**UNIVERSIDAD MARIANO GÁLVEZ DE GUATEMALA**



**FASE DE DISEÑO**

**ASESOR: INGENIERO** **ÁNGEL ATILIO MALTEZ CIFUENTES**

Estudiante: Danny Estuardo Ramírez Socop.

Facultad de ingeniería en sistemas de información y ciencias de la computación

Sección: “A”.

No. De carnet: 3090-19-15542

Curso: Análisis de sistemas ll

Mazatenango 07 de Julio 2022

**INTRODUCCION**

A continuacion se presenta la fase de diseño en la cual resaltamos elementos importantes para esta fase como: la arquitectura implementada que en este caso fue el modelo vista controlador, así también la codificación la de interfaz del login, también los componentes que se utilizaran para llevar a cabo el proyecto, la fase de diseño nos será útil para analizar y diseñar adecuadamente los que vamos a plantear en la fase de desarrollo.

**INDICE**

[ARQUITECTURA IMPLEMETADA 4](#_Toc113478868)

[MODELO VISTA CONTROLADOR 4](#_Toc113478869)

[PORQUE SE VA UTILIZAR 4](#_Toc113478870)

[VENTAJAS 4](#_Toc113478871)

[RESUMEN MODELO VISTA CONTROLADOR (MVC) 4](#_Toc113478872)

[Modelo 4](#_Toc113478873)

[Vista 4](#_Toc113478874)

[Controlador 4](#_Toc113478875)

[ESTRUCTURA DEL CODIGO 5](#_Toc113478876)

[JAVA 5](#_Toc113478877)

[LOGIN 5](#_Toc113478878)

[MYSQL 14](#_Toc113478879)

[COMPONENTES DEL SOFTWARE 14](#_Toc113478880)

[MySQL 14](#_Toc113478881)

[JAVA 14](#_Toc113478882)

[EXPERIANCIA DEL USUARIO 15](#_Toc113478883)

[KAHOOT 15](#_Toc113478884)

[HISTORIA DE USUARIO 18](#_Toc113478885)

[PLANIFICACION 19](#_Toc113478886)

[DIAGRAMA DE COMPONENTES 20](#_Toc113478887)

[LOGIN 21](#_Toc113478888)

[ADMINISTRACION 21](#_Toc113478889)

[REGISTRAR 22](#_Toc113478890)

[FUNDACION 22](#_Toc113478891)

[REPOSITORIO 23](#_Toc113478892)

[KAHOOT 23](#_Toc113478893)

[CONCLUSION 24](#_Toc113478894)

# ARQUITECTURA IMPLEMETADA

# MODELO VISTA CONTROLADOR

## PORQUE SE VA UTILIZAR

Considera que es una arquitectura completa ya que vamos a utilizar un **MODELO** cual va representar la información con la cual el sistema opera gestionando todos loa accesos a dicha información tanto en consulta como como actualizaciones a nuestra base de datos, también utilizaremos una **VISTA** que no es mas que la representación del modelo en forma de la interfaz de usuario(vista donde se presentara la información) y el **CONTROLADOR** es el que responderá a las acciones que el usuario quiera hacer(será el intermediario entre la vista y el modelo).

## VENTAJAS

* La separación del Modelo y la Vista, lo cual logra separar los datos, de su representación visual.
* Facilita el manejo de errores.
* la simplicidad del mantenimiento y la optimización de la escalabilidad de la aplicación

# RESUMEN MODELO VISTA CONTROLADOR (MVC)

Consiste en un patrón de diseño de software que se utiliza para separar en tres componentes los datos, la metodología y la interfaz gráfica de una aplicación. La gran ventaja que posee esta técnica de programación es que permite modificar cada uno de ellos sin necesidad de modificar los demás, lo que permite desarrollar aplicaciones modulares y escalables que se puedan actualizar fácilmente y añadir o eliminar nuevos módulos o funcionalidades de forma paquete utilizada, ya que cada “paquete” utiliza el mismo sistema con sus vistas, modelos y controladores.

## Modelo

El Modelo se encarga de manipular, gestionar y actualizar los datos. Si se utiliza una base de datos aquí es donde se realizan las consultas, búsquedas, filtros y actualizaciones.

## Vista

La Vista sirve para mostrar le al usuario final la interfaz gráfica (pantallas, ventanas, páginas, formularios).

## Controlador

El Controlador es el componente principal de la aplicación, donde se especifican los métodos y funcionalidades que una aplicación. A través del controlador se realizan las consultas al modelo e instrucciones necesarias para poder mostrarlos de una forma legible para el usuario.

