a la aplicación, tenemos la opción de tomar una fotografía del examen o listar los almacenados.

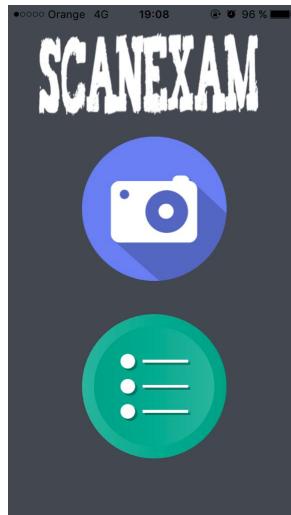


Ilustración 72 - Vista principal App

## 6.2. Tomar foto examen

En la imagen se muestra como en la previa de la cámara, al enfocar el examen, este es identificado y marcado para alertar al usuario que es momento de realizar la foto.

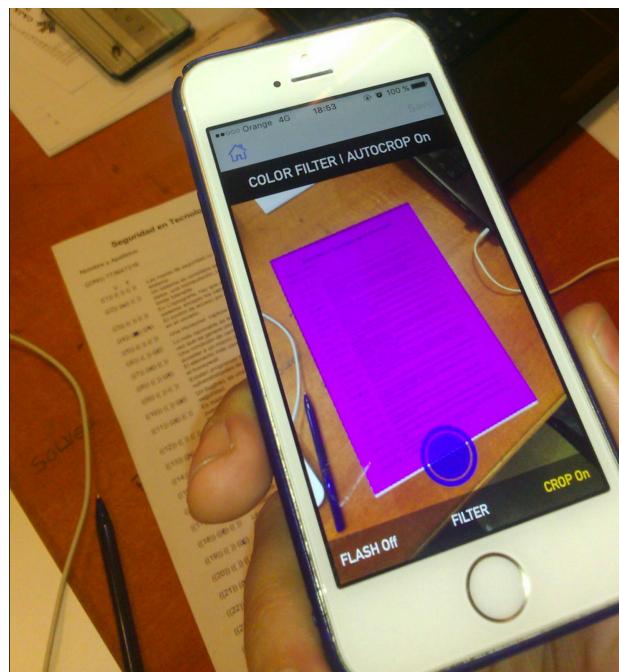


Ilustración 73 - Detección examen

En la siguiente imagen se muestra la corrección de perspectiva que se aplica a la zona donde se encuentra el examen y como se le muestra al usuario. Si no está satisfecho con la fotografía tomada, solo tiene que tocar la imagen para volver a tomar otra. Para confirmarla la fotografía tomada debe pulsar Save.

