

Documentación PocketCare

Ingeniería en Software I
Profesor Juan Pablo Soto

Integrantes

Arce Díaz Christopher
Castillo Machado Nicole
Leyva Conde Gerardo
Thomson Martínez Luis Ernesto
Pivac Alcaráz Vesna Camile

Contenido

Documentación PocketCare	1
Introducción al programa	3
Descripción del programa:.....	3
Herramientas utilizadas	3
Lenguajes	3
Casos de Uso	4
Diagramas de Clases	16
Diagrama de Casos de uso:.....	17
Diagrama de carril	17
Evaluación diaria	17
Crear evento	18
Añadir evento	18
Inicio de sesión	19
Registrar alumno	19
Diagramas de Secuencias	22
Agregar evento como Maestro	22
Iniciar sesión	22
Crear cuenta como Maestro	23
Inscribir Alumno	23
Crear grupo	23
Eliminar grupo	24
Editar grupo	24
Evaluación diaria	24
Revisar evaluación diaria	25
Casos de prueba de Login	25
Archivos necesarios	26
Snippets del código.....	28
Conexión con base de datos	28
Borrar alumno.....	28
Borrar grupo	28
Correo	28
Registro alumno.....	29
Registro grupo	29
Login	30

Introducción al programa

El programa consistirá en una página web que podrá usarse en kínder o guarderías y estará instalada en los teléfonos de los padres y maestros, permitirá a los maestros comunicarse fácilmente con los padres ya que por medio del programa podrán avisar de eventos próximos en la escuela, tareas para los niños, calificaciones, etc.

Los padres recibirán notificaciones de todas estas actualizaciones, y también de los tiempos de entrega y salida de sus hijos, lo cual servirá para que puedan supervisar el estado de su hijo a pesar de no estar con ellos.

Descripción del programa:

El programa consistirá en una página web.

Habrà un login donde se tendrá que crear una cuenta para poder utilizar el programa, las cuentas estarán todas clasificadas como cuenta de padre o cuenta de maestro, y el programa se comportará diferente dependiendo de que tipo de cuenta es la que está afiliada al dispositivo en el que se esté usando.

Al crear una cuenta tipo padre, se requerirá llenar los datos de nombre, correo, número de hijos junto con sus nombres y sus curps, una vez creada la cuenta, el padre necesitará recibir una invitación de una cuenta de maestro para poder ingresar a uno o más de sus hijos en el grupo de ese maestro, cuando esté ingresado en el grupo recibirá todas las notificaciones que el maestro haga.

Al crear una cuenta tipo maestro, se requerirá de los datos de la escuela en la que enseña y de las cuentas de los maestros con los que trabaja, una vez hecho esto el maestro podrá crear y manejar sus diferentes grupos, y podrá agregar alumnos a ellos mandando invitaciones a las cuentas de los padres.

Herramientas utilizadas

- StarUML
- Trello
- SublimeText 3
- Atom 8.0
- SQLWorkbench
- Git/GitHub

Lenguajes

- PHP 7.0.1
- HTML5
- SQL

Casos de Uso

Nombre:	Crear cuenta de Maestro
Autores:	Christopher Arce Nicole Castillo Gerardo Leyva Vesna Pivac
	Luis Thomson

Fecha: 13/02/2019

Descripción:

El caso de uso "Crear cuenta de Maestro" permite a los maestros sin cuenta crear una cuenta en la base de datos de PocketCare

Actores:

Maestro.

Precondiciones:

-No hay precondiciones para este caso de uso

Flujo Normal:

- 1.- El maestro selecciona opción "No tengo cuenta"
- 2.- Maestro Introduce su Nombre de usuario, escuela en la que enseña y la contraseña que desea para su cuenta
- 3.- Maestro selecciona "Crear cuenta"

Flujo Alternativo:

- 1.A.- Maestro sin cuenta no selecciona "No tengo cuenta", el programa no puede ejecutar ninguna función.
- 2.A.- Maestro no introduce correctamente su Nombre, contraseña, y/o escuela en la que enseña, la cuenta no se puede crear.
- 3.A.- Maestro introduce sus datos pero no selecciona "Crear cuenta", la cuenta no se crea.

Poscondiciones:

-La cuenta del maestro es agregada a la base de datos

Nombre:	Iniciar sesión Cuenta Maestro
Autores:	Christopher Arce Nicole Castillo Gerardo Leyva Vesna Pivac Luis Thomson
Fecha:	13/02/2019

Descripción:

El caso de uso “Iniciar sesión cuenta Maestro” permite iniciar sesión y entrar al sistema a todos los usuarios registrados como maestros.

Actores:

Maestro.

Precondiciones:

-Haber creado cuenta de Maestro

Flujo Normal:

- 1.- El maestro introduce sus nombre de usuario y contraseña que estableció al crear su cuenta.
- 2.- El maestro selecciona “Iniciar sesión”.

Flujo Alternativo:

- 1.A.- El maestro no introduce sus nombre de usuario o contraseña, no se puede iniciar sesión.
- 1.B.- El maestro introduce datos erróneos, no se puede iniciar sesión.
- 2.A.- El maestro introduce su nombre de usuario pero no selecciona “Iniciar Sesión”, no se puede iniciar sesión.

Poscondiciones:

-La cuenta queda inicializada y quedan habilitadas las funciones de las cuentas tipo Maestro

Nombre:	Creación de grupos
Autores:	Christopher Arce Nicole Castillo Gerardo Leyva Vesna Pivac Luis Thomson
Fecha:	13/02/2019

Descripción:

El caso de uso "Creación de grupos" crea un grupo de alumnos seleccionados por el maestro.

Actores:

Maestro.

Precondiciones:

- Haber creado una cuenta de maestro.
- Iniciar sesión.

Flujo Normal:

- 1.- Maestro selecciona la opción de crear nuevo grupo.
- 2.- Maestro selecciona el nombre del grupo.
- 3.- Maestro agrega a los alumnos que vayan a estar en ese grupo.
- 4.-Maestro confirma la creación del grupo.

Flujo Alternativo:

- 2.A.- Maestro selecciona el nombre de un grupo ya existente, no se podrá crear el grupo.
- 3.B.- Maestro selecciona a un alumno que ya esté en otro grupo no se podrá agregar al nuevo grupo.

Poscondiciones:

- El grupo sera creado y podrán buscar al alumno perteneciente al grupo desde cualquier tipo de cuenta.

Nombre:	Eliminación de grupo
Autores:	Christopher Arce Nicole Castillo Gerardo Leyva Vesna Pivac Luis Thomson
Fecha:	13/02/2019

Descripción:

El caso de uso "Eliminación de grupo" elimina un grupo de alumnos que ya no serán utilizados.

Actores:

Maestro.

Precondiciones:

- Haber creado una cuenta de maestro.
- Iniciar sesión.
- Haya un grupo.

Flujo Normal:

- 1.- Maestro selecciona la opción de eliminación de grupo.
- 2.- Maestro selecciona el grupo a eliminar.
- 3.- Maestro confirma la eliminación del grupo.

Flujo Alternativo:

2.A.- Maestro selecciona el nombre de un grupo ya existente, no se podrá crear el grupo.

Poscondiciones:

-El grupo será eliminado del sistema, por lo que las cuentas de los padres asociadas al alumno ya no tendrán acceso al sistema al menos que esté vinculada a otro alumno.

