# Historias de Usuario para Triaje

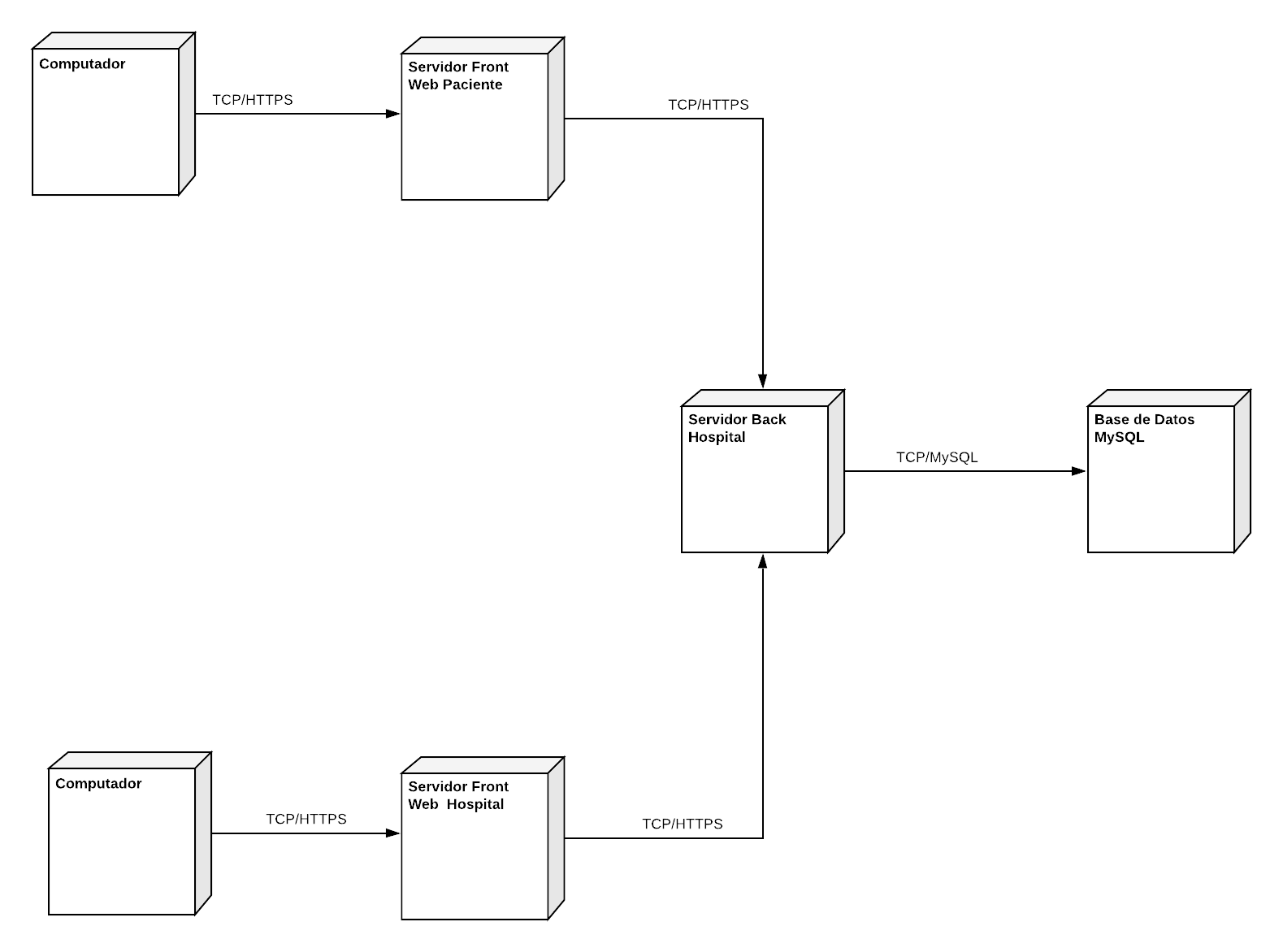
|  |  |  |
| --- | --- | --- |
| Historia de usuario | | |
| Número: 1 | Usuario: Paciente | |
| Nombre historia: Registro de solicitud de atención | | |
| Prioridad: 3 |  | Iteración: 3 |
| Descripción:  Como paciente quiero poder registrar con anterioridad una solicitud de atención para poder ahorrar tiempo de espera. | | |
| Criterios de aceptación:  El sistema debe poder ubicar el establecimiento de salud deseado por paciente.  El sistema debe permitir registrar los siguientes datos:   * Nombres * Apellido Paterno * Apellido Materno * Edad * Género * DNI * Síntomas * Teléfono celular o correo electrónico   El sistema debe a ayudar a identificar el nombre del paciente haciendo uso solo del DNI.  El sistema debe registrar los síntomas del paciente bajo declaración jurada.  El sistema debe poder detectar automáticamente si el paciente tiene seguro SIS o no. | | |

