

Sistema de Gestión de la Información de Registros de Variabilidad de Distensión Arterial

Fink, Sebastián Alberto¹; Jaime, Juan Manuel¹; Passoni, Lucía Isabel²
; Clara, Fernando Mario²

¹Facultad de Ingeniería. Universidad FASTA, Mar del Plata, Argentina

²Laboratorio de Bioingeniería. Facultad de Ingeniería.
Universidad Nacional de Mar del Plata, Argentina

Resumen:

En este trabajo se presenta el diseño y desarrollo de un Sistema de Gestión de la información de registros de Variabilidad de Distensión Arterial (VDA). El sistema facilitará el procesamiento, análisis y almacenamiento de señales de VDA adquiridas por un equipo desarrollado en el Laboratorio de Bioingeniería. El análisis de la onda de pulso tiende a convertirse en una nueva herramienta para la evaluación del estado del sistema circulatorio. Las variaciones de distensión de la arteria son detectadas de forma no invasiva en la zona de palpación de la arteria radial obteniéndose la onda de pulso. Se ha probado que la misma se modifica con la edad en individuos sanos. La obtención de indicadores de cambios progresivos en su morfología permite evaluar el grado de elasticidad o de envejecimiento del árbol arterial. El sistema de gestión de la información propuesto tiene como finalidad generar un repositorio permanente de los registros de VDA y sus análisis correspondientes, con acceso para su consulta vía web. Este sistema asistirá a los investigadores en los procesos de análisis, almacenamiento y acceso remoto, brindando un marco de trabajo adecuado para el del equipo en un entorno amigable y seguro.

Palabras Clave

Gestión de la Información, Variabilidad de Distensión Arterial, Onda de Pulso, Procesamiento de Señales.

Introducción

Esta propuesta tiene como finalidad el desarrollo de un Sistema de Gestión de la Información generada por el uso de un sistema de adquisición de señales biomédicas desarrollado en el Laboratorio de Bioingeniería de la Facultad de Ingeniería de la Universidad Nacional de Mar del Plata (UNMDP) [1]. Este sistema que adquiere y procesa las señales de pulsos de la arteria radial ha sido diseñado específicamente para investigaciones en el área de medicina cardiovascular. El mismo, único en su tipo en función del principio de funcionamiento y la utilidad que brinda, es utilizado para adquirir y almacenar señales bioeléctricas generada por la variabilidad de la distensión de la arteria radial (VDA). Esta información es de interés para la evaluación de patologías asintomáticas (Hipertensión, Diabetes) como así también del deterioro arterial prematuro [2].

La información colectada por el sistema de adquisición debe ser almacenada y posteriormente procesada para la generación de indicadores del grado de deterioro del sistema arterial. Si bien se han diseñado y publicado diversos algoritmos para el procesamiento de esta información [3][4][5], existe una carencia significativa en la organización y automatización de los procesos. En la actualidad la aplicación del protocolo de adquisición de señales ha generado una importante cantidad de registros (más de 1000) y en consecuencia se ha considerado oportuno y necesario el desarrollo

de un Sistema de Gestión de Señales de Variabilidad de Distensión Arterial (SisGeVDA).

El objetivo principal del SisGeVDA es brindar un entorno adecuado de trabajo a los investigadores del área biomédica. Su uso permite la organización de la colección de registros, como así también brinda un soporte para el procesamiento de las señales y la organización de los resultados.

El SisGeVDA generará una base de datos donde se almacenarán las señales adquiridas por el equipo, la información relacionada con el paciente, con el lugar y el profesional que lleva adelante el protocolo, como así también los resultados del procesamiento de los registros. Proveerá una interfaz gráfica para el uso de herramientas de procesamiento digital de señales a ser aplicados en la etapa de procesamiento de los registros, esta función dispondrá de un buen nivel de interactividad con el usuario, facilitando la obtención de los parámetros característicos de la señal de VDA.

En base a los parámetros y la información del paciente, se generará un informe que le brinde herramientas al especialista para el diagnóstico del estado arterial. El sistema tendrá la capacidad de generar reportes de casos individuales, como así también podrá reportar resultados históricos, característica sumamente útil para evaluar evolución de cada paciente como resultado de tratamientos.

El SisGeVDA generará un repositorio de datos permanente, con capacidad de proveer reportes estadísticos de la población de registros de VDA, previéndose como insumo para procesos que utilicen técnicas de Minerías de Datos, en particular algunas que provienen de la Inteligencia Computacional, cuyos modelos han sido propuestos y evaluados [6]. En una próxima etapa del diseño se desarrollarán interfaces gráficas que orientarán al usuario en la aplicación de diversos modelos de Minería de Datos.

Elementos del trabajo y metodología

1. Señales de Variación de Distensión Arterial (VDA)

El desarrollo de la aplicación forma parte de un proyecto de investigación relacionado con registros de señales bioeléctricas generadas por el equipo diseñado y construido por el laboratorio de Bioingeniería de la Universidad Nacional de Mar del Plata. Dicho dispositivo recibe la denominación de "Sistema de Adquisición y Registro de Onda de Pulso" [1].

El dispositivo sensa la variabilidad de la distensión de la arteria radial, de forma no invasiva en la muñeca, utilizando un transductor capacitivo. Para obtener los registros de VDA se utilizó un sensor capacitivo, consistente en una placa metálica rectangular de 2 cm² que se apoya sobre la zona de palpación del pulso radial. Conectando el paciente a potencial de masa, queda formado un capacitor entre ambos. El movimiento de la pared arterial se transmite a la superficie de la piel, provoca el movimiento de la placa, y la capacidad varía al ritmo del ciclo cardíaco. Las variaciones de capacidad son transformadas en variaciones de tensión, constituyendo la salida del detector de VDA. Los cambios sensados en la capacidad, producidos por las variaciones de diámetro de la arteria radial, son digitalizados (frecuencia de muestreo: 200 Hz.), con palabras de ocho bits y adquiridos en una computadora personal. Los temblores fisiológicos del brazo del paciente actúan como artefactos superpuestos a la señal útil. Para reducirlos, se emplea la técnica de promediación. Para este proceso se promedian 8 ciclos cardíacos de señales VDA, previamente filtradas para eliminar artefactos de 50 hz. El análisis de estos registros permite calcular índices para evaluar el grado de elasticidad o de envejecimiento de la arteria

El registro obtenido en la arteria radial de un adulto sano tiene el aspecto indicado en la Fig.1, donde también se halla la descripción de los parámetros de la

señal. Estos parámetros son los que deben ser calculados a partir del procesamiento de los registros, almacenados y disponibles para ser impresos como resultados del estudio [7].