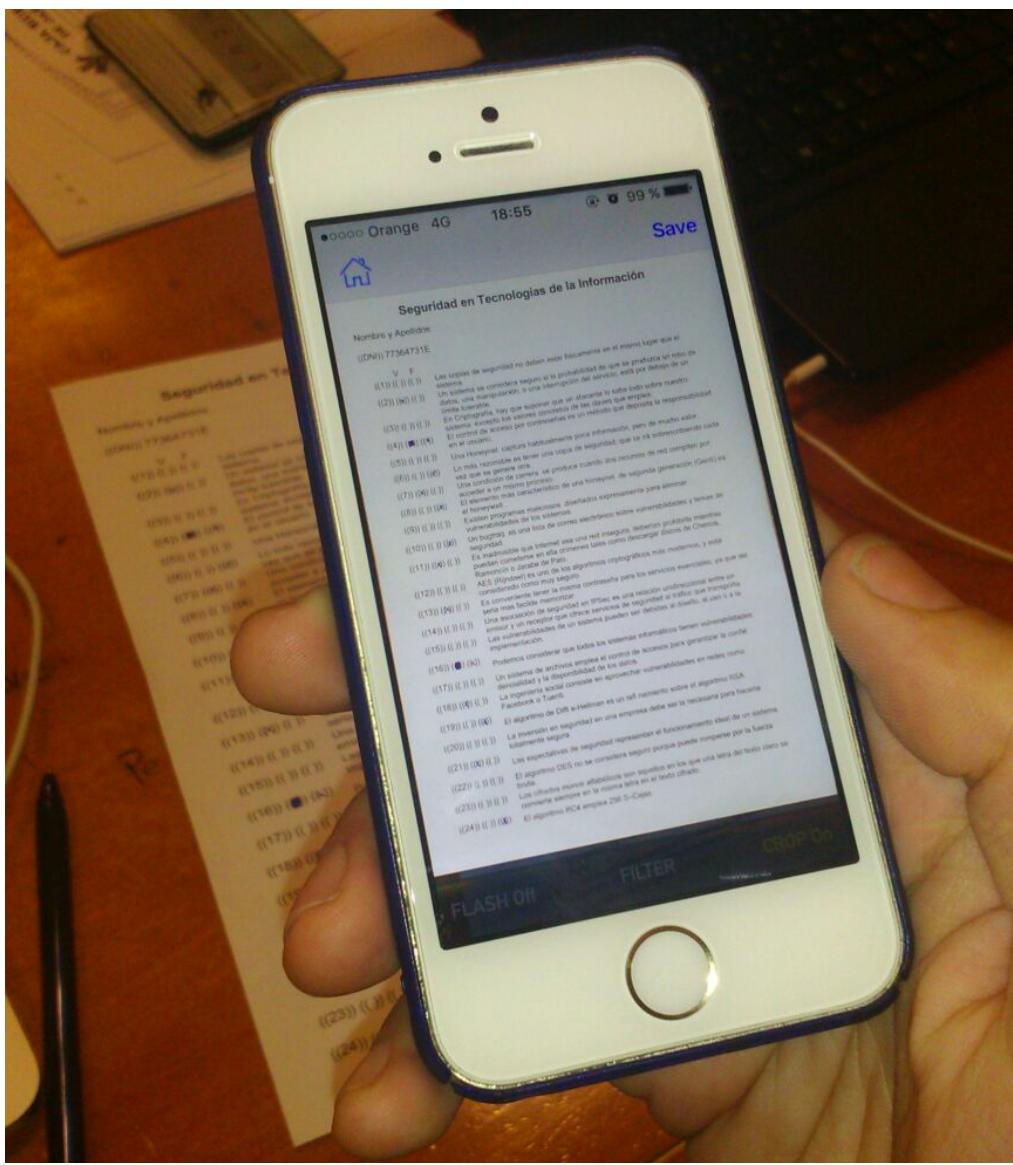


Ilustración 74 - Corrección perspectiva examen

### 6.3. Configuración inicial

Al confirmar la foto tomada, el usuario debe realizar la configuración inicial del examen, indicando el número de preguntas de este. Hasta que usuario no introduce el número de preguntas, no podrá continuar.



Ilustración 75 - Configuración inicial examen

## 6.4. Procesamiento imagen

Tras la configuración inicial del examen, el usuario pasa a la vista de Nuevo examen, en ella tiene la posibilidad de realizar otra fotografía, rotar si fuera necesario la actual o procesarla. Esta última opción se la que se ilustre.

Al confirmar la foto tomada, el usuario debe realizar la configuración inicial del examen, indicando el número de preguntas de este. Hasta que usuario no introduce el número de preguntas, no podrá continuar.

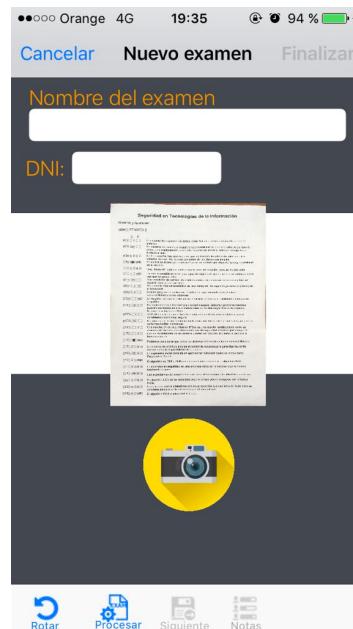


Ilustración 76 - Nuevo examen

El usuario es informado en todo momento en el porcentaje que se encuentra el proceso.



Ilustración 77 - Procesamiento imagen 1



Ilustración 78 - Procesamiento imagen 2



Ilustración 79 - Procesamiento imagen 3



Ilustración 80 - Procesamiento imagen 4

Según el porcentaje de errores obtenidos en la detección de las respuestas se le envía una alerta al usuario.



Ilustración 81 - Detección sin errores



Ilustración 82 - Detección con pocos errores



Ilustración 83 - Detección con demasiados errores

Mientras exista errores en el listado de preguntas, el usuario no podrá calcular la calificación. Para corregirlos tiene que tocar en la alerta que aparece en la barra inferior a la derecha.