|  |  |  |
| --- | --- | --- |
| Historia de usuario | | |
| Número: 2 | Usuario: Licenciado de enfermería | |
| Nombre historia: Gestionar solicitudes de atención | | |
| Prioridad: 3 |  | Iteración: 3 |
| Descripción:  Como licenciado de enfermería quiero visualizar las solicitudes de atención registradas hasta el momento para poder administrarlas. | | |
| Criterios de aceptación:  El sistema debe poder gestionar los siguientes datos por cada solicitud:   * Nombres * Apellido Paterno * Apellido Materno * Edad * Género * DNI * Seguro SIS (Si lo tiene o no) * Síntomas * Teléfono celular o correo electrónico | | |

|  |  |  |
| --- | --- | --- |
| Historia de usuario | | |
| Número: 3 | Usuario: Licenciado de enfermería | |
| Nombre historia: Clasificar a los pacientes | | |
| Prioridad: 5 | Riesgo: | Iteración: 2 y 4 |
| Descripción:  Como licenciado de enfermería quiero poder clasificar a los pacientes para poder asignar la prioridad correspondiente. | | |
| Criterios de aceptación:  El sistema debe asignar una prioridad que sirva de referencia.  El sistema debe permitir derivar al paciente a los siguientes lugares:   * Tópico Diferenciado * Shock Trauma * Consulta Rápida   El sistema debe permitir registrar los siguientes datos:   * Síntomas * Temperatura * Presión arterial * Saturación * Frecuencia Cardiaca * Frecuencia Respiratoria * Peso * Talla   El sistema debe poder calcular los siguientes datos:   * Superficie corporal * Índice de masa corporal * Clasificación del IMC * Riesgo de enfermedad | | |

# Física

La vista física nos dará un mapeo del software a desarrollar desde el punto de vista del hardware reflejando los distintos componentes de hardware conectados para lograr el funcionamiento del programa (Kruchten, s. f.). Para esta vista, se hará uso del diagrama de despliegue.



## Computador de Hospital

Hace referencia al dispositivo usado por los usuarios de los hospitales pertenecientes al área de emergencias, el cual se conectará mediante el uso de algún navegador: Chrome y FireFox.

## Computador de cliente

Hace referencia al dispositivo usado por los pacientes, desde donde se brindará información sobre su situación actual para ser priorizados con anterioridad. Este computador se podrá conectar mediante el uso de algún navegador: Chrome y FireFox.

## Servidor Front Web Paciente

Este es el contenedor Amazon S3 ubicado en una nube privada virtual (VPC) de Amazon Web Services (AWS) en el cual se va a desplegar el código fuente para las interfaces de usuario para los pacientes a través del cual podrán brindar información al hospital deseado.

## Servidor Front Web Hospital

Este es el contenedor Amazon S3 ubicado en una nube privada virtual (VPC) de Amazon Web Services (AWS) en el cual se va a desplegar el código fuente para las interfaces de usuario para los trabajadores de los hospitales del área de emergencias.

## Servidor Back Hospital

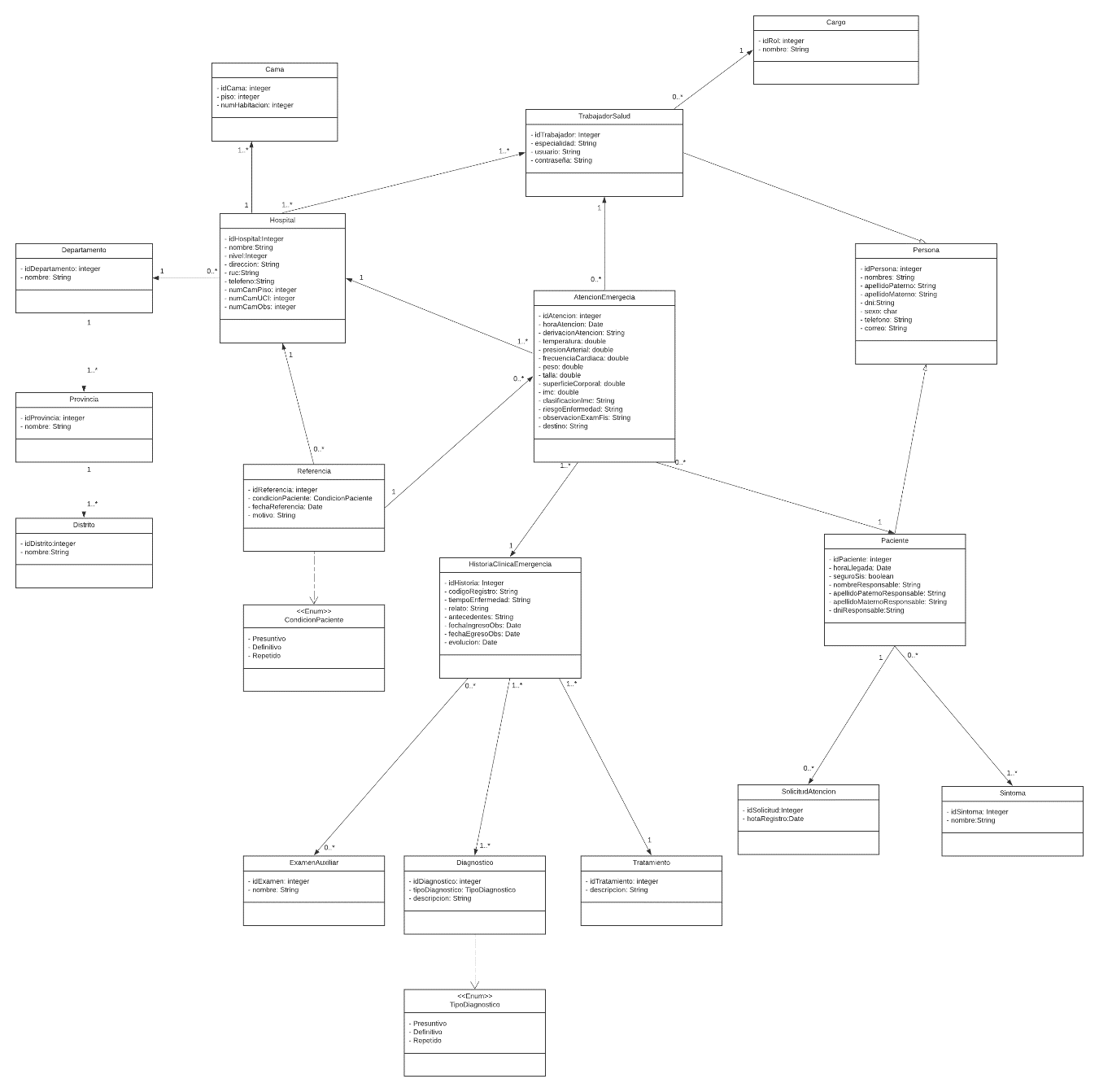
Esta es la instancia Amazon EC2 en la subred privada de la nube privada virtual (VPC) de Amazon Web Services (AWS) en el cual se va a desplegar el código fuente del programa Java y Python que contiene la lógica de negocio para el hospital.