# ESTRUCTURA DEL CODIGO

## JAVA

## LOGIN

package com.login;

import javax.swing.JOptionPane;

public class Login extends javax.swing.JFrame {

/\*\*

\* Creates new form Login

\*/

public Login() {

initComponents();

}

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel1 = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

jLabel2 = new javax.swing.JLabel();

fondo = new javax.swing.JLabel();

jLabel4 = new javax.swing.JLabel();

jLabel5 = new javax.swing.JLabel();

jSeparator1 = new javax.swing.JSeparator();

Usuario = new javax.swing.JTextField();

jSeparator3 = new javax.swing.JSeparator();

Password = new javax.swing.JPasswordField();

jPanel2 = new javax.swing.JPanel();

iniciar = new javax.swing.JButton();

jPanel3 = new javax.swing.JPanel();

jLabel3 = new javax.swing.JLabel();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

jPanel1.setBackground(new java.awt.Color(255, 255, 255));

jPanel1.setForeground(new java.awt.Color(204, 204, 204));

jPanel1.setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());

jLabel1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/com/imagenes/12.png"))); // NOI18N

jPanel1.add(jLabel1, new org.netbeans.lib.awtextra.AbsoluteConstraints(60, 80, 290, 270));

jLabel2.setFont(new java.awt.Font("Roboto Medium", 1, 24)); // NOI18N

jLabel2.setForeground(new java.awt.Color(255, 255, 255));

jLabel2.setText("ASILO CORAZON AMABLE");

jPanel1.add(jLabel2, new org.netbeans.lib.awtextra.AbsoluteConstraints(60, 370, 300, 20));

fondo.setIcon(new javax.swing.ImageIcon(getClass().getResource("/com/imagenes/photo-1538947151057-dfe933d688d1.jpg"))); // NOI18N

jPanel1.add(fondo, new org.netbeans.lib.awtextra.AbsoluteConstraints(0, 0, 420, 520));

jLabel4.setFont(new java.awt.Font("Roboto Light", 1, 14)); // NOI18N

jLabel4.setForeground(new java.awt.Color(0, 0, 0));

jLabel4.setText("USUARIO");

jPanel1.add(jLabel4, new org.netbeans.lib.awtextra.AbsoluteConstraints(470, 160, -1, -1));

jLabel5.setFont(new java.awt.Font("Roboto Light", 1, 14)); // NOI18N

jLabel5.setForeground(new java.awt.Color(0, 0, 0));

jLabel5.setText("CONTRASEÑA");

jPanel1.add(jLabel5, new org.netbeans.lib.awtextra.AbsoluteConstraints(470, 240, 100, -1));

jPanel1.add(jSeparator1, new org.netbeans.lib.awtextra.AbsoluteConstraints(470, 290, 300, 10));

Usuario.setBackground(new java.awt.Color(255, 255, 255));

Usuario.setForeground(new java.awt.Color(204, 204, 204));

Usuario.setBorder(null);

Usuario.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

UsuarioActionPerformed(evt);

}

});

jPanel1.add(Usuario, new org.netbeans.lib.awtextra.AbsoluteConstraints(470, 190, 310, -1));

jPanel1.add(jSeparator3, new org.netbeans.lib.awtextra.AbsoluteConstraints(470, 210, 300, 10));

Password.setBackground(new java.awt.Color(255, 255, 255));

Password.setBorder(null);

Password.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

PasswordActionPerformed(evt);

}

});

jPanel1.add(Password, new org.netbeans.lib.awtextra.AbsoluteConstraints(470, 270, 310, -1));

jPanel2.setBackground(new java.awt.Color(255, 255, 255));

jPanel2.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

iniciar.setBackground(new java.awt.Color(0, 153, 255));

iniciar.setFont(new java.awt.Font("Roboto Light", 1, 14)); // NOI18N

iniciar.setForeground(new java.awt.Color(51, 51, 51));

iniciar.setText("INICIAR");

iniciar.addMouseListener(new java.awt.event.MouseAdapter() {

public void mouseClicked(java.awt.event.MouseEvent evt) {

iniciarMouseClicked(evt);

}

});

iniciar.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

iniciarActionPerformed(evt);

}

});

javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);

jPanel2.setLayout(jPanel2Layout);

jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGap(30, 30, 30)

.addComponent(iniciar, javax.swing.GroupLayout.PREFERRED\_SIZE, 270, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(28, Short.MAX\_VALUE))

