## Nomenclatura y versionamiento

Los diagramas presentados a continuación han sido etiquetados mediante un identificador (DG0X), cabe mencionar que la versión se encuentra establecida en la tabla de revisión presentada posterior a cada diagrama.

DG01.- Diagrama Conceptual

Imagen que contiene mapa, texto

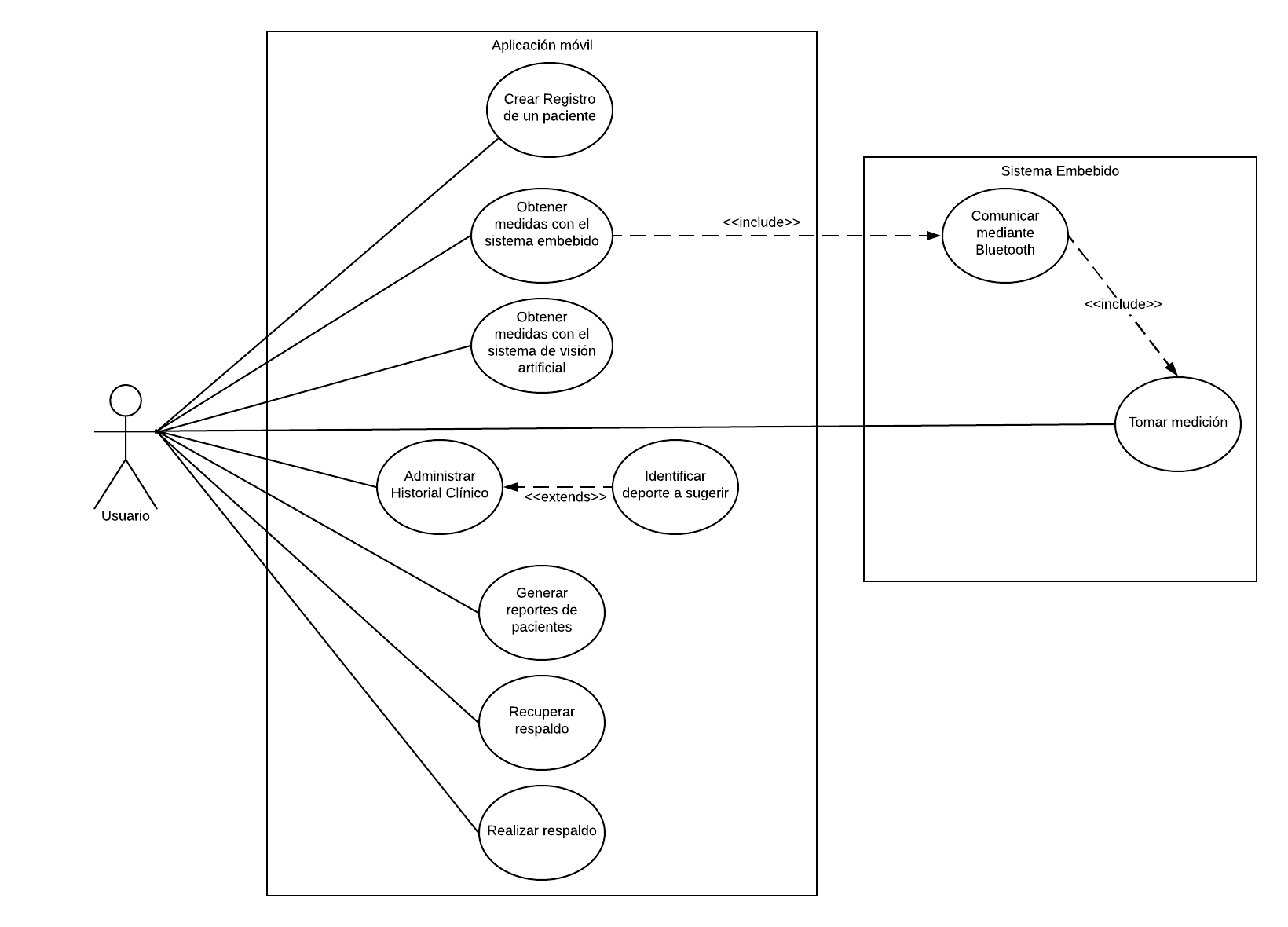
Descripción generada automáticamente

Descripción:

En este diagrama se puede observar la composición o división principal que el proyecto SiCMA contiene, estas ramificaciones presentan el sistema embebido y la aplicación móvil, donde cada uno de ellos contienen las funciones que realizaran.

|  |  |  |  |
| --- | --- | --- | --- |
| No. Revisión | Descripción | Fecha | Estatus |
| 01 | Versión inicial previa a la revisión del equipo de trabajo | 24/03/2020 | Rechazado |
| 02 | Versión posterior a la revisión del equipo de trabajo | 01/04/2020 | Aprobado |

Diagrama de Casos de Uso



*Fuente de notación:* [1]

***Escenarios de Casos de Uso***

CU.1 Crear registro del paciente

|  |  |
| --- | --- |
| Caso de Uso | CU.1 Crear registro del paciente |
| Fuentes | Cliente |
| Actor | Usuario |
| Descripción | Se creará un registro del paciente dando posibilidad a captura de datos |
| Flujo básico | 1. El usuario ingresa a la aplicación. 2. El usuario ingresa a la vista de registro de paciente. 3. El usuario llena el formulario de la información básica del paciente, dicha información es mostrada en Prototipo reporte historial clínico. 4. El usuario confirma el registro del paciente. |
| Flujos alternos | 1. Si el usuario ha registrado datos imprescindibles como vacíos no podrá crear el registro. |
| **Precondiciones** | N/A |
| **Postcondiciones** | 1. Se tiene un registro del paciente del que se ha guardado la información |
| Requerimientos trazados | N/A |

CU. 2 Obtener de medidas con el sistema embebido

|  |  |
| --- | --- |
| Caso de Uso | CU. 2 Obtener de medidas con el sistema embebido. |
| Fuentes | Cliente |
| Actor | Usuario |
| Descripción | Se inicia el proceso de obtención de las medidas de los pliegues cutáneos descritos en el perfil completo de ISAK |
| Flujo básico | 1. Se realiza el caso de uso del registro de una consulta. 2. Dentro de la actividad en que se registra una consulta se inicia el proceso de medición de los pliegues cutáneos, mostrados en Prototipo reporte historial clínico. |
| Flujos alternos | N/A |
| **Precondiciones** | 1. Es necesario que ya exista un registro de una consulta. |
| Post-condiciones | 1. Existirá un registro de las mediciones de los pliegues cutáneos. |
| Requerimientos trazados | RF3,RF4 |

.

