

INFORME DE DESARROLLO

PROYECTO GLUCOMETRÍAS

DIRECTOR DE PROYECTO
ALBEIRO CORTES CABEZAS

DESARROLLADORES
GERMAN ALEJANDRO MORA GOMEZ
RUBIEL CHAPAL RAMOS

UNIVERSIDAD SURCOLOMBIANA
GRUPO DE TRATAMIENTO DE SEÑALES Y TELECOMUNICACIONES GTST
VICERRECTORIA DE INVESTIGACION Y PROYECCION SOCIAL

2016

Tabla de contenido

SERVIDOR DE BASE DE DATOS	3
PLATAFORMA WEB	4
TECNOLOGIAS DE DESARROLLO	5
ESTRUCTURA DEL PROYECTO WEB	6
MODO DESARROLLO PARA PRUEBAS EN NETBEANS	8
CONSIDERACIONES EN EL PROYECTO WEB	8
APLICACIÓN ANDROID	9
TECNOLOGIAS DE DESARROLLO	10
ESTRUCTURA DEL PROYECTO ANDROID	11

Lista de figuras

<i>Figura 1. Tablas de la base de datos</i>	3
<i>Figura 2. Página inicial de la plataforma web</i>	4
<i>Figura 3. Estructura de paginas</i>	6
<i>Figura 4. Estructura del paquete de clases del proyecto.</i>	7
<i>Figura 5. Pantalla de bienvenida GlucUp-2Date</i>	9
<i>Figura 6. Listado de liberías</i>	10
<i>Figura 7. Estructura Android Studio</i>	11

SERVIDOR DE BASE DE DATOS

La base de datos contiene toda la información que se almacena para los servicios de la aplicación. Las tablas guardan los perfiles de usuarios y la información solicitada para cada uno de ellos; también se almacena la información de glucosa para los pacientes.

glucose	patient	personal	root
id INT	id INT(11)	id INT(11)	id INT(11)
identifier JSON	identifier JSON	identifier JSON	ndi VARCHAR(45)
code JSON	name JSON	name JSON	ndiValue VARCHAR(45)
subject JSON	telecom JSON	telecom JSON	name VARCHAR(45)
issued TEXT	gender TINYTEXT	address JSON	phone VARCHAR(45)
performer JSON	birthDate TINYTEXT	gender TINYTEXT	password VARCHAR(45)
valueQuantity JSON	address JSON	birthDate TINYTEXT	address VARCHAR(45)
interpretation JSON	maritalStatus JSON	practitionerRole JSON	email VARCHAR(45)
referenceRange JSON	contact JSON	password VARCHAR(45)	
comments TEXT	communication JSON	token_id VARCHAR(200)	
	managingOrganization LONGTEXT		
	password VARCHAR(45)		
	token_id VARCHAR(200)		
	bloodType VARCHAR(5)		
Indexes	Indexes	Indexes	Indexes

Figura 1. Tablas de la base de datos

Las tablas *patient*, *personal* y *root* contienen la información de los usuarios dependiendo de su perfil como paciente, personal médico o administrador del sistema respectivamente. La tabla *glucose* contiene los parámetros para almacenar la medida de glucosa de los pacientes. De acuerdo al estándar HL7 para el manejo de la información médica, algunos registros guardan la información en formato JSON, lo que permite transmitir los datos de acuerdo al estándar.

PLATAFORMA WEB

La plataforma web permite a los usuarios realizar funciones para el manejo de la información registrada en el sistema.

- **Usuario administrador:** Es registrado a través de la página web. Este usuario tiene la función de registrar a los pacientes y el personal médico. Puede actualizar la información de los usuarios o eliminarlos.
- **Usuario personal:** Puede ver su información personal en la página web, además tiene acceso a los datos del paciente. La plataforma le permite registrar medidas de glucosa del paciente y ver el historial de medidas en forma de tabla o en modo gráfico.
- **Usuario paciente:** Puede ver su información personal en la página web. La plataforma le permite registrar medidas de glucosa y ver el historial de medidas en forma de tabla o en modo gráfico.

