

Práctica Profesional Supervisada

Tecnicatura Universitaria en Programación

Proceso: Finalizado

Materia: Practica Profesional Supervisada.

Coordinador: Ingeniero Osvaldo Giordanini

Universidad: UTN. Regional San Rafael.

Provincia: Mendoza.

País: Argentina.

Alumno: Membrive Denis Andrés

D.N.I.: 35663945

Legajo: 9532

Domicilio: Sarmiento 1314 Cuadro Nacional.

Ciudad: San Rafael.

Provincia: Mendoza.

Código Postal: 5607.

País: Argentina.

Empresa: Asistencias Médicas SRL

Cuit: 30-71383090-5

Gerente de la Empresa: Gettor Sartini Sofía Inés.

Dirección: Manuel Ruibal Este 187

Ciudad: Malargüe.

Provincia: Mendoza.

País: Argentina

Periodo de Practica: Entre mes de Septiembre 2023 a Noviembre 2023.

Horas Reloj: 60

INDICE

- 1. Datos Alumno**
- 2. Datos Empresa**
- 3. Propuesta Practica Profesional**
- 4. Actividades realizadas**
 - a. Digitalización de la Empresa**
 - b. Desarrollo Página web**
 - c. Administración base de datos SQL vinculada a
software Shamman Express.**

1. DATOS DEL ALUMNO



Datos Personales:

- Nombre y Apellido: Membrive Denis Andrés
- Fecha de Nacimiento: 6 de Julio de 1990
- Nacionalidad: Argentina
- Ciudad: San Rafael, Mendoza
- Código Postal: 5.607
- Estado Civil: Casado
- Documento: DU 35.663.945
- CUIL: 20-35663945-7
- Domicilio: Sarmiento Sur 1010 (Cuadro Nacional)
- Teléfono: +54 260-4670003
- Correo Electrónico: denismembrive2@gmail.com

Datos Académicos:

Formación Universitaria: En curso.

Carrera: Tecnicatura Universitaria en Programación.

Establecimiento: Universidad Tecnológica Nacional

2. DATOS DE LA EMPRESA



Razón Social: Asistencias Médicas S.R.L

C.U.I.T.: 30-71383090-5

Domicilio: Manuel Ruibal este 187

Provincia: Mendoza

Localidad: Malargüe

Fecha de contrato social: 2013-03-04

Representante Legal: Sofía Inés Gettor Sartini

Actividad:

- SERVICIOS DE EMERGENCIAS Y TRASLADOS.
- SERVICIOS DE EMERGENCIAS Y TRASLADOS.
- SALUD HUMANA Y SERVICIOS SOCIALES.

3. PROPUESTA PRACTICA PROFESIONAL

Asistencias Médicas SRL nos planteó la necesidad de digitalizar la empresa. Debido a inconvenientes laborales que habían tenido anteriormente, se perdió su sitio web y así su posicionamiento en los motores de búsqueda del navegador.

La propuesta de nosotros como estudiantes de Programación fue la siguiente:

- Desarrollo de sitio web.

Con respecto a los servicios brindados por la empresa, se planteó el desarrollo de una landing page responsive, que pueda adaptarse a cualquier dispositivo para su visualización. En la misma se incorporarían como portada principal una foto de un móvil representativo de la empresa, tendría su barra de navegación para acceder a las distintas secciones de la página. En sus secciones se presentarían sus servicios de forma interactiva, comentarios de clientes, preguntas frecuentes, contacto y por último un footer con el mismo nav bar del principio y los derechos reservados como marca de la empresa.

- Desarrollo App Mobile.

Fue de nuestro interés brindarle la posibilidad a la empresa de contar con una App para que sus clientes pudieran tener la posibilidad de contactarse y solicitar un traslado desde la misma independientemente del sitio de San Rafael en que se encuentren. Esta App estaría desarrollada para dispositivos Android y si bien no estaría disponible en Google playStore por el momento, si se podría descargar en un futuro cercano, su .apk desde github para sus correspondientes pruebas y testeos por parte de los usuarios.

- Administración de Base de Datos.

En este caso la empresa nos plasmó una problemática con respecto al software de gestión que utilizan, el mismo es Shamman Express perteneciente a la empresa Smart Medical Tech S.A., empresa líder en soluciones tecnológicas para la salud. Existía la necesidad de migrar los datos de un servidor Local a otro mediante SQL Server Management Studio (SSMS). Esto requería trasladar la base de datos de un Computador a otro (Este es el servidor local) y realizar la configuración Remota del nuevo Server para poder acceder a él desde cualquier ordenador mediante las credenciales brindadas por la empresa que provee el software de Gestión (Shamman Express).

4. ACTIVIDADES REALIZADAS

a. Digitalización de la empresa

En primera instancia se entrevistó a la Gerente General de la Empresa, Sofía Inés Gettor Sartini con el motivo de plasmar en una idea los servicios que realizaríamos para la empresa. Como Programadores les sugerimos la realización de un sitio web personal de la empresa, donde los usuarios y potenciales clientes pudieran acceder a él desde cualquier navegador web y con cualquier dispositivo móvil o de escritorio, y pudiesen tener en él, una excelente experiencia de usuario, pudiendo interactuar con el mismo de forma interactiva como si se estuviese en la misma oficina de la empresa con un representante de ventas.