Nombre:	Inscribir Alumnos
Autores:	Christopher Arce Nicole Castillo Gerardo Leyva Vesna Pivac Luis Thomson
Fecha:	13/02/2019

Descripción: El caso de uso "Inscribir Alumnos" permite al maestro inscribir alumnos a un grupo.
Actores: Maestro.
Precondiciones: -Haber creado una cuenta de Maestro. -Iniciar sesión. -Seleccionar un grupo.
Flujo Normal: 1.- Maestro selecciona un alumnos para inscribir en el grupo. 2.- Maestro inscribe alumnos en el grupo.
Flujo Alternativo:
Poscondiciones: -Los alumnos seleccionados estarán inscritos en el grupo.

Nombre:	Añadir Evento a Calendario
Autores:	Christopher Arce Nicole Castillo Gerardo Leyva Vesna Pivac Luis Thomson
Fecha:	11/02/2019

Descripción:

El caso de uso “Añadir evento” agrega un evento relevante al calendario escolar propio de la escuela.

Actores:

Maestro.

Precondiciones:

- Haber creado una cuenta de maestro.
- Iniciar sesión.

Flujo Normal:

- 1.- Maestro selecciona la opción de añadir evento.
- 2.- Maestro selecciona la fecha del evento.
- 3.- Maestro selecciona los grupos a los que aplica el evento.
- 4.- Maestro escribe descripción del evento.
- 5.- Maestro publica el evento.

Flujo Alternativo:

- 2.A.- Maestro selecciona fecha anterior al día de publicación, el evento no se puede publicar.
- 3.B.- Maestro no selecciona grupos a los que aplica el evento, el evento no se puede publicar.
- 5.C.- Maestro selecciona fecha válida, selecciona grupos a los que aplica el evento y escribe la descripción del evento pero no selecciona publicar, el evento no se publica.

Poscondiciones:

- Evento queda registrado en el calendario en la fecha seleccionada para todas las cuentas que lo revisen.
- Padres y maestros recibirán la notificación de que se agregó un evento y un día antes de que ocurra, como forma de recordatorio.

Nombre:	Eliminación De Evento
Autores:	Christopher Arce Nicole Castillo Gerardo Leyva Vesna Pivac Luis Thomson
Fecha:	14/02/2019

Descripción:

El caso de uso "Eliminación de evento" elimina un evento del calendario escolar propio de la escuela.

Actores:

Maestro.

Precondiciones:

- Haber creado una cuenta de maestro.
- Iniciar sesión.
- Exista un evento.

Flujo Normal:

- 1.- Maestro selecciona el evento.
- 2.- Maestro selecciona la opción de eliminar evento.
- 3.- Maestro selecciona los grupos a los que se le eliminara el evento.
- 4.- Maestro selecciona la opción eliminar.

Flujo Alternativo:

- 3.A.- Maestro no selecciona grupo al que se le eliminara el evento, el evento no se puede eliminar.
- 4.B.- Maestro selecciona el evento, selecciona los grupos a los que se le eliminara el evento pero o selecciona eliminar, el evento no se elimina.

Poscondiciones:

- Evento se elimina del calendario para todas las cuentas que lo revisen.
- Padres y maestros recibirán la notificación de que se eliminó un evento.

Nombre:	Publicación de evaluación diaria
Autores:	Christopher Arce Nicole Castillo Gerardo Leyva Vesna Pivac Luis Thomson
Fecha:	14/02/2019

Descripción:

El caso de uso “Publicación de evaluación diaria” agrega una publicación sobre cómo se comportó el alumno durante el día escolar en el calendario escolar propio de la escuela.

Actores:

Maestro.

Precondiciones:

- Haber creado una cuenta de maestro.
- Iniciar sesión.
- El alumno esté inscrito en el grupo del maestro.

Flujo Normal:

- 1.- Maestro selecciona el grupo.
- 2.- Maestro selecciona el alumno.
- 3.- Maestro selecciona agregar evaluación diaria.
- 4.- Maestro escribe la evaluación diaria.
- 5.- Maestro selecciona publicar evaluación diaria.

Flujo Alternativo:

- 4.A.- Maestro no escribe nada en la evaluación diaria, la evaluación diaria no se publica.
- 5.B.- Maestro selecciona un grupo valido, un alumno valido, selecciona agregar evaluación diaria, escribe la evaluación diaria pero no selecciona publicar, entonces la evaluación diaria no se publica.

Poscondiciones:

- La evaluación diaria se publica en el calendario de cada alumno para que cada cuenta lo revise.
- Padres recibirán la notificación de que se publicó la evaluación diaria.

Nombre:	Publicación de observación diaria
Autores:	Christopher Arce Nicole Castillo Gerardo Leyva Vesna Pivac Luis Thomson
Fecha:	14/02/2019

Descripción:

El caso de uso "Publicación de observación diaria" agrega una observación sobre cómo se comportó el alumno durante el día escolar en el calendario escolar propio de la escuela.

Actores:

Maestro.

Precondiciones:

- Haber creado una cuenta de maestro.
- Iniciar sesión.
- El alumno esté inscrito en el grupo del maestro.

Flujo Normal:

- 1.- Maestro selecciona el grupo.
- 2.- Maestro selecciona el alumno.
- 3.- Maestro selecciona agregar observación diaria.
- 4.- Maestro escribe la observación diaria.
- 5.- Maestro selecciona publicar observación diaria.

Flujo Alternativo:

- 4.A.- Maestro no escribe nada en la observación diaria, la observación diaria no se publica.
- 5.B.- Maestro selecciona un grupo valido, un alumno valido, selecciona agregar observación diaria, escribe la observación diaria pero no selecciona publicar, entonces la observación diaria no se publica.

Poscondiciones:

- La observación diaria se publica en el calendario de cada alumno para que cada cuenta lo revise.
- Padres recibirán la notificación de que se publicó la observación diaria.

Nombre:	Crear cuenta tipo Padre
Autores:	Christopher Arce Nicole Castillo Gerardo Leyva Vesna Pivac Luis Thomson
Fecha:	13/02/2019

Descripción:

El caso de uso “Crear cuenta tipo Padre” crea una cuenta para el Padre al ser proporcionada una clave de alumno.

Actores:

Padre o tutor.

Precondiciones:

-Tener una clave de alumno.

Flujo Normal:

- 1.- El padre selecciona la opción de crear cuenta tipo Padre.
- 2.- El padre proporciona la clave del alumno.
- 3.- El padre ingresa sus datos.
- 4.- El padre selecciona registrar.
- 5.- Se crea la cuenta.

Flujo Alternativo:

- 2.A.- El padre no proporciona la clave del alumno, no se puede registrar.
- 3.B.- El padre no ingresa todos sus datos, no se puede registrar.
- 5.C.- El padre no selecciona la opción de registrar, no se crea la cuenta.

Poscondiciones:

- El padre inicia sesión.
- Se notifica al Maestro correspondiente el registro de un nuevo Padre.

Nombre:	Iniciar sesión cuenta Padre
Autores:	Christopher Arce Nicole Castillo Gerardo Leyva Vesna Pivac Luis Thomson
Fecha:	13/02/2019

Descripción:

El caso de uso “Iniciar sesión cuenta Padre” permite iniciar sesión y entrar al sistema a todos los usuarios registrados como padres o tutores.

Actores:

Padre o tutor.

Precondiciones:

-Haber creado una cuenta de padre.

