



# **APLICACIÓN: QUIT CHALLENGE REPORTE FINAL**

Aplicaciones Móviles

Profesor: Gerardo Ayala San Martín  
Primavera 2018

Alina Daniela Fernández Jiménez  
150662

# Tabla de contenido

<b>1</b>	<b>Análisis del problema a resolver .....</b>	<b>3</b>
1.1	Identificación y formulación del problema .....	3
1.2	Casos de uso.....	4
<b>2</b>	<b>Impacto socio económico.....</b>	<b>4</b>
<b>3</b>	<b>Ingeniería de software.....</b>	<b>5</b>
<b>4</b>	<b>Diseño computacional. ....</b>	<b>6</b>
4.1	Diagrama UML de Secuencia.....	6
4.2	Esquema de la Base de Datos (Modelo Entidad-Relación):.....	8
<b>5</b>	<b>Diseño de las interfaces.....</b>	<b>9</b>
<b>6</b>	<b>Programación. ....</b>	<b>13</b>
<b>7</b>	<b>Resultados .....</b>	<b>15</b>
	<b>Bibliografía .....</b>	<b>18</b>

# 1 Análisis del problema a resolver

## 1.1 Identificación y formulación del problema

En México existe un grave problema de tabaquismo, a pesar de los distintos esfuerzos que se han realizado con el fin de terminar este vicio. El consumo de tabaco involucra a al menos un 15% de la población mexicana, según datos recientes del Instituto Nacional de Enfermedades Respiratorias. Cada año se estima que mueren más de 5 millones de personas en el mundo a causa del tabaquismo y si no se toman medidas adecuadas en el año 2030 este número puede llegar a los 10 millones. El consumo de esta sustancia afecta la salud de fumadores y de no fumadores expuestos, pudiendo llegar a muerte prematura por enfermedades relacionadas con el tabaquismo.

El consumo de tabaco afecta a los jóvenes, siendo casi el 60 por ciento de los nuevos fumadores menores de 18 años cuando fumaron un cigarrillo por primera vez. Se cree que es debido a que en esta edad los jóvenes y adolescentes son más propensos a fumar debido a influencias psicosociales, como presión por parte de sus compañeros o amigos, aunque según estudios recientes en algunas ocasiones también puede afectar el factor biológico.

El riesgo de desarrollar cáncer debido al tabaco depende de muchos factores, así como el número de años que se estuvo expuesto al tabaco, el número de cigarrillos fumados y la intensidad de la inhalación. Tan grave es el problema que 1 de cada 3 casos de cáncer es provocado por el tabaquismo. Otras enfermedades que puede ocasionar esta sustancia son las siguientes:

- Asma
- Bronquitis crónica
- Enfermedades cerebro-vasculares
- Enfermedad pulmonar obstructiva
- Retraso del crecimiento del feto en mujeres embarazadas
- Aparición de arrugas
- ORL

## 1.2 Casos de uso

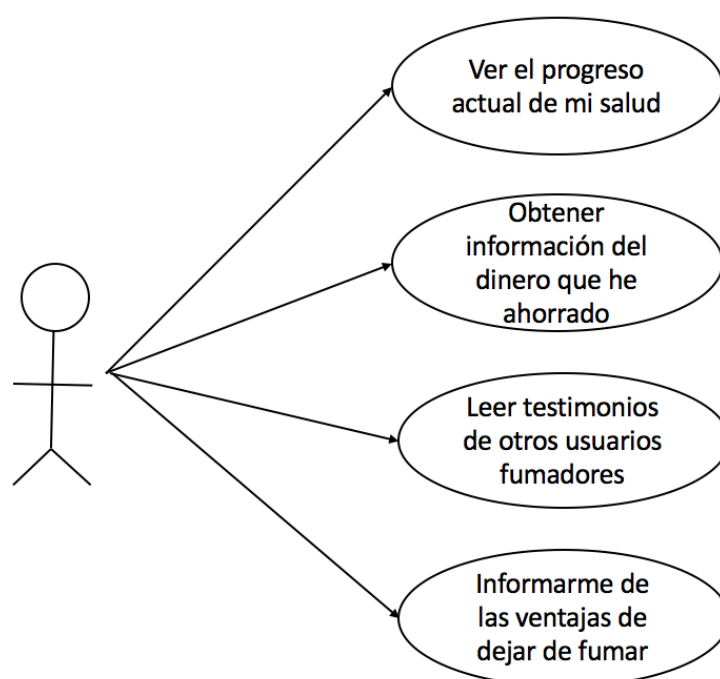


Figura 1. Casos de uso del usuario

Como podemos ver en la imagen anterior, el usuario posee distintas funcionalidades que le permiten usar la aplicación para lograr su principal objetivo y motivación: dejar de fumar.

