



UNIVERSIDAD
NACIONAL
DE COLOMBIA

Universidad Nacional de Colombia - sede Bogotá
Facultad de Ingeniería
Departamento de Sistemas e Industrial
Curso: Ingeniería de Software 1 (2016701)

Grupo Passione

Jorge Alandete-jalandete@unal.edu.co Maicol Garzón-mgarzonm@unal.edu.co
Santiago Maldonado-samaldonado@unal.edu.co Juan Medina-jumedinaca@unal.edu.co

Taller 01

Listado de requerimientos

1. Registro de refugiados

- a. El usuario con rol de personal debe poder registrar datos basicos de registro de un refugiado como: el nombre completo, Número de cédula (si tiene), Edad, Sexo, Ciudad de origen, Estado de salud, Fecha y hora de ingreso y Firma o huella.

2. Control de disponibilidad de camas en tiempo real

- a. El sistema debería asignar cama a un refugiado de forma automática de acuerdo a disponibilidad
- b. El sistema debe registrar una cama como disponible cuando se registre la salida de una persona.
- c. El usuario con rol de personal debe poder habilitar y deshabilitar camas.
- d. El sistema debe mostrar la disponibilidad de camas en tiempo real.
- e. El usuario con rol de personal debe poder asignar camas manualmente.

3. Historial de las personas

- a. El sistema deberá llevar un historial de los registros de las personas.
- b. El usuario debe poder consultar el historial de las personas.
- c. Los usuarios deben poder consultar el historial de otros refugios.
- d. El usuario debe poder abrir y modificar historial de estancia de los refugiados

4. Control de donaciones

- a. El usuario debe poder registrar datos de las donaciones que se realizan a la organización como: tipo de donación, elemento donado, centro de destino, recepción, ubicación.
- b. El usuario podrá crear solicitudes de donación de elementos.

- c. El usuario podrá aceptar solicitudes de donación para trasladar elementos de un centro a otro.
- d. El usuario debe poder ver las solicitudes de donación de otros centros de refugio.
- e. El sistema debe llevar registro de la cantidad de elementos donados
- f. El sistema de control de donaciones debe estar disponible para todos los centros.

5. Panel general

- a. El sistema debe mostrar información de los diferentes refugios al usuario con rol administrativo.
- b. El sistema debe tener la capacidad de generar reportes con la información disponible e histórica.
- c. El usuario debería poder filtrar la información según datos como: fechas, centros, personas.

6. Pre-registros comunitarios

- a. El sistema debe permitir el pre-registro de personas a través de medios externos.

7. Registro y control de empleados

- a. Un usuario con rol de administrador debe poder crear, modificar e inhabilitar cuentas para los empleados de la organización.

8. Autenticación

- a. El sistema debe permitir el acceso a empleados con cuentas autorizadas
- b. El sistema, dependiendo el rol del usuario, permitirá el acceso o no a funcionalidades.

9. Administración e información de turnos

- a. Un usuario con rol de administrador debe poder asignar y modificar turnos.
- b. El sistema debe mostrar información de los turnos.
- c. El usuario puede marcar su entrada y salida de un turno.

10. Implementación

- a. El sistema debe tener la capacidad de funcionar en dispositivos móviles y en computadora
- b. El sistema debería tener tiempo de respuesta eficiente en dispositivos móviles

Clasificación de requisitos

FUNCIONALES	NO FUNCIONALES
<ul style="list-style-type: none"> - RF_1a: El usuario con rol de personal debe poder registrar datos básicos de un refugiado (nombre, cédula, edad, etc.). - RF_2b: El sistema debe registrar una cama como disponible al registrar la salida de una persona. - RF_2c: El usuario con rol de personal debe poder habilitar y deshabilitar camas. - RF_2d: El sistema debe mostrar disponibilidad de camas en tiempo real. - RF_2e: El usuario con rol de personal debe poder asignar camas manualmente. - RF_3a: El sistema deberá llevar un historial de los registros de las personas. - RF_3b: El usuario debe poder consultar el historial de las personas. - RF_3c: Los usuarios deben poder consultar el historial de otros refugios. - RF_3d: El usuario debe poder abrir y modificar historial de estancia. - RF_4a: El usuario debe poder registrar datos de las donaciones (tipo, elemento, ubicación, etc.). - RF_4b: El usuario podrá crear solicitudes de donación. - RF_4c: El usuario podrá aceptar solicitudes de donación. - RF_4d: El usuario debe poder ver solicitudes de otros centros. - RF_4e: El sistema debe llevar 	<ul style="list-style-type: none"> - RFN_2a: El sistema debe asignar cama a un refugiado automáticamente según disponibilidad. - RFN_4f: El sistema de donaciones debe estar disponible para todos los centros. - RFN_5c: El usuario debería poder filtrar la información (fechas, centros, personas). - RFN_8a: El sistema debe permitir acceso a empleados con cuentas autorizadas. - RFN_10a: El sistema debe funcionar en dispositivos móviles y en computadora. - RFN_10b: El sistema debería tener tiempo de respuesta eficiente en dispositivos móviles.