## Base de datos MySQL

Este es el contenedor Relational Database Service ubicada en la subred privada de la nube privada virtual (VPC) de Amazon Web Services (AWS) en el cual se va a desplegar el motor de base de datos MySQL.

# Vista Lógica

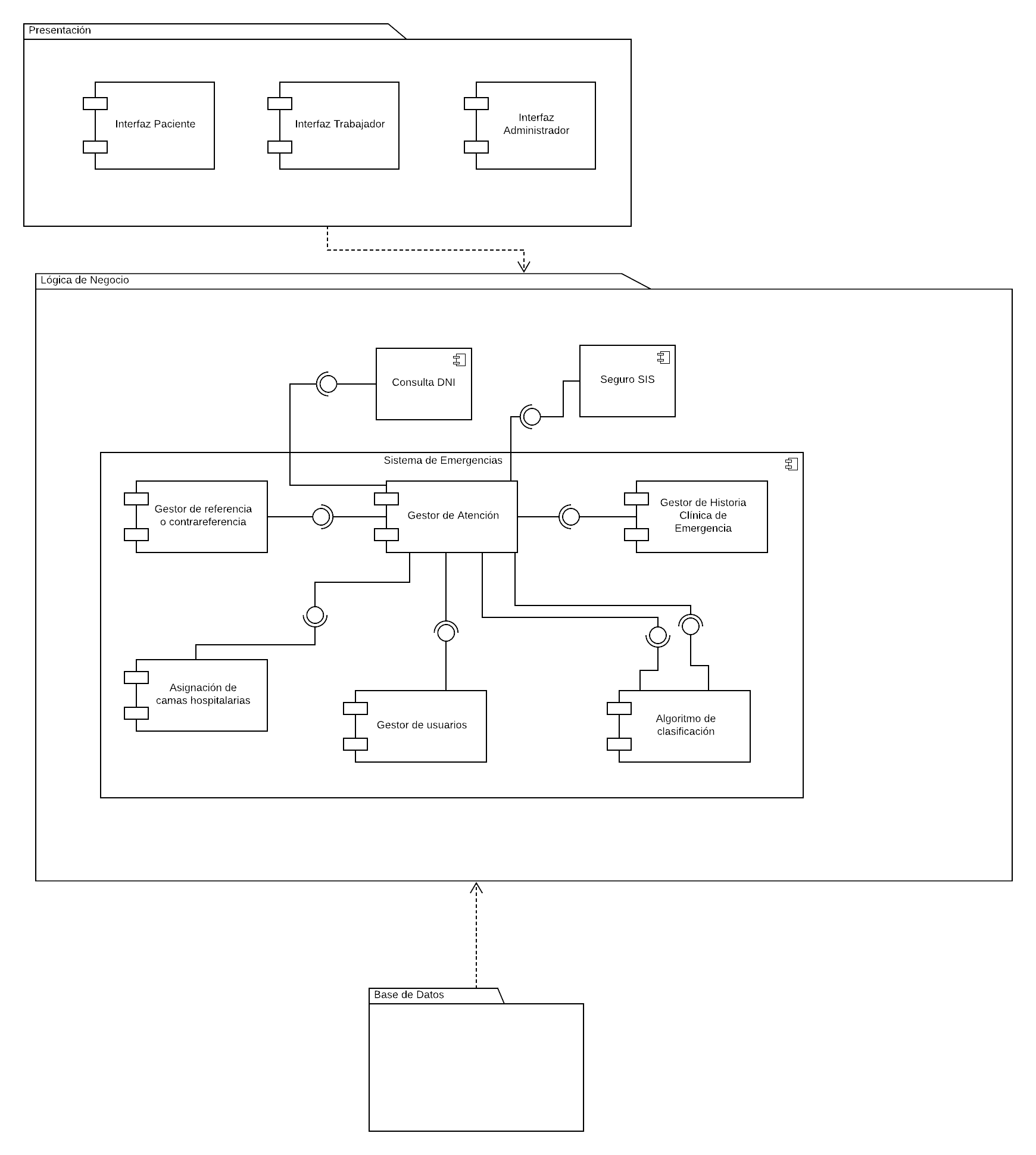
Esta vista describe el modelo de objetos del diseño al usar un método de diseño orientado a objetos (Kruchten, s. f.). Para esta vista, se hará uso del diagrama de clases para representar los objetos (clases) y sus relaciones.



