



```
private CompanyRepository $repository;  
private CompanyEventDispatcher $dispatcher;
```

```
public function __construct(  
    CompanyRepository $repository,  
    CompanyEventDispatcher $dispatcher
```

```
    ) {  
        $this->repository = $repository;  
        $this->dispatcher = $dispatcher;
```

```
    }  
  
    public function __invoke(DTO $dto): DTO
```

```
    {  
        /** @var CreateCompanyDTO $dto */
```

```
        $namespace = $dto->namespace();  
        if($this->repository->namespaceExists($namespace))  
            throw new AlreadyExistsNamespace($namespace);  
    }  
  
    $name = $dto->name();  
    $emailAddress = $dto->emailAddress();  
  
    $company = Company::create($namespace, $name, $emailAddress);  
  
    $this->repository->create($company);  
    $this->dispatcher->created($dto);
```

Samuel Fernández Amorós

TFG – Desarrollo web

FamilyCare Online Application

Tutor: Gregorio Robles Martínez
Junio 2020

Índice

1. Introducción
2. Tecnologías
3. Planificación
4. Requisitos funcionales
5. Requisitos no funcionales
6. Diseño guiado por el dominio
7. Diseño punto vista empresa
8. Diseño punto vista familiar
9. Arquitectura hexagonal
10. Pruebas o test.
11. Puesta a punto de servidores
12. Infraestructura Cloud
13. Despliegue en producción
14. Resultado
15. Conclusiones
16. Agradecimientos



Introducción

- Creación de un CRM diferente.
- Enfoque especial en el área del cuidado de personas.
- Enfoque del desarrollo guiado por el dominio (DDD).
- Construcción de un MVP.
- Diseño visualmente atractivo y sencillo de usar.



Tecnologías

PHP, Symfony, MySQL, Redis, Docker, Git, Ansible, Terraform, Bootstrap, Composer



Planificación temporal

- (29/02 – 08/03) Requisitos
- (09/03 – 28/05) Desarrollo
- (29/02 – 28/05) Documentación
- (29/02 – 12/06) Memoria

Principales requisitos funcionales

Contexto de gestión de empresa

Registro de empresas

Edición de Gestores

“ de trabajadores

“ de clientes

“ de familiares

“ de categorías

Sistema de identificación

Contexto de gestión del familiar

Ficha del familiar

Envío de emails

Edición de notas

Identificación del cliente

Calendario de actividades





Requisitos no funcionales

Información separada



La información de cada empresa se encuentra separada físicamente por bases de datos.

Contraseñas no legibles



Las contraseñas de los usuarios no deben poder ser legibles y se deben guardar cifradas.

Sistema de despliegue



Se debe usar una herramienta para facilitar el despliegue de nuevas versiones en producción.

Detectar el idioma



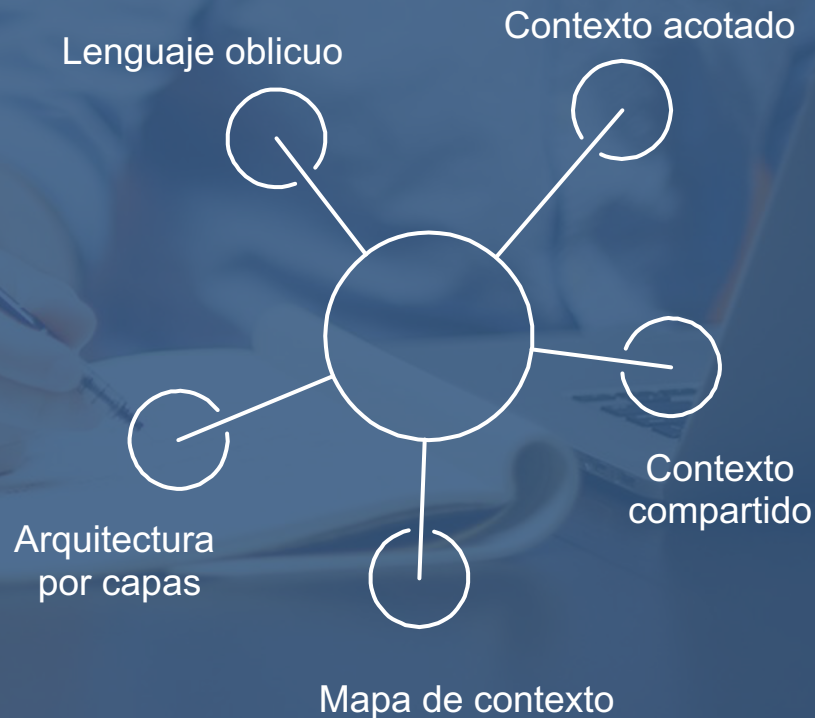
El sistema debe ser capaz de detectar el idioma del usuario.

