



IES JOSÉ RODRIGO BOTET

PROYECTO DE DESARROLLO DE APLICACIONES WEB

APLICACIÓN WEB PARA EL NOMBRAMIENTO DE TAREAS O SERVICIOS DE BASE/COMPAÑÍA

NOMBRE: Germán Orlando Navarro Díaz

DNI: 50516754Z

NIA: 11395657

CURSO: 2018/2019

CONVOCATORIA: Ordinaria



TRIBUNAL

Fdo. PRESIDENTE:

Fdo. TUTOR INDIVIDUAL:

Fdo.TUTOR COLECTIVO:

En Manises, a de de 2019.

NOTA FINAL:

INDICE

Tabla de contenido

1. INTRODUCCIÓN	6
2. HERRAMIENTAS UTILIZADAS	9
2.1. LENGUAJES EMPLEADOS.....	9
2.1.1. PHP (Hypertext Preprocessor).....	9
2.1.2. JavaScript	9
2.1.3. HTML (HyperText Markup Language)	11
2.1.4. CSS (Cascading Style Sheets)	11
2.1.5. MySQL.....	12
2.2. FRAMEWORKS	13
2.2.1. Laravel (PHP)	13
2.2.2. Bootstrap (CSS).....	15
2.3. LA BASE DE DATOS	15
3. ANÁLISIS DE REQUISITOS	18
3.1. PLANTEAMIENTO DE NECESIDADES.....	18
3.2. QUIÉN USARÁ DE LA APLICACIÓN. CASOS DE USO.....	19
4. LA APLICACIÓN WEB	21
4.1. DESCRIPCIÓN	21
4.2. INTERACCIÓN LENGUAJES	21
4.3. CONSTITUCIÓN	22
4.4. PLANTILLAS DE VISUALIZACIÓN.....	25
4.5. PÁGINAS	28
4.5.1. Asignación De Tareas.....	28

4.5.2. Configuración.....	32
4.5.3. Personal.....	34
4.5.4. Ausencias	36
5. CONCLUSIONES.....	39
6. TRABAJOS FUTUROS.	41
7. BIBLIOGRAFÍA.	43
8. MANUAL INSTALACIÓN/CONFIGURACIÓN/USO.	44
8.1. INFORMACIÓN GENERAL.....	44
8.2. INSTALACIÓN DE MYSQL	44
8.3. INSTALACIÓN DE LARAVEL	45
8.4. INICIAR EL SERVIDOR DE PHP	45
8.5. LOGIN	46
8.6. ASIGNACIÓN DE TAREAS.....	47
8.6.1. Borrar tarea asignada	48
8.7. CONFIGURACIÓN	48
8.7.1. Nuevas de tareas, empleos,	48
8.7.2. Relación tarea-empleo.....	50
8.8. PERSONAL.....	51
8.8.1. Nuevo usuario.....	52
8.8.2. Listado de personal fuera de servicio	52
8.8.3. Ascender a usuario de empleo.	53
8.8.4. Editar un usuario.....	54
8.9. AUSENCIAS.....	55
8.9.1. Nueva ausencia	56
8.9.2. Borrar Ausencia	57

INDICE DE ILUSTRACIONES

ILUSTRACIÓN 1, RELACIONES DE LA BASE DE DATOS	16
ILUSTRACIÓN 2, CASOS DE USO DEL USUARIO.	19
ILUSTRACIÓN 3, CASOS DE USO DEL ADMINISTRADOR	20
ILUSTRACIÓN 4, INTERACCIÓN DE LOS LENGUAJES.....	21
ILUSTRACIÓN 5, PÁGINA DE LOGIN.....	22
ILUSTRACIÓN 6, MENÚ DE USUARIO, PANTALLA INICIAL	22
ILUSTRACIÓN 7, TODAS LAS TAREAS	23
ILUSTRACIÓN 8, MENÚ ADMINISTRADOR, PANTALLA INICIAL	23
ILUSTRACIÓN 9, PLANTILLA BASE	25
ILUSTRACIÓN 10, COMPLEMENTO PLANTILLA BASE	26
ILUSTRACIÓN 11, COMPLEMENTO 2 PLANTILLA BASE	26
ILUSTRACIÓN 12, DIAGRAMA PARA ASIGNAR TAREAS	28
ILUSTRACIÓN 13, ASIGNACIÓN DE TAREAS.....	29
ILUSTRACIÓN 14, CÓMO SE OBTIENE EL USUARIO Y NÚMERO DE CONTEO DE TAREA	31
ILUSTRACIÓN 15, TAREAS ASIGNADAS	32
ILUSTRACIÓN 16, DIAGRAMA NUEVO EMPLEO O TAREA	32
ILUSTRACIÓN 17, CONFIGURACIÓN DE TAREAS, EMPLEO, RELACIÓN TAREA- EMPLEO	33
ILUSTRACIÓN 18, EDITAR, BORRAR ELEMENTO DE LA TABLA EMPLEO	33
ILUSTRACIÓN 19, CONFIGURACIÓN TAREA-EMPLEO	34
ILUSTRACIÓN 20, TABLA DE PERSONAL.....	34
ILUSTRACIÓN 21, INFORMACIÓN PERSONAL DE USUARIO	35
ILUSTRACIÓN 22, AUSENCIAS.....	36
ILUSTRACIÓN 23, NUEVA AUSENCIA.....	37
ILUSTRACIÓN 24, TAREAS ASIGNADAS	37
ILUSTRACIÓN 25, EJEMPLO DE AUSENCIA POR BAJA MÉDICA.....	38
ILUSTRACIÓN 26, TAREA REASIGNADA DESPUÉS DE AGREGAR A UN USUARIO COMO AUSENTE	38

1. INTRODUCCIÓN

NOMBRAMIENTO DE SERVICIOS DE BASE/COMPAÑÍA

Desde que existen las fuerzas militares en España ha existido la necesidad de asignar tareas cotidianas o de seguridad al personal militar para su realización. Tareas como la limpieza de espacios habitables, de vehículos, de armamento, etc., al igual que guardias de seguridad o guardias de orden.

En un principio, probablemente, esas tareas eran asignadas por el jefe encargado, a su criterio, o por un subordinado con la autoridad para ello.

Conforme avanzan los tiempos se podría decir que esta asignación de tareas ahora se lleva a cabo de un modo más metódico, siguiendo una serie de criterios y normas preestablecidas bajo un marco legal y organizacional. Algunos de esos criterios a tener en cuenta son la antigüedad en el empleo; si la persona estará ausente el día que le corresponde hacer el servicio ya sea por causas médicas o por estar en un ejercicio, de maniobras, reducción de jornada, etc.

A fecha de hoy, el tiempo normalmente invertido por la persona o personas encargadas de la asignación de las tareas suele ser superior a una hora teniendo en cuenta que el número de personas, el tipo de unidad y el acuartelamiento, puede oscilar entre 10 y 100 personas o más. Este tiempo puede duplicarse o triplicarse debido a la gran cantidad de variables a tener en cuenta cuando se aproximan festividades.

El objetivo principal de este proyecto consiste en desarrollar una herramienta que permita la asignación de dichas tareas empleando el menor tiempo posible. Para ello se propone el uso de una base de datos relacional, a la cual, a través de una página web, se le proporciona la información necesaria con el uso de tecnologías web como MYSQL, PHP, JavaScript, AJAX o JQUERY.

Constaría de una página principal a la cual sólo se accedería mediante usuario y contraseña para garantizar la seguridad de la información. Luego en la página

principal se visualizaría un menú que muestre las opciones dependiendo del tipo de usuario logueado.

Mediante el uso de una base de datos relacional de MySQL se almacenaría toda la información necesaria. Información de los usuarios, los roles, las tareas a realizar, los empleos, las ausencias, además de las relaciones de todos los elementos entre sí.

Para administrar dicha información se implementaría como lenguaje principal de desarrollo PHP en el entorno de trabajo (framework) Laravel, el cual proporciona herramientas y funciones predefinidas que permiten ahorrar tiempo de trabajo.

Es indiscutible que un factor de gran peso cuando se desarrolla una aplicación que trata información puede ser cómo se manipula dicha información y si lo que el usuario solicita es lo que obtiene de un modo preciso, sin embargo, cuando esa información llegue al usuario final no es representa ningún inconveniente que la información que se reciba se vea atractiva, bien organizada, con colores que ayuden a la lectura e interpretación de lo que se tiene en pantalla. Es por ello que para mostrar la información de esta aplicación se usaría Bootstrap, un framework basado en CSS que da formato de un modo llamativo y uniforme a los elementos HTML de las páginas web; y también JavaScript, para enriquecer aún más la funcionalidad del front-end (todo lo que vemos en una página web y cómo se presenta).

La función más importante que debe llevar a cabo esta aplicación es la de asignar tareas a usuarios. Para ello se crearía un controlador dedicado exclusivamente para esta función, que a su vez será alimentado por funciones más pequeñas, por ejemplo que hacen consultas sobre el personal que está ausente, que está ocupado haciendo otras tareas, o que no están presentes porque han terminado una tarea que indica que a su culminación la persona puede marcharse a casa y volver a trabajar al día siguiente a su jornada laboral de modo ordinario.

Además de asignar tareas serían necesarias otras funciones para optimizar el rendimiento y la usabilidad de la aplicación, tales como; agregar a las personas que están ausentes ya se por cuestiones de salud, porque están haciendo un curso, etc; guardar, editar o borrar la información de las personas; ascender en el empleo a

alguien si fuese necesario; configurar información básica como las tareas, los empleos y las relaciones entre ellas; entre muchas más funciones.

También se crearán funciones en JavaScript usando la librería de JQuery para dar interacción a algunas páginas, por ejemplo que se muestren u oculten botones cuando suceda un evento. DataTables sería otra librería de apoyo para mostrar las tablas que contienen la información y poder trabajar con ellas sin solicitar información del servidor, es decir, ordenarlas por columnas, filtrar datos, etc.

A pesar de que la intención, al momento de diseñar esta aplicación, se proyecte para ser usada en España, también existe la posibilidad de colaborar a nivel internacional para mejorarla o adaptarla. Por ello, todos los nombres de las variables, funciones, tablas, comentarios, así como el de todos los elementos en esta aplicación en la parte del código estarían en inglés.

