



Energiza tu vida con

MANUAL TECNICO

Luis Gerardo Marcelino Tax Mantanico

DIAGRAMA ENTIDAD RELACION

En este proyecto, se utiliza un diagrama entidad-relación (ERD) para representar cómo interactúan las entidades, tales como pacientes, objetos y conceptos. Este modelo conceptual de datos facilita la visualización de las relaciones entre los elementos clave del software que estás desarrollando. Al utilizar el diagrama ERD, puedes comprender mejor cómo se conectan y se relacionan estos elementos dentro del sistema.

<https://drive.google.com/file/d/1i2lWKW0yrJy7YRrDcicQI3vgLnu8HWWp/view?usp=sharing>

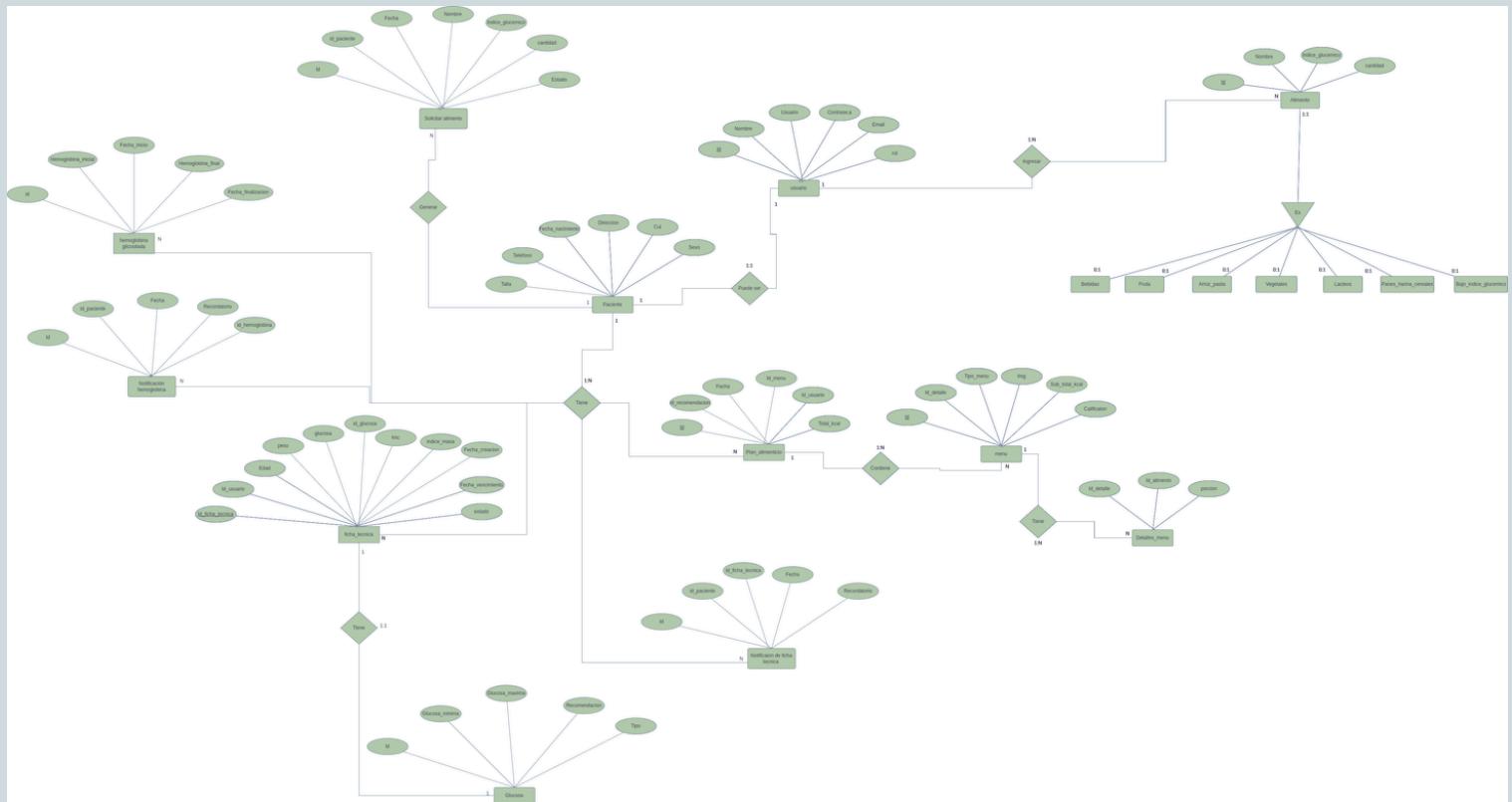
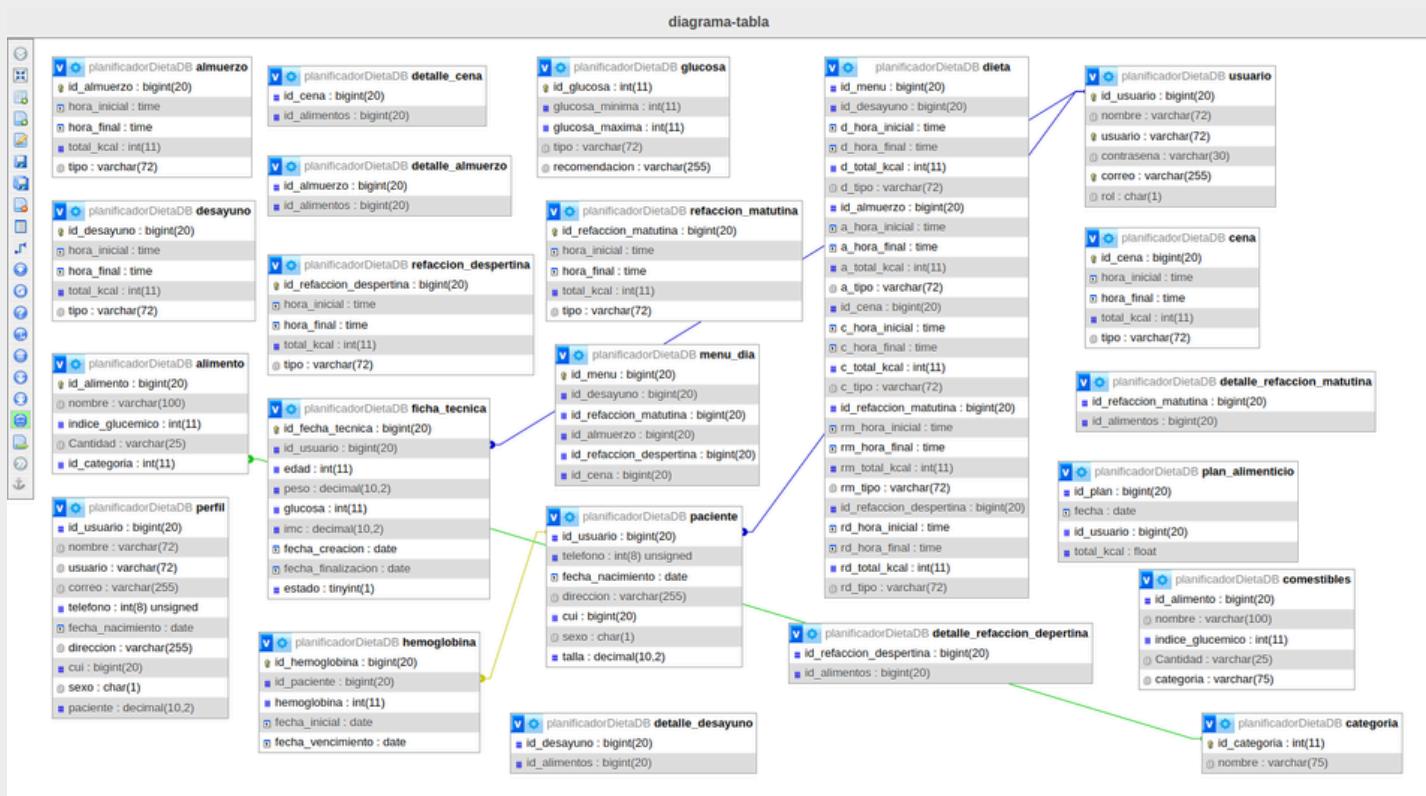


DIAGRAMA DE TABLA

Una tabla en base de datos es un tipo de modelado de datos donde se guarda una información recogida por un sistema. Es decir, son objetos o estructuras que contienen los datos organizados en filas y columnas. Cada fila representa un registro único, y cada columna un campo dentro del registro.