CU.3 Obtener Medidas mediante el sistema de visión artificial

|  |  |
| --- | --- |
| Caso de Uso | CU.3 Obtener Medidas mediante el sistema de visión artificial |
| Fuentes | Cliente |
| Actor | Usuario |
| Descripción | Mediante procesamiento de imágenes obtener las medidas del cuerpo humano definidas en el perfil completo de la certificación ISAK |
| Flujo básico | 1. Crear un registro de consulta. 2. Buscar al paciente al que corresponde la consulta. 3. Iniciar el proceso de captura de imágenes. 4. Se calculan las medidas del cuerpo humano, las cuales se muestran en Prototipo reporte historial clínico.. |
| Flujos alternos | 1. Existe un bloqueo por parte de los permisos de la aplicación para usar la cámara. 2. No se pueden identificar los puntos de referencia por ende no se podrá concluir el proceso de obtención de medidas. |
| Pre-condiciones | 1. Los puntos de referencia han sido marcados para su identificación con el sistema de visión artificial. 2. El registro del paciente ya existe en la base de datos. |
| Post-condiciones | 1. Se tiene un registro de cada una de las mediciones definidas en el perfil completo de la certificación ISAK. |
| Requerimientos trazados | RF1, RF2 |

CU.4 Comunicar mediante bluetooth

|  |  |
| --- | --- |
| Caso de Uso | CU.4 Comunicar mediante bluetooth |
| Fuentes | Cliente |
| Actor | Usuario |
| Descripción | Se realiza la conexión bluetooth al sistema embebido para la medición de pliegues cutáneos |
| Flujo básico | 1. Se efectúa el Caso de uso 3 2. Se entabla conexión Bluetooth con el sistema embebido. 3. El sistema embebido y el dispositivo móvil quedan a la escucha de la acción para el registro de la medición. |
| Flujos alternos | 1. No han sido otorgados los permisos necesarios a la aplicación para comunicarse por Bluetooth. 2. No se puede entablar conexión con el sistema embebido. |
| Pre-condiciones | 1. Exista ya un registro de paciente y de consulta para intentar la conexión Bluetooth. |
| Post-condiciones | 1. Se entabla conexión mediante Bluetooth exitosamente |
| Requerimientos trazados | RF3,RF4 |

CU. 5 Tomar Medición

|  |  |
| --- | --- |
| Caso de Uso | CU. 5 Tomar Medición |
| Fuentes | Cliente |
| Actor | Usuario |
| Descripción | El usuario acciona el sistema embebido y este registra la medición del pliegue cutáneo. |
| Flujo básico | 1. Se efectúa el caso de uso 4 2. Se localiza el pliegue cutáneo que se va a medir. 3. Se mide con el plicómetro 4. Se acciona el sistema embebido para extraer la medición. |
| Flujos alternos | 1. Se pierde la conexión Bluetooth y es necesario reestablecerla |
| Pre-condiciones | 1. Haber efectuado el proceso del caso de uso 4 |
| Post-condiciones | 1. Las mediciones tomadas por el sistema embebido son guardadas en el dispositivo móvil. |
| Requerimientos trazados | RF3,RF4 |

CU. 6 Administrar historial clínico

|  |  |
| --- | --- |
| Caso de Uso | CU. 6 Administrar historial clínico |
| Fuentes | Cliente |
| Actor | Usuario |
| Descripción | Dentro de este caso de uso se dispone varias acciones que puede realizar el actor, como lo son crear y editar historiales clínicos. |
| Flujo básico | 1. Se efectúa el caso de uso 1. 2. Después de que se realiza el registro del paciente se da la posibilidad de la creación de un historial clínico. 3. Llenado del historial clínico, con la información mostrada en Prototipo reporte historial clínico. 4. Para editar: 5. Dentro del menú principal se accede a Editar Historial. 6. Se busca al paciente y se editan los datos necesarios y se procede a guardar. |
| Flujos alternos | N/A |
| Pre-condiciones | 1. Existe un registro de un paciente. |
| Post-condiciones | 1. Existe un registro del historial clínico del paciente. |
| Requerimientos trazados | RF9 |

CU. 7 Identificar deporte a sugerir

|  |  |
| --- | --- |
| Caso de Uso | CU. 7 Identificar deporte a sugerir |
| Fuentes | Cliente |
| Actor | Usuario |
| Descripción | En base a las mediciones e índices del paciente se podrá hacer una sugerencia de un deporte a sugerir. |
| Flujo básico | 1. A partir de las medias tomadas en la consulta se clasifica al paciente de acuerdo con los índices de detección del deporte por somatotipo y sumatoria de pliegues. 2. Se muestran los resultados |
| Flujos alternos | 1. Al menos una medida clasificatoria no existe, por ende, no es posible hacer la clasificación. |
| Pre-condiciones | 1. Existe un registro de una consulta 2. El historial clínico ligado al paciente debe ser de tipo deportivo |
| Post-condiciones | 1. Existirá un registro para el paciente con un deporte a sugerir. |
| Requerimientos trazados | RF7 |

CU. 8 Generar reporte del paciente

|  |  |
| --- | --- |
| Caso de Uso | CU. 8 Generar reporte del paciente |
| Fuentes | Cliente |
| Actor | Usuario |
| Descripción | Se podrá generar un reporte del paciente en el que se visualizará un estado del paciente. |
| Flujo básico | 1. Desde el menú principal se accede al apartado de Generar reporte 2. El usuario busca en la lista de pacientes al paciente en específico del que se quiera genera el reporte. 3. Se seleccionan las consultas que se quieran añadir al reporte. 4. Se genera un documento PDF tal y como se muestra en Prototipo reporte antropométrico. |
| Flujos alternos | 1. No existen los datos suficientes para la realización del reporte, por ende, no se puede generar el reporte. |
| Pre-condiciones | 1. Existe un paciente al que se pueda generar un reporte 2. Existe un paciente con registros de consultas con las que se generará el reporte |
| Post-condiciones | 1. Existirá un reporte en el cual se vería comparaciones entre consultas. |
| Requerimientos trazados | RF6 |

CU. 9 Recuperar respaldo

|  |  |
| --- | --- |
| **Caso de Uso** | CU. 9 Recuperar respaldo |
| **Fuentes** | Cliente |
| **Actor** | Usuario, Google Drive |
| **Descripción** | El usuario conecta una cuenta de Google que tiene almacenado una copia de respaldo de la base de datos de la aplicación y la restaura en el dispositivo móvil. |
| **Flujo básico** | 1. Desde el Menú principal ir al apartado de Respaldo/Restauración. 2. Ingresar el correo electrónico de Google del que se desea recuperar el respaldo. |
| **Flujos alternos** | 1. La cuenta que se ingresa es errónea |
| **Pre-condiciones** | 1. La cuenta que se ingresa tenga una copia de seguridad que restaurar. |
| **Post-condiciones** | 1. La información que se encontraba en la nube ahora tiene una copia en el dispositivo móvil. |
| **Requerimientos trazados** | NRF8 |

