

ULL

Universidad
de La Laguna

Escuela Superior de
Ingeniería y Tecnología
Sección de Ingeniería Informática

Trabajo de Fin de Grado

QuasarApp

QuasarApp

Aarón José Vera Cerdeña

La Laguna, 15 de julio de 2016

Dña. **Silvia Alayón Miranda**, con N.I.F. 43.812.596-B Profesora Contratada Doctor adscrito al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como tutora.

D. **Francisco José Fumero Batista**, con N.I.F. 45.731.321-F investigador adscrito al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como cotutor.

C E R T I F I C A (N)

Que la presente memoria titulada:

"QuasarApp."

ha sido realizada bajo su dirección por D. **Aarón José Vera Cerdeña**, con N.I.F. 78.722.817-G.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna a 15 de Julio de 2016.

Agradecimientos

A toda mi familia por apoyarme siempre.

A mis amigos por aguantar mis comidas de cabeza cada vez que les enseñaba la app.

A mi tutora Silvia que desde que empecé la carrera me ayudó en todas las asignaturas que he cursado con ella y es un placer haber terminado el Grado de Ingeniería Informática siendo ella mi tutora de TFG.

A mi cotutor Fran que me ayudó cada vez que me estancaba en cualquier parte del proyecto y estar atento con cualquier duda que me surgiese.

Al Dr. Tinguaro Díaz Alemán, por haberme explicado todo el proceso de estudio del Glaucoma y felicitarlo por el programa Quasar, muy útil para el estudio de la progresión del Glaucoma.

Al prof. Drago Díaz Alemán, por el diseño del logo de la aplicación.

Licencia

* Si quiere permitir que se compartan las adaptaciones de tu obra y NO quieres permitir usos comerciales de tu obra indica:



© Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial 4.0 Internacional.

Resumen

El TFG consistirá en la adaptación de un software ya existente para el seguimiento de la evolución del glaucoma (enfermedad oftalmológica) a una aplicación móvil.

Se parte de un programa llamado Quasar, desarrollado en Python por el médico que colabora en este TFG. Este programa utiliza análisis de regresión lineal del defecto medio (MD), y de la pérdida de varianza (sLV) o desviación respecto al patrón (PSD), parámetros oftalmológicos muy utilizados en el diagnóstico del glaucoma. El programa almacena los parámetros de cada paciente e indica la progresión de la enfermedad, si detecta un cambio significativo en el MD y en la sLV o PSD.

Palabras clave: Glaucoma, Campo visual, estudio de progresión, perimetría, perímetro Octopus, perímetro Humphrey, aplicación móvil.

Abstract

The objective of this Final Undergraduate Project is the adaptation of already existing software for studying the evolution of the glaucoma (an ophthalmological disease) to a mobile application.

The doctor who collaborates in this work has developed the program, Quasar, in Python. This program applies linear regression analysis to the Mean Defect (MD), and to the Loss of Variance (sLV) or pattern standard deviation(PSD), ophthalmological parameters very used in the diagnosis of the glaucoma. The program stores the parameters of every patient and indicates the progression of the disease, if it detects a significant change in the MD and sLV parameters.

Keywords: Glaucoma, visual field, progression study, perimetry, Octopus perimeter, Humphrey perimeter, Mobile application.

Índice General

1. Introducción	III
2. Descripción del problema médico	4
2.1. ¿Qué es el glaucoma?	4
2.2. Técnicas diagnosticas del glaucoma	5
2.3. Campo visual y perimetría	5
2.4. Índices estadísticos	8
2.5. Modelos de perímetros utilizados en el Hospital Universitario de Canarias	9
3. Objetivo de este TFG	11
3.1. Objetivos proyecto	12
3.2. Objetivos de la app	12
3.3. Antecedentes tecnológicos	13
3.3.1. Funcionamiento del programa Quasar	15
3.3.2. Aplicaciones similares	16
4. Herramientas utilizadas	17
4.1. Apache Cordova	17
4.2. Intel XDK	18
4.3. IndexedDB	19
4.4. Highcharts	21
4.5. Github	22
5. Descripción de la app desarrollada: QuasarApp	23
6. Validación	32
7. Conclusiones y líneas futuras	33
7.1. Conclusiones	33
7.1. Líneas futuras	34
8. Summary and Conclusions	35
8.1. Conclusions	35
8.2. Future Lines	36
9. Presupuesto	37
Apéndice A. Código	38
A.1. Creación base de datos en IndexedDB	38
A.2. Creación paciente	39
A.3. Creación prueba paciente	40
A.4. Búsqueda de paciente	42
A.5. Eliminación paciente	43
Bibliografía	47

Índice de figuras

Figura 1. Esquema del ojo.....	4
Figura 2. <i>Puntos del campo visual que se estudian en la perimetría convencional. Los dos puntos en blanco corresponden al nervio óptico.</i>	6
Figura 3. Ejemplo de perimetría clásica.....	7
Figura 4. A la izquierda Perímetro Octopus y a la derecha Perímetro Humphrey.	9
Figura 5. Resultados en el programa Quasar.....	11
Figura 6. Programa Quasar.....	14
Figura 7. Quasar con pruebas introducidas.....	16
Figura 8. <i>A la izquierda programa TNT y a la derecha GPA.</i>	16
Figura 9. Pantalla principal proyecto en Intel XDK.....	19
Figura 10. Logo IndexedDB.....	20
Figura 11. Logo Highcharts.....	21
Figura 12. Imagen proyecto TFG en Github.....	22
Figura 13. Pantalla Home page.....	23
Figura 14. Pantalla Create Patient.....	24
Figura 15. Pantalla Patient Search.....	25
Figura 16. Pantalla Creación test.....	26
Figura 17. Mensaje error añadir prueba.....	26
Figura 18. Pantalla Main Menu visualización pruebas.....	27
Figura 19. Pantalla Main con resultados del análisis de regresión lineal.....	28
Figura 20. Pantalla Delete test.....	29
Figura 21. Pantalla FAQ.....	30
Figura 22. Gráfica generada por Highchart.....	31
Figura 23. Imagen exportar gráfica.....	31

1. Introducción

Este TFG consiste en el desarrollo de una app móvil, que se llamará QuasarApp, partiendo de un programa llamado Quasar.

El objetivo de Quasar es permitir al especialista el seguimiento de las pruebas realizadas a los pacientes con glaucoma y calcular la progresión de la misma a lo largo de un periodo determinado. Este programa ha sido desarrollado en Python por el médico que colabora con este TFG, el Dr. Tinguaro Díaz Alemán, oftalmólogo del Hospital Universitario de Canarias (HUC). El programa aplica técnicas de análisis de regresión lineal al defecto medio (MD), y a la pérdida de varianza (sLV) o patrón de desviación estándar (PSD), parámetros oftalmológicos muy utilizados en el diagnóstico del glaucoma. El programa almacena los parámetros de cada paciente e indica la progresión de la enfermedad, si detecta un cambio significativo en el MD y en la sLV o PSD.

QuasarApp será una aplicación móvil que ofrecerá a los oftalmólogos expertos en glaucoma la posibilidad de llevar a cabo el seguimiento del paciente de una manera más centralizada y eficiente.

En este TFG se realizará un estudio del programa Quasar, comprobando sus funcionalidades paso a paso, y analizando la inclusión de mejoras, en la usabilidad y accesibilidad del programa, gracias al empleo de nuevas tecnologías.

La aplicación móvil se ha realizado en colaboración con el Servicio de Oftalmología del Hospital Universitario de Canarias (HUC), con la ayuda de expertos en el Glaucoma, ya que ellos serán los usuarios finales de la app.

2. Descripción del problema médico

En este capítulo se va a realizar una breve descripción del problema médico abordado, el estudio de la evolución del glaucoma en los pacientes.

El propósito del TFG consiste en proporcionar al oftalmólogo una herramienta que le ayude a evaluar la progresión del campo visual en pacientes con glaucoma de forma fácil, rápida y eficaz.

2.1. ¿Qué es el glaucoma?

El glaucoma es una enfermedad ocular que se caracteriza por la pérdida del campo visual como consecuencia de un daño en el nervio óptico secundario, provocado generalmente, por un aumento de la presión intraocular (PIO) [1].

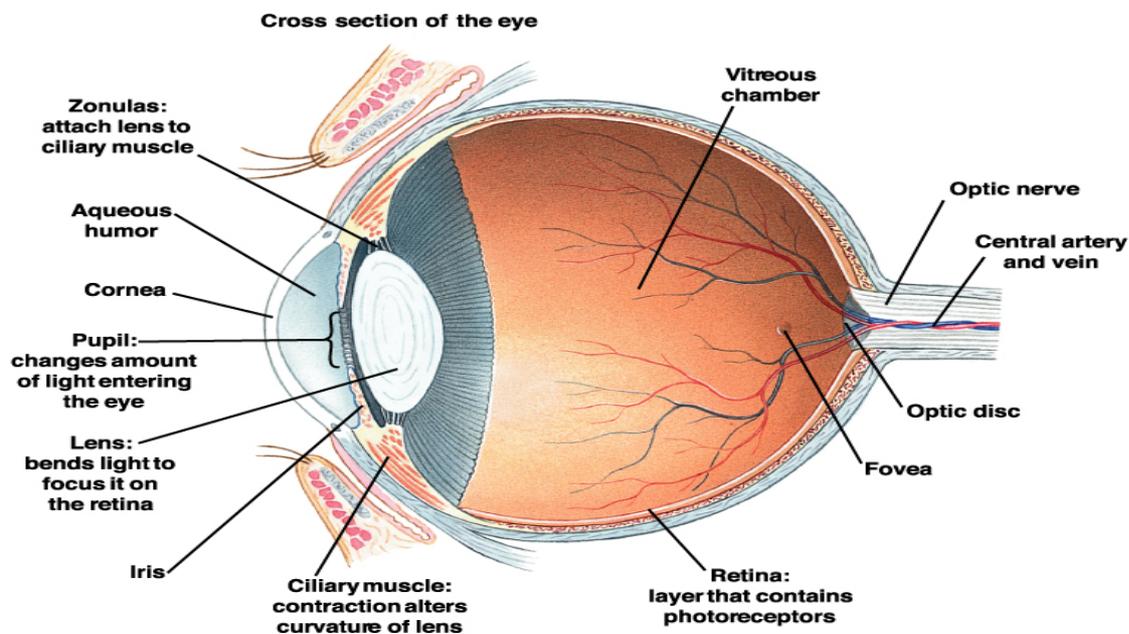


Figura 1. Esquema del ojo.