También la posibilidad que los ya clientes fidelizados pudiesen contactarse vía web, por medio de su correo personal, whatsapp o cualquier medio digital que les pareciese más conveniente.

Para saber dónde estábamos parados, se sugirió a la empresa realizar una pequeña campaña de Marketing dentro de las redes sociales ya existentes, segmentando el público al que queríamos llegar.

Los resultados fueron los esperados por la empresa y de esta manera a nosotros nos daría las herramientas para el desarrollo interactivo del sitio ajustándolo así a las necesidades del rango etario a quien iba dirigido especialmente.

Logramos desarrollar un Producto que se adapta a las necesidades de los más jóvenes, brindándole una experiencia interactiva y una estética adecuada, pero también a su vez, la interfaz es simple y condice con las limitaciones que existen en el rango etario mayor, haciendo simple su navegación y adaptable a cualquier dispositivo móvil en el que se quisiera acceder al mismo.

De la misma manera y ya teniendo los datos de Marketing brindados por la empresa se sugirió crear una App Android para que sus clientes pudieran acceder a ciertos servicios prestados por la empresa desde sus propios dispositivos móviles.

Los Servicios que presta la empresa que están disponibles para su uso en la aplicación son, traslados urbanos y códigos rojos (Donde corre peligro la vida del cliente). De esta manera se generara una mayor eficacia para los afiliados a la hora de solicitar un servicio.

Con respecto al sitio web, fue contratado un servicio de Hosting, en este caso Hostinger, donde se alojó el mismo con su propio dominio para que pudieran visualizarse desde cualquier navegador web.

La App como se mencionó anteriormente, está en fase de pruebas y testeos y no será incorporada por el momento a google playStore. Pero si se puede observar desde gitHub.

b. Desarrollo página web

Para realizar el desarrollo del sitio web se utilizó el entorno de desarrollo Visual Studio Code (VSC) y se empleo el Lenguaje de Programación JavaScript para realizar el backend y funcionalidad de la misma, Html como esqueleto o lenguaje estructural de la pagina y Css para dar los diferentes estilos a los contenedores utilizados en cada etiqueta de Html. También se hizo uso de librerías como Bootstrap y Fontello para agregar interactividad a la página y hacerla full Responsive.

A continuación se muestran imágenes del mismo, luego el link para acceder a su dominio (alojado por el momento en gitHub Page) y también a su repositorio. Al final un link que nos redirigirá a una carpeta de drive donde se encuentra un breve video explicativo del sitio para entender su funcionalidad y su código.



TE OFRECEMOS

Emergencias a Domicilio

contrata ahora

Areas Protegidas

contrata ahora

Traslados

contrata ahora

← **Fernando Calani**
Servicio de emergencia

Exdenete Servicio. Empresa Responsable y los profesionales 10 puntos.