## 2 Impacto socio económico.

La aplicación está pensada para mejorar o facilitar específicamente al usuario el proceso de dejar de fumar. Se sabe que dejar el cigarro no es un proceso fácil y la aplicación no garantiza que el usuario logre su cometido, sin embargo, generará en él una motivación intrínseca aún mayor y la ventaja principal es que logrará llevar un registro de su progreso, así como recibir consejos e información relevante.

Esta aplicación también ayudará económicamente a los usuarios, ya que además de poder seleccionar que el dinero es su principal motivación para dejar de fumar, la aplicación contiene una sección dedicada concretamente a ver el progreso económico que genera dejar de fumar. Adicionalmente, se agregaron elementos de gamificación

como lo son niveles o rangos. En este caso el rango 4 es para los usuarios nuevos que no han ahorrado mucho dinero por dejar de fumar ya que no llevan mucho tiempo sin hacerlo, los rangos se muestran en forma de pirámide, siendo la punta y el mayor rango que se pueda alcanzar el número 1.

### **3 Ingeniería de software.**

La metodología que se siguió para el desarrollo de este proyecto fue una metodología incremental, también conocida como Scrum. Según Joel Francia, “Scrum es un proceso de gestión y control que reduce la complejidad en el desarrollo de productos para satisfacer las necesidades de los clientes”, es decir, el problema se divide en entregas parciales del producto final para obtener pronto resultados.

Después de decidir el diseño de la aplicación se comenzó a trabajar en las pantallas, para así tener el primer entregable o Sprint. Se identificaron los requerimientos del usuario final, el problema principal a resolver y los casos de uso. Finalmente se presentó un prototipo de la aplicación a desarrollar.

En el siguiente Sprint, se trabajó en elementos de diseño UML, el modelo de datos y los métodos principales, aunque pendientes de implementar. En esta etapa también se hicieron cambios a la interfaz gráfica y experiencia de usuario. Finalmente se entregó un proyecto compilable.

Por último, en el tercer Sprint se trabajó en la implementación de los métodos, así como del modelo de datos de la aplicación. Se corrigieron también pantallas y finalmente se entregó una aplicación funcional con todas las características planeadas en un inicio.