La **Figura 1** nos muestra en un corte sagital la anatomía del globo intraocular. La presión intraocular depende del equilibrio entre la producción y drenaje del humor acuoso, fluido que baña la parte más anterior del globo ocular. Las enfermedades oculares que aumenten la producción o disminuyan el drenaje producirán aumento de la PIO.

Existen diversos factores que influyen en la presión intraocular: la herencia familiar, edad, raza, variaciones posturales, movimientos oculares, parpadeo y algunos tratamientos médicos.

2.2. Técnicas diagnosticas del glaucoma

Para diagnosticar el glaucoma es necesario evaluar la presión ocular, pero este no es el único factor a tener en cuenta. Se debe estudiar también el nervio óptico, tanto desde el punto de vista anatómico como funcional.

Para la **exploración anatómica o morfológica** se precisa de personal entrenado y sofisticados sistemas de análisis computerizado de la cabeza del nervio, como por ejemplo el Tomógrafo Retinal de Heidelberg (HRT) [2], la Tomografía de Coherencia Óptica (OCT) [3] y el Escaneo Polarímetro Láser (GDx) [4].

En la **exploración funcional** es básica la realización de campos visuales con perímetros. El programa Quasar analiza los resultados de estas técnicas funcionales, por lo que las explicaremos en más detalle a continuación.

2.3. Campo visual y perimetría

El **campo visual** se puede definir como el espacio que percibe cada ojo por separado cuando mira hacia un punto fijo central. El ojo transmite al cerebro las informaciones que percibe dentro de este campo visual, teniendo en cuenta su forma, sus colores y sus volúmenes.

El campo de visión abarca aproximadamente 180 grados en un individuo cuya visión es normal[5].

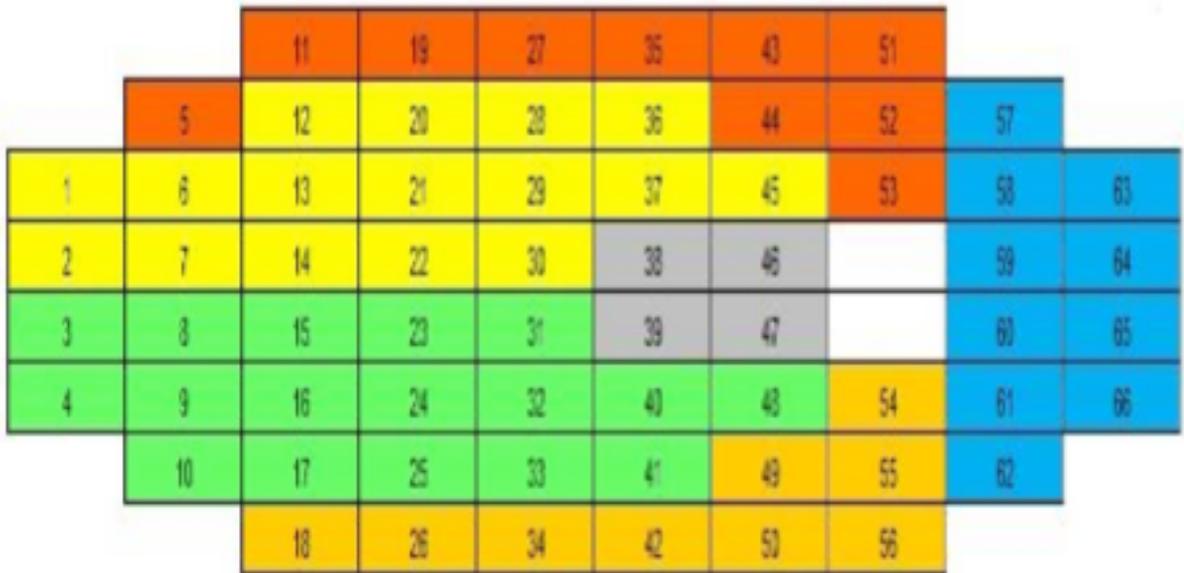


Figura 2. *Puntos del campo visual que se estudian en la perimetría convencional. Los dos puntos en blanco corresponden al nervio óptico.*

El campo visual es estudiado por el oftalmólogo de forma individual para cada ojo. Cuando el campo visual se altera puede afectarse la zona periférica o central. La pérdida localizada de una parte del campo visual se llama escotoma. Es muy típico en el glaucoma, que la pérdida de campo visual comience a nivel periférico y avance lentamente hacia el centro, de aquí la importancia de la detección precoz de la progresión.

El estudio del campo visual se lleva a cabo con la perimetría.

La **perimetría** o campimetría computarizada es un estudio asistido por un ordenador con el que se analiza y cuantifica la sensibilidad de la retina a diferentes estímulos luminosos que varían en intensidad, tamaño y ubicación [6].

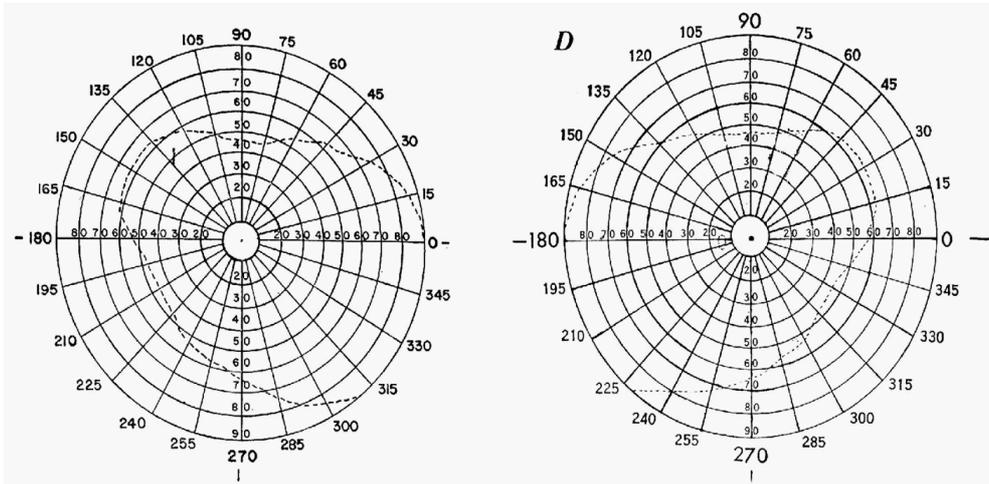


Figura 3. Ejemplo de perimetría clásica.

La perimetría se limita a estudiar la sensibilidad luminosa por medio del umbral diferencial. Se define como umbral a aquel estímulo con un 50% de probabilidad de ser visto y como umbral luminoso diferencial a la mínima intensidad luminosa que puede ser diferenciada de otra (intensidad mínima a la que un estímulo puede ser distinguido del fondo).

El umbral luminoso se mide en decibelios (dB), unidad logarítmica que es inversamente proporcional a la luminosidad del estímulo.

La forma tradicional de búsqueda del estímulo es de manera escalonada. Se presentan estímulos de intensidad igual a la esperada para un sujeto de la misma edad. Según la respuesta del sujeto a este estímulo, el siguiente será de 4 dB más o menos intenso. Una vez producida una inversión en la respuesta del paciente (visto o no visto), se procede con escalones de 2 dB o incluso de 1 dB.

El estudio del campo visual suele limitarse al campo central (30 grados) ya que es suficiente detectar y estudiar la mayoría de las patologías que lo afectan.

2.4. Índices estadísticos

La interpretación de los resultados del campo visual resulta más fácil si se emplean índices estadísticos, así como representaciones gráficas de los resultados en forma de escala de grises.

Los *índices estadísticos más importantes* son [7]:

- **La sensibilidad media (MS)**, es la media aritmética de las sensibilidades de todos los puntos estudiados.
- **El defecto medio (MD)**, es la media aritmética de las diferencias de las sensibilidades de cada punto con respecto al valor para una persona de la misma edad (permite comparaciones entre grupos de edades distintos).
- **La varianza de pérdida en Octopus (sLV) o desviación respecto al patrón en Humphery (PSD)**, es el resultado de dividir la suma de las diferencias cuadráticas entre el defecto medio y el defecto encontrado en cada punto concreto, por el número de casos menos uno. Representa la variabilidad del campo visual. A mayor variabilidad, mayor será el valor de la varianza.

Analizando estos parámetros es posible evaluar el campo visual del paciente, detectar alteraciones precoces y diagnosticar enfermedades de la retina, el nervio óptico y la vía visual.

2.5. Modelos de perímetros utilizados en el Hospital Universitario de Canarias

En el HUC existen perímetros de dos fabricantes diferentes: Octopus [8] y Humphrey [9] se pueden ver en la **figura 4**.

Hay que tener en cuenta que la iluminación en el **Octopus** es 9 veces menos intensa que en el **Humphrey**, y las necesidades de una mayor intensidad luminosa también son mucho menores. Al existir una menor iluminación de fondo los estímulos se percibirán más intensos.

La duración del estímulo deberá ser más corta que el tiempo de latencia de los movimientos oculares, que es de unos 0,25 segundos, el paciente no tendrá tiempo de desviar la mirada para intentar localizar el estímulo. Por ello la duración en el Perímetro de **Humphrey** es de 0,2 segundos y en el de **Octopus** es de 0,1 segundos.



