



## **SOFTWARE HOME LIFE**

Tutor: Maria Alexandra Silva Monsalve

Carolina Arcila Argaez - 2011982751

Julián Eusebio Morantes Ríos - 2011982645

Dairo Fernandez Diaz - 1711021830

Politécnico Gran Colombiano  
Facultada De Diseño E Innovación  
Programa de Ingeniería de Software Bogotá  
2020

## **INTRODUCCIÓN.**

Teniendo claro la importancia y la aclaración sobre el software HOME LIFE, a continuación, describiremos lo que se realizará en la herramienta, el alcance del software, un poco sobre su arquitectura y la metodología que se implementara para la construcción de este y también la implementación de integración continua en este.

## DEFINICIÓN DEL SOFTWARE.

**Home Life**, es una plataforma que permitira conectar a empleados del mundo de la construccion, al mundo de empleados de aseo, y servicios varios con clientes potenciales los cuales tendran una necesidad en especifico.

Este proyecto es una plataforma que nos permitirá solicitar servicios de arreglos de vivienda, en esta época cuando estamos en casa, podemos notar que aumenta el daño de partes en nuestras viviendas partes como enchufes, daños en tuberías, daños en baños, ventanas entre otros tipos de daños, nuestra aplicación nos permitirá realizar la solicitud de un experto en el campo, y de esta forma dar solución a esos accidentes o daños que hay en nuestra vivienda, estos servicios serán categorizados de la siguiente manera : electricidad, acueducto, construcción, obra blanco, obra negra, entre otros tipos de servicios.

Una vez se realice la solicitud del servicio, este será enviado a un profesional en el campo, o incluso puede ser seleccionado por el usuario de una lista de posibles colaboradores según su calificación en la plataforma.

La aplicación se desarrollará en 4 componentes principales, los cuales son:

### **Servicios de Api (Back End) :**

Estos servicios son los que nos permiten realizar todo el core de la plataforma, y es donde se encontrara toda la lógica de negocio de esta.

### **Aplicación Móvil (FrontEnd):**

Es la plataforma que nos permitirá al usuario ingresar y seleccionar el tipo de servicio requerido para su hogar.

### **Aplicación web (FrontEnd)**

Esta plataforma nos permitirá administrar y validar que tipo de servicios existen y también nos permitirá realizar servicios desde un ordenador.

### **Aplicación Móvil Colaborador (FrontEnd)**

Aplicación que permitirá a un colaborador ver y gestionar los servicios asignados a él.

El tiempo es corto por lo tanto nos comprometemos con la primera parte de la aplicación la cual consiste en realizar los servicios Core REST Api. Ya que este es el proceso principal de la aplicación y de este se delega el uso del resto de la aplicación.

Para esto decidimos utilizar el lenguaje de desarrollo. **NetCore 3.1**, con **C#**, y la implementación de **angular 9** por algunas vistas.

También implementaremos **Azure DevOps** para realizar la gestión del proyecto; el cual en cuenta con un repositorio **Git**, para el control de versiones e implementaremos una integración continua con los **Pipelines** que este nos brinda.

## **ALCANCE DEL PROYECTO.**

Para realizar este proyecto se utilizara una estrategia incremental y una metodologia agil llamada scrum, en donde realizamos una plantacion de el proyecto, en el cual realizamos un minimo producto viable MVP el cual sera realizado para la implementacion de este proyecto.

En el MVP se comprometen los siguientes Modulos:

### **Modulo de Inicio de sesion :**

Modulo que tendra la administracion de usuario, logeo y authentication de la aplicación, este modulo permitira a un usuario normal, o usuario colaborador realizar su inicio de sesion y de esta forma validar las opciones permitidas a este.

### **Modulo de registro de servicios:**

Modulos que nos permitira registrar a un usuario normal, una solicitud de servicios, estos servicios solicitaran algunos datos personales para de esta forma tener una informacion completa sobre el servicio a prestar.

### **Modulo de Notificaciones:**

Este modulo nos permite enviar notifficaciones a los usuarios y a el colaborador, logrando comunicarse entre ellos por medio de un chat.

### **Modulo de Calificacion de servicios :**

Este modulo nos permitira dejar comentarios y calificar a la persona que presto el servicio de esta forma se podra tener un puntaje y saber la calidad de persona que presta el servicio.