Flujo Normal:

- 1.- El padre selecciona iniciar sesión como padre.
- 2.- El padre proporciona el nombre de usuario.
- 3.- El padre proporciona su contraseña.
- 4.- El padre inicia sesión.

Flujo Alternativo:

- 1.A.- El padre selecciona iniciar sesión como maestro, al no contar con clave de maestro, no puede iniciar sesión.
- 2.B.- El padre proporciona el nombre de usuario incorrecta o no la proporciona en lo absoluto, no inicia sesión.
- 3.C.- El padre proporciona la contraseña incorrecta o no la proporciona en lo absoluto, no inicia sesión.

Poscondiciones:

-El padre entra al sistema y se le presentan las opciones disponibles.

Nombre:	Revisar Evaluación Diaria
Autores:	Christopher Arce Nicole Castillo Gerardo Leyva Vesna Pivac Luis Thomson
Fecha:	13/02/2019

Descripción:

El caso de uso “Revisión de Evaluación Diaria” permite al padre o tutor revisar el rendimiento del alumno, así como observaciones del maestro.

Actores:

Padre o tutor.

Precondiciones:

-Haber iniciado sesión como padre.

Flujo Normal:

- 1.- El padre inicia sesión como padre.
- 2.- El padre selecciona la opción de revisar evaluación diaria.
- 3.- Se presenta la evaluación (con caritas) y probablemente una observación del maestro.

Flujo Alternativo:

- 2.A.- El padre no selecciona la opción de revisar evaluación diaria, no se le presenta la evaluación.
- 3.B.- La evaluación no se presenta ya que aún no ha sido proporcionada por el maestro.

Poscondiciones:

-Se notifica al maestro que el padre ha revisado la evaluación. (?)

Nombre:	Revisar Notificaciones
Autores:	Christopher Arce Nicole Castillo Gerardo Leyva Vesna Pivac Luis Thomson
Fecha:	13/02/2019

Descripción:

El caso de uso “Revisar Notificaciones” permite al tutor revisar las notificaciones de próximos eventos relevantes en la escuela.

Actores:

Padre o tutor.

Precondiciones:

- Haber creado una cuenta de Tutor.
- Iniciar sesión.

Flujo Normal:

- 1.- Tutor selecciona Revisar Notificaciones.

Flujo Alternativo:

Poscondiciones:

- El tutor será capaz de enterarse de los próximos eventos en la escuela.

Diagramas de Clases

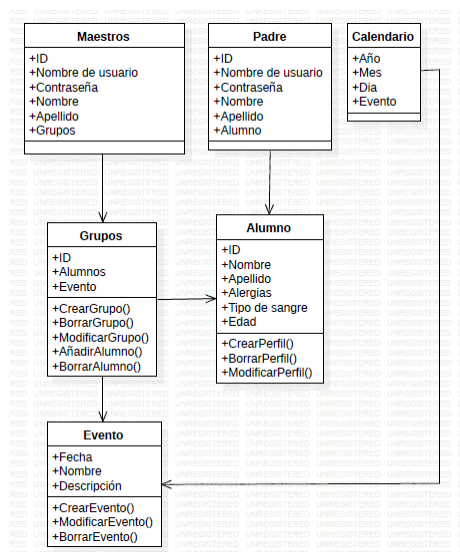


Diagrama de Casos de uso:

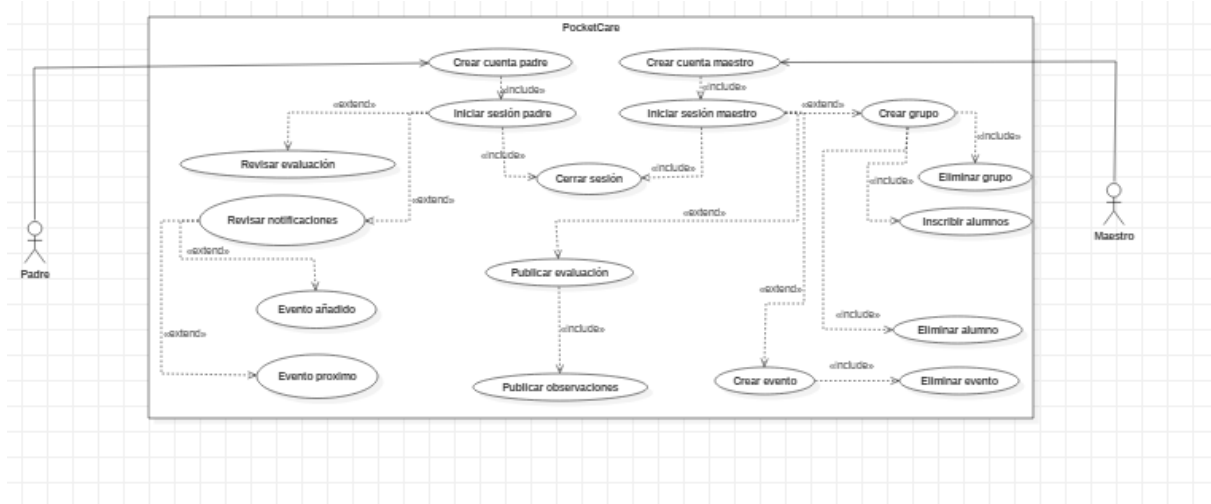
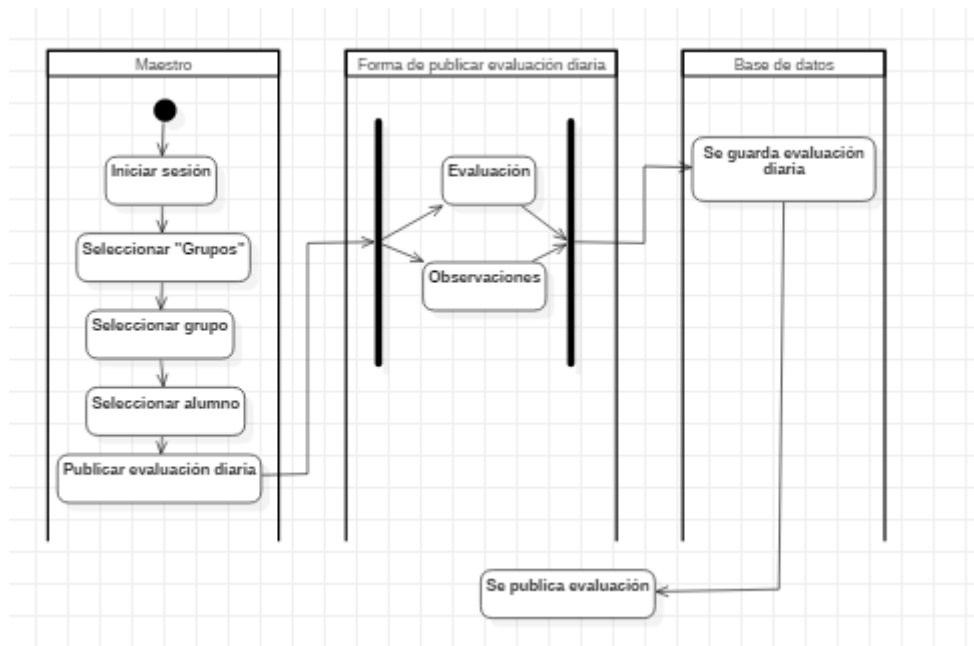
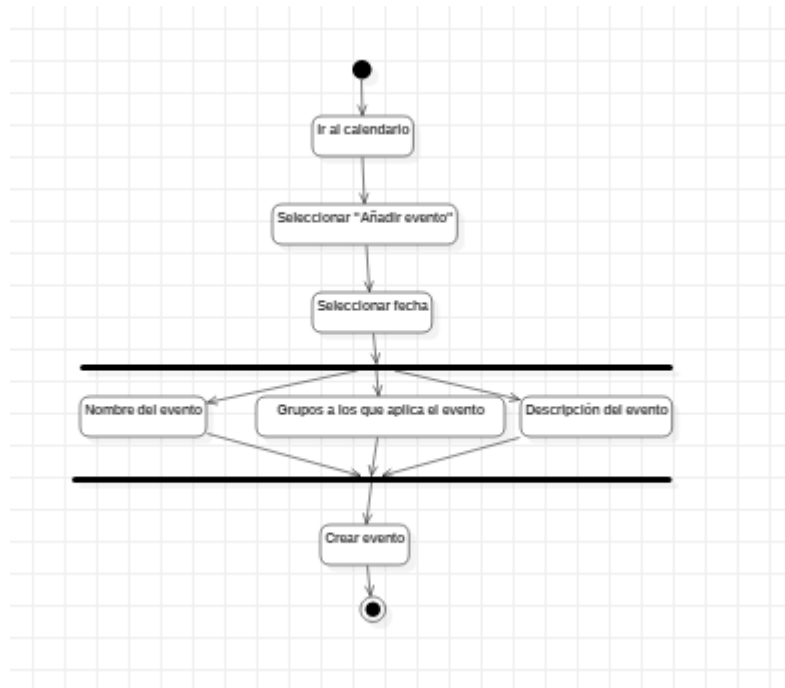


