

MANUAL TECNICO

MENTALHEALTH

16 DE MAYO DE 2016

UNIVERSIDAD TECNOLOGICA DE PEREIRA

Carolina Gómez Trejos

Sebastián Duque Restrepo

Fernando Santa López

William Enrique Choque Valderrama

Nicolás Restrepo Villa

**CONTENIDO**

[CONTENIDO](#h.gjdgxs)

[1. INSTRUCCIONES DE INSTALACION EN LOCALHOST](#h.30j0zll)

[a. INSTALACION](#h.1fob9te)

[b. EJECUCION](#h.3znysh7)

[c. REINICIO](#h.2et92p0)

[2. INSTRUCCIONES DE INSTALACION EN SERVIDOR EXTERNO](#h.tyjcwt)

[3. REQUISITOS DEL SISTEMA](#h.3dy6vkm)

[4. COPIAS DE SEGURIDAD](#h.1t3h5sf)

[5. CREACION DE USUARIO ADMINISTRADOR](#h.4d34og8)

[6. CARACTERISTICAS DE LOS USUARIOS DEL SISTEMA](#h.2s8eyo1)

[a. ADMINISTRADOR (Root)](#h.17dp8vu)

[c. MEDICO GENERAL](#h.3rdcrjn)

[d. MEDICO ESPECIALISTA](#h.26in1rg)

[e. PACIENTE](#h.lnxbz9)

[7. DEFINICION DEL DESARROLLO](#h.35nkun2)

[8. DEFINICION DE IMPLEMENTACION O LIBRERIAS UTILIZADAS](#h.1ksv4uv)

[9. VISTA DE IMPLEMENTACION (O FISICA)](#h.44sinio)

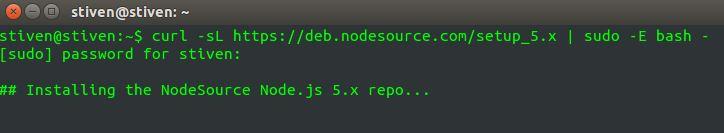
[10. GLOSARIO](#h.2jxsxqh)

1. **INSTRUCCIONES DE INSTALACION EN LOCALHOST**
   1. ***INSTALACION***

Se utilizara el entorno de programación Node.js que está diseñado para programar en el lenguaje JavaScript. Se instala de la siguiente manera:

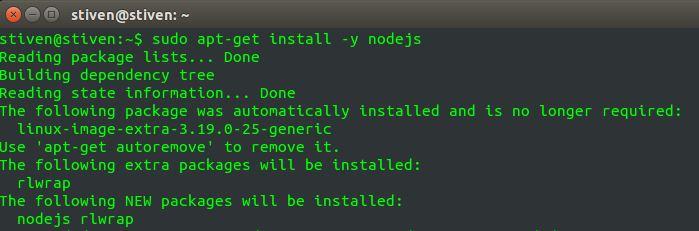
En Ubuntu se debe abrir la terminal y después copiar el siguiente comando:

*curl -sL https://deb.nodesource.com/setup\_5.x | sudo -E bash –*



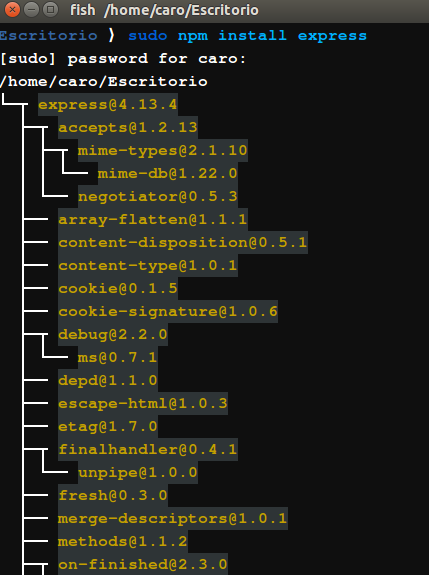
Después de que haya terminado la instalación, se deben instalar las dependencias que se usaran con el siguiente comando:

*sudo apt-get install -y nodejs*



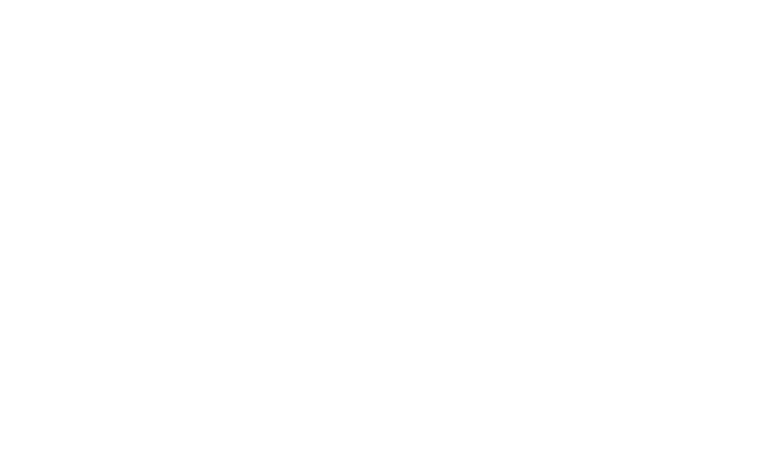
Posteriormente debemos instalar el framework express que es el que se utilizó para manejar la estructura del modelo vista-controlador, el comando es el siguiente:

*npm install express*



Finalmente instalamos la base de datos, que en nuestro caso utilizaremos una base de datos no relacional llamada couchDB. Para la instalación se debe escribir en la terminal el siguiente comando:

*sudo apt-get install couchdb*

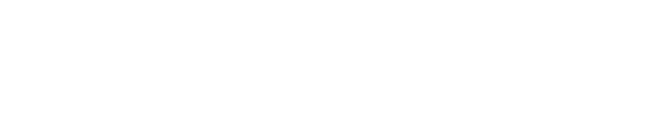


Una vez descargados todas las herramientas, se debe descomprimir la carpeta del proyecto “MentalHealth” en el directorio que se desee.

* 1. ***EJECUCION***

Primero se accede con el siguiente comando a la carpeta en la cual descomprimimos el proyecto “MentalHealth”:

*cd MentalHealth*



Para crear la base de datos se debe usar el siguiente comando:

*curl -X PUT* [*http://localhost:5984/mental\_health*](http://localhost:5984/mental_health)

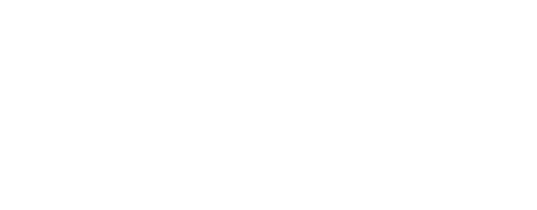
Luego se instalan las dependencias que se están usando con el siguiente comando:

*sudo npm install*



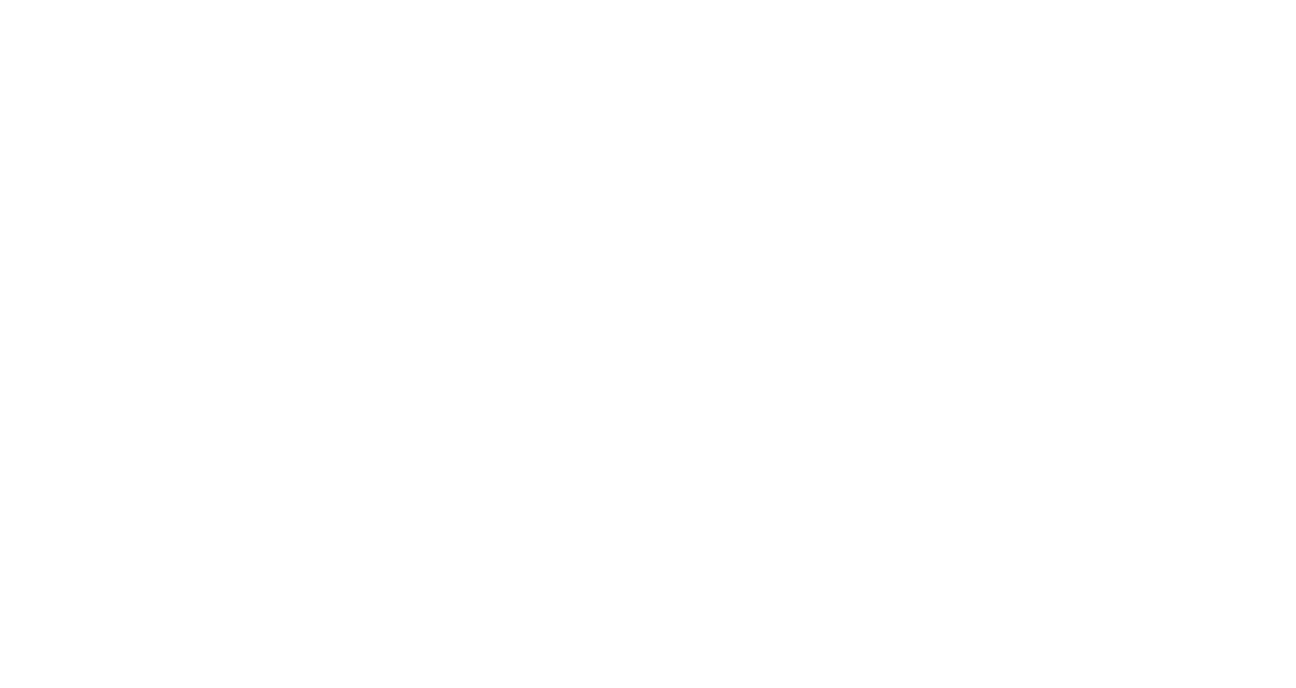
Luego para correr la aplicación se usa el siguiente comando:

*sudo npm start*



Finalmente para visualizar la aplicación se abre cualquier navegador y se digita en la barra de navegación lo siguiente:

*localhost:4000*



* 1. ***REINICIO***

Para borrar la base de datos creada en el punto anterior en la terminal se debe copiar el siguiente comando en la terminal:

*curl -X DELETE http://localhost:5984/mental\_health*

Para reiniciar el motor de base de datos se debe copiar el siguiente comando en la terminal:

*sudo start couchdb*

1. **INSTRUCCIONES DE INSTALACION EN SERVIDOR EXTERNO**

La instalación de la aplicación se hace de igual manera que en el localhost, pero en el equipo servidor, y su ejecución se hace igual que en local host solo que es por medio de un script que se encarga de correr la aplicación cada que el equipo servidor se apague.

