



Sprint Dos

Rol	Integrante	E-mail
Líder del equipo/Tester	Gina Katherine Torres Vega	Kathevega03@gmail.com
Gestor BD	Milena Duarte Gómez	amilenadurte@gmail.com
Desarrollador Frontend	Johann Sebastián Monroy	johannmonroy@hotmail.com
Desarrollador Backend	Alejandro Chiviri	Kairos7071@gmail.com

Proyecto

Mascota Feliz

Grupo 60

Equipo número 4 / Fuerza Delta

Profesor

Carlos Julio Fadul

De acuerdo con los entregables solicitados, nosotros el equipo Fuerza Delta hacemos entrega de nuestro SCRUM.

I. Repositorio (GitHub) donde integrarán el trabajo.

<https://github.com/KatheVega/MascotaFeliz>

En este se encontrará toda la documentación del proyecto.

II. Implementación de la lógica de negocio

- Creacion de entidades en la capa de Dominio (contiene las entidades) y correcto funcionamiento

```
using System;
namespace MascotaFeliz.App.Dominio
{
    public class Dueno:Persona
    {
        public string Correo {get;set;}
    }
}
```

```
using System;
using System.Collections.Generic;
namespace MascotaFeliz.App.Dominio
{
    public class Historia
    {
        public int Id {get;set;}
        public DateTime FechaInicial {get;set;}
        public List<VisitaPyP> VisitasPyP {get;set;}
    }
}
```

PS C:\Mascotas\MascotaFeliz.App> cd MascotaFeliz.App.Dominio
PS C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Dominio> dotnet build
Microsoft (R) Build Engine versión 16.7.2+b60ddb6f4 para .NET
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Determinando los proyectos que se van a restaurar...
Todos los proyectos están actualizados para la restauración.
MascotaFeliz.App.Dominio -> C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Dominio\bin\Debug\netstandard2.0\MascotaFeliz.App.Dominio.dll

Compilación correcta.
0 Advertencia(s)
0 Errores

Tiempo transcurrido 00:00:05.52
PS C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Dominio>

```
using System;
namespace MascotaFeliz.App.Dominio
{
    public class Persona
    {
        public int Id {get;set;}
        public string Nombres {get;set;}
        public string Apellidos {get;set;}
        public string Direccion {get;set;}
        public string Telefono {get;set;}
    }
}
```

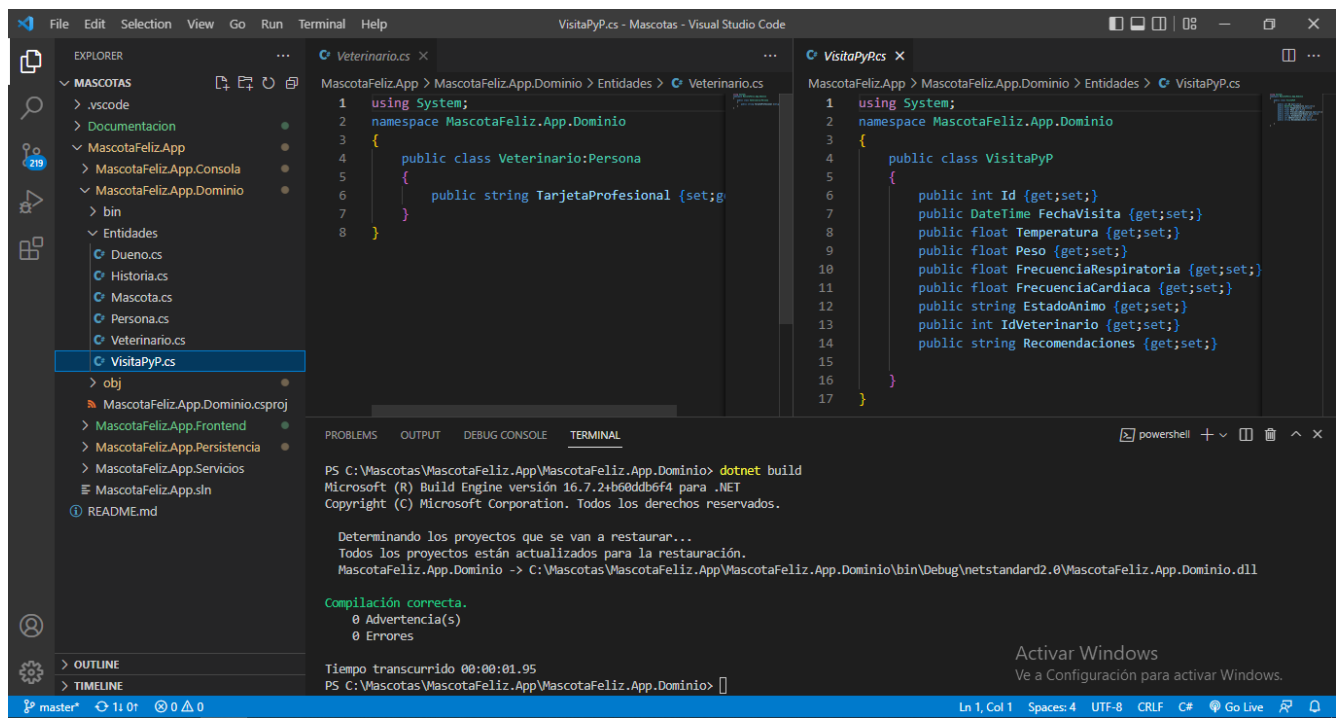
```
using System;
namespace MascotaFeliz.App.Dominio
{
    public class Mascota
    {
        public int Id {get;set;}
        public string Nombre {get;set;}
        public string Color {get;set;}
        public string Especie {get;set;}
        public string Raza {get;set;}
        public Dueno Dueno {get;set;}
        public Veterinario Veterinario {get;set;}
        public Historia Historia {get;set;}
    }
}
```