Diagrama de carril

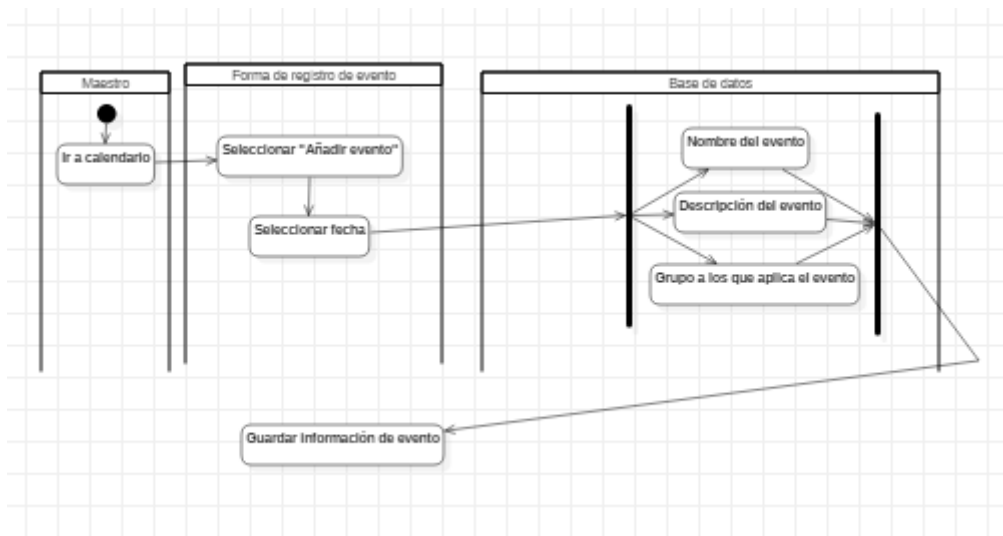
Evaluación diaria



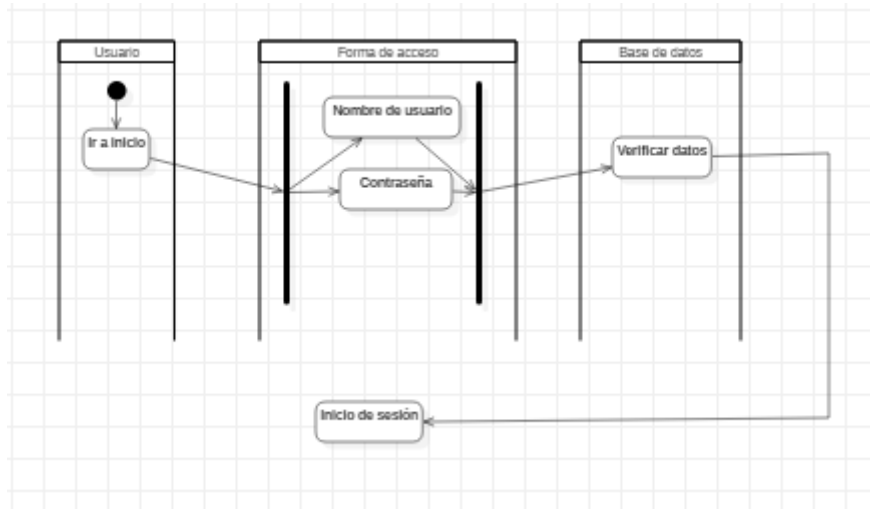
Crear evento



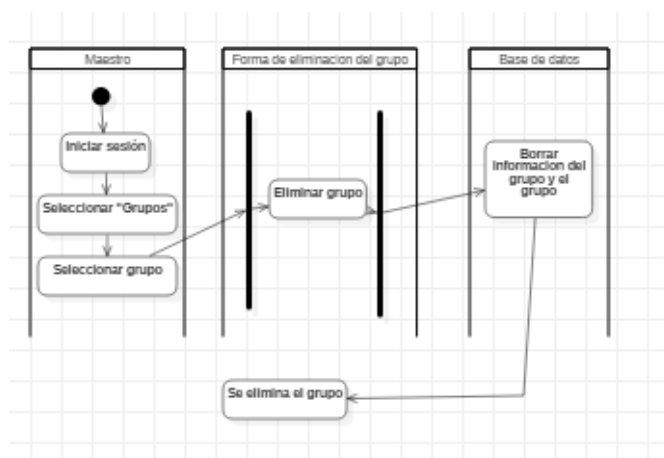
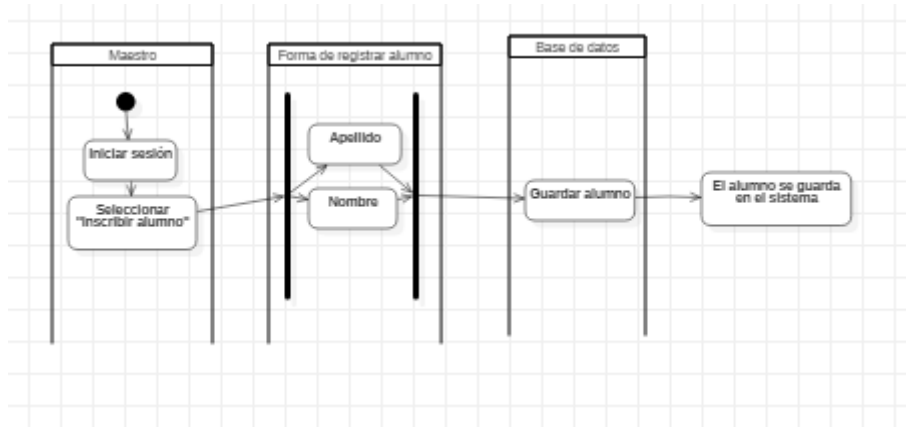
Añadir evento