Ilustración 84 - Corrección error pregunta

## 6.5. Establecer la plantilla de corrección

Ya corregidos los errores producidos en la detección de respuestas, el usuario podrá establecer una plantilla de corrección para el examen.



Ilustración 85 - Botón establecer plantilla activo



Ilustración 86 - Alerta plantilla establecida con éxito

## 6.6. Calcular calificación

Al completar la lista de respuesta, si la plantilla de corrección ya se encuentra establecida, el botón para calcular la nota se activa.



Ilustración 87 - Botón calcular nota activo

Cuando la nota está calculada, se muestra a la derecha del dni del alumno y se activa la opción para guardar y pasar al siguiente alumno en la barra inferior.



Ilustración 88 - Nota calculada y botón guardar calificación activo

## 6.7. Almacenar calificación

Para almacenar la nota de un alumno es obligatorio un dni para su identificación.

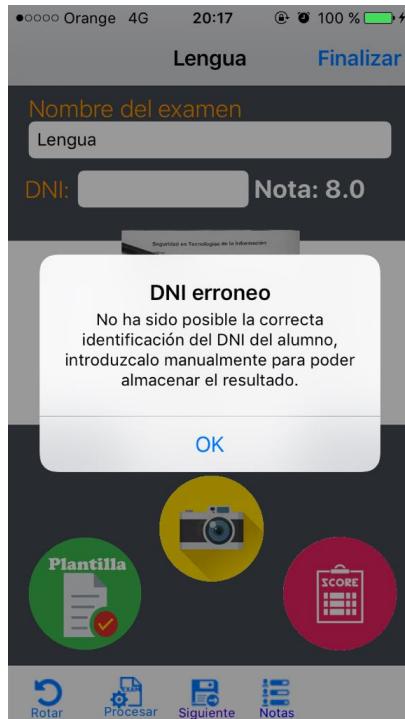


Ilustración 89 - DNI incorrecto

## 6.8. Consultar calificaciones

Al almacenar la primera la calificación de un alumno se activará la opción de listarlas. Al acceder, podemos ver el listado donde podrá seleccionarse alguna de ellas para eliminarla o exportarlos a csv y enviarlos con la opción del menú superior.



Ilustración 90 - Listado calificaciones y opción borrado

## 6.9. Exportar calificaciones

Una vez que el docente acaba de corregir los exámenes de todos los alumnos. Este, podrá enviar los resultados a su ordenador para la publicación o enviárselo directamente a los alumnos.

A continuación, se muestra las opciones que el usuario tendrá disponibles para el envío del fichero. Como opciones principales, aparecen Message, Mail, WhatsApp, Notas y Outlook.

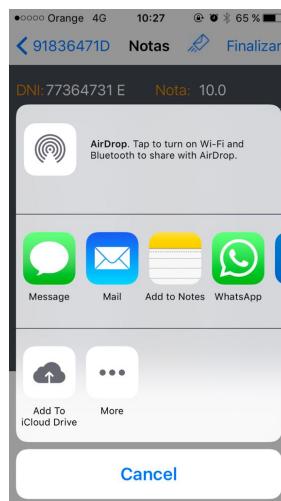


Ilustración 91 - Opciones envío calificaciones

Al seleccionar Outlook para el envío de las calificaciones, se abre automáticamente la creación de un nuevo correo con el fichero csv generado como adjunto. Solo se tiene que introducir el destinatario y enviar.

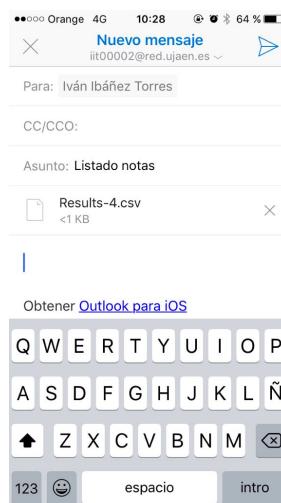


Ilustración 92 - Envío calificaciones Outlook

★ Ivan Ibáñez Torres ⓘ

Notas

Para: Ivan Ibanez Torres

Obtener [Outlook para iOS](#)



Results-5.csv

Ilustración 93 - Mail recibido con calificaciones adjuntas

Para el resultado final, abrimos el documento adjunto recibido en el correo. Una vez en Excel, el listado es completamente editable por el usuario, pudiendo organizarlo de la manera que desee.