Figura 4. A la izquierda Perímetro Octopus y a la derecha Perímetro Humphrey.

2.6. Técnicas diagnósticas de progresión funcional

Existen diversas técnicas para el control de la evolución del glaucoma, como por ejemplo las estrategias Threshold Noiseless Trend (TNT) [10] y Glaucoma Progression Analysis (GPA) [11], pero en este apartado vamos a describir el programa Quasar, ya que es el objeto de estudio de este TFG.

Quasar [12] es un programa diseñado para evaluar la progresión del campo visual en los pacientes con glaucoma, utilizando el análisis de regresión lineal tanto de MD y sLV o PSD, bajo la hipótesis de que la combinación de estos dos parámetros debe mejorar la sensibilidad de cada uno por separado.

El programa de Quasar fue desarrollado por un grupo de investigación del Servicio de Oftalmología del Hospital de Universitario de Canarias. Fue programado en el lenguaje de programación Python 2.7.3 [13] y en el entorno Linux Fedora 17 por uno de los médicos de dicho grupo.

Está basado en el análisis de regresión lineal de MD y sLV o PSD. Los valores de PSD o sLV fueron calculados de nuevo para transformarlos en una distribución lineal, usando una modificación de la fórmula siguiente para perímetros de Octopus:

$$\text{Si, } MD > 16.33; \text{ sLV} = \text{sLV} + ([MD-16.33]/0.84)$$

Esta fórmula tuvo que ser modificada para los perímetros Humphrey:

$$\text{Si } MD < -17.35; \text{ PSD} = \text{PSD} - ([MD+17.35]/0.84)$$

El programa Quasar indica si hay progresión de la enfermedad si detecta un cambio significativo en el MD o en el sLV. También calcula la pérdida de MD por año, y predice la pérdida visual por años si la tendencia no es corregida con tratamiento, como se puede observar en la **Figura 5**. En la actualidad existen dos versiones del programa, una para manejar datos del perímetro Octopus y otra para el de Humphrey.

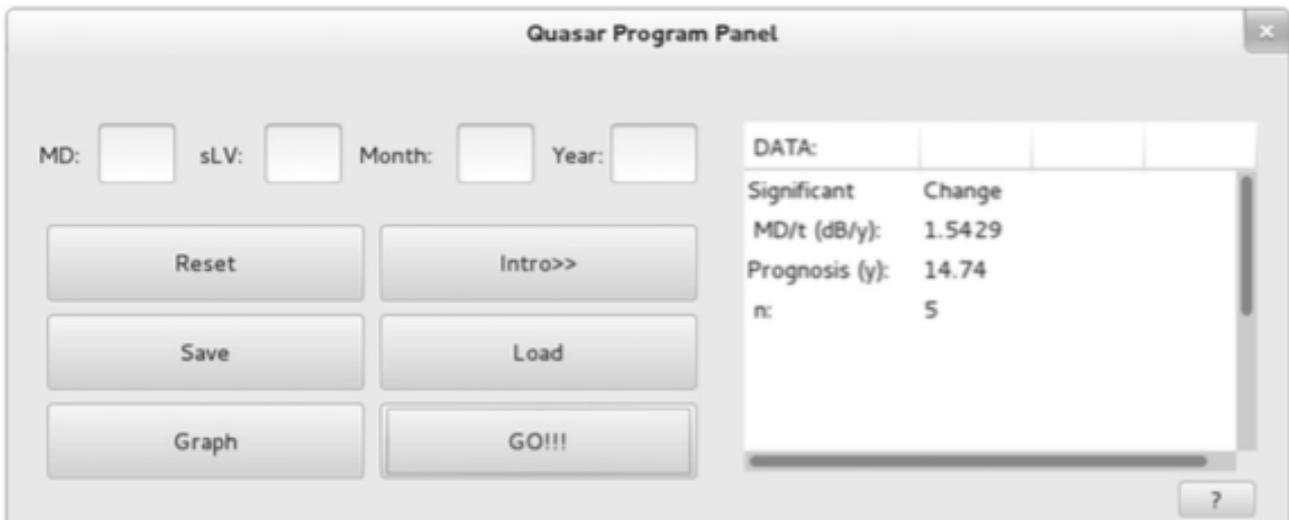


Figura 5. Resultados en el programa Quasar.

3. Objetivo de este TFG

En este proyecto se pretende crear una aplicación para dispositivos móviles partiendo del programa Quasar ya creado previamente para PC.

También se pretende adquirir las competencias relacionadas con la Ingeniería del Software y la programación web utilizando las herramientas que existen en el mercado para ello.

3.1. Objetivos proyecto

Se proponen una serie de objetivos específicos propuestos para este proyecto que serían los siguientes:

- Aplicar los diferentes conocimientos adquiridos en el Grado de Ingeniería Informática de la Universidad de La Laguna.
- Aprendizaje de conocimientos sobre el desarrollo de aplicaciones móviles para múltiples plataformas.
- Planificación del diseño y desarrollo de un proyecto.
- Crear una memoria sobre la aplicación desarrollada en el presente proyecto.

3.2. Objetivos de la app

El objetivo principal de la aplicación móvil es ofrecer al especialista una mayor usabilidad y accesibilidad en el tratamiento de los datos correspondientes a las pruebas realizadas a los pacientes.

Las ventajas que ofrecerá esta aplicación serán las siguientes:

- Una mayor accesibilidad para el especialista a la hora de acceder a las pruebas realizadas a cada paciente.
- La unificación de ambos programas en una sola aplicación ofrece la posibilidad de crear pacientes tanto de tipo "Octopus" como tipo "Humphrey".
- Una mayor usabilidad ya que se podrá usar desde cualquier dispositivo móvil o Tablet sin necesidad de programas adicionales.

- Mostrará la gráfica del análisis de regresión lineal para los estudios de pacientes con Octopus y con Humphrey, ya que será capaz de calcular rangos positivos como negativos sin ninguna dificultad.
- Otra importante ventaja es el almacenamiento de manera segura de los datos personales del paciente, ya que se ha utilizado el algoritmo de encriptación MD5 para cifrar el número de historia del paciente y así respetar la intimidad de los pacientes (cumpliendo así la Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal).
- La app móvil contará con una página de ayuda que explique paso a paso la correcta utilización de la misma.
- La app móvil contará con una página de ayuda que explique paso a paso la correcta utilización de la misma.
- En el diseño de la interface se utilizarán efectos visuales gracias a los diferentes estilos.

3.3. Antecedentes tecnológicos

A día de hoy, con la rápida evolución de las nuevas tecnologías, se han vuelto muy populares las apps en los dispositivos móviles.

En el ámbito médico se están empezando a desarrollar aplicaciones específicas debido a que ofrecen un gran número de ventajas al profesional:

- Acceso a información concreta de forma rápida en consulta facilitando la toma de decisiones.
- Herramientas para crear una nueva relación profesional sanitario-paciente basada en la toma de decisiones compartidas.

- Posibilidad de disponer de herramientas fuera de la consulta o del centro médico (domicilio del paciente, consultorios rurales o servicios de urgencias móviles).
- Posibilidad de comunicación: sanitario-paciente o sanitario-sanitario.

En este proyecto se parte de un programa llamado **Quasar** [12] que se utiliza para almacenar los valores de los parámetros perimétricos de las distintas pruebas realizadas a un paciente, para su posterior comparación y estudio de la progresión de la enfermedad.

Como se ha mencionado anteriormente, el programa está desarrollado en Python [13] y es interpretado, por ello sólo se puede ejecutar en un ordenador que tenga instalado Python y las librerías necesarias para su uso.

Además, existen dos versiones de Quasar, una para cada tipo de perimetría. En la **Figura 6** se muestra la versión correspondiente al perímetro Octopus.

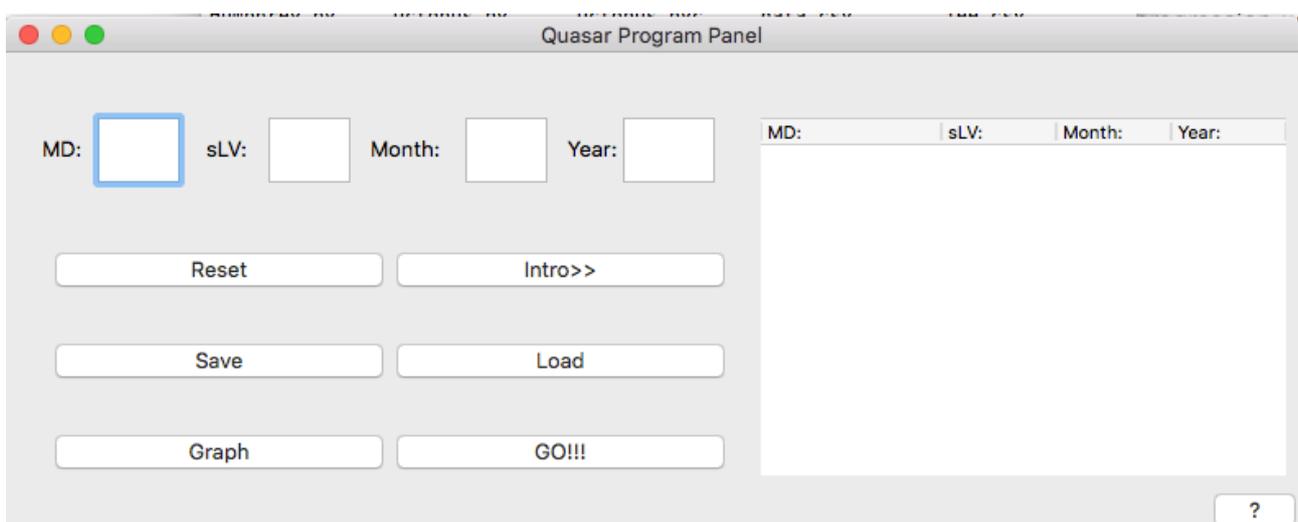


Figura 6. Programa Quasar

A continuación, se describirá más detalladamente el funcionamiento del programa Quásar y se analizarán aplicaciones similares existentes.