1. **REQUISITOS DEL SISTEMA**

**Sistemas operativos compatibles:**

* Windows 7 o superior (x86 y x64)
* Ubuntu 14.04
* CENTOS 7

**Dependencias de software0**

* Node.js versión 5.x
* express versión 4.x
* couchdb versión 1.6.1

**Requisitos mínimos de Hardware**

* Procesador de 1.6 Hz o superior
* RAM 512MB
* Espacio en disco de 2 GB

1. **COPIAS DE SEGURIDAD**

Si se desea sacar una copia de seguridad de la base de datos, basta con copiar o descargar el archivo que se encuentra en la siguiente ubicación:

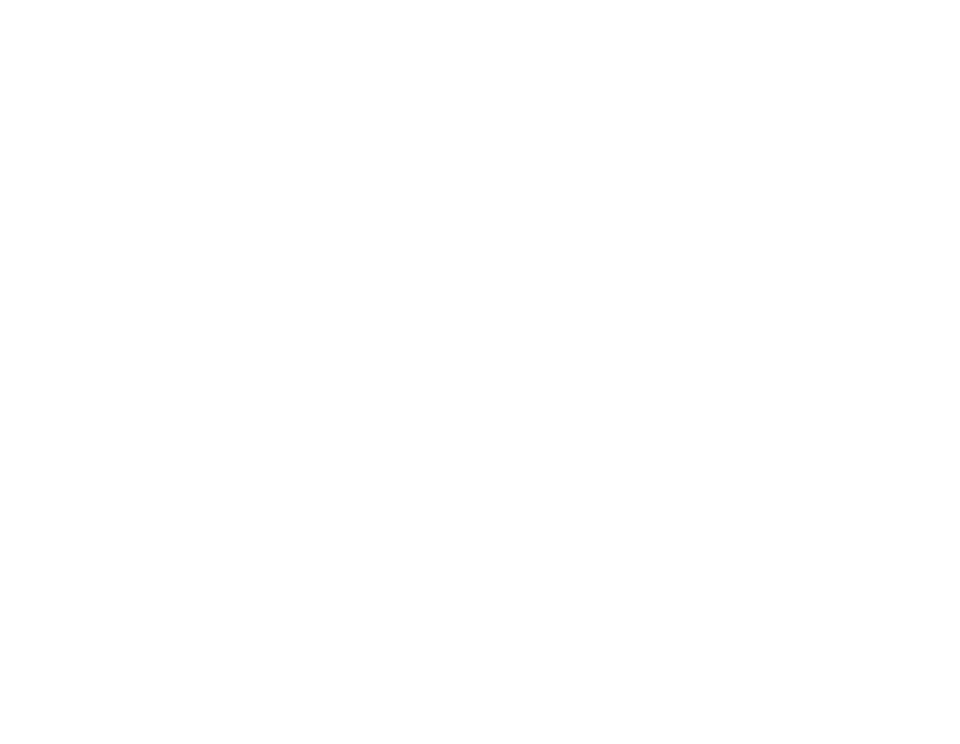
*/var/lib/couchdb/mental\_health.couch*

En cuanto a las copias de seguridad del aplicativo como tal; en el momento se encuentra una copia en cada uno de los equipos de los miembros del grupo y en la carpeta del proyecto en dropbox.

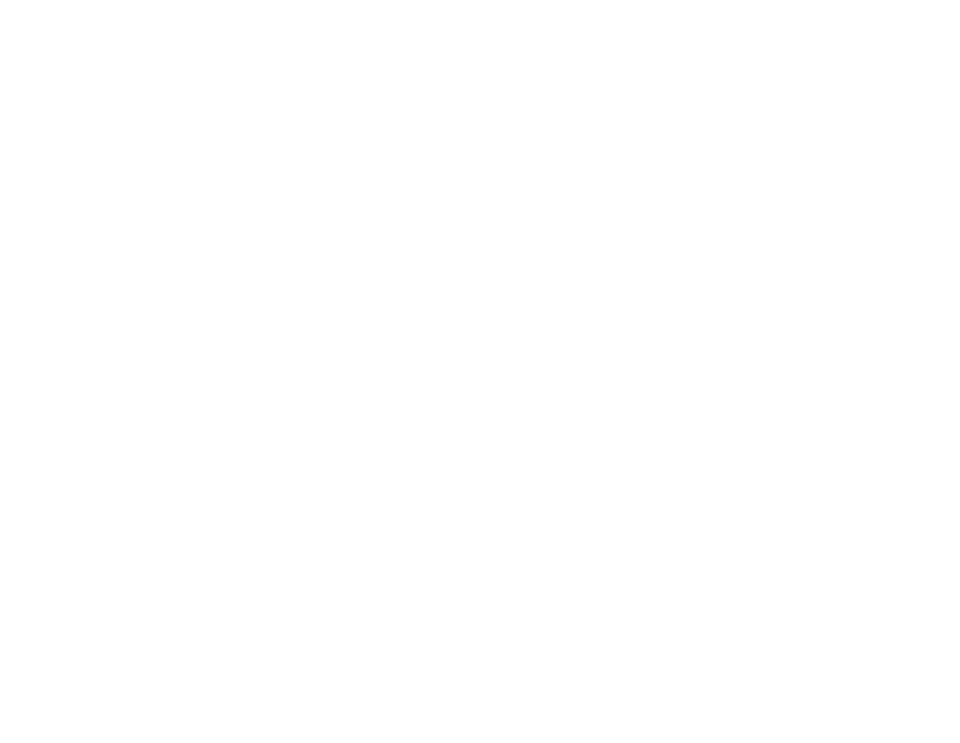
1. **CREACION DE USUARIO ADMINISTRADOR**

Primero se deben seguir los pasos mencionados en el numeral de EJECUCION si nos encontramos de forma local, en caso contrario solo debemos acceder a la siguiente dirección:

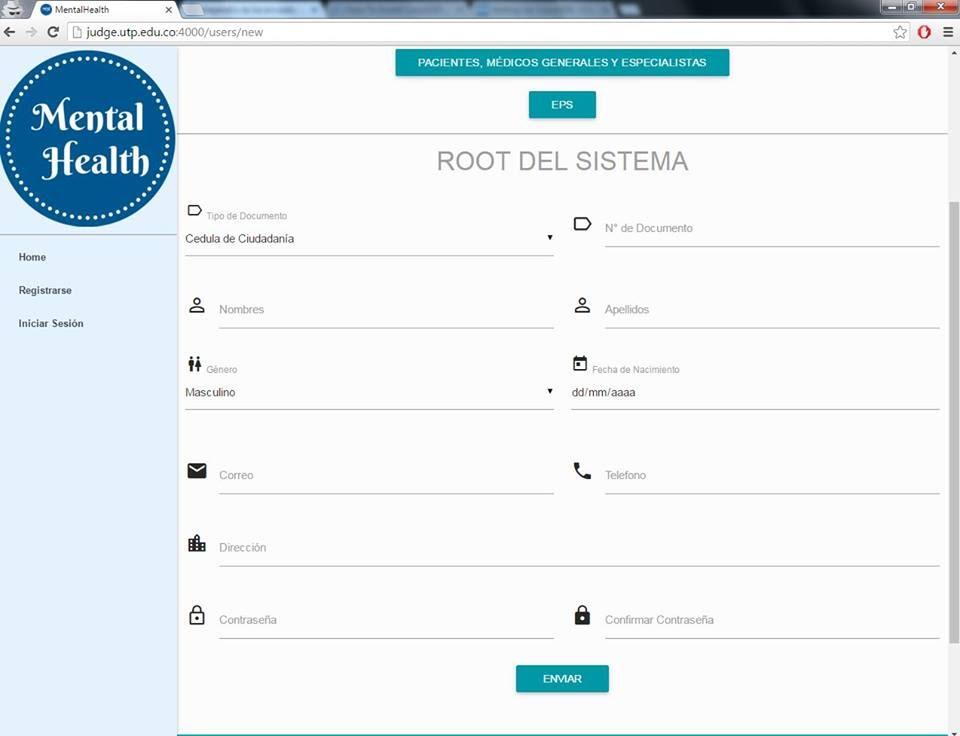
judge.utp.edu.co:4000



Luego le damos clic en registrarse y debe aparecer el siguiente menú:

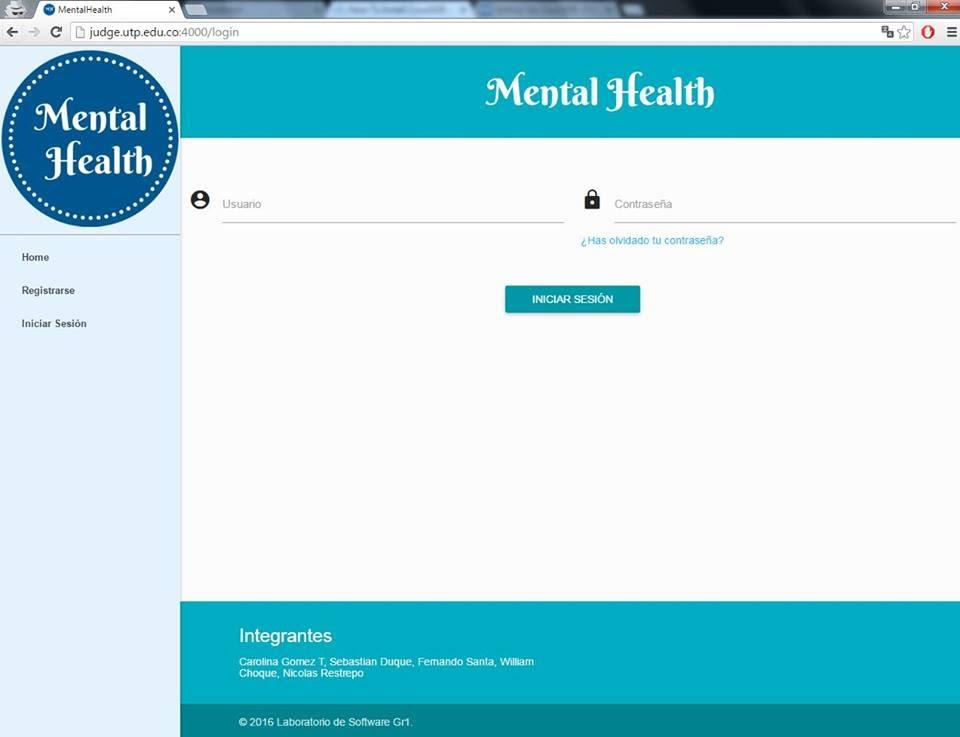


Después le damos clic en la opción ROOT DEL SISTEMA y deben desplegarse los siguientes campos:



Se deben llenar dichos campos y posteriormente darle clic en ENVIAR.

Para iniciar sesión el usuario debe ingresar su número de documento y contraseña:



1. **CARACTERISTICAS DE LOS USUARIOS DEL SISTEMA**

Todos los usuarios aquí mencionados, a excepción del administrador; deben pasar por un proceso de registro previo al login para poder acceder al sistema.

* 1. ***ADMINISTRADOR (Root)***

El administrador es una cuenta única y es el encargado de rechazar o aprobar todas las cuentas de las EPS e informa mediante un correo automático dicho estado de aprobación. También tiene la capacidad de modificar datos de los perfiles de las EPS, el root podría gestionar su propio perfil, también tendría la capacidad de reactivar la cuenta de las EPS.

También se podrán borrar cuentas de las EPS, dicho borrado se realizara de forma lógica en la base de datos.

* 1. **EPS**

Una EPS se encarga de aprobar o rechazar las cuentas de los roles de los usuarios (Pacientes, Médico General, Médico Especialista) e informa por medio de un correo automático el estado de aprobación de dichas cuentas, también tiene la capacidad de modificar los perfiles de cada usuario, además la EPS también podría eliminar las cuentas de los usuario (El borrado en la Base de datos es lógico).

También la EPS podría modificar su propio perfil e inclusive podría eliminar su propia cuenta (Borrado lógico en la base de datos).