PS C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Dominio> dotnet build
Microsoft (R) Build Engine versión 16.7.2+b60ddb6f4 para .NET
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

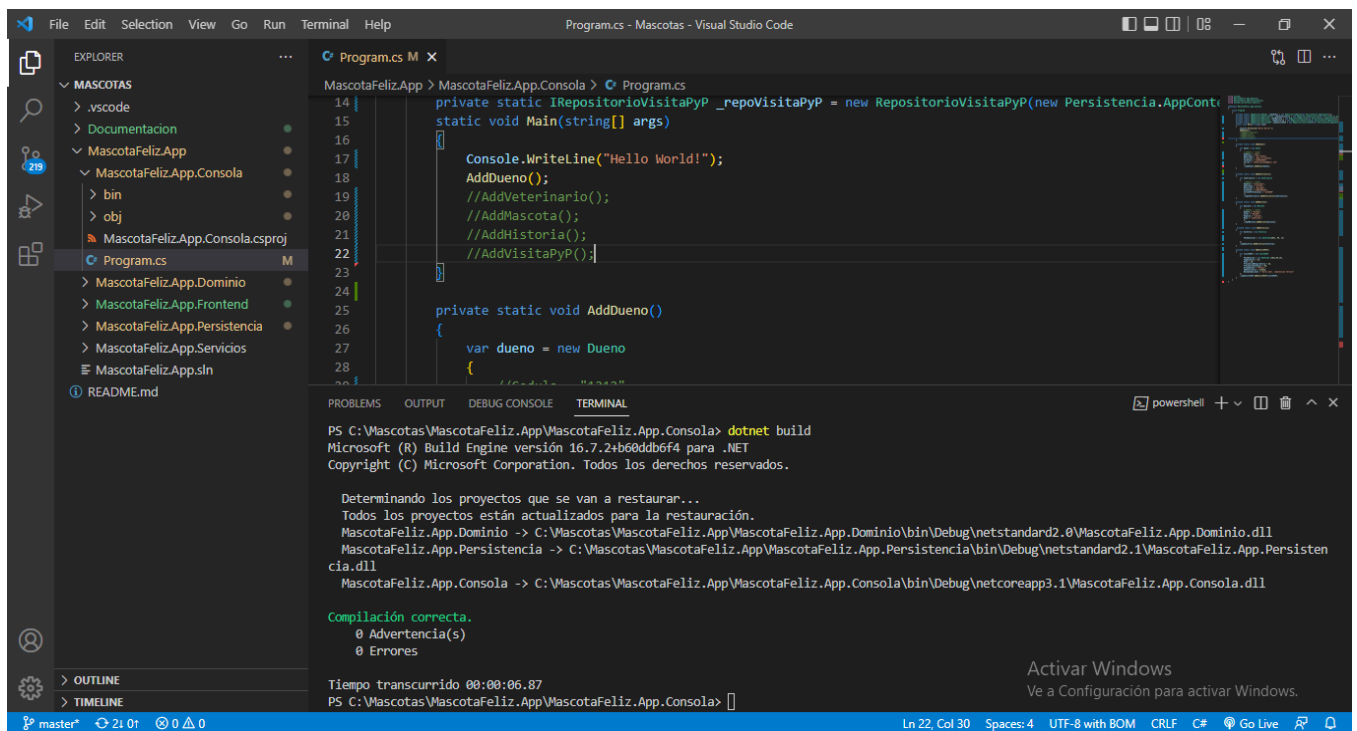
Determinando los proyectos que se van a restaurar...
Todos los proyectos están actualizados para la restauración.
MascotaFeliz.App.Dominio -> C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Dominio\bin\Debug\netstandard2.0\MascotaFeliz.App.Dominio.dll