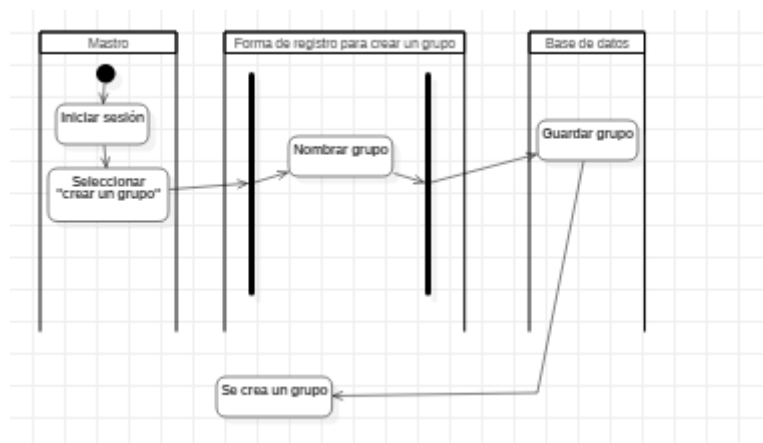
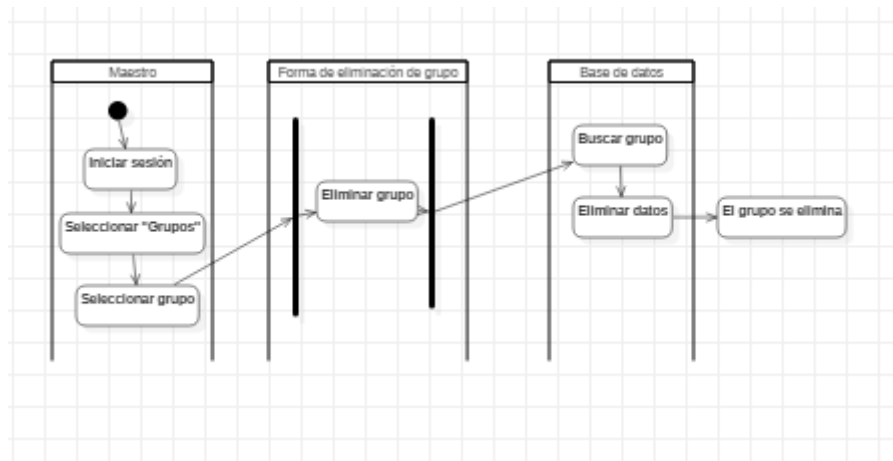
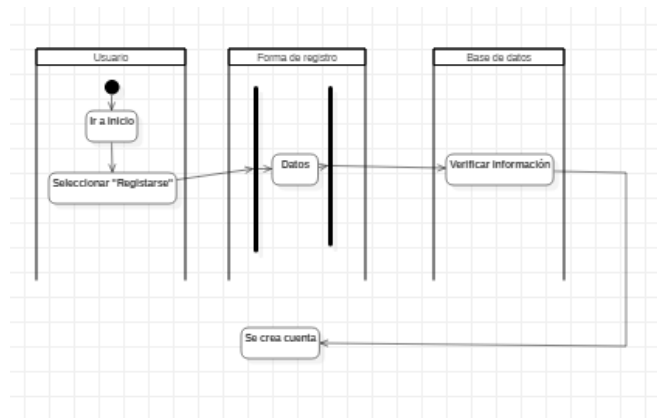