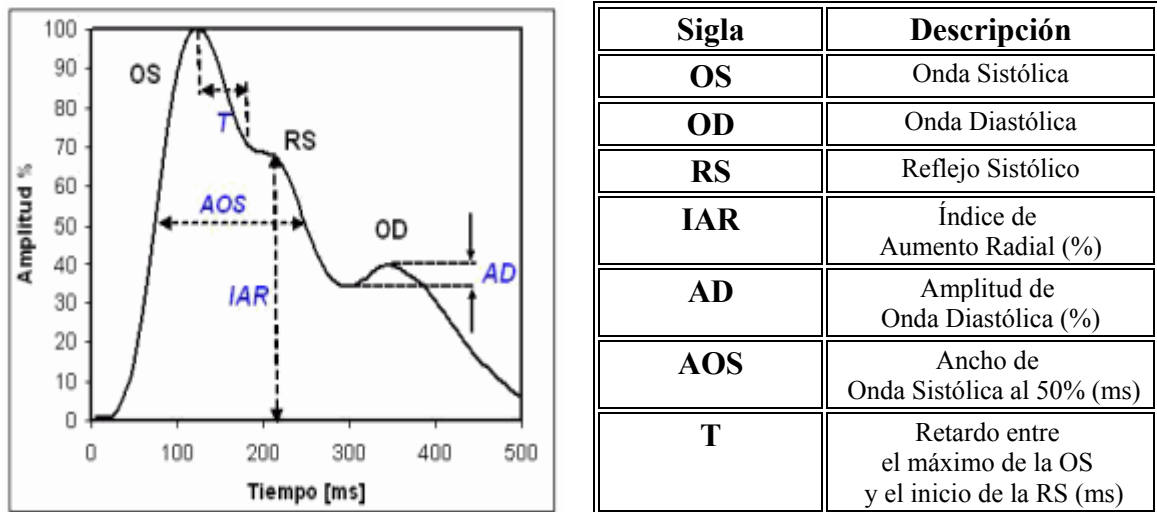


Fig. 1 - Onda de pulso y parámetros característicos.

En Fig. 2 se muestra el caso de una onda de pulso perteneciente a una persona joven sana (menos de 20 años), mientras que en la Fig. 3 se presenta el registro de VDA de una persona sana de más de 70 años.

Hemos desarrollado una técnica a partir del análisis de la VDA para la determinación de indicadores que caracterizan la evolución de la misma. Los resultados demostraron que el sistema empleado puede ser utilizado para la detección temprana de anomalías en el sistema arterial, y para la evaluación o seguimiento de pacientes hipertensos [8]. El conocimiento del IAR en función de la edad permite detectar anomalías que impliquen alteraciones tempranas del sistema cardiovascular [9], previas a la aparición de otros síntomas, y permite tomar medidas preventivas.

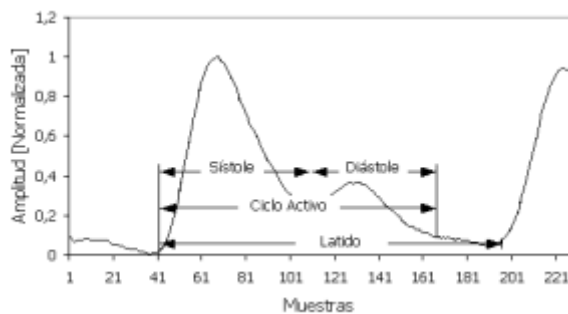


Fig. 2 - VDA, caso Joven. Onda diastólica bien diferenciada

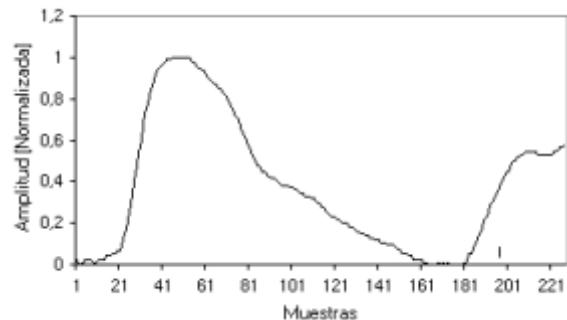


Fig. 3 - VDA, caso Anciano. La onda sistólica reflejada tempranamente ensancha la primera región, desaparece el pico diastólico.

2. Metodología de desarrollo de Software

La metodología de desarrollo de software para este proyecto se basó en RUP (Rational Unified Process), ya que permite administrar el proceso de construcción de una manera ágil, flexible y ordenada. Como resultado de la evaluación de la

problemática y la conformación del grupo de trabajo se presentó un documento preliminar con las características deseables del producto y las posibles soluciones [10].

El paso siguiente fue la selección de casos de uso con el fin de generar diferentes escenarios y evaluar la interacción del usuario con el mismo, estos fueron validados por los interesados y una vez aceptados se refinaron. Con la lista de casos de uso terminada, se establecieron dependencias y prioridades entre los mismos y a partir de esto se plantearon las iteraciones. Cada una de las iteraciones, implica la selección de casos de uso específicos y aborda un análisis en profundidad de la parte de la problemática que resuelven, diseño, implementación y testing unitario y de integración.