## 4 Diseño computacional.

### 4.1 Diagrama UML de Secuencia

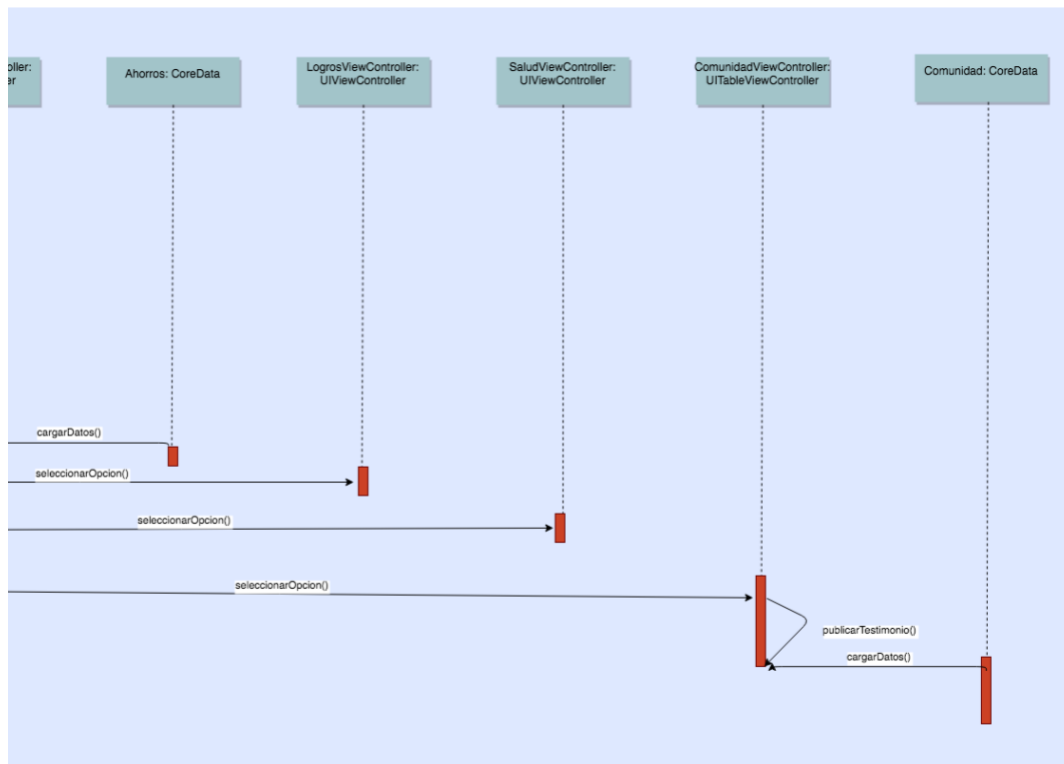
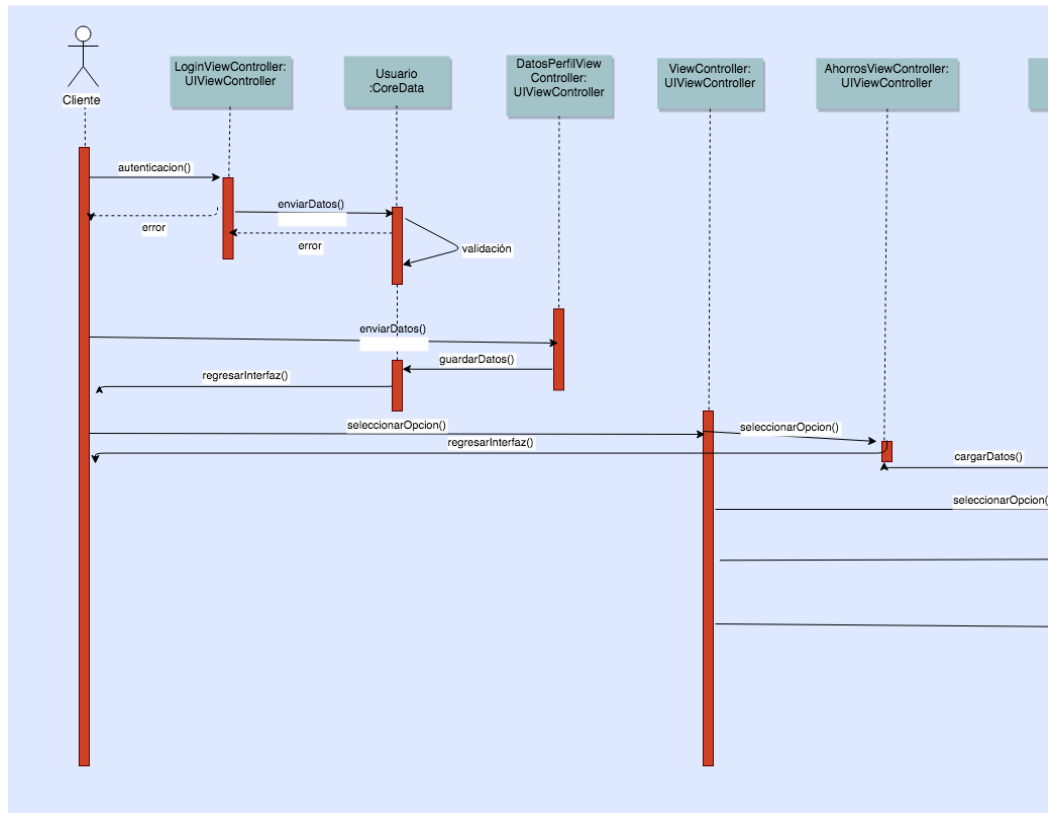


Figura 2. Diagrama UML de Secuencia de acciones del usuario

El diagrama de secuencias se divide en las siguientes acciones:

- Primero se lleva a cabo el proceso de autenticación, en caso de que el usuario inicie sesión para validar sus datos. En caso de crear un nuevo usuario, sus datos se guardarán en la tabla de usuario.
- Después el usuario pasará a la pantalla de inicio, desde la cual podrá seleccionar las siguientes opciones:
- Ahorros: Desde esta pantalla se cargan datos de la tabla de ahorros para ese usuario.
- Logros: Esta pantalla carga los datos de las medallas del usuario.
- Salud: Esta pantalla carga los datos del usuario en cuanto a su salud.
- Comunidad: En esta pantalla se puede publicar un testimonio, el cual se cargará en la tabla de Testimonios.