Compilación correcta.
0 Advertencia(s)
0 Errores

Tiempo transcurrido 00:00:01.95
PS C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Dominio>



- En la Capa de Consola se crea un archivo Programs.cs el cual contiene el main con sus métodos, tenemos funcionando agregar en todas las clases:



- En la Capa de Persistencia creamos el archivo AppContext el cual hace la conexión de las clases con persistencia-base de datos-Frontend:

```

1 using Microsoft.EntityFrameworkCore;
2 using MascotaFeliz.App.Dominio;
3
4 namespace MascotaFeliz.App.Persistencia
5 {
6     public class AppContext : DbContext
7     {
8         public DbSet<Persona> Personas {get;set;}
9         public DbSet<Veterinario> Veterinarios {get;set;}
10        public DbSet<Dueno> Duenos {get;set;}
11        public DbSet<VisitaPyP> VisitasPyP {get;set;}
12        public DbSet<Historia> Historias {get;set;}
13        public DbSet<Mascota> Mascotas {get;set;}
14
15        protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)
16        {
17        }
18    }
19 }

```

Terminal Output:

```

PS C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Persistencia> dotnet build
Microsoft (R) Build Engine versión 16.7.2+b60ddb6f4 para .NET
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Determinando los proyectos que se van a restaurar...
Todos los proyectos están actualizados para la restauración.
MascotaFeliz.App.Dominio -> C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Dominio\bin\Debug\netstandard2.0\MascotaFeliz.App.Dominio.dll
MascotaFeliz.App.Persistencia -> C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Persistencia\bin\Debug\netstandard2.1\MascotaFeliz.App.Persistencia.dll

Compilación correcta.
0 Advertencia(s)
0 Errores

Tiempo transcurrido 00:00:01.13
PS C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Persistencia>

```

- Se crea los archivos que contienen los Repositorios y IRepository de cada clase:

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using MascotaFeliz.App.Dominio;
5 using Microsoft.EntityFrameworkCore;
6
7 namespace MascotaFeliz.App.Persistencia
8 {
9     public interface IRepositoryDueno
10    {
11        IEnumerable<Dueno> GetAllDuenos();
12        Dueno AddDueno(Dueno dueno);
13        Dueno UpdateDueno(Dueno dueno);
14        void DeleteDueno(int idDueno);
15        Dueno GetDueno(int idDueno);
16
17        IEnumerable<Dueno> GetDuenosPorFiltro(string filtro);
18    }
19 }

```

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using MascotaFeliz.App.Dominio;
5 using Microsoft.EntityFrameworkCore;
6
7 namespace MascotaFeliz.App.Persistencia
8 {
9     public class RepositorioDueno : IRepositoryDueno
10    {
11        /// <summary>
12        /// Referencia al contexto de Dueno
13        /// </summary>
14        private readonly AppContext _appContext;
15
16        /// <summary>
17        /// Metodo Constructor Utiliza
18        /// Inyeccion de dependencias para indicar
19        /// </summary>
20        /// <param name="appContext"></param>
21        public RepositorioDueno(AppContext appContext)
22        {
23            _appContext = appContext;
24        }
25
26        public Dueno AddDueno(Dueno dueno)
27        {
28            var duenoAdicionado = _appContext.Duenos.Add(dueno);
29            _appContext.SaveChanges();
30            return duenoAdicionado.Entity;
31        }
32    }
33 }