);

jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel2Layout.createSequentialGroup()

.addContainerGap(212, Short.MAX\_VALUE)

.addComponent(iniciar)

.addGap(33, 33, 33))

);

jPanel1.add(jPanel2, new org.netbeans.lib.awtextra.AbsoluteConstraints(460, 130, 330, 270));

jPanel3.setBackground(new java.awt.Color(102, 204, 255));

jPanel3.setBorder(javax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0, 0)));

jLabel3.setFont(new java.awt.Font("Roboto Medium", 1, 24)); // NOI18N

jLabel3.setForeground(new java.awt.Color(0, 0, 0));

jLabel3.setText("INICIAR SESION");

javax.swing.GroupLayout jPanel3Layout = new javax.swing.GroupLayout(jPanel3);

jPanel3.setLayout(jPanel3Layout);

jPanel3Layout.setHorizontalGroup(

Panel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel3Layout.createSequentialGroup()

.addGap(71, 71, 71)

.addComponent(jLabel3)

.addContainerGap(73, Short.MAX\_VALUE))

);

jPanel3Layout.setVerticalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel3Layout.createSequentialGroup()

.addGap(20, 20, 20)

.addComponent(jLabel3, javax.swing.GroupLayout.PREFERRED\_SIZE, 30, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(18, Short.MAX\_VALUE))

);

jPanel1.add(jPanel3, new org.netbeans.lib.awtextra.AbsoluteConstraints(460, 60, 330, 70));

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, 824, Short.MAX\_VALUE)

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel1, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

pack();

}// </editor-fold>

private void UsuarioActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

private void iniciarMouseClicked(java.awt.event.MouseEvent evt) {

//boton iniciar

String usuario = "admin";

String contraseña = "123";

String pass=new String(Password.getPassword());

if(Usuario.getText().equals(usuario) && pass.equals(contraseña)){

Administracion AD = new Administracion(); //objeto

AD.setVisible(true);

dispose();

}else{

JOptionPane.showMessageDialog(this, "Usuario o contraseña incorrecta");

}

}

private void PasswordActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

private void iniciarActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new Login().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JPasswordField Password;

private javax.swing.JTextField Usuario;

private javax.swing.JLabel fondo;

private javax.swing.JButton iniciar;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JLabel jLabel5;

private javax.swing.JPanel jPanel1;

private javax.swing.JPanel jPanel2;

private javax.swing.JPanel jPanel3;

private javax.swing.JSeparator jSeparator1;

private javax.swing.JSeparator jSeparator3;

// End of variables declaration

}

# ESTRUCTURA DEL CODIGO

# MYSQL

# COMPONENTES DEL SOFTWARE

Motor de base de datos: MySQL

Lenguaje de programación: Java

Entorno de desarrollo (ID): NetBeans

## MySQL

En programación es prácticamente inevitable trabajar con algún tipo de sistema de gestión de bases de datos. Cualquier programa que imaginemos tarde o temprano necesitará almacenar datos en algún lugar, como mínimo para poder almacenar la lista de usuarios autorizados, sus permisos y propiedades.

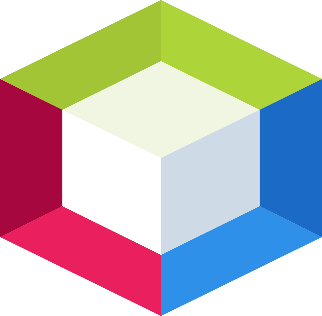
MySQL es el sistema de gestión de bases de datos relacional más extendido en la actualidad al estar basada en código abierto. Desarrollado originalmente por MySQL AB, fue adquirida por Sun MicroSystems en 2008 y esta su vez comprada por Oracle Corporation en 2010, la cual ya era dueña de un motor propio InnoDB para MySQL.

MySQL es un sistema de gestión de bases de datos que cuenta con una doble licencia. Por una parte es de código abierto, pero por otra, cuenta con una versión comercial gestionada por la compañía Oracle.

## JAVA

Java es una plataforma informática de lenguaje de programación creada por Sun Microsystems en 1995. Ha evolucionado desde sus humildes comienzos hasta impulsar una gran parte del mundo digital actual, ya que es una plataforma fiable en la que se crean muchos servicios y aplicaciones. Los nuevos e innovadores productos y servicios digitales diseñados para el futuro también siguen basándose en Java.