MAPEO FÍSICO DE LA BASE DE DATOS

LOS MAPAS FÍSICOS PROPORCIONAN DETALLES ESPECÍFICOS SOBRE EL NÚMERO DE BASES Y LA DISTANCIA FÍSICA QUE EXISTE ENTRE LOS MARCADORES GENÉTICOS. EL MAPEO CITOGENÉTICO ES UN MÉTODO UTILIZADO PARA CONSTRUIR MAPAS FÍSICOS QUE UTILIZA SECCIONES TEÑIDAS DE CROMOSOMAS PARA APROXIMAR LA DISTANCIA ENTRE LOS MARCADORES GENÉTICOS

BASE DE DATOS: 'PLANIFICADORDIETADB'

-- PROCEDIMIENTOS

DELIMITER \$\$

```
CREATE PROCEDURE 'HISTORIAL_FICHA_TECNICA' (IN 'ID' BIGINT) SELECT F.EDAD,
F.PESO, F.GLUCOSA, G.GLUCOSA_MINIMA, G.GLUCOSA_MAXIMA, G.TIPO, F.IMC,
F.FECHA_CREACION, F.FECHA_FINALIZACION, F.ESTADO
FROM FICHA_TECNICA F
INNER JOIN GLUCOSA G
ON F.GLUCOSA BETWEEN G.GLUCOSA_MINIMA AND G.GLUCOSA_MAXIMA
WHERE F.ID_USUARIO = ID
ORDER BY F.FECHA_CREACION ASC$$
```

```
CREATE PROCEDURE 'INSERTAR_PACIENTE' (IN 'P_NOMBRE' VARCHAR(72), IN
'P_USUARIO' VARCHAR(72), IN 'P CONTRASENA' VARCHAR(30), IN 'P_CORREO'
VARCHAR(255), IN 'P_ROL' CHAR(1), IN 'P_TELEFONO' INT(8), IN 'P_FECHA_NACIMIENTO'
DATE, IN 'P_DIRECCIÓN' VARCHAR(255), IN 'P_CUI' BIGINT, IN 'P_SEXO' CHAR(1), IN
'P_TALLA' DECIMAL(10,2)) BEGIN
DECLARE ID LONG;
CALL INSERTAR_USUARIO(P_NOMBRE, P_USUARIO, P_CONTRASENA, P_CORREO,
P_ROL, ID);
INSERT INTO PACIENTE
(ID_USUARIO, TELEFONO, FECHA_NACIMIENTO, DIRECCION, CUI, SEXO, TALLA)
VALUES (ID, P_TELEFONO, P_FECHA_NACIMIENTO, P_DIRECCION,
P_CUI, P_SEXO, P_TALLA);
END$$
```

```
CREATE PROCEDURE 'INSERTAR_USUARIO' (IN 'P_NOMBRE' VARCHAR(72), IN
'P_USUARIO' VARCHAR(72), IN 'P CONTRASENA' VARCHAR(30), IN 'P_CORREO'
VARCHAR(255), IN 'P_ROL' CHAR(1), OUT 'P_ID_USUARIO' BIGINT) BEGIN
INSERT INTO USUARIO (NOMBRE, USUARIO, CONTRASENA, CORREO, ROL)
VALUES (P_NOMBRE, P_USUARIO, P_CONTRASENA, P_CORREO, P_ROL);
SET P_ID_USUARIO = LAST_INSERT_ID();
END$$
```

```
CREATE PROCEDURE 'INSERT_FICHA_TECNICA' (IN 'ID_USUARIO' BIGINT, IN 'EDAD'
DECIMAL(10,2), IN 'PESO' DECIMAL(10,2), IN 'GLUCOSA' INT) BEGIN
INSERT INTO FICHA_TECNICA ('ID_USUARIO', 'EDAD', 'PESO', 'GLUCOSA', 'IMC',
'ESTADO')
VALUES (ID_USUARIO, EDAD, PESO, GLUCOSA, OBTENER_IMC(54, 1), TRUE);
END$$
```

-- FUNCIONES

```
CREATE FUNCTION 'LOGIN' ('CORREO' VARCHAR(255), 'CONTRASENA' VARCHAR(30))
RETURNS BIGINT(20) BEGIN
DECLARE ID BIGINT;
SELECT ID_USUARIO INTO ID
FROM USUARIO WHERE USUARIO.CORREO = CORREO AND
USUARIO.CONTRASENA=CONTRASENA;
RETURN ID;
END$$
```

```
CREATE FUNCTION `OBTENER_IMC` ('PESO' DECIMAL(10,2), 'ID' BIGINT) RETURNS  
DECIMAL(10,2) BEGIN  
DECLARE IMC DECIMAL(10,2);  
DECLARE ALTURA DECIMAL(10,2);  
SELECT TALLA INTO ALTURA FROM PACIENTE WHERE ID_USUARIO=ID;  
SET IMC = PESO / (ALTURA *ALTURA);  
RETURN IMC;  
END$$  
DELIMITER ;
```