## 4.2 Esquema de la Base de Datos (Modelo Entidad-Relación):

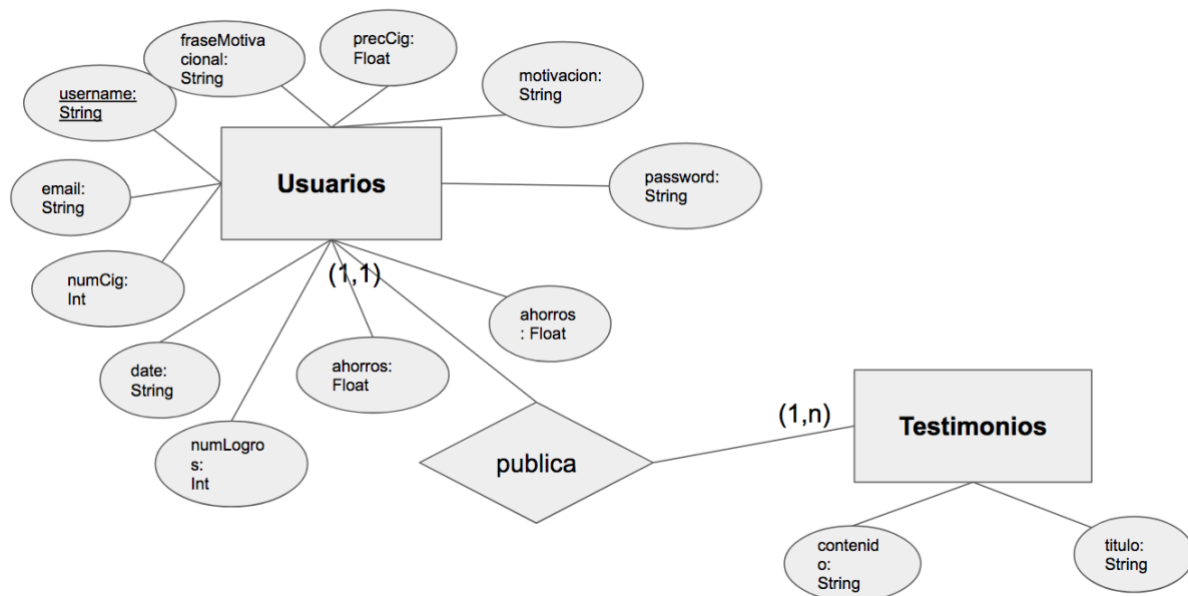


Figura 3. Modelo Entidad-Relación

Las tablas en donde se almacena la información son las siguientes:

- **Usuarios:** Esta tabla almacena un **username** que identifica al usuario, un **email** que corresponde a un correo, un **password** que corresponde a una contraseña, una **fraseMotivacion** que corresponde a una frase personal del usuario, unos **numCig** que corresponden al número de cigarros que el usuario fuma al día, un **precCig** que corresponde al precio individual de cada cigarro (esto para calcular sus ahorros posteriormente), también tiene una **date** que corresponde a la fecha que el usuario decide que dejará de fumar completamente. Para llevar cuenta de los logros tenemos **numLogros** y **ahorros** para contar cuánto lleva ahorrado. Por último, tenemos una **motivación** que se refiere a por qué el usuario dejó de fumar (puede ser Salud, Dinero, Seres Queridos, Estética u Otro).
- **Testimonios:** Esta tabla almacena los testimonios de los usuarios, el **título** y el **contenido** de estos.

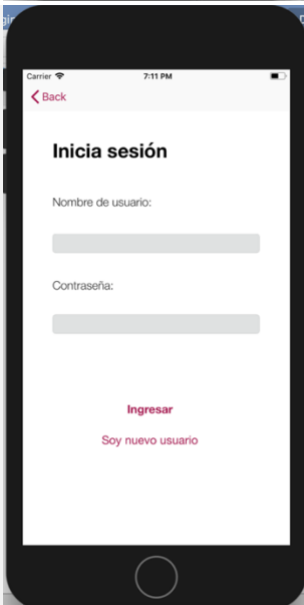


## 5 Diseño de las interfaces.

El diseño gráfico de las vistas de la aplicación se describe a continuación:



Pantalla inicial: Nombre y logo de la aplicación, botón de comienzo.