Aunque la mayoría de aplicaciones Java modernas combinan el tiempo de ejecución y la aplicación de Java, todavía existen algunas aplicaciones e incluso sitios web que no funcionan sin instalar Java para escritorio. El sitio web Java.com está pensado para consumidores que todavía necesitan Java en sus aplicaciones de escritorio, sobre todo las aplicaciones que tienen como destino Java 8. Los desarrolladores y los usuarios que quieran aprender sobre programación Java, deben consultar el sitio web dev.java y los usuarios profesionales, oracle.com/java para obtener más información.

**NEATBEANS**

NetBeans es un IDE (Integrated Development Environment) o entorno de desarrollo integrado, que es gratuito y de código abierto. Si quieres saber qué es Netbeans, en primera instancia, se debe destacar que sirve para el desarrollo de aplicaciones web, corporativas, de escritorio y móviles que utilizan plataformas como Java y HTML5, entre otras.

Pero para saber en concreto qué es NetBeans, debemos aclarar cómo se define un IDE como NetBeans. Es sencillo. Un IDE es una aplicación que tiene como función principal la optimización del desarrollo de software por parte de los programadores.

Se trata de un editor de código fuente, junto con recursos de construcción automáticos y un depurador. Igualmente, brinda la función de autocompletado inteligente de código o IntelliSense. Y en el caso de la plataforma de Netbeans IDE, ofrece un compilador y un intérprete.

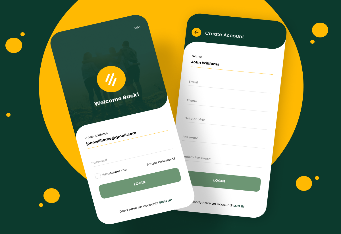
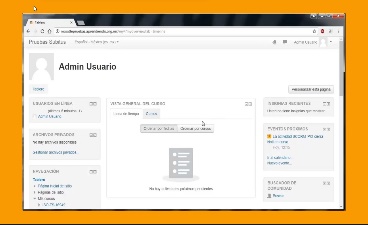
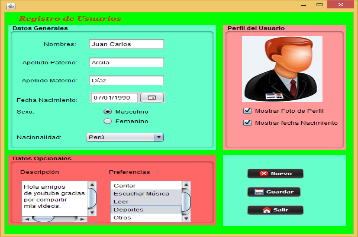
# EXPERIANCIA DEL USUARIO

La experiencia del usuario engloba todas aquellas interacciones del usuario con una marca. Por supuesto, a raíz de la transformación digital, esta experiencia se orienta principalmente a páginas web, aplicaciones móviles, herramientas y otros recursos similares.

Sí, en la actualidad, los altos índices de satisfacción ya no dependen enteramente de la calidad de los productos o servicios.

Para las empresas, esto implica un mayor esfuerzo por desarrollar una presencia digital coherente y atractiva y por adaptar cada vez más sus flujos de interacción a las características y el comportamiento del usuario.

## KAHOOT

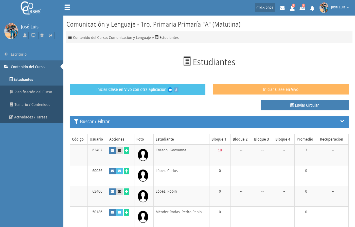
1. ¿El inicio de sesión en el software le fue fácil o difícil?
2. Fácil
3. Difícil
4. Normal
5. Intermedio
6. ¿le gusta el diseño del software?
7. Si
8. No
9. Normal
10. Ordinario
11. ¿Su experiencia en la plataforma ha sido confortable?
12. Excelente
13. Mala
14. Normal
15. ¿los colores de fondo son adecuados para usted?
16. Si
17. No
18. Mala
19. Muy buena
20. ¿Le fue fácil acceder a la plataforma?

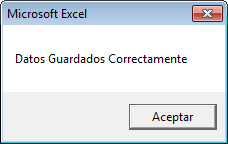
a) Si

b) No

c) Mala

d) Muy buena

1. ¿Del 1 al 10 cuanto le da al software?
2. 1-3
3. 4-5
4. 5-8
5. 8-10
6. ¿Las funciones que posee el software fueron fácil de usar?
7. Si
8. No
9. Malo
10. Muy bueno
11. ¿Podría mejor el software que se le presento?
12. Si
13. No
14. Muy bueno