2. HERRAMIENTAS UTILIZADAS

El desarrollo de esta aplicación se ha realizado haciendo uso de diferentes lenguajes de programación teniendo en cuenta que cada uno posee unas características específicas que le dan valor gracias a las mismas. Unos son ejecutados en el cliente y otros en el servidor.

2.1. LENGUAJES EMPLEADOS.

2.1.1. PHP (Hypertext Preprocessor)

PHP es un lenguaje de script (instrucciones) que se ejecuta en el servidor. Es de código abierto (open source). Se utiliza para desarrollar páginas web estáticas o dinámicas, así como aplicaciones web. PHP es sinónimo de preprocesador de hipertexto, que anteriormente significaba páginas de inicio personales (Personal Home Page). Los scripts PHP solo se pueden interpretar en un servidor que tiene PHP instalado.

PHP es el lenguaje principal usado como base para toda esta aplicación web.

2.1.2. JavaScript

Es un lenguaje de programación interpretado principalmente usado en desarrollo web. Fue desarrollado con el propósito de dar dinamismo e interactividad a los elementos de las páginas web. Está basado en ECMAScript, otro lenguaje de programación desarrollado por Sun Microsystems.

A diferencia de PHP, JavaScript se ejecuta en el cliente y no en el servidor. Esto significa que se pueden ejecutar funciones en el cliente una vez la página ha cargado sin necesidad de que exista comunicación con un servidor. Esto permite a las páginas reaccionar a eventos, exhibir efectos especiales, aceptar texto variable, validar datos, crear cookies, detectar el navegador de un usuario.

Al igual que PHP, JavaScript se puede insertar en el código HTML.

JavaScript fue usado en esta aplicación web para mostrar advertencias, usando Ajax, y JQuery.

2.1.2.1 AJAX (Asynchronous JavaScript And XML)

Siglas de: JavaScript y XML Asíncrono. Ajax no es un lenguaje de programación. Es una nueva técnica para crear mejores, más rápidas, y más interactivas las aplicaciones web con la ayuda de XML, JSON, HTML, CSS, y JavaScript.

Con Ajax cuando se envía un formulario JavaScript realiza la solicitud al servidor, interpreta los resultados, y actualiza la pantalla sin recargar la página actual necesariamente. Todo sucede en segundo plano, sin que el usuario lo note.

Entre las características de Ajax se encuentran:

- Actualizar una página web sin volver a cargar la página.
- Solicitar datos de un servidor - después de que la página se haya cargado
- Recibir datos de un servidor - después de que la página se haya cargado
- Enviar datos a un servidor - en segundo plano.

Ajax fue usado en esta aplicación web para solicitar al servidor un listado de los “empleos” asignados a cada tarea en el momento que se selecciona una tarea.

2.1.2.2 JQuery

Es una librería de JavaScript cuyo principal propósito es simplificar la manipulación de: los elementos de una página web (DOM), las llamadas a servidor usando Ajax, CSS, y el manejo de eventos.

JQuery fue usado en esta aplicación web para modificar elementos del DOM agregando u ocultando botones, mostrando advertencias, haciendo peticiones con Ajax, etc.

2.1.2.3 DataTables

Es una librería que usa JQuery cuyo propósito es facilitar la manipulación, dar formato y añadir algunas características avanzadas a las tablas HTML.

2.1.2.4 SweetAlert

Es una librería que reemplaza la función “*alert*” de JavaScript para mostrarlas notificaciones de un modo más llamativo o elegante.

2.1.3. HTML (HyperText Markup Language)

Lenguaje de Marcas de Hipertexto. Utilizado para el desarrollo de páginas web. Los elementos HTML son la estructura más básica de una página web. Se representan con una etiqueta, por ejemplo: `body`, `table`. La mayoría de las etiquetas tiene una etiqueta inicial y una etiqueta de cierre, ej.: `<body> contenido </body>`; algunas etiquetas no requieren etiqueta de cierre, ej.: `
`, `<hr>`.

HTML fue usado en todas las páginas de esta aplicación web, sin excepción.

2.1.4. CSS (Cascading Style Sheets)

Es un lenguaje declarativo que describe cómo se muestran las páginas web en un navegador. Se puede usar para definir estilos de texto, tamaño de tablas, y otros aspectos que antiguamente sólo se podían definir en el código HTML. El código CSS puede ser definido en un archivo aparte, esto permite que varias páginas puedan reutilizar el mismo código y evitar la repetición de las declaraciones.

CSS3 es una versión mejorada y ampliada de las versiones anteriores, mejoras como esquinas redondeadas, sombras, transiciones, animaciones, etc.

CSS3 es el lenguaje principal usado para dar estilo a toda esta aplicación web.

2.1.5. MySQL

Es un sistema de administración de bases de datos relacionales basado en SQL (Structure Query Language, o Lenguaje de Consultas Estructurado). Su aplicación es altamente diversificada. Sin embargo, la aplicación más común de MySQL se da en las base de datos web. Puede usarse para almacenar desde un simple dato (contador de visitas) hasta un gran inventario de productos de una tienda online.

Usado en conjunto con lenguajes como PHP es posible crear sitios web que interactúan en tiempo real con dichas bases de datos.

2.2. FRAMEWORKS

Esta aplicación web ha sido desarrollada utilizando un **Framework**. De manera general se podría decir que éste es un conjunto de estructuras y guías utilizados para construir algo útil y con sentido.

Un contexto más delimitado del framework utilizado en esta aplicación es el **Web Framework**, el cual se podría definir como un software que ayuda a construir servicios de aplicaciones web y APIs; además permite la reutilización de componentes creados para llevar a cabo tareas repetitivas de modo más fácil.

Elevando los conceptos a un nivel específico, se debe hablar del Framework PHP, teniendo en cuenta que el lenguaje principal utilizado para desarrollar la aplicación web ha sido PHP, este **Framework** se podría decir que es un conjunto de clases PHP que ayudan a construir una aplicación web con facilidad.

2.2.1. Laravel (PHP)

Laravel es un Framework PHP empleado en el desarrollo de aplicaciones web. Sigue la estructura MVC (Modelo Vista Controlador), que hace que sea más rápido crear estructuras web ya que posee funciones como autenticación, servicio de correo, enrutamiento, sesiones, etc.

Gracias a la facilidad para personalizar proyectos el tiempo empleado suele ser más reducido que siguiendo métodos tradicionales de programación

Entre las características de Laravel se encuentran:

- **Modularidad:** Los componentes creados pueden ser separados y reunificados para hacer la aplicación funcional.
- **Autenticación:** Tiene un sistema de autenticación que se puede implementar con la aplicación de una instrucción.

- **Caché temporal:** Crea una propia cache temporal para facilitar la reutilización de los datos usados con más frecuencia.
- **Enrutamiento:** permite agrupar, nombrar, filtrar o vincular rutas a los modelos fácilmente, además de usar rutas amigables.
- **Plantilla de visualización: Blade** es una plantilla de visualización de Laravel. Posee funciones que ayudan a dar formato a los datos dentro de las vistas. Además, implementa la *herencia*, lo que permite usar *estructuras base* (layouts) que pueden ser reutilizadas con facilidad. Todas las vistas de blade tienen la extensión `.blade.php`
- **Constructor de consultas:** sin duda una herramienta muy valiosa. Permite realizar consultas con mucha facilidad de un modo legible para el usuario. También es posible realizar consultas del modo tradicional si fuese necesario.
- **ORM (Object-Relation Mapping) Eloquent:** Mapeo de Objetos y relaciones, algo así como “qué objeto está relacionado con cuál” en la base de datos. Laravel posee una forma muy sencilla de relacionar los objetos a través de los modelos creados.
- **Sistema de migraciones:** Se pueden crear tablas, bases de datos, índices, usando el sistema de migraciones. Es posible modificar tablas o estructuras agregando migraciones.
- **Artisan:** Línea de comandos de Laravel, empleado para ejecutar tareas y aplicaciones en laravel, tales como las migraciones. Ej.: `“php artisan migrate”` esto permitiría ejecutar una migración para modificar, eliminar o agregar algo a la estructura de la base de datos.

Laravel también permite desplegar la aplicación en fase de desarrollo con un solo comando: `php artisan serve --host=localhost --port=80`. Con esta instrucción ya sería posible usar la aplicación desde el navegador web lo cual evita tener que usar aplicaciones de terceros para ejecutar un servidor PHP.

2.2.2. Bootstrap (CSS)

Según la página oficial, “Bootstrap es un conjunto de herramientas de código abierto para desarrollar con HTML, CSS y JS”. Originalmente creado por y para Tweeter ®.

Es un framework para front-end que está desarrollado para dar soporte a la creación de sitios web dinámicos y aplicaciones web. Es uno de los preferidos por muchos por su facilidad para aplicarlo a casi cualquier entorno web.

Un modo de usarlo incluye archivos precompilados que permiten un uso rápido en cualquier proyecto web. Además de los archivos CSS y JS compilados y minificados. El otro modo de usarlo es usando el código fuente.

Su facilidad de uso permite combinar estilos que enriquecen el entorno visual de una página con muy pocos recursos.

2.3. LA BASE DE DATOS

La aplicación web creada hace uso de una base de datos relacional para el almacenamiento de los datos. Consta de 8 tablas interrelacionadas: *user*, *employee*, *absences*, *positions*, *roles*, *task_positions*, *tasks*, *employee_tasks*.

Para esta aplicación web se eligió MySQL como el lenguaje a implementar en la base de datos. Las principales razones para ello son: velocidad, facilidad de uso, escalabilidad, disponibilidad (software libre), fiabilidad, y conectividad.

Todas las tablas tienen una clave primaria de tipo entero autoincrementable y algunas tablas tienen claves ajenas.