CU. 10 Realizar respaldo

|  |  |
| --- | --- |
| **Caso de Uso** | CU. 10 Realizar respaldo |
| **Fuentes** | Cliente |
| **Actor** | Usuario, Google Drive |
| **Descripción** | El usuario conecta una cuenta de Google que para almacenar una copia de respaldo de la base de datos de la aplicación en la nube. |
| **Flujo básico** | 1. Desde el Menú principal ir al apartado de Respaldo/Restauración. 2. Ingresar el correo electrónico de Google en el que se desea guardar una copia de seguridad. |
| **Flujos alternos** | 1. La cuenta que se ingresa es errónea. |
| **Pre-condiciones** | Se tiene una cuenta de Google. |
| **Post-condiciones** | La información que se encontraba en el dispositivo móvil ahora tiene una copia en la nube. |
| **Requerimientos trazados** | NRF8 |

|  |  |  |  |
| --- | --- | --- | --- |
| No. Revisión | Descripción | Fecha | Estatus |
| 01 | Versión inicial previa a la revisión del ET | 24/03/2020 | Rechazado |
| 02 | Versión posterior a la revisión del ET | 25/04/2020 | Rechazado |
| 03 | Versión posterior a la 2da revisión del ET | 25/04/2020 | Pendiente a aprobación |

DG08.-Diagrama de Árbol

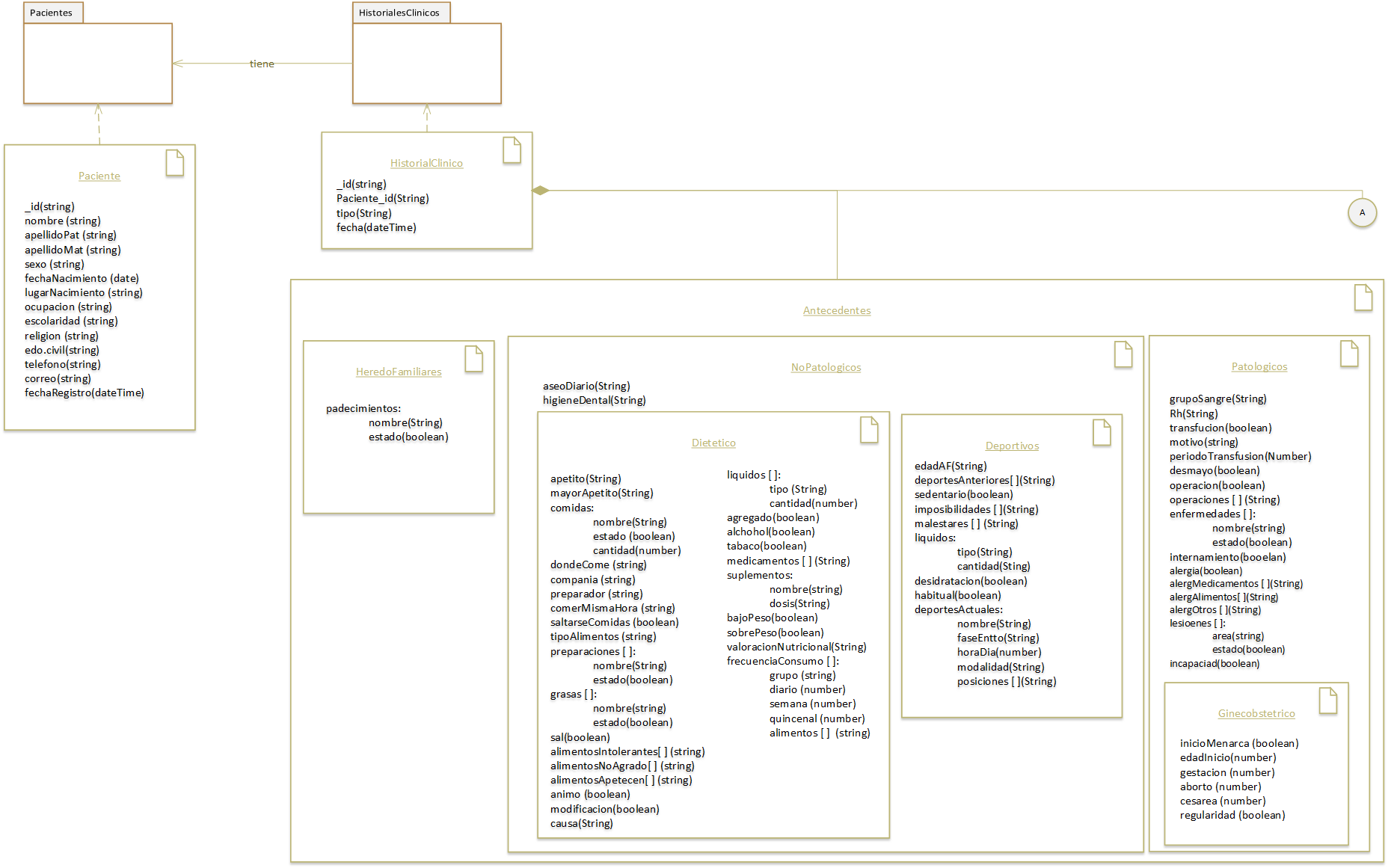


Descripción:

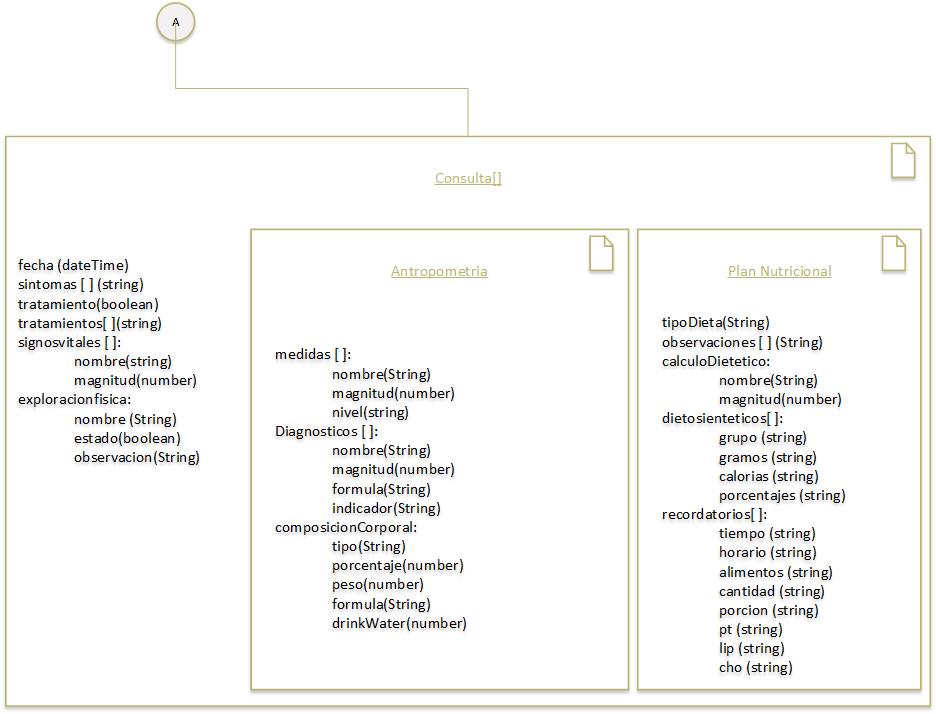
El diagrama de árbol presenta el agrupamiento de los datos que serán almacenados en la base de datos, el motivo de la realización de este diagrama fue para facilitar la visualización y distribución, de los atributos que contendría cada sección, siendo secciones las representadas dentro de un ovalo y sus atributos las ramificaciones. *Fuente de notación:* [2]