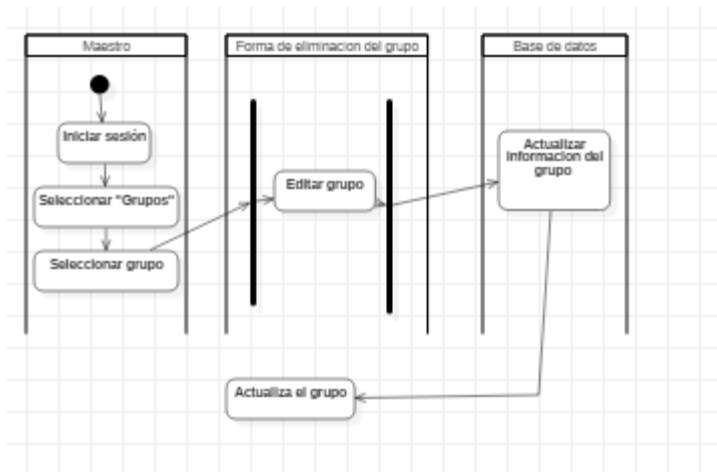
Inicio de sesión



Registrar alumno

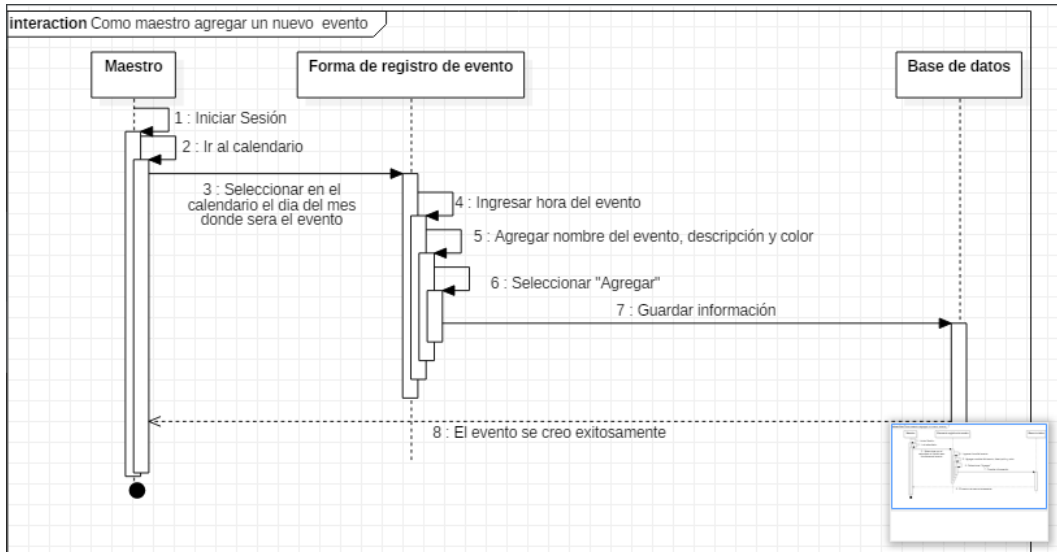




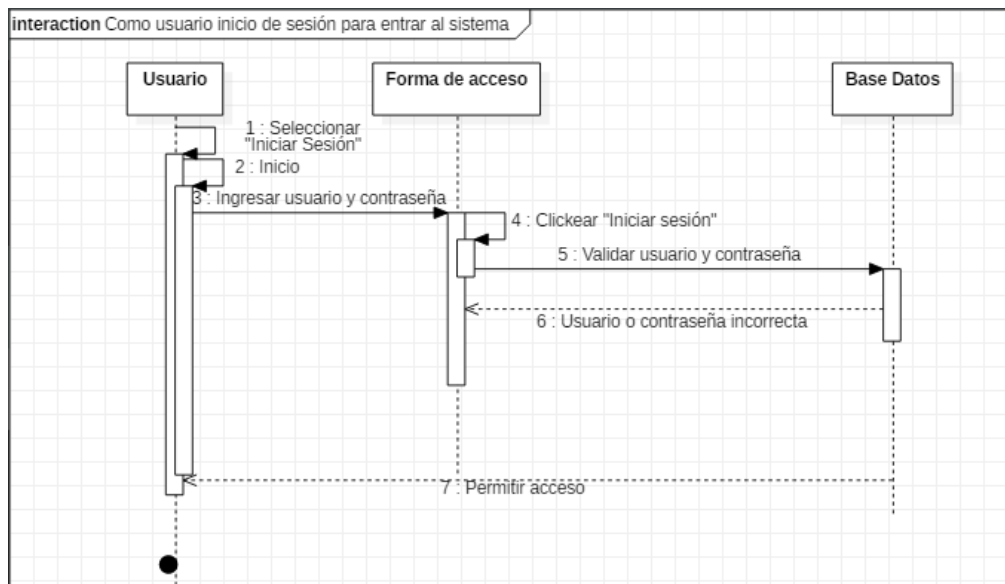


Diagramas de Secuencias

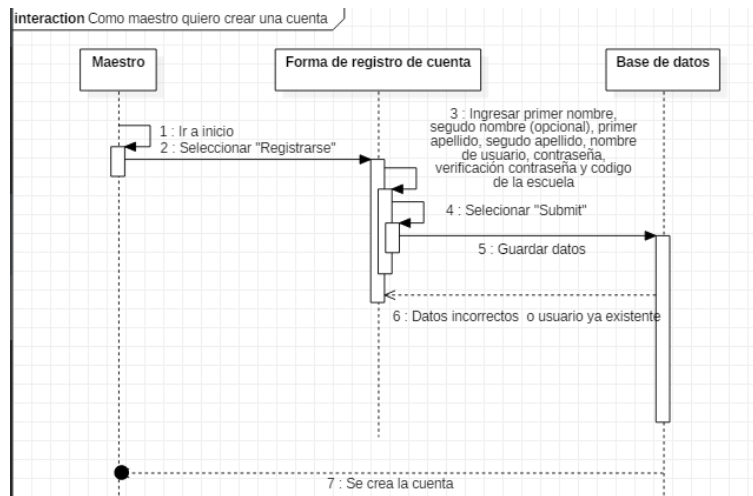
Agregar evento como Maestro



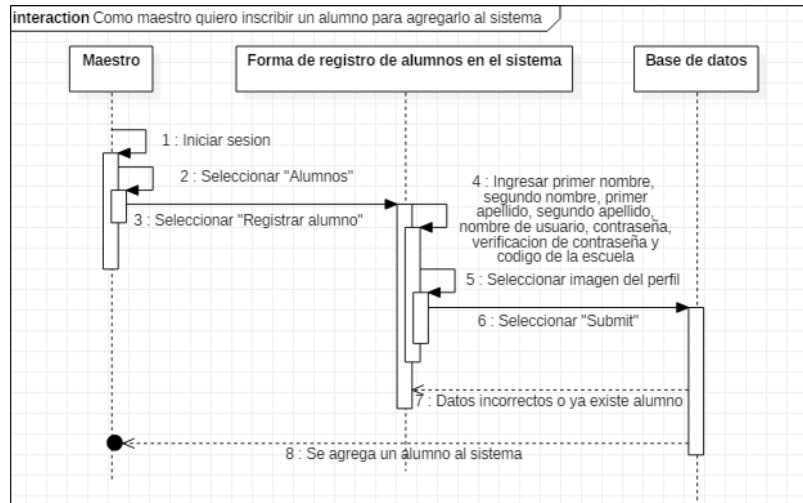
Iniciar sesión



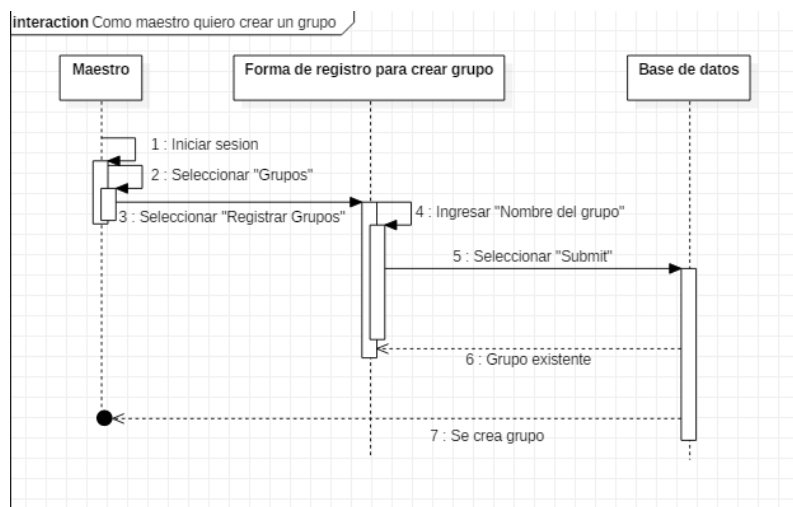
Crear cuenta como Maestro



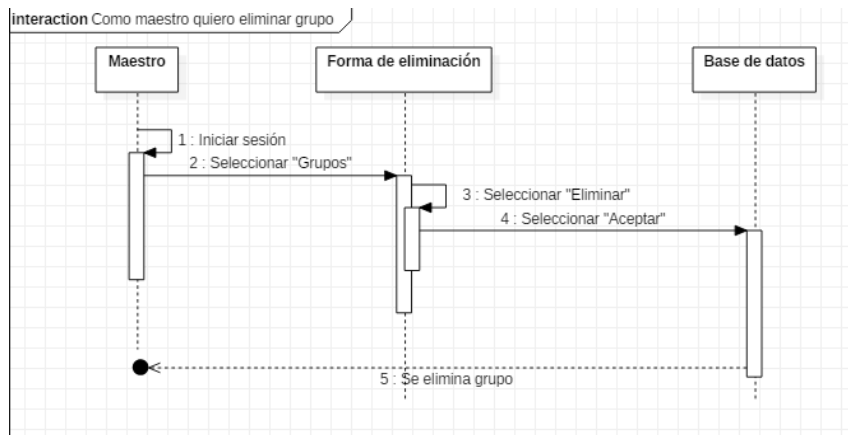
Inscribir Alumno



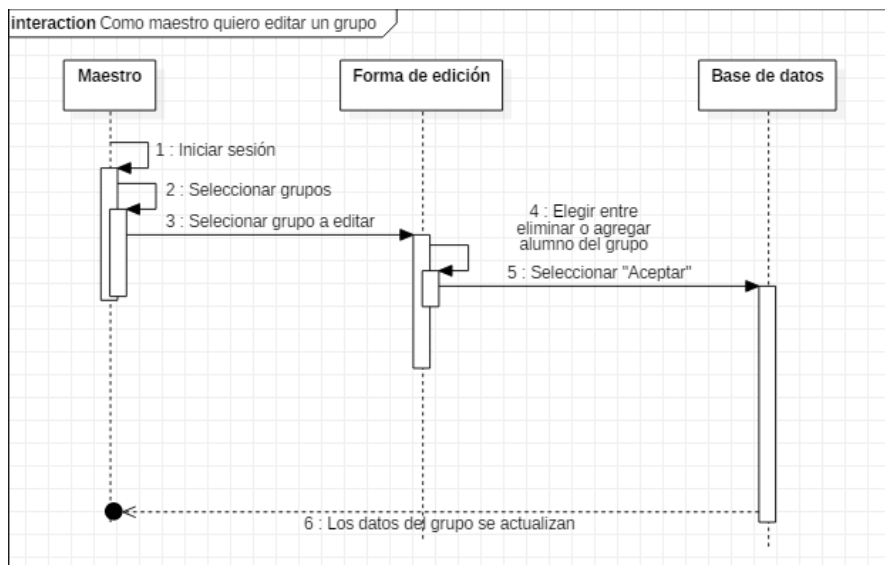
Crear grupo



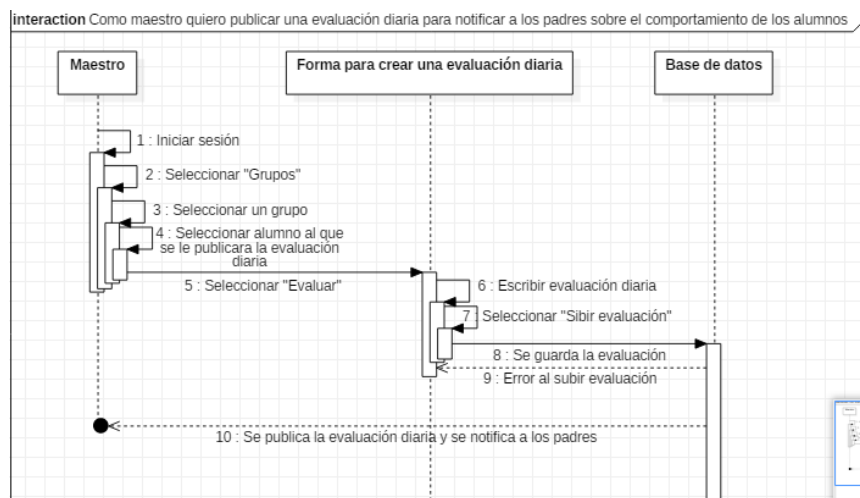
Eliminar grupo



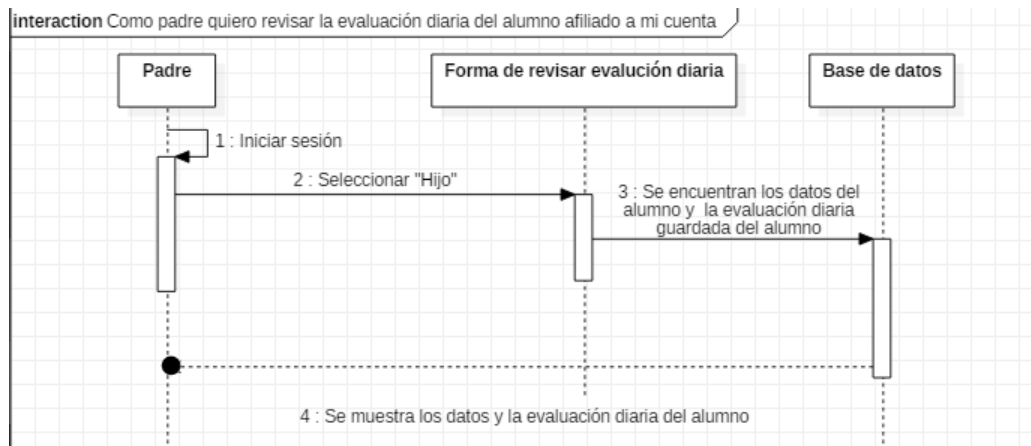
Editar grupo



Evaluación diaria



Revisar evaluación diaria



Casos de prueba de Login

```

#username
Length:
    Null. [property empty]
    <6. [property invalid]
    6. [single]
Content:
    Alphanumeric Characters. [if !empty]
#password
Length:
    Empty. [property empty]
    <6. [property invalid]
    6. [single]
Content:
    Valid Characters. [if !empty]
  
```

```

Test Case 1          <single>
  Length : 6

Test Case 2          <single>
  Length : 6

Test Case 3          (Key = 1.0.1.0.)
  Length : Null
  Content : <n/a>
  Length : Empty
  Content : <n/a>


























Test Case 4          (Key = 1.0.2.0.)
  Length : Null
  Content : <n/a>
  Length : <6
  Content : <n/a>

Test Case 5          (Key = 2.1.1.0.)
  Length : <6
  Content : Alphanumeric Characters
  Length : Empty
  Content : <n/a>

Test Case 6          (Key = 2.1.2.1.)
  Length : <6