* 1. ***MEDICO GENERAL***

El Médico General podrá modificar su perfil, podrá gestionar su agenda (Poner disponibilidad, modificar y eliminar citas), se enviara un correo de confirmación de modificación o eliminación de citas, también podrá eliminar su cuenta (Borrado de la base de datos es lógico).

También podrá crear, modificar y consultar la historia clínica de los pacientes con los cuales tiene asignado una cita.

* 1. ***MEDICO ESPECIALISTA***

El Médico Especialista podrá modificar su perfil, podrá gestionar su agenda (Poner disponibilidad, modificar y eliminar citas), se enviara un correo de confirmación de modificación o eliminación de citas, también podrá eliminar su cuenta (Borrado de la base de datos es lógico).

También podrá crear, modificar y consultar la historia clínica de los pacientes con los cuales tiene asignado una cita.

* 1. ***PACIENTE***

El Paciente podrá modificar su perfil, podrá gestionar sus citas (agendarla, modificarla o eliminarla) en la agenda que tiene cada uno de los médicos (general o especialista) que estén asociados a la misma EPS de dicho paciente y se enviara un correo de confirmación por modificación o eliminación de la cita, además, también podrá eliminar su cuenta (Borrado de la base de datos es lógico).

**NOTA:** Los usuarios (Medico general, Especialista y Paciente) que tengan varios roles podrán intercambiarse entre los diferentes roles que tengan asociadas.

1. **DEFINICION DEL DESARROLLO**

El desarrollo se llevó a cabo en Node.js, el cual es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor (pero no limitándose a ello) basado en el lenguaje de programación de JavaScript, cabe resaltar que Node.js permite construir aplicaciones altamente escalables y escribir código que maneje decenas de miles de conexiones simultáneas en una sólo una máquina física.

Además de usar este entorno se utilizó express.js el cual es un framework de desarrollo de aplicaciones web minimalista y flexible para Node.js, este framework utiliza el MVC (Modelo Vista Controlador) además es robusto, rápido, flexible y muy simple.

Entre otras características, ofrece Router de URL (Get, Post, Put, Delete, etc) los cuales nos facilitan el manejo de las peticiones HTTP que se llevarán a cabo en la ejecución de la aplicación, además también posee facilidades para motores de plantillas (Jade, EJS, JinJS, …).

1. **DEFINICION DE IMPLEMENTACION O LIBRERIAS UTILIZADAS**

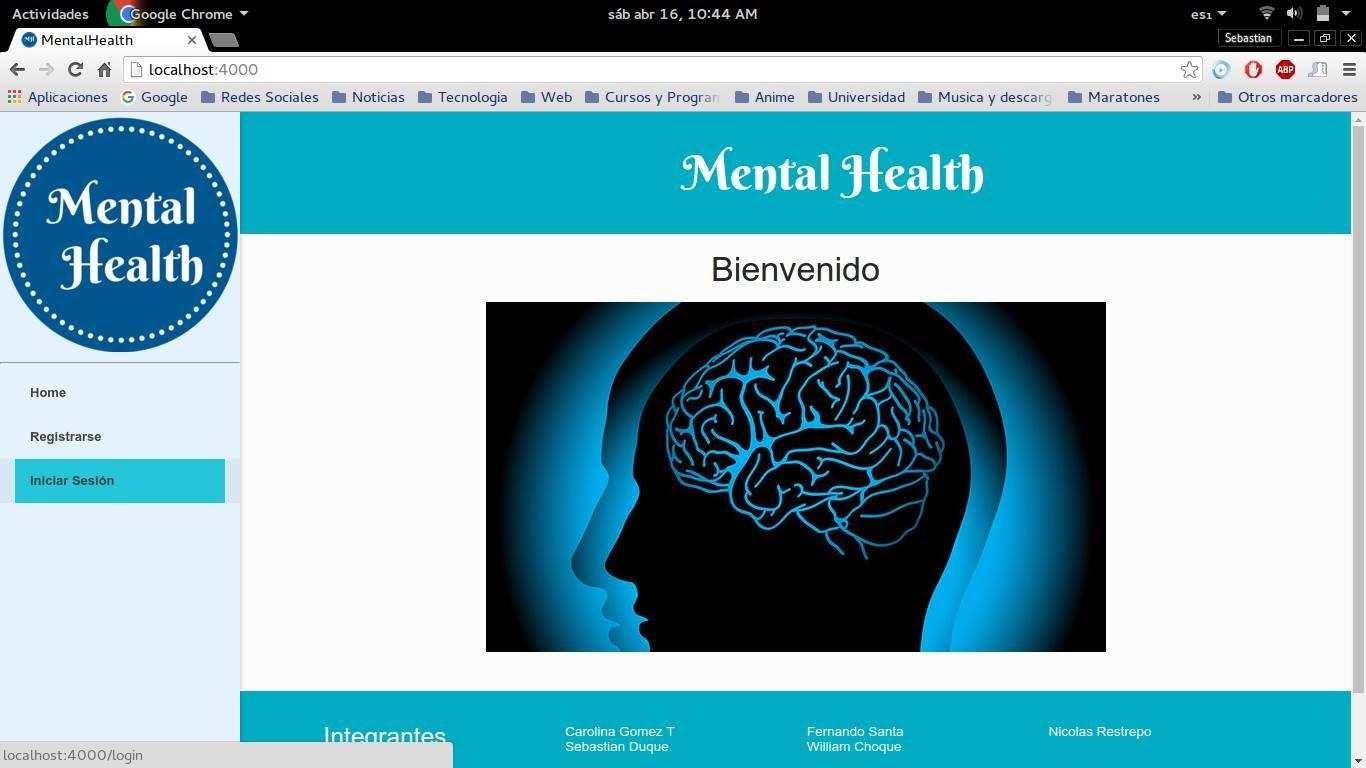
En la implementación se hizo uso de una herramienta de Javascript llamada **JQUERY** de código libre, cuya tarea principal es simplificar la codificación y además permite realizar efectos e interacciones con documentos HTML de una manera muy simple e intuitiva.

Para las diferentes vista desarrolladas de utilizar **EJS** el cual es un lenguaje de plantillas simple que le permite generar el formato HTML con JavaScript, una de las ventajas que nos ofrece EJS es que posee una compilación rápida y renderizado de las vistas y permite la interacción de datos con el servidor lo cual facilita mucho el poder visualizar estos datos en el cliente.

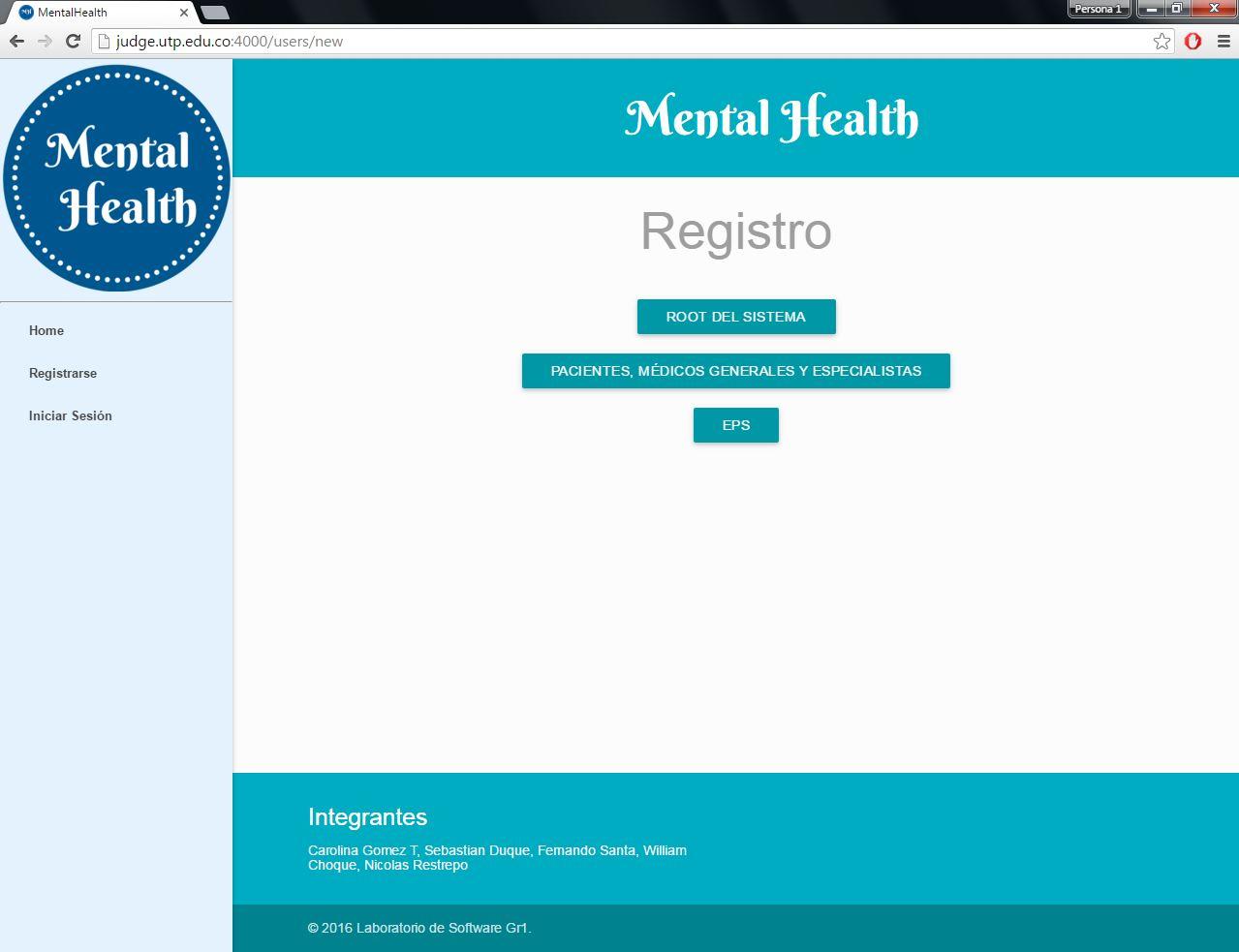
El estilo que poseen las vistas está desarrollado en **Materialize,** el cual es una librería creada en CSS que permite la elaboración de páginas web de muy buena presentación y adaptables a cualquier dispositivo (responsive).