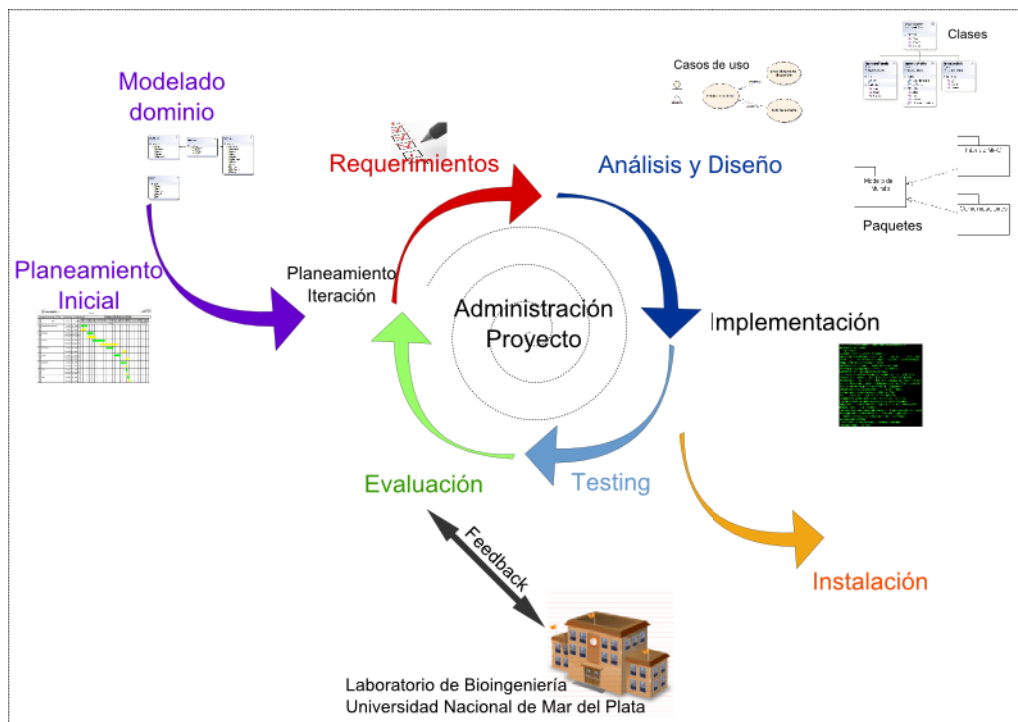


Fig. 4: Iteraciones

Luego de cada iteración, se evalúa en función de los casos de uso implementados si es necesaria su validación o se espera a finalizar la siguiente. En caso de realizarse, se produce un "feedback" que genera nuevos cambios y mejoras a lo ya implementado. Una vez finalizada la secuencia de iteraciones planificadas, se realiza el testing de la aplicación completa.

3. Definición de las etapas del procesamiento de la señal de VDA

La aplicación de escritorio del procesamiento de la señal de VDA tiene como fin la identificación de los parámetros de la forma de la señal, señalados en el apartado 1 de esta sección. Este proceso se inicia con la carga del archivo correspondiente al registro de la señal VDA, adquirido por el equipo descrito y almacenado en el sistema. Estos registros son una secuencia de ciclos de la señal de VDA, en general contaminada con ruido proveniente de diversas fuentes. Las etapas de procesamiento se detallan a continuación:

Filtrado: La señal cruda (tal como fue adquirida) es sometida a un proceso de filtrado con el fin de eliminar ruido. En particular, son filtradas las bandas entre 0 y 5 Hz correspondientes a los movimientos diafragmáticos del paciente (respiración), ansiedades o movimiento muscular. Las bandas que también son eliminadas son

aquellas que superan los 50 Hz. Posterior al proceso de filtrado se eliminan las tendencias o deriva de la señal para suprimir las variaciones de la línea base y se invierten los puntos de la señal generando como salida una nueva entidad: "La señal filtrada".

Identificación de segmentos: La señal filtrada es dividida en segmentos, aislando cada onda de pulso. Para la segmentación se utiliza el concepto de que la onda sistólica es el valor máximo del segmento, se busca dicho máximo dentro de un rango determinado y a partir del mismo se desplaza una cantidad de puntos determinados hacia atrás en busca del inicio del mismo. Una vez establecido el principio del segmento se desplaza hacia adelante una cantidad "N" de puntos y a partir de "N" busca el siguiente máximo en un rango determinado, el mismo es la onda sistólica del siguiente segmento, de ahí se retrocede hasta el inicio del mismo, el cual no solo indica el principio de este, sino el final del anterior.

Selección de los mejores segmentos: Una vez obtenidos los segmentos se lleva a cabo el proceso de descarte de segmentos defectuosos y se seleccionan los mejores según determinados parámetros morfológicos. En esta etapa, a partir de un entorno interactivo, el usuario podrá seleccionar los segmentos que formen parte del conjunto a promediar. Si bien en una próxima versión el sistema realizará automáticamente esta selección utilizando técnicas de aprendizaje automático, dicha pre-selección será presentada como una sugerencia al usuario experto de la aplicación para que quede finalmente en sus manos la decisión de utilizar dichos segmentos o en su defecto poder realizar una selección manual según su propio criterio.

Cálculo del segmento medio: A partir de los segmentos seleccionados previamente se calcula el promedio punto a punto de los mismos, obteniendo una señal que se considera representativa del estado arterial del paciente. A continuación se la normaliza en amplitud entre 0 y 1, obteniéndose una nueva entidad en el sistema la Curva Final Promedio (CFP).

Cálculo de derivadas: Utilizando la CFP se calculan las derivadas segunda y cuarta y se las escala en relación a la CFP para identificar claramente los puntos de corte con la misma.

Extracción de características: Disponiendo de los puntos de corte es posible identificar los parámetros de interés que caracterizan la VDA tales como: IAR (Índice de Aumentación Radial), AOS (Ancho de Onda Sistólica) y AOD (Amplitud Onda Diastólica). Como ejemplo, el IAR se calcula utilizando el valor de la señal de VDA normalizada donde su cuarta derivada tiene un cambio de signo dentro de una región de interés (ROI). Este punto es identificado por el operador con la ayuda de una interfaz gráfica.