```

- Se crea la capa Servicios ejecutada correctamente:

The screenshot shows the Visual Studio Code interface with the file explorer on the left displaying the project structure. The main editor shows the `Startup.cs` file in the `MascotaFeliz.App.Services` namespace. The terminal at the bottom shows the command `dotnet build` being executed, resulting in a successful compilation with no warnings or errors.

```

9 using Microsoft.Extensions.Configuration;
10 using Microsoft.Extensions.DependencyInjection;
11 using Microsoft.Extensions.Hosting;
12 using Microsoft.Extensions.Logging;
13
14 namespace MascotaFeliz.App.Services
15 {
16     public class Startup
17     {
18         public Startup(IConfiguration configuration)
19         {
20             Configuration = configuration;
21         }
22
23         public IConfiguration Configuration { get; }
24     }
25 }

```

Terminal Output:

```

PS C:\Mascotas\MascotaFeliz.App> cd MascotaFeliz.App.Services
PS C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Services> dotnet build
Microsoft (R) Build Engine versión 16.7.2+b60ddb6f4 para .NET
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

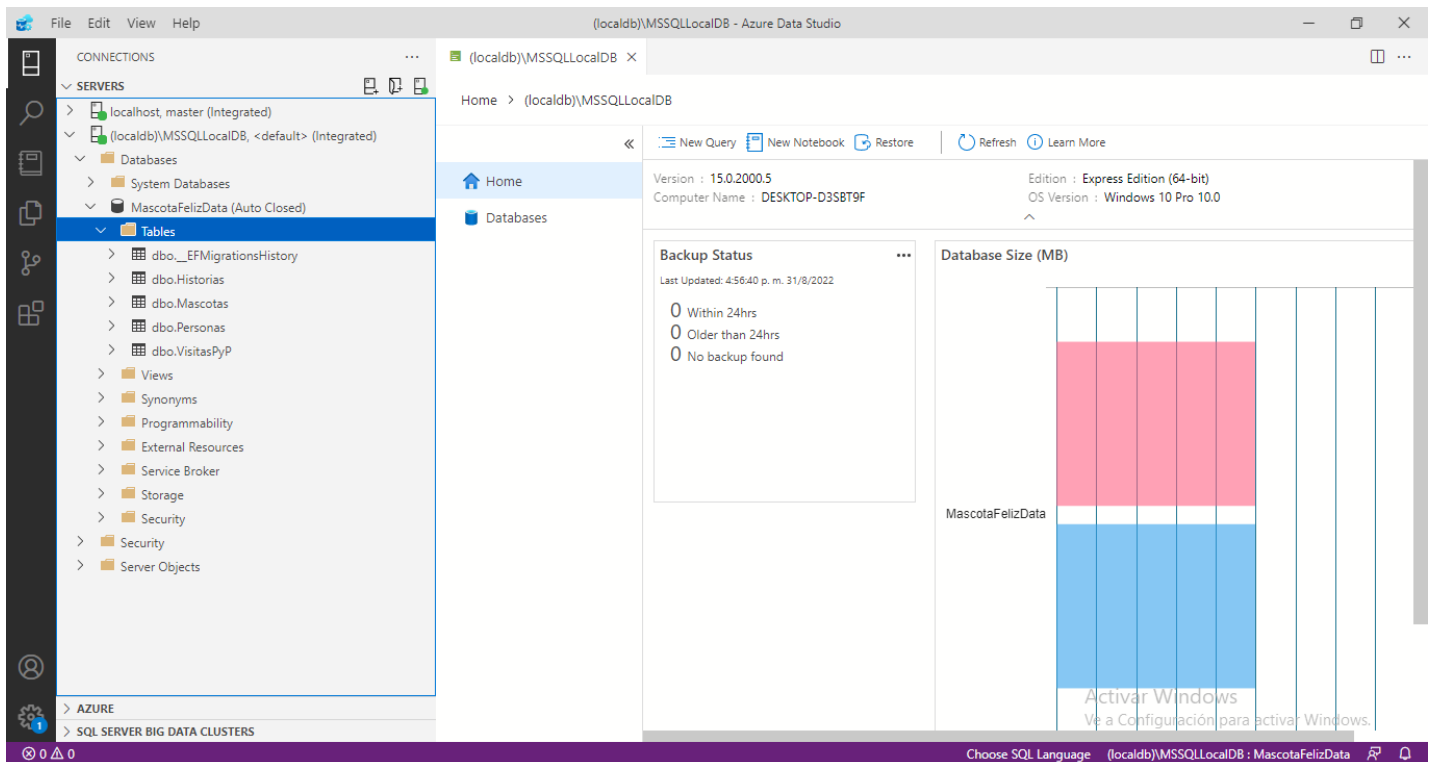
Determinando los proyectos que se van a restaurar...
Se ha restaurado C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Services\MascotaFeliz.App.Services.csproj (en 391 ms).
MascotaFeliz.App.Services -> C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Services\bin\Debug\netcoreapp3.1\MascotaFeliz.App.Services.dll

Compilación correcta.
0 Advertencia(s)
0 Errores

Tiempo transcurrido 00:00:07.55
PS C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Services>