Diseño guiado por el dominio

Siguiendo el enfoque de organización de la información se detectan dos contextos:

- Gestión de empresa.
- Familiares.

Aplicamos la parte de diseño estratégica y la parte táctica.



Diseño desde el punto de vista de la empresa



- Actores:
Gestor, trabajador, cliente y familiar.



- Casos de uso:
Registro de empresa. Edición de gestor, cliente, familiar y trabajador. Vinculación de familiar y cliente. Edición de actividad,



- Objetos de valor:
Dirección de email, Contraseña, DNI, etc.

Diseño desde el punto de vista del familiar



- **Actores:**
Gestor, trabajador, cliente y familiar.

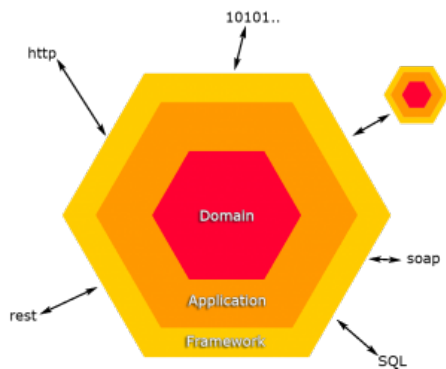


- **Casos de uso:**
Ver calendario de actividades, crear una nota, enviar un email, apuntar familiar a una actividad.



- **Objetos de valor:**
Dirección de email, Contraseña, DNI, etc.

Arquitectura hexagonal o de puertos y adaptadores



Fuente: <https://laraveles.com/introduccion-arquitectura-hexagonal/>

Capa de infraestructura

Capa de aplicación

Capa de dominio

Pruebas unitarias / Integración / Funcionales

Validamos casos de uso

- Estados normales.
- Límites de los casos
- Situaciones invalidas.



| | Normal | Limite | Invalido |
|---------------------|--------|--------|----------|
| Crear actividad | ✓ | | ✓ |
| Registro de empresa | ✓ | ✓ | ✓ |
| Enviar un email | ✓ | | |

Entorno online

Puesta a punto de servidores

- Configuración de VM
- Configuración de DNS
- Registro de dominio

www.samuel-fa-uoc.cf



Google Cloud Platform



CLOUDFLARE

Máquina
virtual

DNS

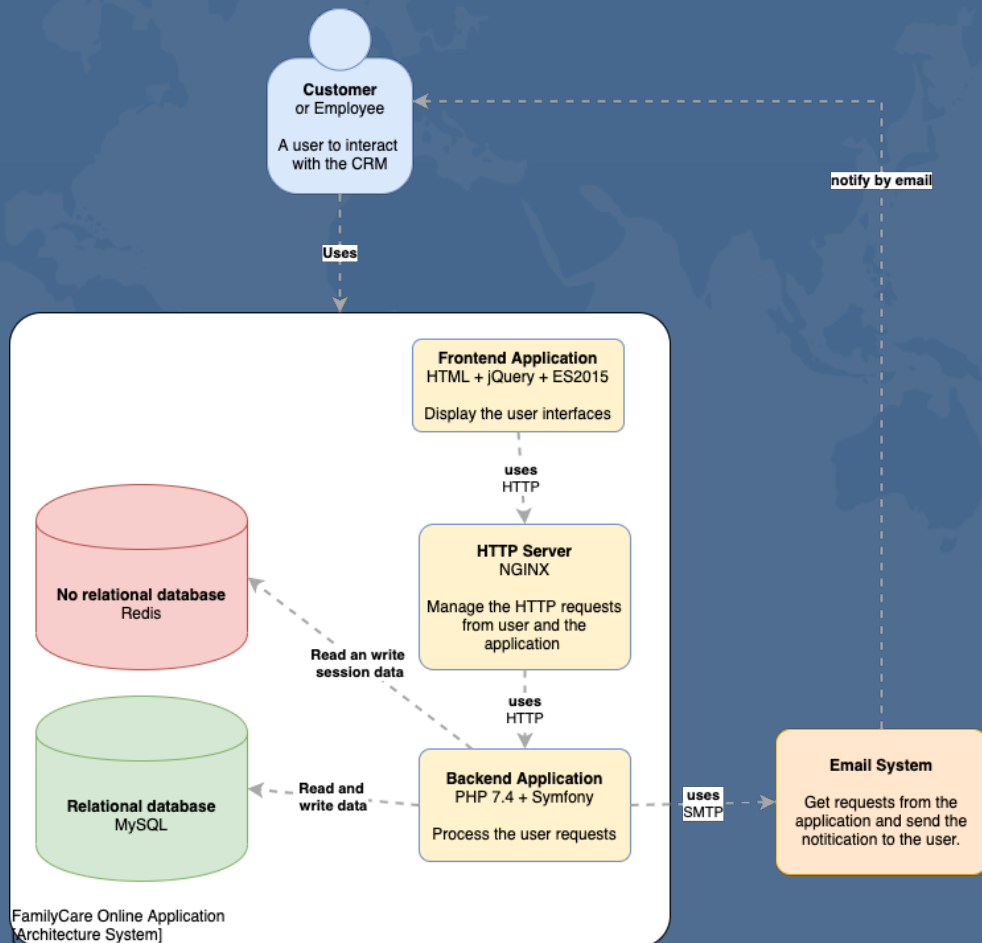
Dominio

freenom

Infraestructura Cloud



Google Cloud Platform





ANSIBLE



Despliegue en producción

Usamos Ansistrano.

- Es una extensión de Ansible.
- Despliegue de nuevas versiones del proyecto.
- Recuperar versiones anteriores en caso de errores.

Usamos Ansible.

- Instalación del software en los servidores.
- Configuración del software en los servidores.

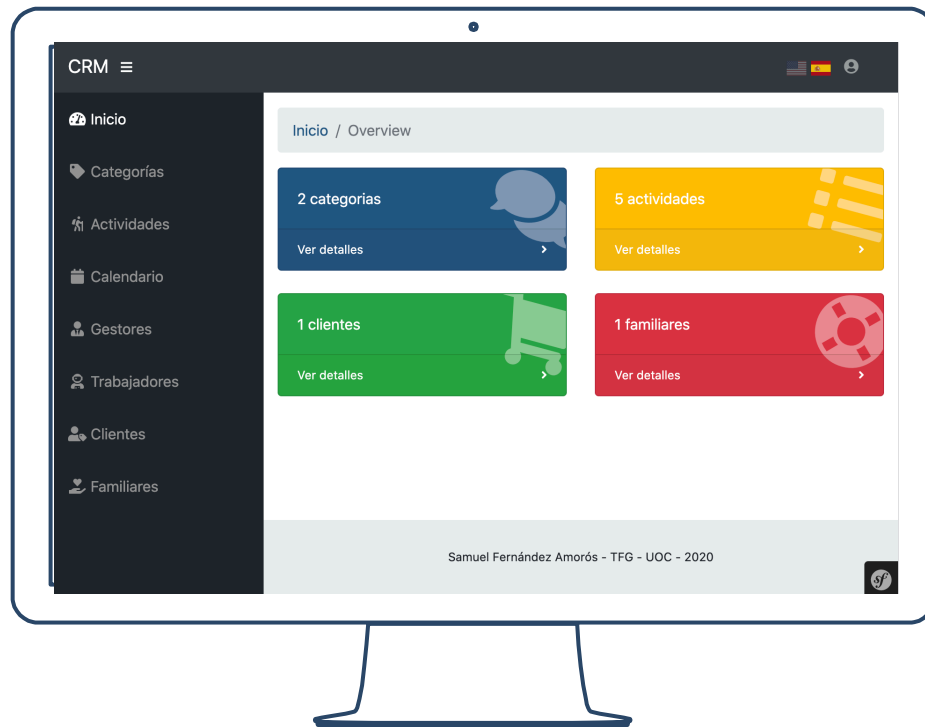
Usamos Terraform.

- Gestionar maquinas virtuales.
- Gestionar la red.
- Gestionar la seguridad.

Resultado

Se puede probar el proyecto en:

www.samuel-fa-uoc.cf



Conclusiones

MVP

Creación de un Proyecto que puede seguir **creciendo** en cuanto al número de funcionalidades.



Objetivos

Se han **cumplido** los **objetivos** iniciales en los **plazos** propuestos.



Experiencia

Desarrollo del proyecto pasando por diferentes etapas, desde el **diseño**, la **programación** hasta el **despliegue**.



Futuro

Futuros pasos del proyecto podrían ser estudiar la **viabilidad** en el **mercado** empresarial.





¡Gracias!



Llego el momento de las preguntas,
enviarme vuestras preguntas a:
samuelfa@uoc.edu

Agradecimientos

Mi pareja, a la familia, amigos, compañeros de trabajo, profesores, consultores, tutor y a mi perrita Pei. Sin vuestro apoyo no hubiera podido llegar hasta aquí.