Almacenamiento en la Base de Datos: Las características de la señal y la información del paciente (previamente ingresados desde la aplicación web o bien dados de alta en la actual sesión) son almacenados de manera persistente en una base de datos.

El sistema tiene la capacidad de brindar informes generados por consultas a la base de datos. Estos informes pueden corresponder a:

- El análisis de un registro de un paciente, que detalla antecedentes clínicos y permite el ingreso de observaciones por parte del operador. Este informe individual tiene como destinatario final a un profesional médico, quien podrá utilizarlo para evaluar el estado arterial del paciente.
- Los análisis de registros de un mismo paciente, con el propósito de evaluar su evolución.

- Listados y estadísticas de descriptivas por edades de pacientes, patologías, hábitos de vida (fumadores, deportistas, sedentarios). Las tablas generadas pueden ser insumo de procesos de Minerías de Datos (vistas minables).

4. Arquitectura

Debido al elevado consumo de recursos que implica el procesamiento de la señal de pulso sumado a la necesidad de poder efectuar un procesamiento independiente de la existencia de conexión con un servidor (remoto o local) se optó por una arquitectura Cliente/Servidor tradicional [11].

La aplicación fue diseñada en dos partes, donde cada parte opera en un determinado ambiente y contiene funcionalidades específicas. En líneas generales, la aplicación ejecutada en el ámbito de escritorio se encuentra orientada al procesamiento de los registros de pulso y la generación de informes médicos. Como una extensión, se planteó a la aplicación web como un repositorio de señales y con funcionalidades respecto al acceso, consulta y administración de datos personales y clínicos de pacientes, antecedentes familiares, administración de médicos, visualización de registros e informes, estadísticas y la posibilidad de realizar el seguimiento del paciente.

En la Fig. 6 se muestra la distribución y arquitectura de cada aplicación por separado y la arquitectura subyacente.

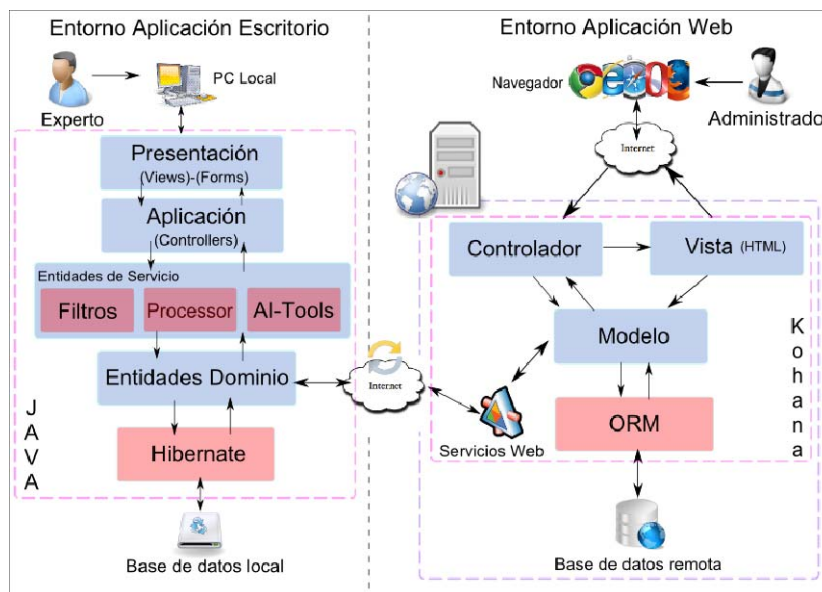


Fig. 6: Arquitectura Cliente/Servidor y representación ámbitos web y de escritorio

Dentro del ámbito de escritorio, la aplicación fue diseñada con una arquitectura MVC (Modelo-Vista-Controlador)[12] e implementada en el lenguaje JAVA[13]. El empleo del patrón arquitectónico MVC permite la separación lógica de la interfaz de usuario (vista), de lógica de control (controlador) y lógica de negocio y entidades (modelo) impactando positivamente en la mantenibilidad y extensibilidad del sistema.

Los controladores dirigen la ejecución del sistema creando nuevas vistas, nuevos modelos y nuevos controladores para el flujo de la información a través del sistema. En cuanto a los modelos o entidades, distinguimos dos tipos: las entidades persistentes y las entidades lógicas. Las entidades persistentes hacen referencia a objetos del dominio en cuestión y se almacenan de forma persistente. Ej: datos de pacientes, de médicos y registros VDA.

Las entidades lógicas, en cambio, aluden al encapsulamiento de módulos y funcionalidades particulares de la aplicación pero que no son persistentes: filtros digitales, segmentador y graficador entre otros.

Dentro de las capacidades de la aplicación de escritorio se destacan: Administración de Pacientes, Administración de Registros de Pulso, Procesamiento de señales de Pulso y generación de informes.

Para conseguir cierta flexibilidad e independencia respecto a la base de datos, se delega la funcionalidad de acceso a datos y persistencia al framework Hibernate[14] integrado en la aplicación.

Cabe remarcar que se priorizó y se enfocó en la usabilidad del sistema aplicando patrones de interfaz de usuario buscando agilizar las tareas involucradas en el proceso. En particular, en la selección de segmentos.

La aplicación de escritorio fue diseñada para poder operar incluso en modo "offline" con lo cual cuenta con un almacenamiento local. Dicho almacenamiento contiene un subconjunto de los datos almacenados de la base de datos remota y se sincroniza con esta a través del uso de un servicio web. En el caso que no existiese conexión disponible con el servidor, la información procesada por la aplicación sería almacenada de forma local. Si luego la conexión se logra establecer en la sesión actual, o se inicia una nueva en otro momento, se realizará una sincronización automáticamente con el servidor para actualizar los valores de la base de datos remota manteniendo una consistencia general

La estructura de la base de datos tanto en el ámbito web como en el ámbito de escritorio son equivalentes, y fue diseñada teniendo en cuenta el manejo de históricos mediante la replicación de datos, para en un futuro aplicar técnicas de Minería de Datos para identificación de patrones a partir de los históricos.