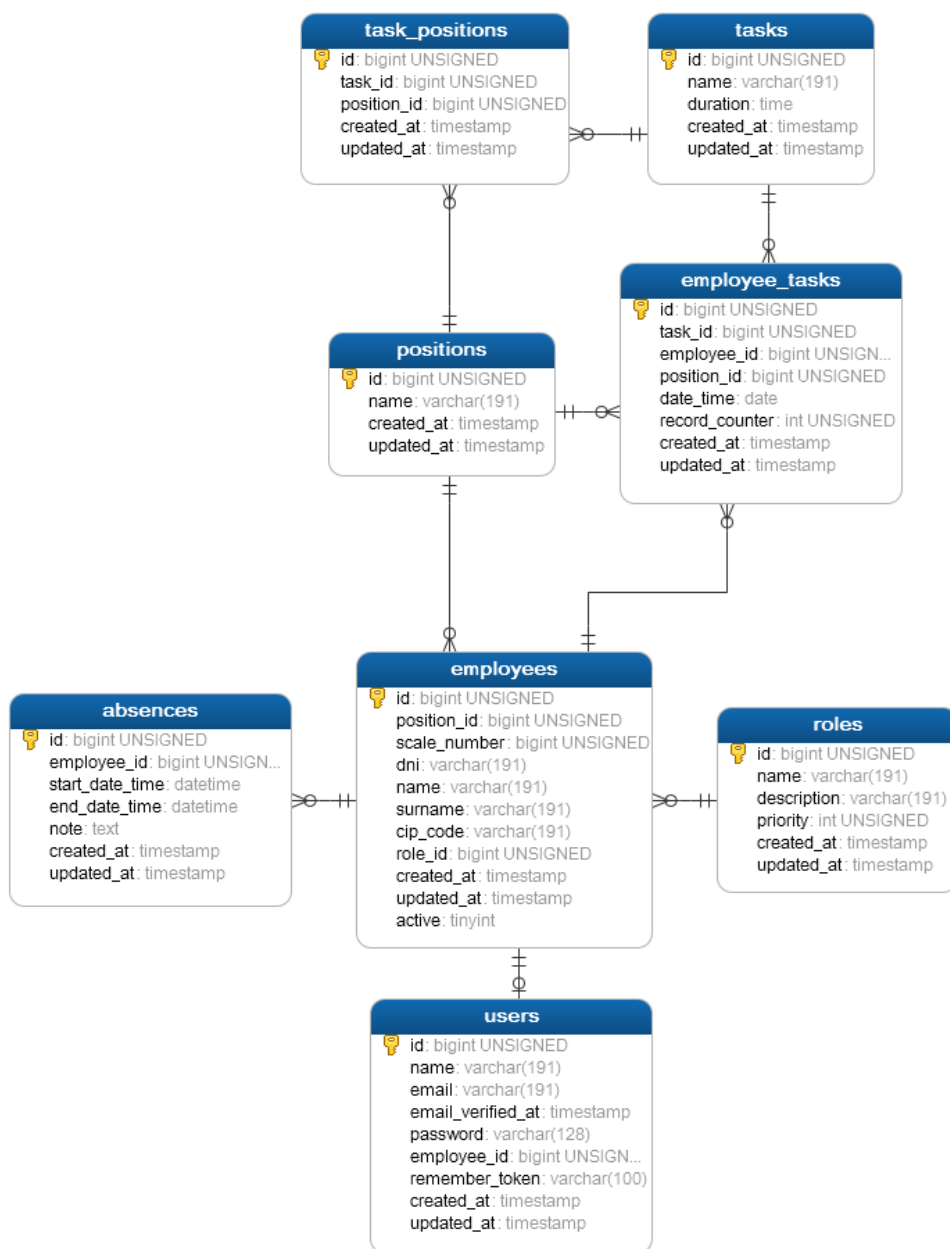


Ilustración 1, Relaciones de la base de datos

La estructura de la base de datos, tablas, relaciones y claves ajenas, es creada usando el sistema de migraciones de Laravel. En el siguiente ejemplo se muestra el código usado para crear la estructura de la tabla “tasks”:

<?php

```
use Illuminate\Support\Facades\Schema;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Database\Migrations\Migration;

class CreateTasksTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('tasks', function (Blueprint $table) {
            $table->bigIncrements('id');
            $table->string('name');
            $table->time('duration')->nullable();
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('tasks');
    }
}
```

3. ANÁLISIS DE REQUISITOS

3.1. PLANTEAMIENTO DE NECESIDADES

Se requiere una aplicación para asignar tareas (también llamados servicios) a un grupo de personas. Todos los usuarios deben poder acceder con usuario y contraseña. Los usuarios básicos no pueden registrarse por su cuenta, el administrador se encargará de ello.

Para el adecuado funcionamiento de la aplicación hay que atender ciertos criterios al momento de la asignación de tareas. Entre las necesidades para asignarlas se establece que todas las personas tienen un Empleo (rango o categoría) y una antigüedad en el empleo.

Se debe llevar un conteo de cada tarea asignada a cada persona para que todos hagan la misma cantidad de tareas por cada tarea asignada.

Se debe poder incluir como ausentes a las personas que lo estarán por cualquiera que sea la razón en una fecha o rango de fechas. Si al momento de añadir a una persona esa persona ya tiene una tarea asignada, esa tarea debe anularse y se reasignarán todas las tareas ya asignadas a partir de esa fecha para conservar el conteo de cada tarea asignada.

Las condiciones para la asignación son las siguientes:

- Si una persona estará ausente en una fecha específica no se le podrá asignar una tarea.
- Si una persona estará asignada a una tarea cualquiera, ese día ya no se le podrán asignar más tareas.
- Si una persona estará asignada a un servicio de armas, ese día y el día siguiente no se le podrán asignar más tareas.
- Servicio de armas, 24 horas de duración:
 - El orden empieza por la persona con más antigüedad en el empleo.

- Al terminar el servicio la persona tiene el resto del día libre y debe presentarse a trabajar al día siguiente.
- Servicio de orden, termina antes de que acabe la jornada laboral:
 - El orden empieza por la persona más moderna en el empleo.
 - Al terminar el servicio la persona se incorpora a su puesto de trabajo y continúa con la jornada laboral si procede.

3.2. QUIÉN USARÁ DE LA APLICACIÓN. CASOS DE USO

Se determina que podrá ser usada por dos usuarios. Uno con acceso básico y otro con privilegios de administrador que podrá acceder a todas las funciones de la aplicación.

Usuario: Puede ver sus propias tareas, ordenadas cronológicamente en orden descendente, un listado que muestra arriba las tareas próximas por hacer; y las tareas de todos los usuarios con tareas asignadas pendientes, ya que algunas tareas se deben realizar en grupo y es necesario ver el equipo de trabajo.

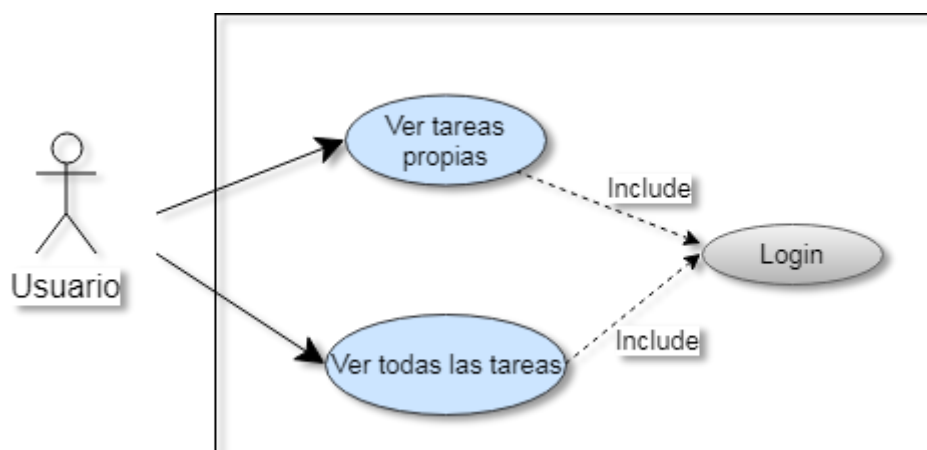


Ilustración 2, Casos de uso del usuario.

Administrador: Además de tener los mismos privilegios de un usuario normal, también puede: asignar, mostrar y borrar de tareas a los usuarios; CRUD de Tareas, Empleos, Relación tarea-empleo; CRUD de personal; alta, mostrar y borrar ausencias

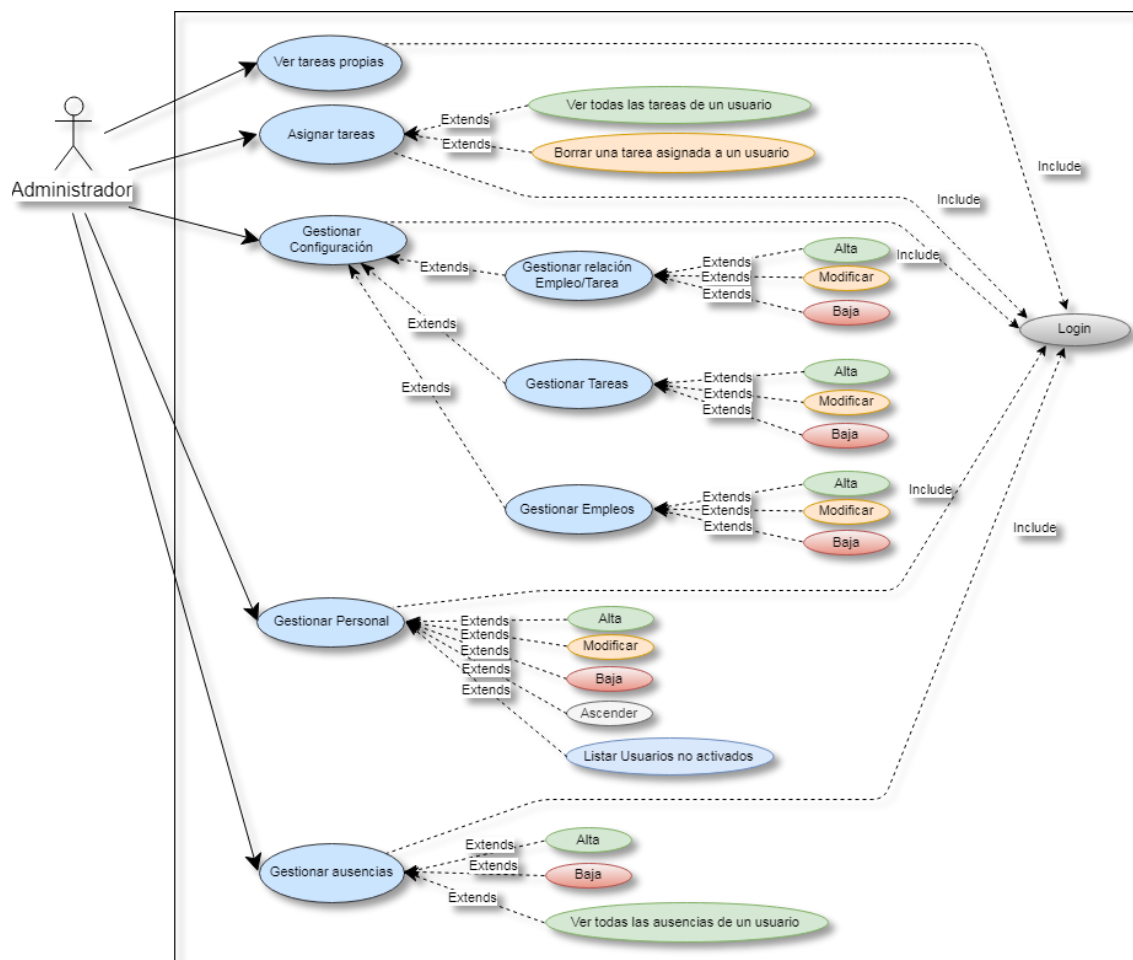


Ilustración 3, Casos de uso del administrador

4. LA APLICACIÓN WEB

4.1. DESCRIPCIÓN

Esta aplicación web ha sido desarrollada para que un administrador asigne tareas a usuarios teniendo en cuenta su empleo (categoría), antigüedad en el empleo, la fecha en la que se realizará la tarea, y el número de veces que un usuario ha realizado una tarea.

En la mayoría de las funciones de la aplicación donde se quiera editar o borrar información se ha configurado con la librería *SweetAlert* un mensaje de advertencia informando del posible riesgo que conlleva la alteración de dicha información. Si se intenta borrar algo que es vital para el sistema éste no lo permitirá.

4.2. INTERACCIÓN LENGUAJES

En general, Laravel se encarga de la manipulación de toda la información que circula en la aplicación. Un ejemplo de interacción sería cuando un usuario hace clic para ver sus tareas asignadas, a través de Blade Laravel recoge la solicitud, consulta la base de datos y devuelve a blade los datos para mostrarlos formateándolos usando Bootstrap y la librería de DataTables.

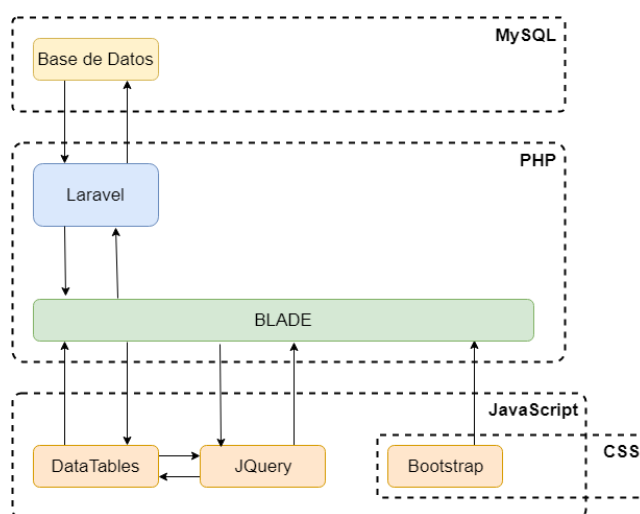


Ilustración 4, Interacción de los lenguajes

4.3. CONSTITUCIÓN

Consta de una página inicial para que el usuario ingrese sus credenciales y pueda hacer uso de ella.



Ilustración 5, Página de Login

Al ingresar la página se dividirá en *Encabezado*, *Menú lateral* y *Cuerpo*. Hay un menú en la parte izquierda que muestra 2 ítems si es un usuario, o el menú completo si es el administrador.

Usuario: Se muestran sólo las tareas asignadas al usuario en orden cronológico descendente.



| Servicio | Fecha |
|-------------|-------------|
| Guardia L/V | 27 Jun 2019 |

Ilustración 6, Menú de usuario, pantalla inicial

En esta sección “Todas las tareas”, se muestran todos los usuarios con tareas asignadas para que pueda saber si trabaja en grupo o individual.



Ministerio de Defensa
Sistema de gestion de tareas

David ▾

Usuario
Todas las tareas

TAREAS PROGRAMADAS

Listado de las tareas asignadas a todos a los usuario a partir de hoy.

Mostrar 10 registros Búsqueda en la tabla:

| Empleo | Nombre | Servicio | Fecha |
|---------|------------------|-------------|-------------|
| Soldado | German Correa | Guardia L/V | 27 Jun 2019 |
| Soldado | David Saavedra | Guardia L/V | 27 Jun 2019 |
| Soldado | Noelia Herrero | Guardia L/V | 27 Jun 2019 |
| Soldado | Fátima Negrete | Guardia L/V | 27 Jun 2019 |
| Soldado | Sandra Lorente | Guardia L/V | 27 Jun 2019 |
| Soldado | Fátima Negrete | Guardia L/V | 30 Jun 2019 |
| Soldado | Gabriel Santiago | Guardia L/V | 27 Jun 2019 |

Mostrando 1 a 7 de 7 registros Anterior 1 Siguiente

Ilustración 7, Todas las tareas

Administrador, al igual que al usuario al administrador también le aparecen tareas si las tuviese asignadas.



Ministerio de Defensa
Sistema de gestion de tareas

German ▾

Usuario
Asignación de tareas
Configuración
Personal
Ausencias

Tareas de: Soldado German Correa

Las tareas se muestran en orden cronológico. Abajo está la más antigua.

| Servicio | Fecha |
|-------------|-------------|
| Guardia L/V | 27 Jun 2019 |

Ilustración 8, Menú administrador, pantalla inicial

A continuación se muestra una parte de una plantilla de Blade usada para seleccionar qué se muestra dependiendo del tipo de rol que tiene el usuario:

```
{{-- Check user is logged in --}}
@if (!auth()->guest() )

    {{-- verify user role --}}
    @if(auth()->user()->employee->role->id === 2)
        <a href="/home" >
            <button class="bh3 col-xl-12 col-lg-12 col-md-12 btn-top text-left
            menu_item

                {{-- add class .active --}}
                <?php if(request()->path() == 'home' || request()->path() ==
                '/')echo 'active-side';?>">
                    &ensp; Usuario
                </button>
        </a>

        <a href="/tasks" >
            <button class="bh3 col-xl-12 col-lg-12 col-md-12 btn-top text-left
            menu_item

                {{-- add class .active --}}
                <?php if(request()->path() == 'assigntask' || request()->path() ==
                'tasks')echo 'active-side';?>">
                    &ensp; Todas las
                    tareas
                </button>
        </a>
    @endif
@endif
```


4.4. PLANTILLAS DE VISUALIZACIÓN

Gracias a que se usa Blade como plantilla de visualización, la pantalla principal usa una única plantilla base que a medida que se llaman otras páginas conforman un todo, es decir, el Encabezado y el menú lateral izquierdo son una plantilla, cuando se presiona un botón del menú este carga una página que es un segmento que complementa la plantilla base.

A continuación se muestra una parte de la plantilla principal que usa el elemento “@yield” para mostrar una sección de código proveniente de otra plantilla. Esta es la página `main.blade.php`:

```
.
.
<title>@yield('title')</title>
</head>
<body @yield('onload')>
.
.

@if(auth()->check())
    <div class="main col-xl-9 col-lg-9 col-md-9 col-sm-12 col-xs-12">
@else
    <div class="main col-xl-12 col-lg-12 col-md-12 col-sm-12 col-xs-12">
12">
@endif

    @yield('content')
</div>
.
.
```

En la siguiente imagen se enmarca la sección que se está mostrando con el código anteriormente expuesto.

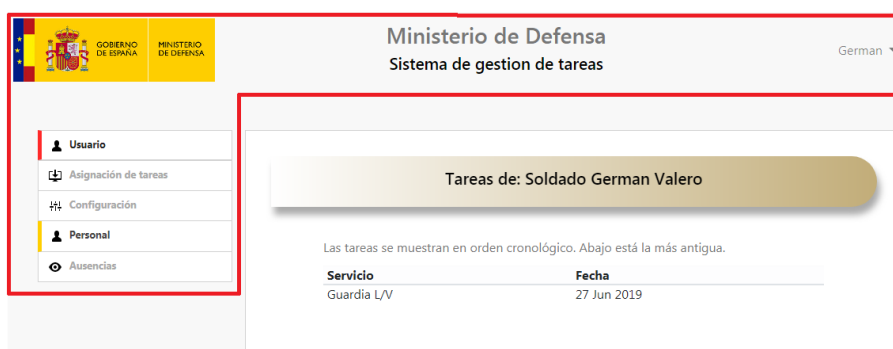


Ilustración 9, Plantilla base

El siguiente código inserta todo lo que está contenido entre “@section” y “@endsection” dentro del campo “@yield” que se encuentra en la plantilla `main.blade.php`

```
@extends('main')

@section('title', 'Ausencias')

@section('content')

    <div class="title py-3">
        .
        .
    </div>

@endsection
```

La siguiente imagen muestra la parte de la página que está representada por el código anterior.

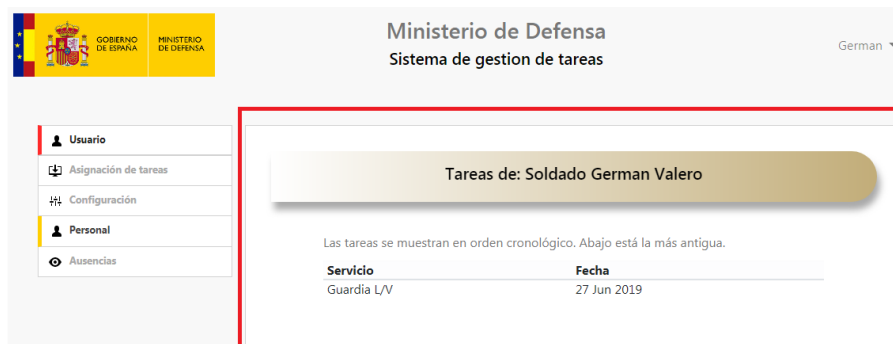


Ilustración 10, Complemento plantilla base



Ilustración 11, Complemento 2 plantilla base

Otra función que se ha implementado usando *Blade* en esta aplicación es la de mostrar selectivamente elementos en las páginas dependiendo de qué usuario esté activo en el sistema. De ahí que se muestre un menú para el Usuario y otro diferente para el Administrador.