-- ESTRUCTURA DE TABLA PARA LA TABLA 'ALIMENTO'

```
CREATE TABLE `ALIMENTO` (  
`ID_ALIMENTO` BIGINT(20) NOT NULL,  
`NOMBRE` VARCHAR(100) NOT NULL,  
`INDICE_GLUCEMICO` INT(11) NOT NULL,  
`CANTIDAD` VARCHAR(25) NOT NULL,  
`ID_CATEGORIA` INT(11) NOT NULL  
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;
```

```
CREATE TABLE `ALMUERZO` (  
`ID_ALMUERZO` BIGINT(20) NOT NULL,  
`HORA_INICIAL` TIME NOT NULL,  
`HORA_FINAL` TIME NOT NULL,  
`TOTAL_KCAL` INT(11) NOT NULL,  
`TIPO` VARCHAR(72) DEFAULT NULL  
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;
```

```
CREATE TABLE `CATEGORIA` (  
`ID_CATEGORIA` INT(11) NOT NULL,  
`NOMBRE` VARCHAR(75) NOT NULL  
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;
```

```
CREATE TABLE `CENA` (  
`ID_CENA` BIGINT(20) NOT NULL,  
`HORA_INICIAL` TIME NOT NULL,  
`HORA_FINAL` TIME NOT NULL,  
`TOTAL_KCAL` INT(11) NOT NULL,  
`TIPO` VARCHAR(72) DEFAULT NULL  
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;
```

```
CREATE TABLE `COMESTIBLES` (  
`ID_ALIMENTO` BIGINT(20)  
`NOMBRE` VARCHAR(100)  
`INDICE_GLUCEMICO` INT(11)  
`CANTIDAD` VARCHAR(25)  
`CATEGORIA` VARCHAR(75)  
);
```

```
CREATE TABLE `DESAYUNO` (  
`ID_DESAYUNO` BIGINT(20) NOT NULL,  
`HORA_INICIAL` TIME NOT NULL,  
`HORA_FINAL` TIME NOT NULL,  
`TOTAL_KCAL` INT(11) NOT NULL,  
`TIPO` VARCHAR(72) DEFAULT NULL  
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;
```

```
CREATE TABLE `DETALLE_ALMUERZO` (  
`ID_ALMUERZO` BIGINT(20) NOT NULL,  
`ID_ALIMENTOS` BIGINT(20) NOT NULL  
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;
```

```
CREATE TABLE `DETALLE_CENA` (  
`ID_CENA` BIGINT(20) NOT NULL,  
`ID_ALIMENTOS` BIGINT(20) NOT NULL  
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;
```

```

CREATE TABLE `DETALLE_DESAYUNO` (
  `ID_DESAYUNO` BIGINT(20) NOT NULL,
  `ID_ALIMENTOS` BIGINT(20) NOT NULL
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;

CREATE TABLE `DETALLE_REFACCION_DEPERTINA` (
  `ID_REFACCION_DESPERTINA` BIGINT(20) NOT NULL,
  `ID_ALIMENTOS` BIGINT(20) NOT NULL
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;

CREATE TABLE `DETALLE_REFACCION_MATUTINA` (
  `ID_REFACCION_MATUTINA` BIGINT(20) NOT NULL,
  `ID_ALIMENTOS` BIGINT(20) NOT NULL
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;

CREATE TABLE `DIETA` (
  `ID_MENU` BIGINT(20),
  `ID_DESAYUNO` BIGINT(20),
  `D_HORA_INICIAL` TIME,
  `D_HORA_FINAL` TIME,
  `D_TOTAL_KCAL` INT(11),
  `D_TIPO` VARCHAR(72),
  `ID_ALMUERZO` BIGINT(20),
  `A_HORA_INICIAL` TIME,
  `A_HORA_FINAL` TIME,
  `A_TOTAL_KCAL` INT(11),
  `A_TIPO` VARCHAR(72),
  `ID_CENA` BIGINT(20),
  `C_HORA_INICIAL` TIME,
  `C_HORA_FINAL` TIME,
  `C_TOTAL_KCAL` INT(11),
  `C_TIPO` VARCHAR(72),
  `ID_REFACCION_MATUTINA` BIGINT(20),
  `RM_HORA_INICIAL` TIME,
  `RM_HORA_FINAL` TIME,
  `RM_TOTAL_KCAL` INT(11),
  `RM_TIPO` VARCHAR(72),
  `ID_REFACCION_DESPERTINA` BIGINT(20),
  `RD_HORA_INICIAL` TIME,
  `RD_HORA_FINAL` TIME,
  `RD_TOTAL_KCAL` INT(11),
  `RD_TIPO` VARCHAR(72)
);

CREATE TABLE `FICHA_TECNICA` (
  `ID_FECHA_TECNICA` BIGINT(20) NOT NULL,
  `ID_USUARIO` BIGINT(20) NOT NULL,
  `EDAD` INT(11) NOT NULL,
  `PESO` DECIMAL(10,2) NOT NULL,
  `GLUCOSA` INT(11) NOT NULL,
  `IMC` DECIMAL(10,2) NOT NULL,
  `FECHA_CREACION` DATE DEFAULT CURDATE(),
  `FECHA_FINALIZACION` DATE NOT NULL,
  `ESTADO` TINYINT(1) NOT NULL
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;
-- DISPARADORES `FICHA_TECNICA`
DELIMITER $$

CREATE TRIGGER `CALCULAR_FECHA` BEFORE INSERT ON `FICHA_TECNICA` FOR EACH ROW BEGIN
    SET NEW.FECHA_FINALIZACION = DATE_ADD(NEW.FECHA_CREACION, INTERVAL 7 DAY);
END
$$
DELIMITER ;