La arquitectura web fue diseñada teniendo en cuenta sus funciones generales tales como: centralización de los datos, repositorio de históricos y acceso y consultas a la base de datos. La idea surgió como una manera de expandir las funcionalidades existentes en la aplicación de escritorio y permitir que la información esté disponible para una mayor cantidad de usuarios, entre ellos, médicos que puedan visualizar informes de registros de pulso sobre pacientes a los cuales diagnostican.

Mientras que la aplicación de escritorio se concentra en el procesamiento de las señales en si, la aplicación web permite:

- Administración de Usuarios y sus roles dentro del sistema (Administrador, Data Entry, Consultor), donde cada rol tiene un conjunto de permisos específico.
- Administración de los datos personales de pacientes.
- Administración de los registros de pulso sin procesar de pacientes, en conjunto con información clínica relevada al momento del registro.
- Administración de los datos personales de médicos.
- Visualización de informes de registros procesados de pacientes
- Generación de reportes estadísticos respecto a parámetros como: sexo, edad, altura y peso de pacientes.
- Seguimiento de los parámetros de la onda de pulso del paciente y sus datos clínicos a lo largo del tiempo.
- Resguardo y restauración de los datos registrados

Por otra parte, al centralizar los datos, la información ingresada vía web debía quedar disponible para las aplicaciones cliente de escritorio y debía poder resolver la problemática de la sincronización con las estaciones de trabajo, las cuales subían información a ser almacenada en la base remota. Es por ello, que fue necesaria la

implementación de un servicio web [15] con intercambio de mensajes SOAP [16] para resolver este requerimiento.

5. Tecnologías y Herramientas Utilizadas

Las tecnologías se han seleccionado en base a los requerimientos no funcionales y sin perder de vista que se trata de un producto que se utilizará en el marco de un proyecto de investigación de una universidad pública. Se priorizaron las herramientas "open source" con el fin de evitar costosas licencias, esto lejos de limitar el proyecto, lo enriqueció con abundante documentación y la disponibilidad del código para ser modificado según las necesidades.

Desde un principio la base de datos planteada y consensuada con el laboratorio fue MySQL[17], dado que la misma brinda toda la funcionalidad necesaria para un proyecto de estas características, y el equipo de desarrollo cuenta con vasta experiencia en dicha herramienta. Sin embargo el hecho de trabajar con distintos niveles de abstracción, entre ellos una capa de persistencia y la utilización de un ORM (Object-Relational Mapper) [18], hace que el sistema sea independiente de la base de datos elegida, con lo cual la misma puede ser sustituida por otra sin cambio alguno en la funcionalidad general.

Para el diseño de la base de datos se utilizó DBDesigner [19] y para la manipulación de los datos se contó con dos aplicaciones, SQLyog **¡Error! No se encuentra el origen de la referencia.** y phpMyAdmin[21], la primera es una GUI (interfaz gráfica de usuario) de escritorio y la segunda una interfaz web.

Para la aplicación de escritorio se optó por Java ya que es totalmente independiente de la plataforma, cuenta con un importante conjunto de librerías que facilitan el desarrollo de distintas funcionalidades y se dispone de mucha documentación al respecto. También influyó en esta decisión la popularidad y corta curva de aprendizaje del lenguaje, para que en un futuro pueda ser continuado o modificado por otros desarrolladores. Para la generación y manipulación de gráficos se utilizó la librería JFreeChart[22] la cual es sumamente popular en desarrollos que requieran gráficos y cuenta con varios ejemplos de su uso con fines educativos.

Finalmente, los entornos de trabajo utilizados para el desarrollo en Java fueron Netbeans[23] y Eclipse[24], utilizando el primero especialmente para GUI (Interfaz gráfica de usuario).

Respecto a la aplicación web, la misma se construyó utilizando tecnologías presentes en la hoy llamada "Web 2.0" [25], es decir, mucha interactividad con el usuario y transmisiones asíncronas AJAX [26] con JSON (JavaScript Object Notation) **¡Error! No se encuentra el origen de la referencia.** como formato de intercambio de datos con el objetivo de desarrollar una aplicación ágil, usable y con una operatividad similar a las aplicaciones de escritorio tradicionales.

Con la intención de fomentar buenas prácticas de desarrollo y facilitar las tareas de mantenibilidad y extensibilidad de la aplicación web, se llevó a cabo la implementación en base a un framework basado en PHP [28] versión 5 con soporte para orientación a objetos que implementa el patrón Modelo-Vista-Controlador y cuenta con librerías, módulos y utilitarios que aceleran el proceso de desarrollo.

El framework de desarrollo rápido de aplicaciones elegido fue Kohana[29] integrado con Doctrine [30] un framework de persistencia para PHP que implementa conceptos análogos a Hibernate y que cuenta con su propio lenguaje de consultas.

En lo que concierne a las tecnologías involucradas del lado del navegador y que facilitan la interactividad con el usuario respecto a interfaz y comunicaciones AJAX con el servidor se utilizó la librería jQuery[31], un framework javascript multi-navegador y

plug-ins basados en el mismo concepto. Cabe destacar entre ellas, jqGrid [32], la cual permitió que las tareas de gestión sobre objetos del dominio fuesen simples para el usuario al emplear un patrón de interfaz similar a una planilla de cálculo en lugar de los formularios tradicionales.

Aptana Studio [33], basado en Eclipse, fue el entorno de desarrollo integrado elegido para las tareas de codificación para la web.

El modelado de las aplicaciones se realizó con UML utilizando StarUML [34], basado en diagramas de casos de uso, diagramas de clase, diagramas de paquetes, diagramas de estado y diagramas de secuencia.

La administración del desarrollo se llevó a cabo mediante Google Docs, aprovechando la facilidad que brinda de poder compartir los documentos en forma remota y poder editarlos de manera concurrente, dando la posibilidad de participar no solo al grupo de desarrollo sino también al los miembros del laboratorio. Esta misma tecnología fue utilizada para llevar el control de las horas trabajadas. Finalmente se utilizó el hosting de Google [35] para alojar el código fuente y manejar un control de versiones, permitiendo trabajar en forma remota, y resolver posibles problemas de concurrencia.