3.3.1. Funcionamiento del programa Quasar

En la pantalla principal(figura 7) aparecen todas las funcionalidades del programa:

- En la parte superior izquierda está un formulario donde se introducen los resultados después de haberse realizado la prueba al paciente. Se deberá introducir el valor MD (defecto medio), el valor sLv (pérdida de varianza), el mes y el año. Posteriormente, pulsando **Intro>>**, se introducen los valores de la prueba en el programa.
- La parte derecha de la pantalla muestra las pruebas introducidas. Se pueden introducir tantas pruebas como se desee.
- El botón **Reset** borra la última prueba introducida.
- El botón **Save** tiene la función de guardar todas las pruebas que se hayan introducido por pantalla en un archivo con extensión .csv.
- Si se quiere cargar algún fichero con pruebas creadas anteriormente, se puede utilizar el botón **Load**, que permite seleccionar un fichero .csv y cargar las pruebas por pantalla.
- El botón **GO!!!** tiene la función de analizar todas las pruebas que se le hayan realizado al paciente y especificar si hay cambios significativo o no
- Al pulsar el botón **Graph** se generará una gráfica donde se muestran los valores MD y sLV en función del tiempo correspondiente a las pruebas.

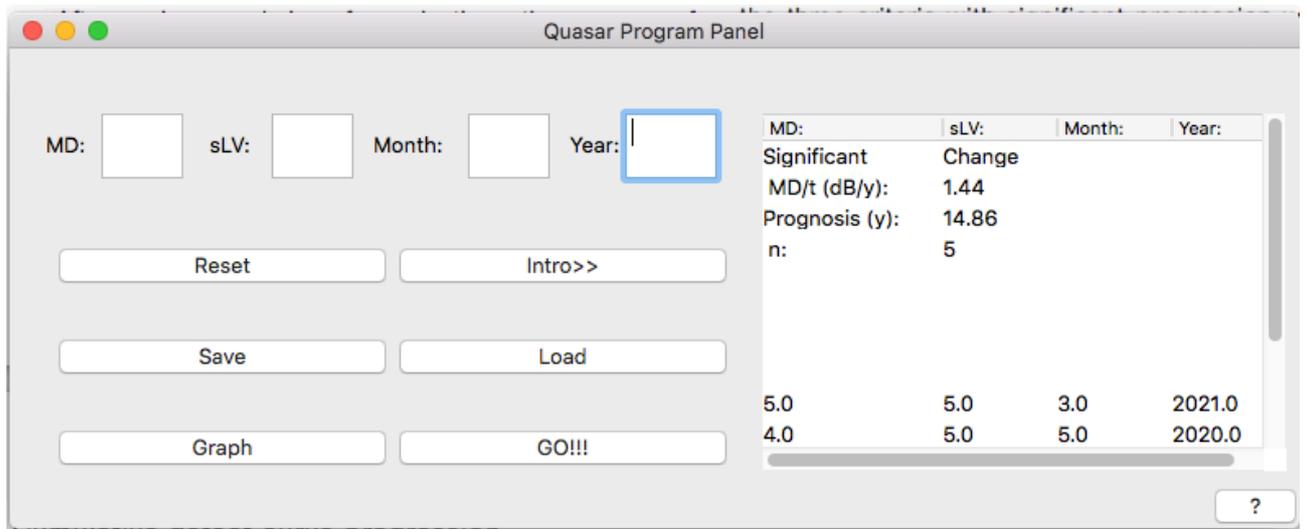


Figura 7. Quasar con pruebas introducidas

3.3.2. Aplicaciones similares

Se ha realizado una búsqueda exhaustiva de aplicaciones similares, encontrándose algún programa para PC para el estudio de la progresión del Glaucoma, como los anteriormente citados en la sección 2.6, TNT [10] y GPA [11].

Pero, a día de hoy, en el mercado no existe ninguna aplicación móvil que realice el análisis de la progresión del glaucoma, por lo que el desarrollo de esta app es muy novedoso.

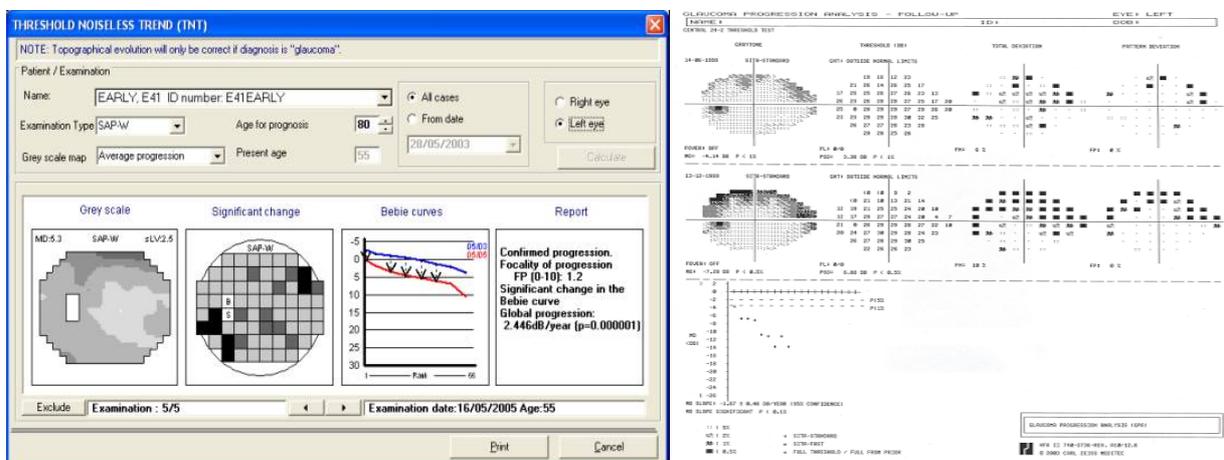


Figura 8. A la izquierda programa TNT y a la derecha GPA.

4. Herramientas utilizadas

Se ha hecho un análisis de las herramientas que existen en el mercado sobre el desarrollo de aplicaciones móviles, tanto comerciales como de software libre.

En un primer momento se eligió la herramienta PhoneGap Build [14] junto al Framework PhoneGap, ya que ofrecía una serie de ventajas para el diseño y desarrollo de la aplicación móvil, como por ejemplo la posibilidad de compilar a varias plataformas móviles, pero se acabó descartando por ser una aplicación comercial.

Después de buscar mucha información y probar varias herramientas de software libre tales como Monaca [15], Titanium [16], etc. se optó por la utilización de Intel XDK, que utiliza el Framework Apache Cordova, que es similar a PhoneGap ya que vienen del mismo fabricante.

4.1. Apache Cordova

Apache Cordova [17] es un framework de software libre que cuenta con muchas apis de diversos dispositivos móviles para desarrollar aplicaciones nativas dentro de un smartphone. Cada vez es más popular ya que para el desarrollo de las aplicaciones se utilizan las tecnologías web HTML5 [18], CSS [19] y JavaScript [20].

Una de las grandes peculiaridades de este entorno de trabajo es la posibilidad de desarrollar para iOS, Android y demás sistemas operativos sin la necesidad de programar en sus lenguajes nativos (Java, Objective-C, etc.)

4.2. Intel XDK

Intel XDK [21] es una herramienta para desarrollar apps cross-plataform utilizando HTML5.

Con XDK, los desarrolladores pueden programar usando tecnologías estándar como HTML5 y desde una misma base de código generar apps para distintas plataformas, en concreto para las siguientes:

- Aplicaciones móviles:
 - iOS
 - Android (nativo, Cordova, Crosswalk), Windows 8 Store,
 - Windows Phone 8, Tizen, y Nook.

- Aplicaciones web:
 - Web.
 - Chrome App.
 - Facebook App.

El software Intel XDK cuenta con un entorno de desarrollo que permite emular apps en dispositivos virtuales para analizar cómo se verá la app en distintos dispositivos (iPhone, Microsoft Surface, Google Nexus, entre otros). Intel XDK ofrece la capacidad de que los desarrolladores puedan almacenar su código en la nube de manera gratuita.

Intel XDK es gratuito y se puede utilizar tanto en Windows como Mac y Linux.

En la **figura 9** se muestra el proyecto creado con la herramienta Intel XDK para el desarrollo de la app objeto de este TFG.

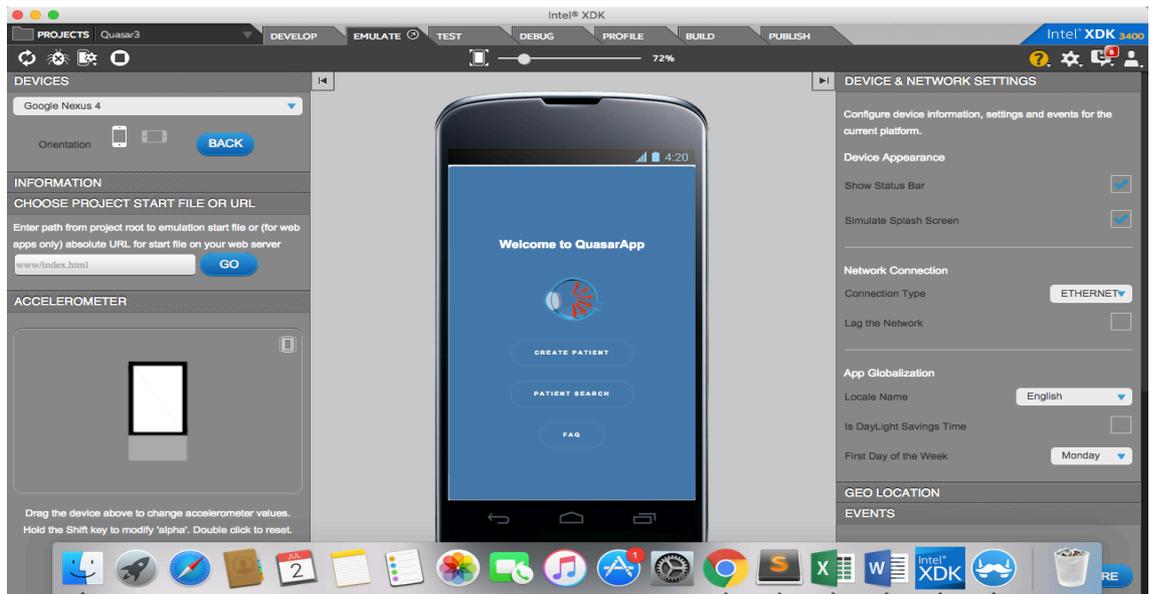


Figura 9. Pantalla principal proyecto en Intel XDK.