```

- Se realizo la conexión de la base de datos



- Prueba Unitaria de Agregar Mascota, con conexión a BD

The screenshot shows the Visual Studio Code interface with the `Program.cs` file open. The `AddMascota` method is implemented, creating a new `Mascota` object with the following properties:

```

//Cedula = "1212",
Nombre = "Horus",
Color = "cafe",
Especie = "canino",
Raza = "Golden",

```

The `repoMascota.AddMascota(mascota);` line is highlighted. The terminal shows the command `dotnet run` and the output "Hello World!".

On the right, the SQL Query window shows the results of a query:

```

SELECT TOP (1000) [Id]
, [Nombre]
, [Color]
, [Especie]
, [Raza]
, [DuenoId]
, [VeterinarioId]
, [HistoriaId]
FROM [MascotaFelizData].[dbo].[Mascotas]

```

Id	Nombre	Color	Especie	Raza	DuenoId
1	Gaia	Negra	canina	siberiano	NULL
2	Sam	cafe	gatuno	angora	NULL
3	Horus	cafe	canino	Golden	NULL

- Prueba Unitaria de Agregar Veterinario, con conexión a BD

The screenshot shows the Visual Studio Code interface with the `Program.cs` file open. The `AddVeterinario` method is implemented, creating a new `Veterinario` object with the following properties:

```

//Cedula = "1212",
Nombres = "Alexander",
Apellidos = "Torres",
Direccion = "Duitama",
Telefono = "3105755335",
TarjetaProfesional = "ac45689",

```

The `repoVeterinario.AddVeterinario(veterinario);` line is highlighted. The terminal shows the command `dotnet run` and the output "Hello World!".

On the right, the SQL Query window shows the results of a query:

```

SELECT TOP (1000) [Id]
, [Nombres]
, [Apellidos]
, [Direccion]
, [Discriminator]
, [Telefono]
, [Correo]
, [TarjetaProfesional]
FROM [MascotaFelizData].[dbo].[Personas]

```

Id	Nombres	Apellidos	Direccion	Discriminator
1	Juan	Sin Miedo	Bajo un puente	Dueno
2	Gina	Torres	Calle 10a #19a 142	Veterinario
3	Alexander	Torres	Duitama	Veterinario

- Prueba Unitaria de Agregar Dueño, con conexión a BD

The screenshot displays the Visual Studio Code interface during a unit test execution. The Explorer on the left shows the project structure for 'MASCOTAS'. The center editor shows the 'Program.cs' file with a private static void AddDueno() method. The right pane shows a SQL query in a separate window, and the bottom pane shows the terminal output of a successful compilation and execution.

```

private static void AddDueno()
{
    var dueno = new Dueno
    {
        //Cedula = "1212",
        Nombres = "Katherine",
        Apellidos = "Vega",
        Direccion = "calle 2a #10b 92",
        Telefono = "1234567891",
        Correo = "Kathevega@gmail.com"
    };
    _repoDueno.AddDueno(dueno);
}

private static void AddVeterinario()
{
}

```

SQLQuery_1 - (localdb)\MSSQLLocalDB\MascotaFelizData

```

SELECT TOP (1000) [Id]
, [Nombres]
, [Apellidos]
, [Direccion]
, [Discriminator]
, [Telefono]
, [Correo]
, [TarjetaProfesional]
FROM [MascotaFelizData].[dbo].[Personas]