```

```
CREATE TABLE `GLUCOSA` (
  `ID_GLUCOSA` INT(11) NOT NULL,
  `GLUCOSA_MINIMA` INT(11) NOT NULL,
  `GLUCOSA_MAXIMA` INT(11) NOT NULL,
  `TIPO` VARCHAR(72) NOT NULL,
  `RECOMENDACION` VARCHAR(255) NOT NULL
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;
```

```
CREATE TABLE `HEMOGLOBINA` (
  `ID_HEMOGLOBINA` BIGINT(20) NOT NULL,
  `ID_PACIENTE` BIGINT(20) NOT NULL,
  `HEMOGLOBINA` INT(11) NOT NULL,
  `FECHA_INICIAL` DATE DEFAULT CURDATE(),
  `FECHA_VENCIMIENTO` DATE DEFAULT NULL
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;
```

```
-- DISPARADORES 'HEMOGLOBINA'
```

```
DELIMITER $$  
CREATE TRIGGER `VENCIMIENTO_HEMOGLOBINA` BEFORE INSERT ON  
`HEMOGLOBINA` FOR EACH ROW BEGIN  
    SET NEW.FECHA_VENCIMIENTO = DATE_ADD(NEW.FECHA_INICIAL, INTERVAL 30  
DAY);  
END  
$$  
DELIMITER ;
```

```
CREATE TABLE `MENU_DIA` (  
  `ID_MENU` BIGINT(20) NOT NULL,  
  `ID_DESAYUNO` BIGINT(20) NOT NULL,  
  `ID_REFACCION_MATUTINA` BIGINT(20) NOT NULL,  
  `ID_ALMUERZO` BIGINT(20) NOT NULL,  
  `ID_REFACCION_DESPERTINA` BIGINT(20) NOT NULL,  
  `ID_CENA` BIGINT(20) NOT NULL  
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;
```

```
CREATE TABLE `PACIENTE` (  
  `ID_USUARIO` BIGINT(20) NOT NULL,  
  `TELEFONO` INT(8) UNSIGNED NOT NULL,  
  `FECHA_NACIMIENTO` DATE NOT NULL,  
  `DIRECCION` VARCHAR(255) NOT NULL,  
  `CUI` BIGINT(20) NOT NULL,  
  `SEXO` CHAR(1) NOT NULL,  
  `TALLA` DECIMAL(10,2) NOT NULL  
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;
```

```
CREATE TABLE `PERFIL` (  
  `ID_USUARIO` BIGINT(20)  
  ,`NOMBRE` VARCHAR(72)  
  ,`USUARIO` VARCHAR(72)  
  ,`CORREO` VARCHAR(255)  
  ,`TELEFONO` INT(8) UNSIGNED  
  ,`FECHA_NACIMIENTO` DATE  
  ,`DIRECCION` VARCHAR(255)  
  ,`CUI` BIGINT(20)  
  ,`SEXO` CHAR(1)  
  ,`PACIENTE` DECIMAL(10,2)  
);
```

```
CREATE TABLE `PLAN_ALIMENTICIO` (
  `ID_PLAN` BIGINT(20) NOT NULL,
  `FECHA` DATE NOT NULL,
  `ID_USUARIO` BIGINT(20) NOT NULL,
  `TOTAL_KCAL` FLOAT NOT NULL
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;
```

```
CREATE TABLE `REFACCION_DESPERTINA` (
  `ID_REFACCION_DESPERTINA` BIGINT(20) NOT NULL,
  `HORA_INICIAL` TIME NOT NULL,
  `HORA_FINAL` TIME NOT NULL,
  `TOTAL_KCAL` INT(11) NOT NULL,
  `TIPO` VARCHAR(72) DEFAULT NULL
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;
```

```
CREATE TABLE `REFACCION_MATUTINA` (
  `ID_REFACCION_MATUTINA` BIGINT(20) NOT NULL,
  `HORA_INICIAL` TIME NOT NULL,
  `HORA_FINAL` TIME NOT NULL,
  `TOTAL_KCAL` INT(11) NOT NULL,
  `TIPO` VARCHAR(72) DEFAULT NULL
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;
```

```
CREATE TABLE `USUARIO` (
  `ID_USUARIO` BIGINT(20) NOT NULL,
  `NOMBRE` VARCHAR(72) NOT NULL,
  `USUARIO` VARCHAR(72) NOT NULL,
  `CONTRASENA` VARCHAR(30) NOT NULL,
  `CORREO` VARCHAR(255) NOT NULL,
  `ROL` CHAR(1) NOT NULL
) ENGINE=INNODB DEFAULT CHARSET=UTF8MB4 COLLATE=UTF8MB4_UNICODE_CI;
```

```
DROP TABLE IF EXISTS `COMESTIBLES`;
```

```
CREATE ALGORITHM=UNDEFINED DEFINER='ROOT'@'LOCALHOST' SQL SECURITY DEFINER VIEW `COMESTIBLES` AS SELECT `AL`.`ID_ALIMENTO` AS `ID_ALIMENTO`, `AL`.`NOMBRE` AS `NOMBRE`, `AL`.`INDICE_GLUCEMICO` AS `INDICE_GLUCEMICO`, `AL`.`CANTIDAD` AS `CANTIDAD`, `CA`.`NOMBRE` AS `CATEGORIA` FROM (`ALIMENTO` `AL` JOIN `CATEGORIA` `CA` ON(`CA`.`ID_CATEGORIA` = `AL`.`ID_CATEGORIA`)) ;
```

```
-- ESTRUCTURA PARA LA VISTA `DIETA`
```

```
DROP TABLE IF EXISTS `DIETA`;
```

```
CREATE ALGORITHM=UNDEFINED DEFINER='ROOT'@'LOCALHOST' SQL SECURITY DEFINER VIEW `DIETA` AS SELECT `M`.`ID_MENU` AS `ID_MENU`, `D`.`ID_DESAYUNO` AS `ID_DESAYUNO`, `D`.`HORA_INICIAL` AS `D_HORA_INICIAL`, `D`.`HORA_FINAL` AS `D_HORA_FINAL`, `D`.`TOTAL_KCAL` AS `D_TOTAL_KCAL`, `D`.`TIPO` AS `D_TIPO`, `A`.`ID_ALMUERZO` AS `ID_ALMUERZO`, `A`.`HORA_INICIAL` AS `A_HORA_INICIAL`, `A`.`HORA_FINAL` AS `A_HORA_FINAL`, `A`.`TOTAL_KCAL` AS `A_TOTAL_KCAL`, `A`.`TIPO` AS `A_TIPO`, `C`.`ID_CENA` AS `ID_CENA`, `C`.`HORA_INICIAL` AS `C_HORA_INICIAL`, `C`.`HORA_FINAL` AS `C_HORA_FINAL`, `C`.`TOTAL_KCAL` AS `C_TOTAL_KCAL`, `C`.`TIPO` AS `C_TIPO`, `RM`.`ID_REFACCION_MATUTINA` AS `ID_REFACCION_MATUTINA`, `RM`.`HORA_INICIAL` AS `RM_HORA_INICIAL`, `RM`.`HORA_FINAL` AS `RM_HORA_FINAL`, `RM`.`TOTAL_KCAL` AS `RM_TOTAL_KCAL`, `RM`.`TIPO` AS `RM_TIPO`, `RD`.`ID_REFACCION_DESPERTINA` AS `ID_REFACCION_DESPERTINA`, `RD`.`HORA_INICIAL` AS `RD_HORA_INICIAL`, `RD`.`HORA_FINAL` AS `RD_HORA_FINAL`, `RD`.`TOTAL_KCAL` AS `RD_TOTAL_KCAL`, `RD`.`TIPO` AS `RD_TIPO` FROM (((`MENU_DIA` `M` JOIN `DESAYUNO` `D` ON(`M`.`ID_DESAYUNO` = `D`.`ID_DESAYUNO`)) JOIN `ALMUERZO` `A` ON(`M`.`ID_ALMUERZO` = `A`.`ID_ALMUERZO`)) JOIN `CENA` `C` ON(`M`.`ID_CENA` = `C`.`ID_CENA`)) JOIN `REFACCION_MATUTINA` `RM` ON(`RM`.`ID_REFACCION_MATUTINA` = `M`.`ID_REFACCION_MATUTINA`)) JOIN `REFACCION_DESPERTINA` `RD` ON(`RD`.`ID_REFACCION_DESPERTINA` = `M`.`ID_REFACCION_DESPERTINA`)) ;
```

```
DROP TABLE IF EXISTS `PERFIL`;
```

```

CREATE ALGORITHM=UNDEFINED DEFINER='ROOT'@'LOCALHOST' SQL SECURITY
DEFINER VIEW `PERFIL` AS SELECT `U`.`ID_USUARIO` AS `ID_USUARIO`, `U`.`NOMBRE`
AS `NOMBRE`, `U`.`USUARIO` AS `USUARIO`, `U`.`CORREO` AS `CORREO`,
`P`.`TELEFONO` AS `TELEFONO`, `P`.`FECHA_NACIMIENTO` AS `FECHA_NACIMIENTO`,
`P`.`DIRECCION` AS `DIRECCION`, `P`.`CUI` AS `CUI`, `P`.`SEXO` AS `SEXO`, `P`.`TALLA` AS
`PACIENTE` FROM (`USUARIO` `U` LEFT JOIN `PACIENTE` `P` ON(`P`.`ID_USUARIO` =
`U`.`ID_USUARIO`)) ;
ALTER TABLE `ALIMENTO`
    ADD PRIMARY KEY (`ID_ALIMENTO`),
    ADD KEY `ID_CATEGORIA` (`ID_CATEGORIA`);
ALTER TABLE `ALMUERZO`
    ADD PRIMARY KEY (`ID_ALMUERZO`);
ALTER TABLE `CATEGORIA`
    ADD PRIMARY KEY (`ID_CATEGORIA`);
ALTER TABLE `CENA`
    ADD PRIMARY KEY (`ID_CENA`);
ALTER TABLE `DESAYUNO`
    ADD PRIMARY KEY (`ID_DESAYUNO`);
ALTER TABLE `FICHA_TÉCNICA`
    ADD PRIMARY KEY (`ID_FECHA_TECNICA`),
    ADD KEY `ID_USUARIO` (`ID_USUARIO`);

ALTER TABLE `GLUCOSA`
    ADD PRIMARY KEY (`ID_GLUCOSA`);
ALTER TABLE `HEMOGLOBINA`
    ADD PRIMARY KEY (`ID_HEMOGLOBINA`),
    ADD KEY `ID_PACIENTE` (`ID_PACIENTE`);
ALTER TABLE `MENU_DIA`
    ADD PRIMARY KEY (`ID_MENU`);
ALTER TABLE `PACIENTE`
    ADD KEY `ID_USUARIO` (`ID_USUARIO`);
ALTER TABLE `REFACCION_DESPERTINA`
    ADD PRIMARY KEY (`ID_REFACCION_DESPERTINA`);
ALTER TABLE `REFACCION_MATUTINA`
    ADD PRIMARY KEY (`ID_REFACCION_MATUTINA`);

ALTER TABLE `USUARIO`
    ADD PRIMARY KEY (`ID_USUARIO`),
    ADD UNIQUE KEY `USUARIO` (`USUARIO`),
    ADD UNIQUE KEY `CORREO` (`CORREO`);
-- AUTO_INCREMENT DE LAS TABLAS VOLCADAS
ALTER TABLE `ALIMENTO`
    MODIFY `ID_ALIMENTO` BIGINT(20) NOT NULL AUTO_INCREMENT,
    AUTO_INCREMENT=114;

-- -- AUTO_INCREMENT DE LA TABLA `ALMUERZO`
-- ALTER TABLE `ALMUERZO`
--     MODIFY `ID_ALMUERZO` BIGINT(20) NOT NULL AUTO_INCREMENT,
--     AUTO_INCREMENT=3;

-- -- AUTO_INCREMENT DE LA TABLA `CATEGORIA`
-- ALTER TABLE `CATEGORIA`
--     MODIFY `ID_CATEGORIA` INT(11) NOT NULL AUTO_INCREMENT,
--     AUTO_INCREMENT=12;

-- -- AUTO_INCREMENT DE LA TABLA `CENA`
-- ALTER TABLE `CENA`
--     MODIFY `ID_CENA` BIGINT(20) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=3;