4.3. IndexedDB

IndexedDB[22] es un tipo de base de datos NoSQL no relacional.

Se puede definir IndexedDB como una herramienta que HTML5 proporciona a los programadores para poder trabajar con una base de datos desde la propia aplicación web, sin tener que conectarse a ningún servidor, disponible offline (en entornos sin conexión web) y persistente (al cerrar la aplicación, el contenido no se borra).

Las ventajas que ofrece IndexedDB son [23]:

- **Es un sistema de almacenamiento local:** Esta característica es muy útil. Gracias a que la información almacenada en IndexedDB está en el propio equipo, nuestra aplicación puede seguir funcionando.
- **No tiene límite de tamaño:** IndexedDB es un sistema de almacenamiento limitado tan sólo por el disco duro.

- **Permite múltiples "tablas":** IndexedDB no funciona con tablas, pero permite tener en una misma base de datos varias tablas para almacenar contenidos.
- **Permite múltiples bases de datos:** Se puede conectar a varias bases de datos vía IndexedDB.



Figura 10.Logo IndexedDB

En nuestro caso, después de crear la base de datos, hemos utilizado dos almacenes de objetos:

- "pacientes", donde se guarda el número de historia del paciente y el tipo de prueba que se le ha realizado.
- "pruebas", donde se guardan los datos de los resultados de las pruebas que se le han realizado al paciente.

4.4. Highcharts

Highcharts es un Frameworks para construir gráficas con JavaScript [24]. Es uno de los frameworks más conocidos el sector.

Entre sus características principales destacan[25]:

- **Javascript:** Esta creado con JavaScript que es uno de los lenguajes más utilizados por parte de los desarrolladores.
- **Compatibilidad:** La compatibilidades máxima, ya que soporta incluso Explorer 6, adecuándose a todos los navegadores.
- **Tipos de Gráficas:** Otra de las cosas que más destaca de HighChart es el amplio soporte de gráficas de todo tipo que ofrece.
- **Facilidad de uso:** Por último destacar es que un framework muy sencillo de utilizar.



Figura 11. Logo Highcharts.

4.5. Github

GitHub [26] es una plataforma de desarrollo colaborativo de software para alojar proyectos utilizando el sistema de control de versiones Git.

GitHub aloja tu repositorio de código y te brinda herramientas muy útiles para el trabajo en equipo, dentro de un proyecto.

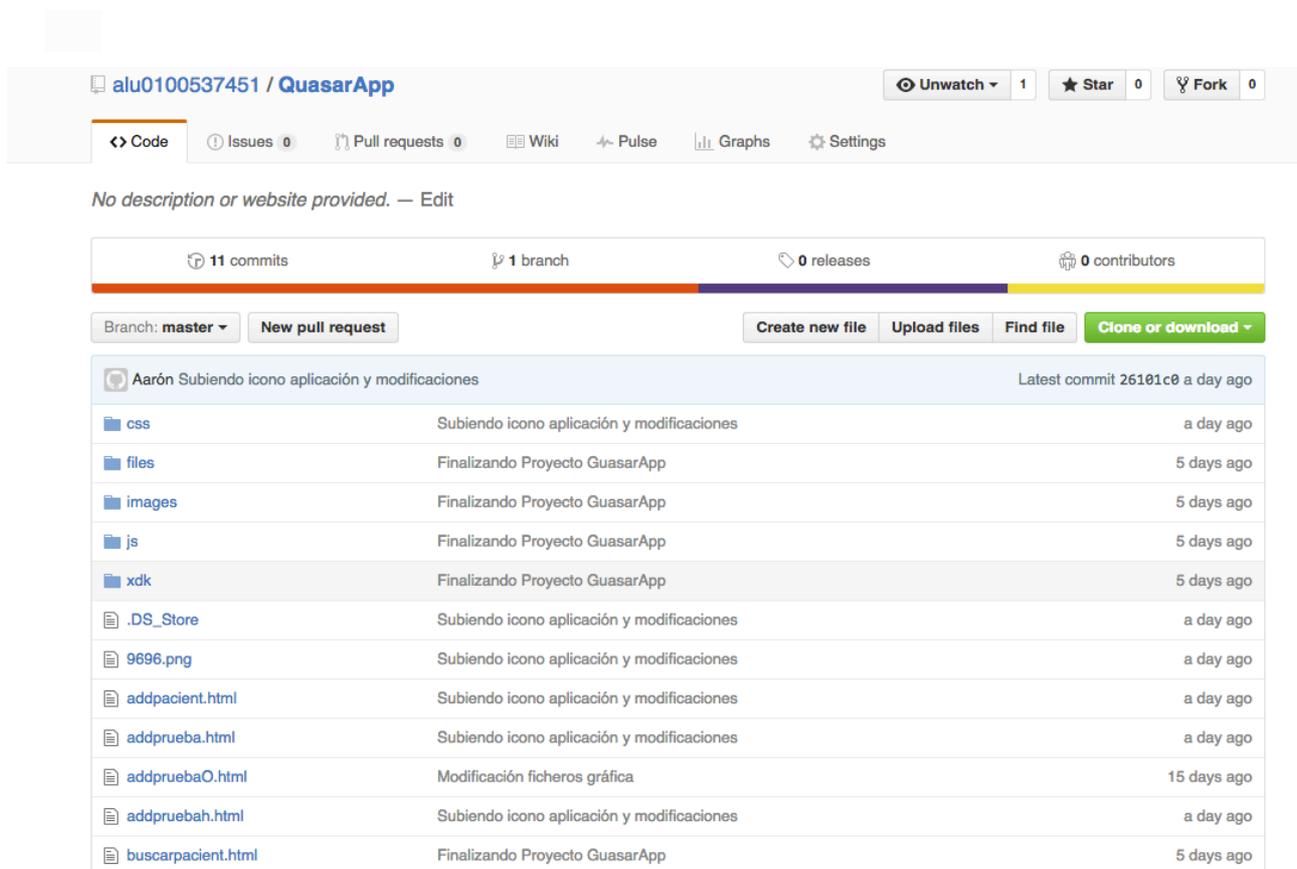


Figura 12. Imagen proyecto TFG en Github.

5. Descripción de la app desarrollada: QuasarApp

A continuación se describen las **funcionalidad básicas** que ofrece la aplicación:

- **Página Home:**

La página de inicio muestra tres botones: Create Patient, Patient Search y FAQ, que se explicarán en los siguientes puntos detalladamente.

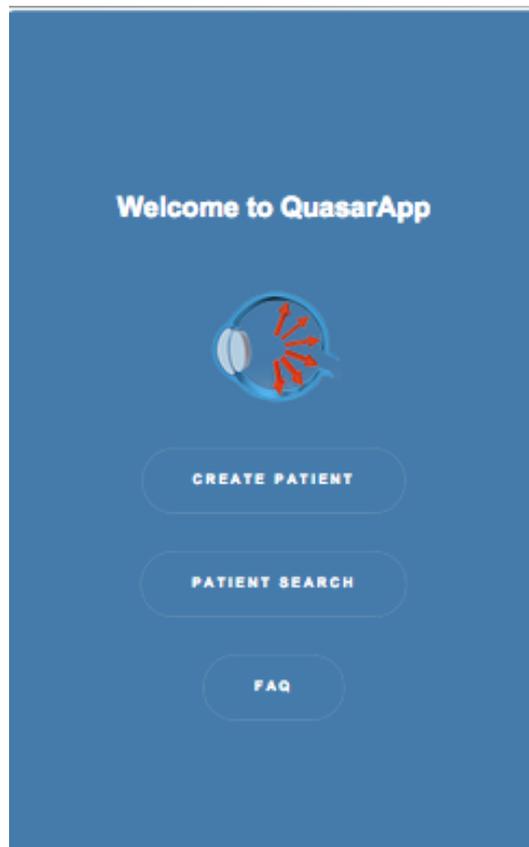


Figura 13. Pantalla Home page

- **Crear paciente:** cuando se invoca por medio de un botón aparece en la pantalla un formulario que solicita un número de historia y un tipo de prueba.

El número de historia es el número de historia del paciente en cuestión y el tipo de prueba sería la prueba que se le debe realizar al paciente en función de sus necesidades, que pueden ser de tipo "Octopus" o "Humphrey".