Preguntas Frecuentes

Preguntas Frecuentes

Pregunta 1



Pregunta 2

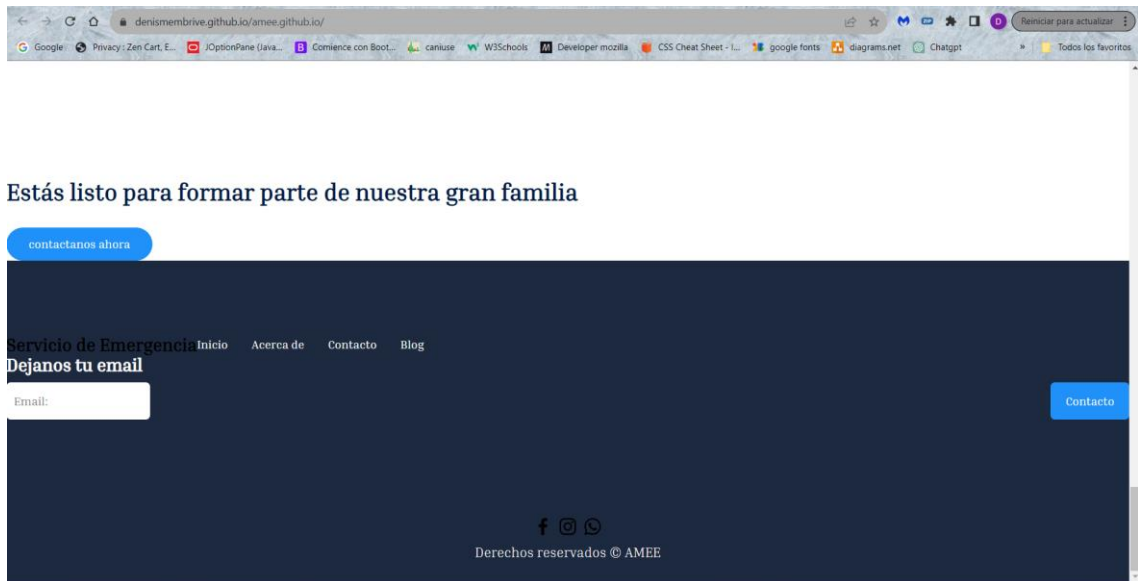


Pregunta 3



Pregunta 4





Link Pagina web

<https://denismembrive.github.io/amee.github.io/>

Link Repositorio

<https://github.com/denismembrive/amee.github.io>

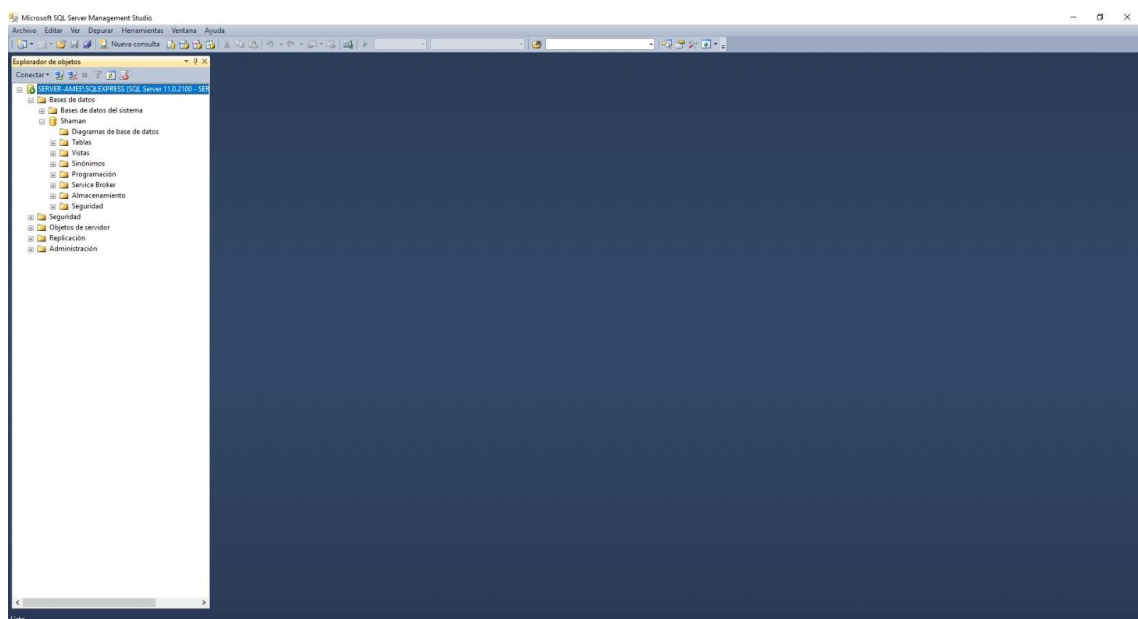
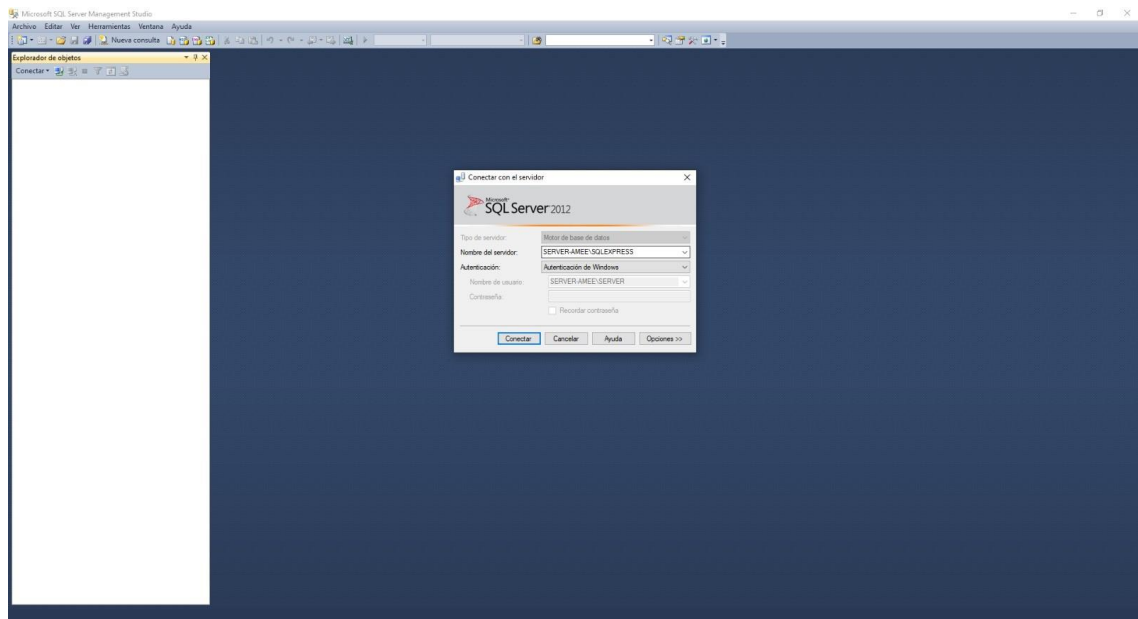
Link Video demostrativo

https://drive.google.com/drive/folders/1g_NprJrCSd6G8gMKjo2CJ8tG_Y7e-j1X?usp=sharing

c. Administración base de datos SQL vinculada a software Shamman Express.

Como primer tarea se realizó la migración de los datos de un server utilizado (viejo computador) a otro (nuevo computador). Este procedimiento se realizó mediante el uso del software SQL Server Management Studio (SSMS). Por medio de las credenciales de usuario brindadas por la empresa, se accedió con el uso del software a la base de datos y se extrajeron los datos de clientes.