Para mayor información sobre las funciones de la plataforma web, puedes descargar el manual de usuario [Aqui](#).



Figura 2. Página inicial de la plataforma web

TECNOLOGIAS DE DESARROLLO

Software de desarrollo

Para el desarrollo del proyecto web usamos programas con licencia de software libre

- *Proyecto desarrollado en:*
NetBeans IDE 8.1 (Build 201510222201)
Java: 1.8.0_77; Java HotSpot(TM) 64-Bit Server VM 25.77-b03
Runtime: Java(TM) SE Runtime Environment 1.8.0_77-b03
- *Base de datos en:*
MySQL Workbench 6.3.6 build 511 CE (64 bits) Community
Puerto de para base de datos en MySQL : 3306
- *Servidor de aplicaciones:*
Apache Tomcat 8.0.27.0

Lenguajes de programación

HTML, CSS y JS: Para el manejo de vista y las funciones script en el cliente. HTML5 organiza los parámetros para el etiquetado de la estructura de la página HTML, CSS3 construye la apariencia de la página para un estilo visual interesante y JS permite la construcción de funciones de página en el cliente para evitar carga en el servidor.



JSF y JAVA: JSF es un framework de aplicaciones java con el cual se realiza el despliegue y desarrollo las páginas web del proyecto. El servidor recibe las peticiones del cliente y realiza el control de las vistas y de la base de datos usando java.



Librerías y Frameworks

- *Primefaces:* Esta librería permite la creación de contenido enriquecido para el manejo de vistas y las funciones de la plataforma. Es una librería de componentes para JSF.
- *Bootfaces:* Al igual que Primefaces permite el manejo de vistas con JSF, esta librería integra las vistas de las páginas con estilos visuales comparados con bootstrap de twitter.

- [Font Awesome](#): Esta librería contiene un compilado de iconos para darle mayor estilo visual a las páginas.
- [javax.mail versión 1.4.7](#) : permite el envío de mensajes de correo desde el servidor a los usuarios. Se usa para enviar un correo al usuario cuando solicita una recuperación de contraseña.
- [Gson versión 2.6.2](#): librería de clases para el manejo JSON usando el lenguaje java.
- [Mysql conector versión 5.1.39](#): librería para el control de las conexiones a la base de datos de MySQL desde java.

ESTRUCTURA DEL PROYECTO WEB

El proyecto en netbeans se estructuró con Maven que permite descargar las librerías directamente desde los repositorios de Maven. Las páginas .html son desplegadas de acuerdo a los perfiles de cada usuario y del control de las vistas desarrolladas en el servidor.