```

```
-- AUTO_INCREMENT DE LA TABLA 'DESAYUNO'
-- ALTER TABLE 'DESAYUNO'
  MODIFY `ID_DESAYUNO` BIGINT(20) NOT NULL AUTO_INCREMENT,
  AUTO_INCREMENT=3;

-- AUTO_INCREMENT DE LA TABLA 'FICHA_TECNICA'
-- ALTER TABLE 'FICHA_TECNICA'
  MODIFY `ID_FECHA_TECNICA` BIGINT(20) NOT NULL AUTO_INCREMENT,
  AUTO_INCREMENT=4;

-- AUTO_INCREMENT DE LA TABLA 'GLUCOSA'
-- ALTER TABLE 'GLUCOSA'
  MODIFY `ID_GLUCOSA` INT(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=6;

-- AUTO_INCREMENT DE LA TABLA 'HEMOGLOBINA'
-- ALTER TABLE 'HEMOGLOBINA'
  MODIFY `ID_HEMOGLOBINA` BIGINT(20) NOT NULL AUTO_INCREMENT;

-- AUTO_INCREMENT DE LA TABLA 'MENU_DIA'
-- ALTER TABLE 'MENU_DIA'
  MODIFY `ID_MENU` BIGINT(20) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=3;

-- AUTO_INCREMENT DE LA TABLA 'REFACCION_DESPERTINA'
-- ALTER TABLE 'REFACCION_DESPERTINA'
  MODIFY `ID_REFACCION_DESPERTINA` BIGINT(20) NOT NULL AUTO_INCREMENT,
  AUTO_INCREMENT=3;

-- AUTO_INCREMENT DE LA TABLA 'REFACCION_MATUTINA'
-- ALTER TABLE 'REFACCION_MATUTINA'
  MODIFY `ID_REFACCION_MATUTINA` BIGINT(20) NOT NULL AUTO_INCREMENT,
  AUTO_INCREMENT=3;
```

```
-- AUTO_INCREMENT DE LA TABLA 'USUARIO'  
--  
ALTER TABLE 'USUARIO'  
    MODIFY 'ID_USUARIO' BIGINT(20) NOT NULL AUTO_INCREMENT,  
    AUTO_INCREMENT=18;  
  
-- RESTRICCIONES PARA TABLAS VOLCADAS  
--  
  
-- FILTROS PARA LA TABLA 'ALIMENTO'  
--  
ALTER TABLE 'ALIMENTO'  
    ADD CONSTRAINT 'ALIMENTO_IBFK_1' FOREIGN KEY ('ID_CATEGORIA')  
REFERENCES 'CATEGORIA' ('ID_CATEGORIA');  
  
-- FILTROS PARA LA TABLA 'FICHA_TECNICA'  
--  
ALTER TABLE 'FICHA_TECNICA'  
    ADD CONSTRAINT 'FICHA_TECNICA_IBFK_1' FOREIGN KEY ('ID_USUARIO')  
REFERENCES 'USUARIO' ('ID_USUARIO');  
  
-- FILTROS PARA LA TABLA 'HEMOGLOBINA'  
--  
ALTER TABLE 'HEMOGLOBINA'  
    ADD CONSTRAINT 'HEMOGLOBINA_IBFK_1' FOREIGN KEY ('ID_PACIENTE')  
REFERENCES 'PACIENTE' ('ID_USUARIO');  
  
-- FILTROS PARA LA TABLA 'PACIENTE'  
--  
ALTER TABLE 'PACIENTE'  
    ADD CONSTRAINT 'PACIENTE_IBFK_1' FOREIGN KEY ('ID_USUARIO')  
REFERENCES 'USUARIO' ('ID_USUARIO');  
COMMIT;
```

DIAGRAMA DE PAQUETE

EN NUESTRO PROYECTO, UTILIZAMOS DIAGRAMAS DE PAQUETES PARA REFLEJAR LA ORGANIZACIÓN DE LOS PAQUETES Y SUS ELEMENTOS. ESTOS DIAGRAMAS SON FUNDAMENTALES PARA PROPORCIONAR UNA VISUALIZACIÓN CLARA DE LOS ESPACIOS DE NOMBRES, ESPECIALMENTE CUANDO LOS EMPLEAMOS PARA REPRESENTAR LOS ELEMENTOS DE CLASE. LOS DIAGRAMAS DE PAQUETES SON HERRAMIENTAS CLAVE EN LA ORGANIZACIÓN DE OTROS DIAGRAMAS UML, COMO LOS DIAGRAMAS DE CASOS DE USO Y LOS DIAGRAMAS DE CLASE. AUNQUE SU USO MÁS COMÚN ES EN ESTOS CONTEXTOS, LOS DIAGRAMAS DE PAQUETES TIENEN UNA FLEXIBILIDAD QUE PERMITE APLICARLOS A UNA VARIEDAD DE ELEMENTOS UML, ADAPTÁNDOSE A LAS NECESIDADES ESPECÍFICAS DE NUESTRO PROYECTO.



DIAGRAMA DE DESPLIEGUE

UN DIAGRAMA DE DESPLIEGUE ES UN TIPO DE DIAGRAMA UML QUE MUESTRA LA ARQUITECTURA DE EJECUCIÓN DE UN SISTEMA, INCLUYENDO NODOS COMO ENTORNOS DE EJECUCIÓN DE HARDWARE O SOFTWARE, Y EL MIDDLEWARE QUE LOS CONECTA.

LOS DIAGRAMAS DE DESPLIEGUE SE UTILIZAN NORMALMENTE PARA VISUALIZAR EL HARDWARE Y EL SOFTWARE FÍSICO DE UN SISTEMA, USÁNDOLO PUEDES ENTENDER CÓMO EL SISTEMA SE DESPLEGARÁ FÍSICAMENTE EN EL HARDWARE.