<p>registro de elementos donados.</p> <ul style="list-style-type: none">- RF_5a: El sistema debe mostrar información de los refugios al usuario administrador.- RF_5b: El sistema debe generar reportes con información histórica.- RF_6a: El sistema debe permitir pre-registro de personas desde medios externos.- RF_7a: El administrador debe poder crear, modificar e inhabilitar cuentas de empleados.- RF_8b: El sistema debe controlar acceso a funcionalidades según rol del usuario.- RF_9a: El administrador debe poder asignar y modificar turnos.- RF_9b: El sistema debe mostrar información de turnos.- RF_9c: El usuario puede marcar entrada y salida de turnos.	
--	--

Priorización MoSCoW:

MUST

- **RF_1a:** El usuario con rol de personal debe poder registrar datos basicos de registro de un refugiado como: el nombre completo, Número de cédula (si tiene), Edad, Sexo, Ciudad de origen, Estado de salud, Fecha y hora de ingreso y Firma o huella.
- **RF_2b:** El sistema debe registrar una cama como disponible cuando se registre la salida de una persona.
- **RF_2c:** El usuario con rol de personal debe poder habilitar y deshabilitar camas para que el sistema pueda o no asignarlas.
- **RF_2d:** El sistema debe mostrar la disponibilidad de camas en tiempo real.
- **RF_2e:** El usuario con rol de personal debe poder asignar camas manualmente.
- **RF_3a:** El sistema debe llevar un historial de los registros de las personas.
- **RF_3b:** El usuario con rol de personal debe poder consultar el historial de las personas.

- **RF_3c:** Los usuarios deberán poder consultar el historial de otros refugios.
- **RF_3d:** El usuario debe poder abrir y modificar historial de estancia de los refugiados
- **RF_4a:** El usuario con rol de personal debe poder registrar datos de las donaciones que se realizan a la organización como: tipo de donación, elemento donado, centro de destino, recepción, ubicación.
- **RF_4b:** El usuario con rol de personal debe poder crear solicitudes de donación de elementos.
- **RF_4c:** El usuario con rol de administrador debe poder aceptar solicitudes de donación para trasladar elementos de un centro a otro.
- **RF_4d:** El usuario con rol de administrador debe poder ver las solicitudes de donación de otros centros de refugio.
- **RF_4e:** El sistema debe llevar registro de la cantidad de elementos donados
- **RNF_4f:** El sistema de control de donaciones debe estar disponible para todos los centros.
- **RF_5a:** El sistema debe mostrar información de los diferentes refugios al usuario con rol administrativo.
- **RF_7a:** Un usuario con rol de administrador debe poder crear, modificar e inhabilitar cuentas para los empleados de la organización.
- **RNF_8a:** El sistema debe permitir el acceso a empleados con cuentas autorizadas
- **RF_8b:** El sistema, dependiendo el rol del usuario, debe permitir el acceso o no a las funcionalidades.
- **RNF_10a:** El sistema debe tener la capacidad de funcionar en dispositivos móviles y en computadora

SHOULD

- **RNF_2a:** El sistema debería asignar cama a un refugiado de forma automática de acuerdo a disponibilidad
- **RNF_5c:** El usuario con rol de personal o administrador debería poder filtrar la información de consulta según datos como: fechas, centros, personas.

COULD

- **RF_5b:** El sistema podría tener la capacidad de generar reportes con la información disponible e histórica.
- **RF_9a:** Un usuario con rol de administrador podría asignar y modificar turnos.
- **RF_9b:** El sistema podría mostrar información de los turnos.

- **RF_9c:** El usuario podría marcar su entrada y salida de un turno.
- **RNF_10b:** El sistema podría tener tiempo de respuesta menor a 5s en dispositivos móviles

WONT

- **RF_6a:** No es necesario que el sistema permita el pre-registro de personas a través de medios externos.

Estimación Fibonacci y argumentación

Requisito		Estimación	Argumento
RF_1a	El usuario con rol de personal debe poder registrar datos básicos de un refugiado (nombre, cédula, edad, etc.)	5	Es complejo por el hecho de realizar el diseño y desarrollo de la base de datos
MUST			
RF_2b	El sistema debe registrar una cama como disponible al registrar la salida de una persona	1	Es de dificultad mínima ya que solo es modificar el valor un atributo booleano en la base de datos en la salida de un refugiado
MUST			
RF_2c	El usuario con rol de personal debe poder habilitar y deshabilitar camas	2	Es simple ya que aparte de modificar el valor un atributo booleano en la base de datos se necesita una interfaz con la que interactúa el usuario
MUST			
RF_2d	El sistema debe mostrar disponibilidad de camas en tiempo real	3	Es moderada principalmente por la implementación en tiempo real.
MUST			
RF_2e	El usuario con rol de personal debe poder asignar camas manualmente	2	Es simple porque solo se implementa una interfaz para que el usuario pueda relacionar camas con refugiados
MUST			
RF_3a	El sistema debe llevar un historial de los registros de las personas	5	Es complejo ya que es necesario implementar el historial en la base de datos
MUST			