Debido a que se está realizando un patrón de modelo vista controlador, las clases tienen métodos que se limitan a obtener los datos que tienen. Por otro lado, los controladores de cada clase se encargan de gestionar las principales funciones para la adquisición de datos desde la base de datos.

# Vista de desarrollo

Esta vista describe la organización real de los módulos de software en el ambiente de desarrollo del software en donde se empaqueta en partes pequeñas para ser desarrollado por uno persona o grupo desarrollador (Kruchten, s. f.). Para esta vista, se hará uso del diagrama de componentes.



## Presentación

En este paquete se encuentran agrupados los componentes visuales y representa la parte del sistema con la que interactúa los usuarios.

Dentro se encuentra los siguientes componentes:

* **Interfaz del paciente:** Comprende la interfaz que se encarga de recolectar información previa de la condición del paciente y se convierta en una solicitud de atención.
* **Interfaz del trabajador:** Comprende las interfaces para gestionar su cuenta, registrar las atenciones realizadas, registrar las historias clínicas de emergencia y el registro de una referencia o contrareferencia.
* **Interfaz del administrador:** Comprende las interfaces para gestionar la información correspondiente a un hospital y la gestión de cuentas para su hospital.

## Lógica de Negocios

En este paquete se encuentran agrupados los componentes encargados de la operación que solicita el usuario del sistema a través de la interfaz.

Dentro se encuentran los siguientes componentes:

* **Gestor de Usuarios:** Realiza toda operación con creación y modificación de cuentas correspondiente a un hospital y de información general sobre el mismo hospital.
* **Gestor de Atención:** Realiza toda operación de creación y actualización de datos correspondiente a la atención en emergencias de un paciente.
* **Gestor de Historias Clínicas de Emergencia:** Realiza toda operación de creación y modificación de Historias Clínicas de Emergencia correspondiente a los pacientes en una atención por emergencia.
* **Gestor de referencia o contrareferencia:** Realiza toda la operación de creación y modificación de referencias o contrareferencias realizadas a pacientes además de mostrar el estado actual del establecimiento de salud destino.
* **Asignación de camas hospitalarias:** Realiza toda la operación con la asignación de camas dando a conocer el estado actual de capacidad y obtener sugerencias de ubicación para un paciente.
* **Algoritmo de clasificación de pacientes:** Se alimenta de la información sobre la condición actual del paciente para poder asignar una prioridad al paciente.

Dentro se encuentran los siguientes componentes externos:

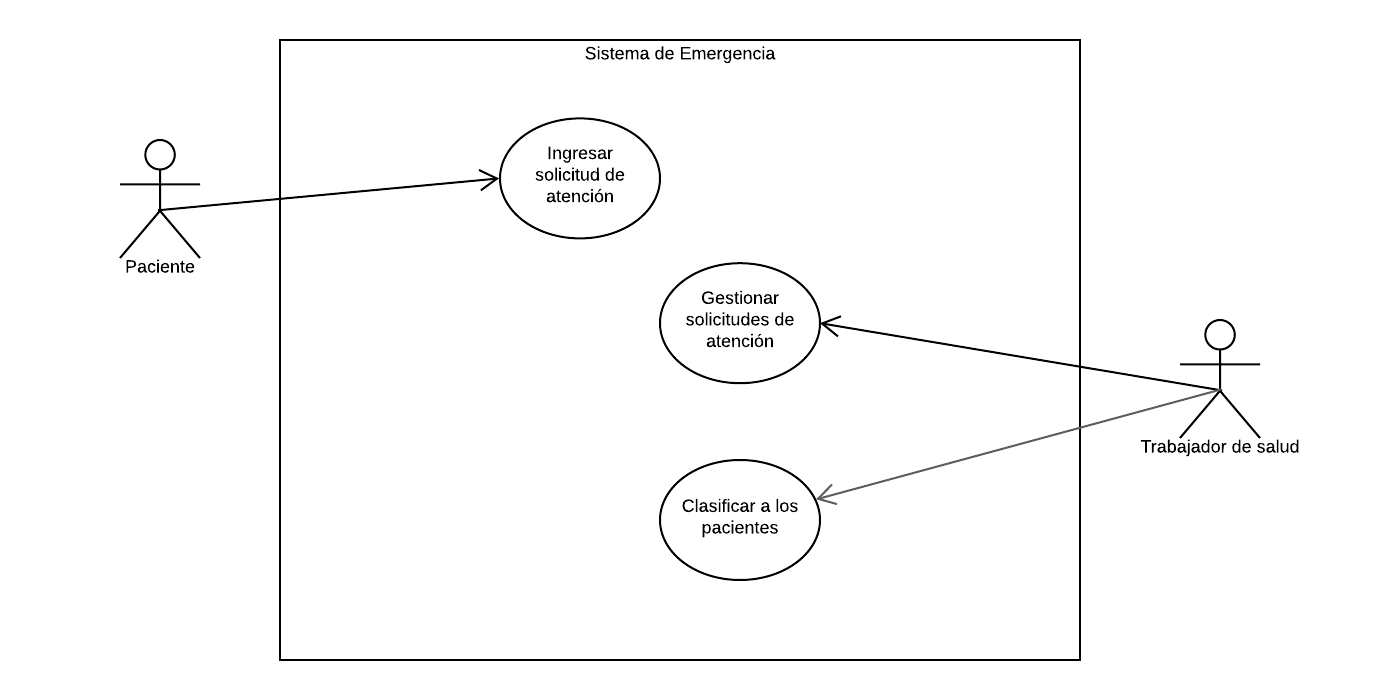
* **Consulta DNI:** Servicio externo que provee el nombre correspondiente al DNI que se digita.
* **Seguro SIS:** Servicio externo que indica si el DNI brindado corresponde a una persona con seguro SIS.