	A	B	C
1	DNI	Nota	
2	77364731 E	10.0	
3	72801711A	8.0	
4	85748619A	4.0	
5	83254465Z	5.0	
6	765656973D	5.0	
7	26143835H	4.0	
8	91836471D	3.0	
9			
10			
11			

Ilustración 94 - Listado calificaciones Excel

## 7. Conclusiones y posibles mejoras

Una vez finalizado el desarrollo de la aplicación y la documentación, ha llegado el momento de sacar conclusiones acerca del proyecto y la manera en la que se ha desarrollado.

Lo primero a destacar es la amplia variedad de ventajas que ofrece Swift, he encontrado muchas librerías que han sido de gran ayuda a la hora de implementar una solución al problema.

Desde mi punto de vista, Swift es un lenguaje de programación muy potente, sencillo e intuitivo. Al comenzar este proyecto, nunca había trabajado con él antes. Realizando el tutorial para principiantes oficial de Apple y gratuito, he conseguido desarrollar la aplicación, aunque un poco ajustado, en el tiempo establecido.

Si nos centramos en la metodología de desarrollo software utilizada, he quedado muy satisfecho. Me ha ayudado a hacer una planificación de todas las tareas incluidas en el proyecto con la que poder organizarse mucho mejor el tiempo e ir viendo avances notables en este.

El problema que intenta resolver este TFG se caracteriza por una necesidad de un gran número de pruebas. Estas pruebas, requieren de mucho tiempo para poder detectar errores y posibles mejoras. Debido a la gran inversión de tiempo en este aspecto, surge mi única contra referente a este TFG, creo que el tiempo para su realización es un poco ajustado y que con algo más de tiempo se podría llegar a realizar un software mucho más avanzado y competo.

Por encima de todo, tengo que destacar la gran satisfacción producida al enfrentarme a una nueva tecnología totalmente desconocida para mí, conseguir utilizarla, comprender como funciona y poder aplicarlo para desarrollar e implementar este proyecto.

Para la realización de este TFG hubo que realizar una investigación y recogida de información previa, acerca del problema. He hablado con distintos docentes interesados en la idea de una App móvil de este tipo y comentaban, que algo muy

útil sería la posibilidad de corregir también, exámenes de respuesta múltiple. Esta funcionalidad aportaría mayor versatilidad a la aplicación.

Para llevar a cabo la implementación de esta funcionalidad, sería necesaria la modificación de la plantilla del examen, añadiendo las nuevas opciones a la derecha de las ya presentes.

El método que analiza el texto reconocido por el motor OCR, tendrá que ser modificado para identificar el número de pregunta y su nuevo formato de cuatro opciones de respuesta.

El proceso de clasificación de preguntas también se tendría que modificar. Hay que actualizar el proceso de análisis de la cadena, para que al igual que en el caso anterior, identifique correctamente el nuevo formato de respuestas.

El reconocimiento manuscrito del DNI del alumno es una idea que se ha intentado incluir en este TFG, pero que por falta de tiempo y varias pruebas realizadas sin buenos resultados se tuvo que abandonar.

También se podría incluir como posible mejora. Ayudaría al docente en la impresión del examen, al no tener que combinarlo con el listado de alumnos o introducirlo manualmente en cada corrección de la App.