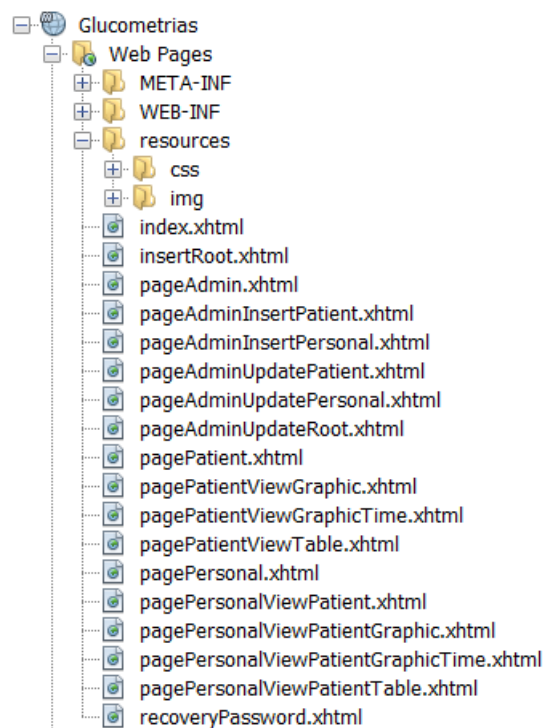


Figura 3. Estructura de paginas

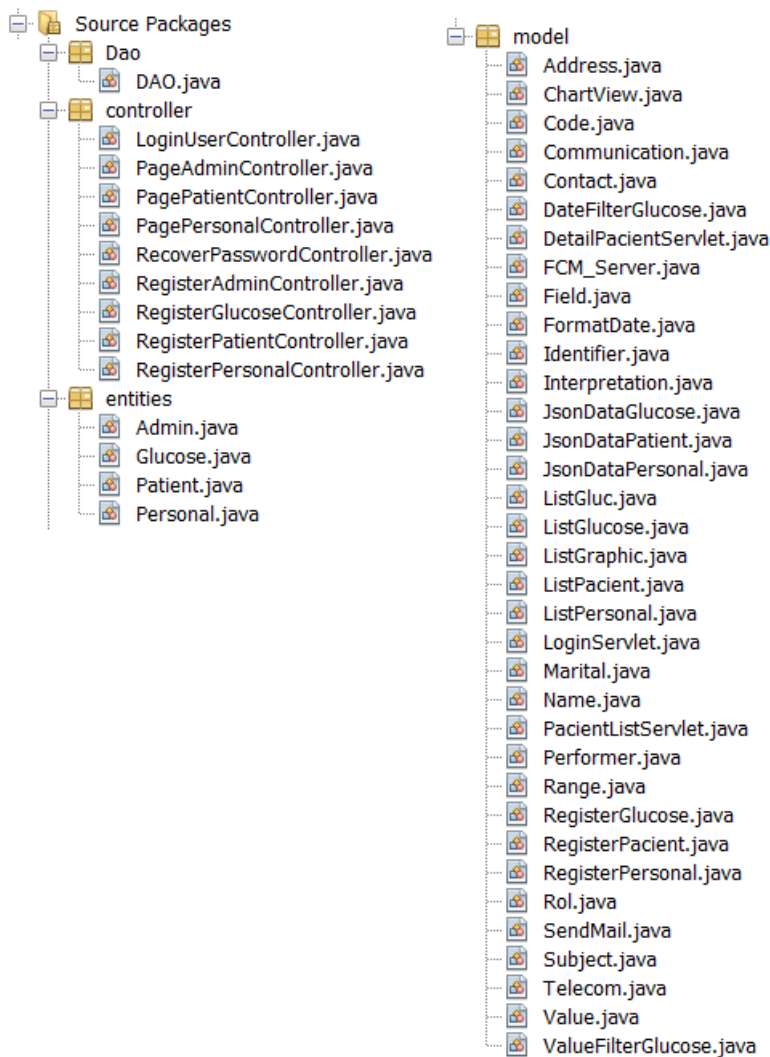


Figura 4. Estructura del paquete de clases del proyecto.

Paquete Dao: contiene la clase DAO.java que permite en control de las conexiones con la base de datos MySQL. En él se pueden encontrar todos los parámetros de conexión.

Paquete controller: contiene las clases manejadoras para el control de vistas y funciones del sistema.

Paquete entities: contiene las entidades que se usan en el proyecto. Cada entidad contiene los datos necesarios para representar a cada usuario y de la glucosa.

Paquete model: contiene las clases generales para desarrollo en el proyecto. Entre ellas encontramos las clases que permiten generar los JSON, las gráficas, envío de correos, los servlets de control de la aplicación Android, entre otras.

En el Java Doc del proyecto puedes ampliar información sobre los métodos y clases de la aplicación.