## Base de datos

Este paquete representa el uso de una base de datos, realizado en MySQL en este caso, para almacenar toda la información a usar durante todo el proceso.

# Casos de Uso

Se encargan de modelar el comportamiento de un sistema capturando los requisitos funcionales del sistema a desarrollar (Universitat Politecnica de Valencia, 2017). A continuación, se mostrará los diagramas de casos de uso.



## Descripción de actores

|  |  |  |
| --- | --- | --- |
| **Actor** | **Descripción** | **Abreviatura** |
| Paciente | Es la persona que será atendida en el establecimiento de salud con la opción de adelantar su condición actual por medio la aplicación. | PAC |
| Trabajador de salud | Es el encargado de atender al paciente durante su estadía en el área de emergencia. | TSA |

## Descripción de casos de uso

|  |  |  |  |
| --- | --- | --- | --- |
| **Código** | **Título** | **Descripción** | **Actor(es)** |
| CU.01 | Ingresar solicitud de atención | En este caso de uso se permite a los pacientes el ingreso de una solicitud de atención para adelantar su situación actual. | PAC |
| CU.02 | Gestionar solicitudes de atención | En este caso de uso se permite al trabajador de salud la visualización, modificación, borrado y conversión de las solicitudes de atención. | TSA |
| CU.03 | Clasificar a los pacientes | En este caso de uso se permite la asignación de prioridad al paciente. | TSA |

## Especificación de casos de uso

### Caso de uso CU.01: Ingresar solicitud de atención

|  |  |
| --- | --- |
| Código caso de uso | CU.01 |
| Nombre de caso de uso | Ingresar solicitud de atención |
| Actores | Paciente |
| Descripción | En este caso de uso se permite a los pacientes el ingreso de una solicitud de atención para adelantar su situación actual. |
| Pre condición | - |
| Post condición | Se ha registrado una solicitud de atención al sistema. |
| Flujo Básico | 1. El usuario ingresa a la URL del sistema y selecciona el hospital donde quiere atenderse. 2. El sistema muestra un formulario donde se solicita los siguientes datos:  * Edad * DNI * Síntomas * Teléfono celular o correo electrónico  1. El usuario ingresa los siguientes datos:  * Edad * DNI * Síntomas * Teléfono celular o correo electrónico  1. El sistema llena automáticamente los campos de Nombre, Apellido Paterno y Apellido Materno haciendo uso del DNI. 2. El usuario selecciona Registrar. 3. El sistema muestra un mensaje de confirmación. |
| Flujo excepcional | DNI no encontrado   1. El sistema no completa Nombres, Apellido Paterno y Apellido Materno en el paso 4. 2. El sistema muestra un mensaje de error.   Síntomas faltantes   1. El usuario no ingresa ningún síntoma en el paso 3. 2. El usuario selecciona Registrar. 3. El sistema muestra un mensaje de campo síntomas obligatorio.   DNI faltante   1. El usuario no ingresa el DNI en el paso 3. 2. El usuario selecciona Registrar. 3. El sistema muestra un mensaje de campo DNI obligatorio.   Edad faltante   1. El usuario no ingresa la edad en el paso 3. 2. El usuario selecciona Registrar. 3. El sistema muestra un mensaje de campo edad obligatorio.   Teléfono o correo faltante   1. El usuario no ingresa el teléfono o correo en el paso 3. 2. El usuario selecciona Registrar. 3. El sistema muestra un mensaje de campo teléfono o correo obligatorio.   Cancelar registro   1. En cualquier paso, el usuario puede cancelar el registro de solicitud de atención. |