RF_3b	El usuario con rol de personal debe poder consultar el historial de las personas	2	Es simple porque se debe implementar una interfaz para realizar consultas a la base de datos
MUST			
RF_3c	Los usuarios con rol de personal deberán poder consultar el historial de otros refugios	1	Es de dificultad mínima ya que muchos los motores de bases de datos están diseñadas para ser consultadas desde diferentes dispositivos y el tamaño de la organización es reducido
MUST			
RF_3d	El usuario con rol de personal debe poder abrir y modificar historial de estancia	2	Es simple debido a que es un interfaz que muestra una consulta de la base de datos en base a unos parámetros
MUST			
RF_4a	El usuario con rol de administrador debe poder registrar datos de las donaciones (tipo, elemento, ubicación, etc.)	5	Es complejo ya que es necesario implementar lo necesario del apartado de donaciones en la base de datos
MUST			
RF_4b	El usuario con rol de administrador podrá crear solicitudes de donación	5	Es complejo ya que es necesario implementar una tabla de solicitudes de donaciones
MUST			
RF_4c	El usuario con rol de administrador podrá aceptar solicitudes de donación	3	Es de dificultad media pues es necesario realizar operaciones en la base de datos y además implementar la interfaz necesaria para la interacción con el usuario
MUST			
RF_4d	El usuario con rol de administrador debe poder ver solicitudes de otros centros	2	Es simple debido a que es un interfaz que muestra una consulta de la base de datos en base a unos parámetros
MUST			
RF_4e	El sistema debe llevar registro de elementos donados	1	Es de dificultad mínima, ya que las herramientas de los motores bases de datos brindan las funciones para llevarlo a cabo
MUST			
RF_5a	El sistema debe mostrar	2	Es simple debido a que es un

MUST	información de los refugios al usuario administrador		interfaz que muestra una consulta que sintetice gran parte de la base datos
RF_5b	El sistema podría generar reportes con información histórica	3	Es de dificultad media, pues además de la interfaz de interacción con el usuario y las consultas a la base de dato, es necesario procesar los datos obtenidos para devolver un reporte
COULD			
RF_6a	El sistema debe permitir pre-registro de personas desde medios externos	2	Es simple porque por medio de un formulario web se insertan registros a una tabla
WON'T			
RF_7a	El usuario con rol de administrador debe poder crear, modificar e inhabilitar cuentas de empleados	3	Es de dificultad media porque es necesario implementar una interfaz que permita al administrador modificar un registro en la base de datos
MUST			
RF_8b	El sistema debe controlar acceso a funcionalidades según rol del usuario	2	Es de dificultad simple a causa de las facilidades que otorgar las bases datos modernas para la gestión de usuarios
MUST			
RF_9a	El usuario con rol de administrador podría asignar y modificar turnos	3	Es de dificultad media ya que solo es modificar un registro en la tabla de personal
COULD			
RF_9b	El sistema podría mostrar información de turnos	2	Es simple debido a que es mostrar la información de una consulta de la tabla de personal
COULD			
RF_9c	El usuario con rol de personal podría marcar entrada y salida de turnos	5	Es moderada porque debemos relacionar la tabla de empleados a una tabla de registro de turnos
MUST			
RNF_2a	El sistema debería asignar cama a un refugiado automáticamente según disponibilidad	3	Es de dificultad media ya que se debe consultar por una cama y asignarla a un refugiado
SHOULD			

RNF_4f	El sistema de donaciones debe estar disponible para todos los centros	1	Es de dificultad mínima ya que muchos los motores de bases de datos están diseñadas para ser consultadas desde diferentes dispositivos y el tamaño de la organización es reducido
MUST			
RNF_5c	El usuario con rol de personal debería poder filtrar la información (fechas, centros, personas)	2	Es simple por las herramientas de consulta que nos otorga los motores de bases de datos
SHOULD			
RNF_8a	El sistema debe permitir acceso a empleados con cuentas autorizadas	8	Es difícil debido a la falta de conocimientos del equipo relacionados a la seguridad en el software
MUST			
RNF_10a	El sistema debe funcionar en dispositivos móviles y en computadora	8	Es difícil a causa de la incertidumbre que se tiene del tiempo de respuesta de los procesos de la base de datos, la transmisión de los datos y el poder de procesamiento de los dispositivos
MUST			
RNF_10b	El sistema debería tener tiempo de respuesta eficiente en dispositivos móviles	8	Es difícil a causa de la incertidumbre que se tiene del tiempo de respuesta de los procesos de la base de datos, la transmisión de los datos y el poder de procesamiento de los dispositivos
MUST			