## Índice de Ilustraciones

ILUSTRACIÓN 1 – ALUMNOS EXAMINÁNDOSE .....	6
ILUSTRACIÓN 2 - EXAMEN MULTIRESPUESTA .....	7
ILUSTRACIÓN 3 - EXAMEN RESPUESTA SIMPLE .....	8
ILUSTRACIÓN 4 - HOJA RESPUESTAS SIMPLES .....	9
ILUSTRACIÓN 5 - INNOCAN .....	10
ILUSTRACIÓN 6 - EJEMPLO EXAMEN GEXCAT .....	11
ILUSTRACIÓN 7 - EXAMIO .....	11
ILUSTRACIÓN 8 - PLANTILLA EJEMPLO EXAMIO .....	12
ILUSTRACIÓN 9 - HOJA RESPUESTAS GEXCAT .....	12
ILUSTRACIÓN 10 - DIAGRAMA GANTT .....	18
ILUSTRACIÓN 11 - DIAGRAMA DE PERT .....	19
ILUSTRACIÓN 12 - MICROBLINK .....	25
ILUSTRACIÓN 13 - ANYLINE .....	26
ILUSTRACIÓN 14 - SWIFTOCR .....	26
ILUSTRACIÓN 15 - ABBYY .....	26
ILUSTRACIÓN 16 - SCANBOT .....	27
ILUSTRACIÓN 17 - ORBIT SDK .....	27
ILUSTRACIÓN 18 - TESSERACT OCR .....	28
ILUSTRACIÓN 19 - GPUIMAGE2 .....	29
ILUSTRACIÓN 20 - TOUCAN .....	29
ILUSTRACIÓN 21 - IMAGEMAGICK .....	30
ILUSTRACIÓN 22 - OPENCV .....	30
ILUSTRACIÓN 23 - DIAGRAMA DE CLASES .....	32
ILUSTRACIÓN 24 - DIAGRAMA DE CASOS DE USO .....	34
ILUSTRACIÓN 25 - DIAGRAMA DE ACTIVIDAD .....	35
ILUSTRACIÓN 26 - PATRÓN MVC .....	36
ILUSTRACIÓN 27 - DIAGRAMA SECUENCIA NUEVO EXAMEN .....	37
ILUSTRACIÓN 28 - DIAGRAMA SECUENCIA ELIMINAR EXAMEN .....	38
ILUSTRACIÓN 29 - DIAGRAMA SECUENCIA EDITAR EXAMEN .....	39
ILUSTRACIÓN 30 - DIAGRAMA SECUENCIA CORREGIR EXAMEN .....	40
ILUSTRACIÓN 31 - DIAGRAMA SECUENCIA ELIMINAR CALIFICACIÓN .....	42
ILUSTRACIÓN 32 - DIAGRAMA SECUENCIA EXPORTAR CALIFICACIONES .....	43
ILUSTRACIÓN 33 - STORYBOARD PRINCIPAL .....	44
ILUSTRACIÓN 34 - STORYBOARD CÁMARA .....	45
ILUSTRACIÓN 35 - STORYBOARD CAPTURA FOTO .....	46
ILUSTRACIÓN 36 - STORYBOARD CONFIGURACIÓN INICIAL NUEVO EXAMEN .....	46
ILUSTRACIÓN 37 - STORYBOARD VISTA NUEVO EXAMEN .....	47
ILUSTRACIÓN 38 - STORYBOARD ACTUALIZACIONES VISTA NUEVO EXAMEN .....	48
ILUSTRACIÓN 39 - STORYBOARD LISTADO EXÁMENES .....	49
ILUSTRACIÓN 40 - STORYBOARD ACTUALIZACIONES VISTA LISTADO EXAMENES .....	50
ILUSTRACIÓN 41 - STORYBOARD LISTADO CALIFICACIONES .....	51
ILUSTRACIÓN 42 - XCode .....	52
ILUSTRACIÓN 43 - LENGUAJE C .....	53
ILUSTRACIÓN 44 - C++ .....	53
ILUSTRACIÓN 45 - OBJECTIVE C .....	53
ILUSTRACIÓN 46 - SWIFT .....	54
ILUSTRACIÓN 47 - PLANTILLA PRUEBAS CON CORCHETES A LA DERECHA .....	55
ILUSTRACIÓN 48 - PLANTILLA PRUEBAS CUADRANTE RESPUESTAS INFERIOR .....	56
ILUSTRACIÓN 49 - PLANTILLA PRUEBA CORCHETES IZQUIERDA .....	56
ILUSTRACIÓN 50 - PLANTILLA EXAMEN CON BORDES .....	57
ILUSTRACIÓN 51 - PLANTILLA EXAMEN FINAL .....	58
ILUSTRACIÓN 52 - FORMATO RESPUESTA ALUMNO .....	59
ILUSTRACIÓN 53 - REESCALADO HD A 4K .....	60
ILUSTRACIÓN 54 - IMAGEN BLANCO/NEGRO .....	60
ILUSTRACIÓN 55 - IMAGEN CON RUIDO DIGITAL .....	61
ILUSTRACIÓN 56 - RECTIFICACIÓN ÁNGULO ROTACIÓN TEXTO .....	61
ILUSTRACIÓN 57 - DETECCIÓN RECTÁNGULO Y CAMBIO PERSPECTIVA .....	62
ILUSTRACIÓN 58 - DETECCIÓN RECTÁNGULO .....	63