LOS DIAGRAMAS DE DESPLIEGUE AYUDAN A MODELAR LA TOPOLOGÍA DE HARDWARE DE UN SISTEMA EN COMPARACIÓN CON OTROS TIPOS DE DIAGRAMAS UML, QUE EN SU MAYORÍA ESBOZAN LOS COMPONENTES LÓGICOS DE UN SISTEMA.

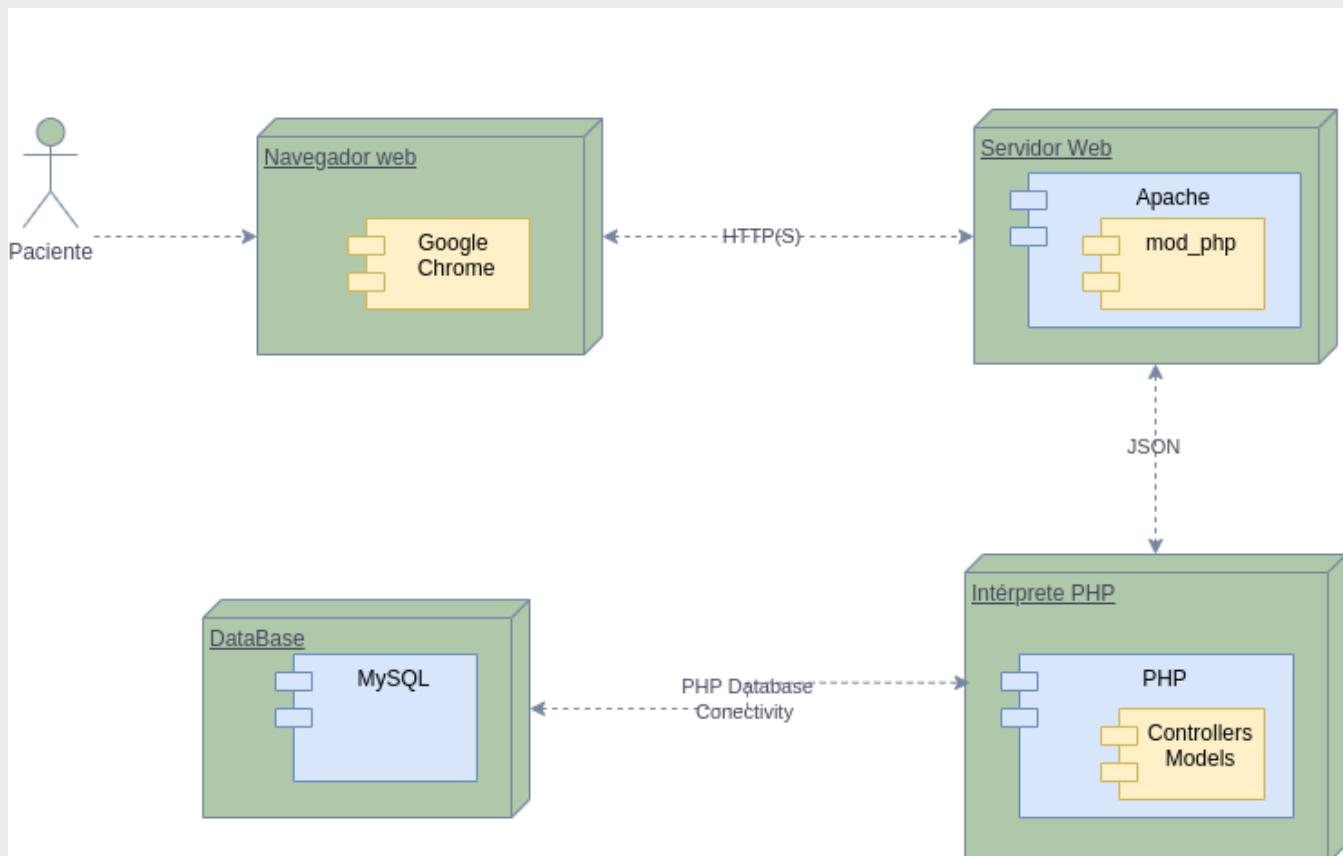
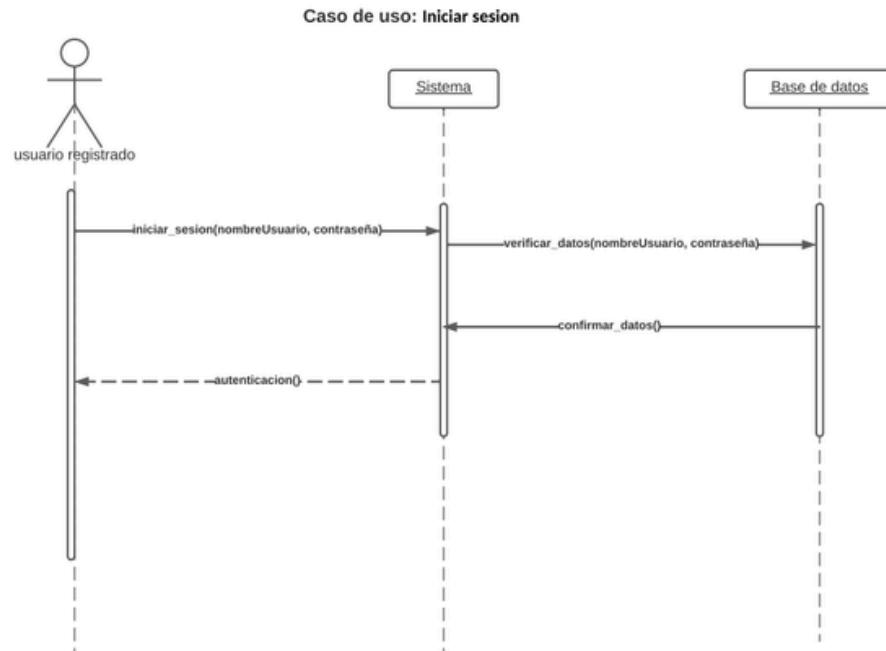