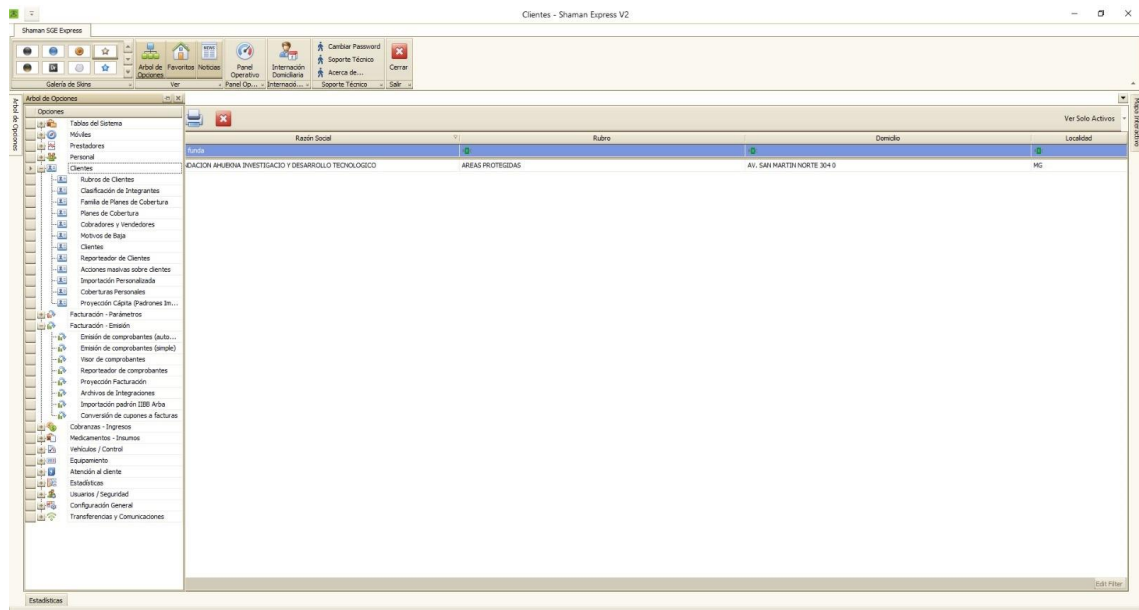
Por razones comerciales y de seguridad no se muestran datos de los clientes.



Una vez realizada esta tarea, se instaló en el nuevo ordenador el software SQL Server Management Studio. Se crearon los usuarios correspondientes y se realizó la migración a esta nueva base de datos.

Desde la empresa del software Shamman se colaboró conjuntamente con soporte técnico para vincular la base de datos a este software de gestión. Cabe destacar que esta tarea es realizada por la empresa Smart Medical Tech S.A , ya que Asistencias Médicas SRL hace el pago de un canon mensual por soporte Técnico.

En la siguiente imagen vemos el software de gestión Shamman Express instalado y funcionando correctamente en el nuevo ordenador.



d. Desarrollo App Mobile

De acuerdo a los datos recopilados por el área de marketing de la empresa se consiguió desarrollar una App móvil pensada para sistemas operativos Android, por ser el mismo el más utilizado en Mendoza, de acuerdo a estudios pertinentes realizados por la Cámara Argentina de Software.

El concepto de sistema operativo como algo que primordialmente presenta a sus usuarios una interfaz cómoda, es una perspectiva descendente (es decir de arriba hacia abajo). Desde una perspectiva alterna, la ascendente, el sistema operativo tiene como misión administrar todos los elementos de un sistema complejo. Las computadoras modernas constan de procesadores, memorias, temporizadores, discos, puertos de E/S, interfaces de red y una amplia gama de otros dispositivos. En la perspectiva alterna, la tarea del sistema operativo, consiste en efectuar un reparto ordenado y controlado de los procesadores, memorias y dispositivos de E/S, entre los diversos programas que compiten por obtenerlos.

Android es un sistema operativo inicialmente pensado para teléfonos móviles, al igual que iOS, Symbian y BlackBerry. Lo que lo hace diferente es que está basado en Linux, un núcleo de sistema operativo libre, gratuito y multiplataforma.

Para este trabajo se utilizó el entorno de desarrollo integrado (IDE) ANDROID STUDIO, ya que el mismo cuenta con las herramientas necesarias para el diseño y desarrollo de aplicaciones en el sistema operativo Android, de manera que en este software, es posible

elaborar en conjunto el diseño estético de la aplicación y los protocolos de comunicación con el servidor.

Para el diseño de la aplicación se tuvo en cuenta lo siguiente:

- **ARQUITECTURA:**

A la hora de llevar a cabo un desarrollo, es de vital importancia realizar una estructura del código ordenada y que éste sea agrupado según diferentes características específicas. Por ejemplo puede resultar de utilizar realizar una agrupación que incluya lo necesario para presentar la interfaz, otra que una toda la gestión de acceso a una fuente de datos, una tercera que tome unos datos y los manipule según las necesidades existentes. Una adecuada estructura del código aporta muchas ventajas , como por ejemplo; permite una mejor comprensión del código y un aprendizaje más rápido de su manejo, facilita el mantenimiento y actualización de la aplicación, mejora el proceso de testeo del software al organizar el código en función de sus características, etc.

