



**Universidad Nacional de Colombia - sede Bogotá**

**Facultad de Ingeniería**

**Departamento de Sistemas e Industrial**

**Curso: Ingeniería de Software 1 (2016701)**

## **Taller 01**

### **Nombres:**

Juan David Serrano Ruiz  
Federico Hernández Montaña  
Diego Ospina Ladino  
Juan David Montenegro

### **Requerimientos:**

- **Req\_1: Ingreso y registro de personas:** Un sistema que guarde la información de cada persona que ingrese, tomada por el usuario. Esta información es tomada por un usuario, al momento de un tentativo ingreso de un beneficiario; la información incluye: Nombre completo, numero de cedula (si se tiene), edad, sexo, ciudad de origen, estado de salud, fecha y hora de ingreso y firma o huella. Usando una interfaz en forma de un formulario con una plantilla única y generalizada.
- **Req\_2: Generación de llave única:** Se genera una llave única en el momento del primer registro de un beneficiario que facilitara la identificación de sus datos y el manejo de los futuros
- **Req\_3: Autocompletado del registro:** Sistema automático que mediante una llave única generada en el primer ingreso del beneficiario; la cual idealmente es la huella (o en dado caso el nombre); va a permitir autocompletar los datos invariables del beneficiario en el formulario para agilizar el ingreso. Seguirá siendo necesario completar campos como la fecha y hora de ingreso, el estado de salud y la firma.
- **Req\_4: Asignación de camas:** Sistema automático de control de disponibilidad y asignación de camas en tiempo real. Debe asignar un número de cama según disponibilidad en tiempo real al ser aprobada una entrada. Entregando una respuesta inmediata; en forma de un mensaje flotante; al usuario para poder proceder en la entrada.
- **Req\_5: Historial de estancia:** Sistema automático que guarda y actualiza un historial para cada persona que entra a los centros. El sistema de la recolección de información va a tener un apartado específico llamado historial de estancia, en el cual se registra cada ingreso de un beneficiario a los diferentes centros. Este sistema funcionaria como una mejora para el sistema de ingresos al generar nuevos criterios al momento de una entrada.
- **Req\_6: Turnos del personal:** Sistema manual donde se asignan y administran los turnos de los empleados y permite la consulta de este. Dicho sistema busca una mejor organización en la distribución de turnos además de servir como herramienta para los administradores a la hora de controlar el trabajo de sus empleados

- **Req\_7: Registro y seguimiento de donaciones:** Sistema manual que guarda la información pertinente de las donaciones que reciben los centros (qué entra, a dónde va, si ya se entregó, cuánto queda). Dicho sistema debe ser propio de un punto
- **Req\_8: Vista general de las donaciones:** Interfaz que permita la visualización general de las donaciones que entran en la empresa; por parte de los administrativos; para poder gestionar las mismas.
- **Req\_9: Panel general:** Sistema que permita consultar la información de los centros: ingresos, inventario y camas de los centros. Mediante una nueva interfaz para computador con único acceso administrativo y un manejo supervisado por el usuario
- **Req\_10: Aplicación móvil:** Debido a la comodidad de los usuarios se define que la aplicación para usuarios de registro debe desarrollarse en un modelo móvil, específicamente para dispositivos Android
- **Req\_11: Aplicación administrativa de escritorio:** Para todos los sistemas de acceso administrativo se desarrolla una aplicación compatible con Windows 7 y Windows 10, pues son las tecnologías disponibles.
- **Req\_12: Alerta de necesidades:** Un sistema de comunicación que permita compartir y alertar de las necesidades de cada centro. El cliente propone un sistema para el manejo y visualización de las problemáticas internas de cada centro.
- **Req\_13: Asegurar un sistema comprensible:** Primordialmente los sistemas con interacción de usuario de registro deben asegurar interfaces comprensibles y de fácil uso.
- **Req\_14: Actualización en tiempo real:** Todos los sistemas que registren datos deben hacer una actualización en tiempo real a la base de datos para permitir un trabajo más efectivo para el personal administrativo