Resultados

Actualmente, la aplicación de escritorio del producto se encuentra en etapa de puesta en funcionamiento y pruebas. Mientras que el entorno Web, aún se encuentra en fase de elaboración.

Como resultado del desarrollo se implementaron vistas del sistema representadas por los formularios y componentes gráficos. Las imágenes que se muestran a continuación siguen la secuencias de pasos descripta para el proceso.

A partir de la lectura de un registro de onda de pulso, asociado a un paciente, en la siguiente figura se observa una pantalla donde se muestra al operador la gráfica tanto la señal cruda (gráfico superior) como de la señal ya filtrada e invertida para permitir su posterior segmentación y extracción de parámetros (gráfico inferior). En la parte izquierda de la ventana, se muestra un menú que contiene las secciones importantes del análisis en progreso. En una de dichas secciones, se muestran los datos correspondientes al paciente seleccionado (Fig. 7).

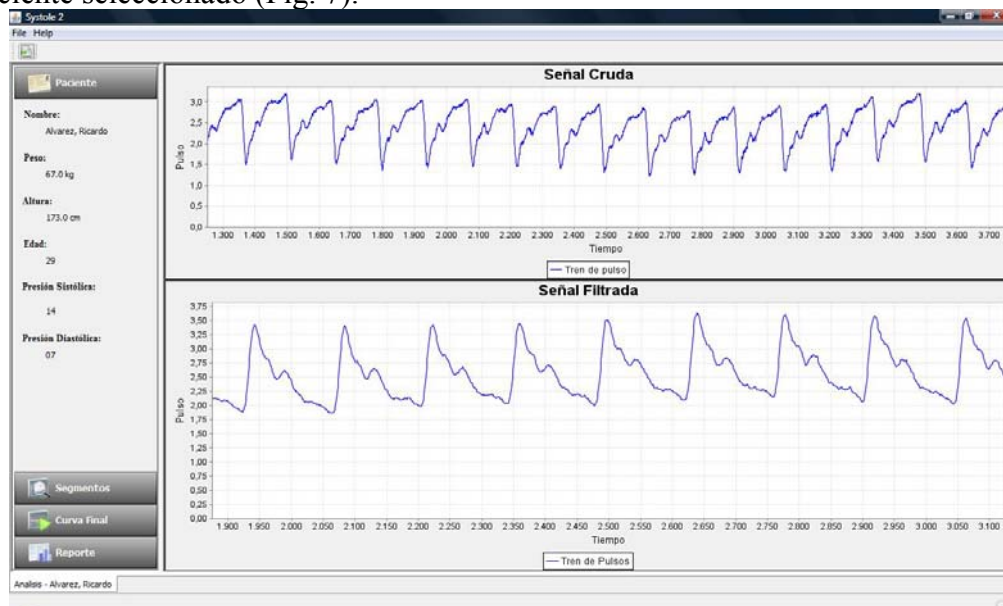


Fig. 7 - Pantalla Principal

Una vez que el tren de pulsos es segmentado como se describió en las etapas del proceso, los segmentos resultantes son exhibidos mediante una galería de gráficos, pudiendo recorrerla y elegir los segmentos deseados (Fig. 8).

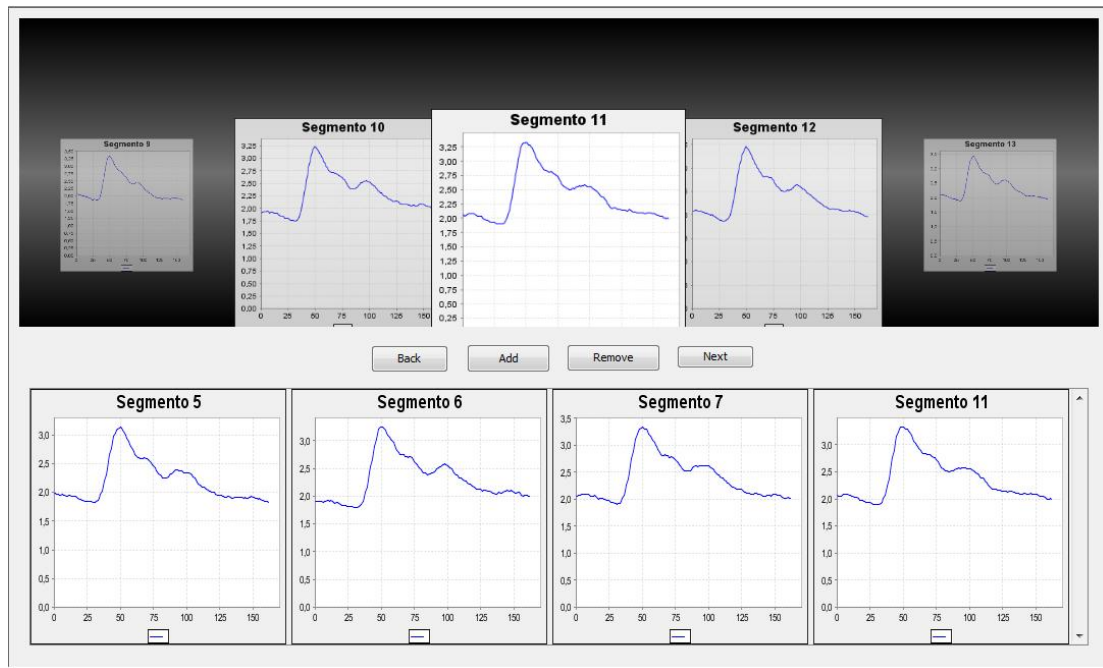


Fig. 8 -Selección de Segmentos

Luego de promediar los segmentos seleccionados, se genera la curva final que será utilizada para el cálculo de los índices (Fig.9). Sobre la curva final, se calculan las segunda y cuarta derivadas (Fig. 10), las cuales permiten obtener los índices deseados (Fig.1).