- **DISEÑO INTERFAZ DE USUARIO:**

Al crear la interfaz del usuario se tuvo en cuenta el estudio realizado de marketing, segmentado a los posibles usuarios, ya que, muchas veces ellos son los que juzgan el sistema por su interfaz más que por su funcionalidad, por lo tanto se tuvieron en cuenta los siguientes principios para crear esta interfaz.

Familiaridad: utilizar términos que sean conocidos a los usuarios

Consistencia: los menús con el mismo formato y significado en toda la aplicación

Mínima sorpresa: Misma acción en contextos comparables produzcan efectos

Comparables.

Recuperabilidad: Permitir la recuperación frente a errores cometidos por el usuario.

- **MANTENIBILIDAD Y PORTABILIDAD:**

1. Disponibilidad para todo tipo de dispositivo Android con este sistema operativo

2. La aplicación Android esta soportada para versiones 7.0 y superiores abarcando la mayoría de dispositivos móviles.

- INTERFAZ Y USABILIDAD

1. La aplicación consta de una interfaz sencilla, atractiva e intuitiva. De tal forma que su uso no suponga un impedimento o esfuerzo al usuario a la hora de hacer uso de la aplicación.

2. La introducción de datos está estructurada procurando evitar errores.

- RENDIMIENTO:

1. Tiempos de respuesta no superiores a un segundo en las peticiones al servidor y menores en las consultas a la base de datos.

2. Tanto los accesos a base de datos como algún cálculo que se realice en la aplicación no supone demasiada carga para el dispositivo, por lo que el rendimiento es óptimo.

- BASE DE DATOS:

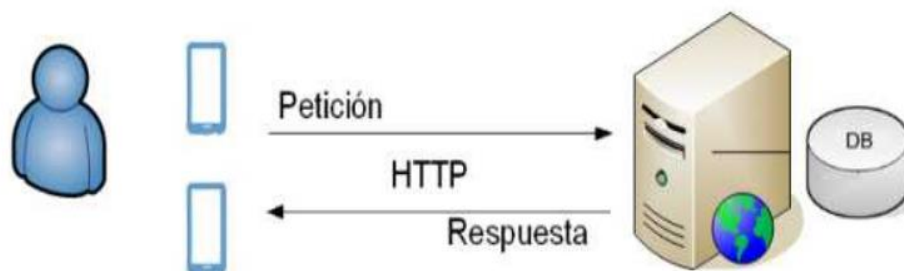
(EN FASE DE DESARROLLO)

La base de datos no puede ser local en el propio dispositivo, por lo que se implementa una base de datos externa situada en el servidor y a la que se accede a través de VISUAL STUDIO.

La base de datos almacenará toda la información de usuarios.

- ARQUITECTURA DE LA APLICACION

Como se menciona anteriormente, la arquitectura de la aplicación se basa en el modelo cliente servidor. En el caso de este proyecto, y como puede verse en la siguiente figura, la aplicación móvil y desde cualquier dispositivo electrónico constituirá la parte cliente, y la base de datos estará alojada en un servidor al que se accederá mediante llamadas http.



- TECNOLOGIA UTILIZADA:

En este caso se utilizó el lenguaje de programación Kotlin. Al ser un lenguaje basado en JAVA nos dio la posibilidad de tener una sintaxis similar a uno de los lenguajes vistos en la carrera. De esta manera nos fue más fácil aprender de él y utilizarlo debidamente.

El entorno de desarrollo es ANDORID STUDIO, este software nos permitió llevar un desarrollo ordenado de la App debido a que sus componentes (Layout, Activity, Manifest, Service, Local storage and Database, conexión API REST) están ordenados en distintos package y acceder a ellos individualmente nos dio el ordenamiento y la posibilidad de cometer menos errores en el desarrollo de la misma.

CONCLUSION:

La arquitectura del proyecto se basa en el modelo cliente-servidor, donde el cliente será la Aplicación móvil, y el servidor una base de datos.

Durante todo el proyecto, se han ido cambiando y retocando partes que ya estaban implementadas debido a diversos factores.

El proyecto se encuentra alojado en gitHub, en el link que se mostrara a continuación y como se dijo anteriormente, está en fase de pruebas y pronto estará disponible para su descarga y su uso por parte de los clientes de la empresa de manera real.

