|  |
| --- |
| *Documento de Arquitectura del software* |

Índice de Contenido

1 Introducción 3

2 Resumen Arquitectónico 3

2.1 Estilo Arquitectónico 3

2.2 Objetivos de la Arquitectura 3

3 Diagrama de Componentes. 4

3.1 View 6

3.2 Model 6

3.3 Data 7

3.4 Common 7

3.5 Tests 7

4 Vista de Casos de Uso 8

5 Diagrama de despliegue. 8

5.1 Visión General 8

6 Diagrama de Base de Datos 9

**Documento de Arquitectura del Software**

# Introducción

# Resumen Arquitectónico

## Estilo Arquitectónico

El sistema se desarrollará utilizando una estructura conformada por cinco capas: model, view, data, common y test.

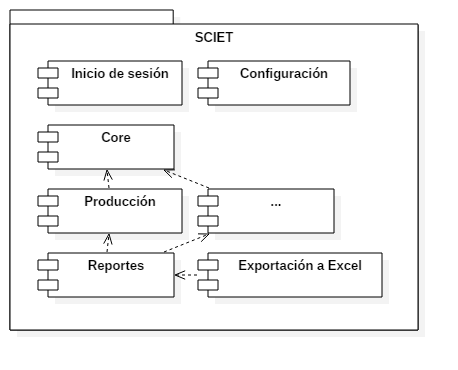
* Model: almacenará todos los POJOS del sistema.
* View: almacenará todo lo relacionado con la interfaz de usuario del sistema.
* Data: presentará una interfaz de comunicación con la base de datos.
* Common: en esta capa se almacenarán archivos de uso común en el sistema.
* Test: contendrá las pruebas unitarias del sistema.

## Objetivos de la Arquitectura

La presente arquitectura pretende cubrir los objetivos de calidad esperados, siendo estos los siguientes:

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo de calidad** | **Rango Requerido** | **Motivo** | **Estrategias para lograr el objetivo** |
| Escalabilidad | N/A | Aunque el cliente únicamente solicitó que el sistema abarcara las áreas de producción y ventas, también mencionó que a largo plazo el sistema deberá incluir módulos para cada área de la empresa. | * Utilizar una arquitectura en capas, aunada a una separación en módulos, que permitan reducir el acoplamiento de los componentes de tal forma que el agregar módulos al sistema se haga de forma transparente. |
| Velocidad de respuesta | Máximo 3 segundos | Aun cuando el cliente no solicitó explícitamente este atributo, el equipo de desarrollo considera que al ser un sistema que será usado frecuentemente, éste debería responder las consultas lo más rápido posible. | * Definir índices en las tablas dentro de la base de datos para asegurar que las consultas se realizarán de forma eficiente. |

# Diagrama de Componentes.



A continuación describiremos cada componente:

| C-01: Inicio de sesión | |
| --- | --- |
| Descripción: | Permite acceder al sistema a los usuarios, validando sus credenciales. |
| Requerimientos: | *RF\_005 – Inicio de sesión* |
| Interfaces Disponibles: | Este componente es independiente |

| C-02: Configuración | |
| --- | --- |
| Descripción: | Permite establecer los datos de conexión para el servidor de base de datos, así como otras configuraciones del sistema. |
| Requerimientos: | *RF\_006 - Configurar parámetros de conexión* |
| Interfaces Disponibles: | Este componente es independiente. |

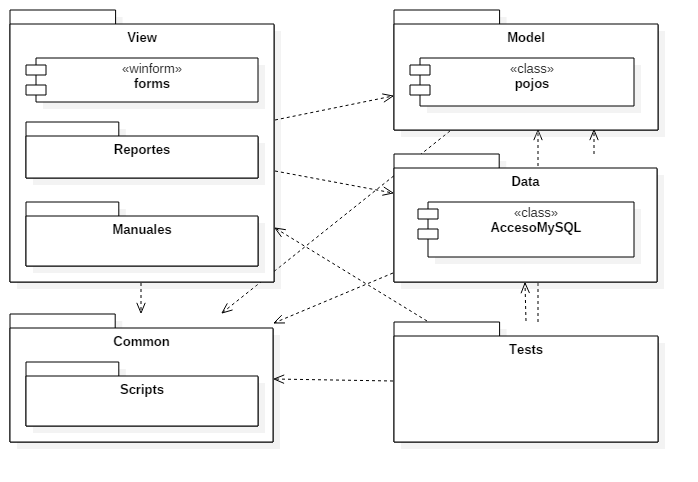
| C-03: Core | |
| --- | --- |
| Descripción: | Es el núcleo del sistema. Permite, en caso de requerirlo, intercomunicar los módulos de cada área |
| Requerimientos: | *No aplica* |
| Interfaces Disponibles: | Este componente deberá poder trabajar con todos los componentes de cada área. |

| C-04: Producción | |
| --- | --- |
| Descripción: | Permite gestionar el área de producción de la empresa. |
| Requerimientos: | *RF\_002 – Catálogo de productos* |
| Interfaces Disponibles: | Este componente trabajará de la mano del componente C-03, así como con el componente de reportes |

| C-05: Reportes | |
| --- | --- |
| Descripción: | Permite generar reportes de todas las áreas. |
| Requerimientos: | *RF\_003 – Reportes* |
| Interfaces Disponibles: | Trabaja con datos de los componentes de las áreas de producción |

| C-06: Exportación de reportes | |
| --- | --- |
| Descripción: | Permite exportar a Excel reportes generados por el componente C-05. |
| Requerimientos: | *RF\_004 – Exportación de reportes a Excel* |
| Interfaces Disponibles: | Trabaja de la mano del componente C-05. Además depende de configuraciones del componente C-02 |