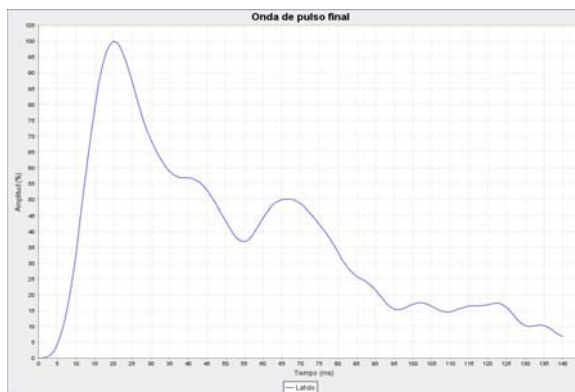


Fig. 9- Segmento Final

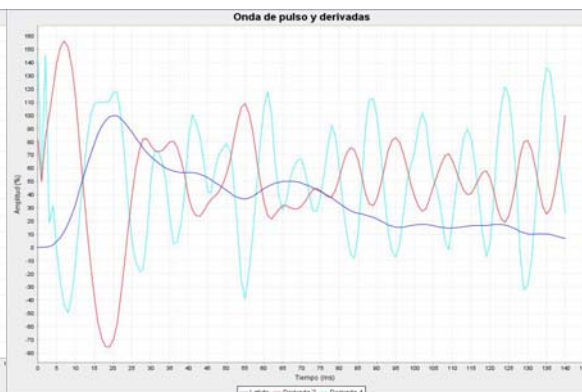


Fig. 10 - Derivadas

El informe de un registro correspondiente a un paciente, que puede emitirse desde la aplicación de escritorio donde se realiza el proceso o bien desde una conexión remota, contiene (Fig. 11)

- los datos del paciente: edad, Peso [Kg], Presión sistólica [mmHg], Presión diastólica [mmHg], Índice de masa corporal, Frecuencia cardíaca
- el perfil de la señal de VDA, similar al Segmento Final de la Fig. 10

- los índices de la misma: Amplitud OD, IAR, Ancho 50% OS y los valores de nominales de los mismos correspondientes a la década a la que pertenece el paciente.
- comentarios sobre la morfología de la VDA ingresados por el profesional que procesa la señal.
- antecedentes: medicación, nivel de colesterol y niveles de glucemia, antecedentes familiares cardiovasculares y diabéticos, hábitos de vida.
- a modo de conclusión: comentarios relacionados al diagnóstico del sistema arterial en relación a la descripción de la VDA

Discusión

El Sistema de Gestión para el procesamiento, análisis y almacenamiento de los registros de señales de Variabilidad de Distensión Arterial ha sido diseñado a partir de los requerimientos funcionales de un grupo de investigadores del área biomédica. Dado que el producto no se corresponde con una aplicación estándar, el proceso de evaluación de las características deseables y sus posibles soluciones ha insumido una buena proporción de recursos.

Se considera que la elección de RUP como el proceso de desarrollo de software basada en casos de uso ha sido adecuada, dado que debido a la complejidad del problema el abordaje a partir de los casos de uso y el método de las iteraciones para el refinamiento de las soluciones es conveniente.

Las primeras iteraciones tuvieron una fuerte carga de diseño, durante las mismas se determinó la arquitectura del sistema y surgieron los diagramas de clases iniciales, el de base de datos, y la relaciones entre los distintos componentes del sistema. En consecuencia, fue necesaria la previsión de un lapso de tiempo mayor para estas iteraciones con respecto a las posteriores. Algunas iteraciones presentaron una importante carga de recursos dedicados a la investigación de técnicas específicas, como por ejemplo la etapa del procesamiento de señales de VDA. En esta etapa los desarrolladores de software recibieron soporte por los investigadores del área de bioingeniería quienes desarrollaron prototipos de algoritmos en lenguaje M (de Matlab ®) [36] para su posterior implementación en el SisGeVDA.



INFORME
12/02/2010 16:45 hs

Médico: Juan Alvarez				
Paciente: José Perez				
Edad	Peso [Kg]	PS [mmHg]	PD [mmHg]	Fuma
45	79	130	95	NO
INDICES				
Amplitud OD %	Aumentación %	Ancho 50% OS ms	Masa Corporal	Frecuencia Card. ppm
10.3	61	215	24	72.4

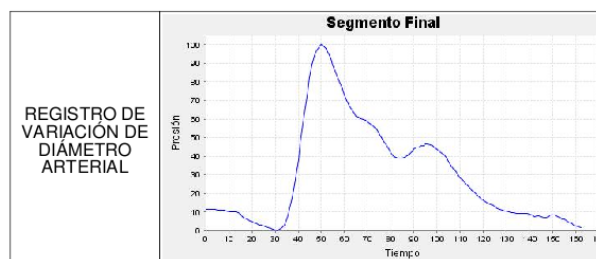


Fig. 11. Informe individual de un registro de VDA.

En las primeras iteraciones se dedicó buena parte de los recursos a la elección de las herramientas a utilizar en el desarrollo del sistema. Si bien finalmente se optó por JAVA, se consideró inicialmente el uso de Python [37] como lenguaje de desarrollo y particularmente el uso de las librerías NumPy [38] y SciPy [39], dado su creciente uso en los ámbitos académicos y científicos. El hecho de no disponer de un entorno de desarrollo satisfactorio en dichas herramientas, como así también la relativamente poca compatibilidad entre plataformas, aportó elementos para la decisión de realizar el desarrollo en tecnologías basadas en JAVA. Además la experiencia previa del equipo en desarrollos exitosos con esta tecnología jugó un papel determinante en la toma de decisiones.

El diseño y desarrollo del SisGeVDA, como herramienta para la investigación en medicina, ha resultado una experiencia novedosa para los autores informáticos de la propuesta. La conformación de un grupo de trabajo interdisciplinario *ad-hoc* generó una sinergia positiva que permitió la implementación del sistema con un inversión de recursos adecuada. En una próxima etapa se propone integrar el sistema con herramientas de Inteligencia Computacional para el reconocimiento automático de patrones derivados de la morfología de la señales de VDA.