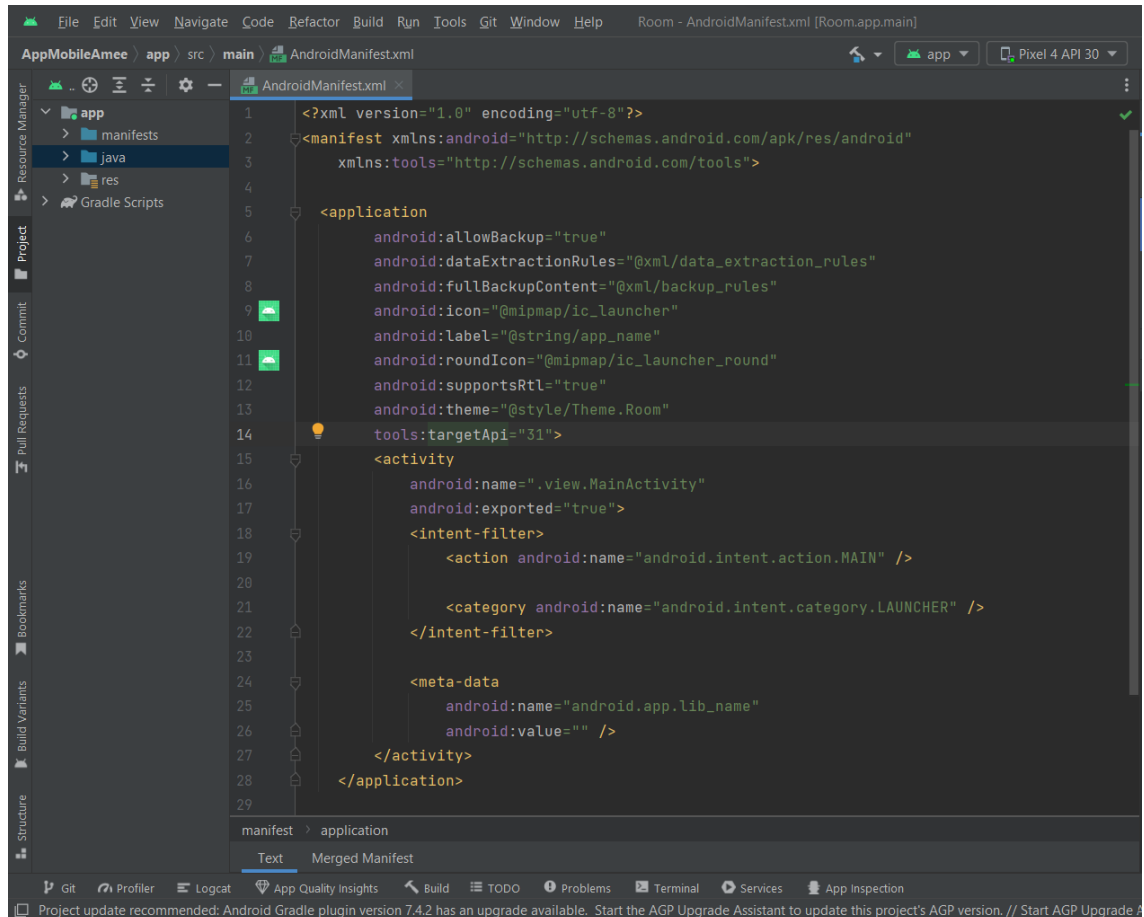
Link repositorio gitHub

<https://github.com/denismembrive/AppMobileAmees>

Link Video demostrativo

https://drive.google.com/drive/folders/1g_NprJrCSd6G8qMKjo2CJ8tG_Y7e-j1X?usp=sharing

Compartimos fotos dentro del entorno de desarrollo integrado (Android studio) del código y de la emulación de la App dentro del mismo entorno.



```
1 package com.curso.android.app.practica.counter.view
2
3 import android.os.Bundle
4 import androidx.activity.viewModels
5 import androidx.appcompat.app.AppCompatActivity
6 import com.curso.android.app.practica.counter.databinding.ActivityMainBinding
7
8 class MainActivity : AppCompatActivity() {
9
10     private lateinit var binding: ActivityMainBinding
11     private val mainViewModel: MainViewModel by viewModels()
12
13     override fun onCreate(savedInstanceState: Bundle?) {
14         super.onCreate(savedInstanceState)
15         binding = ActivityMainBinding.inflate(layoutInflater)
16         setContentView(binding.root)
17
18         mainViewModel.counter.observe(owner: this) { it: Counter?
19             println("Recibimos un nuevo valor de counter. $it")
20             binding.counter.text = "${it.number}"
21             binding.timestamp.text = "Actualizado ${it.timestamp}"
22         }
23
24         binding.incrementButton.setOnClickListener { it: View?
25             mainViewModel.incrementCounter()
26         }
27         binding.decrementButton.setOnClickListener { it: View?
28             mainViewModel.decrementCounter()
29         }
30     }
31 }
```

Project update recommended: Android Gradle plugin version 7.4.2 has an upgrade available. Start the AGP Upgrade Assistant to update this project's AGP version. // Start AGP Upgrade Assistant

```
1 package com.curso.android.app.practica.counter.view
2
3 import androidx.lifecycle.ViewModel
4 import androidx.lifecycle.MutableLiveData
5
6 class MainViewModel: ViewModel() {
7
8     // Solo queremos que se pueda leer el contador
9     private var _counter: MutableLiveData<Counter> = MutableLiveData<Counter>(Counter(0, Date()))
10
11     fun incrementCounter() {
12         val next = (_counter.value?.number ?: 0) + 1
13         updateCounter(next)
14     }
15
16     fun decrementCounter() {
17         val next = (_counter.value?.number ?: 0) - 1
18         updateCounter(next)
19     }
20
21     private fun updateCounter(next: Int) {
22         viewModelScope.launch { this: CoroutineScope
23             _counter.value = Counter(next, Date())
24         }
25     }
26 }
```