1. **VISTA DE IMPLEMENTACION (O FISICA)**

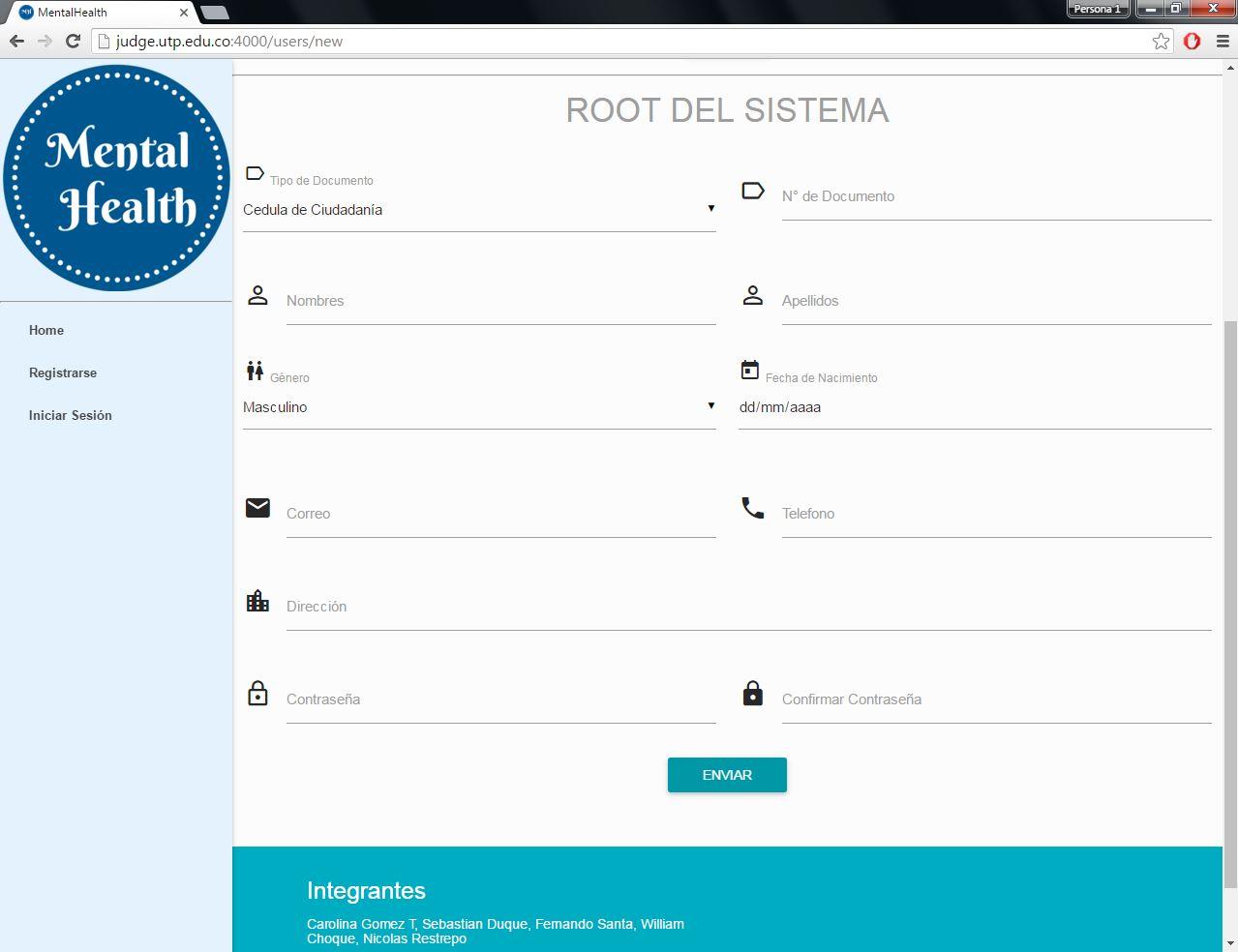
Página de Inicio:



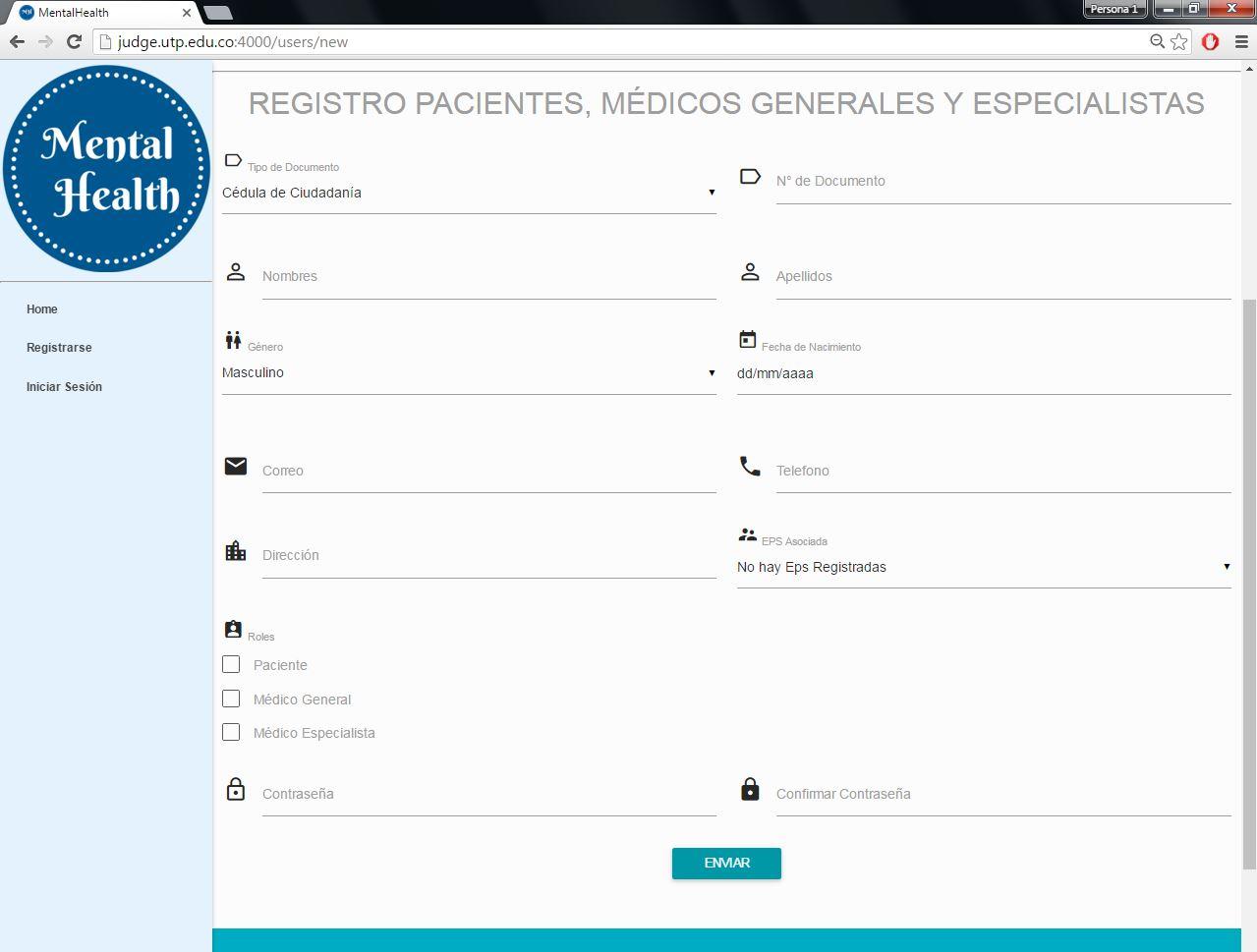
Página de Registro:



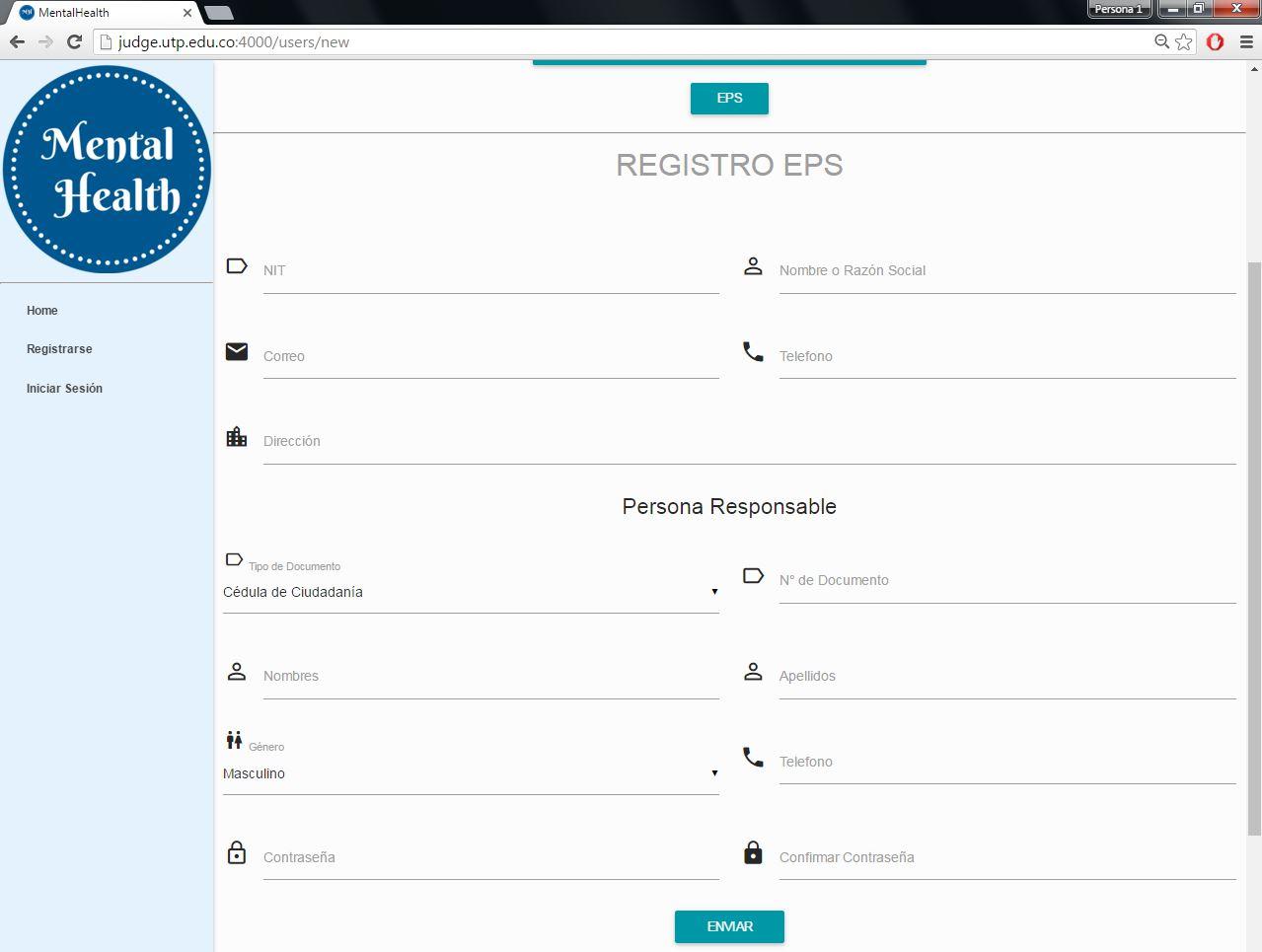
Registro ROOT:



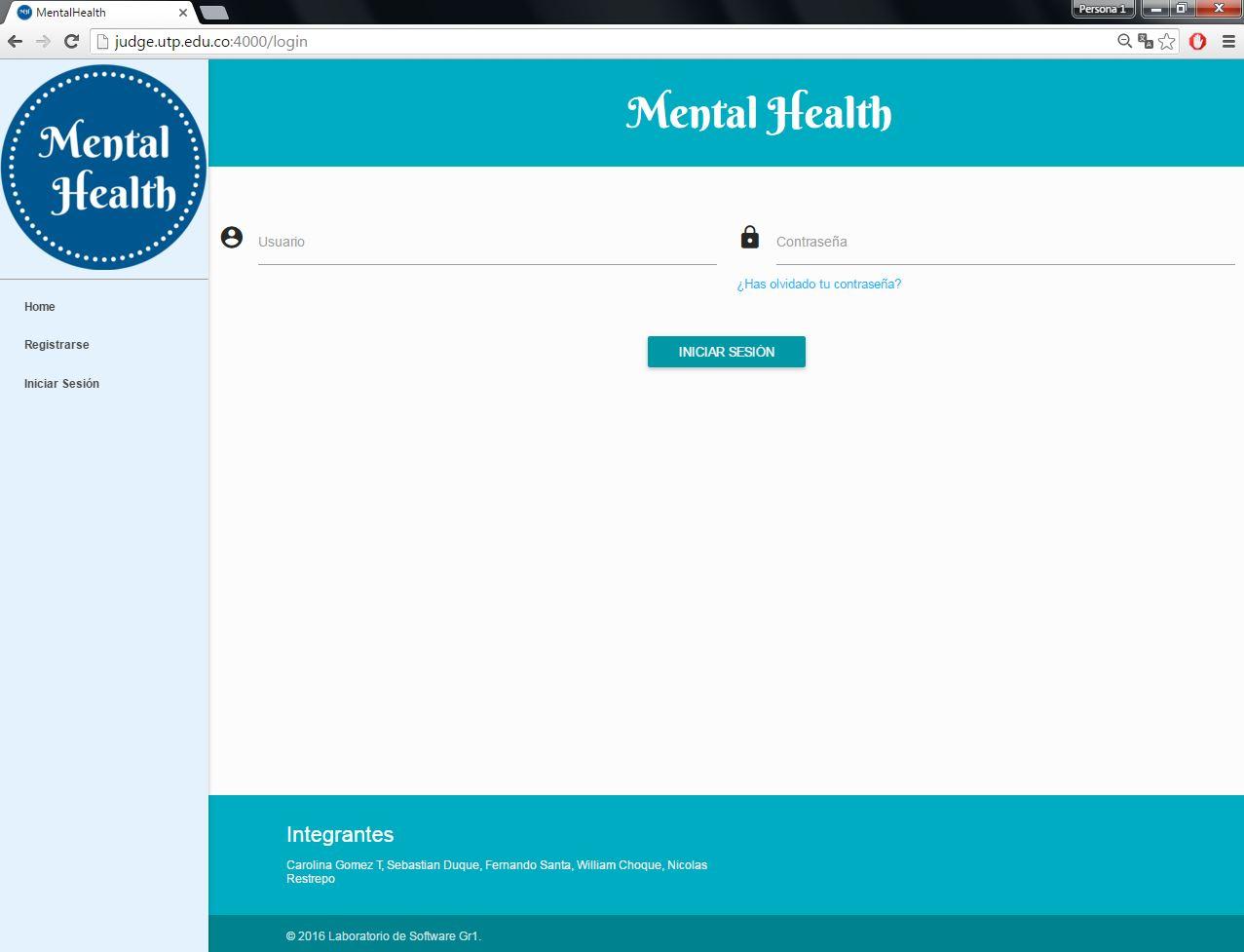
Registro Paciente, Medico General y Especialista:



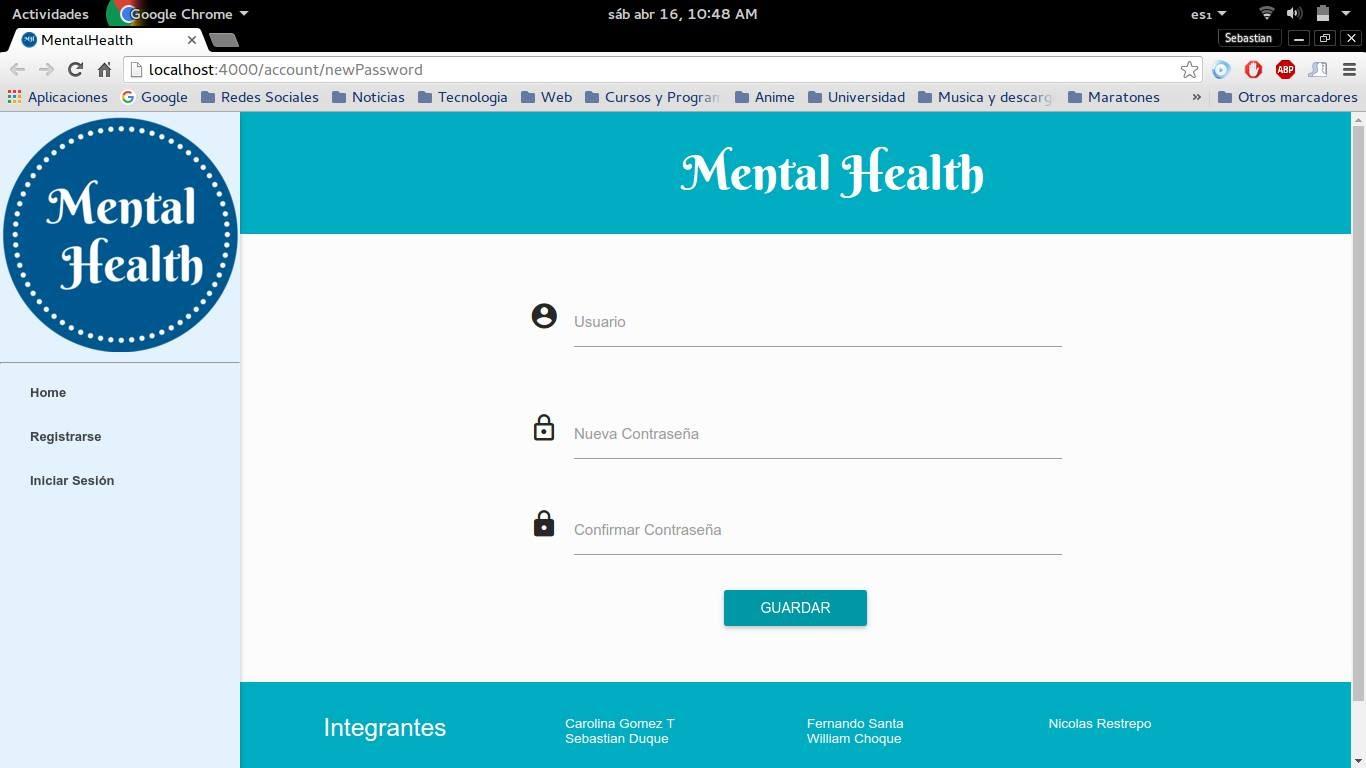
Registro EPS:



Inicio de Sesión:



Recuperar Contraseña:



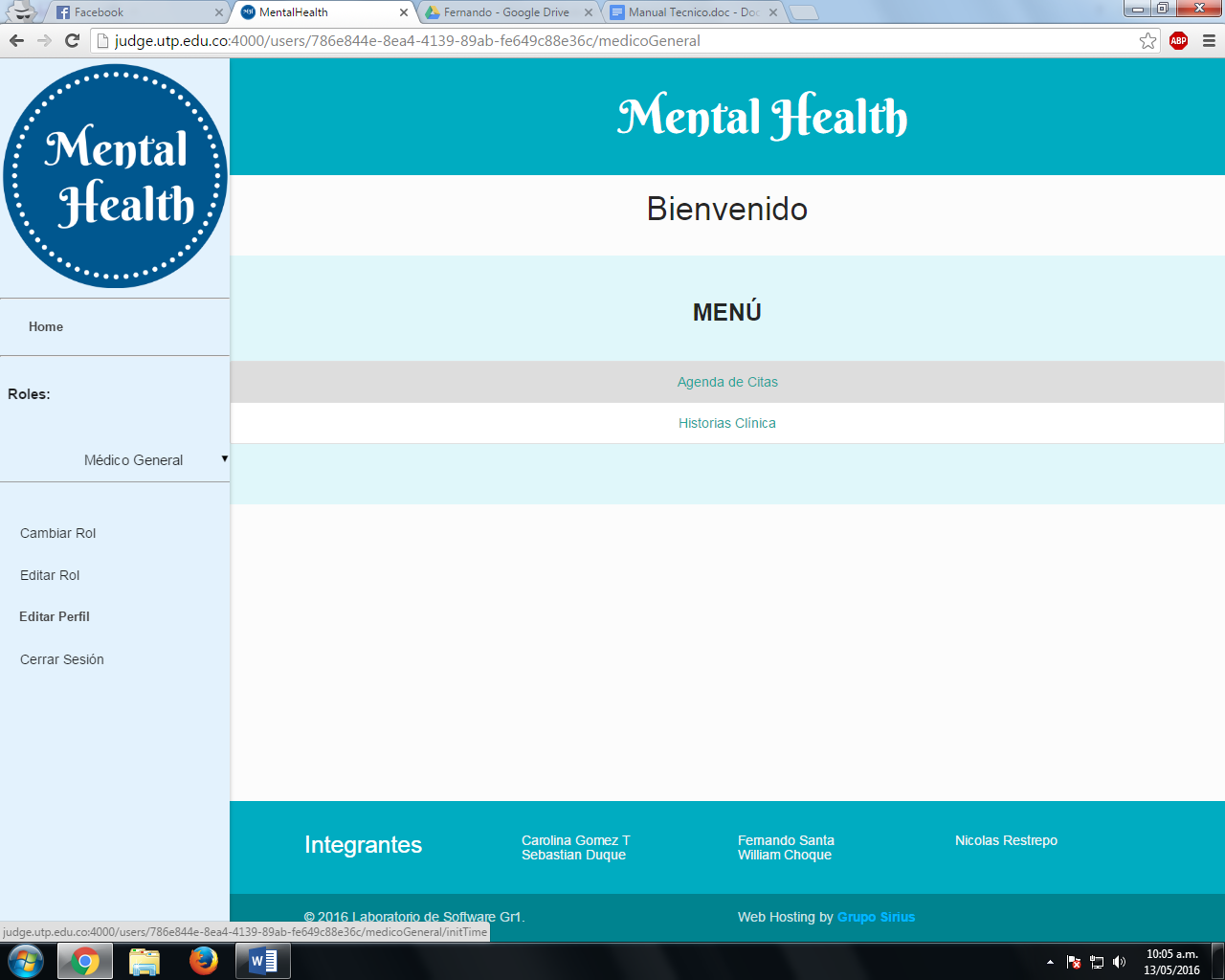
Gestión de usuario Administrador:



Gestión de usuario EPS:



Gestión de usuario Medico General y Especialista:



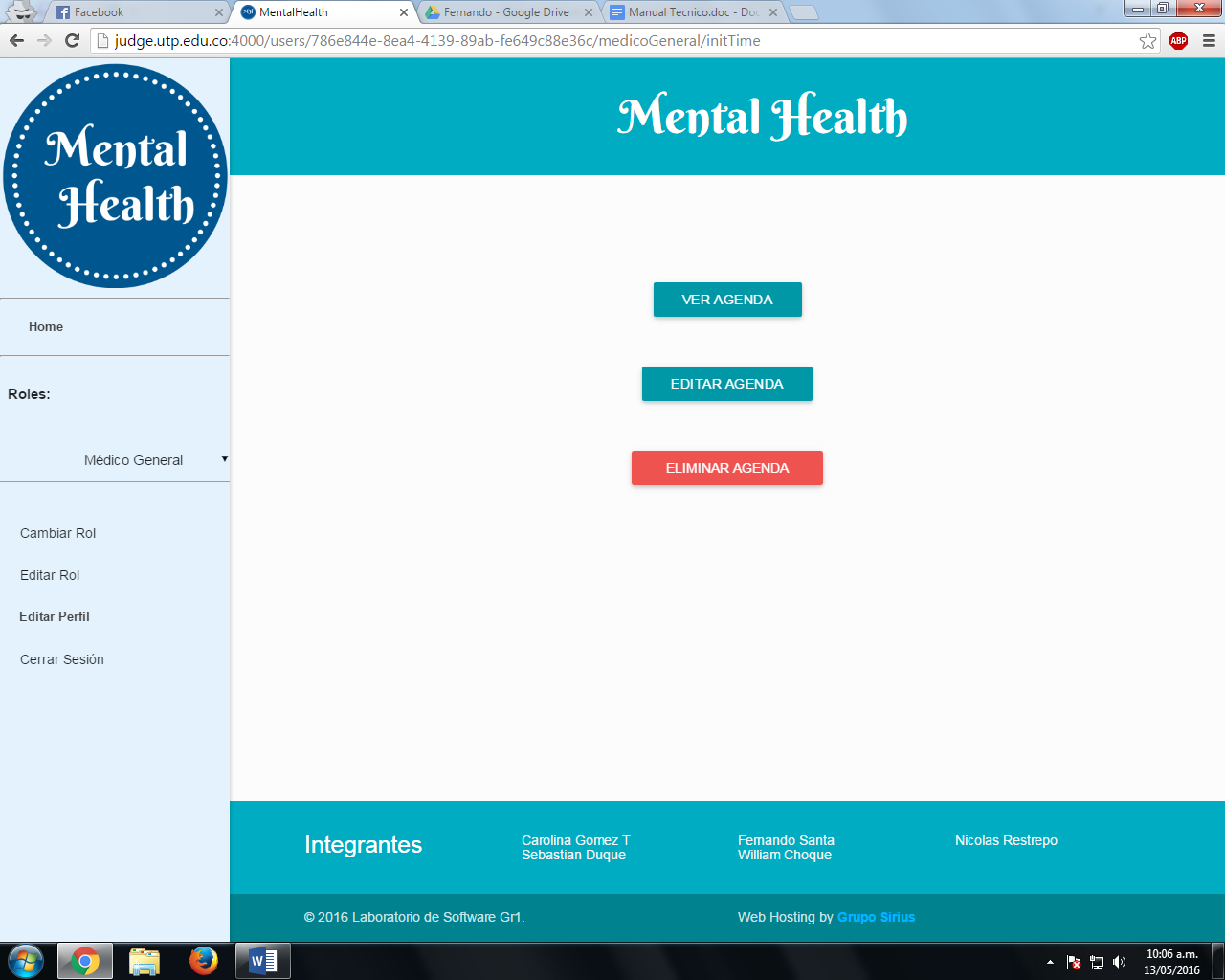
Gestión de usuario Paciente:

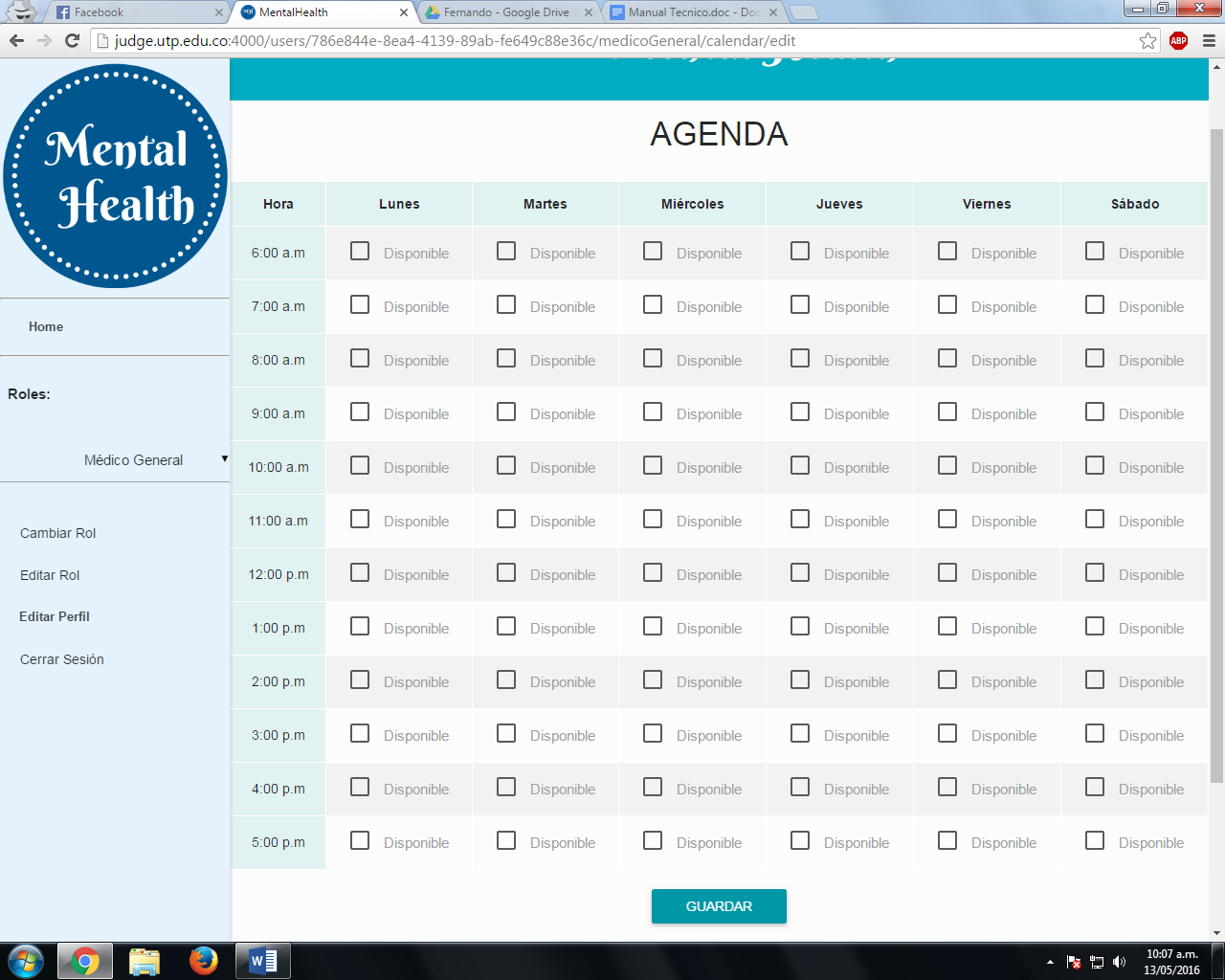


Correo de confirmación:



Gestión de Citas:

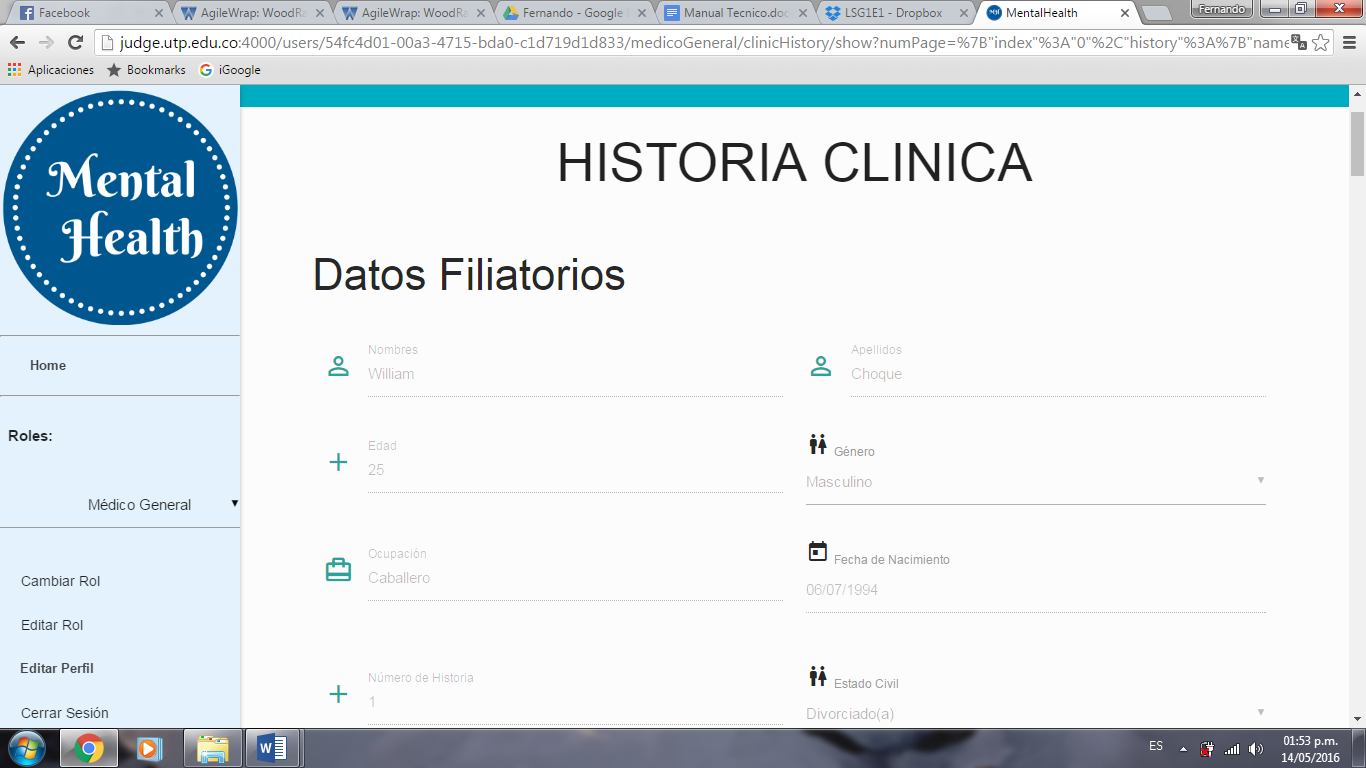


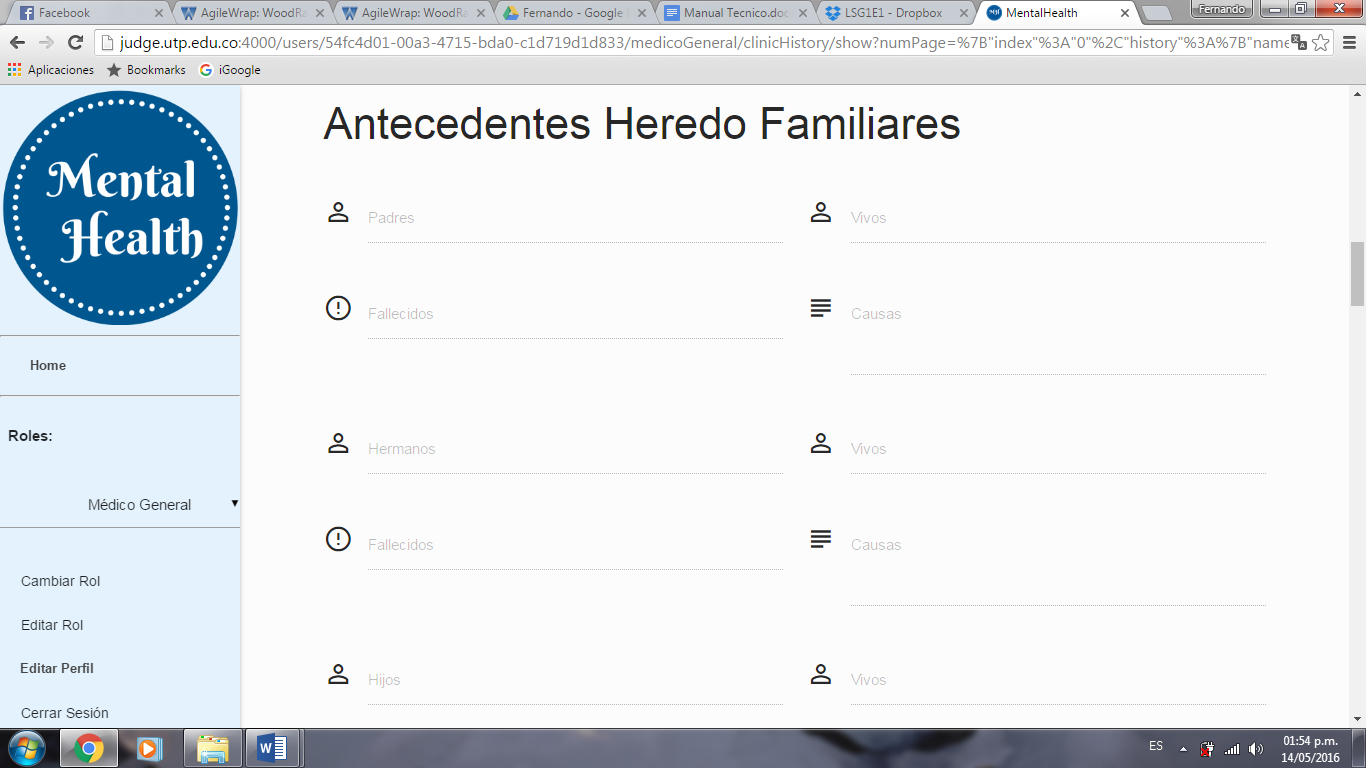
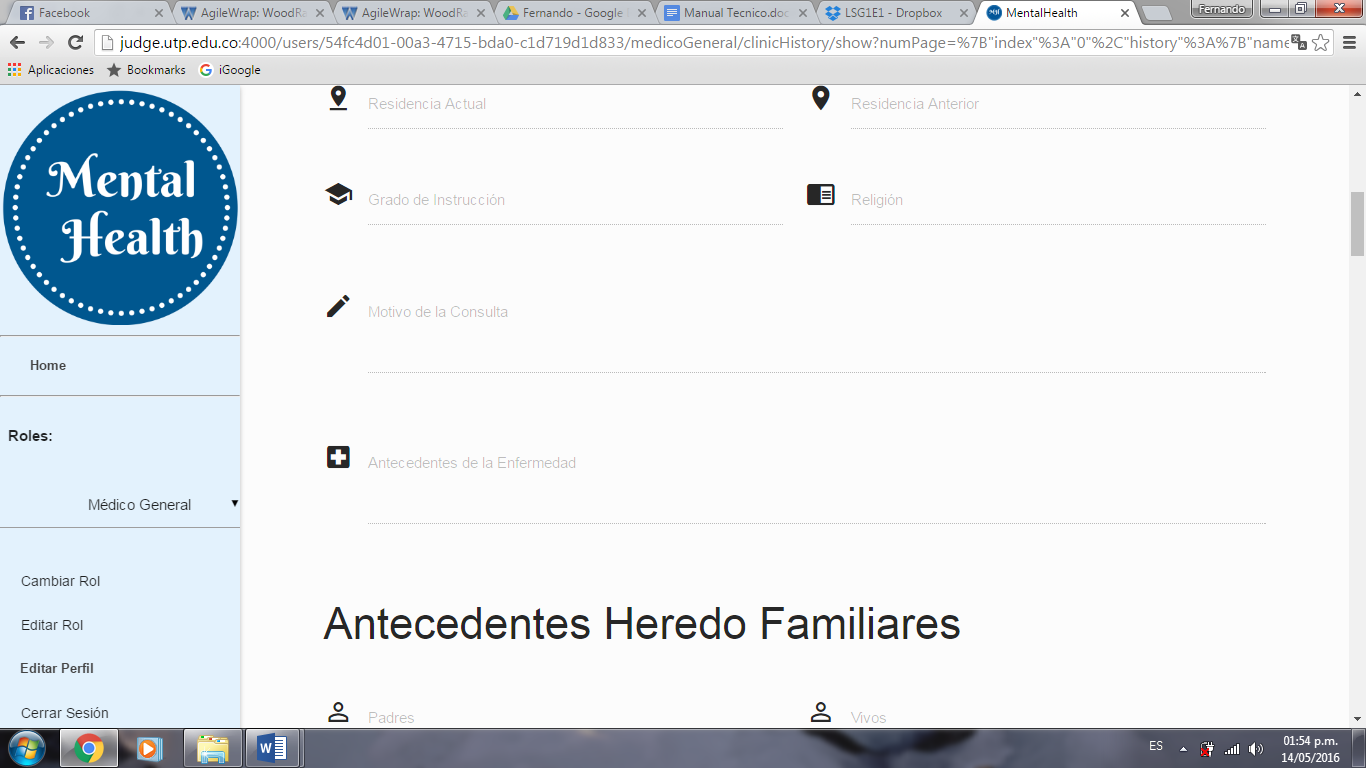


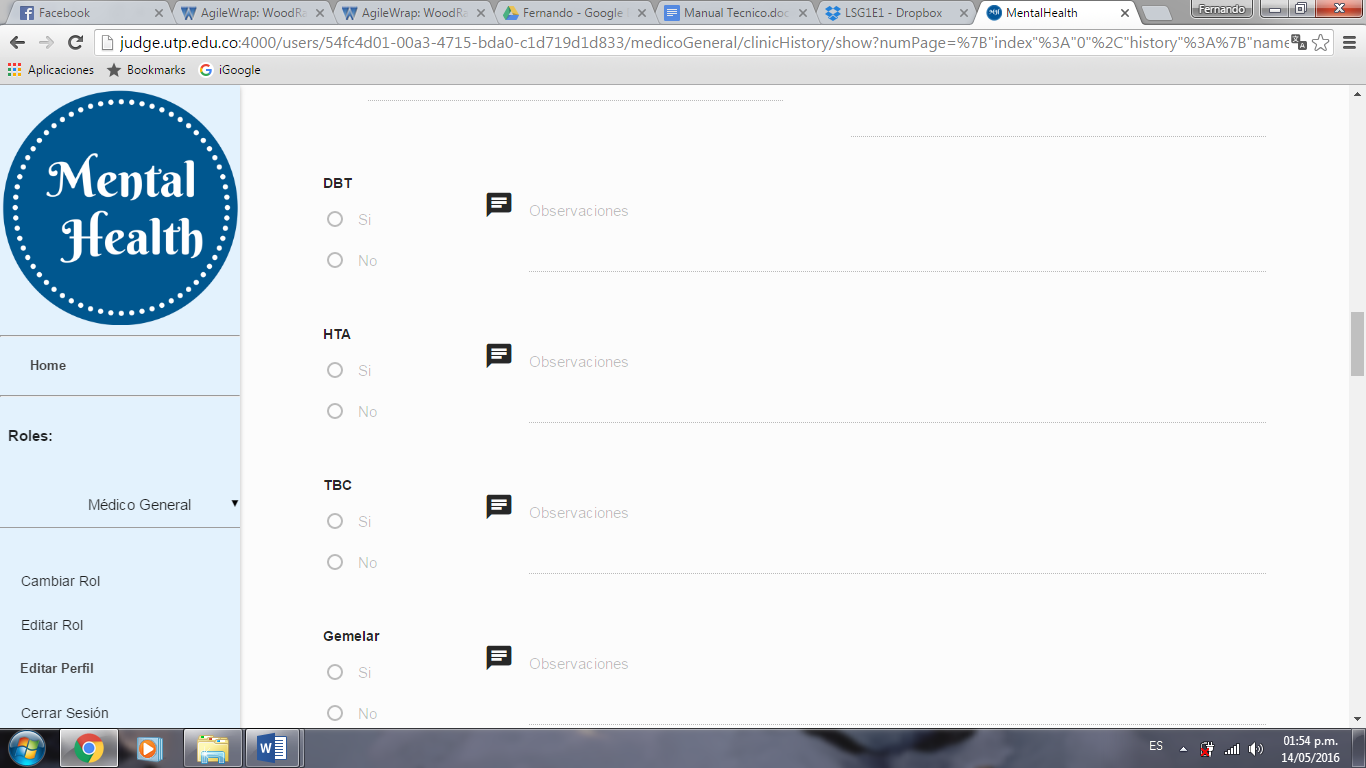
Agenda:

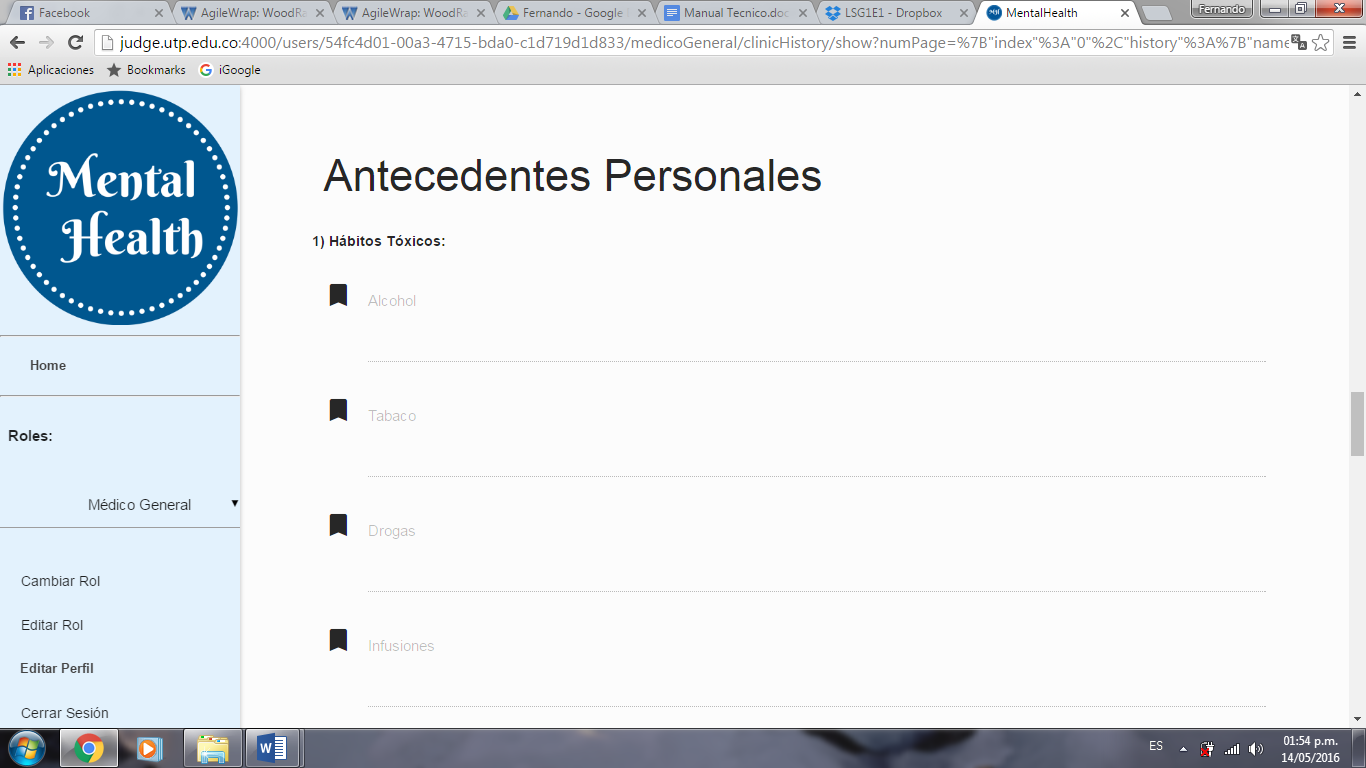
Historia Clinica:

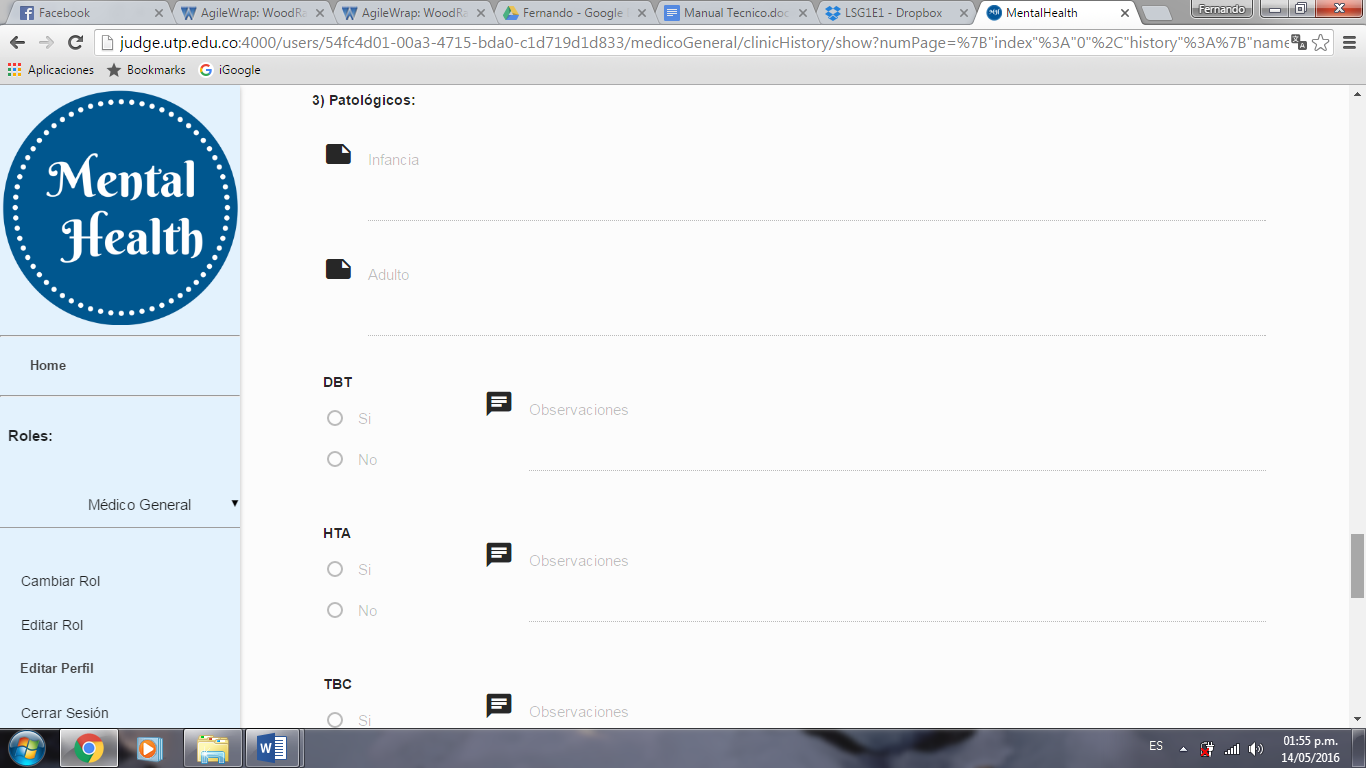
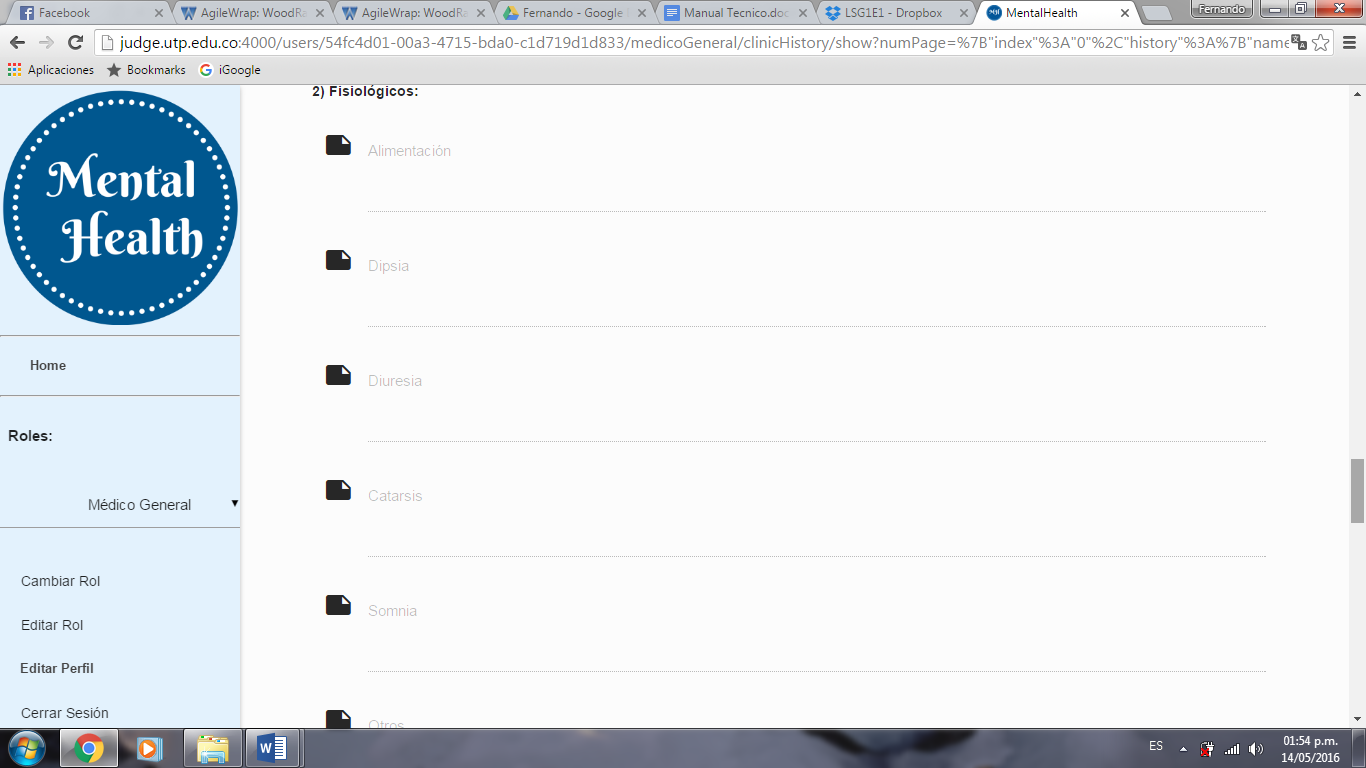


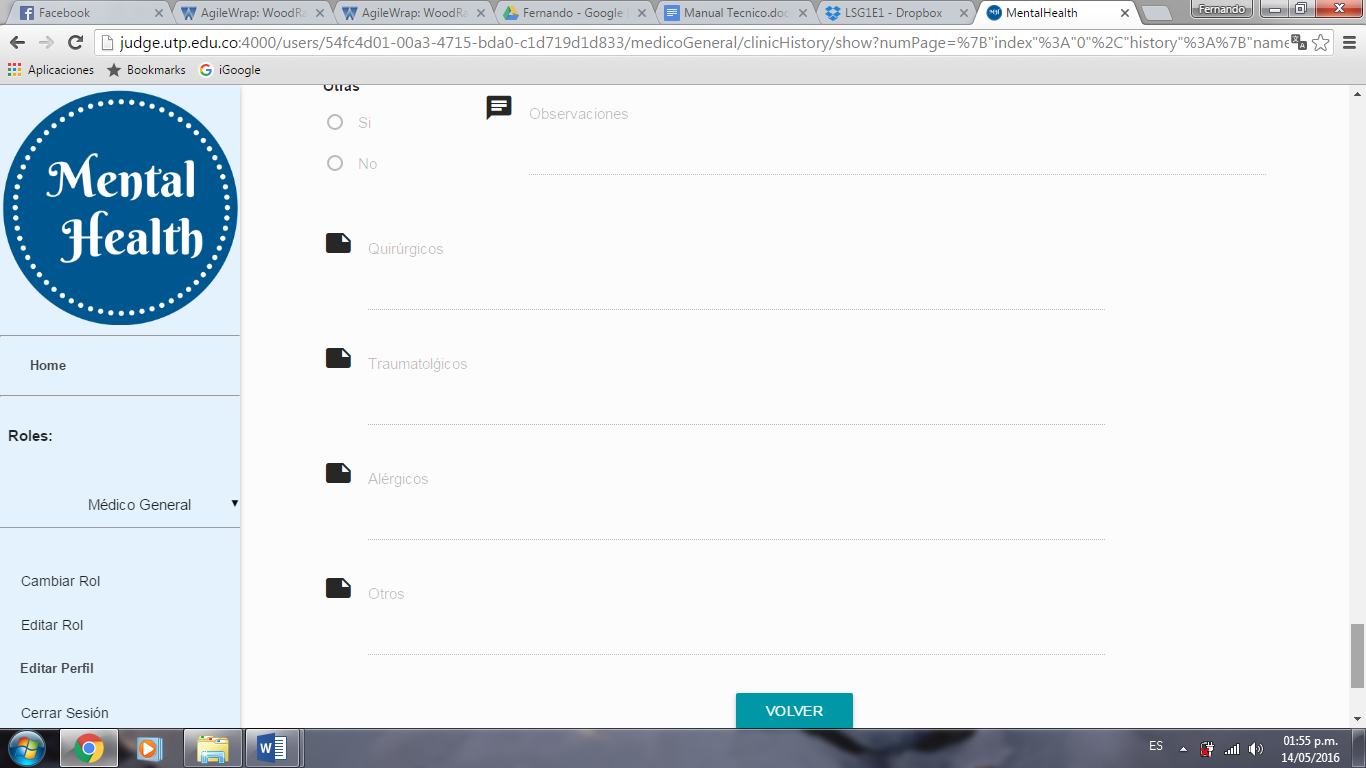












1. **GLOSARIO**

* <https://nodejs.org/en/download/package-manager/>
* <http://www.tutorialspoint.com/couchdb/couchdb_installation.htm>
* <https://www.digitalocean.com/community/tutorials/how-to-install-couchdb-and-futon-on-ubuntu-14-04>