Inicio de sesión: Si el usuario ya tiene una cuenta únicamente ingresa su nombre de usuario y contraseña, de lo contrario crea una nueva cuenta.

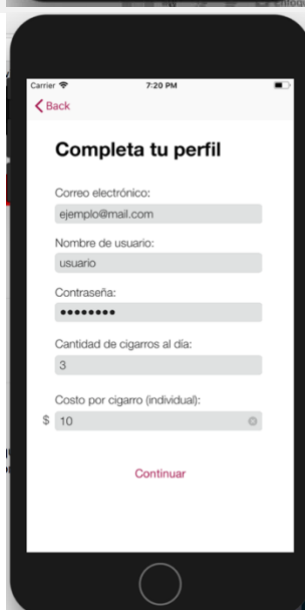


En esta pantalla, el usuario tiene la opción de elegir una fecha para dejar de fumar, esto lo motiva a dejarlo definitivamente para esa fecha.



Por otra parte, al crear su perfil el usuario puede seleccionar una motivación para dejar de fumar. Estas pueden ser:

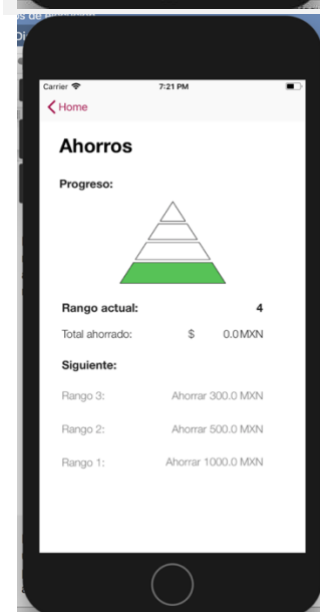
- Salud
- Dinero
- Seres queridos
- Estética
- Otro



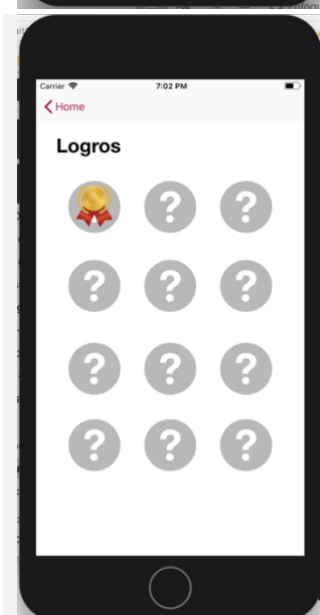
En esta pantalla el usuario puede completar los datos de su perfil tales como correo, nombre de usuario, contraseña, la cantidad de cigarros que fuma al día y el costo de cada cigarro. Esto último servirá para calcular el ahorro promedio del usuario.



Esta pantalla muestra el inicio de la aplicación con las diferentes opciones y una frase motivacional hasta arriba (editable). Las opciones son: Ahorros, Logros, Salud y Comunidad. Abajo se encuentra el botón de configuración, en el cual podrá editar los datos del perfil, así como un botón de consejos con el cual se mostrará algún consejo para dejar de fumar al presionarlo.



Pantalla de ahorros: Muestra al usuario el total de dinero ahorrado y se clasifica en rangos, siendo 4 el más bajo y el 1 el más alto. También le indica qué cantidad le falta para subir de rango.



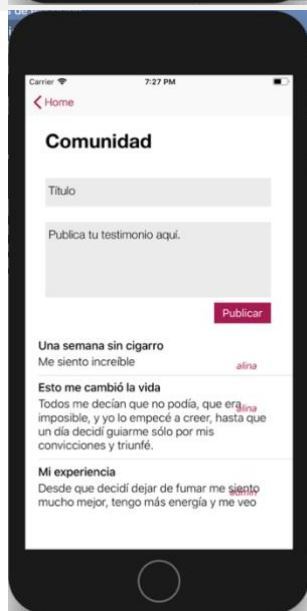
Pantalla de logros: En esta pantalla se muestran los logros del usuario, estos pueden ir desde completar su perfil hasta lograr avances significativos en la aplicación.