MODO DESARROLLO PARA PRUEBAS EN NETBEANS

1. Cargue el "ScriptSqlDatabase.sql" al servidor de base de datos.
2. Abra el archivo de proyecto "Glucometrias" en netbeans.
3. En el archivo web.wxml cambie la variable "javax.faces.PROJECT_STAGE" de "Production" a "Development"; esto con el fin de visualizar las excepciones.
4. En la clase "Dao.DAO.java" cambie las variables "userDb" y "passDb" por las credenciales que usa en su base de datos MySQL. el puerto usado en la base de datos es el 3306. si ha cambiado su puerto, modifique la dirección de la base de datos en el metodo "conectar" de la clase.
5. En la clase "model.SendMail.java" modificar las variables "sender" y "senderPass" por sus correspondientes, de acuerdo a la cuenta de correo que use como administrador en gmail. Para evitar errores en la mensajería se debe habilitar el uso del correo para aplicaciones externas a las de google. Para ello debe loguearse en el navegador con su cuenta de gmail, luego acceder a la siguiente dirección "<https://www.google.com/settings/security/lesssecureapps>" y activar Acceso de aplicaciones menos seguras; también acceder a la siguiente dirección "<https://accounts.google.com/b/0/DisplayUnlockCaptcha>" y Permitir el acceso a tu cuenta de Google.
6. Hacer un build al proyecto para descargar las librerías desde los repositorios Maven. Para ello, en el proyecto netbeans darle "Build project" o F11.
7. Correr el proyecto desde de netbeans para visualizarlo. Para ello darle "Run project" o F6.

CONSIDERACIONES EN EL PROYECTO WEB

Se usa la cuenta de Gmail con usuario : glucometriasgtst@gmail.com y contraseña: [gtst2525](#) para el envío de mensajes de correo.

En el google drive de la cuenta de Gmail se agregó el APK de la aplicación móvil donde se puede descargar e instalar en los dispositivos:

APK ANDROID:

<https://drive.google.com/file/d/0By1VjmcVY702RGF3UE40LUxpT0k/view?usp=sharing>

APLICACIÓN ANDROID

GlucUp-2Date es una aplicación móvil que permite a los usuarios realizar funciones para el manejo de la información registrada en el sistema.

- **Usuario personal:** La aplicación le muestra todos los pacientes relacionados o le permite buscar un paciente. Puede registrar medidas de glucosa del paciente y ver el historial de medidas en forma de tabla o en modo gráfico.
- **Usuario paciente:** Puede ver su información personal en la página web. La plataforma le permite registrar medidas de glucosa y ver el historial de medidas en forma de tabla o en modo gráfico.

Para mayor información sobre las funciones de GlucUp-2Date, puedes descargar el manual de usuario [Aqui](#).



Figura 5. Pantalla de bienvenida GlucUp-2Date

TECNOLOGIAS DE DESARROLLO

Software de desarrollo

Para el desarrollo del proyecto web usamos programas con licencia de software libre

- *Proyecto desarrollado en:*
Android Studio 2.2 (Build #AI-145.327617
JRE: 1.8.0_76-release-b03 amd64
JVM: OpenJDK 64-Bit Server VM by JetBrains s.r.o
- *Base de datos en:* Igual que la plataforma web
- *Servidor de aplicaciones:* Igual que la plataforma web

Librerías y Frameworks

```
dependencies {  
    compile fileTree(include: ['*.jar'], dir: 'libs')  
    compile 'com.android.support:appcompat-v7:24.2.1'  
    compile 'com.android.support:design:24.2.1'  
    compile 'com.android.support:recyclerview-v7:24.2.1'  
    compile 'com.google.code.gson:gson:2.4'  
    compile 'com.android.support:support-v4:24.2.1'  
    compile 'com.github.barteksc:android-pdf-viewer:2.1.0'  
    compile 'com.github.PhilJay:MPAndroidChart:v2.0.9'  
    compile 'com.android.support:cardview-v7:24.2.1'  
    compile 'com.code-troopers.betterpickers:library:3.0.0'  
    compile 'com.android.support.constraint:constraint-layout:1.0.0-alpha9'  
    compile 'com.google.firebase:firebase-messaging:9.6.1'  
    testCompile 'junit:junit:4.12'  
}
```