Los componentes se organizarán en una estructura de capas:



A continuación se describen cada una de ellas:

## View

| C-10: forms | |
| --- | --- |
| Descripción: | Se refiere a los componentes de interfaz gráfica de usuario. Éste componente contendrá:   * Todos los formularios de la interfaz de usuario. * Manejo de eventos para las interfaces de usuario y canalización. * Carga de datos desde la capa de datos a la presentación en sus componentes. |
| Requerimientos: | Tecnología: C#.Net Windows Forms, Framework 4.6  Plataforma: Windows 7, 8 y 10. |
| Interfaces Disponibles: | Se comunican con data, model y con common. |

| C-11: Reportes | |
| --- | --- |
| Descripción: | Se refiere a los componentes de interfaz gráfica de usuario que involucran reportes. |
| Requerimientos: | Tecnología: C#.Net Windows Forms, Framework 4.6  Plataforma: Windows 7, 8 y 10. |
| Interfaces Disponibles: | Se comunican con data, model y con common. |

| C-11: Reportes | |
| --- | --- |
| Descripción: | Se refiere a los manuales del sistema. |
| Requerimientos: | Tecnología: C#.Net Windows Forms, Framework 4.6  Plataforma: Windows 7, 8 y 10. |
| Interfaces Disponibles: | Este componente es independiente |

## Model

| C-20: Pojos | |
| --- | --- |
| Descripción: | Almacena todas las clases tipo pojo que representan los objetos del sistema. |
| Requerimientos: | Tecnología: C#.Net Windows Forms, Framework 4.6  Plataforma: Windows 7, 8 y 10. |
| Interfaces Disponibles: | Este componente es independiente |

## Data

| C-30: Reportes | |
| --- | --- |
| Descripción: | Incluye la clase conexión a la base de datos. Además, incluye todos los DAOs del sistema. |
| Requerimientos: | Tecnología: C#.Net Windows Forms, Framework 4.6  Plataforma: Windows 7, 8 y 10. |
| Interfaces Disponibles: | Se comunican con model. |

## Common

| C-40: Scripts | |
| --- | --- |
| Descripción: | Almacena scripts de la base de datos con respaldos y datos de ejemplo. |
| Requerimientos: | Tecnología: MySQL 8 Community  Plataforma: Windows 7, 8 y 10. |
| Interfaces Disponibles: | Este componente es independiente |

## Tests

| C-50: Tests | |
| --- | --- |
| Descripción: | Incluye las pruebas unitarias de los componentes del sistema para asegurar la calidad. |
| Requerimientos: | Tecnología: C#.Net Windows Forms, Framework 4.6  Plataforma: Windows 7, 8 y 10. |
| Interfaces Disponibles: | Se comunican con todas las capas. |

En resumen, el presente proyecto se desarrollará bajo las siguientes tecnologías:

* **C# .net**. Para las capas de presentación y servicios, ya que se cuenta con amplia experiencia en este lenguaje.
* **MySQL 8 Community**. Para parte de la persistencia se utilizara dicho gestor de base de datos, ya que no tiene costo de licencia por el uso comercial del mismo.

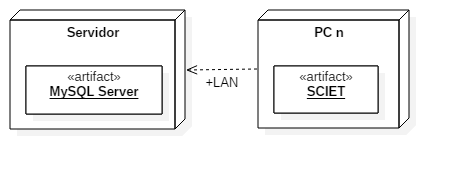
# Vista de Casos de Uso

*Para ver los casos de uso del sistema revisar el documento COLOCA AQUÍ EL NOMBRE DEL DOCUMENTO DONDE TENDRAN ALMACENADOS LOS CASOS DE USO.*

# Diagrama de despliegue.

## Visión General

A continuación se mostrará el diagrama de despliegue de cómo se implantará el sistema:



En el diagrama se muestra la distribución física requerida para implementar la aplicación y se muestran los componentes requeridos en el lado del cliente y en el lado del servidor.

El sistema consistirá en un servidor de base de datos que almacene la información del sistema; y en una computadora cliente que tendrá instalada la aplicación. En caso de requerirse, una única computadora podrá fungir como servidor y como cliente.

# Diagrama de Base de Datos

A continuación se lista el modelo de datos del sistema de manera parcial, con el motivo de que éste será refinado mediante el desarrollo iterativo del mismo.

**Nombre de la base de datos**: SCIETBD

**SGBD:** MySQL 8 Community

**Diagrama Relacional**

|  |  |
| --- | --- |
| **Rastreabilidad con los Requerimientos** | |
| **Tabla** | **Requerimiento(s) relacionados.** |
|  | |
| **NOMBRE DE LA BD** | |
| **COLOCA AQUÍ EL NOMBRE DE CADA UNA DE LAS TABLAS** | **COLOCA AQUÍ LOS REQUERIMIENTOSQUE SE RELACIONAN CON CADA TABLA** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |