

# Implementación de la metodología de modelado OOHDM en el desarrollo de un sistema web para la asignación de causas penales y apelaciones en el Juzgado de Control y Enjuiciamiento Penal del Estado de Guerrero en Acapulco

Nava Santana Rigoberto de Jesús, Gutiérrez Mata Francisco Javier,  
Gazga Portillo José Francisco, Carranza Gómez Jorge

Tecnológico Nacional de México campus Acapulco, División de Estudios de Postgrado e Investigación, Av. Institituto Tecnológico s/n Crucero del Cayaco C.P. 39905, Guerrero, México.

## Resumen

En el presente documento se exponen los resultados obtenidos en la etapa de análisis y diseño de un sistema web, que tiene como finalidad sistematizar los procesos administrativos correspondientes a la asignación de causas penales a jueces y apelaciones a magistrados en el Juzgado de Control y Enjuiciamiento Penal del estado de Guerrero con sede en Acapulco de Juárez, ya que dicha dependencia gubernamental carece de un sistema que auxilie al personal a realizar estos procesos complejos. Es por ello que, para comprender de mejor manera los requerimientos que se solicitan para la construcción de este sistema, se ha implementado la metodología de modelado OOHDM (*Object Oriented Hypermedia Design Method*) ya que proporciona una mejor perspectiva para el modelaje y diseño de aplicaciones y sistemas web a partir de las cuatro etapas que componen a esta metodología. A lo largo de este documento se desglosarán los aspectos teóricos de esta metodología de modelado, para posteriormente dar paso a la documentación de su implementación, las conclusiones se centrarán en los resultados de la implementación, haciendo énfasis en los beneficios que trae la metodología en el desarrollo de aplicaciones y sistemas web.

## Abstract

This document presents the results obtained in the analysis and design stage of a web system, which aims to systematize the administrative processes corresponding to the assignment of criminal cases to judges and appeals to magistrates in the Court of Control and Criminal Prosecution of the state of Guerrero based in Acapulco de Juarez, since this government agency lacks a system that helps staff to perform these complex processes. That is why, in order to better understand the requirements requested for the construction of this system, the OOHDM (*Object Oriented Hypermedia Design Method*) modeling methodology has been implemented, since it provides a better perspective for the modeling and design of web applications and systems from the four stages that make up this methodology.

Throughout this document the theoretical aspects of this modeling methodology will be broken down, to later give way to the documentation of its implementation, the conclusions will focus on the results of the implementation, emphasizing the benefits that the methodology brings in the development of web applications and systems.

**Palabras clave:** Sistematización, metodología, implementación.

**Keywords:** Systematization, Methodology, Implementation.

## 1. INTRODUCCIÓN.

En la actualidad, la mayoría de las dependencias gubernamentales cuentan con algún sistema o aplicación web para la sistematización de procesos complejos, dando como resultado la agilización de ciertos pasos que antes causaban retraso, así como facilitar la organización y búsqueda de información.

El Juzgado de Control y Enjuiciamiento Penal del Estado de Guerrero con sede en Acapulco de Juárez es una dependencia gubernamental que cumple con la alta función de impartir justicia, en el fuero común, a los gobernados, para dar vigencia a la garantía de acceso a la justicia [1], por lo cual, una parte de este proceso de impartir justicia es el de recibir solicitudes de causas penales por parte de los ministerios públicos o de la población que se presenta a declarar una denuncia, por lo que el Juzgado de Control realiza el proceso de asignación de causas penales a jueces, donde el personal, a través de una serie de criterios, selecciona a un juez entre todos los que conforman el Juzgado de Control para dar seguimiento a dicha solicitud y finalizar con su correspondiente veredicto.

Por otro lado, cuando se ha finalizado una solicitud de causa penal con su respectivo veredicto y una de las partes procesales desea impugnar tal decisión, el Juzgado de Control se ve en la necesidad de redactar una solicitud de apelación, por lo que se realiza el proceso de asignación de apelaciones a magistrados, similar al proceso de asignación de causas penales, a través de una serie de criterios se selecciona a un magistrado entre todos los que conforman el Juzgado de Control.

Sin embargo, el personal del Juzgado de Control realiza ambos procesos de manera convencional, es decir, buscando los datos necesarios que se requieren para calcular cada criterio en archiveros llenos de documentos, además de realizar los cálculos necesarios en el programa Excel. Si bien el programa Excel ha sido usado por un largo periodo de tiempo, no satisface por completo las necesidades del personal, debido a que su uso provoca retrasos en la realización de los procesos. Es por esta razón que se propone el desarrollado un sistema web como una solución a la ausencia de una herramienta computacional a la medida, que auxilie al Juzgado de Control.

Por lo cual, para tener un mejor entendimiento de los requerimientos que se solicita sean cubiertos, se ha implementado la metodología de modelado OOHD (Object Oriented Hypermedia Design Method).

## 2. METODOLOGÍA.

Si bien las metodologías de ingeniería de software funcionan como un conjunto de procedimientos para el desarrollo de sistemas, ninguna de ellas brinda el soporte adecuado o necesario para el desarrollo de sistemas web en específico, es por esta problemática que surgieron nuevas metodologías como una solución ante la falta de soporte para el desarrollo de sistemas y aplicaciones basados en tecnologías web, algunas de esas metodologías son: UWE, WAE, OOWS y entre ellas OOHD. El objetivo de estas metodologías es orientar a los desarrolladores y diseñadores para la construcción de aplicaciones y sistemas basados en web, ofreciendo herramientas para comprender la arquitectura, navegabilidad y diseño. En este artículo se habla sobre la implementación de la metodología OOHD.

### 2.1. OOHD.

Propuesta por Schwabe y Rossi, esta metodología, surge a partir de la necesidad de modelar aplicaciones de hipermedia dinámicas a gran escala que combinan patrones de navegación complejos con un comportamiento computacional sofisticado [2].

El tamaño, la complejidad y el número de aplicaciones crecen en forma acelerada en la actualidad, por lo cual una metodología de diseño sistemática es necesaria para disminuir la complejidad y admitir evolución y reusabilidad [3].

Esta metodología hace énfasis en separar el desarrollo de una aplicación o sistema web en cuatro etapas, las cuales son:

1. Diseño conceptual.
2. Diseño navegacional.
3. Diseño de interfaz abstracta.
4. Implementación.

Sin embargo, los autores Molina, Zea, Contento y García [4] consideran la obtención de requerimientos como el primer paso de la metodología OOHDM, dando como resultado cinco etapas.

Se menciona que esta metodología permite desarrollar aplicaciones web a partir de la utilización de modelos especializados como: conceptual, navegación e interfaz de usuario, teniendo como objetivo simplificar y hacer más eficaz el diseño de aplicaciones [4].

#### **2.1.1. DISEÑO CONCEPTUAL.**

Primera etapa de la metodología OOHDM, se construye un esquema conceptual representado por los objetos de dominio, las relaciones y colaboraciones existentes establecidas entre ellos [3].

Se construye un modelo del dominio de la aplicación, a través de las técnicas del modelado orientado a objetos, se puede partir de un modelo E/R. Se identifican las clases y sus relaciones, que pueden ser de asociación, agregación, composición, y generalización-especialización. El resultado de esta etapa es un modelo estructural compuesto por clases, asociaciones y atributos, y es similar al diagrama de clases del UML [5].

#### **2.1.2. DISEÑO NAVEGACIONAL.**

Un modelo navegacional es construido como una vista sobre un diseño conceptual, admitiendo la construcción de modelos diferentes de acuerdo con los diferentes perfiles de usuarios [3].

Representa los diferentes caminos que puede ejecutar la aplicación dependiendo del tipo de usuario. Es decir, brinda un contexto navegacional capaz de realizar acciones a través de enlaces, vínculos o índices que están relacionados dentro de la aplicación web dependiendo del perfil de usuario para mostrar sus vistas correspondientes [4].

#### **2.1.3. DISEÑO DE INTERFAZ ABSTRACTA.**

En OOHDM utilizamos el enfoque de diseño ADV (*Abstrac Data View*) para describir la interfaz de usuario de una aplicación hipermedia. Los ADV son abstractos en el sentido de que solo representan la interfaz y el estado y no la implementación [2].

En esta etapa de la metodología se especifican las interfaces de usuario que se visualizaran en la aplicación web [4].

#### 2.1.4. IMPLEMENTACIÓN.

Al llegar a esta etapa, el primer paso que debe realizar el diseñador es definir los ítems de información que son parte del dominio del problema. Debe de identificar también, como son organizados los ítems de acuerdo con el perfil del usuario y sus tareas; definir que interfaz debería ver y como debería comportarse [3].

### 3. RESULTADOS.

Los resultados obtenidos contemplan las cuatro etapas originales de la metodología más la obtención de requerimientos propuesta por Molina, Zea, Contento y García, a continuación, se exponen los diferentes esquemas que propone la metodología OOHDM.

#### 3.1. REQUERIMIENTOS.

Como ya se mencionó anteriormente, esta etapa no forma parte de la propuesta original de Schwabe y Rossi, sin embargo, se implementó al igual que las otras cuatro etapas al ser contemplada por otros autores.

Esta etapa consiste en plasmar los requerimientos del sistema en diagramas de casos de uso, para la identificación de roles y sus respectivos privilegios, lo cual dio como resultado la identificación de cinco roles para el manejo del sistema, los cuales son:

1. Recepcionista.
2. Jefe de recepcionistas.
3. Administrador.
4. Juez.
5. Magistrado.

En la figura 1 se puede observar el diagrama correspondiente a los roles recepcionista y jefe de recepcionistas, donde se describen las principales tareas de estos roles a realizar dentro del sistema, destacando el registro de solicitudes para ambos roles y la validación de solicitudes para el jefe de recepcionistas.

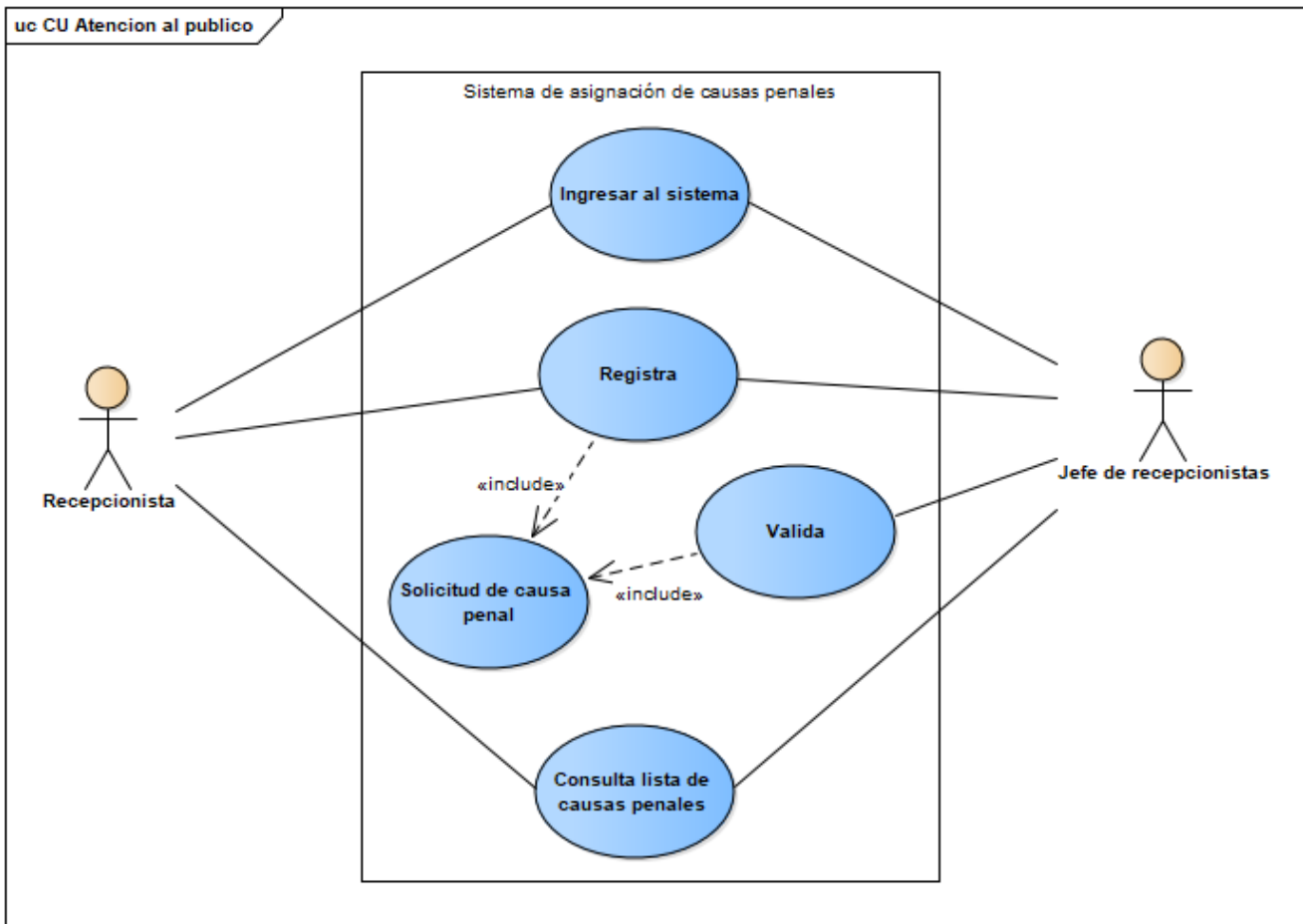


Figura 1. Diagrama de caso de uso de los roles recepcionista y jefe de recepcionistas (elaboración propia).

En la figura 2 se puede observar el diagrama correspondiente al rol administrador, el cual fue identificado como el rol con los privilegios más altos dentro del sistema, pues es el responsable del departamento donde se realizan los procesos a sistematizar, para este rol destacan las actividades de asignación de causas penales a jueces, asignación de apelaciones a magistrados y registro de solicitud de apelación, siendo esta última tarea única para el rol administrador.

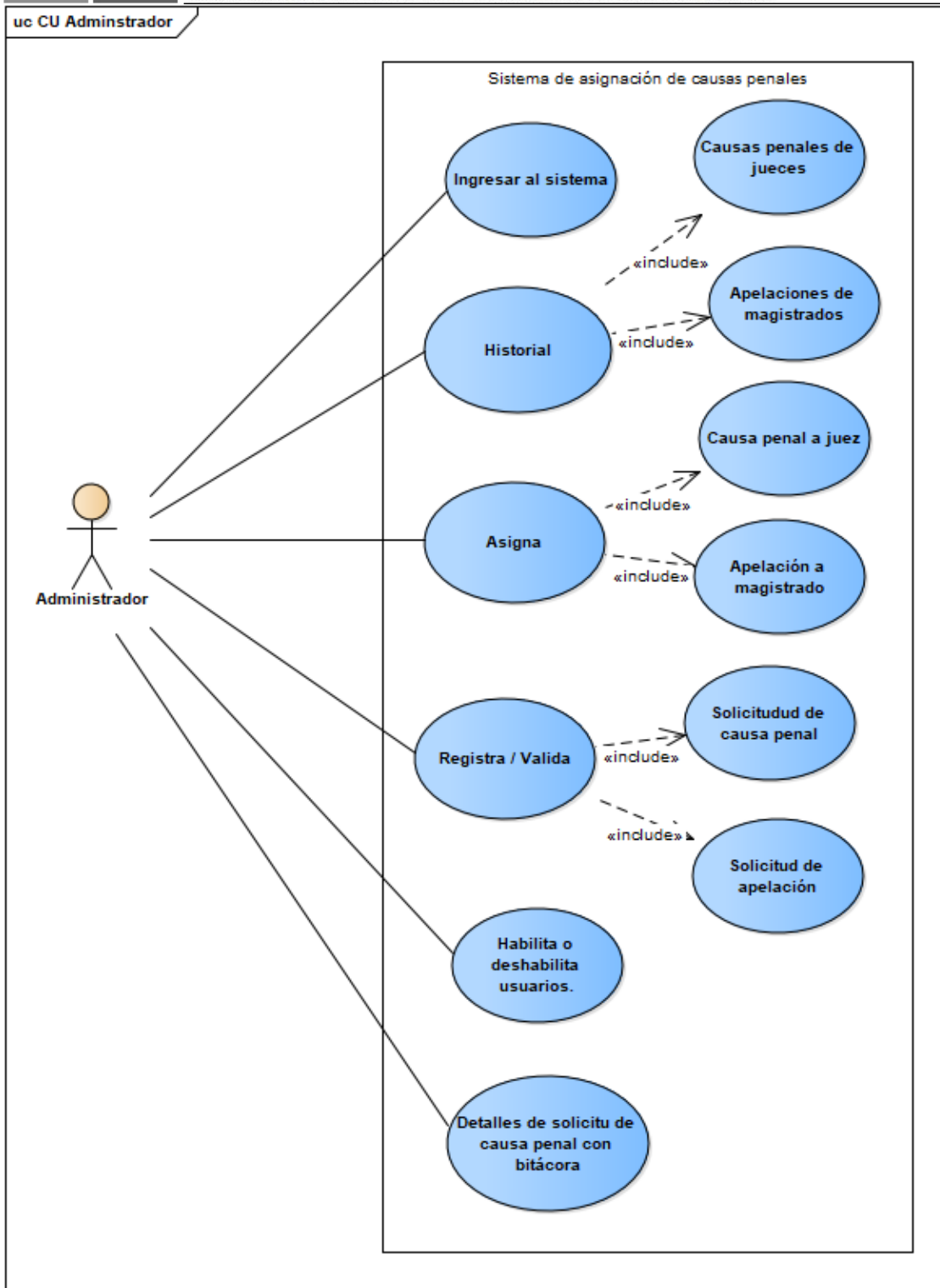


Figura 2. Diagrama de caso de uso del rol Administrador (elaboración propia).

En la figura 3 se puede observar el diagrama correspondiente al rol juez y la figura 4 corresponde al rol magistrado, ambos son los roles con los privilegios más bajos del sistema ya que sus únicas acciones son las de ingresar al sistema, consultar información y registrar indisponibilidades.

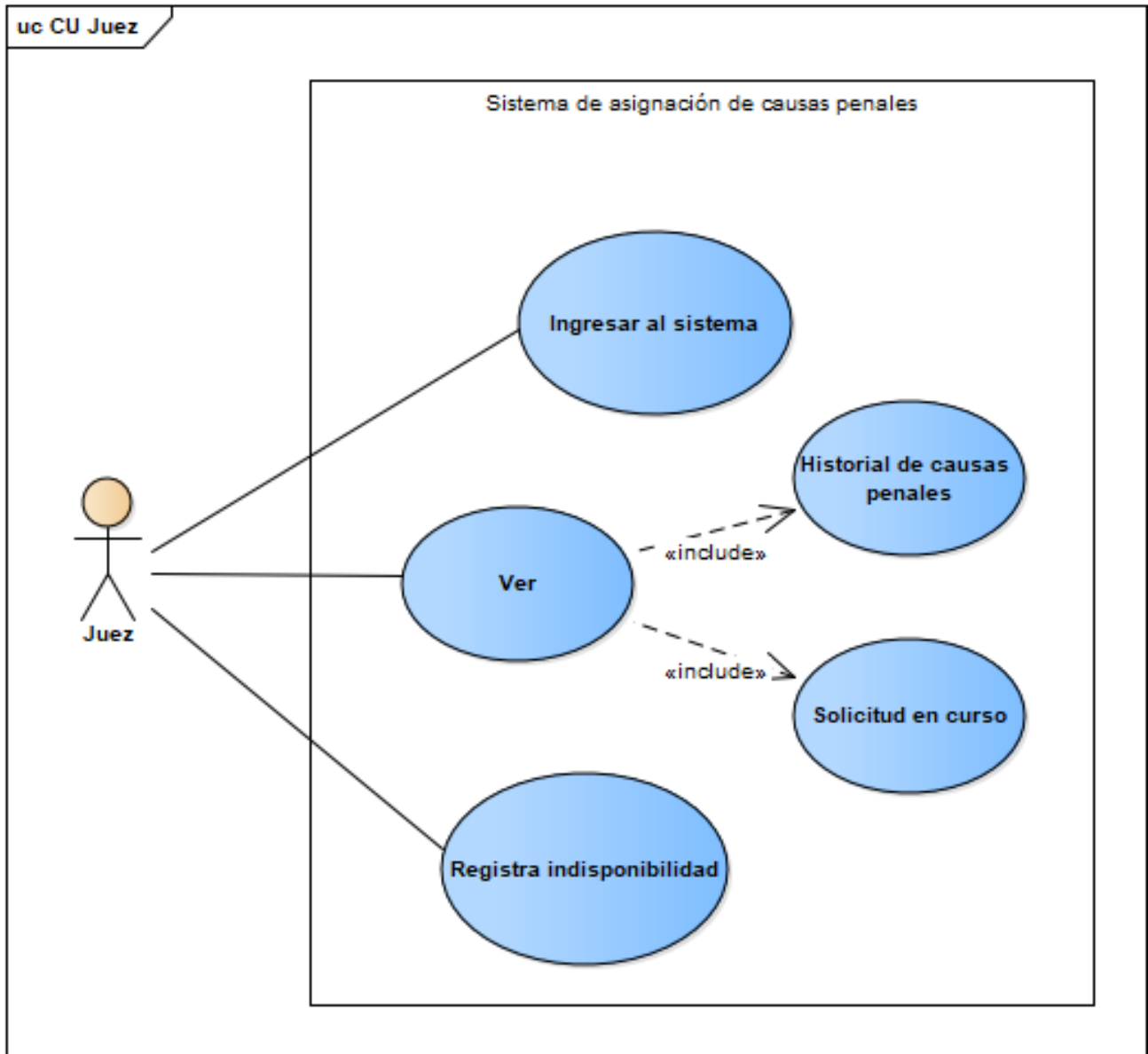


Figura 3. Diagrama de caso de uso del rol juez (elaboración propia).

uc CU Magistrado

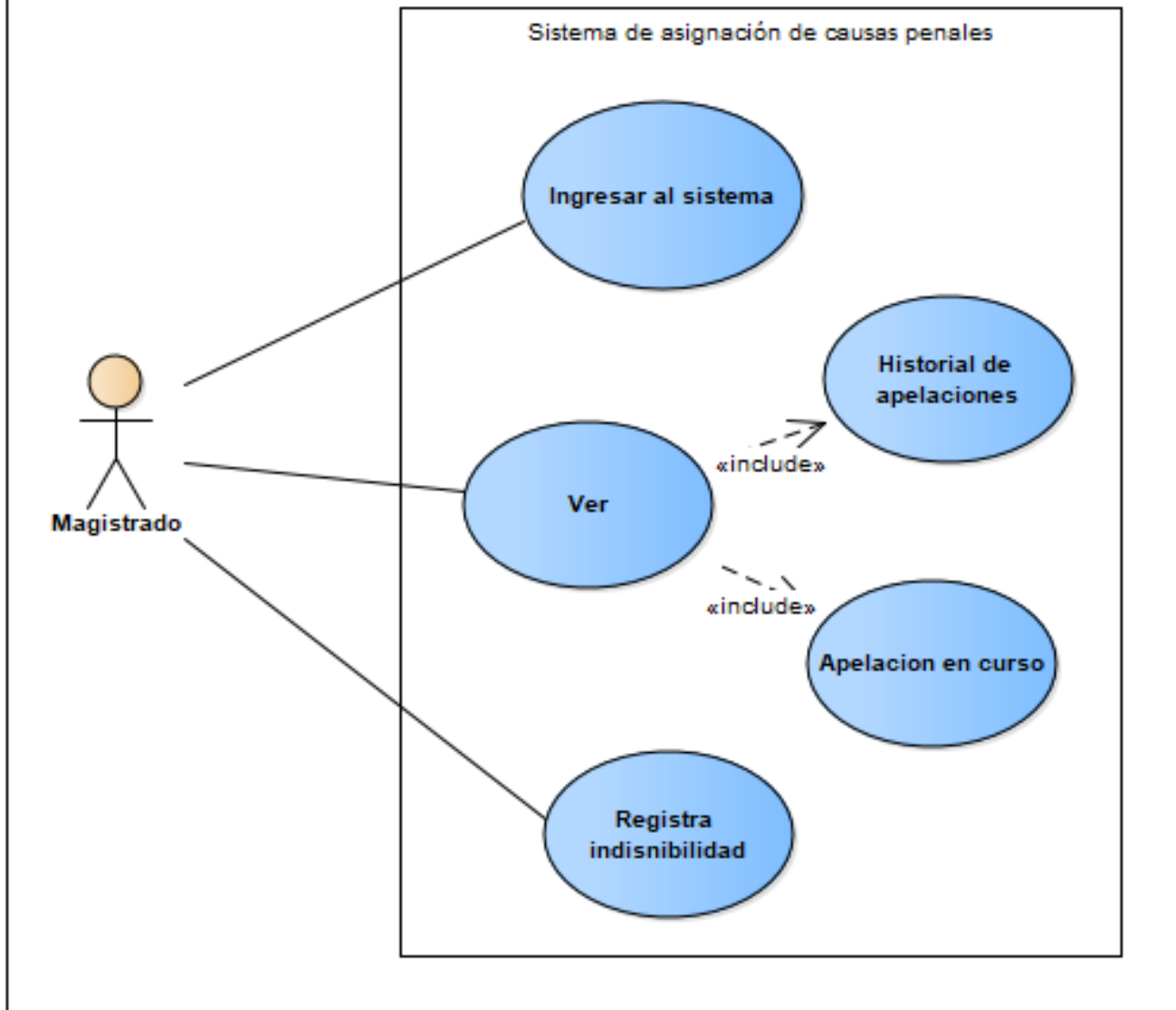


Figura 4. Diagrama de caso de uso del rol magistrado (elaboración propia).