Figura 6. Listado de librerías

- *Support AppCompat v7:* Librería que permite el uso de material de diseño y manejo de barras de acción a través de patrones de diseño. Incluye el manejo de actividades para mostrarlas como diálogos *AppCompatActivity*.
- *Support Design:* Librería complementaria para AppCompat. Permite manejo sobre componentes especiales: Navigation Drawer, Floating Action Button, Snackbars y Tabs
- *Support RecyclerView:* Librería complementaria para AppCompat. Específica para hacer uso de la clase *RecyclerView* y manejar listas en vistas recicladas
- *Code GSON:* librería de clases de Google para el manejo JSON usando el lenguaje java.
- *Support Support v4:* Librería para hacer manejo de APIs para comunicación con servidores y manejo de excepciones.

- [Barteksc AndroidPdfViewer](#): Usada en la sección de ayuda para mostrar los manuales de usuarios en la misma aplicación.
- [PhliJay MPAndroidChart](#): Usada en la creación de las gráficas para la muestra de resultados. Gráficas de barras, lineales y juntas.
- [Support CardView](#): Librería complementaria para AppCompatActivity. Específica para hacer uso del elemento *CardView* para crear ítems en listas reciclables.
- [Code-troopers Betterpickers](#): Librería para crear diseños de selección de hora y fecha.
- [Support ConstraintLayout](#): Librería que permite el manejo del *Layout Constraint*, un nuevo Layout que permite crear pantallas responsivas a diferentes dispositivos.
- [Firebase Messaging](#): Librería para el manejo de notificaciones haciendo uso de FireBase Cloud Messaging.

ESTRUCTURA DEL PROYECTO ANDROID

El proyecto en Android Studio tiene la siguiente distribución:

- **App:** Se encuentra todos los archivos relacionados con el diseño de la aplicación.
- **Gradle Scripts:** Archivos de configuración en donde se encuentran las librerías usadas, la versión de la aplicación, la API principal y la versión de compilación.

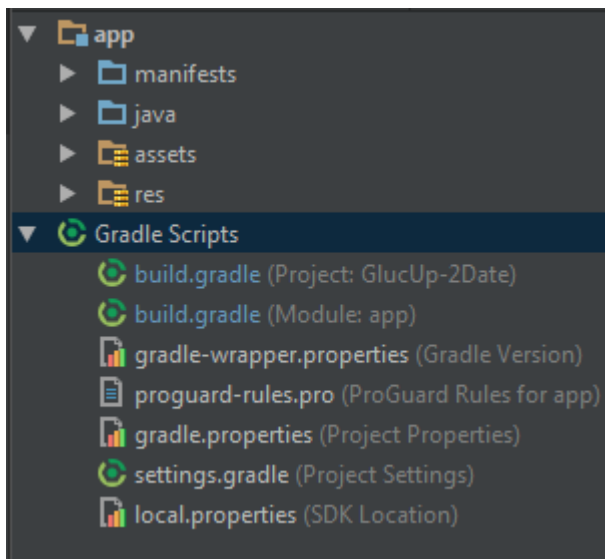


Figura 7. Estructura Android Studio

Carpeta App

- **Manifests:** Se encuentra el archivo *AndroidManifest.xml* que brinda información esencial acerca de la aplicación: Nombre del paquete JAVA para la compilación, descripción de actividades, servicios, proveedores y receptores
- **Java:** Se encuentran todas las clases, actividades, fragmentos y demás archivos usados en el desarrollo de la aplicación. Existen dos subcarpetas:
 - **Controler:** Están todos archivos que reciben las acciones del usuario, actividades, fragmentos, adaptadores.
 - **Model:** Archivos que se encargan de la comunicación entre la aplicación y el servidor. También existen clases para el manejo de datos
- **Assets:** Se encuentran los manuales de usuario que se integran en la compilación de la aplicación.
- **Res:** Están todos los recursos como imágenes, sonidos, ficheros y archivos xml que son usados por los archivos de la carpeta Java para crear la vista y entregar la información