  Content : Alphanumeric Characters
  Length : <6
  Content : Valid Characters
  
```

Archivos necesarios

Lista de archivos en carpeta principal:

 EnviarCorreo.php	Update EnviarCorreo.php	4 days ago
 ListaAlumnos.php	Terminado Evaluacion Diaria	14 days ago
 Manual de usuario.docx	Manual de usuario	4 days ago
 ManuallInstalacion.txt	Manual de instalación	6 days ago
 alumnos.php	crud completo sin poder cambiar imagen	15 days ago
 basedatos.php	Primer commit	a month ago
 borrarAlumno.php	crud alumnos 0.9	15 days ago
 borrarGrupo.php	Borrar grupo	a month ago
 calendario.html	Agrega una funcion	8 days ago
 correo.php	Update correo.php	7 days ago
 datosAlumno.php	crud completo sin poder cambiar imagen	15 days ago
 evaluacionDiaria.php	Cambios evaluacion	7 days ago
 eventos.php	Eventos	15 days ago
 grupos.php	Evaluacion Diaria	15 days ago
 guardarCambios.php	edite el calendario padres para que puedan ver la descripcion de los ...	8 days ago
 hijo.php	Update hijo.php	6 days ago
 imprimirImagen.php	crud completo sin poder cambiar imagen	15 days ago
 index.php	Update index.php	7 days ago
 login.php	Botón de inicio sesion	8 days ago
 logout.php	Primer commit	a month ago
 registroAlumno.php	edite el calendario padres para que puedan ver la descripcion de los ...	8 days ago
 registroGrupo.php	Ya agregué el registroGrupo	a month ago
 registroMaestro.php	Primer commit	a month ago
 registroPadre.php	Agregar email al registrar padre	22 days ago
 signup.php	Botones	8 days ago