DIAGRAMA DE SECUENCIA DE LOS CASOS DE USO



Caso de uso: Registro de usuario

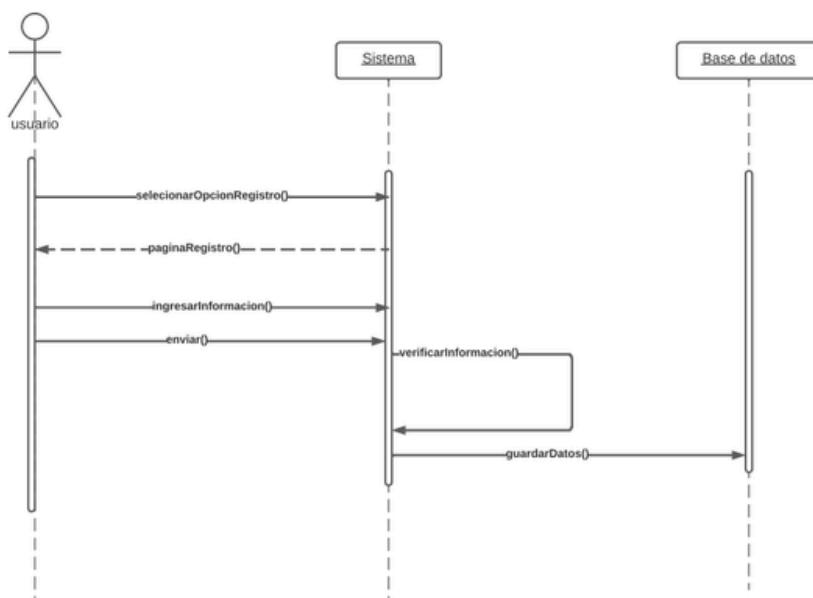
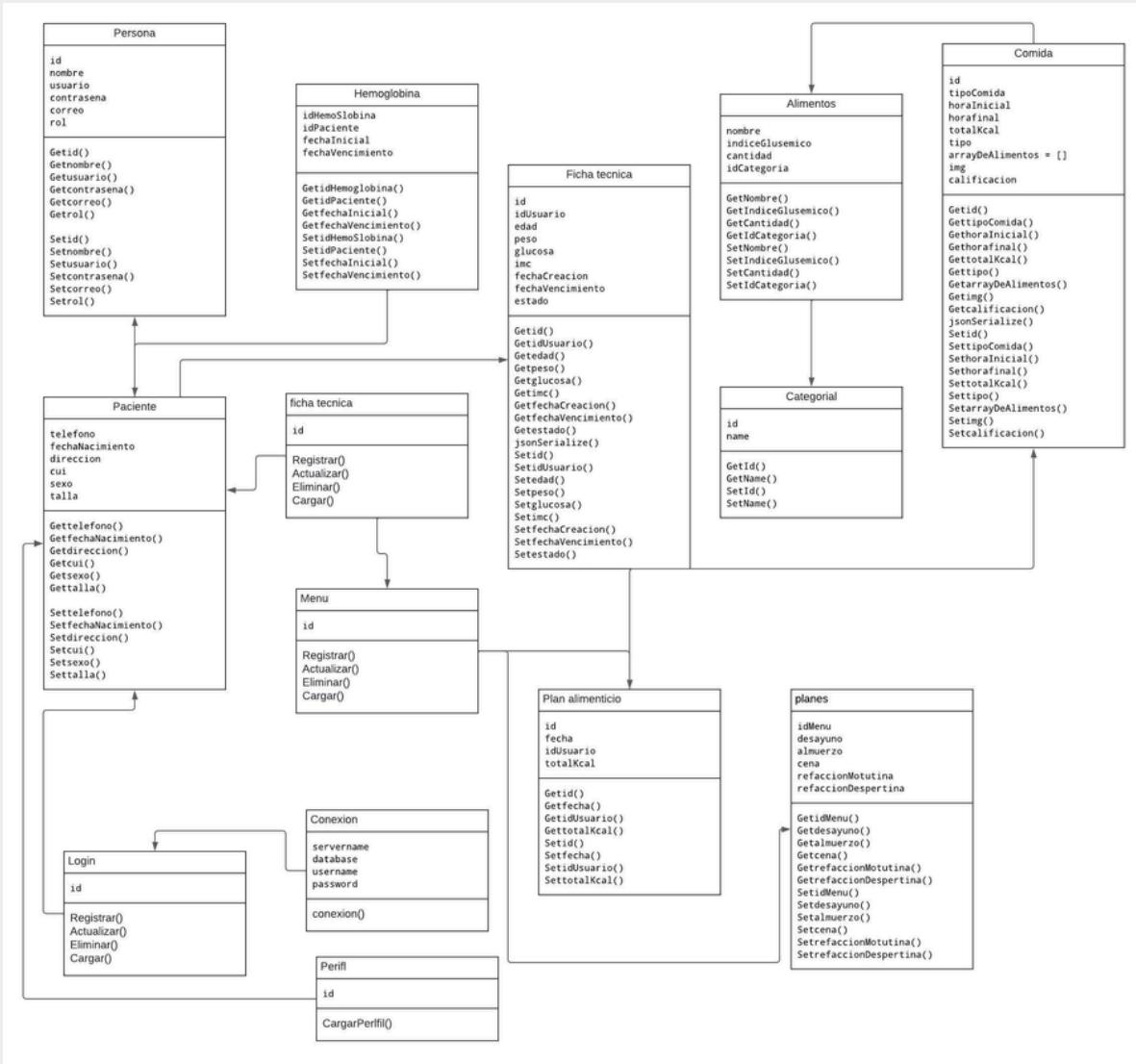


DIAGRAMA DE DESPLIEGUE

UN DIAGRAMA DE CLASES ES UNA ESTRUCTURA ESTÁTICA QUE SE USA PARA MOSTRAR LOS TIPOS DE RELACIONES ENTRE LOS OBJETOS QUE SE ESTÁN PROGRAMANDO. TAMBIÉN ES UNA BUENA MANERA DE MOSTRAR LA ESTRUCTURA DE CLASES EN UN SISTEMA. EN EL CASO DE LOS NEGOCIOS, SE PUEDEN UTILIZAR PARA ILUSTRAR LAS RELACIONES DE CLASE EN UNA APLICACIÓN EMPRESARIAL. UN DIAGRAMA DE CLASES ES ESPECIALMENTE ÚTIL PARA COMUNICAR LAS JERARQUIAS DE CLASES Y LAS COLABORACIONES DE CLASES CON LAS PARTES INTERESADAS O UN EQUIPO.



Caso de uso de alto nivel

Numero:	CU001
Caso de uso	Registro Usuario
Actores	Usuario
Descripción	El sistema permite al usuario nuevo crear una cuenta proporcionando información personal y credenciales. El usuario completa un formulario de registro. Tras enviar el formulario, el sistema valida la información, crea la cuenta y confirma el registro al usuario.
Tipo	Primario

Numero:	CU002
Caso de uso	Iniciar sesión
Actores	Usuario
Descripción	El sistema permite al usuario registrado acceder a su cuenta utilizando su correo y contraseña. Tras la autenticación exitosa, el usuario puede acceder a las funcionalidades y servicios del sistema según sus permisos.
Tipo	Primario

Numero:	CU003
Caso de uso	Ver Perfil
Actores	Usuario
Descripción	El sistema permite al usuario registrado visualizar la información de su perfil, incluyendo datos personales y configuraciones de cuenta. El usuario accede a su perfil a través de la interfaz del sistema y puede revisar la información detallada que se muestra en la página de perfil.
Tipo	Primario

Numero:	CU004
Caso de uso	Editar perfil
Actores	Usuario
Descripción	El sistema permite al usuario registrado modificar la información de su perfil. El usuario accede a la sección de edición del perfil, realiza los cambios deseados en sus datos personales y configuraciones, y guarda las modificaciones. El sistema valida la información actualizada y la guarda en la base de datos. Tipo: Primario
Tipo	Primario

Numero:	CU005
Caso de uso	Historial de Dietas
Actores	Usuario (paciente)
Descripción	El sistema permite al paciente revisar el historial de dietas

	previamente asignadas y su efectividad.
Tipo	Primario

Numero:	CU006
Caso de uso	Solicitud de nueva dieta
Actores	Usuario (paciente)
Descripción	El sistema permite al paciente solicitar una nueva dieta basada en su ficha técnica y en el historial de dietas
Tipo	Primario

Numero:	CU007
Caso de uso	Generar plan de dieta
Actores	Usuario(paciente) Sistema
Descripción	El sistema genera un plan de dieta personalizado para el paciente basado en la información de la ficha técnica y en la hemoglobina glicosilada y en las solicitudes de nuevas dietas.
Tipo	Primario

Numero:	CU008
Caso de uso	Revisar Dieta Generada
Actores	Administrador
Descripción	El sistema permite al administrador revisar y aprobar el plan de dieta generado para el paciente antes de su entrega.
Tipo	Primario

Numero:	CU009
Caso de uso	Enviar Plan de dieta al paciente
Actores	Sistema
Descripción	El sistema envía el plan de dieta aprobado al paciente por correo electrónico o a través de la plataforma del sistema. Tipo: Primario
Tipo	Primario