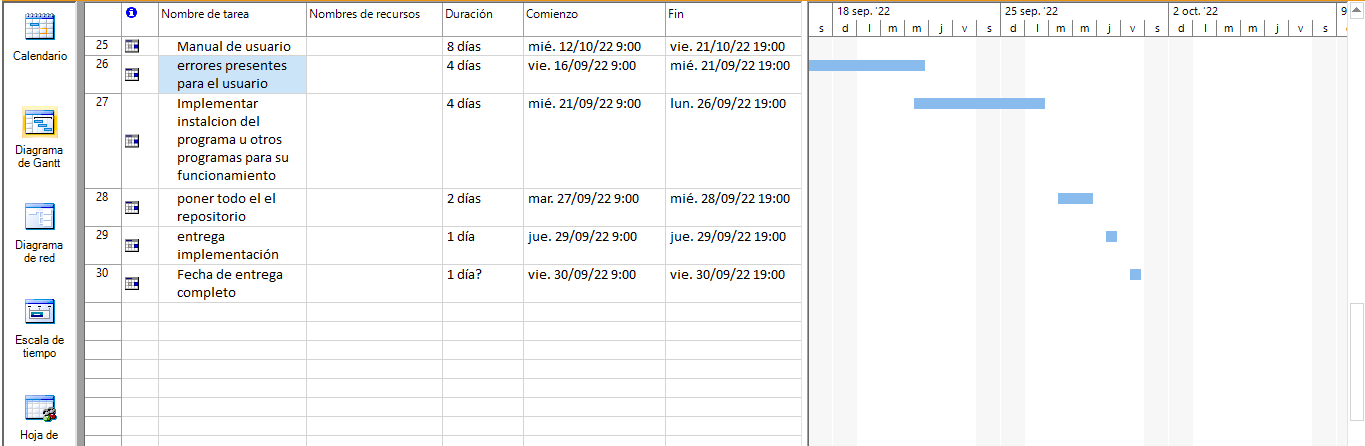
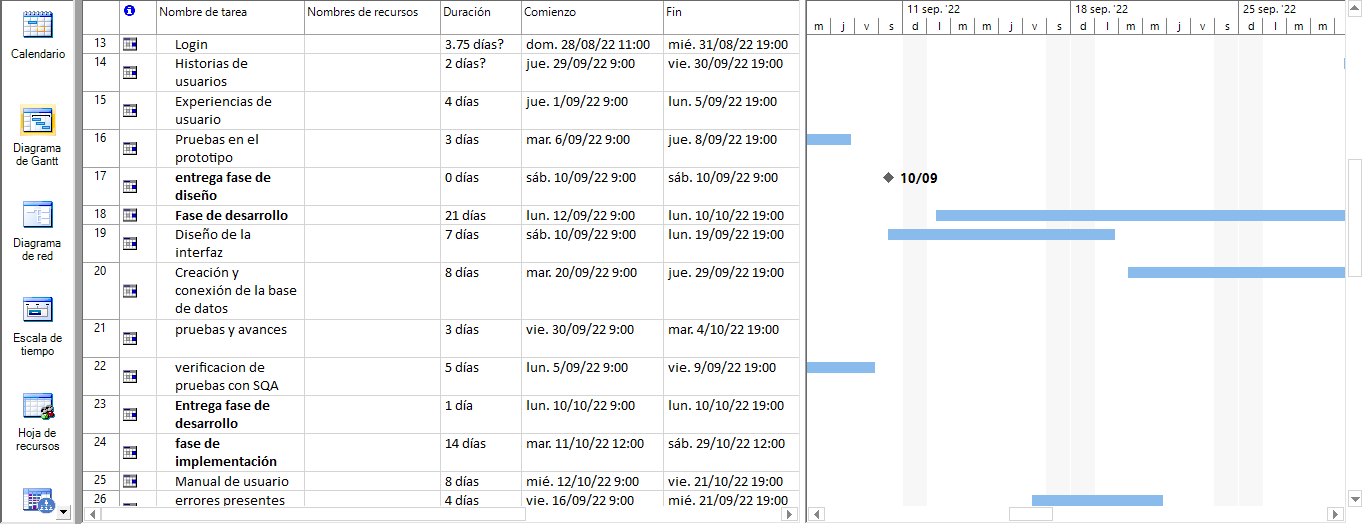
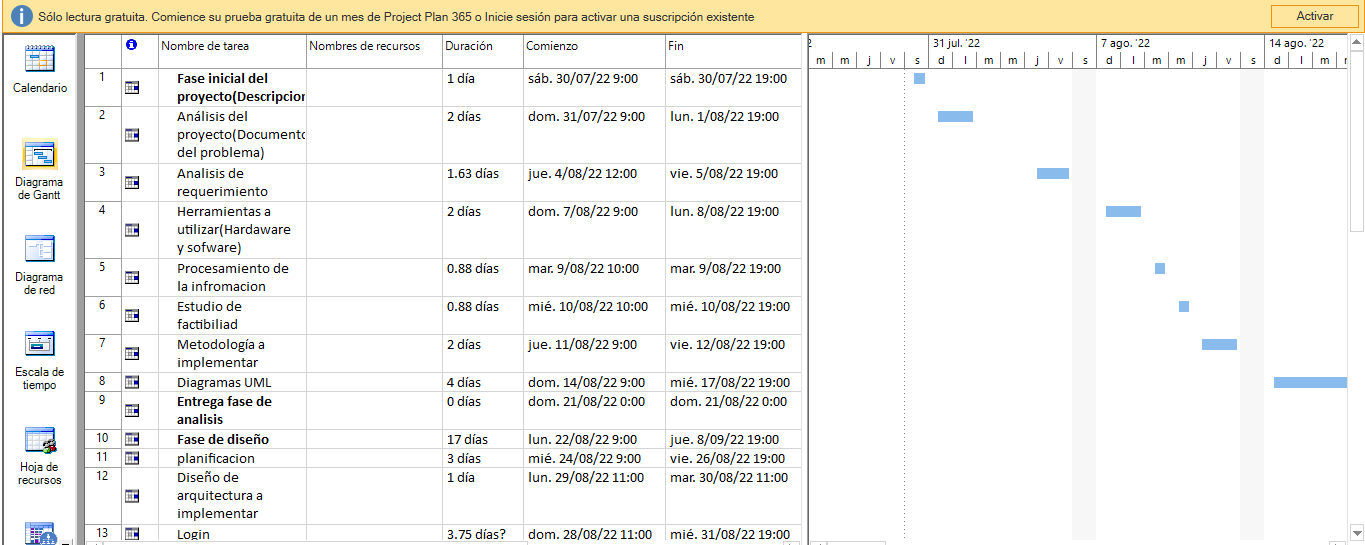
1. ¿Sus datos se guardan correctamente?
2. Si
3. No
4. ¿Sintió alguna molestia al utilizar el software?
5. Si
6. No
7. Dejar comentario