Tras la obtención de requerimientos e identificación de roles y su posterior modelado, se logra comprender con mayor claridad que es lo que el usuario final desea en cuanto a funcionalidad se refiere, pues los modelos de caso de uso facilitan la identificación de privilegios para cada rol de usuario.

### 3.2. ESQUEMA CONCEPTUAL.

Utilizando una notación similar a UML la figura 5 muestra el esquema que representa el modelo conceptual, donde se identificaron ocho clases, las cuales son:

1. Solicitud.
2. Recepcionista.



3. Jefe de recepcionistas.
4. Administrador.
5. Juez.
6. Magistrado.
7. Imputado.
8. Víctima.

En este esquema se destaca la relación de todas las clases con la clase solicitud, pues el sistema está pensado para dar solución a la asignación de solicitudes tanto de causas penales como de apelaciones, además de obtener datos de relevancia de las demás clases.

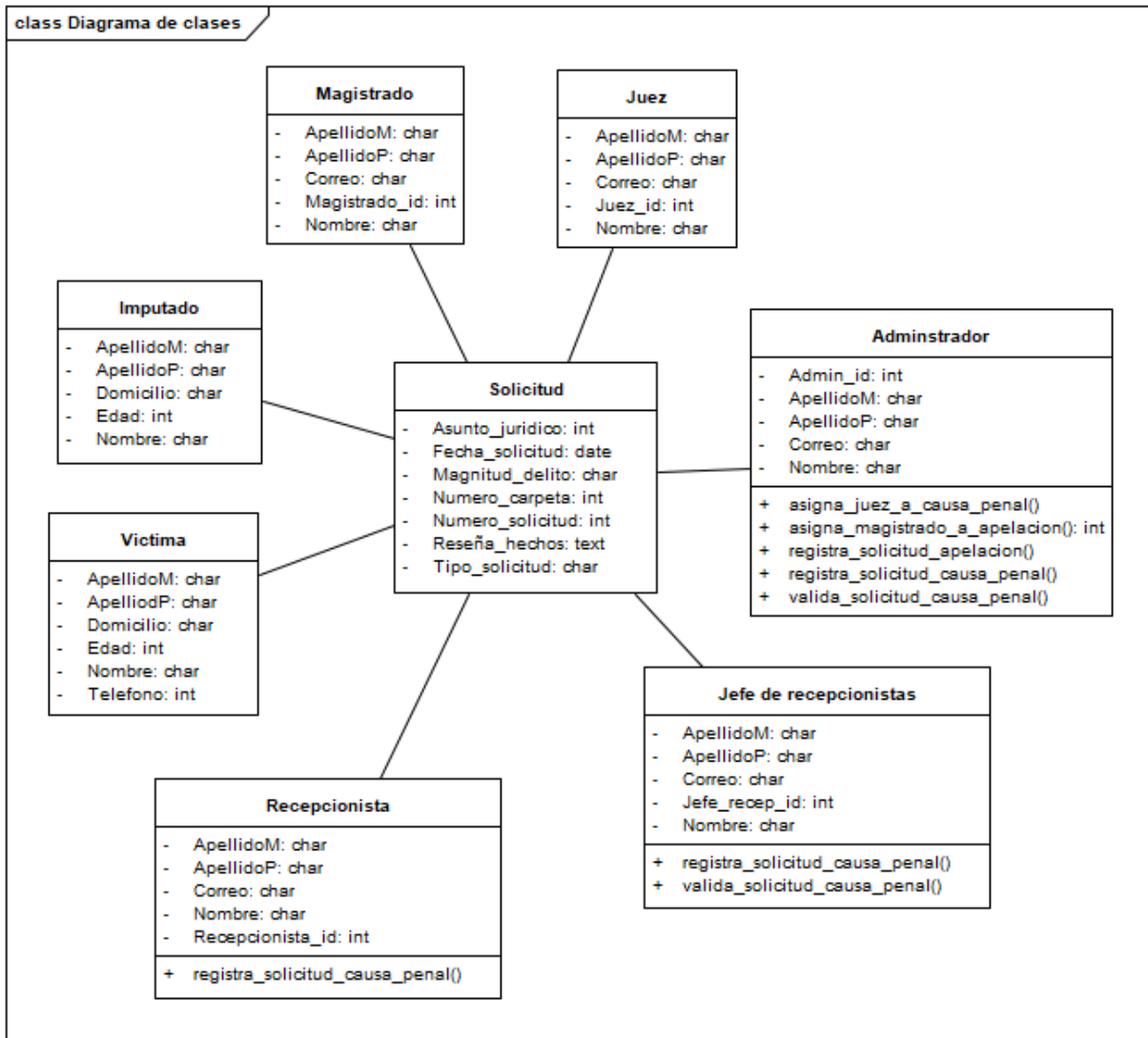


Figura 5. Esquema de modelo conceptual (elaboración propia).

### 3.3. ESQUEMA NAVEGACIONAL.

Gracias a la fase de obtención de requerimientos y su posterior modelado a casos de uso, se logró identificar las actividades a realizar por rol, lo que facilitó la construcción del esquema navegacional.

Para la construcción de este esquema se utilizó una notación similar a la empleada por Schwabe y Rossi. La figura 6 se puede observar el resultado del análisis navegacional, obtenido en buena parte de los diagramas de caso de uso de la fase de requerimientos, el esquema busca representar la navegabilidad del sistema a construir, comenzando con un *login* de acceso mediante nombre de usuario y contraseña, para posteriormente redirigir a un menú principal para cada rol de usuario, en donde se especifican las acciones a realizar.