```

Id	Nombres	Apellidos	Direccion	Discriminator
1	Juan	Sin Miedo	Bajo un puente	Dueno
2	Gina	Torres	Calle 10a #19a ...	Veterinario
3	Alexander	Torres	Duitama	Veterinario
4	Katherine	Vega	calle 2a #10b 92	Dueno

ia.dll
MascotaFeliz.App.Console -> C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Console.dll
Compilación correcta.
0 Advertencia(s)
0 Errores
Tiempo transcurrido 00:00:01.39
PS C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Console> dotnet run
Hello World!
PS C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Console>

- Prueba Unitaria conexión Backend/Frontend/Bases de Datos

The screenshot displays the Visual Studio Code interface during a unit test execution. The Explorer on the left shows the project structure for 'MASCOTAS'. The center editor shows the 'Program.cs' file with a namespace MascotaFaeliz.App.Frontend and a public class Program. The right pane shows the terminal output of a successful compilation and execution.

```

namespace MascotaFaeliz.App.Frontend
{
    public class Program
    {
        public static void Main(string[] args)
        {
            CreateHostBuilder(args).Build().Run();
        }

        public static IHostBuilder CreateHostBuilder(string[] args) =>
        {
            Host.CreateDefaultBuilder(args)
                .ConfigureWebHostDefaults(webBuilder =>
                {
                    webBuilder.UseStartup<Startup>();
                });
        }
    }
}

```

Tiempo transcurrido 00:00:05.31
t run
info: Microsoft.Hosting.Lifetime[0]
Now listening on: https://localhost:5001
info: Microsoft.Hosting.Lifetime[0]
Now listening on: http://localhost:5000
info: Microsoft.Hosting.Lifetime[0]
Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
Content root path: C:\Mascotas\MascotaFeliz.App\MascotaFeliz.App.Frontend

- Prueba Unitaria conexión Lista dueños

MascotaFeliz.App.Frontend Home Lista de dueños Registrar de dueños Lista de mascotas Lista de veterinarios

hola

Nombres	Apellidos		
Juan	Sin Miedo	Detalle Dueno	Editar Dueno
Katherine	Vega	Detalle Dueno	Editar Dueno

- Prueba Unitaria conexión Lista mascotas

MascotaFeliz.App.Frontend Home Lista de dueños Registrar de dueños Lista de mascotas Lista de veterinarios

hola

Nombre	Color	Especie	Raza		
Gaia	Negra	canina	siberiano	Detalle Mascota	Editar Mascota
Sam	cafe	gatuno	angora	Detalle Mascota	Editar Mascota
Horus	cafe	canino	Golden	Detalle Mascota	Editar Mascota

- Prueba Unitaria conexión Lista veterinarios

MascotaFeliz.App.Frontend Home Lista de dueños Registrar de dueños Lista de mascotas Lista de veterinarios

hola

TarjetaProfesional		
gk10224	Detalle Veterinario	Editar Veterinario
ac45689	Detalle Veterinario	Editar Veterinario

IV. Informe de retrospectiva

Evidencia del acta, si se desea ver con detalle se encuentra en

<https://github.com/KatheVega/MascotaFeliz/tree/master/Documentacion/Actas/Acta02>

ArchivoHerramientasVista2.Restrospectiva_Sprint2 - Modo de compatibilidad • Guardado en Este PC

Fecha de la reunión:	15 de septiembre de 2022		
Hora de inicio:	20:00	Hora de fin:	21:00
Convocada por:	Johann Monroy (Scrum Master)		
Asunto:	Reunión de retrospectiva - Mascota Feliz		

CONVOCADOS / ASISTENTES

Nombre	Área / Proceso / Rol	Asistió (Si/No)
Gina Katherine torres vega (KV)	Lider	Si
Milena Duarte Gómez (MD)	Gestor BD	Si
Johan Sebastián Monroy (JM)	Desarrollador Frontend	Si
Alejandro Chiviri (AC)	Desarrollador Backend	Si
Gina Katherine torres vega (KV)	Tester	Si

Temas:

1. ¿Qué se hizo bien? / Nuevas best practices
2. ¿Qué se puede mejorar? / Plan de acciones de mejora
3. ¿Qué se puede intentar? / Acuerdos de equipo actualizados
4. Compromisos / Impedimentos a escalar

Tema 1: ¿Qué se hizo bien?

Resumen

- Entrada a las reuniones y comunicación constante por WhatsApp
- Asignación de tareas y roles dentro del Sprint
- Manejo de tiempos y cumplimientos
- Lluvia de ideas de visualizaciones y decisiones entre todo el equipo
- Colaboración constante en temas fáciles para unos, complejos para otros.

Tema 2: ¿Qué se puede mejorar?

Resumen

- Mayor participación en las decisiones del equipo
- Más dinamismo en las reuniones
- Ser más proactivos individualmente

Tema 3: ¿Qué se puede intentar?

Resumen

- Acoplamiento entre el frontend y el backend
- Asistir de manera más constante a clases
- Mejorar la disciplina y dinamismo individual y colectivo

Tema 4: Compromisos.

Resumen

- Estudiar como se desarrolla en HTML y CSS
- Estudiar e investigar como acoplar el Backend con el Frontend

En muestra de acuerdo con los puntos descritos en la presente acta, firman:

Katherine Vega (Scrum Máster)
Milena Duarte (Product owner)
Johan Monroy (Development Team)
Alejandro Chiviri (Development Team)

Final del documento

V. Historias de usuario desarrolladas en el Sprint 2.

Este documento se encuentra completo en la siguiente ruta

<https://github.com/KatheVega/MascotaFeliz/tree/master/Documentacion>

github.com/users/KatheVega/projects/1/views/1

KatheVega / Projects / @KatheVega's untitled project

Search or jump to...

@KatheVega's untitled project

View 1 + New view

Filter by keyword or by field

No Status 6

MascotaFeliz #16
Sprint 3 / Asignar un veterinario a una mascota

MascotaFeliz #17
Sprint 3 / Actualizar datos de dueño, mascota, veterinario

MascotaFeliz #18
Sprint 3 / Registrar visitas

MascotaFeliz #19
Sprint 3 / Consultar listado de mascotas por veterinario

+ Add item

Todo 1

MascotaFeliz #15
Sprint 2 / Formulario Visita HTML

+ Add item

In Progress 4

MascotaFeliz #11
Sprint 2 / Formulario Registrar mascotas HTML

MascotaFeliz #12
Sprint 2 / Formulario Registro consultar mascotas HTML

MascotaFeliz #13
Sprint 2 / Formulario Registro propietario HTML

MascotaFeliz #14
Sprint 2 / Formulario Registro veterinario

+ Add item

Done 9

MascotaFeliz #9
Sprint 1 / creación de la base de datos

MascotaFeliz #10
Sprint 1 / formato de historias de usuario

MascotaFeliz #6
Sprint 1 / tablero Kanban en Github

MascotaFeliz #7
Sprint 1 / implementación de la capa de dominio

+ Add item

Activar Windows
Ve a Configuración para activar Windows.

HISTORIAS DE USUARIO			
ID de referencia:	4	Nombre:	Formulario Inicio de sesion
Usuario:	Desarrollador		
Modifica la historia:	N/A	Sprint:	2/4
Prioridad:	Alta	PF estimados:	3
Riesgos de desarrollo:	Medio	PF reales:	4
Descripción: Yo como veterinario necesito acceder a la aplicación para poder ingresar los datos de la visita realizada a la mascota, esto con el fin de tener seguridad y solo ingrese el personal autorizado.			
Observaciones: En este formulario se debe visualizar un campo de usuario y contraseña para garantizar seguridad, boton para su respectivo ingreso.			

VI. Historias de usuario a desarrollar en el sprint 3.

Este documento se encuentra completo en la siguiente ruta

<https://github.com/KatheVega/MascotaFeliz/tree/master/Documentacion>

HISTORIAS DE USUARIO			
ID de referencia:	4	Nombre:	Asignar un veterinario a una mascota
Usuario:	Desarrollador		
Modifica la historia:	N/A	Sprint:	3/4
Prioridad:	Alta	PF estimados:	5
Riesgos de desarrollo:	Medio	PF reales:	5
Descripción: Yo como veterinario necesito identificar el veterinario que atendio a la mascota.			
Observaciones:			