ILUSTRACIÓN 59 - IDENTIFICACIÓN CUATRO PUNTOS .....	64
ILUSTRACIÓN 60 - CORRECCIÓN PERSPECTIVA .....	64
ILUSTRACIÓN 61 - INDICADOR DETECCIÓN RECTÁNGULA EN CÁMARA .....	65
ILUSTRACIÓN 62 - FUNCIÓN PROCESSIMAGE() .....	65
ILUSTRACIÓN 63 - FUNCIÓN IMAGEN ESCALA GRISES .....	66
ILUSTRACIÓN 64 - FUNCIÓN IMAGEN ESCALA DE GRISES CON LIMPIEZA DE RUIDO .....	66
ILUSTRACIÓN 65 - MÉTODO PROCESAMIENTO TEXTO 1 .....	67
ILUSTRACIÓN 66 - MÉTODO PROCESAMIENTO TEXTO 2 .....	68
ILUSTRACIÓN 67 – MÉTODO CLASIFICACIÓN DE RESPUESTAS 1 .....	69
ILUSTRACIÓN 68 - MÉTODO CLASIFICACIÓN DE RESPUESTAS 2 .....	70
ILUSTRACIÓN 69 - MÉTODO EXPORTACIÓN Y ENVÍO DE RESULTADOS .....	71
ILUSTRACIÓN 70 - EXAMEN PRUEBAS 1 .....	72
ILUSTRACIÓN 71 - EXAMEN PRUEBAS 2 .....	73
ILUSTRACIÓN 72 - VISTA PRINCIPAL APP .....	74
ILUSTRACIÓN 73 - DETECCIÓN EXAMEN .....	74
ILUSTRACIÓN 74 - CORRECCIÓN PERSPECTIVA EXAMEN .....	75
ILUSTRACIÓN 75 - CONFIGURACIÓN INICIAL EXAMEN .....	76
ILUSTRACIÓN 76 - NUEVO EXAMEN .....	77
ILUSTRACIÓN 77 - PROCESAMIENTO IMAGEN 1 .....	77
ILUSTRACIÓN 78 - PROCESAMIENTO IMAGEN 2 .....	78
ILUSTRACIÓN 79 - PROCESAMIENTO IMAGEN 3 .....	78
ILUSTRACIÓN 80 - PROCESAMIENTO IMAGEN 4 .....	78
ILUSTRACIÓN 81 - DETECCIÓN SIN ERRORES .....	79
ILUSTRACIÓN 82 - DETECCIÓN CON POCOS ERRORES .....	79
ILUSTRACIÓN 83 - DETECCIÓN CON DEMASIADOS ERRORES .....	80
ILUSTRACIÓN 84 - CORRECCIÓN ERROR PREGUNTA .....	80
ILUSTRACIÓN 85 - BOTÓN ESTABLECER PLANTILLA ACTIVO .....	81
ILUSTRACIÓN 86 - ALERTA PLANTILLA ESTABLECIDA CON ÉXITO .....	81
ILUSTRACIÓN 87 - BOTÓN CALCULAR NOTA ACTIVO .....	82
ILUSTRACIÓN 88 - NOTA CALCULADA Y BOTÓN GUARDAR CALIFICACIÓN ACTIVO .....	82
ILUSTRACIÓN 89 - DNI INCORRECTO .....	83
ILUSTRACIÓN 90 - LISTADO CALIFICACIONES Y OPCIÓN BORRADO .....	83
ILUSTRACIÓN 91 - OPCIONES ENVÍO CALIFICACIONES .....	84
ILUSTRACIÓN 92 - ENVÍO CALIFICACIONES OUTLOOK .....	84
ILUSTRACIÓN 93 - MAIL RECIBIDO CON CALIFICACIONES ADJUNTAS .....	85
ILUSTRACIÓN 94 - LISTADO CALIFICACIONES EXCEL .....	85

## Índice de Tablas

TABLA 1 – DIVISIÓN DE TAREAS .....	16
TABLA 2 - PLANIFICACIÓN HORAS .....	17
TABLA 3 - COSTE SOFTWARE PROYECTO .....	21
TABLA 4 - COSTE HARDWARE PROYECTO .....	22
TABLA 5 - COSTE DEL PROYECTO .....	23