4.5. PÁGINAS

4.5.1. Asignación De Tareas

Al momento de **asignar una tarea** se hace teniendo en cuenta los criterios anteriormente descritos en el PLANTEAMIENTO DE NECESIDADES. A continuación se muestra un diagrama de flujo, la explicación de cada función, código de ejemplo, Los pasos son los siguientes:

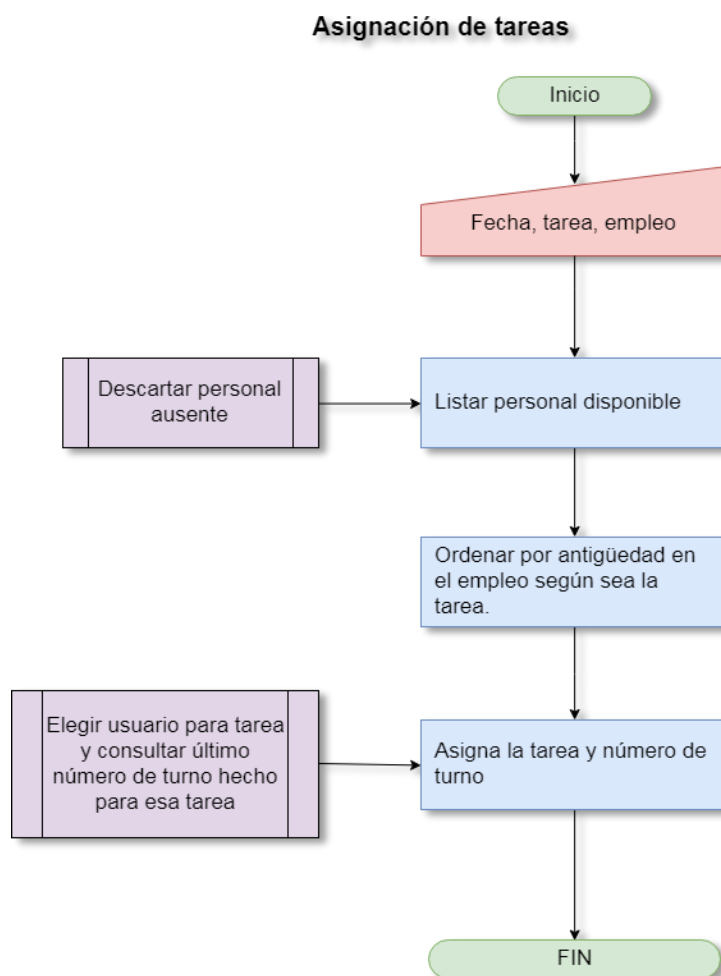


Ilustración 12, Diagrama para asignar tareas

- El administrador:
 - inserta la Fecha, Tarea, Empleo (se muestran sólo los que corresponden a la tarea, cargados a través de Ajax), y PAX.

Ilustración 13, Asignación de tareas

- El sistema internamente:
 - Hace listado: de los ausentes, de aquellos con tareas asignadas, de aquellos que tienen servicio de armas el día anterior, y los agrupa todos. El siguiente código fuente es un ejemplo de cómo se hace:

```
protected function _whoIsUnavailable($date, $position_id, $task_id){

    //convert received date to Carbon format
    $carbonDate = (Carbon::createFromFormat('Y-m-d', $date)->startOfDay());

    //filter inactive employees
    $inact = Employee::where('active', 0)->get()->pluck('id');

    //Filter employee ids who are absent
    $abss = Absence::
        where('start_date_time', '<=', $carbonDate)->
        where('end_date_time', '>=', $carbonDate)->
        get()->pluck('id');

    //filter employees doing a task on $date
    $onduty = EmployeeTask::
        where('date_time', '=', $carbonDate)->
        get()->pluck('employee_id');

    //filter employees who have a day off after task
    $beforeDutyDate = $carbonDate->subDay();
    $afterDuty = DB::table('employee_tasks')->
        where('date_time', '=', $beforeDutyDate)->
        where(function ($query) {
            $query->
                where( 'task_id', '=', 1)->
                orWhere( 'task_id', '=', 2);})->
        get()->
        pluck('employee_id');
```

```
//merge ids
$id = $inact->merge($abss);
$id = $id->merge($onduty);
$id = $id->merge($afterDuty);
$id = $id->unique();

$sids = $id->toArray();

//No value to send
if (!isset($id)){
    return ;
}else{
    return $sids;
}
}
```

- Compara el listado de todo el personal de ese empleo en la base de datos y descarta el listado anterior (todos los ausentes durante esa fecha), dejando únicamente el personal disponible y presente.
- Ordena el listado del personal disponible según la tarea a realizar.
- Obtiene el usuario y el número de veces que dicho usuario ha hecho la tarea que en cuestión. El siguiente diagrama de flujo muestra el proceso:

Escoger usuario y número de conteo de tarea

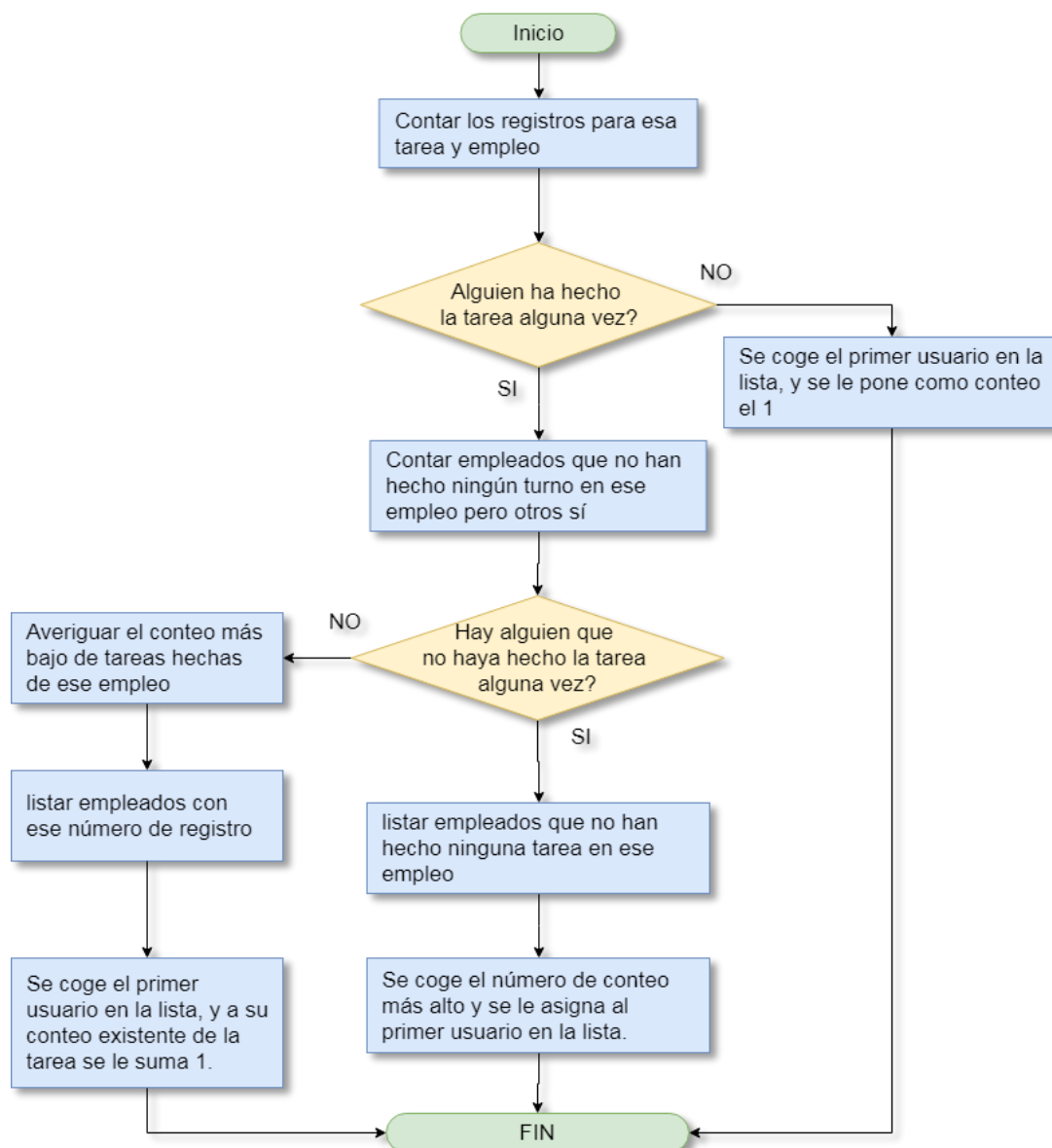


Ilustración 14, Cómo se obtiene el usuario y número de conteo de tarea

- Se almacena en la base de datos el usuario con el número de orden de la tarea.
- Se muestra la pantalla donde se visualizan todas las tareas a partir de hoy.

Listado de las tareas asignadas a todos a los usuario a partir de hoy.