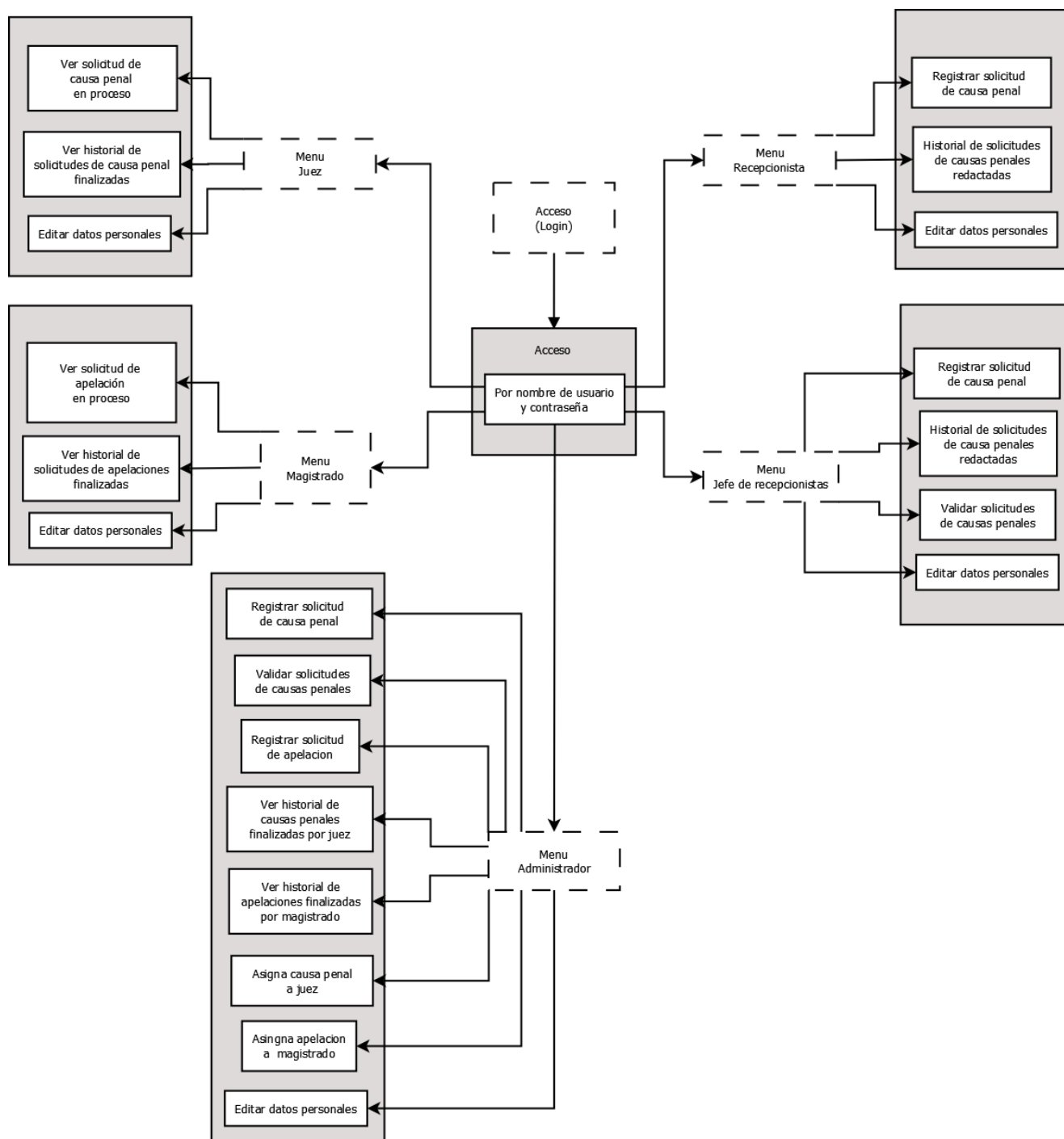


Figura 6. Esquema navegacional (elaboración propia).

### 3.4. INTERFACES ABSTRACTAS.

Para la realización de las interfaces abstractas se tomó como ejemplo lo mostrado en por Schwabe y Rossi, donde se hace uso de ADV (*Abstract Data View*) para la representación de estas. La figura 9 se puede observar el ADV que representa el diseño del acceso al sistema (*login*) mediante correo electrónico y contraseña.

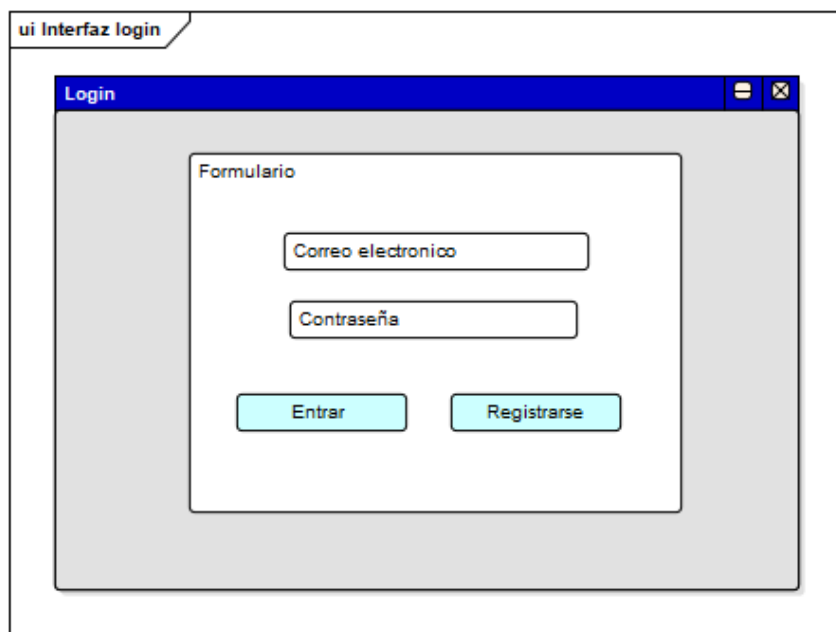


Figura 7. ADV correspondiente al acceso al sistema (elaboración propia).

La figura 10 muestra el ADV correspondiente al diseño del menú principal del sistema, compuesto por una barra de navegación, pestañas para cada sección y un espacio para el logotipo del Juzgado de Control.

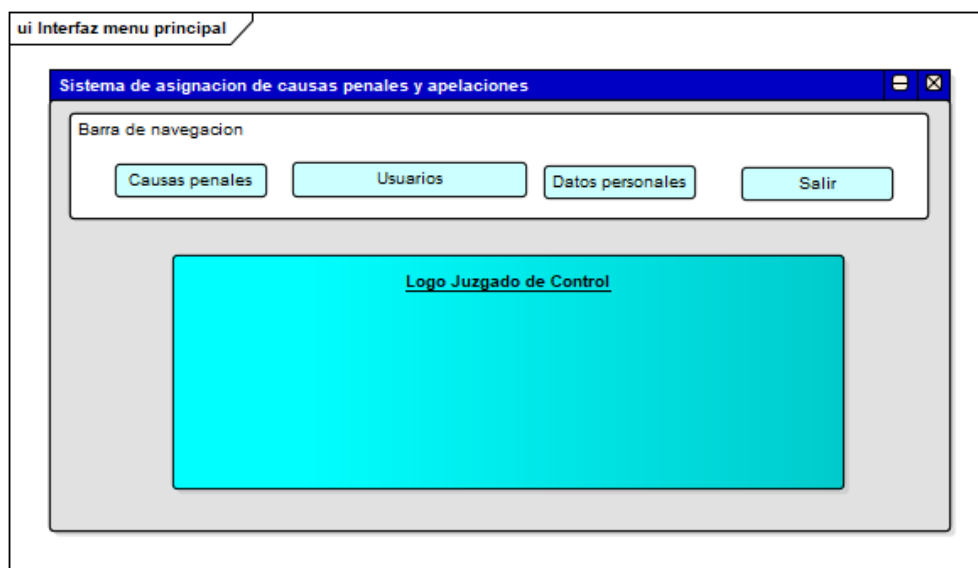


Figura 8. ADV correspondiente al menú principal (elaboración propia).

Para el registro de solicitudes, el Juzgado de Control solicito poder registrar más de una víctima, más de un imputado y un solo denunciante para cada solicitud, por lo que se optó hacer el registro de una solicitud en dos partes, la primera en donde se capturen los datos básicos y la segunda donde se registren las víctimas, imputados y denunciantes, la figura 9 muestra el diseño del formulario para la captura de datos básicos de una solicitud.

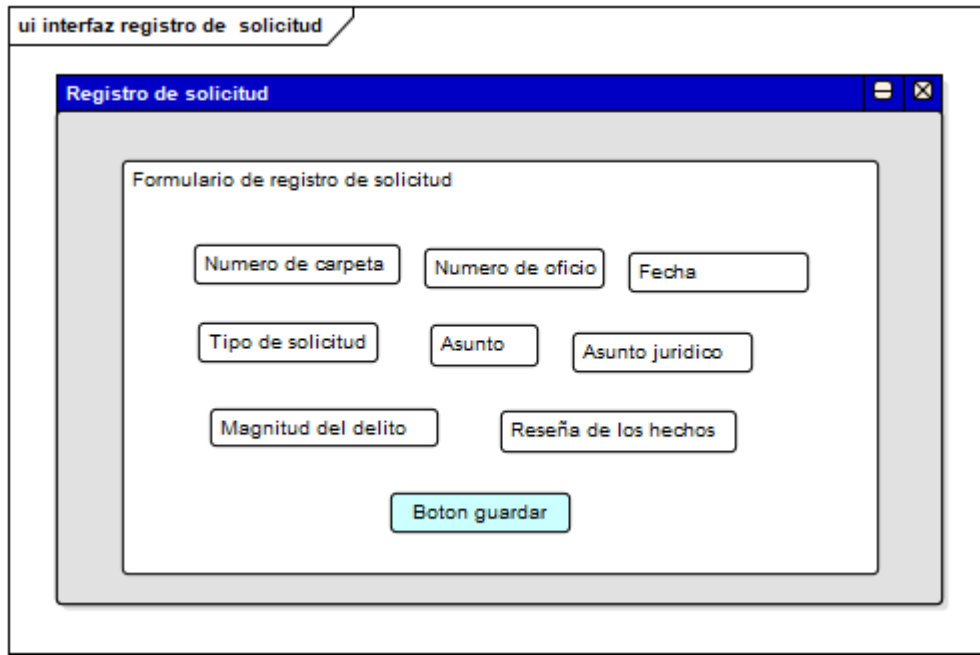


Figura 9. ADV Correspondiente al registro de solicitud (elaboración propia).