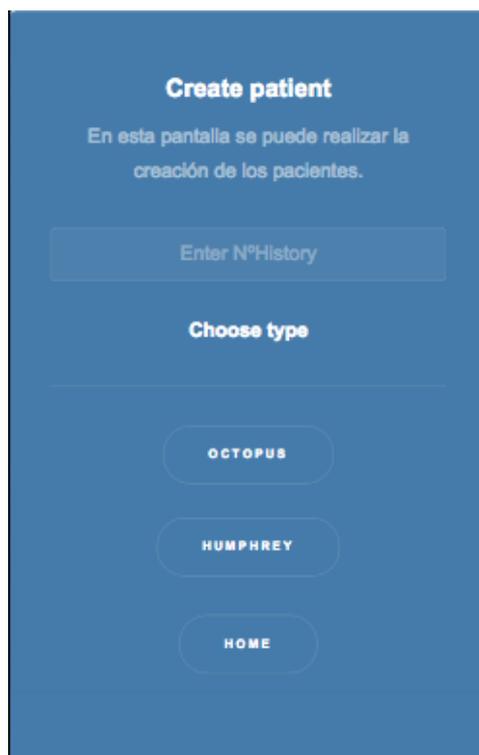


Figura 14. Pantalla Create Patient.

- **Búsqueda y eliminación de paciente:** esta funcionalidad sirve para buscar y eliminar pacientes de la base de datos.

Si el paciente está creado se mostrará un mensaje de confirmación y se redirigirá a la página Main Menu.

La funcionalidad de borrar un paciente ofrece la posibilidad de borrarlo introduciendo su número de historia, y antes de borrarlo se mostrará un mensaje de confirmación para comprobar que el usuario está seguro de la acción llevada a cabo.

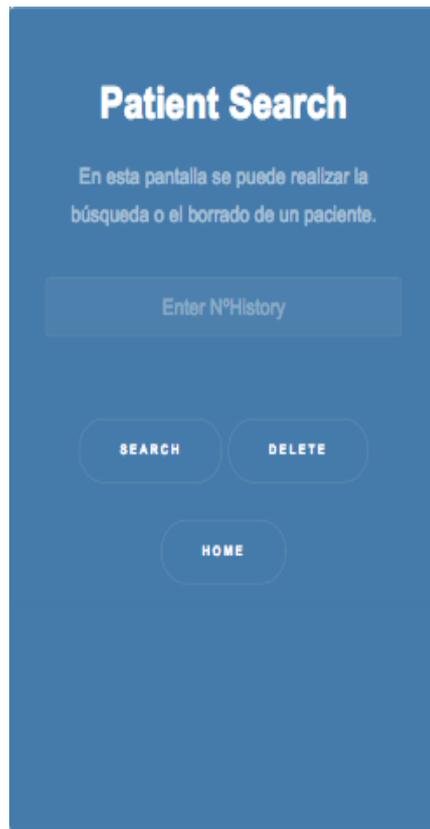


Figura 15.Pantalla Patient Search

- **Crear prueba:** esta funcionalidad se puede utilizar desde el menú principal, cuando previamente se haya utilizado la función de búsqueda de paciente.

Al pulsar Crear prueba se redirige al usuario a otra página que muestra un formulario con 4 campos para introducir: el defecto medio(MD), la perdida de varianza(sLV) o patrón de desviación estándar(PSD), el mes y el año en el que se ha realizado la prueba.

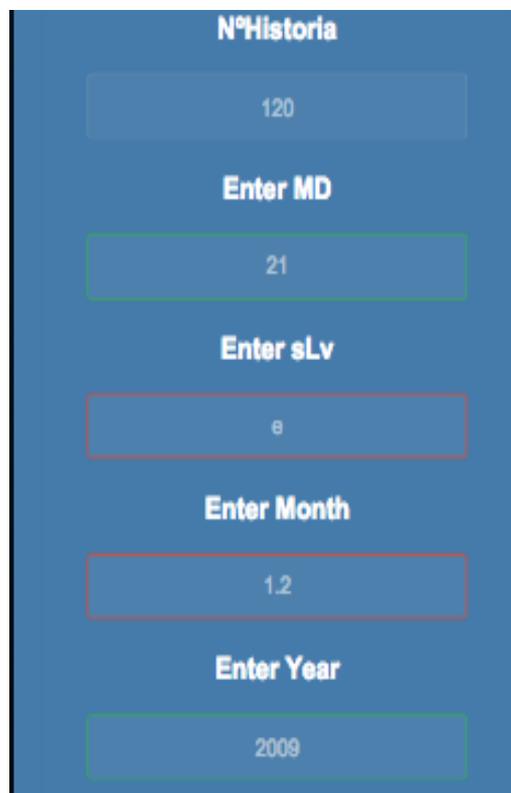
Internamente se considera el tipo de prueba que se le realiza al paciente. Si es de tipo "Octopus" se hace una comprobación de si el defecto medio es menor de 16.33 y se modifica la perdida de varianza utilizando esta fórmula:

$$sLV = sLV + ([MD-16.33]/0.84)$$

En el caso de "Humprey" se hace una comprobación de si el MD es menor de 17.35 y se aplicaría la siguiente fórmula:

$$PSD = PSD - ([MD+17.35]/0.84)$$

Además se ha implementado un control de errores utilizando una función de JavaScript para que no se pueda introducir datos no permitidos en los campos. El texto introducido en cada campo aparecerá en rojo en caso de ser inadecuado. Si el usuario pulsa el botón save y los campos contienen datos incorrectos se visualizará un mensaje de error como el mostrado en la **figura 17**.



The screenshot shows a form titled "N°Historia" on a blue background. It contains five input fields with labels above them: "Enter MD" (value 21), "Enter sLv" (value e), "Enter Month" (value 1.2), and "Enter Year" (value 2009). The "Enter MD" and "Enter Year" fields have green borders, while the "Enter sLv" and "Enter Month" fields have red borders, indicating errors.

Figura 16. Pantalla Creación test

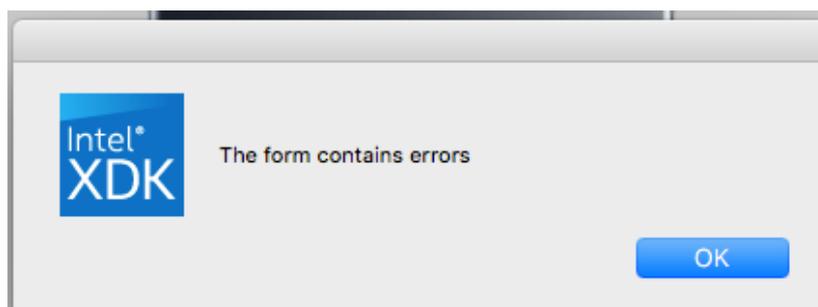


Figura 17. Mensaje error añadir prueba.

- **Visualización de pruebas:** esta función se mostrará después de haberse utilizado la función Buscar paciente previamente, habiéndose introducido el número de historia del mismo, lo que conducirá a un menú donde se muestran todas las pruebas asociadas a ese paciente.



The screenshot shows a mobile application interface with a blue background. At the top, it says "Main menu". Below that, there is a section labeled "N°History" with a white box containing the number "120". Underneath is a section labeled "Tests" which contains a table with four columns: "Month", "Year", "MD", and "sLv". The table lists four rows of data.

Month	Year	MD	sLv
4	2015	21	8.560
9	2018	20	6.869
4	2021	19.5	5.774
3	2019	19	6.179

Figura 18. Pantalla Main Menu visualización pruebas

- **Análisis de regresión:**

Los resultados del análisis de regresión lineal se mostrarán en la pantalla Main Menu del paciente seleccionado.

Los resultados mostrados serán el p-value de MD, p-value sLV o PSD, la pérdida de dB por año y si hay cambios significativos o no en función de los valores del p-value de los valores estudiados.

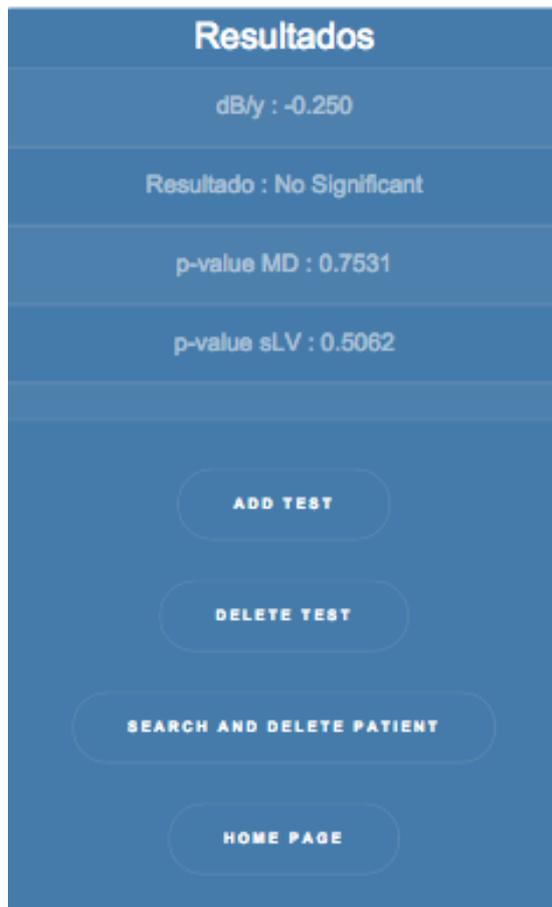


Figura 19. Pantalla Main con resultados del análisis de regresión lineal.

- **Eliminar prueba:** esta funcionalidad ofrece la posibilidad de borrar una prueba del paciente que tengamos seleccionado. Mostrará en la pantalla principal las pruebas que tiene asociadas ese paciente y el id de cada prueba.

El borrado de una prueba se realiza por el id asociado a la prueba dentro del almacén de objetos de la base de datos.

Delete test

N°History

120

Id Test

Enter Id

DELETE TEST

Tests

Id	Month	Year	MD	sLv
1	4	2015	21	8.560
2	9	2018	20	6.869

Figura 20. Pantalla Delete test

- **Ayuda:**

La pantalla FAQ se explicará paso a paso la manera de utilizar correctamente la aplicación y todas las funcionalidades que ofrece la misma.

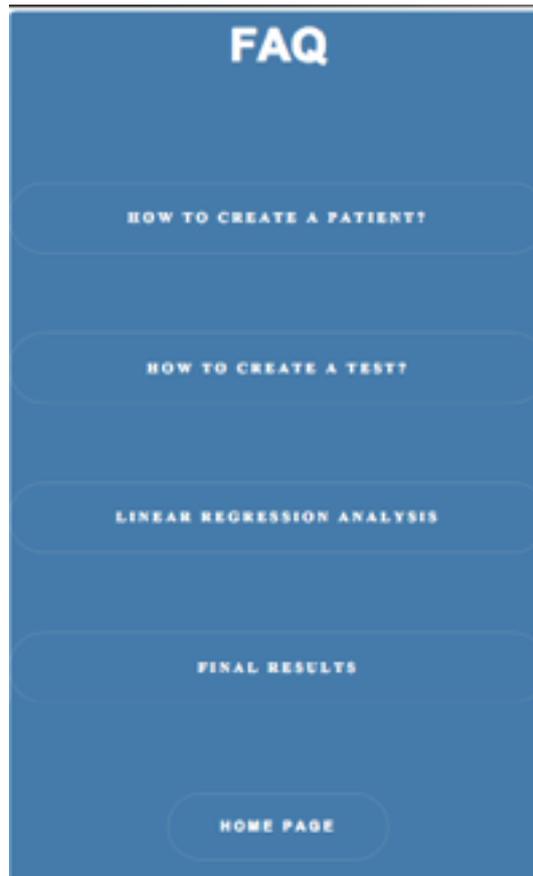


Figura 21. Pantalla FAQ

- **Gráfica:** esta funcionalidad muestra los datos estadísticos analizados anteriormente, representados en una gráfica en la que se muestran dos rectas: la roja se corresponde a los datos del MD y la azul a los datos de la sLv o PSD, después de haberse realizado el análisis de regresión lineal.

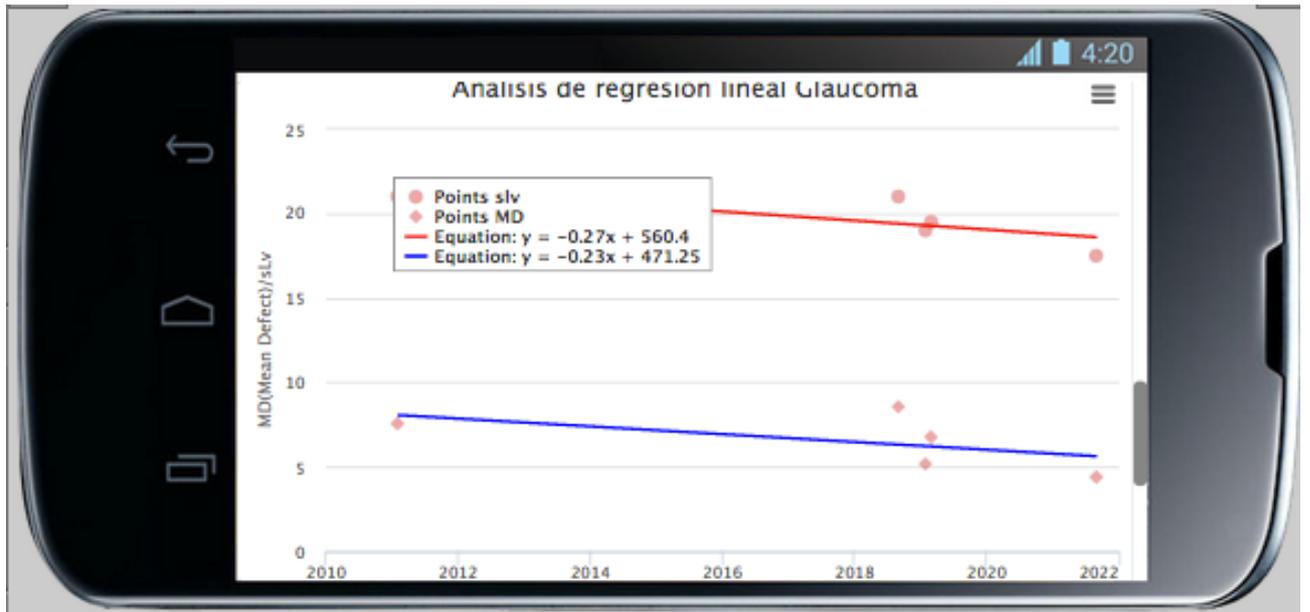


Figura 22.Gráfica generada por Highchart.

La funcionalidad gráfica ofrece la posibilidad de exportar la gráfica generada en múltiples formatos, mostrados en la **figura 22**.

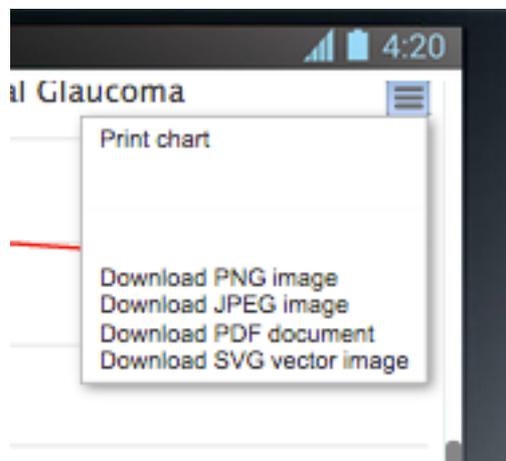


Figura 23.Imagen exportar gráfica.

6. Validación

Para la validación de la aplicación se solicitó la colaboración de los profesionales médicos del Servicio de Oftalmología del Hospital Universitario de Canarias. La aplicación fue instalada en sus dispositivos móviles y validada en la visita médica durante una semana. Como la eficacia del programa Quasar ya ha sido analizada y publicada [7, 12], hemos pedido a los profesionales médicos que valorasen la facilidad de uso de la aplicación, siendo el resultado positivo.

El principal problema que plantearon los profesionales al inicio de este proyecto fue que necesitaban una herramienta de trabajo intuitiva, rápida y de fácil uso para que, durante la visita médica, se pudiera analizar la progresión de los pacientes con glaucoma. En este aspecto la aplicación QuasarApp cumple las expectativas. Pero no se ha podido realizar una encuesta completa de usabilidad de la app para confirmar su grado de satisfacción por falta de tiempo.

7. Conclusiones y líneas futuras

7.1. Conclusiones

En este TFG se ha abordado la implementación de una app móvil, QuasarApp, partiendo de un programa llamado Quasar, para el seguimiento de la evolución del glaucoma. La app incluye además mejoras sobre el programa original del que se parte.

QuasarApp es una aplicación móvil que permite a los oftalmólogos expertos en glaucoma llevar a cabo el seguimiento del paciente de una manera más centralizada.

La realización de este proyecto me ha permitido aplicar los conocimientos técnicos adquiridos durante los años cursados en el Grado de Ingeniería Informática, el desarrollo de varias competencias transversales (como por ejemplo, el aprender a trabajar en equipos multidisciplinares) y la adquisición nuevos conocimientos, como el aprendizaje de nuevos lenguajes de programación (JavaScript), y el perfeccionamiento de otros (CSS y HML5).

Una de las dificultades de este proyecto ha sido el manejo de la base de datos, ya que al ser una base de datos no relacional han surgido algunos problemas a la hora de relacionar las pruebas con los pacientes, pero se han solventado correctamente mediante el uso del resto de lenguajes utilizados en el desarrollo.

La otra dificultad ha sido la programación de los cálculos estadísticos involucrados en el análisis de regresión lineal. Al no existir ninguna librería específica para ello, se han tenido que implementar funciones internas.

Por último, esta experiencia me ha permitido conocer el grado de implicación que conlleva crear una aplicación totalmente funcional en un ámbito real.

La aplicación QuasarApp ayudará a los expertos a realizar un mejor control de los pacientes, y hará que su trabajo sea más eficiente gracias a la utilización de las nuevas tecnologías.

7.1. Líneas futuras

Como líneas futuras de desarrollo, se pueden mencionar las siguientes:

- Apoyarse en un servicio de almacenamiento de datos en la nube de datos.
- La posibilidad de generar un informe que contenga todos los datos de los resultados del análisis de regresión lineal.
- Importar los datos desde la nube a la app sin necesidad de introducirlos, y hacer que el perímetro exporte a la nube los datos desde la realización de la perimetría.

8. Summary and Conclusions

8.1. Conclusions

In this *Final Undergraduate Project* has the implementation of a mobile app , QuasarApp, has been implemented from a program called Quasar. This program is focused on the study of the evolution of glaucoma. The app also includes improvements over the original program.

QuasarApp is a mobile application for glaucoma experts for developing a more centralized monitoring of the patients.

The realization of this project has allowed me to apply the acquired knowledge during the Computer Science Grade, the development of several generic skills (such as learning to work in multidisciplinary teams) and to learn new programming languages (JavaScript) , and to improve others (CSS and HML5).

One of the difficulties of this project has been handling with the database. This is a nonrelational database, and some problems have emerged when linking tests with patients. Finally, the problems have been properly solved by using other languages during the development.

The other difficulty has been the programming of the statistical calculations involved in the linear regression analysis .In the absence of any specific library for it , they have been implemented with internal functions.

Finally , this experience has allowed me to know the degree of responsibility that involves creating a fully functional application in a real environment.

The QuasarApp application will help experts to perform a better control of the patients, improving the efficiency of their work thanks to the use of new technologies .

8.2. Future Lines

Future work lines could be the following ones:

- To use a data storage service in the cloud.
- To add the ability to generate a report containing all resultant data from the linear regression analysis.
- To directly import data from the cloud to the app, and make the perimeter able to export perimetric data to the cloud.

9. Presupuesto

El presupuesto de este TFG se ha desglosado en la cantidad de horas de análisis y de implementación que se han empleado en cada funcionalidad de la aplicación multiplicada por el precio/hora estimado por el Colegio de Ingenieros Técnicos Informáticos de Canarias que sería de 20€ por hora de desarrollo.

Funcionalidad	Horas de análisis	Horas de implementación	Total
Configuración Base de datos	10	15	25
Creación de pacientes	5	10	15
Creación pruebas médicas	8	25	33
Encriptación datos	6	20	26
Gráfica	7	18	25
Diseño Estilo	8	20	28
Eliminación pruebas médicas	15	30	45
Búsqueda Pacientes	16	30	46
Análisis de regresión lineal	20	37	57
		Total	300

El presupuesto total de este TFG sería de un total de 6.000€.

Apéndice A. Código

A.1. Creación base de datos en IndexedDB

```
var dataBase = null;

dataBase = indexedDB.open("objectDb", 1);
dataBase.onupgradeneeded = function (e) {
    var active = dataBase.result;
    var objectDb = active.createObjectStore("pacientes",
    {keyPath: 'id', autoIncrement: true});
    objectDb.createIndex('by_dni', 'dni',
    {unique:true});
    var objectDb1 = active.createObjectStore("pruebas",{
    keyPath: 'id', autoIncrement: true});
    objectDb1.createIndex('by_dni', 'dni', {unique:
    false});
};

dataBase.onsuccess = function (e) {
    alert('Se ha cargado correctamente');
};

dataBase.onerror = function (e) {
    alert('Error loading database');}
};
```

A.2.Creación paciente

```
function addpatient(t) {

    var active = dataBase.result;
    var data = active.transaction(["pacientes"], "readwrite");
    var objectDb2 = data.objectStore("pacientes");

    var request = objectDb2.put({
        dni:calcMD5(document.querySelector("#dni").value),
        tipo: t
    });

    request.onerror = function (e) {
        alert('The indicated patient already created.');
```

```
        document.querySelector('#dni').value = '';
    };

    request.onsuccess = function () {
        alert('The patient has been createdsuccessfully');
```

```
    };

    data.oncomplete = function (e) {
        document.querySelector('#dni').value = '';
    };
}
```

A.3.Creación prueba paciente

```
function addprueba() {

    if(!document.querySelector('form').checkValidity()){
        alert('The form contains errors');}
    else{

var active2 = dataBase.result;
var data1 = active2.transaction(["pacientes"],
    "readwrite");
var objectDb = data1.objectStore("pacientes");
var index1 = objectDb.index("by_dni");
var request1 =
    index1.get(calcMD5(document.querySelector("#dni1").value)
);

request1.onsuccess = function () {

md1 = parseFloat(document.querySelector("#md").value);
slv1 = parseFloat(document.querySelector("#slv").value);
var result = request1.result;
    tipo1 = result.tipo;
        if( md1 >16.33 && tipo1 == "O" ){//octopus
            slv1 = slv1 + ((md1-16.33)/0.84);
        }else if( md1 <-17.35 && tipo1 =="H"){//humfrey
            slv1 = slv1 - ((md1+17.35)/0.84)}
//ABRIENDO TRANSACCIÓN ALMACÉN DE OBJETOS PRUEBAS
var active = dataBase.result;
var data = active.transaction(["pruebas"], "readwrite");
var objectDb1 = data.objectStore("pruebas");
var index = objectDb1.index("by_dni");
```

```

var request=index.get(calcMD5(document.querySelector
("#dni1").value));
    var request = objectDb1.put({
        md: document.querySelector("#md").value,
        slv: slv1,
        mes: document.querySelector("#mes").value,
        anyo: document.querySelector("#anyo").value,
        orde: parseFloat((
parseFloat(document.querySelector("#mes").value))+(12*
parseFloat((document.querySelector("#anyo").value)-2000))),
dni:calcMD5(document.querySelector("#dni1").value)
        });
    request.onerror = function (e) {
        alert('Error en la transacción');
    };

data.oncomplete = function (e) {

    document.querySelector('#md').value = '';
    document.querySelector('#slv').value = '';
    document.querySelector('#mes').value = '';
    document.querySelector('#anyo').value = '';
    alert('The medical test has been created
    correctly');
};
};
}
}

```

A.4. Búsqueda de paciente

```
function buscarpacient() {

    var active = dataBase.result;
    var data = active.transaction(["pacientes"], "readwrite");
    var object = data.objectStore("pacientes");

    var index = object.index("by_dni");
    var request = index.get(calcCMD5(document.querySelector
("#dni1").value));

        request.onsuccess = function () {
            if(request.result == undefined){
                alert('No such patient');
            }else{
                var result = request.result;
                alert("There the patient and has the ID: " +
result.id);
            }

            passdata(document.querySelector("#dni1").value);
        };
    };
}
```

A.5. Eliminación paciente

```
function deletedni() {
    var active = dataBase.result;
    var data = active.transaction(["pruebas"], "readwrite");
    var objectDb = data.objectStore("pruebas");

    objectDb.openCursor().onsuccess = function (e) {

        var result = e.target.result;
        if (result == null) {
            return;}
        if(result.value.dni == calcMD5(document.querySelector
("#dni1").value)){
            objectDb.delete(result.value.id);}

        result.continue();
    }
}

//ABRIENDO TRANSACCIÓN ALMACÉN DE OBJETOS PRUEBAS
var active = dataBase.result;
var data1 = active.transaction(["pacientes"], "readwrite");
var objectDb2 = data1.objectStore("pacientes");
var index = objectDb2.index("by_dni");
var request = index.get(calcMD5(document.querySelector
("#dni1").value));

request.onsuccess = function (){
    var result1 = request.result;
    objectDb2.delete(result1.id);}
data.oncomplete = function (e) {
    alert('The patient has been deleted successfully');
    document.querySelector('#dni1').value = '';
};
}
```

A.6. Borrar pruebas

```
function deleteprueba() {

    var active = dataBase.result;
    var data1 = active.transaction(["pruebas"], "readwrite");
    var objectDb2 = data1.objectStore("pruebas");
    objectDb2.openCursor().onsuccess = function (e) {
        var result1 = e.target.result;
        if (result1 == null){
            return;}
        if(result1.value.dni ==
        calcMD5(document.querySelector("#dni1").value) &&
        result1.value.id ==document.querySelector("#id").value)
        {
            if (confirm("Would you like delete this test?") ==
            true){
                objectDb2.delete(result1.value.id);
                alert('The medical test has been deleted
                correctly');}
            else
                alert('Deletion canceled by user');}
            else{
                result1.continue();}
        };

    data1.oncomplete = function (e){
        if (confirm("Would you like continue deleting test?")
        == true){
            pasdata(document.querySelector("#dni1").value, 'deletepru
            eba');}
            else
                pasdata(document.querySelector("#dni1").value, 'menu')
            };
    }
}
```

En este apéndice se ha incluido las funciones más significativas de la aplicación y algunas no se han podido incluir debido a su gran extensión.

El código completo se puede encontrar en el repositorio de Github del proyecto **[27]**.

Bibliografía

- [1] <http://www.clinicabaviera.com/glaucoma>
- [2] <https://www.heidelbergengineering.com/international/products/hrt/hrt-glaucoma/>
- [3] http://www.zeiss.es/meditec/es_es/productos---soluciones/oftalmologia-y-optometria/glaucoma/diagnostics/optical-coherence-thomography/familia-cirrus/cirrus-hd-oct.html
- [4] <http://www.glaucoma.org.il/glaucoma/diagnosis/gdx>
- [5] <http://es.slideshare.net/wildercito35/interpretacion-del-campo-visual-computadorizado>
- [6] <http://www.oftalmologiaprivada.com/innovanet/macros/TextContentWithMenu.jsp?contentid=45&channel=innova.net>
- [7] Valentín Tinguaro Díaz Alemán. Tesis doctoral sobre el Estudio comparativa de procedimientos para la evaluación de la progresión campimétrica en pacientes glaucomatosos, 2008.
- [8] http://www.oftalmoseoformacion.com/wp-oftalmoseo/documentacion/Cap_08-02_Perimetria_Octopus.pdf
- [9] http://www.zeiss.es/meditec/es_es/productos---soluciones/oftalmologia-y-optometria/glaucoma/diagnostics/perimetry/humphrey-hfa-ii-i.html
- [10] <http://www.oculus.de/es/productos/perimetria/easyfield/software-perimetro/threshold-noiseless-trend-tnt-analisis-de-progresion/>
- [11] <http://www.zeiss.com/content/dam/Meditec/us/download/Glaucoma%20Landing%20Page/masteringgpasupplement.pdf>
- [12] Valentín T. Díaz-Alemán, Marta González-Hernández, Daniel Perera-Sanz and Karintia Armas-Domínguez. RESEARCH REPORT: Evaluation of Visual Field Progression in Glaucoma: Quasar Regression Program and Event Analysis, 2015.
- [13] <https://www.python.org/>
- [14] <https://build.phonegap.com/>
- [15] <https://monaca.io/es/>

- [16] <http://www.batanga.com/tech/13241/que-es-apache-cordova>
- [17] <http://www.appcelerator.org/#titanium>
- [18] <https://html5up.net/>
- [19] https://es.wikipedia.org/wiki/Hoja_de_estilos_en_casca_da
- [20] <https://es.wikipedia.org/wiki/JavaScript>
- [21] <https://software.intel.com/es-es/intel-xdk>
- [22] <https://developer.mozilla.org/es/docs/IndexedDB>
- [23] <https://rolandocaldas.com/html5/indexeddb-tu-base-de-datos-local-en-html5>
- [24] <http://www.highcharts.com/>
- [25] <http://www.genbetadev.com/javascript/highcharts-graficas-con-javascript>
- [26] <https://github.com/>
- [27] <https://github.com/alu0100537451/QuasarApp>