Requerimientos funcionales y no funcionales:

| FUNCIONALES  | NO FUNCIONALES  |
|--|---|
| <ul style="list-style-type: none"> <li>- RF_1 (Req_1: Ingreso y registro de personas)</li> <li>- RF_2(Req_2: Generación de llave única)</li> <li>- RF_3(Req_3: Autocompletado del registro)</li> <li>- RF_4(Req_4: Asignación de camas)</li> <li>- RF_5(Req_5: Historial de estancia)</li> <li>- RF_6 (Req_6: Turnos del personal)</li> <li>- RF_7(Req_7: Registro y seguimiento de donaciones)</li> <li>- RF_8(Req_8: Vista general de las donaciones)</li> <li>- RF_9(Req_9: Panel general)</li> <li>- RF_10(Req_12: Alerta de necesidades)</li> </ul> | <ul style="list-style-type: none"> <li>- RNF_1(Req_10: Aplicación móvil)</li> <li>- RNF_2(Req_11: Aplicación administrativa de escritorio)</li> <li>- RNF_3(Req_13: Asegurar un sistema comprensible)</li> <li>- RNF_4(Req_14: Actualización en tiempo real)</li> </ul> |

## MoSCoW

### **MUST (DEBE - OBLIGATORIO SIN ÉL EL SISTEMA PIERDE SENTIDO O NO FUNCIONA)**

- **RF\_1:** Se implementará un formulario digital único que permitirá al usuario registrar la información básica del beneficiario en su primer ingreso. Este formulario será fácil de usar y almacenará los datos en una base de datos segura.
- **RF\_2:** Al registrar por primera vez a un beneficiario, el sistema generará automáticamente un identificador único (llave) que permitirá relacionar y acceder fácilmente a sus datos en futuras ocasiones.
- **RF\_4:** Se creará un sistema que consulte en tiempo real la disponibilidad de camas, al registrar un ingreso aprobado, se asignará automáticamente una cama disponible y el sistema mostrará un mensaje con el número de cama y su ubicación.
- **RNF\_1:** El módulo de registro de beneficiarios se desarrollará como una aplicación para dispositivos Android, facilitando el ingreso de datos desde el terreno o dispositivos móviles.
- **RNF\_3:** Las interfaces estarán diseñadas para ser intuitivas, con un enfoque visual claro y simple, pensando siempre en la facilidad de uso para el personal de registro.

### **SHOULD (DEBERÍA - NO OBLIGATORIO, PERO APORTA MUCHO A LA EXP DE USUARIO)**

- **RF\_3:** Cuando un beneficiario se registre nuevamente, el sistema detectará su llave única y completará automáticamente los datos que no cambian. Solo será necesario llenar información específica del nuevo ingreso.
- **RF\_5:** Cada ingreso se registrará en una sección específica del sistema llamado Historial de estancia, que permitirá llevar un control detallado de todas las veces que un beneficiario ha ingresado a los centros.
- **RF\_7:** Se desarrollará un sistema manual por centro para ingresar, seguir y gestionar las donaciones recibidas, indicando qué se recibió, dónde está, si se entregó y cuánto queda.
- **RF\_9:** Se construirá un sistema administrativo con una interfaz centralizada para acceder a información clave de cada centro (ingresos, inventario, camas).
- **RNF\_2:** Los sistemas con funciones administrativas se desarrollarán como una aplicación de escritorio, compatible con sistemas operativos Windows 7 y 10, adaptándose a las tecnologías actuales del cliente.
- **RNF\_4:** Toda la información registrada en los sistemas se sincronizará inmediatamente con la base de datos, garantizando que el personal administrativo siempre cuente con información actualizada.

### **COULD (PODRÍA - ES UN PLUS USUALMENTE PERSONALIZACIÓN, SE DESARROLLA USUALMENTE CUANDO QUEDA TIEMPO)**

- **RF\_8:** Se ofrecerá un sistema especial para los administrativos que muestre una interfaz general y organizada de todas las donaciones que han llegado a los distintos centros.