### Caso de Uso CU.02: Gestionar solicitudes de atención

|  |  |
| --- | --- |
| Código caso de uso | CU.02 |
| Nombre de caso de uso | Gestionar solicitudes de atención |
| Actores | Trabajador de salud |
| Descripción | En este caso de uso se permite al trabajador de salud la visualización, modificación y borrado de las solicitudes de atención. |
| Pre condición | El usuario debe haber iniciado sesión en el sistema. |
| Post condición | Se ha actualizado la información de las solicitudes en el sistema. |
| Flujo Básico | 1. El usuario selecciona “Solicitudes de atención”. 2. El sistema muestra la relación de solicitudes de atención. 3. El usuario selecciona “Crear”. 4. El sistema muestra un formulario en blanco con los siguientes datos:  * Nombres * Apellido Paterno * Apellido Materno * Edad * Seguro SIS * DNI * Síntomas * Teléfono celular o correo electrónico  1. El usuario ingresa los siguientes datos:  * Edad * DNI * Síntomas * Teléfono celular o correo electrónico  1. El sistema completa automáticamente Nombres, Apellido Paterno, Apellido Materno y Seguro SIS. 2. El usuario selecciona “Aceptar”. 3. El sistema muestra un mensaje de confirmación. |
| Flujo Alternativo | Modificar una solicitud   1. El usuario selecciona “Solicitudes de atención”. 2. El sistema muestra la relación de solicitudes de atención. 3. El usuario selecciona una solicitud de atención. 4. El usuario selecciona “Modificar”. 5. El sistema muestra un formulario con los siguientes datos completos y modificables:  * Nombres * Apellido Paterno * Apellido Materno * Seguro SIS * Edad * DNI * Síntomas (Modificable) * Teléfono celular o correo electrónico (Modificable)  1. El usuario modifica alguno de los siguientes datos:  * Nombres * Apellido Paterno * Apellido Materno * Edad * DNI * Síntomas * Teléfono celular o correo electrónico  1. El usuario selecciona “Aceptar”. 2. El sistema muestra un mensaje de confirmación.   Eliminar una solicitud   1. El usuario selecciona “Solicitudes de atención”. 2. El sistema muestra la relación de solicitudes de atención. 3. El usuario selecciona una solicitud de atención. 4. El usuario selecciona “Borrar”. 5. El sistema muestra un mensaje de verificación. |
| Flujo excepcional | Campo faltante encontrado   1. Si luego del paso 6 del flujo alternativo de modificación el usuario deja un campo en blanco, el sistema muestra un mensaje de error de campo faltante.   DNI no encontrado   1. El sistema no completa Nombres, Apellido Paterno y Apellido Materno en el paso 5 del flujo base. 2. El sistema muestra un mensaje de error.   Síntomas faltantes   1. El usuario no ingresa ningún síntoma en el paso 5 del flujo base. 2. El usuario selecciona Registrar. 3. El sistema muestra un mensaje de campo síntomas obligatorio.   DNI faltante   1. El usuario no ingresa el DNI en el paso 5 del flujo base. 2. El usuario selecciona Registrar. 3. El sistema muestra un mensaje de campo DNI obligatorio.   Edad faltante   1. El usuario no ingresa la edad en el paso 3. 2. El usuario selecciona Registrar. 3. El sistema muestra un mensaje de campo edad obligatorio.   Teléfono o correo faltante   1. El usuario no ingresa el teléfono o correo en el paso 5 del flujo base. 2. El usuario selecciona Registrar. 3. El sistema muestra un mensaje de campo teléfono o correo obligatorio.   Cancelar modificación, borrado o creación   1. En cualquier paso, el usuario puede cancelar la modificación, borrado o creación de la solicitud de atención. |

### Caso de Uso CU.03: Clasificar a los pacientes

|  |  |
| --- | --- |
| Código caso de uso | CU.03 |
| Nombre de caso de uso | Clasificar a los pacientes |
| Actores | Trabajador de salud |
| Descripción | En este caso de uso se permite la asignación de prioridad al paciente. |
| Pre condición | El usuario debe haber iniciado sesión en el sistema. |
| Post condición | Se ha actualizado la información de las atenciones en el sistema con su prioridad. |
| Flujo Básico | 1. El usuario selecciona “Triaje”. 2. El sistema muestra una relación de solicitudes de atención. 3. El usuario selecciona una solicitud de atención. 4. El usuario selecciona “Realizar triaje”. 5. El sistema muestra un formulario con los siguientes datos completos, modificables y en blanco:  * Nombres * Apellido Paterno * Apellido Materno * Edad * DNI * Seguro SIS * Síntomas (Modificable) * Teléfono celular o correo electrónico * Temperatura (En blanco) * Presión arterial (En blanco) * Saturación (En blanco) * Frecuencia cardiaca (En blanco) * Frecuencia Respiratoria (En blanco) * Peso (En blanco) * Talla (En blanco) * Superficie corporal (En blanco) * Clasificación del IMC (En blanco) * Riesgo de enfermedad (En blanco)  1. El usuario ingresa los siguientes datos:  * Temperatura * Presión arterial * Frecuencia cardiaca * Frecuencia respiratoria * Peso * Talla  1. El sistema calcula los siguientes datos:  * Superficie corporal * Índice de masa corporal * Clasificación del IMC * Riesgo de enfermedad  1. El usuario selecciona “Asignar prioridad”. 2. El sistema muestra una prioridad de referencia. 3. El usuario decide la prioridad a asignar. 4. El usuario selecciona “Guardar”. 5. El sistema muestra un mensaje de confirmación. |
| Flujo Alternativo | Modificar los síntomas   1. En el paso 6 puede modificar los síntomas. 2. Seguir con los pasos 7 al 12. |
| Flujo excepcional | Campo faltante encontrado   1. Si luego del paso 6 el usuario deja un campo en blanco, el sistema muestra un mensaje de error de campo faltante.   Cancelar clasificación   1. En cualquier paso, el usuario puede cancelar la clasificación del paciente. |