## **PRIMERA ENTREGA**

1. Para esta primera entrega se realizará la arquitectura del sistema.  
Crearemos la línea base del sistema en Net Core 3.1 contando con 4 capas, las cuales son Capa de presentación API, Capa de negocio, Capa de modelos y dto y por último una capa de repositorios en relación con la línea base.
2. Como segunda parte entregaremos el módulo de seguridad – Login del sistema, el cual contendrá usuario y contraseña y el sistema identificará el rol que el usuario tiene y así determinará a que módulos tiene acceso.

## **SEGUNDA ENTREGA**

1. La segunda entrega se entregará el modulo de autogestión el cual consiste en que el usuario pueda realizar el registro de sus datos personales y selección de cargo que va a tener su perfil – Usuario colaboradores.
2. También se realizará el modulo de solicitud de un servicio, el cual consiste en que el usuario “Solicitante” pueda diligenciar un formulario, donde registra los datos importantes para que los colaboradores puedan tomar el servicio que están solicitando y cumplir con la labora.

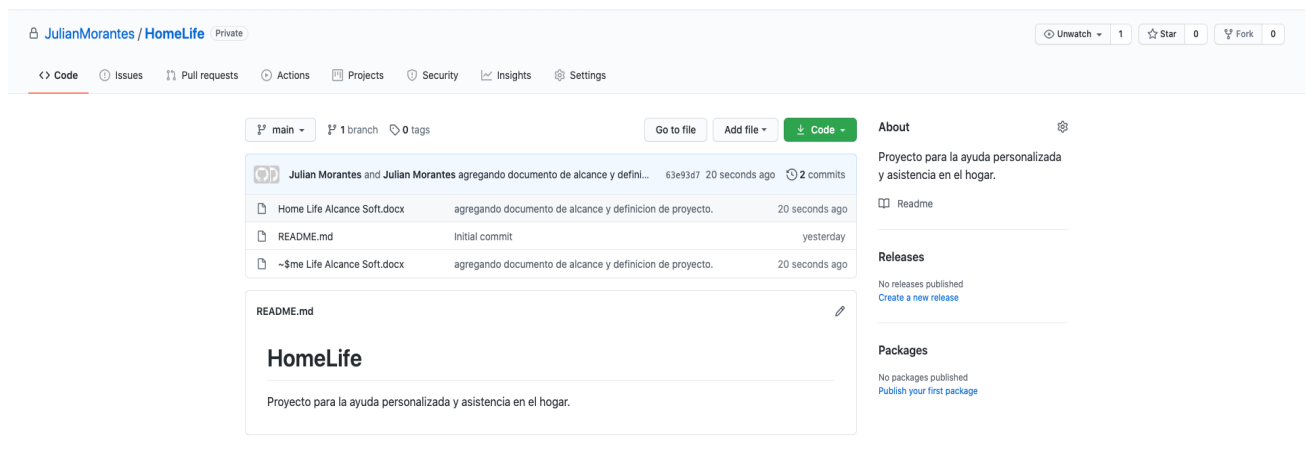
## **TERCERA ENTREGA**

1. Para la tercera entrega se realizará el modulo de calificación del colaborador, el cual tendrá como opciones los valores 1,2,3,4,5 estrellas. Donde el 1 es el servicio mas inconforme y el 5 es el servicio más conforme.
2. También se realizará el modulo de Comentarios donde los clientes podrán agregar observaciones sobre sus experiencias en cuanto al servicio que presto el colaborador.