|  |  |  |  |
| --- | --- | --- | --- |
| No. Revisión | Descripción | Fecha | Estatus |
| 01 | Versión inicial previa a la revisión del equipo de trabajo | 24/03/2020 | Aprobado |

 DG09.-Diagrama de documentos 1/2



DG09.-Diagrama de documentos 2/2



Descripción:

El diagrama anterior presenta la distribución de los datos que seran almacenados en la base de datos del sistema, como se menciono con anterioridad es un modelo no relacional basado en documentos, teniendo dos colecciones, pacientes: donde se almacenaran los datos generales y de contacto e historial clinico: la sección más extensa del docuemento, presentando docuementos embebidos que son la representación del seccionamiento visto anteirormente en el diagrama de arbol. Este diagrama contiende una tabla descriptiva de cada uno de las figuras utilizadas con el fin de mejorar la conprension del mismo. *Fuente de notación:* [2] *y* [3]

|  |  |  |  |
| --- | --- | --- | --- |
| No. Revisión | Descripción | Fecha | Estatus |
| 01 | Versión inicial previa a la revisión del ET | 28/04/2020 | Aprobado |
| 02 | Versión posterior al análisis de datos. | XX/XX/2020 |  |

DG04.-Diagrama de Clases

Descripción:

El sistema hará uso de clases de control como lo son *Main, DBController, DriveController y EmbebidoController*, para el control total de la aplicación, para la comunicación con la base de datos, la comunicación con el servicio de Google Drive y comunicación con el sistema embebido respectivamente. *HistorialClinico, Consulta, Paciente, Reporte, Somatocarta, Formulario y Elemento* son clases entidad, las cuales están principalmente relacionadas con los datos, mientras que *SistemaReconociemiento* será la clase enfocada en el procesamiento de las imágenes y finalmente *SistemaVision* la interfaz de interacción con la cámara. *Fuente de notación:*[1]

|  |  |  |  |
| --- | --- | --- | --- |
| No. Revisión | Descripción | Fecha | Estatus |
| 01 | Versión inicial previa a la revisión del equipo de trabajo | 30/04/2020 | Pendiente a aprobación |

***DG06.-Diagrama de Secuencia***

Diagrama de Secuencia: Interacción "General”



Descripción:

Diagrama de secuencia para el escenario inicial de la aplicación. A la izquierda se ubica la clase *Main* que será el control total de la aplicación, proporcionando desde la interfaz hasta el orden de cada proceso. El mensaje *iniciarAplicacion()* es la representación del comienzo de ejecución de la aplicación, la cual se verá efectuada por la interacción del usuario a la clase *Main* .

Una vez iniciada la aplicación esta deberá conectar con la base de datos, en caso de no existir la base, se creará una nueva (principalmente cuando la aplicación este recién instalada en el dispositivo), para ello *Main* envía el mensaje *conectarBD() o crearDB()* a la clase *DBController*, quien a su vez realizara el proceso necesario para conectar o crear -según sea el caso- con la base de datos local. La siguiente actividad se ve modelada en el diagrama de secuencia Menu. Finalmente, las líneas de vida de las clases visualizadas en este diagrama se verán concluidas al cerrar la ejecución de la aplicación. *Fuente de notación:*[4]*,* [1] *y* [5]

Diagrama de Secuencia: Interacción "Menú”



Descripción:

Diagrama de secuencia para la presentación y ejecución del menú de la aplicación. Para simplificar el diagrama se utilizó la referencia de diagramas, Registro de Paciente, Registro de Consulta, Generación de Reporte, Respaldo/Restauración, Buscar Historial y Editar Historial, cada uno de estos opcionales a iniciar su ejecución por activación del usuario. *Fuente de notación:*[4]*,* [1] *y* [5]

Diagrama de Secuencia: Interacción "Registro Paciente”



Descripción:

Diagrama de secuencia para el registro de un paciente. Iniciando con la solicitud al *Main* de *nuevoPaciente()* por parte del usuario. El *Main* mostrará en la interfaz gráfica el formulario necesario para la creación de paciente, el parámetro editable será “verdadero” dado que es necesario el llenado de los datos; una vez concluido este paso y realizada la solicitud de guardado*, Main* realizará la creación del objeto *p1* de tipo *Paciente* con la información del formulario anteriormente mencionado. Ya creado este objeto Main solicita *agregarPaciente()* enviando como parámetro al objeto *p1, DBController* se encargara de realizar la creación de un paciente en la base de datos. Para concluir el proceso *Main* notifica al usuario. La actividad siguiente se ve referenciada en el diagrama de secuencia Crear historial. Es importante mencionar que ningún objeto será destruido dado que podrá ser utilizado en secuencias posteriores. *Fuente de notación:*[4]*,* [1] *y* [5]

Diagrama de Secuencia: Interacción "Crear Historial”



Descripción:

Diagrama de secuencia para la creación de historial. El *Main* mostrará en la interfaz gráfica el formulario necesario para la creación de historial, el parámetro editable será “verdadero” dado que es necesario el llenado de los datos; una vez concluido este paso y realizada la solicitud de guardado*, Main* realizará la creación del objeto *h1* de tipo *HistorialClinico* con la información del formulario anteriormente mencionado. Ya creado este objeto *Main* solicita *setHistorial()* enviando como parámetro al objeto *p1 – el cual ya debió de ser creado con anterioridad-*. El proceso *Main* envía el mensaje agregarHistorial() con p1 como parámetro, esta clase se encargará del proceso agregación en la base. H1 deberá ser destruido una vez obtenida la respuesta del *DBController* e informar al usuario el fin del proceso*.*

*Fuente de notación:*[4]*,* [1] *y* [5]

Diagrama de Secuencia: Interacción "Registro Consulta”



Descripción:

Diagrama de secuencia para el registro de consulta. Una vez solicitado *nuevaConsulta()* por el usuario, *Main* creará *c1* objeto *de Consulta.* Ahora es necesario tener un objeto Paciente para continuar con la secuencia, es por ello por lo que se reutilizara la secuencia “registro de paciente” en caso de no existir en el sistema o “buscar paciente” para cuando sea existente. Con un p1 ya existente en el sistema se procede a asignar la c1 a p1 por medio del mensaje *asignarConsulta().* Posterior a esta relación, *Main* muestra el formulario necesario para el llenado de información referente a una consulta. Similar a procesos anteriores una vez solicitado *guardar(), c1* cambia los nuevos valores con la información figurada en el formulario. Las siguientes actividades serán la secuencia Foto medidas y Medidas embebido,una vez finalizadas *Main* envía *generarResultados() a c1* para concluir esta solicitud por parte del usuario *Main* muestra los resultados por medio de la interfaz. Aceptando esto el usuario envía guardar(), mensaje de activación para solicitar a *DBController* guardar esta consulta, enviando *p1* para tener toda la información necesaria. Concluiremos con la notificación al usuario. Como secuencia opcional se encentra el diagrama generar reporte. *Fuente de notación:*[4]*,* [1] *y* [5]

Diagrama de Secuencia: Interacción "Buscar Paciente”



Descripción:

Diagrama de secuencia para la búsqueda de paciente. *Main* solicita *getPacientes()* a *DBController*, este último procede a solicitar a la base de datos todos los pacientes que existen, retornando a esta búsqueda el identificador y nombre de paciente, con esta información *Main* mostrará al usuario por medio de la interfaz. Una vez seleccionado el paciente por parte del usuario *Main* solicita a *DBController* el resto de información necesaria para el proces, con esto ultimo *Main* crea *p1* objeto de tipo *Paciente*. *Fuente de notación:*[4]*,* [1] *y* [5]

Diagrama de Secuencia: Interacción "Buscar Historial”



Descripción:

Diagrama de secuencia para la búsqueda de historial. Para la ejecución de esta secuencia será necesario tener un objeto *p1* de tipo Paciente inicializado y con la información necesaria, es por ello por lo que se hará uso de la secuencia Búsqueda de Paciente. Dado que *p1* existe *Main* solicita a *DBController* el historial de *p1* mediante getHistorial(), como en casos anteriores esta ultima clase consulta en la base el historial y retorna la información a *Main*, siguiente paso es crear el objeto *h1* de tipo *Historial*Clinico con la información obtenida de *DBController*. Finalizamos la secuencia mostrando los datos contenidos en h1 mediante un formulario y editable “falso” dado que únicamente se mostrará la información. *Fuente de notación:*[4]*,* [1] *y* [5]

Diagrama de Secuencia: Interacción "Editar Historial”



Descripción:

Diagrama de secuencia para la edición de historial. Tal y como muestra el *diagrama Menu* para la ejecución de este diagrama es necesario tener un objeto *h1* previamente inicializado. Previniendo una cancelación del proceso se creará un objeto *h2* como copia exacta del *h1*, efectuando un respaldo. Posterior a esta acción se muestra un formulario precargado con la información de h1, permitiendo así al usuario editar los cambios que desee realizar. Como se mencionó existirá la alternativa de *guardar()* , en este caso *Main* envía el mensaje *setHistorial()* a *h1* para guardar los cambios efectuados en el formulario. Hecho esto *Main* solicita a *DBController editarHistorial()* enviando como parámetro p1. En el caso de *cancelar()* únicamente h1 asumirá los valores de *h2*, y este último será destruido. Para concluir se notifica al usuario y se muestra el formulario con *h1* a fin de mostrar cambios. *Fuente de notación:*[4]*,* [1] *y* [5]

Diagrama de Secuencia: Interacción "FotoMedidas”



Descripción:

Diagrama de secuencia para la toma de Foto Medidas. Una vez iniciada la secuencia *Main* crea v1 objeto de la clase *VisionArtificial(),* este objeto permitirá que la interfaz de usuario utilice la cámara del dispositivo. De igual manera *Main* creará r1 objeto de clase *SistemaReconocimiento().*

Por cada posición el usuario podrá *tomarFoto()*, mensaje solicitado por usuario hacia la interfaz que *Main* controla, el siguiente mensaje será *capturar()* enviado a *v1*. Debido a este último paso se creará la *im1* objeto de tipo *Imagen.* Posteriormente *Main* solicita los datos referentes a las medidas al *r1*, enviando como parámetros la im1 y la posición a analizar. Una vez obtenidas estas medidas serán mostradas al usuario, de ser aceptadas este solicita el guardado, con lo que Main envía *guardarMedias()* a *c1*. El resto de los objetos a excepción de *Main* y *Consulta* serán destruidos. *Fuente de notación:*[4]*,* [1] *y* [5]

Diagrama de Secuencia: Interacción "Medidas Embebido”



Descripción:

Diagrama de secuencia para la toma de Medidas desde Sistema Embebido. Una vez iniciada la secuencia, *Main* crea *e1* de clase *EmbebidoController. A* su vez *e1* como parte de su creación conecta con el sistema embebido. La siguiente acción consiste en mostrar el formulario referente a la toma de medidas en pligues, de esta manera el sistema embebido funcionara como sistema de entrada para el llenado de esta sección.

Por cada medida a tomar *Main* solicita a *EmbebidoController recibirMedida()*, este en comunicación con el embebido recibirá la información requerida y se vera visualizada en la interfaz. Ya habiendo realizando la toma de todas las medidas el usuario envia guardar(), con ello *Main* utiliza esta información para guardarla en *c1*. *Fuente de notación:*[4]*,* [1] *y* [5]

Diagrama de Secuencia: Interacción "Generación Reporte”



Descripción:

Diagrama de secuencia para la generación de reporte. La activación de esta secuencia se genera por parte de solicitud del paciente proporcionando el parámetro proyección, referente al periodo de evaluación en el que se basara el reporte. En caso de no tener un objeto Paciente asignado a p1 se buscará uno. *Main* se comunica con *DBController* para pedir las consultas de *p1*, por cada consulta encontrada -teniendo proyección como condición de paro- crea una consulta con los datos obtenidos y cada una de ellas serán asignadas al objeto *p1*.