Numero:	CU010
Caso de uso	Ver dieta actual
Actores	Usuario
Descripción	El sistema permite al paciente acceder a una visión completa de su dieta actual asignada, incluyendo una lista detallada de los alimentos permitidos y las restricciones específicas. Además, ofrece información adicional como el valor nutricional de cada alimento, las cantidades recomendadas y la frecuencia de las comidas
Tipo	Primario

Numero:	CU011
---------	-------

Caso de uso	llenar ficha tecnica
Actores	Usuario(paciente)
Descripción	El sistema solicitará al paciente que increse su información personal y credenciales. El usuario completa un formulario de registro. Tras enviar el formulario, el sistema valida la información, crea una ficha técnica del usuario.
Tipo	Primario

Numero:	CU012
Caso de uso	Editar ficha tecnica
Actores	Usuario (paciente)
Descripción	El sistema permitirá al paciente acceder a su ficha técnica existente y realizar modificaciones. El usuario selecciona la opción para editar la ficha técnica, completa los cambios necesarios en el formulario presentado, y envía las modificaciones. El sistema valida la información editada y actualiza la ficha técnica del usuario con los nuevos datos proporcionados
Tipo	Primario

Numero:	CU013
Caso de uso	Actualizacion de ficha tecnica
Actores	Usuario(paciente)
Descripción	El sistema permite al paciente actualizar la información de su ficha técnica. El usuario accede a su ficha técnica y elige la opción para actualizarla. Luego, completa los campos del formulario con la información actualizada y envía el formulario. El sistema valida los datos proporcionados y actualiza la ficha técnica del usuario con la nueva información ingresada.
Tipo	Primario

Numero:	CU014
Caso de uso	Notificacion de actualizacion de ficha técnica
Actores	Usuario(paciente), Sistema
Descripción	El sistema enviará un recordatorio al paciente para que actualice su ficha técnica.
Tipo	Primario

Numero:	CU015
Caso de uso	Historial de ficha tecnica
Actores	Usuario(Paciente)
Descripción	El sistema permite al paciente consultar el historial de cambios realizados en su ficha técnica.
Tipo	Primario

Número:	CU016
Caso de uso	Registro de hemoglobina glicosilada
Actores	Usuario(Paciente)
Descripción	El sistema solicitará al paciente que increse su información personal y credenciales. El usuario completa un formulario de registro. Tras enviar el formulario, el sistema valida la información, crea una ficha técnica del usuario.
Tipo	Primario

Número:	CU017
Caso de uso	Editar hemoglobina glicosilada
Actores	Usuario(paciente)
Descripción	El sistema permitirá al paciente acceder a su hemoglobina glicosilada existente y realizar modificaciones. El usuario selecciona la opción para editar, completa los cambios necesarios en el formulario presentado, y envía las modificaciones. El sistema valida la información editada y actualiza la hemoglobina glicosilada del usuario con los nuevos datos proporcionados
Tipo	Primario

Número:	CU018
Caso de uso	Actualizar hemoglobina glicosilada
Actores	Usuario(paciente)
Descripción	El sistema permite al paciente actualizar la información de su hemoglobina glicosilada. El usuario accede a su hemoglobina glicosilada y elige la opción para actualizarla. Luego, completa los campos del formulario con la información actualizada y envía el formulario. El sistema valida los datos proporcionados y actualiza la hemoglobina glicosilada del usuario con la nueva información ingresada.
Tipo	Primario

Número:	CU019
Caso de uso	Notificar actualización de hemoglobina glicosilada
Actores	Usuario(paciente), Sistema
Descripción	El sistema enviará un recordatorio al paciente para que actualice su hemoglobina glicosilada.
Tipo	Primario

Número:	CU020
Caso de uso	Historial de hemoglobina glicosilada
Actores	Usuario(paciente)
Descripción	El sistema permite al paciente consultar el historial de cambios realizados en su hemoglobina glicosilada.

Tipo	Primario
-------------	----------

Numero:	CU021
Caso de uso	Ver Estadística de proceso
Actores	Usuario (paciente)
Descripción	El sistema permite al paciente visualizar estadísticas y gráficos relacionados con su progreso en el seguimiento de la dieta
Tipo	Primario

Numero:	CU022
Caso de uso	Agregar nuevos alimentos
Actores	Administrador
Descripción	El sistema permite al administrador agregar nuevos alimentos y sus propiedades nutricionales para incluirlos en los planes de dieta.
Tipo	Secundario

Numero:	CU023
Caso de uso	Eliminar alimentos
Actores	Administrador
Descripción	El sistema permite al administrador eliminar alimentos obsoletos o incorrectos del sistema de base de datos.
Tipo	Segundario

Numero:	CU024
Caso de uso	Editar alimentos
Actores	Administrador
Descripción	El sistema permite al administrador poder editar alimentos y sus propiedades nutricionales para incluirlos en los planes de dieta.
Tipo	Segundario

Numero:	CU025
Caso de uso	Historial de usuario
Actores	administrador
Descripción	El sistema permite al Administrador ver el historial de los usuario registrados en el sistema.
Tipo	Primario

Numero:	CU026
Caso de uso	Calificar dieta
Actores	Usuario(paciente)
Descripción	El sistema permite al paciente calificar su dieta actual para proporcionar retroalimentación sobre su efectividad y satisfacción.
Tipo	Primario

Numero:	CU027
Caso de uso	Reporte de las dietas mas gustas
Actores	administrador
Descripción	El sistema permite al administrador ver el reporte de las dietas mas gustadas.
Tipo	Primario

Numero:	CU028
Caso de uso	Exportacion de reporte de dietas mas gustadas
Actores	administrador
Descripción	El sistema permite al administrador exportar el reporte en formato PDF para su revisión y almacenamiento.
Tipo	Primario

Numero:	CU029
Caso de uso	Exportacion de plan de dieta
Actores	Usuario(paciente)
Descripción	El sistema permite al paciente exportar su plan de dieta en formato PDF para su revisión y almacenamiento.
Tipo	Primario

Numero:	CU030
Caso de uso	Desactivar Cuenta
Actores	Administrador
Descripción	El sistema permite al administrador desactivar cuentas de paciente
Tipo	Secundario

Numero:	CU031
Caso de uso	Historial de notificacion de actualizacion de ficha técnica
Actores	Usuario(paciente)
Descripción	El sistema permite al paciente ver el historial de las notificaciones de las ficha tecnicas registradas en el sistema.
Tipo	Primario

Numero:	CU032
Caso de uso	Historial de Notificacion de actualizacion de la hemoglobina glicosilada
Actores	Usuario(paciente)
Descripción	El sistema permite al paciente ver el historial de las

	notificaciones de las hemoglobinas glicosiladas registradas en el sistema.
Tipo	Primario

Numero:	CU033
Caso de uso	Carga de archivo
Actores	Administrador
Descripción	El administrar puede cargar diferentes tipos de archivos con extensión JSON, para luego poder utilizarlo en el sistema
Tipo	Primario

Numero:	CU034
Caso de uso	Carga de archivo
Actores	Administrador
Descripción	El administrar puede cargar diferentes tipos de archivos con extensión JSON, para luego poder utilizarlo en el sistema
Tipo	Primario

Numero:	CU035
Caso de uso	Descarga de reportes
Actores	Usuario
Descripción	Los diferentes tipos de usuario pueden descargar los reportes correspondientes en formato de pdf.
Tipo	segundario