Pantalla de salud: Se muestran los avances generales en cuanto a tiempo que lleva sin fumar, la cantidad de cigarros que no ha fumado, el tiempo de vida ganada y más detalles.



En los avances detallados de salud podemos ver más datos que le ayudan al usuario a visualizar sus mejoras de salud con base en el tiempo que lleva sin fumar, estos datos fueron obtenidos de: [https://www.clarin.com/buena-vida/salud/pasa-cuerpo-dejas-fumar\\_0\\_r1NY5FKPml.html](https://www.clarin.com/buena-vida/salud/pasa-cuerpo-dejas-fumar_0_r1NY5FKPml.html)



Por último, la pantalla de comunidad les permite a los usuarios publicar un testimonio o simplemente compartir opiniones con los demás usuarios.

## 6 Programación.

La aplicación móvil hace uso del framework de Apple; CoreData. Core Data simplifica la gestión de modelo de datos de las aplicaciones iOS, nos ayuda con la parte de la persistencia, para que los datos sigan guardados incluso cuando la aplicación deja de ejecutarse.

Los datos son guardados en entidades, en este caso tenemos la entidad “usuario” y la entidad “testimonio”. Se usa un contexto para guardar los datos en un contenedor persistente, así, le asignamos valor a cada variable al crear un nuevo usuario, lo mismo sucede al guardar un nuevo testimonio. Para visualizar los datos se maneja por medio de solicitudes usando `NSFetchRequest`.

Para evitar repetir código y ahorrar tiempo de ejecución se utilizó el método “prepare for segue” en Swift, el cual ayuda para facilitar el manejo de secuencias y cambios entre pantallas. En este caso, tuvimos que transferir la variable de “username” una vez que el usuario inicia sesión o crea una nueva cuenta.

```
override func prepare(for segue: UIStoryboardSegue, sender: Any?) {
    if(segue.identifier == "conf") {
        let controller = segue.destination as? ConfigurationViewController
        controller?.user = username
    }
    if(segue.identifier == "ahorros"){
        let controller = segue.destination as? AhorrosViewController
        controller?.usuario = username
    }
    if(segue.identifier == "logros"){
        let controller = segue.destination as? LogrosViewController
        controller?.usuario = username
    }
    if(segue.identifier == "com"){
        let controller = segue.destination as? ComunidadViewController
        controller?.usuario = username
    }
}
```

Figura 4. Transferencia de variable *username*

Otra de las funciones importantes fue la de calcular el rango del usuario para la sección de ahorros y desplegar el texto adecuado dependiendo del rango, ya que este contiene una operación matemática que calcula la cantidad exacta de dinero que falta ahorrar para pasar al siguiente rango (Figura 5).

```

func cargarRango() {
    //Esta función obtiene info del nivel del usuario y la carga a la pantalla
    calcularRango = Float(Model.returnParameter(user: usuario, type: "rango"))!
    totalAhorrado.text = String(calcularRango)
    if calcularRango < 300.0{
        rangoActual.text = "4"
        rango3.text = "Ahorrar \(300.0-calcularRango) MXN"
        rango2.text = "Ahorrar \(500.0-calcularRango) MXN"
        rango1.text = "Ahorrar \(1000.0-calcularRango) MXN"
    }else if calcularRango < 500.0{
        rangoActual.text = "3"
        rango3.text = "Alcanzado. ¡Felicidades!"
        rango2.text = "Ahorrar \(500.0-calcularRango) MXN"
        rango1.text = "Ahorrar \(1000.0-calcularRango) MXN"
    }else if calcularRango < 1000.0{
        rangoActual.text = "2"
        rango3.text = "Alcanzado. ¡Felicidades!"
        rango2.text = "Alcanzado. ¡Felicidades!"
        rango1.text = "Ahorrar \(1000.0-calcularRango) MXN"
    }else{
        rangoActual.text = "1"
        rango3.text = "Alcanzado. ¡Felicidades!"
        rango2.text = "Alcanzado. ¡Felicidades!"
        rango1.text = "Alcanzado. ¡Felicidades!"
    }
}
}