# Vista de procesos

La vista de procesos nos muestra al conjunto de tareas que forman una unidad ejecutable que componen a un proceso (Kruchten, s. f.). Para esta vista, se han modelado, en formato BPMN, los procesos de atención en emergencia.

Para encontrar los modelos, se debe revisar siguiente anexo: **AnexoF\_ModelosProcesoAtenciónEmergencia.**

# Referencias

Kruchten, P. (s. f.). *Planos Arquitectónicos: El Modelo de “4+1” Vistas de la Arquitectura del Software*. 16.

Universitat Politecnica de Valencia. (2017). *Diagrama de casos de uso | | UPV* [MP4]. https://www.youtube.com/watch?v=orvAkFFWo5o

Contexto del caso:

## Objetivos

A continuación, se mencionará el objetivo general y los objetivos específicos del presente trabajo.

### Objetivo general

Desarrollar una herramienta que brinde información oportuna en triaje, en la hospitalización o referencia y al generar Historias Clínicas de Emergencia en entidades de salud para innovar la asignación de prioridades a pacientes mediante una referencia inteligente, la asignación de camas hospitalarias y referencias o contra referencias con información detallada del establecimiento de salud destino, y la gestión del formato de Historias Clínicas de Emergencias.

### Objetivos específicos

A continuación, se detallan los objetivos específicos:

1. Implementar un componente que permita obtener información oportuna para usarlo de referencia a la hora de asignar una prioridad a un paciente.
2. Implementar un componente que permita obtener información oportuna sobre las camas hospitalarias y condición de establecimientos de salud destino al momento de asignar una cama hospitalaria o hacer una referencia.
3. Implementar un componente que permita obtener información oportuna de las Historias Clínicas de Emergencia.

### Resultados esperados

1. Implementar un componente que permita obtener información oportuna para usarlo de referencia a la hora de asignar una prioridad a un paciente.

* Módulo que permita el registro de datos tomados en triaje permitiendo que el paciente pueda adelantar síntomas mediante la aplicación bajo declaración jurada.
* Algoritmo que asigne prioridad propuesta a un paciente para que luego el personal de salud decida la prioridad del paciente en base a los síntomas que padece y la prioridad propuesta.

1. Implementar un componente que permita obtener información oportuna sobre las camas hospitalarias y condición de establecimientos de salud destino al momento de asignar una cama hospitalaria o hacer una referencia.

* Módulo que muestre y permita actualizar los estados de las camas hospitalarias brindando sugerencia sobre donde ubicar a los pacientes atendidos en emergencia y que necesitan ser hospitalizados.
* Módulo que permita la referencia o contra referencia hospitalaria mostrando el detalle del estado del hospital destino.

1. Implementar un componente que permita obtener información oportuna de las Historias Clínicas de Emergencia.

* Módulo que permita el registro y consulta de la información ingresada por el personal de salud contenida en el formato de la Historia Clínica de Emergencia de manera segura.