Habiendo concluido la recopilación de consultas, *Main* crea *r1*, objeto de tipo *Reporte*, al momento de su ejecución este auto envía los mensajes *calcularPorcentajes()* y *generarSomatocarta(),* lo cuales funcionan como activadores a procesos internos del objeto. Finalizando con la presentación del reporte al usuario por medio de la interfaz de usuario. *Fuente de notación:*[4]*,* [1] *y* [5]

Diagrama de Secuencia: Interacción "Respaldo”



Descripción:

Diagrama de secuencia para el respaldo de información. *Main* deberá conectar con los servicios de GoogleDrive por medio de la clase GoogleDriveBackup, teniendo una conexión exitosa la clase *Main* solicita todos los documentos almacenados en la base de datos, una vez recuperados se enviara el mensaje *guardarDocs()* con los documentos recuperados como parámetro. Finalmente la clase *GoogleDriveBackup* se encargara de almacenar dichos documentos. *Fuente de notación:*[4]*,* [1] *y* [5]

Diagrama de Secuencia: Interacción "Restauración”

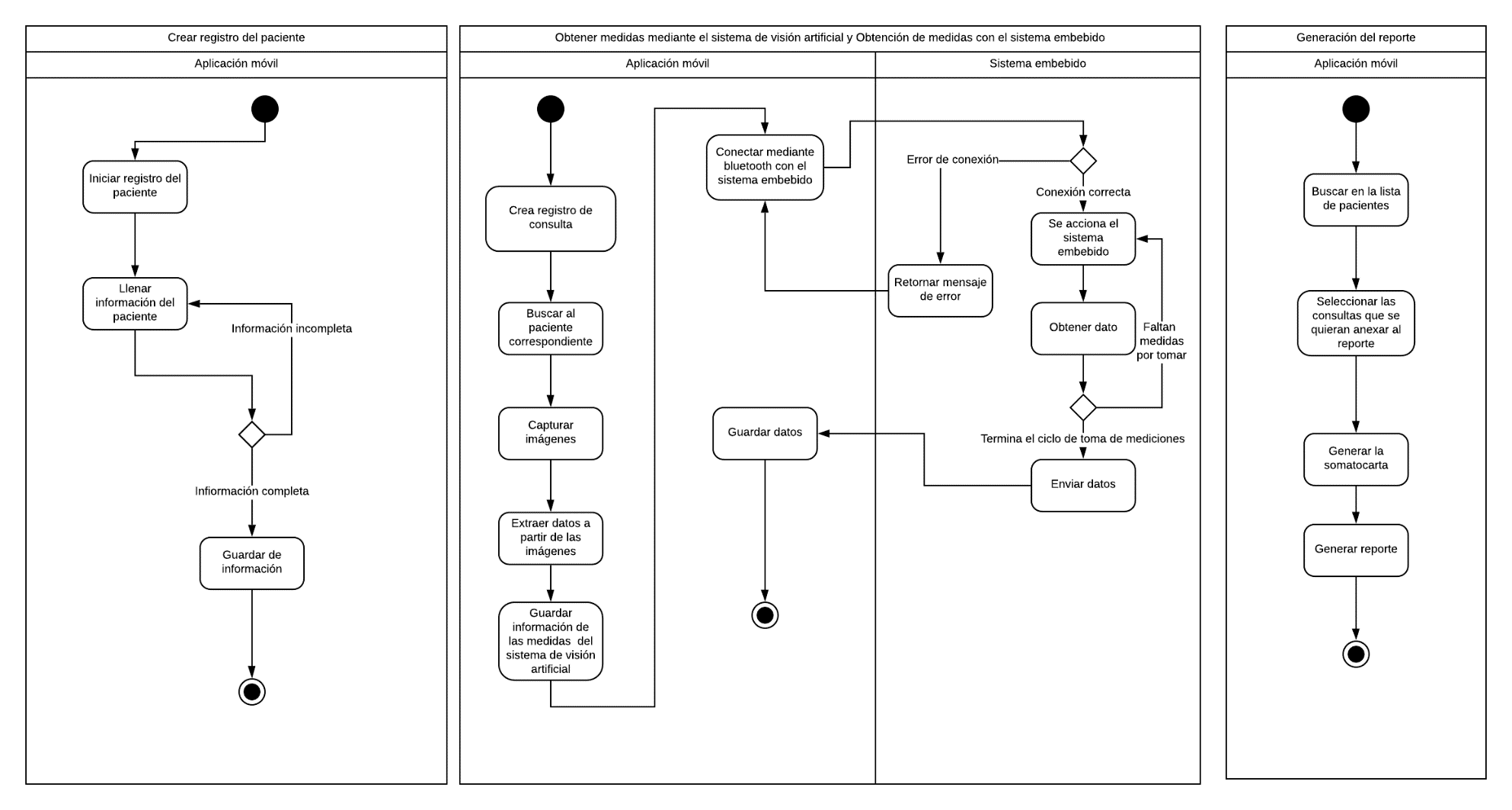


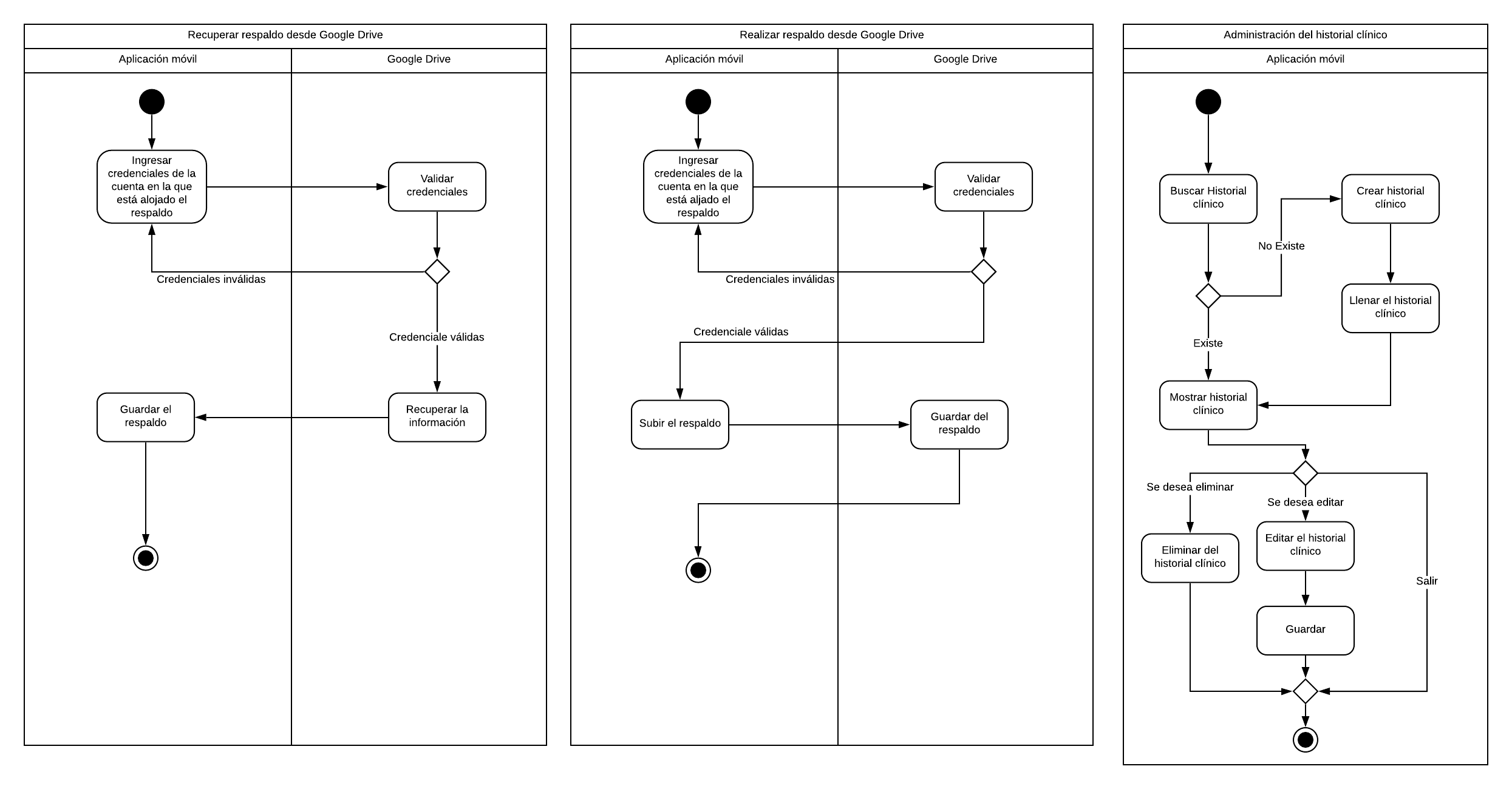
Descripción:

Diagrama de secuencia para la restauración de información. Funcionando de manera inversa a la secuencia Respaldo, iniciando con la conexión a los servicios de GoogleDrive por medio de *GoogleDriveBackup,* una vez realizada la conexión *Main* solicita los documentos almacenados por medio de *getDocs().* Una vez obtenidos los documentos estos serán enviados a *DBController* para que se encargue de almacenarlo en la base de datos, para ello esta clase realiza un borrado de la información en la base y la restaura con los documentos antes obtenidos. *Fuente de notación:*[4]*,* [1] *y* [5]

|  |  |  |  |
| --- | --- | --- | --- |
| No. Revisión | Descripción | Fecha | Estatus |
| 01 | Versión inicial previa a la revisión del equipo de trabajo | 28/04/2020 | Aprobado |

DG07.-Diagrama de Actividad





Descripción:

Se definen las actividades de los diagramas haciendo referencia a los procesos encontrados dentro de los casos de usos posibles establecidos en la descripción del software, tomando en cuenta la funcionalidad del sistema y la representación de los procesos en los distintos componentes del sistema.

**Crear registro del paciente**: Dentro de la aplicación se puede iniciar el proceso de registro del paciente, en donde se llena la información del paciente, en caso de que falte información del paciente que sea indispensable, el proceso retornará al llenado de información, de no ser así, el proceso termina con el guardado de la información.

**Obtener medidas mediante el sistema de visión artificial y Obtención de medidas con el sistema embebido**: Cuando se inicia el proceso de obtención de medidas estas deben de estar relacionadas con una consulta, entonces para iniciar este proceso se crea un registro de la consulta, después de busca un paciente ya registrado al que con el que estará relacionada esa consulta, se procede a realizar el proceso de captura de imágenes, se extraen los datos de las imágenes y se hace el guardado de información de las medidas del sistema de visión artificial para continuar con el sistema embebido, en donde se realiza la conexión por medio de protocolo bluetooth al sistema embebido, si existe algún error de conexión se retorna a en donde puede establecerse la conexión, en caso contrario el sistema embebido estará a la espera de que se accione para obtener el dato, se repite el ciclo para los datos faltante y una vez terminado, los datos son enviados a la aplicación móvil.

**Generación del reporte**: Para el proceso de generación del reporte, se busca un paciente que haya tenido registro de al menos una consulta, después se procede a agregar las consultas que se quieran incluir en el reporte, se genera la somatocarta para el reporte y se genera el reporte en formato PDF.

**Recuperar respaldo desde Google Drive**: Para la realización de esta actividad, el proceso requiere ingresar las credenciales de una cuenta de Google que ya tenga almacenado un respaldo en Drive, la parte de Google Drive se encarga de la validación de las credenciales y de ser correctas éste retornará el respaldo almacenado para su respectivo guardado en la aplicación móvil

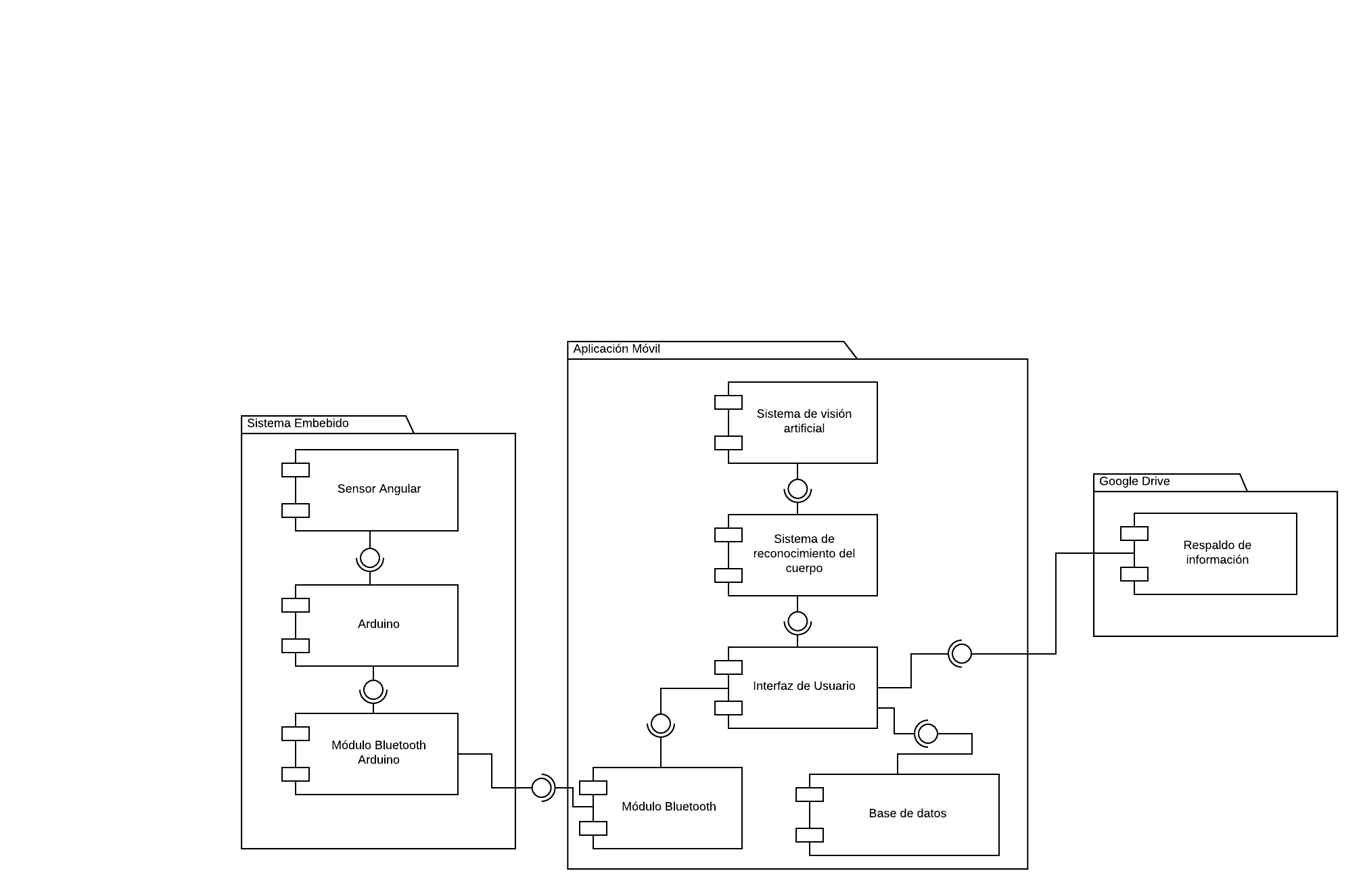
**Realizar respaldo desde Google Drive**: Igualmente como la anterior actividad, el proceso, requiere ingresar credenciales de una cuenta de Google para el guardado de una copia de restauración del estado actual de la aplicación, se sube el respaldo y regresa a la aplicación móvil.

**Administración del historial clínico**: Se busca el Historial clínico que se quiera administrar, en el caso de que no se encuentre puede crearse uno donde se tiene que llenar la información indispensable para su guardado, en el caso de que exista o ya se haya llenado un nuevo historial clínico, se muestra. Llegado a este punto del proceso, se puede optar por distintas opciones, editar el historial clínico en donde se guardarán los datos nuevamente capturados, se puede eliminar, en caso de que ya no se quiera conservar en la aplicación, o si simplemente se quería mostrar el historial, puede salir sin efectuar ningún cambio.

*Fuente de notación:*[1]

|  |  |  |  |
| --- | --- | --- | --- |
| No. Revisión | Descripción | Fecha | Estatus |
| 01 | Versión inicial previa a la revisión del equipo de trabajo |  | Aprobado |

DG05.-Diagrama de Componentes



Descripción:

Se estima que SiCMA para su correcto funcionamiento debe de estar definido por sus componentes y subcomponentes que serán de utilidad para llevar a cabo todas las acciones previstas en la fase de análisis del proyecto.

Los componentes principales son:

|  |  |
| --- | --- |
| Componente | Descripción |
| Sistema Embebido | Un sistema embebido implementado en un plicómetro que contará con un sensor angular y un módulo de comunicación bluetooth para la medición de los pliegues cutáneos. |
| Aplicación Móvil | La aplicación móvil en donde se implementará la interfaz de usuario para la gestión de información, la cual contará con una base de datos local en el dispositivo, además se encontrará implementado el sistema de reconocimiento del cuerpo alimentado por el sistema de visión artificial. |
| Google Drive | Se utilizará Google drive para la realización de respaldos a la nube y restauraciones desde la nube. |

*Fuente de notación:*[5]

|  |  |  |  |
| --- | --- | --- | --- |
| No. Revisión | Descripción | Fecha | Estatus |
| 01 | Versión inicial previa a la revisión del equipo de trabajo | 24/03/2020 | Pendiente aprobación |

DG06.-Diagrama de Despliegue



Descripción:

El presente diagrama presenta los nodos complementarios del sistema, tal y como se había representado en el diagrama de componentes, aquí podemos observar dichos componentes, prestando principal atención en los ubicados en el nodo Aplicación, dado que será aquí donde se deberá monitorear y tener presente durante el proceso de despliegue. Tal es el caso del componente base de datos, el cual al momento del despliegue deberá contener o proveer el artefacto referente a la definición del esquema de base de datos. Además, se presenta los protocolos por los cuales estos nodos presentaran una comunicación.

*Fuente de notación:* [4]

|  |  |  |  |
| --- | --- | --- | --- |
| No. Revisión | Descripción | Fecha | Estatus |
| 01 | Versión inicial previa a la revisión del equipo de trabajo | 11/05/2020 | Pendiente aprobación |

DG10.- Diagrama de flujo



Descripción:

El diagrama presentado dicta el flujo del algoritmo que se desarrollará para el sistema embebido, esté diagrama considera las señales entrantes como son los botones de apagado, reset y activación, así como la señal obtenida por el pin de conexión con el sensor angular, de igual manera el modulo de blueetooth es considerado, teniendo un proceso constante de cuestionamiento si la señal con el dispositivo móvil continua activa.

*Fuente de notación:*[4]

|  |  |  |  |
| --- | --- | --- | --- |
| No. Revisión | Descripción | Fecha | Estatus |
| 01 | Versión inicial previa a la revisión del equipo de trabajo | XX/XX/XXXX | Aprobado |

DG12 Diagrama de flujo Interacción de vistas

Imagen que contiene texto, mapa

Descripción generada automáticamente

Descripción:

Este diagrama presenta el flujo de interacción para el entendimiento del prototipado, dado que el prototipo no funcional fue presentado mediante la herramienta Adobe Xd, proporcionando una funcionalidad de secuencia.

*Fuente de notación:*[4]

|  |  |  |  |
| --- | --- | --- | --- |
| No. Revisión | Descripción | Fecha | Estatus |
| 01 | Versión inicial previa a la revisión del equipo de trabajo | XX/XX/XXXX | Aprobado |

Referencias

[1] K. Kendall y J. Kendall, *Análisis y Diseños de Sistemas 8va Edición - ISBN: 9786073205771*. 2011 [Online]. Disponible en: http://cotana.informatica.edu.bo/downloads/ld-Analisis y Diseno de Sistemas\_Kendall-8va.pdf

[2] A. Hernández Chillón, S. Feliciano Morales, J. García Molina, y D. Sevilla Ruiz, “Visualización de esquemas en bases de datos NoSQL basadas en documentos”, *Actas las 22nd Jornadas Ing. del Softw. y Bases Datos, JISBD 2017*, núm. July, 2017.

[3] J. P. Poveda Galvis, “Propuesta de notación gráfica para el modelo orientado a documentos de MongoDB”, *J. Chem. Inf. Model.*, p. 96, 2015.

[4] G. Booch, J. Rumbaugh, y I. Jacobson, “Lenguaje Unificado de Modelado Guía del Usuario.” p. 276, 2006.

[5] P. López, “Ingeniería Del Software I”, *Ing. Del Softw. I*, pp. 12–15, 2012.