La figura 10 muestra el diseño para la captura de víctimas, imputados, denunciante y termino de registro.

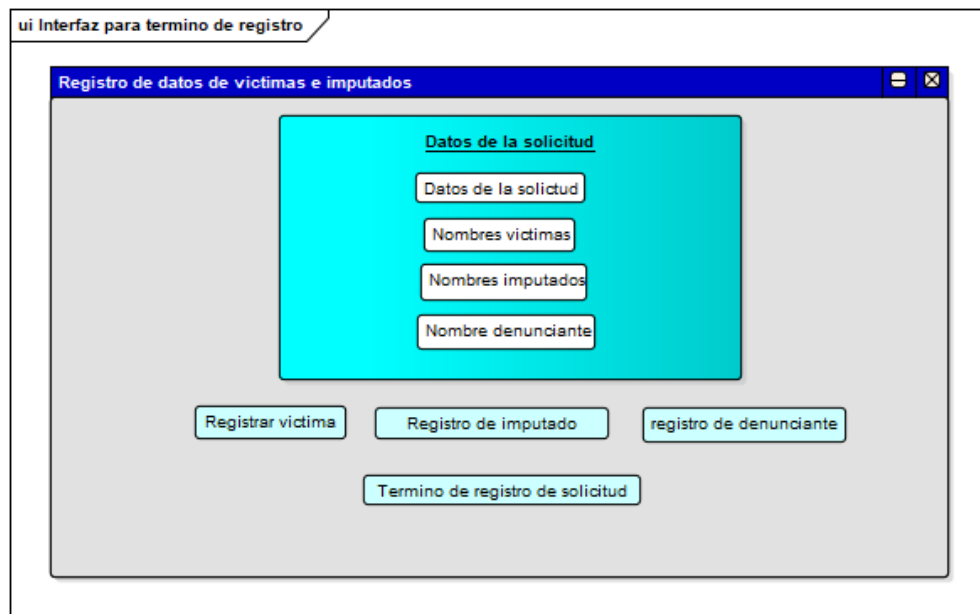


Figura 10. ADV correspondiente al termino de registro (elaboración propia).

Para representar la lista de solicitudes registradas, el Juzgado de Control opto por un diseño similar a tarjetas de presentación con información relevante para cada solicitud, en lugar de una lista convencional de filas y columnas, la figura 11 muestra el diseño para la lista de solicitudes.

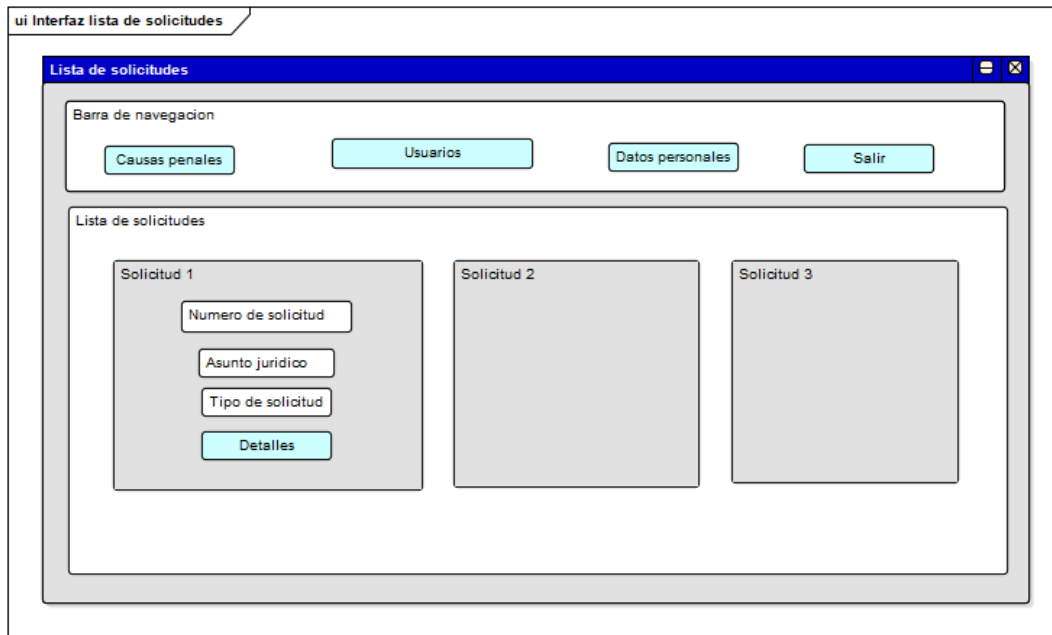


Figura 11. ADV correspondiente a la lista de solicitudes (elaboración propia).

### 3.5. IMPLEMENTACIÓN.

Esta última fase consiste en plasmar el análisis logrado mediante la construcción de los esquemas al desarrollo del sistema, comenzado con el esquema conceptual. Dado que se utiliza una arquitectura MVC (Modelo Vista Controlador), así como el *framework* CodeIgniter para la construcción del sistema, la programación de las clases identificadas en el modelo conceptual se realizó respetando esta estructura, por lo tanto, cada clase del esquema conceptual es un controlador, identificado mediante el nombre de la clase como se muestra en la figura 12.

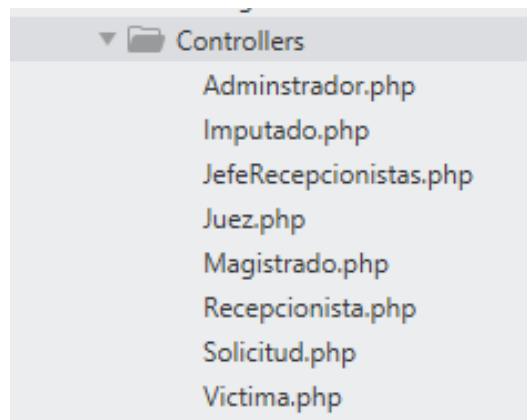


Figura 12. Representación de las clases del esquema conceptual como controladores (elaboración propia).

Una vez realizado el análisis navegacional y la posterior construcción del esquema, se pueden apreciar las rutas de navegación para cada rol. Para ello el *framework CodeIgniter* proporciona una opción de acceso a los controladores por medio de rutas por nombre, la figura 13 muestra la obtención de rutas de navegación para el rol Administrador a través del nombre del controlador que es “Administrador” y el nombre de la función ah acceder.

```
//Rutas Adminstrador
if (session("tipo_usr") == "Adminstrador")
{
    $routes->get("/registrar_solicitud_causa_penal", "Adminstrador::registrar_solicitud_causa_penal");
    $routes->get("/registrar_solicitud_apelacion", "Adminstrador::registrar_solicitud_apelacion");
    $routes->get("/asignar_solicitud_causa_penal", "Adminstrador::registrar_solicitud_causa_penal");
    $routes->get("/asignar_solicitud_apelacion", "Adminstrador::asignar_solicitud_apelacion");
}
```

Figura 13. Acceso a ruta mediante nombre de controlador y nombre de función (elaboración propia).

Finalmente, para el diseño de la interfaz de usuario, se utilizaron los ADV construidos, la figura 14 muestra el acceso (login) del sistema.