```

Figura 5. Cargar el rango del usuario

Por último, una función muy importante para el correcto funcionamiento de la aplicación es la de autenticar al usuario, para esto se hace una solicitud al modelo de datos mandando el usuario y contraseña ingresados, si encuentra el objeto es verdadero, de lo contrario, falso.

```

static func authenticate(user: String, password: String) -> Bool{
    let appDelegate = UIApplication.shared.delegate as! AppDelegate
    let managedContext = appDelegate.persistentContainer.viewContext
    let request = NSFetchRequest<NSFetchRequestResult>(entityName: "Users")
    request.returnsObjectsAsFaults = false
    do {
        let results = try managedContext.fetch(request)
        if results.count > 0 {
            for result in results as! [NSManagedObject]{
                if user == result.value(forKey: "username") as? String && password ==
                    result.value(forKey: "password") as? String{
                    print("Usuario y contraseña autenticada")
                    return true
                }
            }
        }
    } catch {
        print("Usuario o contraseña incorrectos")
    }
    return false
}

```

Figura 6. Función autenticación.

En la pantalla de inicio de sesión se manda a llamar la función `shouldPerformSegue`, que indica si se debe pasar a la pantalla de inicio o no. En caso de que el resultado de la función de autenticar sea falso, la aplicación despliega una alerta de usuario o contraseña incorrectos, en caso de que sea verdadero pasa a la pantalla de inicio.

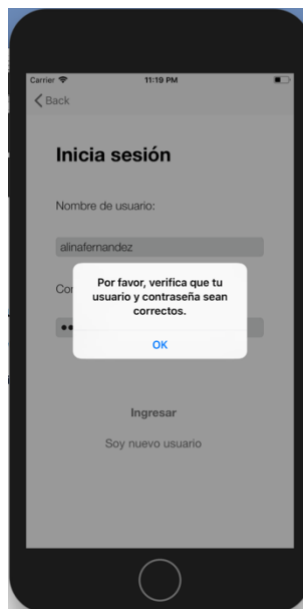
```
override func shouldPerformSegue(withIdentifier identifier: String, sender: Any?) -> Bool{
    if(identifier == "auth"){
        if authentication{
            return true
        }else{
            self.present(alert, animated: true)
            return false
        }
    }
    return true
}
```

Figura 7. Función para validar cambio de pantalla.

## 7 Resultados

Se presentan los siguientes ejemplos de ejecución de la aplicación:

1. Inicio de sesión con usuario o contraseña incorrectos.



La aplicación no nos deja iniciar sesión ya que ese usuario no existe.

2. Creación de nuevo usuario



Los datos son guardados correctamente.

### 3. Visualización de logros



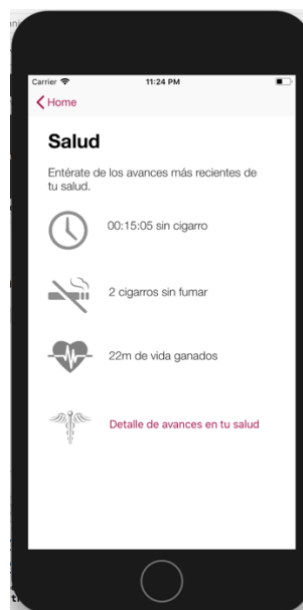
Al dar clic en la medalla nos despliega una alerta con información de ese logro.

### 4. Visualización de un consejo.





##### 5. Visualización de cronómetro, tiempo sin fumar.



Estos fueron algunos de los principales ejemplos de ejecución de la aplicación. Se piensa añadir más funcionalidades en un futuro para mejorar la experiencia del usuario y que se motive más al usar esta aplicación

## Bibliografía

Abuse, N. I. (1 de marzo de 2010). *Advancing Adiction Science*. Recuperado el 10 de febrero de 2018, de El Tabaquismo y los Adolescentes:

<https://www.drugabuse.gov/es/publicaciones/serie-de-reportes/adiccion-al-tabaco/el-tabaquismo-y-los-adolescentes>

Baena, A. (2017). *Tabaquisme*. Recuperado el abril de 2018, de 10 pasos para dejarlo:

<http://www.tabaquisme.cat/es/dejar-de-fumar/estoy-decidido/10-pasos-para-dejarlo/>

Francia, J. (septiembre de 2017). *Qué es Scrum?* Recuperado el mayo de 2018, de

Scrum.org: <https://www.scrum.org/resources/blog/que-es-scrum>

*Salud CCM*. (mayo de 2017). Recuperado el 10 de febrero de 2018, de Consecuencias del Tabaco en la Salud: <http://salud.ccm.net/contents/481-consecuencias-del-tabaco-en-la-salud>