



**Universidad
de Valparaíso**
CHILE

FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA CIVIL BIOMÉDICA

Implementación de un Sistema de información Hospitalaria de código abierto.

Natalia Valentina Espinoza Carvajal

Trabajo para optar al Título de
Ingeniera Civil Biomédica

Profesor Guía:
Cesar Galindo

Diciembre - 2022

VALPARAÍSO ::: CHILE



Universidad de Valparaíso
Facultad de Ingeniería
Escuela de Ingeniería Civil Biomédica

Información para registro oficial en Biblioteca:

- Proyecto de Ingeniería Biomédica: Área informática médica
- Prof. Coordinador del Área: Cesar Galindo
- Prof. Guía: Cesar Galindo
- Prof. Co-Guía: Carolina Saavedra Ruíz
- Palabras clave: Sistemas de Información, OpenMRS, Registro de Pacientes

Autorización del prof. Guía:

- Fecha versión final para depósito en Biblioteca UV: ___/___/___
- Nombre Prof. Guía: _____
- V°B° Prof. Guía: _____

TABLA DE CONTENIDO

1.	INTRODUCCIÓN	4
2.	METODOLOGÍA E IMPLEMENTACIÓN	5
	Etapa 1: Estudiar el sistema de información hospitalario	5
	Etapa 2: Levantamiento de requerimientos técnicos, funcionales y diseño	5
	Etapa 3: Instalar Bahmni	6
	Etapa 4: Configuración de Bahmni	7
	4.1 Cargar el diccionario de conceptos	8
	4.2 Internacionalización	8
	4.3 Crear usuarios	9
	4.4 Configurar variables globales	9
	4.5 Configurar atributos de paciente en el registro de pacientes	9
	4.5 Configurar jerarquía de direcciones en el registro de nuevos pacientes	11
3.	RESULTADOS Y DISCUSIÓN	11
4.	CONCLUSIÓN	16
5.	ANEXOS	18
	Anexo 1: Instalar Virtual Box, configurar CentOS y configurar el acceso a internet	18
	Anexo 2: Instalar Bahmni	21
	Anexo 3: Contenido del archivo para la internacionalización del módulo registro.	25
	Anexo 4: Contenido del archivo para la internacionalización del módulo clínico.	27
	Anexo 5: Contenido del app.json	30
6.	REFERENCIAS	33

1. INTRODUCCIÓN

En América Latina y el Caribe es prioritario reducir las severas restricciones a la atención médica oportuna y de calidad, especialmente para las poblaciones más pobres y aquellas que viven en áreas remotas. A nivel mundial, existe evidencia de que la transformación digital de las organizaciones de salud impulsa el proceso de cambio. [1] [2]

En nuestro país, existe la Estrategia Digital en Salud, la cual busca digitalizar procesos clínicos y administrativos en establecimientos de salud, incorporando TIC, con la finalidad de contar con información que apoye la toma de decisiones en los distintos niveles de gestión del sector salud [3]. Según el reporte del ministerio de salud en el año 2015, el 53% de los hospitales en Chile tenía implementado un registro clínico electrónico (RCE), mientras que en las organizaciones de atención primaria de salud (APS) el 79% contaba con esta aplicación. [4]

Hay dos aspectos clave en la información de salud: es confidencial, pero al mismo tiempo necesita ser compartida para que los pacientes reciban una atención oportuna y de calidad. La interoperabilidad es la capacidad de diferentes sistemas para conectarse y posteriormente intercambiar información.

La incorporación de tecnologías de la información y de las comunicaciones (TIC) a los sistemas de salud se denomina eSalud. Las principales motivaciones para la aplicación de las TIC por las organizaciones de salud, es que son herramientas que favorecen la calidad, la seguridad y la continuidad del cuidado asistencial. Otros beneficios son la reducción de las limitaciones del acceso a la salud, evitar la duplicidad de infraestructura e información y el beneficio de la gestión integrada de la información administrativa, clínica y de salud [1].

Las aplicaciones que aborda este proyecto son el sistema de información hospitalario (HIS) y la historia clínica electrónica (HCE). Desde la perspectiva del paciente un proceso de atención en salud, como la consulta médica en un hospital, involucra la creación de datos e información de salud. Los síntomas, observaciones o diagnósticos, serán registrados en la historia clínica del paciente (HCE). La información de cada servicio es integrada por un HIS, es un sistema integrado de información diseñado para gestionar todos los aspectos clínicos, administrativos y financieros de un hospital. Además, permite obtener estadísticas generales de pacientes, datos epidemiológicos, de salud laboral y salud pública, entre otros. [5]

En el camino a la digitalización es necesario contar con sistemas de información de calidad y que tengan la característica de poder interoperar. En el contexto de salud en Chile, existen iniciativas de proveedores privados que implican una inversión de recursos significativa. A pesar de la variedad de proyectos, se observa una escasa definición técnica y de estándares a nivel nacional, además de una falta de regulación por la ausencia de políticas específicas articuladas con una estrategia nacional.

Afortunadamente en la actualidad existen software denominados de código abierto que son gratuitos y poseen alta confiabilidad. El software de código abierto es un código fuente que cualquiera puede inspeccionar, modificar y mejorar. El código fuente se puede manipular para cambiar el funcionamiento de una aplicación o para agregar nuevas características [6]. OpenMRS es a la vez un software y una comunidad. Como software, sirve como sistema de historia clínica electrónica (HCE) diseñado

originalmente para los países en desarrollo. Gracias a su comunidad de código abierto, se ha convertido en una plataforma informática médica utilizada en todos los continentes [7]

La motivación de este proyecto a largo plazo es implementar un sistema de información hospitalaria de código abierto, con el fin de habilitar una herramienta de acceso gratuito y asequible para organizaciones de salud que posean recursos limitados para invertir en sistemas de información. En esta etapa inicial del proyecto el objetivo alcanzable es implementar el HIS en una maquina local y configurar para ajustar las funciones a la realidad chilena, incluyendo el registro de paciente, la creación de la ficha clínica respectiva y el registro de diagnóstico en el módulo clínico.

2. METODOLOGÍA E IMPLEMENTACIÓN

La planificación del proyecto comprende cinco etapas que se desarrollan consecutivamente. La primera consiste en realizar un estudio de la documentación del software y programar las tareas que se deben realizar para completar la implementación. La segunda etapa es el diseño y levantamiento de requerimientos funcionales. A continuación, se completará la instalación del software, para seguir con la siguiente etapa de configuración. Por último, se deben realizar pruebas funcionales.

Etapa 1: Estudiar el sistema de información hospitalario

Bahmni es un sistema de información hospitalaria (HIS) y de historia clínica electrónica (EMR) de código abierto, en continuo desarrollo desde el 2012. El objetivo de Bahmni es satisfacer las necesidades de los hospitales en contextos de bajos recursos. Bahmni cuenta con una gran comunidad que mantiene actualizada la información disponible. La documentación es muy completa para desarrolladores, implementadores y usuarios. También cuenta con un curso corto sobre como iniciar en bahmni que incluye videos explicativos en inglés. La documentación oficial se puede encontrar en <https://bahmni.atlassian.net/wiki/spaces/BAH/overview>.

Bahmni EMR es una aplicación para el cliente, construida en AngularJS y HTML, e interactúa con el back-end de OpenMRS usando servicios web REST [8]. El sistema se basa en una estructura de base de datos conceptual que no depende de los tipos de información médica que deben recopilarse ni de formularios concretos de recopilación de datos, por lo que puede personalizarse para distintos usos [9].

OpenMRS tiene una arquitectura modular, que permite añadir o eliminar fácilmente funciones especiales del sistema. Los módulos tienen acceso total al sistema y pueden modificar o mejorar su comportamiento. Bahmni aprovecha otros productos open source disponibles y ofrece los siguientes modulos: Registro de pacientes, Servicios clínicos, Laboratorio, Pacientes hospitalizados, Gestión de existencias, Informes, Facturación y Contabilidad. Se pueden conocer en detalle en <https://www.bahmni.org/feature-list>

Etapa 2: Levantamiento de requerimientos técnicos, funcionales y diseño

El programa Bahmni requiere ser instalado en una maquina con sistema operativo CentOS 7.6. Sin embargo, este proyecto se realizó en una maquina física con sistema operativo Windows 10 utilizando el programa VirtualBox que permite crear máquinas virtuales. Una máquina virtual es un entorno virtual que funciona como sistema informático virtual con su propia CPU, memoria, interfaz de red y almacenamiento. La máquina virtual es de tipo Linux, con sistema operativo CentOS 7.6. El tamaño del disco es de 20 GB y

el tamaño de memoria es de 2048 MB. La documentación de github y el anexo 1 describe la metodología con mayor detalle sobre las configuraciones en VirtualBox, las configuraciones del sistema operativo y como configurar el acceso a internet.

El terminal que ofrece VirtualBox es poco amigable y dificulta el trabajo, para mejorar este aspecto se instaló el cliente SSH 'Putty'. Un cliente SSH es una aplicación en el computador que se usa para ejecutar comandos de terminal. Para poder utilizarlo se realizó una redirección de puertos, utilizando el puerto 22. Otras reglas de redirección de puertos fueron incluidas para que los servicios de Bahmni sean desplegados desde el navegador web del computador. La metodología detallada de cómo realizar la redirección de puertos se encuentra en la documentación de github y en el Anexo 1.

Para este caso de implementación, se definieron los siguientes aspectos de requerimientos y diseño: contar con tres usuarios con roles diferentes, el módulo de registro de paciente debe ser habilitado para el contexto chileno, habilitar parcialmente el módulo clínico incluyendo el registro de diagnósticos, y por último documentar toda la metodología utilizada en github.

Etapa 3: Instalar Bahmni

La instalación del software se realizó siguiendo las instrucciones detalladas en el anexo 2 y que se pueden encontrar en el github. En la consola se deben correr las líneas una a una. Es importante recalcar que las versiones de las paqueterías deben ser las señaladas. Es recomendable utilizar el comando 'sudo' en cada línea a ejecutar. Comenzando, se instalaron los pre-requisitos. Lo siguiente fue instalar pip de python. Con esto listo, el tercer paso fue instalar la paquetería utilizando pip.

Luego, se instaló el rpm de bahmni desde el repositorio, un archivo RPM es un elemento propio de Red Hat cuya finalidad es la de guardar paquetes de instalación dentro de Linux. Con estos archivos rpm será mucho más fácil gestionar el software ya que los tenemos encapsulados en un único sitio. Después de lo anterior, se configuró el setup.yml en el directorio /etc/bahmni-installer para configurar la zona horaria e ingresar el ip que referencia el software en el puerto 8069, el contenido del archivo se encuentra en el Anexo 2. Los pasos finales corresponden a cargar las variables de entorno, correr el instalador y crear un respaldo total del software. A continuación, se detallan algunos comandos útiles para conocer y modificar el estado del software instalado.

```
# conocer el status del programa
systemctl status openmrs
# detener el programa
systemctl stop openmrs
# iniciar el programa
systemctl start openmrs
```

Para acceder al programa, se debe iniciar sesión con un usuario y clave. Por defecto existe el usuario "superman" que posee todos los permisos para editar.

Tabla 1: Usuario y clave configurado por defecto

Usuario	superman
Clave	Admin123

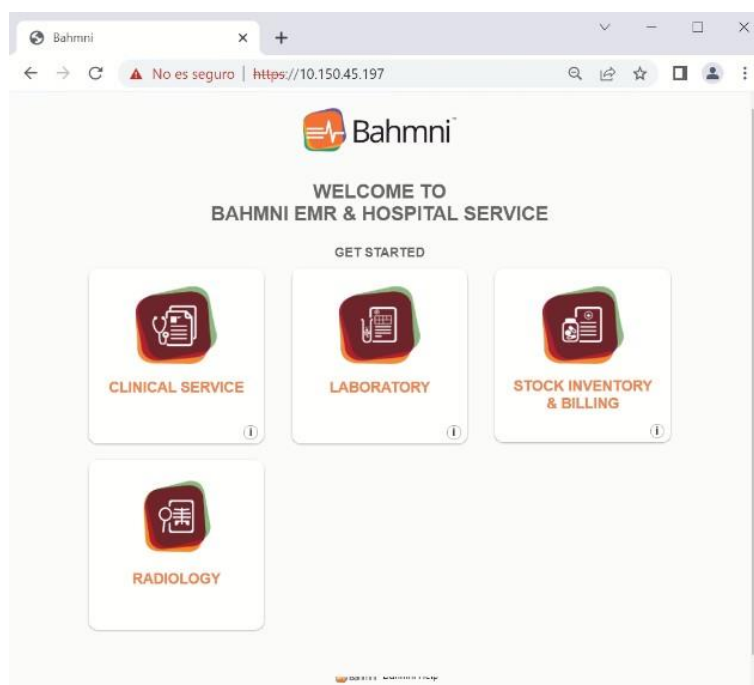


Ilustración 1: Página de inicio de Bahmni

Etapa 4: Configuración de Bahmni

Cuando Bahmni se ha instalado en la máquina, los módulos pueden ser utilizados con las configuraciones por defecto. El objetivo de esta etapa es configurar el software para que sea utilizado en las organizaciones de salud en Chile. Las tareas principales son; cargar el diccionario de conceptos, habilitar las traducciones al español de los módulos que serán utilizados, configurar la información demográfica que corresponde al paciente chileno y crear los usuarios del programa.

A continuación, se detallan las configuraciones que fueron realizadas en esta etapa del proyecto. Parte de las configuraciones se realizaron desde la pantalla de administración de OpenMRS. Otro método de configuración fue modificar directamente archivos que se encuentran en directorios específicos. Para la modificación de estos archivos se utilizó el terminal y el comando 'nano' de linux.

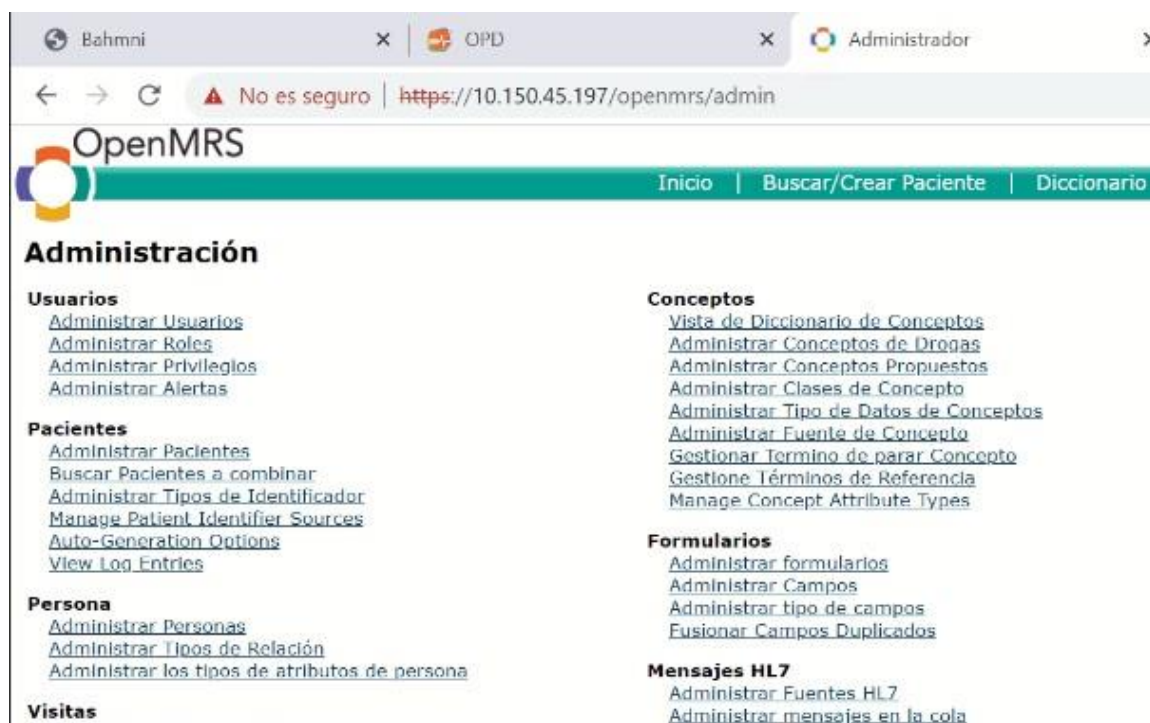


Ilustración 2: Interfaz de configuración de OpenMRS

4.1 Cargar el diccionario de conceptos

El diccionario de conceptos contiene todos los términos con los que trabaja el software. Los términos son variados, algunos ejemplos corresponden a enfermedades, anatomía, diagnósticos, síntomas, fármacos, exámenes de laboratorio, exámenes de radiología, parámetros vitales, características demográficas, etc. La configuración se realizó desde la página de administración de OpenMRS, en la interfaz 'Search index', hacer click en 'Rebuild Search Index'.

4.2 Internacionalización

La internacionalización permite que la aplicación esté disponible en varios idiomas. La internacionalización también se conoce como i18n porque hay 18 caracteres entre la 'i' y la 'n' en la palabra 'internacionalización'.

Cada módulo tiene archivos JSON que soportan las diferentes configuraciones locales. El archivo 'locale_es.json' soporta el idioma español y se encuentra en la carpeta llamada 'i18n'. La estructura del directorio refleja los distintos módulos de la aplicación. Los archivos JSON contienen un objeto JSON formado por pares clave-valor. Todas las claves se asignan internamente al texto que se muestra en la página web del módulo al que pertenece el archivo JSON.

Los pasos para configurar la internacionalización en el módulo de registro y el módulo clínico están contenidos en la documentación en github. Esta configuración permite habilitar el registro de diagnósticos. El contenido del archivo JSON de registro y clínico se encuentran en el anexo 3 y 4 respectivamente.

4.3 Crear usuarios

La creación de un nuevo usuario implica asignar privilegios, estos definen lo que se puede o no hacer en el sistema (ver información, editar información, añadir información, etc.). Los roles agrupan los privilegios en conjuntos manejables. Esta configuración se realizó desde la interfaz de administración de OpenMRS y el paso a paso se encuentra detallado en la documentación del Github. Se crearon tres usuarios diferentes utilizando los datos de la tabla.

Nombre (real) de usuario	Apellido usuario	Nombre de usuario	Clave	Roles		
user	uno	unouser	ASdf1234	Registration-App-Read-Only	Provider	
user	dos	dosuser	ASdf1234	Clinical App-Diagnosis	Provider	Registration-App
user	tre	treuser	ASdf1234	Clinical-App-Read-Only	Provider	Registration-App

Tabla 2: Características de los usuarios creados en bahmni

4.4 Configurar variables globales

Las variables globales definen las reglas que siguen ciertas variables. Una de las variables modificadas permite desplegar la lista de géneros con las palabras en español, y la otra permite registrar nombre con tildes y la letra ñ. Esta configuración se realizó desde la interfaz de administración de OpenMRS



Ilustración 3: Modificación de variables globales para el despliegue de generos

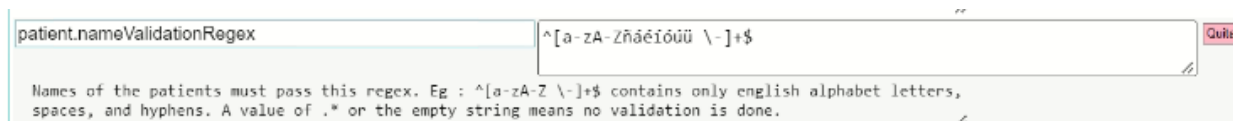


Ilustración 4: Modificación de las variables globales para validación del nombre del paciente

4.5 Configurar atributos de paciente en el registro de pacientes

El módulo de registro de pacientes permite buscar los pacientes ya ingresados en el sistema y crear un nuevo paciente. Las configuraciones por defecto incluyen datos demográficos de los pacientes que no se ajustan al perfil de paciente que trabaja Chile, como se puede ver en las ilustraciones 5 y 6.

The screenshot shows a web browser window with two tabs: 'Bahmni' and 'Registro de Pacientes'. The address bar shows a local IP address: 'https://10.150.45.197/bahm...'. The page header is dark green and contains a home icon, a search icon, the text 'Buscar', a plus icon, and the text 'Crear nueva'. Below the header, there are three search sections:

- ID:** A text input field with the placeholder 'Introduzca la ID' and a 'Buscar' button.
- Nombre:** A text input field with the placeholder 'Ingrese su nombre' and a 'Buscar' button.
- रुग्णाचे नाव / Rural Ward:** Two text input fields. The first has the placeholder 'रुग्णाचे नाव' and the second has 'Enter ward'. A 'Buscar' button is located below these fields.

Ilustración 5: Configuración por defecto de la interfaz de búsqueda de pacientes

The screenshot shows a form for creating a new patient, divided into three sections:

- Información de dirección:**
 - CASA SIN., CALLE: Text input field.
 - Village *: Text input field.
 - District: Text input field.
 - Gram Panchayat: Text input field.
 - Tehsill: Text input field.
 - Expresar: Text input field.
- Identificador adicional:**
 - Identificación nacional: Dropdown menu with 'NAT' selected.
 - Introduzca la ID:
- Otra información:**
 - Casta: Text input field.
 - Clase: Dropdown menu with 'General' selected.
 - Detalles de educación: Dropdown menu.
 - Contacto secundario: Text input field.
 - Nombre del padre / esposo: Text input field.
 - Identificador secundaria: Text input field.
 - Tenencia de la tierra: Text input field with '2' entered.

Ilustración 6: Configuración por defecto de la interfaz de crear nuevo paciente

Las características demográficas son datos que completan la descripción de una persona, como la edad, sexo, donde vive, etc. Los datos demográficos que incluye una ficha clínica de paciente son diferentes para cada país. En esta implementación utilizamos el perfil de paciente nacional, creado por La Organización HL7 Chile [<https://hl7chile.cl/fhir/ig/CoreCL/StructureDefinition-CorePacienteCl.html>] que describe las características que deben ser incluidas. La configuración de estos atributos se realiza en la pantalla de administración de OpenMRS.

Por otro lado, se modificó la aplicación para que los campos de la interfaz de registro contengan la información adecuada. Esto se logró editando el contenido del archivo 'app.json' desde el terminal, ingresando en el directorio /etc/bahmni-web/bahmni_config/openmrs/apps/registration. Este archivo contiene toda la configuración a nivel de aplicación para el módulo de registro. El contenido de 'app.json' se encuentra en la documentación en github y en el anexo 5.

4.5 Configurar jerarquía de direcciones en el registro de nuevos pacientes

Esta configuración permite que, al escribir en el apartado de comuna, los campos de provincia y región se autocompletan. Para esto fue necesario cargar en bahmni un archivo CSV que contiene la información ordenada en forma jerarquizada (Región, Provincia, Comuna) El archivo CSV se encuentra en la documentación de github.

3. RESULTADOS Y DISCUSIÓN

Las configuraciones realizadas permitieron que el sistema de información hospitalario y la ficha clínica electrónica de paciente que ofrece Bahmni fueran customizados, ofreciendo un producto ajustado a la realidad nacional. La metodología permite instalar estas aplicaciones de manera local utilizando una cantidad de recursos limitados.

El HIS funciona para tres usuarios nuevos con privilegios diferentes. En la página de inicio que se despliega para 'unouser' unicamente aparece el módulo de registro. El usuario 'unouser' puede buscar pacientes y ver su información, pero no es posible modificarla ni crear nuevos pacientes.

Los usuarios 'dosuser' y 'treuser' tienen acceso completo al módulo de registro, lo que les permite crear un nuevo paciente. También tienen acceso al módulo de clínica, sin embargo, el usuario dos puede modificar diagnósticos, mientras que el usuario tres no tiene permisos para editar en este módulo. Esto se evidencia en las siguientes imágenes.

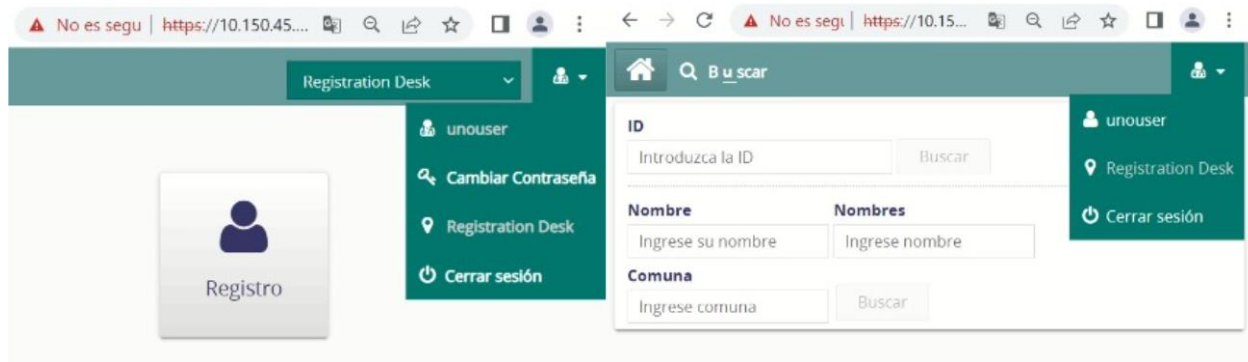


Ilustración 7: Pantallas al iniciar sesión con el usuario 'unouser'; a) Pantalla principal; b) Pantalla de registro de paciente

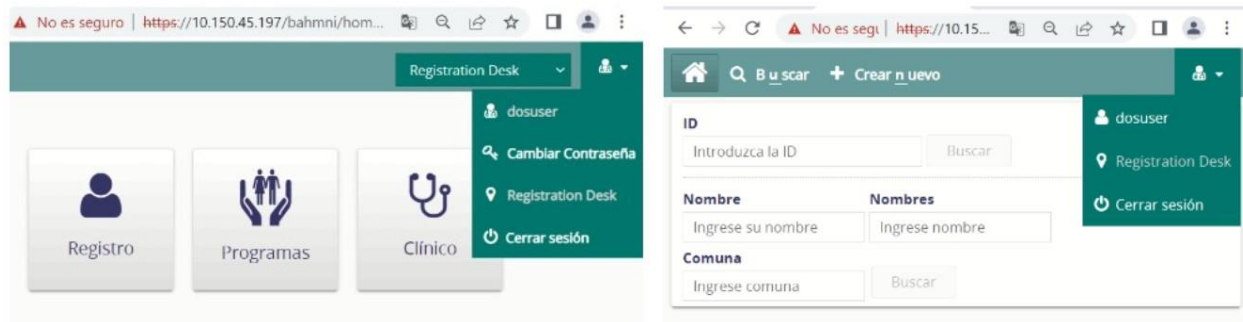


Ilustración 8: Pantallas al iniciar sesión con el usuario 'dosuser' y 'treuser'; a) Pantalla principal; b) Pantalla de registro de paciente

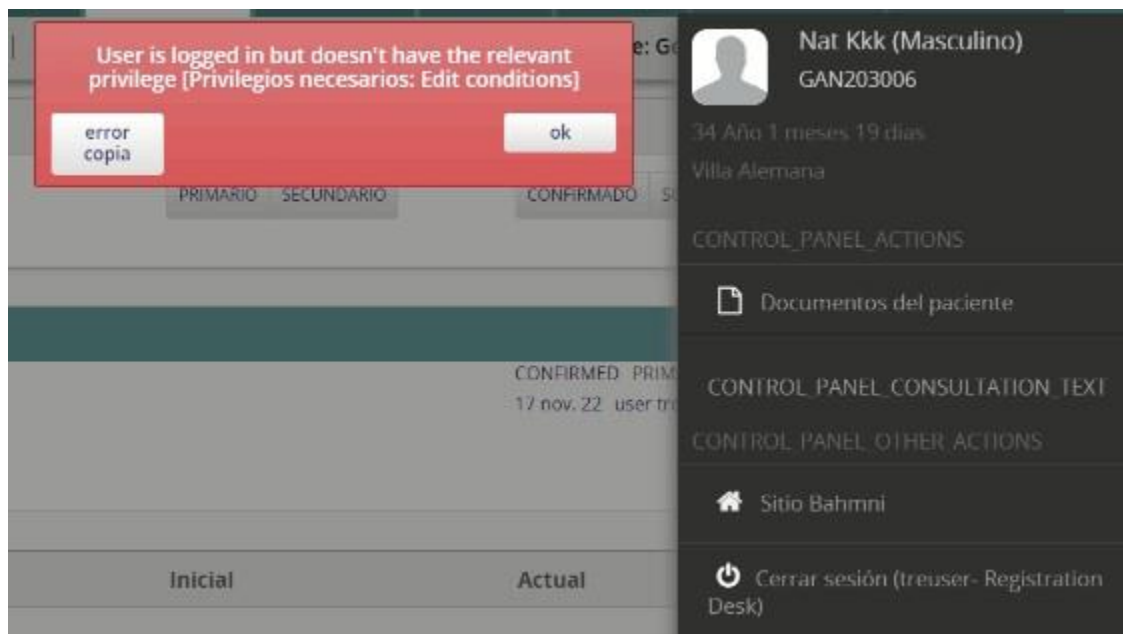


Ilustración 9: El usuario 'treuser' no posee los privilegios para guardar cambios en el módulo clínico. En la esquina inferior derecha aparece el nombre de usuario.

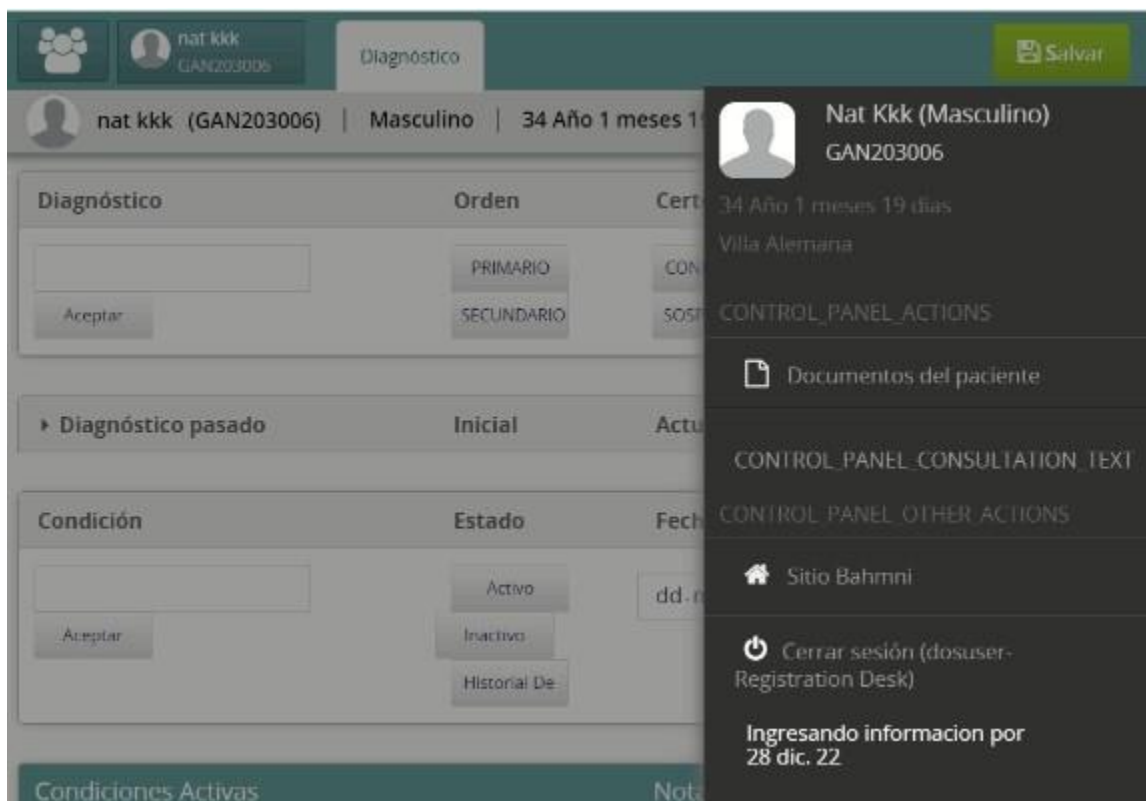


Ilustración 10: El usuario 'dosuser' tiene acceso solo a la pantalla de diagnóstico del módulo clínico. En la esquina inferior derecha aparece el nombre de usuario.

El módulo de registro de paciente fue exitosamente habilitado. Es posible buscar a los pacientes ya registrados, la pantalla para crear nuevo paciente está en español y los campos solicitados son los que corresponden al paciente chileno como se muestra en las siguientes imágenes.

ID	National ID	Nombre	Given Name Local	Middle Name Local	Family Name Local	Género	Edad	Fecha de Registro
GAN203006	NAT2804	nat ali kkk	nat	ali	kkk	M	34	08 nov. 22
GAN203007	NAT2805	ñaña caña maña				F	33	08 nov. 22
GAN203008	NAT2806	ñaña nnnn kkk	nata	mania	kkk	F	3	08 nov. 22
GAN203010	NAT2808	nata cvcgh				M	3	11 nov. 22

Ilustración 11: Pantalla de búsqueda de pacientes con algunos pacientes desplegados

Ilustración 12: Primera parte de la pantalla de registro de nuevo paciente final

The screenshot displays a patient registration form with three main sections:

- Información de dirección:** Includes text input fields for 'Dirección 1 (calle, número)', 'Comuna', 'Provincia', and 'Región'.
- Identificador adicional:** Features a dropdown menu for 'Identificación nacional' (set to 'NAT') and a checkbox labeled 'Introduzca la ID'.
- Otra información:** Contains dropdown menus for 'Nivel educacional' and 'Ocupación', and text input fields for 'Nombre de contacto principal', 'Nombre de contacto secundario', 'Apellido Materno', 'Contacto principal', 'Contacto secundario', 'Identificador secundaria', and 'Lenguaje que utiliza el paciente para comunicarse'.

Ilustración 13: Segunda parte de la pantalla de registro de paciente final

El registro de diagnósticos en el módulo clínico funciona correctamente, permite ingresar diagnósticos y condiciones en forma de texto, se puede seleccionar si este diagnóstico es primario o secundario, confirmado o en sospecha y si corresponde a un diagnóstico activo o inactivo. Además, permite visualizar un historial de diagnósticos pasados detallado. La ilustración 14 presenta la interfaz del registro de diagnósticos.

The screenshot shows a web browser window with the URL <https://10.150.45.197/bahmni/>. The interface includes a navigation bar with tabs for Observaciones, Diagnóstico, Disposición, Consulta, Órdenes, Medicamentos, and Bacteriología. The user is logged in as 'nat kkk' with ID 'GAN203006'. The main area is divided into two sections:

Diagnóstico Form:

Diagnóstico	Orden	Certeza	Estado
<input type="text"/>	<input type="button" value="PRIMARIO"/> <input type="button" value="SECUNDARIO"/>	<input type="button" value="CONFIRMADO"/> <input type="button" value="SOSPECHADO"/>	<input type="button" value="Inactive"/>

Diagnóstico pasado Table:

Diagnóstico pasado	Inicial	Actual	Acciones
hepatitis		CONFIRMED PRIMARY 17 nov. 22 user tre	Agregar como condición <input type="button" value="✎"/> <input type="button" value="✖"/>
diabetes	CONFIRMED PRIMARY 08 nov. 22 Super Man	CONFIRMED PRIMARY Inactive 15 nov. 22 Super Man	Agregar como condición <input type="button" value="✎"/> <input type="button" value="✖"/>
hepatitis		CONFIRMED PRIMARY 15 nov. 22 Super Man	Agregar como condición <input type="button" value="✎"/> <input type="button" value="✖"/>

Ilustración 14: Registro de diagnósticos en el módulo clínico

La documentación se encuentra en el siguiente enlace de github <https://github.com/natijam/bahmni-proyecto-titulo>

Los módulos seleccionados para esta implementación son básicos y estarán presentes transversalmente en las organizaciones de salud chilenas. Sin embargo, para considerar otros servicios y procesos que se ajusten a la realidad nacional, es necesario realizar un análisis sistemático o levantamiento de los procesos básicos y sus alcances. De igual manera, conocer los procesos va a permitir un análisis de las brechas que presenta Bahmni para lograr cumplir con los requerimientos levantados y las necesidades específicas para cada implementación.

4. CONCLUSIÓN

El sistema de información hospitalario Bahmni ofrece una oportunidad para hacer frente a los desafíos actuales de la digitalización en salud en Chile. Bahmni es un sistema informático efectivo para la integración de la gestión de los diferentes servicios del hospital y de la información de salud y hospitalaria. Permite la creación, almacenamiento y actualización de los registros médicos electrónicos de

los pacientes.

El proyecto entrega la metodología para ajustar los módulos básicos de un sistema informático hospitalario gratuito, a la realidad de la salud chilena. Un posible siguiente paso para completar el objetivo a largo plazo del proyecto corresponde a la instalación del software en un servidor, de manera que otros computadores de la organización puedan acceder a la aplicación. Trabajos futuros pueden comprender la habilitación de reportes y el agendamiento. Los reportes son los informes que se pueden desprender del análisis de la información almacenada en el sistema respecto a la utilización de recursos, la estadística de los pacientes, la ocupación de algunos servicios, entre otros, es un módulo que apoya fuertemente la toma de decisiones y está incluido en las funciones que ofrece Bahmni. De la misma forma, contar con el módulo de agendamiento de citas, es una importante herramienta de gestión que puede mejorar la calidad y la eficiencia del servicio.

5. ANEXOS

Anexo 1: Instalar Virtual Box, configurar CentOS y configurar el acceso a internet

Requisitos técnicos

Vamos a crear una máquina virtual CentOS v7.6 utilizando VirtualBox.

- [VirtualBox](www.virtualbox.org) para Windows Host
- [CentOS-7-x84_64-DVD-1810.iso](http://ftp.ijj.ad.jp/pub/linux/centos-vault/7.6.1810/isos/x86_64/CentOS-7-x86_64-DVD-1810.iso)

Configuraciones en VirtualBox

En VirtualBox seleccionar “Nueva”

General

- Básico
- Nombre : `CentOS 7.6`
- Tipo : `Linux`
- Version : `Red Hat (64-bit`

Sistema

- Tamaño de memoria: `2048 MB`

1. Disco duro: Seleccionar "crear un disco duro virtual ahora". Next.
2. Tipo de archivo de disco duro: VDI (VirtualBox Disk Image). Next.
3. Almacenamiento en unidad de disco duro física: Reservado dinámicamente. Next
4. Ubicación del archivo y tamaño: modificar el tamaño del disco a 20 GB. Crear.
5. La máquina está creada en VirtualBox

Almacenamiento

Seleccionar “Vacío” debajo de “Controlador: IDE”, seleccionar el ícono de disco a la derecha de la ventana. Click en “Seleccionar archivo de disco óptico virtual”.

- Controlador : IDE
- `CentOS-7-x84_64-DVD-1810.iso`

Configuraciones de instalación de CentOS

Al Iniciar la máquina se abrirá el asistente de instalación de CentOS

- Idioma: `Español. Chile`
- En la sección “Sistema”, seleccionar “Destino de la instalación”.
- Aparecerá seleccionado el disco duro que creamos. Click en `Listo`.

- Empezar instalación.

Durante la instalación

- Crear clave de administrador "root".
- Crear usuario y clave.
- [] Hacer administrador al usuario

Finalizar instalación

- `Reiniciar`
- License information
- Aceptar el acuerdo de la licencia

Configuraciones de sistema CentOS

Configuración de Red

En el terminal

...

para comprobar si hay conexión a internet utilizar el comando `nmcli` en el terminal
nmcli

acceder al archivo de configuración de redes sólo visualización

cat /etc/sysconfig/network-scripts/ifcfg-enp0s3

el comando vi permite editar el archivo

sudo vi /etc/sysconfig/network-scripts/ifcfg-enp0s3

comando para insertar caracteres en el archivo

i

Cambiar `ONBOOT:no` a `ONBOOT:yes`

Presionar la tecla "esc"

el comando :wq guarda los cambios realizados en el archivo

:wq

sudo systemctl restart network.service

nmcli

ahora debe mostrar enp0s3: conectado to enp0s3

para conocer el ip de la máquina virtual

ip a

...

Configuraciones en VirtualBox

* Herramientas

* Preferencias

* Detalles de la Red NAT

- `Reenvío de puertos`

Reglas de reenvío de puertos

IPv4 IPv6

Nombre	Protocolo	IP anfitrión	Puerto anfitrión	IP invitado	Puerto invitado
Rule 1	TCP	10.150.45.197	23	10.0.2.4	22
Rule 10	TCP	10.150.45.197	3306	10.0.2.4	3306
Rule 11	TCP	10.150.45.197	5432	10.0.2.4	5432
Rule 12	TCP	10.150.45.197	80	10.0.2.4	80
Rule 13	TCP	10.150.45.197	11112	10.0.2.4	11112
Rule 14	TCP	10.150.45.197	443	10.0.2.4	443
Rule 2	TCP	10.150.45.197	8050	10.0.2.4	8050
Rule 3	TCP	10.150.45.197	8051	10.0.2.4	8051
Rule 4	TCP	10.150.45.197	8052	10.0.2.4	8052
Rule 5	TCP	10.150.45.197	8053	10.0.2.4	8053
Rule 6	TCP	10.150.45.197	8054	10.0.2.4	8054
Rule 7	TCP	10.150.45.197	8069	10.0.2.4	8069
Rule 8	TCP	10.150.45.197	8055	10.0.2.4	8056
Rule 9	TCP	10.150.45.197	8057	10.0.2.4	8057

Aceptar Cancelar

Anexo 2: Instalar Bahmni

#prerequisitos.

```
sudo yum install -y
https://kojipkgs.fedoraproject.org//packages/zlib/1.2.11/19.fc30/x86_64/zlib-1.2.11-
19.fc30.x86_64.rpm
sudo yum install -y epel-release
#si no funciona, usar : sudo yum install https://dl.fedoraproject.org/pub/epel/epel-
release-latest-7.noarch.rpm
```

#instalar pip

```
sudo yum install python-pip
sudo pip install pip==v19.0

sudo pip install --upgrade pip
sudo pip install babel==v1.0 python-stdnum urllib3==1.21.1 idna==2.5 chardet==3.0.2
certifi==2017.4.17 qrcode pyserial pypdf python-chart psycogreen passlib ofxparse
requests
```

#instalar paqueteria por pip

```
sudo pip install babel==v1.0 python-stdnum urllib3==1.21.1 idna==2.5 chardet==3.0.2
certifi==2017.4.17 qrcode pyserial pypdf python-chart psycogreen passlib ofxparse
requests
```

#En caso que falten paquetes, pueden ser algunos de estos.

```
sudo uninstall click
sudo pip install click==v7.0,
sudo pip install pyusb1
sudo pip install decorator==v3.4.0
sudo pip install beautifulsoup4
sudo yum install http://repo.mybahmni.org/releases/ansible-2.4.6.0-
1.el7.ans.noarch.rpm
```

#instalar rpm de bahmni desde repositorio.

```
sudo yum install -y https://repo.mybahmni.org/releases/bahmni-installer-0.93-
219.noarch.rpm
```

#instalar postgres manual por RPM

```
rm -f /opt/pgdg-redhat-repo-*
cd /opt && wget https://repo.mybahmni.org/releases/pgdg-redhat-repo-42.0-
23.noarch.rpm
```

#Configurar Setup.yml

```
sudo nano /etc/bahmni-installer/setup.yml
```

#este es el contenido de setup.yml.

To see the list of valid variables in Bahmni please refer to:

```
#
https://bahmni.atlassian.net/wiki/display/BAH/List+Of+Configurable+Installation+Variables

timezone: America/Santiago
implementation_name: default
selinux_state: disabled
mysql_root_password: poner clave
mysql_old_root_password: poner clave
openerp_url: http://poner ip :8069
# bahmni_repo_url: http://repo.mybahmni.org.s3-website-ap-southeast-1.amazonaws.com/rpm/bahmni/
bahmni_repo_url: https://repo.mybahmni.org/releases/
#Configurar Setup.yml y local segun necesidad

sudo nano /etc/bahmni-installer/local
#este es el contenido de local para instalacion en misma maquina full

localhost ansible_connection=local

[nagios-server]

[bahmni-emr]
localhost

[bahmni-emr-db]
localhost

[bahmni-emr-db-slave]

[bahmni-erp]
localhost

[bahmni-erp-db]
localhost

[bahmni-erp-db-slave]

[bahmni-lab]
localhost

[bahmni-lab-db]
localhost

[bahmni-lab-db-slave]

[bahmni-reports]
localhost

[bahmni-reports-db]
localhost
```

[bahmni-reports-db-slave]

[atomfeed-console]
localhost

[pacs-integration]
localhost

[pacs-integration-db]
localhost

[pacs-integration-db-slave]

[dcm4chee]
localhost

[dcm4chee-db]
localhost

[dcm4chee-db-slave]

[bahmni-event-log-service]
localhost

[bahmni-offline]

[mysql-backup-tool]
localhost

[postgres-backup-tool]
localhost

[bahmni-backup-artifacts]
localhost

[local:children]
nagios-server
bahmni-emr
bahmni-emr-db
bahmni-emr-db-slave
bahmni-lab
bahmni-lab-db
bahmni-lab-db-slave
bahmni-erp
bahmni-erp-db
bahmni-erp-db-slave
bahmni-reports
bahmni-reports-db
bahmni-reports-db-slave
pacs-integration
pacs-integration-db
pacs-integration-db-slave
dcm4chee
dcm4chee-db

```
dcm4chee-db-slave
bahmni-event-log-service
bahmni-offline
atomfeed-console
mysql-backup-tool
postgres-backup-tool
bahmni-backup-artifacts
#cargar variables de entorno

echo "export BAHMNI_INVENTORY=local" >> ~/.bashrc
source ~/.bashrc
#correr instalador

sudo bahmni -i local install
#backup de bahmni. total

bahmni -i local backup --backup_type=all --options=all
#restaurar bahmni

#archivos quedan en /data/ y en /home/bahmni que deben ponerse a mano en maquina
destino.

bahmni -i local restore --restore_type=db --options=openmrs --strategy=dump --
restore_point=openmrs_dump_20170221084218.sql.gz
#notificaciones bahmni email

sudo curl -L https://github.com/Lopior/Instalador/edit/main/email-
notification.properties >> /opt/openmrs/email-notification.properties
```

Anexo 3: Contenido del archivo para la internacionalización del módulo registro.

```
{
  "REGISTRATION_SEARCH_CONFIG_KEY": "B <u> u </u> scar",
  "REGISTRATION_CREATE_NEW_CONFIG_KEY": "Crear <u> n </u> uevo",
  "REGISTRATION_SEARCH_CONFIG_ACCESS_KEY": "mi",
  "REGISTRATION_CREATE_NEW_CONFIG_ACCESS_KEY": "norte",
  "REGISTRATION_VIEW_CONFIG_KEY": "Vista",
  "REGISTRATION_PRINT_REG_CARD_LOCAL_KEY" : "Imprimir tarjeta reg. (<u> l </u> ocal)",
  "REGISTRATION_PRINT_REG_CARD_KEY": "Tarjeta de registro impresa <U> P </u> ",
  "REGISTRATION_PRINT_SUPPLEMENTAL_PAPER": "Im <u> p </u> rimir suplementario de
  papel",
  "REGISTRATION_AGE_ERROR_KEY": "La edad debería ser un valor positivo",
  "REGISTRATION_TELEPHONE_NUMBER_ERROR_KEY": "El número de teléfono debe tener un
  mínimo de 6 dígitos",
  "REGISTRATION_CASTE_TEXT_ERROR_KEY": "La casta debe contener solo caracteres
  alfanuméricos",
  "REGISTRATION_BRING_CARD_TEXT_KEY": "Es bastante importante llevar la tarjeta al
  hospital",
  "REGISTRATION_TAKE_MEDICINES_TEXT_KEY": "Tome medicamentos de acuerdo con las
  instrucciones del médico",
  "REGISTRATION_OPD_DAYS_TEXT_KEY" : "Los días de OPD son los lunes, miércoles,
  viernes",
  "REGISTRATION_SUNDAYS_HOLIDAY_TEXT_KEY": "Los domingos son vacaciones",
  "REGISTRATION_SMOKING_INJURIOUS_TEXT_KEY" : "Fumar, la masticación del tabaco puede
  causar cáncer, enfermedades cardíacas y asma",
  "REGISTRATION_INSTITUTE_NAME_TEXT_KEY": "Centro Medico Y dental Fundación",
  "REGISTRATION_INSTITUTE_REGISTERED_TEXT_KEY": "(Registrado)",
  "REGISTRATION_PRINT_WITH_BARCODE": "Imprimir <u> c </u> odigo de barra",
  "REGISTRATION_NAME_IN_LOCAL_LANGUAGE": "Nombre Social",
  "REGISTRATION_MIDDLENAMELOCAL": "Segundo Nombre",
  "REGISTRATION_NATIONAL_ID": "Identificación nacional",
  "REGISTRATION_HOUSE_NO_STREET": "Direccion",
  "REGISTRATION_GRAM_PANCHAYAT": "Direccion 2",
  "REGISTRATION_VILLAGE": "Comuna",
  "REGISTRATION_TEHSIL": "Ciudad",
  "REGISTRATION_DISTRICT": "Provincia",
  "REGISTRATION_STATE": "Región",
  "PATIENT_ATTRIBUTE_CASTE": "Casta",
  "PATIENT_ATTRIBUTE_CLASS": "Previsión de salud",
  "PATIENT_ATTRIBUTE_EDUCATION_DETAILS": "Nivel educacional",
  "PATIENT_ATTRIBUTE_OCCUPATION": "Ocupación",
  "PATIENT_ATTRIBUTE_SECONDARY_CONTACT": "Contacto secundario",
}
```

```
"PATIENT_ATTRIBUTE_FATHERSHUSBANDS_NAME": "Apellido Materno",
"PATIENT_ATTRIBUTE_PRIMARY_RELATIVE": "Familiar primario",
"PATIENT_ATTRIBUTE_SECONDARY_IDENTIFIER": "Identificador secundaria",
"PATIENT_ATTRIBUTE_LAND_HOLDING_IN_ACRES": "Tenencia de la tierra",
"PATIENT_ATTRIBUTE_DEBT_IN_RS": "Deuda",
"PATIENT_ATTRIBUTE_DISTANCE_FROM_CENTER_IN_KM": "Distancia desde el centro",
"PATIENT_ATTRIBUTE_URBAN": "Es urbana",
"PATIENT_ATTRIBUTE_CLUSTER": "Racimo",
"PATIENT_ATTRIBUTE_RATION_CARD_TYPE": "Cartilla de racionamiento",
"PATIENT_ATTRIBUTE_FAMILY_INCOME_PER_MONTH_IN_RS": "Ingresos familiares",
"NUTRITIONAL_VALUES_LOCALE_KEY": "Valores nutricionales",
"FEE_INFORMATION_LOCALE_KEY": "Información de tarifas",
"PATIENT_ATTRIBUTE_EMAIL_ADDRESS": "Email"
}
```

Anexo 4: Contenido del archivo para la internacionalización del módulo clínico.

```
{
  "MODULE_LABEL_ACTIVE_KEY": "Activo",
  "MODULE_LABEL_PROGRAMS_KEY": "Programas",
  "MODULE_LABEL_MY_PATIENTS_KEY": "Mis Pacientes",
  "MODULE_LABEL_ACTIVE_PATIENTS_BY_LOCATION": "PacientesPorUbicacion",
  "MODULE_LABEL_OPD_1_KEY": "OPD-1",
  "MODULE_LABEL_ALL_KEY": "Todos",

  "DASHBOARD_TITLE_LAB_RESULTS_KEY": "Resultados de Laboratorio",
  "DASHBOARD_TITLE_LAB_ORDERS_DISPLAY_CONTROL_KEY": "Cumplimiento de órdenes de Laboratorio",
  "DASHBOARD_TITLE_PACS_KEY": "Pacs",
  "DASHBOARD_TITLE_RADIOLOGY_KEY": "Radiología",
  "DASHBOARD_TITLE_VISITS_KEY": "Visitas",
  "DASHBOARD_TITLE_PROGRAMS_KEY": "Programas",
  "DASHBOARD_TITLE_TREATMENTS_KEY": "Tratamientos",
  "DASHBOARD_TITLE_DISPOSITION_KEY": "Disposición",
  "DASHBOARD_TITLE_DIAGNOSIS_KEY": "Diagnóstico",
  "DASHBOARD_TITLE_ADMISSION_DETAILS_KEY": "Detalles de admisión",
  "DASHBOARD_TITLE_NUTRITIONAL_VALUES_KEY": "Valores Nutricionales",
  "DASHBOARD_TITLE_SECOND_VITALS_KEY": "Signos Vitales secundarios",
  "DASHBOARD_TITLE_GYNAECOLOGY_KEY": "Ginecología",
  "DASHBOARD_TITLE_HISTORY_AND_EXAMINATIONS_KEY": "Historial y exámenes",
  "DASHBOARD_TITLE_OBSTETRICS_KEY": "Obstetricia",
  "DASHBOARD_TITLE_NAVIGATION_LINKS_CONTROL_KEY": "Enlaces de controles de navegación",
  "DASHBOARD_TITLE_RADIOLOGY_ORDERS_KEY": "Ordenes de radiología",
  "DASHBOARD_TITLE_VITALS_KEY": "Signos vitales",
  "DASHBOARD_TITLE_PATIENT_INFORMATION_KEY": "Información del paciente",
  "DASHBOARD_TITLE_BACTERIOLOGY_RESULTS_KEY": "Resultados de Bacteriología",
  "DASHBOARD_TITLE_FORMS_DISPLAY_CONTROL_KEY": "Formularios",

  "DASHBOARD_TITLE_GROWTH_CHART": "Gráfico crecimiento",
  "DASHBOARD_TITLE_BLOOD_PRESSURE": "PA",
  "DASHBOARD_TITLE_WEIGHT_BMI": "Peso/IMC",
  "DASHBOARD_TITLE_DIABETES_BLOOD_SUGAR": "Diabetes/Glicemia",
  "DASHBOARD_TITLE_PULSE": "Pulso",
  "DASHBOARD_TITLE_FORMS2_DISPLAY_CONTROL_KEY": "Formulario-2",
  "DASHBOARD_TITLE_FORMS2_OBS_TO_OBS_DISPLAY_CONTROL_KEY": "Formulario-2 ObsToObs",

  "DASHBOARD_TAB_GENERAL_KEY": "General",
  "DASHBOARD_TAB_TRENDS_KEY": "Tendencias",
  "DASHBOARD_TAB_PATIENT_SUMMARY_KEY": "Resumen del paciente",
  "DASHBOARD_TAB_DISCHARGE_SUMMARY_KEY": "Resumen de alta",
  "DASHBOARD_TAB_ORDERS_KEY": "Órdenes",

  "LAB_INVESTIGATIONS_KEY": "Investigaciones de Laboratorio",
```

"PACS_ORDERS_VISIT_PAGE_KEY": "Visitar página de órdenes PACS",

"DISCHARGE_SUMMARY_DIAGNOSIS_KEY" : "Diagnóstico",
"DISCHARGE_SUMMARY_ADVICE_ON_DISCHARGE_KEY": "Consejos para el alta",
"DISCHARGE_SUMMARY_AT_ADMISSION_KEY": "En admisión",
"DISCHARGE_SUMMARY_ADMISSION_DETAILS_KEY": "Detalles de admisión",
"DISCHARGE_SUMMARY_CLINICAL_NOTES_KEY": "Notas Clínicas",
"DISCHARGE_SUMMARY_AT_DISCHARGE_KEY": "En Alta",
"DISCHARGE_SUMMARY_TREATMENTS_KEY": "Tratamientos",

"OBSERVATIONS_BOARD_LABEL_KEY": "Observaciones",
"MEDICATIONS_BOARD_LABEL_KEY": "Medicamentos",
"ORDERS_BOARD_LABEL_KEY": "Órdenes",
"BACTERIOLOGY_BOARD_LABEL_KEY": "Bacteriología",
"CONSULTATION_BOARD_LABEL_KEY": "Consulta",
"DISPOSITION_BOARD_LABEL_KEY": "Disposición",
"DIAGNOSIS_BOARD_LABEL_KEY": "Diagnóstico",

"CLINICAL_ORDER_RADIOLOGY_NEED_PRINT": "Necesita impresora para esta orden",
"CLINICAL_ORDER_RADIOLOGY_NEED_PRINT_BUTTON": "Necesita impresora",

"CONSULTATION_PAGE_KEY" : "Consulta",
"PROGRAM_MANAGEMENT_PAGE_KEY": "Página de administración del programa",
"DISCHARGE_SUMMARY_PAGE_KEY": "Página de resumen de Alta",
"PATIENT_ADT_PAGE_KEY": "Patient ADT Page",
"PATIENT_DASHBOARD_KEY": "Panel del paciente",
"PATIENT_VISIT_PAGE_KEY": "Página de visitas del paciente",
"HOME_DASHBOARD_KEY": "Panel de inicio",

"MEDICATION_CHANGE_DURATION_BUTTON": "Cambiar Duración",
"DASHBOARD_TITLE_Consultation_KEY": "Notas de la consulta",
"DASHBOARD_TITLE_APPOINTMENTS_KEY": "Cita",
"DASHBOARD_APPOINTMENTS_SLOT_KEY": "Ranura",
"DASHBOARD_APPOINTMENTS_SERVICE_KEY": "Servicio",
"DASHBOARD_APPOINTMENTS_SERVICE_TYPE_KEY": "Tipo de cita de servicio",
"DASHBOARD_APPOINTMENTS_DATE_KEY": "Fecha",
"DASHBOARD_APPOINTMENTS_PROVIDER_KEY": "Proveedor",
"DASHBOARD_APPOINTMENTS_STATUS_KEY": "Estados",
"DASHBOARD_PAST_APPOINTMENTS_KEY": "Citas pasadas",
"DASHBOARD_UPCOMING_APPOINTMENTS_KEY": "Citas futuras",
"DASHBOARD_UPCOMING_APPOINTMENTS_KEY_LINK": "Vista de listado",
"DASHBOARD_NO_PAST_APPOINTMENTS_KEY": "Sin citas pasadas",
"DASHBOARD_NO_UPCOMING_APPOINTMENTS_KEY": "Sin citas futuras",

"DASHBOARD_TELECON_KEY" : "Teleconsulta",
"DASHBOARD_JOIN_TELECON_KEY" : "Iniciar teleconsulta",
"DASHBOARD_TITLE_FORMS2_DISPLAY_EDIT_FORMS2_FORMS": "Visualizar y editar Forms (2.0)",

```
"PROGRAM_HIV_PROGRAM": "Programa HIV ",  
"PROGRAM_TB_PROGRAM": "Programa TB",  
"PROGRAM_ID_NUMBER": "Numero ID",  
"PROGRAM_TREATMENT_DATE": "Fecha de Tratamiento",  
"PROGRAM_COMORBIDITES": "Co-morbilidades",  
"PROGRAM_DOCTORINCHARGE": "Doctor a cargo",  
"PROGRAM_PATIENT_STAGE": "Etapa de Paciente"  
}
```

Anexo 5: Contenido del app.json

```

{
  "id": "bahmni.registration",
  "instanceOf": "bahmni.template.registration",
  "description": "Bahmni Patient Registration App",
  "extensionPoints": [
    {
      "id": "org.bahmni.registration.patient.search.result.action",
      "description": "Bahmni Patient Search Result Action"
    },
    {
      "id": "org.bahmni.registration.patient.next",
      "description": "After Bahmni Patient Registration, forward url configuration"
    },
    {
      "id": "org.bahmni.registration.navigation",
      "description": "Navigation within registration first and second page"
    },
    {
      "id": "org.bahmni.registration.conceptSetGroup.observations",
      "description": "Registration Second Page Forms"
    }
  ],
  "contextModel": [
    "patientUuid",
    "activeVisitUuid"
  ],
  "config": {
    "patientInformation": {
      "additionalPatientInformation": {
        "title": "Additional Patient Information",
        "attributes": [
          "email"
        ]
      },
      "hidden": {
        "attributes": [
          "cluster",
          "RationCard",
          "familyIncome",
          "debt",
          "distanceFromCenter",
          "landHolding",
          "caste",
          "isUrban"
        ]
      }
    }
  },

```

```
"defaults": {
  "class": "General",
  "landHolding": 2
},
"addressHierarchy": {
  "showAddressFieldsTopDown": false,
  "strictAutocompleteFromLevel": "stateProvince"
},
"relationshipTypeMap": {
  "Doctor": "provider",
  "Parent": "patient"
},

"autoCompleteFields": ["familyName", "caste"],
"defaultIdentifierPrefix": "GAN",
"defaultVisitType": "OPD",
"searchByIdForwardUrl": "/patient/{{patientUuid}}",
"showMiddleName": true,
"showLastName": true,
"isLastNameMandatory": true,
"patientNameDisplayOrder": ["firstName", "middleName", "lastName"],
"showSaveConfirmDialog": false,
"showBirthTime": true,
"showCasteSameAsLastNameCheckbox": false,
"printOptions": [
  {
    "translationKey": "REGISTRATION_PRINT_REG_CARD_LOCAL_KEY",
    "templateUrl":
"/bahmni_config/openmrs/apps/registration/registrationCardLayout/print_local.html",
    "shortcutKey": "l"
  },
  {
    "translationKey": "REGISTRATION_PRINT_REG_CARD_KEY",
    "templateUrl":
"/bahmni_config/openmrs/apps/registration/registrationCardLayout/print.html",
    "shortcutKey": "p"
  },
  {
    "translationKey": "REGISTRATION_PRINT_SUPPLEMENTAL_PAPER",
    "templateUrl":
"/bahmni_config/openmrs/apps/registration/supplementalPaperLayout/print.html",
    "shortcutKey": "r"
  },
  {
    "translationKey": "REGISTRATION_PRINT_WITH_BARCODE",
```

```

        "templateUrl":
"/bahmni_config/openmrs/apps/registration/registrationCardLayout/printWithBarcode.html",
        "shortcutKey": "c"
    }
],
"conceptSetUI": {
    "REGISTRATION FEES": {
        "required": false,
        "label": "Fee"
    },
    "FHS": {
        "buttonSelect": true
    },
    "defaults":{
        "FHS": "Present"
    }
},
"fieldValidation" : {
    "email" : { "pattern" :
"^(?=.{1,40}$)(([^\<>()\\[\]\\\\\.\,;:\s@\\"]+)(\\.[^\<>()\\[\]\\\\\.\,;:\s@\\"]+)*)(\\.+\\")@((([a-zA-Z\\-0-9]+\\.)+[a-zA-Z]{2,4})$)", "errorMessage" : "Should be a valid email address"},
    "primaryContact" : { "pattern" : "[0-9]{8,10}", "errorMessage" : "Should be 12 characters starting with 91 (country code)"},
    "landHolding" : { "pattern" : "[0-9]{0,3}", "errorMessage" : "Should be between 1 to 999 acres"},
    "familyName" : { "pattern" : "[a-zA-Zñáéíóúü\\s]{0,}", "errorMessage" : "Should contain characters"},
    "givenName" : { "pattern" : "[a-zA-Zñáéíóúü\\s]{0,}", "errorMessage" : "Should contain characters"},
    "middleName" : { "pattern" : "[a-zA-Zñáéíóúü\\s]{0,}", "errorMessage" : "Should contain characters"},
    "address1" : { "pattern" : "[a-zA-Z0-9\\s]{2,}", "errorMessage" : "Should contain at least 2 characters"}
},
"patientSearch": {
    "address": {
        "label": "Comuna",
        "placeholder": "Ingrese comuna",
        "field": "address2"
    },
    "customAttributes": {
        "label": "Nombres",
        "placeholder": "Ingrese Nombre",
        "fields": ["givenNameLocal", "middleNameLocal", "familyNameLocal"]
    }
}
}
}
}

```

6. REFERENCIAS

- [1] CEPAL, «Newsletter 12: Salud y TIC,» 2010.
- [2] ehCOS, «Oportunidades y desafíos del sector salud en Chile,» 9 Septiembre 2022. [En línea]. Available: <https://www.ehcos.com/oportunidades-desafios-salud-chile/>.
- [3] Slim, «Salud Digital,» 2022. [En línea]. Available: <https://saluddigital.com/es/avance-de-la-ciencia/el-ministerio-de-salud-de-chile-minsal-anuncia-la-creacion-del-departamento-de-salud-digital/>.
- [4] M. G. L. Adamo, «"Historia Clínica Electrónica: Continuidad de la atención, seguridad, calidad y mejor gestión en salud",» 2016. [En línea]. Available: https://na.eventscloud.com/file_uploads/725cd4c92e038fce92de352981e26709_GabrielaLissi.pdf.
- [5] A. García, El sistema de información del hospital, CEPAL, 2012.
- [6] «¿Qué es el software de código abierto? | IBM,» [En línea]. Available: <https://www.ibm.com/cl-es/topics/open-source>.
- [7] B. Mamlin, «Introduction to OpenMRS,» OpenMRS, 2021. [En línea]. Available: <https://wiki.openmrs.org/display/docs/Introduction+to+OpenMRS>.
- [8] Bahmni Wiki, «Setting up Bahmni Dev Environment,» 2022. [En línea]. Available: <https://bahmni.atlassian.net/wiki/spaces/BAH/pages/32604585/Setting+up+Bahmni+Dev+Environment>.
- [9] OpenMRS, «Medical Free/Libre and Open Source Software,» [En línea]. Available: <https://www.medfloss.org/node/40>.