### Mapeo de objetivos, resultados y verificación

|  |  |  |
| --- | --- | --- |
| Tabla 1: | | |
| *Relación de objetivos específicos, resultado, medio de verificación e indicador objetivamente verificable* | | |
| **Objetivo:** Implementar un componente que permita obtener información oportuna para usarlo de referencia a la hora de asignar una prioridad a un paciente. | | |
| **Resultado** | **Medio de verificación** | **Indicador objetivamente verificable** |
| Módulo que permita el registro de datos tomados en triaje permitiendo que el paciente pueda adelantar síntomas mediante la aplicación bajo declaración jurada. | * Código fuente del componente que muestra información sobre enfermedades comunes por prioridad. * Manual técnico y funcional | * Resultado de las pruebas unitarias aprobadas al 100% al cumplir con los casos de uso. * Resultado de la evaluación del juicio de experto aprobatoria por parte de los tres médicos. |
| Algoritmo que asigne prioridad propuesta a un paciente para que luego el personal de salud decida la prioridad del paciente en base a los síntomas que padece y la prioridad propuesta. | * Código fuente del algoritmo encargado de proponer un turno al paciente. * Manual técnico y funcional. | * Resultado de las pruebas unitarias aprobadas al 100% al cumplir con los casos de uso. * Resultado de la evaluación del juicio de experto aprobatoria por parte de los tres médicos. |
| **Objetivo:** Implementar un componente que permita obtener información oportuna sobre las camas hospitalarias y condición de establecimientos de salud destino al momento de asignar una cama hospitalaria o hacer una referencia. | | |
| **Resultado** | **Medio de verificación** | **Indicador objetivamente verificable** |
| Módulo que muestre y permita actualizar los estados de las camas hospitalarias brindando sugerencia sobre donde ubicar a los pacientes atendidos en emergencia y que necesitan ser hospitalizados. | * Código fuente del componente que muestra y permita actualizar los estados de las camas hospitalarias. * Manual técnico y funcional. | * Resultado de las pruebas unitarias aprobadas al 100% al cumplir con los casos de uso. * Resultado de la evaluación del juicio de experto aprobatoria por parte de los tres médicos. |
| Módulo que permita la referencia o contra referencia hospitalaria mostrando el detalle del estado del hospital destino. | * Código fuente del componente que muestra a los pacientes que necesitan una cama hospitalaria. * Manual técnico y funcional. | * Resultado de las pruebas unitarias aprobadas al 100% al cumplir con los casos de uso. * Resultado de la evaluación del juicio de experto aprobatoria por parte de los tres médicos. |
| **Objetivo:** Implementar un componente que permita obtener información oportuna de las Historias Clínicas de Emergencia. | | |
| **Resultado** | **Medio de verificación** | **Indicador objetivamente verificable** |
| Módulo que permita el registro y consulta de la información ingresada por el personal de salud contenida en el formato de la Historia Clínica de Emergencia de manera segura. | * Código fuente del componente que recibe la información ingresada por el personal de salud contenida en la Historia Clínica de Emergencia. * Manual técnico y funcional | * Resultado de las pruebas unitarias aprobadas al 100% al cumplir con los casos de uso. * Resultado de la evaluación del juicio de experto aprobatoria por parte de los tres médicos. |

## AWS (Amazon Web Services):

Es una plataforma en la nube que ofrece distintos servicios integrales (*¿Qué es AWS?*, s/f). Se usará para poder hacer uso de los servicios EC2 y RDS para el despliegue del software a desarrollar.

## Vue.js:

Es un framework usado para desarrollar interfaces de usuario (*Vue.Js - The Progressive JavaScript Framework*, s/f). Se usará para poder desarrollar las interfaces necesarias en base a los lenguajes JavaScript, CSS y HTML que hace uso el framewok.

## Spring Boot:

Es un framework basado en Java que permite crear programas de manera rápida, fácil y segura (*Why Spring?*, s/f). Se usará para poder desarrollar la parte de la lógica y procesamiento de datos de los componentes y algoritmo a desarrollar.

## The Jupyter Notebook:

Es una aplicación web libre que te ayuda a crear documentos que contienen código en vivo, ecuaciones, texto narrativo y visualizaciones (*Project Jupyter*, s/f). Se usará para poder experimentar y probar el algoritmo de priorización de clientes en triaje realizado en lenguaje python.

## Visual Studio Code:

Es un editor de código fuente que tiene la capacidad de soportar múltiples lenguajes de programación (*Documentation for Visual Studio Code*, s/f). Se hará uso para poder codificar las interfaces haciendo uso del framework Vue.js.

## Eclipse:

Es un entorno integrado de desarrollo que da soporte para codificar en distintos lenguajes y permite su compilación (Guindon, s/f). Se hará uso para poder codificar la parte de la lógica y procesamiento de datos junto con la ayuda de Spring Boot.

## GitLab:

Es una plataforma DevOps libre que ayuda al desarrollo, seguridad y al equipo de desarrollo en la forma de construir y colaborar un software (*The DevOps Lifecycle with GitLab*, s/f). Se hará uso para mantener seguro el código de desarrollado de los componentes y algoritmo. Además, servirá para manejar las versiones del código fuente y para tenerlo accesible a las personas cuando el proyecto se termine y sea adecuada a las necesidades de los interesados.

## MySQL:

Es una base de datos libre que ayuda con el desarrollo de aplicaciones de bases de datos escalables y de alto rendimiento (*MySQL :: MySQL Products*, s/f). Se hará uso para modelar y desarrollar la base de datos necesaria para que los complementos desarrollados guarden y consulten la información que necesiten para su funcionamiento.

## MySQL Workbench:

Es una herramienta visual para arquitectos y desarrolladores de bases de datos, el cual hace uso de la base de datos MySQL (*MySQL :: MySQL Workbench*, s/f). Se hará uso para facilitar el modelamiento de base de datos para los componentes a desarrollar.

## Metodología Scrum:

La metodología Scrum es un proceso de desarrollo de software ágil diseñada para dar claridad, enfoque, energía y transparencia a los equipos desarrolladores de sistemas de software (Phalnikar et al., 2009). Se usará para poder definir cada resultado esperado como un conjunto de historias de usuario, las cuales definen lo que necesita el usuario como funcionalidad (Joskowicz, s/f), y poder relacionarlos con un sprint en específico, el cual es un periodo de un mes o menos para crear parte del producto a desarrollar y que funcione (Schwaber & Sutherland, 2017).