## Bibliografía

- Que es Swift: <https://www.apple.com/es/swift/>
- Comenzar en el desarrollo de aplicaciones iOS (Swift):  
[https://developer.apple.com/library/content/referencelibrary/GettingStarted/DevelopiOSAppsSwift/ImplementNavigation.html#/apple\\_ref/doc/uid/TP40015214-CH16-SW1](https://developer.apple.com/library/content/referencelibrary/GettingStarted/DevelopiOSAppsSwift/ImplementNavigation.html#/apple_ref/doc/uid/TP40015214-CH16-SW1)
- “Modelo incremental”, procesossoftware.wikispaces.com, disponible en:  
<https://procesossoftware.wikispaces.com/Modelo+Incremental>
- “Técnicas para identificar requisitos funcionales y requisitos no funcionales”, sites.google.com, disponible en:  
<https://sites.google.com/site/metodologiareq/capitulo-ii/tecnicas-para-identificar-requisitos-funcionales-y-no-funcionales>
- OpenCV - Detect skew angle: <http://felix.abecassis.me/2011/09/opencv-detect-skew-angle/>
- OpenCV – Image denoising:  
[http://docs.opencv.org/trunk/d5/d69/tutorial\\_py\\_non\\_local\\_means.html](http://docs.opencv.org/trunk/d5/d69/tutorial_py_non_local_means.html)
- OpenCV – Rectangle Detection: <http://answers.opencv.org/question/94492/rectangle-detection-opencv-2412/>
- OpenCV – Geometric Image Transformations:  
[http://docs.opencv.org/2.4/modules/imgproc/doc/geometric\\_transformations.html](http://docs.opencv.org/2.4/modules/imgproc/doc/geometric_transformations.html)
- Tesseract – Improve Quality: <https://github.com/tesseract-ocr/tesseract/wiki/ImproveQuality>
- Storyboards tutorial in iOS9: <https://www.raywenderlich.com/113388/storyboards-tutorial-in-ios-9-part-1>
- ImageMagick – Noise Reduction: <http://www.imagemagick.org/discourse-server/viewtopic.php?t=13086>
- ImageMagick – Perspective transformations:  
<http://www.imagemagick.org/discourse-server/viewtopic.php?t=16173>
- Displaying Alert Dialogs:  
<https://developer.apple.com/library/content/documentation/Cocoa/Conceptual/Dialog/Tasks/UsingAlerts.html>

- OpenCV – Shape detections: <http://www.pyimagesearch.com/2016/02/08/opencv-shape-detection/>
- Background Threads: <https://thatthinginswift.com/background-threads/>
- Text line detection in handwritten documents:  
<http://www.sciencedirect.com/science/article/pii/S0031320308001775>
- OCR of Hand-written Data using kNN: [http://opencv-python-tutorials.readthedocs.io/en/latest/py\\_tutorials/py\\_ml/py\\_knn/py\\_knn\\_opencv/py\\_knn\\_opencv.html](http://opencv-python-tutorials.readthedocs.io/en/latest/py_tutorials/py_ml/py_knn/py_knn_opencv/py_knn_opencv.html)
- Digit Recognition using OpenCV, sklearn and Python:  
<http://hanzratech.in/2015/02/24/handwritten-digit-recognition-using-opencv-sklearn-and-python.html>
- Simple Digit Recognition OCR in OpenCV-Python:  
<http://opencvpython.blogspot.com.es/2012/04/simple-digit-recognition-ocr-in-opencv.html>
- SwiftOCR: <https://github.com/garnele007/SwiftOCR>
- CocoaControls: <https://www.cocoacontrols.com/controls?language=2-swift>
- IPDCameraViewController: <https://github.com/mmackh/IPDFCameraViewController>
- ABBY FineReader Engine: <https://www.abbyy.com/en-eu/ocr-sdk/>
- GexCat: <http://www.gexcat.com/>
- ExamIO: <http://www.examio.com/>
- GanttProject: <http://www.ganttproject.biz/>
- MicroBlinkOCR: <https://microblink.com/en>
- ScanBot: <https://scanbot.io/en/index.html>

- OCR Labs: <http://www.ocrlabs.com/>
- Tesseract OCR: <https://github.com/tesseract-ocr/>
- GPUImage2: <https://github.com/BradLarson/GPUImage2>
- Toucan: <https://github.com/gavinbunney/Toucan>