# HISTORIA DE USUARIO

Una historia de usuario es una explicación general e informal de una función de software escrita desde la perspectiva del usuario final. Su propósito es articular cómo proporcionará una función de software valor al cliente.

|  |  |
| --- | --- |
| **HISTORIO DE USUARIO** | |
| **Numero:** \* | **Usuario:** Administrador |
| **Nombre de Historio**: Registros de clientes | |
| **Prioridad del Negocio:** Alta | **Riesgo en Desarrollo:** Medio |
| **Programador Responsable:** \*\*\*\*\*\*\*\*\*\*\*\*\*\* | |
| **Descripción:** Yo como recepcionista, necesito un registro de clientes que me muestre los datos de todos los clientes que vienen al negocio. | |
| **Criterios de aceptación:** Los datos requeridos son: Nombre, Apellidos, CI, Fecha de Ingreso, y de Entrega | |

# PLANIFICACION



# DIAGRAMA DE COMPONENTES

# LOGIN

El usuario deberá de ingresar su usuario y contraseña correctamente y dar clic en el botón iniciar para dar inicio a la interfaz donde podrá ver todas las opciones de administración.

# ADMINISTRACION

Interfaz donde tendremos las opciones de entrar al registro de los nuevos internos también a los servicios de la fundación y la ficha de los internos.



# REGISTRAR

Esta interfaz nos servirá para agregar a los nuevos internos que ingresaran al asilo poder llevar control de toda su información personal que se ingrese.

# FUNDACION

En esta interfaz nos encontraremos los servicios que nos brinda la fundación tales como los médicos, Farmacia y Laboratorio cabe recalcar que cada servicio ofrece algo distinto.



# REPOSITORIO

<https://github.com/dannysocop/Tarea-Reposittorio.git>

# KAHOOT

<https://create.kahoot.it/share/preguntas-de-software/84d54dd6-4eb3-45e5-bfcc-da809852496c>

# CONCLUSION

La fase de diseño nos es muy útil para poder identificar factores como la utilización de la arquitectura que se está implantando los componentes a utilizar, así mismo también nos es útil para verificar las interfases que nuestro proyecto tendrá acabo y poder presentarlo en forma de un prototipo para que sean evaluadas por el SQA a cargo y poder presentar un ben proyecto y poder cumplir con todos los lineamientos que se están pidiendo.