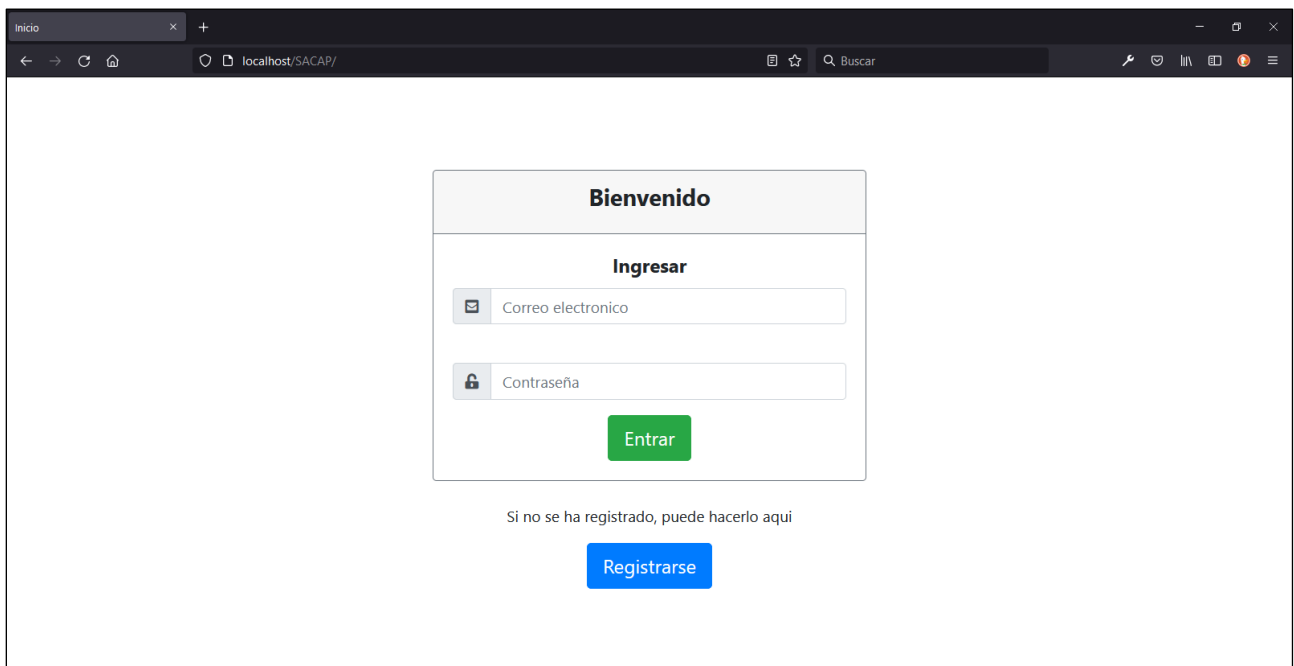


Figura 14. Acceso al sistema.

La figura 15 muestra la vista del menú principal en este caso la del rol administrador.

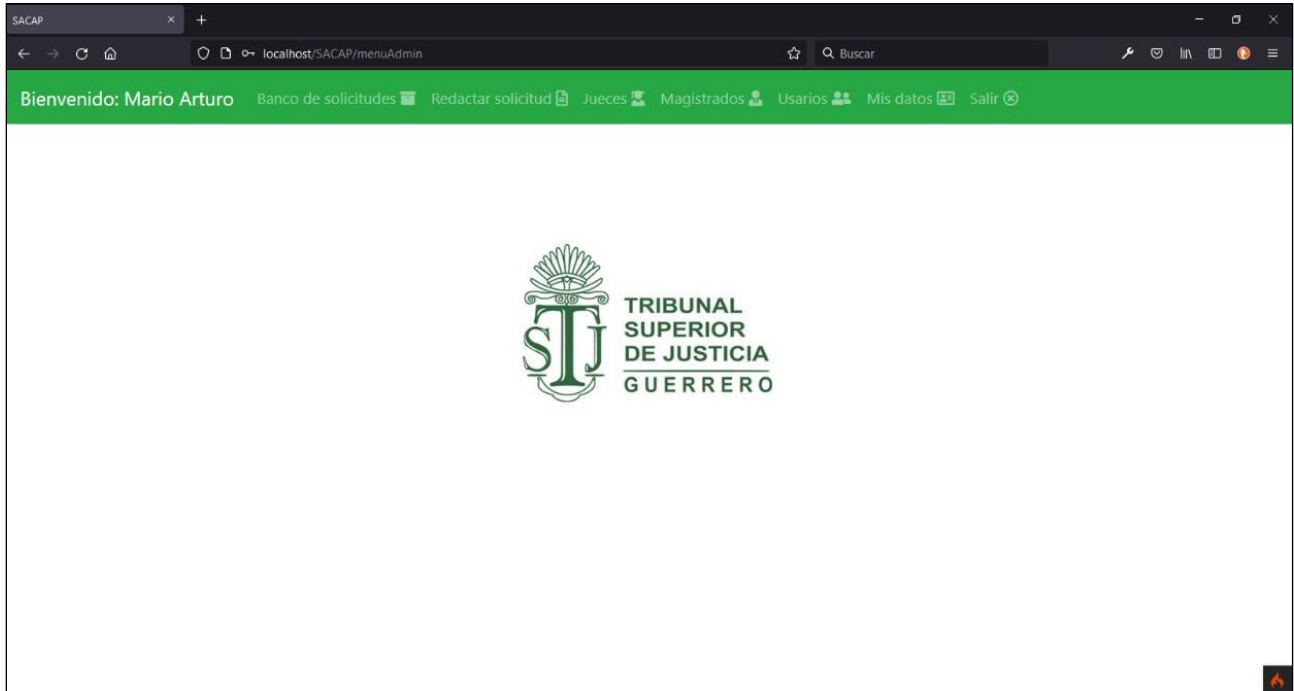


Figura 15. Vista del menú.

Como se mencionó antes, el Juzgado de Control solicito poder hacer el registro de solicitudes de tal manera que se logre registrar más de una víctima, más de un imputado y un solo denunciante por cada solicitud, la figura 16 muestra la vista del registro de datos básicos de una solicitud.

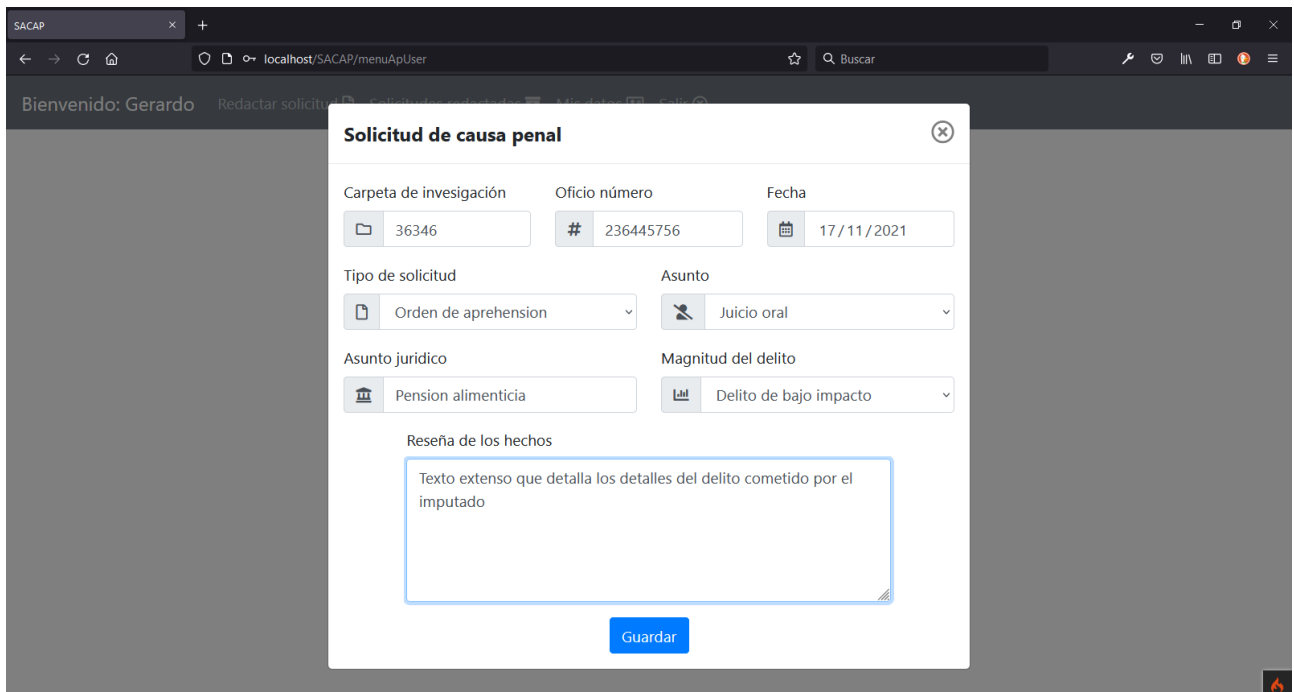


Figura 16. Vista del registro de solicitud.

La figura 17 muestra la vista general de una solicitud, donde se pueden registrar víctimas, imputados, denunciante y finalizar el registro.