Mostrar	10	registros	Búsqueda en la tabla:	
Empleo	Nombre	Servicio	Fecha	
Soldado	German Correa	Guardia L/V	27 Jun 2019	Borrar
Soldado	David Saavedra	Guardia L/V	27 Jun 2019	Borrar
Soldado	Noelia Herrero	Guardia L/V	27 Jun 2019	Borrar
Soldado	Fátima Negrete	Guardia L/V	27 Jun 2019	Borrar
Soldado	Sandra Lorente	Guardia L/V	27 Jun 2019	Borrar
Soldado	Fátima Negrete	Guardia L/V	30 Jun 2019	Borrar
Soldado	Gabriel Santiago	Guardia L/V	27 Jun 2019	Borrar
Mostrando 1 a 7 de 7 registros				
			Anterior	1 Siguiente

Ilustración 15, Tareas asignadas

4.5.2. Configuración

En esta sección se agregan, listan, actualizan y eliminan las tareas, empleos y la relación empleo-tarea. Esta sección debe manipularse con cuidado ya que la eliminación de un elemento de la base de datos podría alterar la configuración actual. Para agregar un elemento nuevo a las tareas o empleos, o relación de tarea-empleo, en la parte inferior de la tabla aparece la opción para hacerlo. Cuando se intenta guardar un nombre que ya existe el sistema vuelve al formulario indicando que ese nombre ya existe y que se debe especificar uno diferente. El siguiente diagrama de flujo muestra el proceso:

Nuevo Empleo / Tarea

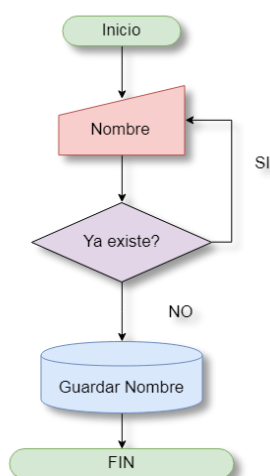


Ilustración 16, Diagrama nuevo empleo o tarea

Configuración

Empleo/Tarea
Empleo
Tarea

Configuración de los empleos (categorías)

Empleos existentes. Haga click en un empleo para modificarlo

Mostrar 10 registros

Búsqueda en la tabla:

Empleo ▼

Soldado

Cabo

Cabo 1º

Cabo Mayor

Sargento

Mostrando 1 a 5 de 5 registros

Anterior 1 Siguiente

Nuevo empleo

Ilustración 17, Configuración de tareas, empleo, relación tarea-empleo

Para activar la edición o borrado un elemento de los empleos o tareas se debe hacer clic sobre él. El elemento sólo pueden borrarse si no está siendo implementado en la base de datos. Si se intenta borrar el sistema no lo permite. Para cancelar la acción de *Guardar* o *Borrar* se presiona la tecla *ESC*.

Empleo ▼

Soldado Guardar Borrar

Cabo

Ilustración 18, Editar, Borrar elemento de la tabla empleo

Los elementos de la tabla de relaciones tarea-empleo pueden borrarse y no representan alteración alguna al funcionamiento correcto de la aplicación. No se pueden editar. Para activar la función de borrado hay que hacer clic sobre el empleo. Para cancelar la acción de *Borrar* se presiona la tecla *ESC*.

Configuración

Tarea/Empleo
Empleo
Tarea

Empleos asignados a las tareas

Qué tareas puede hacer quién.

Para borrar una relación haga click sobre el empleo.

Mostrar 10 registros
Búsqueda en la tabla:

Tarea	Empleo
Conductor Ambulancia	Soldado
Conductor Ambulancia	Cabo Borrar
Guardia L/V	Soldado

Ilustración 19, Configuración tarea-empleo

4.5.3. Personal

En esta sección se muestra todo el personal que está en servicio activo. Por motivos de visualización de elementos de la base de datos los usuarios no se pueden eliminar ya que al visualizar fechas anteriores los campos estarían vacíos. Para evitar eso lo que se hace es desactivar el usuario y deja de estar disponible para todas las opciones de la aplicación excepto para visualizarse en fechas anteriores.

Personal

Nuevo
Fuera de servicio

Para editar un empleado haga click sobre el nombre

Mostrar 10 registros
Búsqueda en la tabla:

Empleo	Nº Esc.	Nombre	DNI	Código CIP
Cabo	16	Cristian de Jesús	30683348Z	LN_274
Cabo	17	Irene Camacho	12554326Y	LN_720
Cabo	18	Julia Carrión	11223092N	LN_481
Cabo	19	Daniel Toledo	84428759Z	LN_799
Cabo	20	Cristian Ruíz	15204448E	LN_244
Cabo	21	Marina Montañez	67196834B	LN_317
Cabo	22	Sara Lázaro	04164360A	LN_842
Cabo	23	Diana Villalba	92915930R	LN_478
Cabo	24	Naïara Anaya	82125801T	LN_104
Cabo	25	Ana Almonte	86614271C	LN_858

Mostrando 1 a 10 de 50 registros

Anterior

1

2

3

4

5

Siguiente

Ilustración 20, Tabla de Personal

La visualización de todos los usuarios se muestra en una tabla HTML normal la cual es formateada con la librería *DataTables* de JavaScript. Con esta librería se puede ordenar la tabla por cualquier columna, buscar en todas las celdas a modo de filtrado en tiempo real, mostrar en modo paginación por cantidad de filas (10, 25, 50 o 100 filas).

Para editar la información de un usuario se debe hacer clic sobre el nombre.




Ilustración 21, Información personal de usuario

Desde este menú es posible editar la información del usuario, ascenderle en el empleo, y cambiar su estado entre activo y *no activo*.

Ascender a un usuario en el empleo no tiene repercusiones a la hora de visualizarse en las tareas asignadas, allí se mostrará el usuario siempre con el empleo que tenía al momento de asignársele la tarea.

Cuando un usuario deja de estar en servicio activo, en la aplicación, éste de visualizarse en las tareas asignadas y las ausencias. Si el usuario tiene una tarea asignada pendiente de realizar en el futuro, en el momento que pasa a estar *no activo* todas las tareas asignadas son eliminadas desde esa fecha hacia adelante y son reasignadas automáticamente.

4.5.4. Ausencias

Para un correcto funcionamiento de la aplicación es necesario mantenerla actualizada con los usuarios con que no estarán presentes para realizar las tareas.

Ausencias

Búsqueda en la base de datos

- Aquí se muestran las ausencias de hoy en adelante, para buscar fechas anteriores debe realizarse una búsqueda manualmente.
- Para ver todas las ausencias de un usuario haga click sobre el nombre.

Mostrar registros
Búsqueda en la tabla:

Nombre	F. inicio	F. fin	Notas
Carla Lomeli	27 Jun 2019	27 Jun 2019	
Leyre Carrera	27 Jun 2019	27 Jun 2019	
Raquel Viera	27 Jun 2019	27 Jun 2019	

Mostrando 1 a 3 de 3 registros

Anterior

Siguiente

Ilustración 22, Ausencias

Las opciones para esta sección: Nueva ausencia, Búsqueda en la base de datos y borrado de una ausencia. No se permite la edición. Para borrar una ausencia hay que hacer clic en el nombre del usuario y luego elegir la ausencia que se quiere borrar.

Para agregar una nueva ausencia se debe escoger al usuario del listado disponible. El orden de la lista es alfabético por apellido.

Nueva ausencia

Agregar

14058994Z : Lucas Aguayo

Fecha inicio

Fecha fin

Notas

Ilustración 23, Nueva ausencia

Una función relevante a destacar es la actualización automática de las tareas asignadas cuando se agrega una nueva ausencia, es decir, cuando se agrega una persona el sistema captura los datos del que identifican al usuario y las fechas, busca en la tabla de tareas asignadas si esa persona tiene tareas asignadas en dichas fechas y de ser así elimina todas las tareas asignadas de ahí en adelante y reasigna nuevamente todas las tareas anteriormente eliminadas. A continuación se muestra un ejemplo de lo anteriormente descrito:

Mostrar 10 registros
Búsqueda en la tabla:

Empleo	Nombre	Servicio	Fecha	
Soldado	Clara Ontiveros	Guardia L/V	20 Jun 2019	Borrar
Soldado	Izan Costa	Guardia L/V	20 Jun 2019	Borrar
Soldado	Nayara Páez	Guardia L/V	20 Jun 2019	Borrar
Soldado	Ana Costa	Guardia L/V	21 Jun 2019	Borrar
Soldado	Erika Rubio	Guardia L/V	21 Jun 2019	Borrar
Soldado	Javier Terrazas	Guardia L/V	21 Jun 2019	Borrar

Mostrando 1 a 6 de 6 registros

Anterior

1

 Siguiente

Ilustración 24, Tareas asignadas

Luego se agrega a un usuario al listado de bajas, para este ejemplo se agregará a “Nayara Pérez” que tiene una guardia asignada para el 20 de junio de 2019.

Mostrar registros

Búsqueda en la tabla:

Nombre	F. inicio	F. fin	Notas
Nayara Pérez	20 Jun 2019	20 Jun 2019	Baja médica

Mostrando 1 a 1 de 1 registros

Anterior

1

Siguiente

Ilustración 25, Ejemplo de ausencia por baja médica

En la siguiente ilustración se puede ver como a “Nayara Pérez” se reasigna la tarea moviéndola del 20 de junio que no estará presente al 21 de junio, y la tarea programada para el 20 de junio es asignada a quien corresponde manteniendo el orden de antigüedad del empleo del listado.

Mostrar 10 registros

Búsqueda en la tabla:

Empleo	Nombre	Servicio	Fecha	
Soldado	Clara Ontiveros	Guardia L/V	20 Jun 2019	Borrar
Soldado	Izan Costa	Guardia L/V	20 Jun 2019	Borrar
Soldado	Ana Costa	Guardia L/V	20 Jun 2019	Borrar
Soldado	Nayara Pérez	Guardia L/V	21 Jun 2019	Borrar
Soldado	Erika Rubio	Guardia L/V	21 Jun 2019	Borrar
Soldado	Javier Terrazas	Guardia L/V	21 Jun 2019	Borrar

Mostrando 1 a 6 de 6 registros

Anterior

1

Siguiente

Ilustración 26, Tarea reasignada después de agregar a un usuario como ausente

5. CONCLUSIONES

Seleccionar el lenguaje ideal o los lenguajes en que se va a programar es esencial, más importante aún es crear un planeamiento previo antes de empezar a desarrollar código siguiendo pasos que, está demostrado que si se siguen, funcionan.

Gracias a los frameworks se ahorra tiempo que normalmente había que dedicar a funciones que hoy en día suelen desarrollarse para la mayoría de las aplicaciones web, funciones como controlar el acceso, restablecer contraseñas, registro, etc., dichas funciones ahora ya están preestablecidas y esto se traduce en tiempo ahorrado. Hacer uso de los frameworks no es garantía de que la aplicación estará disponible en menos tiempo o quedará mejor estructurada pero son de gran ayuda indiscutiblemente. Está claro que si no se conocen las funciones nativas del lenguaje en el que se está programando intentar usar un framework puede ser contraproducente causando pérdidas de tiempo además de experimentar frustración.

Durante el desarrollo de esta aplicación se hizo uso de frameworks como Laravel y Bootstrap. La principal razón para haber elegido Laravel fue su gran facilidad para crear y manipular consultas a una base de datos. Y Bootstrap porque da estilo a las páginas de un modo muy sencillo y directo. Las demás librerías usadas, JQuery, DataTables, o SweetAlert, también han sido herramientas de gran ayuda. Es relevante decir que durante la implementación de todas ellas la línea de aprendizaje se incrementaba directamente proporcional a las ganas de querer perfeccionar cada función, estructura, página, botón, etc., que con dichas herramientas sería posible de lograr.

Esta aplicación se ha desarrollado con la intención real de ponerla en marcha y que no fuese simplemente un trabajo destinado a ser un proyecto más al final de un curso; otro documento más que sirve de ejemplo y guía para los cursos siguientes. El destino final sería las oficinas de las unidades militares del ejército de España, allí surge esta necesidad. A fecha de hoy estas tareas algunos las organizan tablas o listados en papel, otros usan hojas de cálculo como Microsoft Excel® o Calc® de LibreOffice, y

probablemente no sea lo más acertado, por experiencia propia se podría decir que intentar organizar las tareas para un grupo más de 70 personas en épocas vacacionales con una antelación de dos semanas, teniendo en cuenta todas las variables que se necesitan, es muy agotador.

Sería de gran ayuda a esas personas encargadas de asignar tales tareas, porque cuando hay que hacerlo toma mucho tiempo, pero cuando hay que hacer modificaciones porque aquellas personas que se suponía estarían presentes ya no lo estarán, entonces hace falta reajustar todo para que se reasignen de nuevo las tareas; porque lo ideal es que todos los que tienen hacerlas tengan el mismo número de ellas a lo largo del año.

Desarrollar una aplicación web requiere mucho tiempo y dedicación, sin duda ésta no ha sido la excepción. Han sido muchas más horas de las que se supone que se deben dedicar, pero por todo lo aprendido verdaderamente ha merecido la pena. Sólo a través de la práctica y esfuerzo se puede mejorar cada día para aspirar a dedicarse a la programación web o cualquier tipo de programación.

6. TRABAJOS FUTUROS.

Durante el desarrollo de esta aplicación web surgieron muchas opciones para incluir como parte del desarrollo principal pero debido al tiempo que representaba incluirlas no fue posible.

Entre las distintas funciones que se pretendía incluir estaba la posibilidad de que el administrador, aplicando la ley de protección de datos europea, no mostrase el nombre del usuario sino el CODIGO_CIP.

También la posibilidad de que un usuario pudiese informar al administrador a través de la aplicación las fechas en que estaría ausente.

Igualmente, sería necesario agregar una tabla de “Excluidos” en donde se pueda registrar los usuarios quienes por una condición particular, como puede ser una reducción de jornada, se verían exentos de realizar algunas tareas.

En algunas situaciones sería necesario que existiesen más roles aparte del administrador y usuario. Así por ejemplo podría existir un rol intermedio que se encargue sólo de asignar tareas y ausencias pero que no pueda manipular nada de la configuración.

Escoger a un usuario específicamente al momento de asignar una tarea sería otra posibilidad, podría ser un caso de voluntariado excepcional.

Cambiar el idioma de la interfaz de usuario según corresponda a la configuración regional.

El usuario podría hacer simulaciones de las posibles tareas en un futuro para prever cuándo se vería afectado.

También existe la posibilidad de poder adaptar la aplicación realizando variaciones en la estructura y en las funciones para gestionar turnos de trabajo, no tareas. Esta necesidad surge a las personas encargadas de jefaturas de enfermería o a las

empresas donde hay trabajadores que realizan turnos de 8 horas que van rotando durante cada día de la semana, como los de vigilancia privada, en los que debe haber cierto equilibrio para que todos hagan aproximadamente las mismas jornadas durante el mes.

7. BIBLIOGRAFÍA.

- 8 agosto 2014, <https://techterms.com/definition/javascript>
- Margaret Rouse, noviembre 2006.
<https://whatis.techtarget.com/definition/PHP-Hypertext-Preprocessor>
- 2019, https://www.tutorialspoint.com/ajax/what_is_ajax.htm
- Refsnes Data, 1999-2019, https://www.w3schools.com/xml/ajax_intro.asp
- Mozilla, 2019, <https://developer.mozilla.org/es/docs/Glossary/jQuery>
- Mozilla, 2019, <https://developer.mozilla.org/en-US/docs/Web/CSS/CSS3>
- Esepe studio, 16 agosto 2005, <https://www.espestudio.com/noticias/que-es-mysql>
- 5 octubre 2018, <https://www.larashout.com/what-is-laravel-and-why-you-should-learn-it>
- Bootstrap, <https://getbootstrap.com/>
- S. Bharath, 2013, <https://vmokshagroup.com/blog/bootstrap-advantages/>

8. MANUAL INSTALACIÓN/CONFIGURACIÓN/USO.

8.1. INFORMACIÓN GENERAL

Las siguientes instrucciones indican qué se debe hacer previo a la utilización de la aplicación web desarrollada. Ésta, gracias a que todo el software utilizado para su desarrollo es de carácter gratuito y de libre distribución se puede usar en cualquier sistema operativo que soporte los lenguajes utilizados. Para este manual el sistema operativo de aplicación ha sido Microsoft Windows® 7 por lo que las instrucciones son para éste.

8.2. INSTALACIÓN DE MYSQL

Para poder ejecutar esta aplicación es necesario tener un servidor de MySQL. Para instalarlo se debe descargar desde internet del siguiente vínculo:
<https://dev.mysql.com/downloads/installer/>

Durante la instalación se debe especificar una contraseña que servirá para futuros usos a través de consola y conexiones.

Una vez instalado por completo se debe crear la base de datos. Esto se puede hacer a través de la consola con las siguientes instrucciones:

1. Primero hay que entrar con usuario y contraseña
 - ***mysql -u username -p***
2. Se ingresa la contraseña para avanzar. Una vez dentro del sistema MySQL creamos la base de datos
 - ***create database taskmanager;***

A partir de este momento ya se puede continuar con la ejecución de los demás programas para el desarrollo de la aplicación.

8.3. INSTALACIÓN DE LARAVEL

Laravel usa *Composer* para sus administrar sus subprogramas, lo que hace necesario que *Composer* esté instalado en el ordenador antes de poder utilizar la aplicación web.

Composer se puede instalar desde la ruta:

- <https://getcomposer.org/Composer-Setup.exe>

8.4. INICIAR EL SERVIDOR DE PHP

Una vez instalado *Composer* ya se puede ejecutar la aplicación web. Desde una consola de comandos (command, powershell, GitBash) y estando dentro de la carpeta donde se encuentra la aplicación, se ejecuta la instrucción para iniciar el servidor web

```
php artisan serve --host=localhost --port=80
```

```
MINGW64:/d/TaskManager  
german@german-PC MINGW64 /d/TaskManager (master)  
$ php artisan serve --host=localhost --port=80  
Laravel development server started: <http://localhost:80>
```

8.5. LOGIN

En el navegador se abre la página `http://localhost`

Sistema de Gestion de tareas

E-Mail Address
Password

☐ Remember Me

Login

email

contraseña

Una vez logueado la primera página muestra las tareas asignadas.

GOBIERNO DE ESPAÑA
 MINISTERIO DE DEFENSA

Ministerio de Defensa
 Sistema de gestion de tareas

German ▼

Usuario

[Asignación de tareas](#)

[Configuración](#)

[Personal](#)

[Ausencias](#)

Tareas de: Soldado German Correa

Las tareas se muestran en orden cronológico. Abajo está la más antigua.

Servicio	Fecha
Guardia L/V	27 Jun 2019

8.6. ASIGNACIÓN DE TAREAS

Seleccionar **Asignación de tareas** en el menú de la izquierda.

Seleccionar la fecha en la que se quiere programar la tarea

Seleccionar la tarea

Seleccionar el empleo del usuario que realizará la tarea. Sólo se muestran aquellos que pueden hacer esa tarea.

¿Cuántos usuarios realizarán esa tarea ese mismo día?

Listado de las tareas asignadas.

Listado de las tareas asignadas a todos a los usuario a partir de hoy.

Mostrar	10	registros	Búsqueda en la tabla:	
Empleo	Nombre	Servicio	Fecha	
Soldado	German Carrión	Guardia L/V	27 Jun 2019	Borrar
Soldado	David Delrio	Guardia L/V	27 Jun 2019	Borrar
Soldado	Beatriz Toledo	Guardia L/V	27 Jun 2019	Borrar
Soldado	Nicolás Ibarra	Guardia L/V	27 Jun 2019	Borrar
Soldado	Leyre Huerta	Guardia L/V	27 Jun 2019	Borrar
Soldado	Nicolás Ibarra	Guardia L/V	30 Jun 2019	Borrar
Soldado	Gael Segura	Guardia L/V	27 Jun 2019	Borrar
Mostrando 1 a 7 de 7 registros			Anterior	1 Siguiente

8.6.1. Borrar tarea asignada

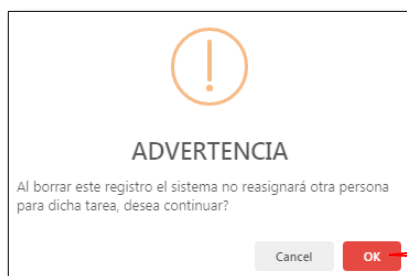
Listado de las tareas asignadas a todos a los usuario a partir de hoy.

Mostrar registros Búsqueda en la tabla:

Empleo	Nombre	Servicio	Fecha	
Soldado	German Carrión	Guardia L/V	27 Jun 2019	Borrar
Soldado	David Delrio	Guardia L/V	27 Jun 2019	Borrar
Soldado	Beatriz Toledo	Guardia L/V	27 Jun 2019	Borrar
Soldado	Nicolás Ibarra	Guardia L/V	27 Jun 2019	Borrar
Soldado	Leyre Huerta	Guardia L/V	27 Jun 2019	Borrar
Soldado	Nicolás Ibarra	Guardia L/V	30 Jun 2019	Borrar
Soldado	Gael Segura	Guardia L/V	27 Jun 2019	Borrar

Mostrando 1 a 7 de 7 registros Anterior Siguiente

Clic **Borrar** para borrar una tarea asignada.



Confirmar borrado.