Para el funcionamiento del módulo calendario de PocketCare, se utilizó código de libre uso encontrado en <https://www.fullcalendar.io/> El código utilizado fue:

```
36 <script>
37     $(document).ready(function(){
38         $('#Calendario').fullCalendar({
39             header:{
40                 left:'today,prev,next',
41                 center:'title',
42                 right:'month,basicWeek,basicDay,agendaWeek,agendaDay'
43             },
44
45             dayClick:function(date,jsEvent,view){
46
47                 $('#btnAgregar').prop("disabled", false);
48                 $('#btnModificar').prop("disabled", true);
49                 $('#btnEliminar').prop("disabled", true);
50
51                 limpiarFormulario();
52                 $('#txtFecha').val(date.format());
53                 $('#ModalEventos').modal();
54             },
55
56             events: 'http://localhost/PocketCare/eventos.php',
57
58             eventClick:function(calEvent,jsEvent,view){
59
60                 $('#btnAgregar').prop("disabled", true);
61                 $('#btnModificar').prop("disabled", false);
62                 $('#btnEliminar').prop("disabled", false);
63
64                 // H2
65                 $('#tituloEvento').html(calEvent.title);
66                 // Mostrar la información del evento en los inputs
67                 $('#txtDescripcion').val(calEvent.descripcion);
68                 $('#txtID').val(calEvent.id);
69                 $('#txtTitulo').val(calEvent.title);
70                 $('#txtColor').val(calEvent.color);
71                 FechaHora = calEvent.start._i.split(" ");
72                 $('#txtFecha').val(FechaHora[0]);
73                 $('#ModalEventos').modal();
74             },
75             editable:true,
76             eventDrop:function(calEvent){
77                 $('#txtID').val(calEvent.id);
78                 $('#txtTitulo').val(calEvent.title);
79                 $('#txtColor').val(calEvent.color);
80                 $('#txtDescripcion').val(calEvent.descripcion);
81                 var fechaHora = calEvent.start.format().split("T");
82                 $('#txtFecha').val(fechaHora[0]);
83                 $('#txtHora').val(fechaHora[1]);
84
85                 RecolectarDatosGUI();
86                 EnviarInformacion('modificar', NuevoEvento, true);
87             });
88     });
89 }
```

Snippets del código

Conexión con base de datos

```
<?php
$server = 'localhost';
$username = 'root';
$password = '';
$dbase = 'bd_pocketcare';

try {
    $conn = new PDO("mysql:host=$server; dbname=$dbase;", $username, $password); /* almacena conexión con la base de datos */
} catch (PDOException $e) {
    die('Error: ' . $e->getMessage());
}
?>
```

Borrar alumno

```
1 <?php require 'basedatos.php';
2
3 header('Location: alumnos.php');
4 $id = isset($_GET["id"]) ? $_GET["id"] : 0;
5 $sql = "DELETE FROM alumnos WHERE ID = {$id}";
6 $conn->query($sql);
7 die();
```

Borrar grupo

```
1 <?php require 'basedatos.php';
2
3 header('Location: grupos.php');
4 $id = isset($_GET["id"]) ? $_GET["id"] : 0;
5 $sql = "DELETE FROM grupos WHERE ID = {$id}";
6 $conn->query($sql);
7 die();
```

Correo

```
1 <?php
2
3 require "PHPMailer/Exception.php";
4 require "PHPMailer/PHPMailer.php";
5 require "PHPMailer/SMTP.php";
6
7 use PHPMailer\PHPMailer\PHPMailer;
8 use PHPMailer\PHPMailer\Exception;
9
10 $para = $_POST['to'];
11 $asunto = $_POST['asunto'];
12
13
14 $m = new PHPMailer;
15
16 $m->isSMTP();
17 $m->SMTPAuth = true;
18
19 $m->Host = "smtp.gmail.com";
20 $m->Username = "pocketcare@gmail.com";
21 $m->Password = "cuidate22";
22 $m->SMTPSecure = "tls";
23 $m->Port = "587";
24
25 $m->SMTPOptions = array(
26     'ssl' => array(
27         'verify_peer' => false,
28         'verify_peer_name' => false,
29         'allow_self_signed' => true
30     )
31 );
32
33 $m->isHTML();
34
35
36 $m->Subject = $asunto;
37 $m->Body = "<h2>Hola, se ha realizado una modificación en el calendario de PocketCare. <br>Visitar PocketCare para checar los cambios.</h2>";
38
39 $m->FromName = "PocketCare";
40
41 $m->addAddress($para, 'Usuario');
42
43 if($m->send()){
44     <script type="text/javascript">
45         alert("Enviado...");
46         window.close();
47     </script>
48 }else{
49     <script type="text/javascript">
50         alert("<?php echo $m->ErrorInfo; ?>");
51         window.close();
52     </script>
53 <?php }
54
55 ?>
56
```

Registro alumno

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <title>Sign Up</title>
  <link href="https://fonts.googleapis.com/css?family=Roboto" rel="stylesheet">
  <link rel="stylesheet" href="assets/css/style.css">
</head>
<body>
  <?php require 'partials/header.php'?>

  <!--</header><!-- /header >>

  </header><!-- /header >>-->

  <a href="index.php" ></a>

  <?php if(!empty($message)): ?>
    <p> <?= $message ?></p>
  <?php endif; ?>

  <h1>Registrar Alumno</h1>

  <form action="registroAlumno.php" method="post">
    <t> Seleccionar imagen del alumno </t><br>

    <input name = "Imagen" type="file" name="imagen">
    <input name="Nombre" type="text" placeholder="Primer nombre">
    <input name="Nombre2" type="text" placeholder="Segundo nombre (opcional)">
    <input name="Apellido" type="text" placeholder="Primer Apellido">
    <input name="Apellido2" type="text" placeholder="Segundo Apellido">
    <input name="Edad" type="text" placeholder="Edad">
    <input name="TipoSangre" type="text" placeholder="Tipo de sangre">
    <input name="Alergias" type="text" placeholder="Alergias">
    <input type="submit" value="submit">
  </form>

</body>
</html>
```

Registro grupo

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <title>Sign Up</title>
  <link href="https://fonts.googleapis.com/css?family=Roboto" rel="stylesheet">
  <link rel="stylesheet" href="assets/css/style.css">
</head>
<body>
  <?php require 'partials/header.php'?>

  <!--</header><!-- /header >>

  </header><!-- /header >>-->

  <a href="index.php" ></a>

  <?php if(!empty($message)): ?>
    <p> <?= $message ?></p>
  <?php endif; ?>

  <h1>Registrar grupo</h1>

  <form action="registroGrupo.php" method="POST">
    <input name="Nombre" type="text" placeholder="Nombre del grupo">
    <input type="submit" value="Submit">
  </form>

</body>
</html>
```

Login

```
session_start();

require 'basedatos.php';

if (empty($_POST['usuario']) && empty($_POST['contraseña'])) {
    $records = $conn->prepare('SELECT id, tipocuenta, nombreusuario, contrasena FROM cuentas WHERE nombreusuario = :usuario');
    $records->bindParam(':usuario', $_POST['usuario']);
    $records->execute();
    $results = $records->fetch(PDO::FETCH_ASSOC);

    $message = '';

    if (/*count($results) > 0 && password_verify($_POST['contraseña'], $results['contrasena'])*/ $_POST['contraseña'] == $results['contrasena']) {
        $_SESSION['id_usuario'] = $results['id'];
        header('location: /PocketCare');
    } else {
        $message = 'Lo sentimos, los datos no coinciden';
    }
}

<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title>Login</title>
    <link href="https://fonts.googleapis.com/css?family=Roboto" rel="stylesheet">
    <link rel="stylesheet" href="assets/css/style.css">
</head>
<body>

    <?php require 'partials/header.php' ?>
    <a href="index.php" ></a>
    <h1>Login</h1>
    <input type="reset" value="Sign Up" onclick="location='signup.php'">

    <?php if(empty($message)): ?>
    <p> <?= $message ?></p>
    <?php endif; ?>

    <form action="login.php" method="POST">
        <input type="text" name="usuario" placeholder="Ingresa su nombre de usuario">
        <input type="password" name="contraseña" placeholder="Ingresa su contraseña">
        <input type="submit" value="Iniciar Sesión">
    </form>
```