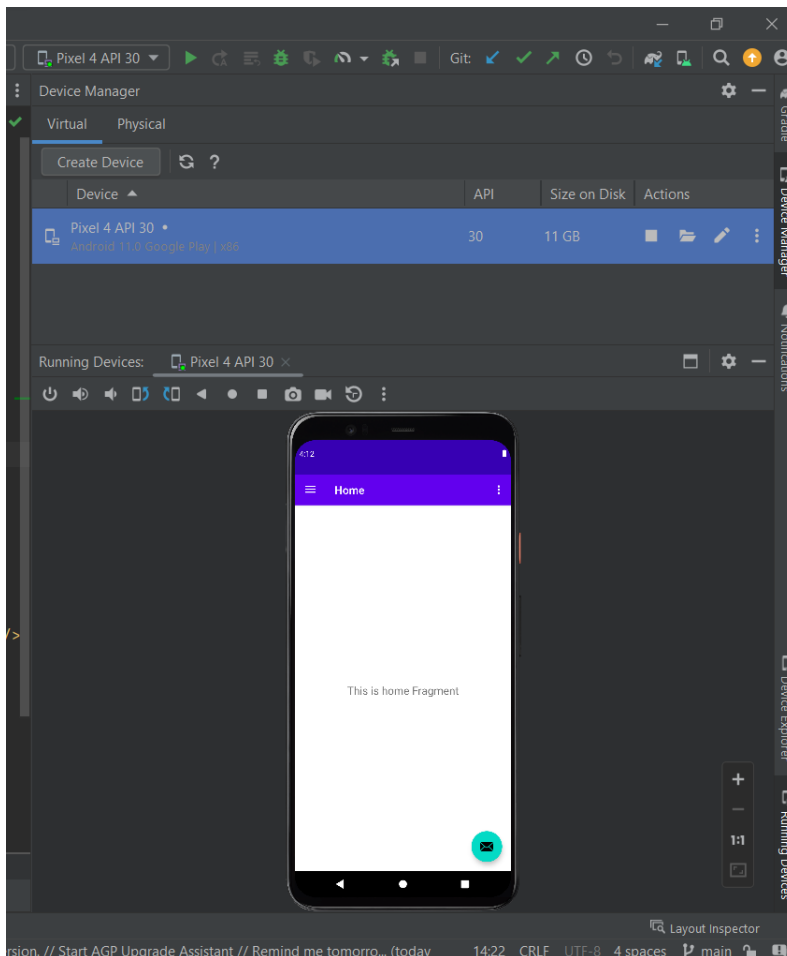
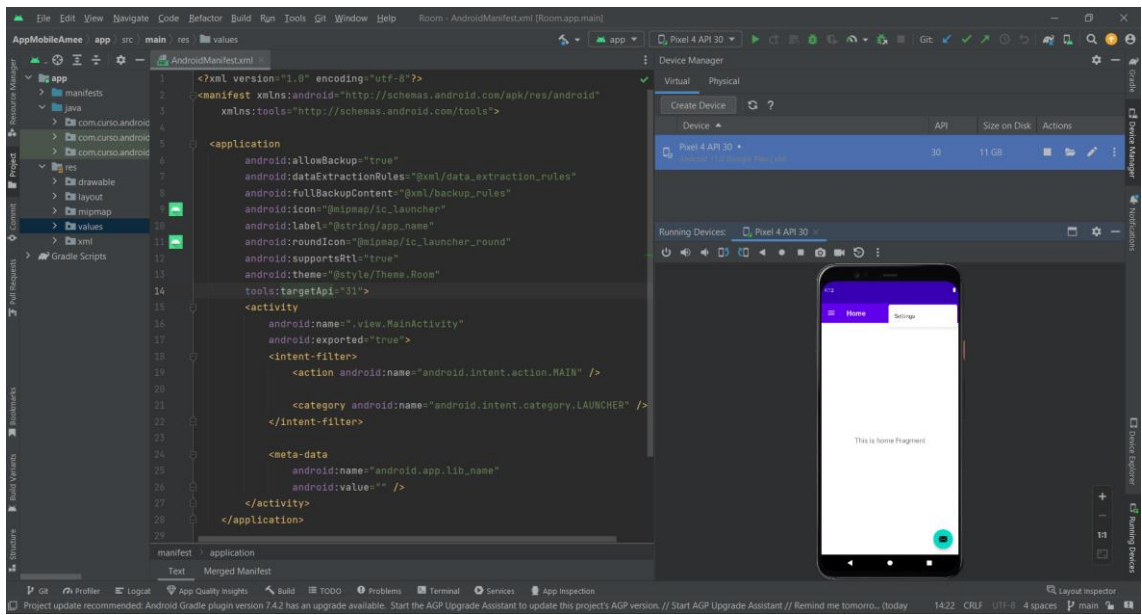
Project update recommended: Android Gradle plugin version 7.4.2 has an upgrade available. Start the AGP Upgrade Assistant to update this project's AGP version. // Start AGP Upgrade Assistant