8.7. CONFIGURACIÓN

8.7.1. Nuevas de tareas, empleos,



Seleccionar **Configuración** en el menú de la izquierda

Clic en Empleo

Listado de todos los empleos en la base de datos.

En la parte de abajo al hacer clic sobre "Nuevo empleo" se activa el campo para escribir el nuevo empleo y aparece el botón para guardar.

Se escribe el nombre y se presiona Guardar. En caso de querer cancelar se presiona la tecla ESC.

Mensaje de advertencia después de presionar Guardar.

El procedimiento para la creación de tareas es igual que el procedimiento para crear empleos.

8.7.2. Relación tarea-empleo

Clic en Empleo/Tarea

Listado de todas las relaciones actuales.

En la parte de abajo al hacer clic sobre la tarea o el empleo se activan los selectores para escoger el empleo y la tarea. Aparece también el botón para guardar.

Se selecciona el empleo y la tarea a relacionar y clic en Guardar.

Mensaje de advertencia después de hacer clic en Guardar.

Configuración

Empleo/Tarea Empleo Tarea

Empleos asignados a las tareas

Qué tareas puede hacer quién.
Para borrar una relación haga click sobre el empleo.

Mostrar 10 registros
Búsqueda en la tabla:

Tarea	Empleo
Conductor Ambulancia	Soldado
Conductor Ambulancia	Cabo
Guardia L/V	Cabo
Guardia L/V	Cabo 1º
Guardia S, D, Fs	Soldado
Guardia S, D, Fs	Cabo
Guardia S, D, Fs	Cabo 1º
Limpieza HQ	Soldado
Limpieza HQ	Cabo

Mostrando 1 a 9 de 9 registros Anterior 1 Siguiente

Nueva Relación.

Guardia L/V Soldado

Nueva Relación.

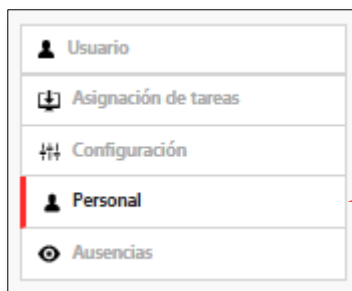
Guardia S, D, Fs Sargento Guardar

Guardar

Desea guardar esta entrada?

Cancel OK

8.8. PERSONAL



Seleccionar **Personal** en el menú de la izquierda

Personal fuera de servicio

Para visualizar la información de un usuario haga clic sobre el nombre.

Personal				
<div>Nuevo</div> <div>Fuera de servicio</div>				
Para editar un empleado haga click sobre el nombre				
Mostrar	10	registros	Búsqueda en la tabla:	
Empleo	Nº Esc.	Nombre	DNI	Código CIP
Cabo	16	Julia Roca	64647033J	LN_415
Cabo	17	Francisco Roca	75582821Z	LN_297
Cabo	18	Izan Sisneros	37284561C	LN_732
Cabo	19	Alma Sisneros	91519367K	LN_357
Cabo	20	Jan Alfonso	22597441X	LN_793
Cabo	21	Jan Mena	08544687A	LN_037
Cabo	22	Álvaro Solís	70023432R	LN_204
Cabo	23	Isaac Arredondo	21770417C	LN_711
Cabo	24	Marcos Córdova	81664048H	LN_923
Cabo	25	Jon Parra	32560373D	LN_180
Mostrando 1 a 10 de 50 registros				
<div>Anterior</div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div> <div>Siguiente</div>				

Información del usuario.

Julia Roca

Información personal

Nuevo

Editar

Ascender

☒ Usuario activo

Empleo

Nº Escalafón

Rol de Usuario

Cabo

16

usuario

Nombre

Apellidos

Julia

Roca

DNI

Código CIP

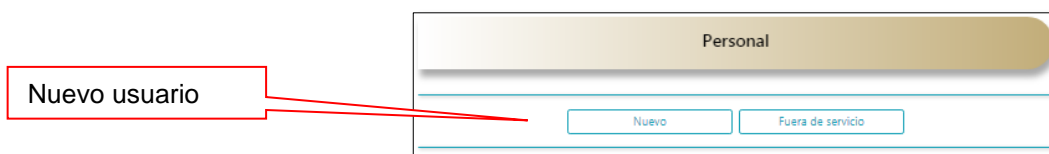
Email

64647033J

LN_415

marina08@gmail.com

8.8.1. Nuevo usuario



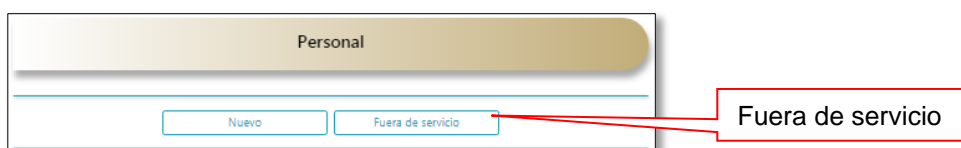
Formulario para crear un nuevo usuario.

A screenshot of a form titled 'Nuevo Información personal'. The form contains several input fields and dropdown menus. At the top, there is a header bar with the word 'Nuevo' and the subtitle 'Información personal'. Below this, there are three dropdown menus: 'Empleo' (with 'Soldado' selected), 'Nº Escalafón' (with 'Nº Escalafón' selected), and 'Rol Usuario' (with 'usuario' selected). Below these, there are two text input fields: 'Nombre' and 'Apellidos'. Below these, there are three text input fields: 'DNI', 'Código CIP', and 'Email'. At the bottom left, there is a blue button labeled 'Guardar'.

Se rellena toda la información y clic en Guardar, si la información que debe ser única como el DNI, CODIGO_CIP, o email ya existen en la base de datos la información no se guardará y se volverá al formulario para corregir la información.

8.8.2. Listado de personal fuera de servicio

Listado del personal que no está disponible. Estos usuarios no aparecen en los listados de personal en servicio activo ni en las ausencias.



Para reactivar un usuario hay que hacer clic sobre el nombre.

Personal fuera de servicio

Mostrar 10 registros Búsqueda en la tabla:

Empleo	Nº sc.	Nombre	DNI	Código CIP
Cabo	16	Julia Roca	64647033J	LN_415

Mostrando 1 a 1 de 1 registros Anterior 1 Siguiente

Luego marcar la casilla “Usuario activo”

Julia Roca
Información personal

Nuevo Editar Ascender ☒ Usuario activo

Empleo N° Escalafón Rol de Usuario

Cabo 16 usuario

Nombre Apellidos

Julia Roca

DNI Código CIP Email

64647033J LN_415 marina08@gmail.com

8.8.3. Ascender a usuario de empleo.

Clic Ascender

Julia Roca
Información personal

Nuevo Editar Ascender ☒ Usuario activo

Empleo N° Escalafón Rol de Usuario

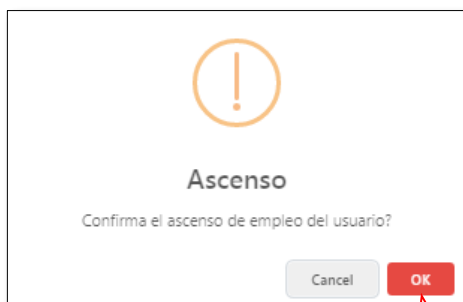
Cabo 16 usuario

Nombre Apellidos

Julia Roca

DNI Código CIP Email

64647033J LN_415 marina08@gmail.com



Confirmar

8.8.4. Editar un usuario.

Julia Roca
Información personal

[Nuevo](#) [Editar](#) [Ascender](#) ☐ Usuario activo

Empleo: Cabo N° Escalafón: 16 Rol de Usuario: usuario

Nombre: Julia Apellidos: Roca

DNI: 64647033J Código CIP: LN_415 Email: marina08@gmail.com

Se modifica la información, al hacer clic en Guardar se verifica que la información que debe ser única no se repita. Si esta repetida se volverá al formulario para corregir la información.

Julia Roca
Información personal

Empleo: Cabo N° Escalafón: 16 Rol de Usuario: usuario

Nombre: Julia Apellidos: Roca

DNI: 64647033J Código CIP: LN_415 Email: marina08@gmail.com

[Guardar](#) [Cancelar](#)

8.9. AUSENCIAS



Seleccionar **Ausencias** en el menú de la izquierda

Búsqueda de ausencias en la base de datos.

Listado de personas ausentes.

Clic sobre el nombre para ver todas las ausencias de ese usuario.

Ausencias

Búsqueda en la base de datos

- Aquí se muestran las ausencias de hoy en adelante, para buscar fechas anteriores debe realizarse una búsqueda manualmente.
- Para ver todas las ausencias de un usuario haga click sobre el nombre.

Mostrar 10 registros Búsqueda en la tabla:

Nombre	F. inicio	F. fin	Notas
Sara Agosto	08 Jun 2019	09 Jun 2019	
Leire Téllez	27 Jun 2019	27 Jun 2019	
Vera Lucas	27 Jun 2019	27 Jun 2019	
Sara Agosto	28 Jun 2019	29 Jun 2019	

Mostrando 1 a 4 de 4 registros

Anterior
1
Siguiente

8.9.1. Nueva ausencia

Clic para una Nueva Ausencia

Ausencias

Nueva ausencia

Búsqueda en la base de datos

01/06/2019 30/06/2019 Buscar

Elegir el usuario.

Nueva ausencia

DNI : Nombre Apellido

20622159Z : Sara Agosto

Fecha inicio Fecha fin

dd/mm/aaaa dd/mm/aaaa

Notas

Anotaciones

Agregar

8.9.2. Borrar Ausencia

Clic sobre el nombre para ver todas las ausencias de ese usuario.

Mostrar registros

Búsqueda en la tabla:

Nombre	F. inicio	F. fin	Notas
David Carrillo	27 Jun 2019	28 Jun 2019	
Leire Téllez	27 Jun 2019	27 Jun 2019	
Sara Agosto	28 Jun 2019	29 Jun 2019	

Clic sobre el la fecha a borrar

Ausencias de: Soldado David Carrillo		
Para eliminar una ausencia haga click sobre ella.		
1.	Fecha Inicio	Fecha fin
	27 Jun 2019	28 Jun 2019

Eliminar

Confirma que desea eliminar esta ausencia?

Cancel
OK

Confirmar borrado