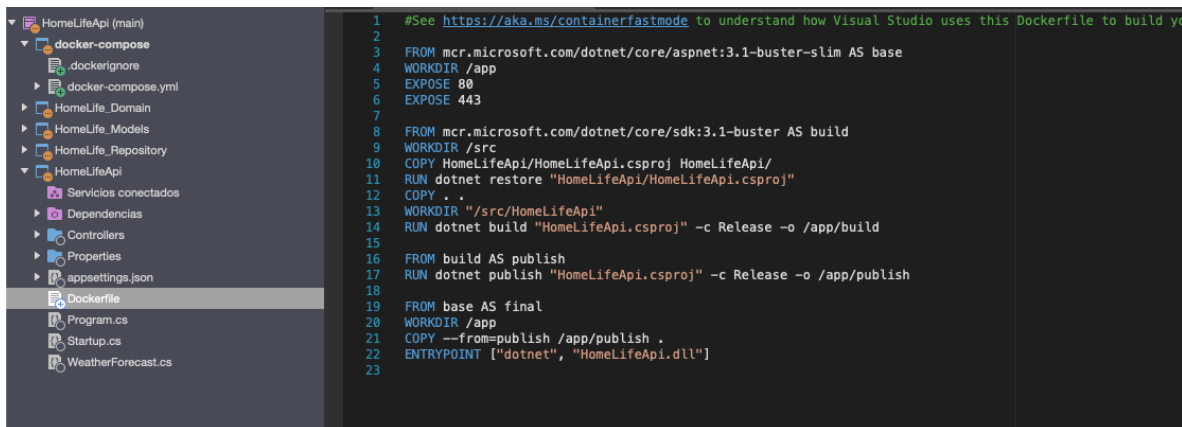
## CONTROL DE VERSIONES

Nuestro código se encuentra versionado en Git, y su repositorio se encuentra dividido en dos, uno reposado en git hub, por solicitud de entrega para la actividad, y el otro en Azure DevOps, administrador de procesos de integración continua.

Link GitHub : <https://github.com/JulianMorantes/HomeLife.git>

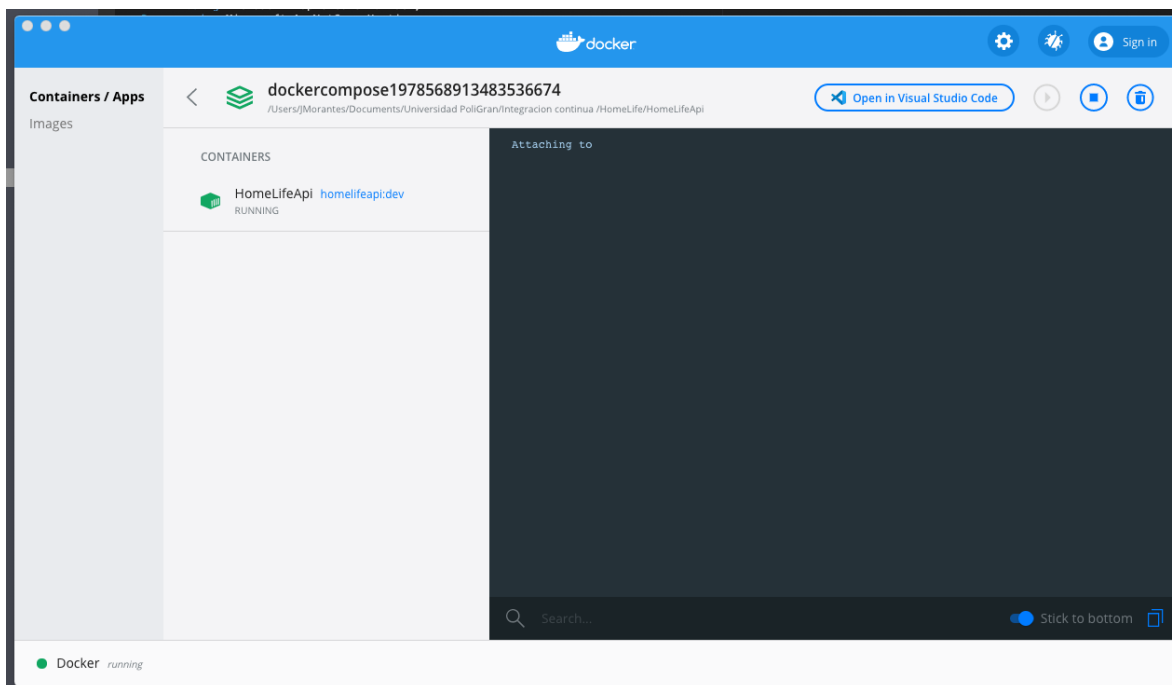


En nuestro proyecto estamos implementando una api contenida en docker, esto lo implementamos en nuestro proyecto principal de compilacion



Se está utilizando por ahora un servidor docker instalado localmente, de esta forma nos permite correr la aplicación.

Evidencia de que la aplicación esta ejecutando contenida en un archive de docker



Y la api corriendo de forma exitosa :