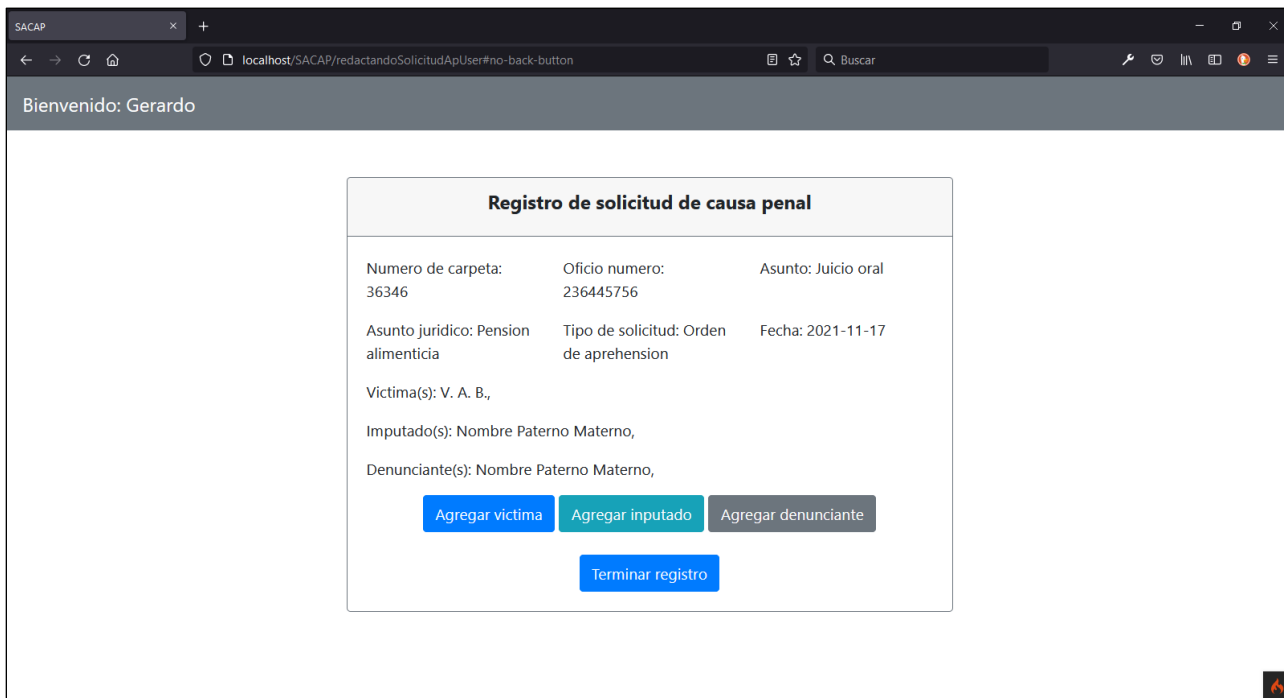


Figura 17. Vista general del registro de solicitud.

La figura 18 muestra la lista de solicitudes en forma de tarjetas de presentación.

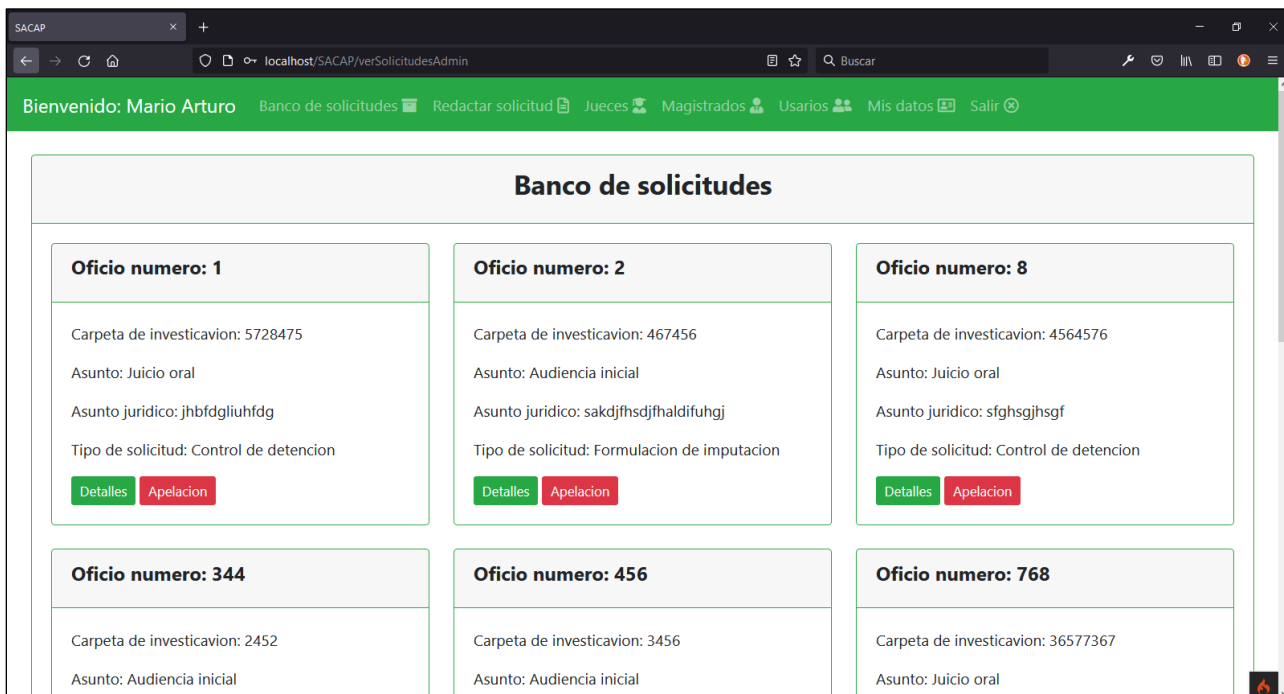


Figura 18. Lista de solicitudes.



#### 4. CONCLUSIONES.

Una vez terminado todo el análisis que se propone en la metodología OOHDM, se puede decir que cumple su objetivo de ser una guía eficiente para el modelado, pues logra guiar a los desarrolladores y diseñadores a entender los aspectos de estructura navegabilidad y diseño, si bien las metodologías de ingeniería de software buscan el desarrollo de sistemas de forma ordenada orientando a los equipos de desarrollo, no brindan el soporte necesario para la construcción de aplicaciones y sistemas web, sin embargo, no se pretende que OOHDM tenga que sustituir a una metodología de ingeniería de software.

Por otro lado, las etapas que componen a esta metodología pueden ser divididas en dos bloques que son: análisis *back-end* y diseño *front-end*, abarcando el diseño conceptual y diseño navegacional para el *back-end* y diseño de interfaz abstracta para el *front-end*, dejando a la fase de implementación como independiente, ya que en esta última se desarrollan los esquemas y se traducen a programación y diseño.

Finalmente, la metodología OOHDM demuestra ser un buen complemento a una metodología de ingeniería de software, logrando implementarse durante la fase de análisis y diseño para una comprensión de requerimientos más eficiente, al otorgar a los desarrolladores herramientas para el análisis.

#### REFERENCIAS.

- [1] Poder Judicial del Estado de Guerrero, «¿Quiénes somos?», 2020. [En línea]. Available: <http://tsj-guerrero.gob.mx/2020/?p=139>.
- [2] D. Schwabe y G. Rossi, «An Object Oriented Approach to Web-Based Application Design,» Enero 1998. [En línea]. Available: [https://www.researchgate.net/publication/220346789\\_An\\_Object\\_Oriented\\_Approach\\_to\\_Web-Based\\_Applications\\_Design](https://www.researchgate.net/publication/220346789_An_Object_Oriented_Approach_to_Web-Based_Applications_Design).
- [3] D. Silva y B. Mercerat, «Construyendo aplicaciones web con una metodología de diseño orientada a objetos,» *Revista Colombiana de Computación*, vol. 2, n° 2, 2001.
- [4] J. R. Molina Ríos, M. P. Zea Ordóñez, M. J. Contento Segarra y F. G. García Zerda, «Comparación de metodologías en aplicaciones web,» 3c *Tecnología: glosas de innovación aplicadas a la pyme*, vol. 7, n° 1, pp. 1-19, 2018.
- [5] A. N. Del Valle Rodríguez, *Metodologías de diseño usadas en ingeniería web, su vinculación con las NTICS*, La plata, Arg.: Universidad Nacional de La plata, 2009.

Correo de autor: [rigojns19@gmail.com](mailto:rigojns19@gmail.com)