Agradecimientos

Dr. Ing. Gustavo Meschino. Universidad Nacional de Mar del Plata y Universidad FASTA
Ing. Nicolas Martínez Saez. Universidad FASTA

Referencias

- [1]. Introzzi A., Clara F., Valiente S.M. Sistema de registro de ondas de dilatación arterial. *VI Congreso Argentino de Bioingeniería. Tomo de resúmenes* pp9, (1988).
- [2]. Aníbal R. Introzzi, Lucía I. Passoni, Adriana G. Scandurra, Gustavo J. Meschino, Fernando M. Clara "Relación entre el Envejecimiento y la Morfología de los Registros de Variación de

- Diámetro Arterial: Clasificación con Redes Neuronales Artificiales”. *Rev Argent Cardiol - Sociedad Argentina de Cardiología - ISSN 0034-7000*. Vol. 73 - Nro. 5 - Pág. 352-357 (2005)
- [3]. L.I.Passoni, G.Meschino, A. Scandurra, F.Clara, A. Introzzi “Evaluación de las características de la variación del diámetro de la arteria radial por medio de mapas auto-organizados.” *Revista Argentina de Bioingeniería*. ISSN 03295257 Vol 8:1 pag 30-37.(2002).
- [4]. L. Passoni, J. Fritschy, A. Introzzi F. Clara."Clustering algorithms as classifiers of blood-pressure recordings" *Computer Simulations in Biomedical*. Editors: H.Powr, R.T. Hart Proceedings.Editor Computational Mechanics Publications Southampton, Boston. ISBN 1-85312-321-8. Catalogado en Britnish Library.pp:513:520- (1995)
- [5]. Clara F., Scandurra A., Meschino G, Passoni L., Introzzi A. “Evaluación de la Aumentación Radial en Base a Registros de Variación de Diámetro Arterial”. *Trabajos Completos del XV Congreso Argentino de Bioingeniería. IV Jornadas de Ingeniería Clínica*. ISBN: 950-698-155-8. Paraná. (2005)..
- [6]. L. I. Passoni, A. Scandurra, G. Meschino, A.L. Dai Pra, F. M. Clara “Sistema de Soporte a las Decisiones Médicas utilizando Herramientas Neuro-Fuzzy.”. *Anales JAIIO 2002*. Vol. 31 (CD).ISSN 1666-1141. SIS’2002 Simposio Argentino de Informática y Salud. Vol. 5. ISSN 1666-1125. pág. 211-217. (2002)
- [7]. Clara F., Passoni L., Scandurra A., Meschino G., Introzzi A: ”Detección de riesgo cardiovascular basada en registros de variación de diámetro arterial”. *XIV Congreso Argentino de Bioingeniería – Córdoba, Octubre 2003*.
- [8]. Kohara K, Jiang Y, Igase M, Hiwada K “Effect of reflection of arterial pressure on carotid circulation in essential hypertension”. *Am J Hypertens*. Oct; Vol 12; 1015-20. (1999)
- [9]. Duprez D, Kaiser D, Whitwam W y col: Determinants of Radial Artery Pulse Wave Analysis in Asymptomatic Individuals. *Am J Hypertens*. 2004; 17(8):647-53.
- [10]. Kruchten, P. 2000 *The Rational Unified Process: an Introduction, Second Edition*. 2nd. Addison-Wesley Longman Publishing Co., Inc.
- [11]. Arquitectura Cliente/Servidor (<http://es.wikipedia.org/wiki/Cliente-servidor>)
- [12]. Patrón MVC (http://es.wikipedia.org/wiki/Modelo_Vista_Controlador)
- [13]. Java J2Se (<http://java.sun.com/javase/index.jsp>)
- [14]. Hibernate (<http://www.hibernate.org/>)
- [15]. Servicio Web (http://es.wikipedia.org/wiki/Servicio_web)
- [16]. SOAP (<http://es.wikipedia.org/wiki/SOAP>)
- [17]. MySQL (http://www.mysql.com/?bydis_dis_index=1)
- [18]. ORM (http://es.wikipedia.org/wiki/Mapeo_objeto-relacional)
- [19]. DBDesigner4 documents. FabForce.net.(<http://www.fabforce.net/dbdesigner4/docs.php>)
- [20]. SQLYog Gui (<http://www.webyog.com/en/>)
- [21]. PhpMyAdmin (<http://www.phpmyadmin.net/>)
- [22]. JFreeChart (<http://www.jfree.org/>)
- [23]. NetBeans (<http://www.netbeans.org>)
- [24]. Eclipse (<http://www.eclipse.org/>)
- [25]. Web 2.0 (http://es.wikipedia.org/wiki/Web_2.0)
- [26]. AJAX (<http://es.wikipedia.org/wiki/AJAX>)
- [27]. JSON (<http://es.wikipedia.org/wiki/JSON#Referencias>)
- [28]. PHP (<http://www.php.net/index.php>)
- [29]. Kohana PHP 5 Framework (<http://kohanaframework.org/>)
- [30]. Doctrine Project. (<http://www.doctrine-project.org/>)
- [31]. jQuery (<http://jquery.com>)
- [32]. jqGrid (<http://www.trirand.com/jqgridwiki/doku.php>)
- [33]. Aptana Studio (<http://www.aptana.org/>)
- [34]. StarUML (<http://staruml.sourceforge.net/>)
- [35]. Google Code Hosting (<http://code.google.com>)
- [36]. Matlab (<http://www.mathworks.com>)
- [37]. Python (<http://www.python.org>)
- [38]. NumPy (<http://numpy.scipy.org>)
- [39]. SciPy (<http://www.scipy.org>)

Datos de Contacto:

Sebastián Alberto Fink. Universidad FASTA, Mar del Plata. CP: 7600. safink@gmail.com
 Juan Manuel Jaime. Universidad FASTA, Mar del Plata. CP: 7600. jmj155@gmail.com