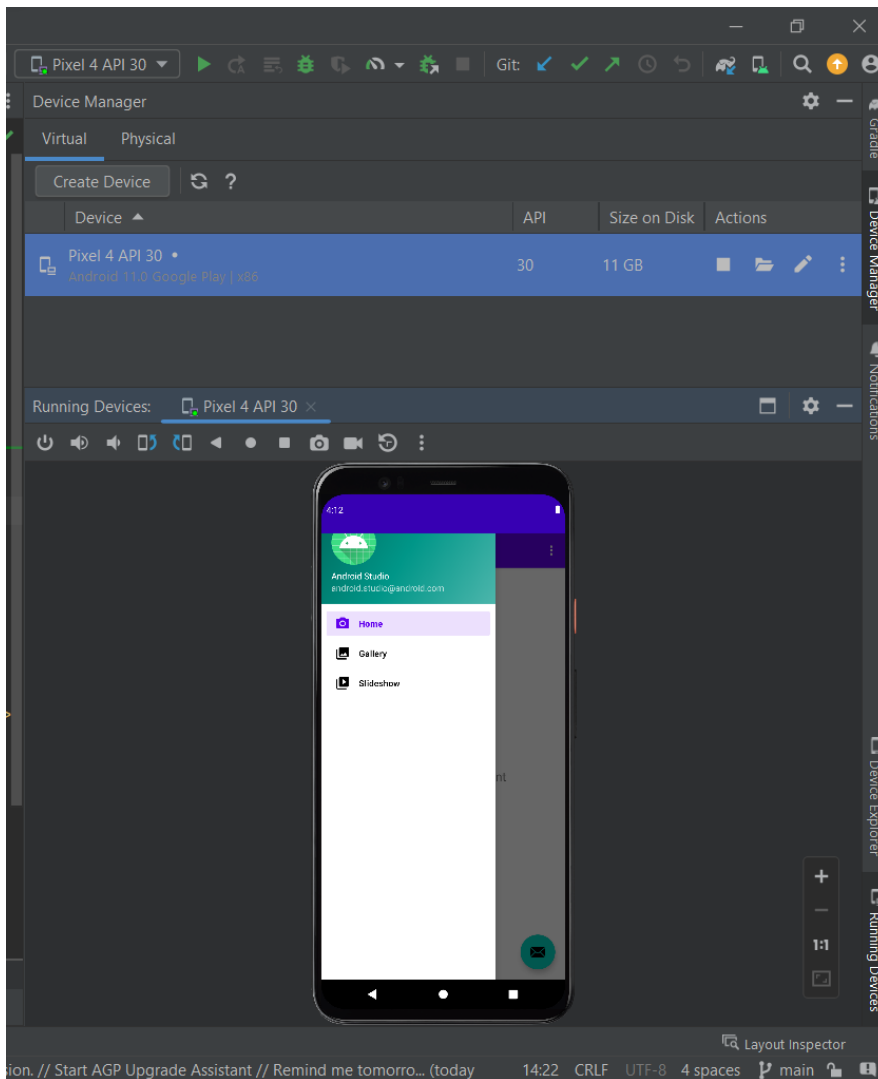

```
Room - MainActivityTest.kt [Room.app.androidTest]
File Edit View Navigate Code Refactor Build Run Tools Git Window Help
AppMobileAmees > app > src > androidTest > java > com > curso > android > app > practica > counter > view > MainActivityTest
AndroidManifest.xml x MainActivityTest.kt x
1 package com.curso.android.app.practica.counter.view
2
3 import ...
4
5 @RunWith(AndroidJUnit4::class)
6 class MainActivityTest {
7
8     @get: Rule
9     var rule: ActivityScenarioRule<*> = ActivityScenarioRule(MainActivity::class.java)
10
11     @Before
12     fun setUp() {
13
14     }
15
16     @After
17     fun tearDown() {
18
19     }
20
21     @Test
22     fun mainActivity_incrementCounter() {
23         Espresso.onView(
24             ViewMatchers.withId(R.id.increment_button)
25         ).perform(
26             ViewActions.click()
27         )
28
29         Espresso.onView(
```

Project update recommended: Android Gradle plugin version 7.4.2 has an upgrade available. Start the AGP Upgrade Assistant to update this project's AGP version. // Start AGP Upgrade Assistant

```
Room - MainViewModelUnitTest.kt [Room.app.unitTest]
File Edit View Navigate Code Refactor Build Run Tools Git Window Help
AppMobileAmees > app > src > test > java > com > curso > android > app > practica > counter > MainViewModelUnitTest
AndroidManifest.xml x MainViewModelUnitTest.kt x activity_main.xml x
1 package com.curso.android.app.practica.counter
2
3 import ...
4
5 /**
6  * Example local unit test, which will execute on the development machine (host).
7  *
8  * See [testing documentation](http://d.android.com/tools/testing).
9  */
10 class MainViewModelUnitTest {
11
12     private lateinit var viewModel: MainViewModel
13
14     @get: Rule
15     val instantTaskRule = InstantTaskExecutorRule()
16     private val dispatcher = StandardTestDispatcher()
17
18     @Before
19     fun setUp() {
20         Dispatchers.setMain(dispatcher)
21         viewModel = MainViewModel()
22     }
23
24     @After
25     fun tearDown() {
26         Dispatchers.resetMain()
27     }
28 }
```

Project update recommended: Android Gradle plugin version 7.4.2 has an upgrade available. Start the AGP Upgrade Assistant to update this project's AGP version. // Start AGP Upgrade Assistant





Denis Andrés Membrive

35663945

Sofía Inés Gettor Sartini

Asistencias Médicas S.R.L.

Gerente General