### **WONT (NO NECESARIO - PUEDE ESPERAR AL PUNTO QUE ESTÉ O NO PASA NADA)**

**SI NUNCA SE DESARROLLA)**

- **RF\_6:** Un sistema para que los administradores asignen, editen y consulten los turnos del personal. Esto facilitará la organización interna y el seguimiento del trabajo del equipo.
- **RF\_10:** Se implementará un sistema interno de comunicación donde los centros podrán reportar sus necesidades o problemáticas, y éstas podrán visualizarse fácilmente para su atención.

**Estimación de tiempo/esfuerzo Fibonacci:**

| Requisito |                                      | Estimación | Argumento  |
|-----------|--------------------------------------|------------|--|
| RF_1      | RF_1: Ingreso y registro de personas | 9          | Es una tarea de alta dificultad. El equipo no tiene experiencia previa en estas funcionalidades; se requiere integración con base de datos, validaciones y pruebas, además de un sistema de autenticación. |
| MUST      |                                      |            |  |
| RF_2      | RF_2: Generación de llave única      | 3          | Requiere lógica de generación única y almacenamiento básico. Es técnicamente sencillo y con bajo riesgo de errores.  |
| MUST      |                                      |            |  |
| RF_3      | RF_3: Autocompletado del registro    | 3          | No muy complejo, pero involucra recuperación de datos desde la base, validación de identidad y experiencia limitada del equipo con este tipo de lógica.  |
| SHOULD    |                                      |            |  |
| RF_4      | RF_4: Asignación de camas            | 8          | Implica lógica compleja para la asignación eficiente y justa de camas, conexión con múltiples datos (disponibilidad, estado) y pruebas rigurosas.  |
| MUST      |                                      |            |  |
| RF_5      | RF_5: Historial de estancia          | 5          | Es de complejidad media. Requiere una estructura clara de almacenamiento y actualización constante del historial de cada ingreso por beneficiario.   |
| SHOULD    |                                      |            |  |
| RF_6      | RF_6: Turnos del personal            | 5          | Tiene dificultad media-alta debido a la cantidad de personal y la necesidad de manejar   |
| WONT      |                                      |            |  |

|        |  |   |   |
|--------|--|---|---|
|        |  |   | asignaciones en tiempo real o semiautomáticas.  |
| RF_7   | RF_7: Registro y seguimiento de donaciones     | 5 | Involucra formularios, almacenamiento estructurado de datos y autenticación. Su impacto es medio y la complejidad razonable.                |
| SHOULD |  |   |   |
| RF_8   | RF_8: Vista general de las donaciones          | 3 | Es una interfaz que depende de datos ya registrados. Su desarrollo es mayormente visual y con integración a dashboards.                     |
| COULD  |  |   |   |
| RF_9   | RF_9: Panel General                            | 6 | Requiere condensar múltiples fuentes de datos en una interfaz clara, con sincronización en tiempo real. Moderadamente complejo.             |
| SHOULD |  |   |   |
| RF_10  | RF_10: Alerta de necesidades                   | 3 | Tiene utilidad operativa, pero es técnicamente simple. Su implementación básica puede iniciar como un sistema de notificaciones.            |
| WONT   |  |   |   |
| RNF_1  | RNF_1: Aplicación móvil                        | 8 | Es compleja debido al desarrollo nativo, pruebas en dispositivos, compatibilidad y poca experiencia del equipo.                             |
| MUST   |  |   |   |
| RNF_2  | RNF_2: Aplicación administrativa de escritorio | 5 | Tiene complejidad media: se necesita interfaz clara, sincronización con base de datos y compatibilidad con Windows.                         |
| SHOULD |  |   |   |
| RNF_3  | RNF_3: Asegurar un sistema comprensible        | 3 | Requiere enfoque UX/UI, pruebas con usuarios y una estructura visual simple. Su complejidad técnica es baja.                                |
| MUST   |  |   |   |
| RNF_4  | RNF_4: Actualización en tiempo real            | 5 | Requiere tecnologías como sockets o polling para mantener sincronización. Si no hay infraestructura reactiva, puede aumentar la dificultad. |
| SHOULD |  |   |   |

