

Diseño Centrado en el Usuario y Claims Analysis: Metodologías para Introducir Mejoras a un Software de Terapia de las Afasias

Bartolucci, Carlos Alberto
Departamento de Ingeniería en Sistemas de
Información
Universidad Tecnológica Nacional
La Plata Av.60 esq. 124
carlosalbertobartolucci@outlook.com

Colombini, Matias
Departamento de Ingeniería en Sistemas de
Información
Universidad Tecnológica Nacional
La Plata Av.60 esq. 124
matiascolombini@gmail.com

Abstract

El Diseño Centrado en el Usuario (DCU) es una metodología que busca lograr la mejora de la interfaz de usuario (IU) y Claims Analysis (CA) busca una recolección de la mayor cantidad de necesidades de los usuarios para obtener los requerimientos de diseño. En este artículo, los usuarios considerados serán terapeutas del lenguaje y personas afásicas. Tanto DCU como CA serán revisadas en cuanto a su aplicabilidad para proponer mejoras a una aplicación web destinada a la rehabilitación del lenguaje en afásicos usando la metodología de telepráctica.

1. Introducción

El Diseño Centrado en el Usuario (DCU) considera al diseño de interfaz de usuario (IU) como el proceso por el cual un ingeniero del software crea, analiza y construye mecanismos que proporcionen el mejor canal de comunicación entre el software y el usuario. Consiste en las etapas de: descubrimiento, diseño, desarrollo [1]. “Un diseño cuidadoso de la IU es una parte fundamental del diseño general del software. Si se quiere que el software alcance su máximo potencial, su interfaz debe diseñarse de manera que se ajuste a las habilidades, experiencia y expectativas de los usuarios previstos” [2]. Claims Analysis (CA) es una metodología para obtener la mayor información posible de las necesidades de los usuarios desde diferentes fuentes.

La afasia se define como "la pérdida de la capacidad para comprender y usar el lenguaje que resulta en una habilidad reducida para comunicarse, establecer y mantener interacciones y cumplir con los roles sociales en la vida" (Lasker, Garret y Fox, 2007). Dado que no existen estadísticas en Argentina, se calcula (por extrapolación de USA y Europa) que existen 150 afásicos

por millón de habitantes anualmente, debido a cualquier causa -accidentes cerebro-vasculares, traumatismos craneoencefálicos, tumores, y otros-; lo que constituye una población estable de 164.000 afásicos basándonos en la estadística realizada por el INDEC en el año 2001 (36.200.000 habitantes con una estimación para el año 2014 de 39.700.000 habitantes) (FUNDAFASIA).

La afasia dificulta comunicarse, establecer y mantener interacciones. En la afasia Leve el paciente manifiesta dificultades para nombrar objetos, personas, lugares y más tiempo del usual para comunicarse. Pueden mantener una conversación normal en muchas circunstancias pero suelen tener problemas para comprender el lenguaje cuando es extenso o complejo y en hallar las palabras para expresar una idea o explicarse -similar a tener una palabra en la "punta de la lengua". Está limitado para comprender un chiste verbal, anotar rápidamente un número telefónico, encontrar una palabra determinada para expresar una opinión de manera precisa, etc., limitando sus posibilidades de desempeño socio-laboral.

2. Lev_Aphasia

Lev_Aphasia es una aplicación de plataforma WEB, desarrollada por un equipo interdisciplinar de ingenieros en sistemas de información, fonoaudiólogos, psicólogos y becarios de la carrera de Ingeniería en Sistemas de Información, que propone estimular el lenguaje, la memoria de trabajo y atención dividida (dominios cognitivos que también se alteran en los cuadros afásicos).

Para el módulo de lenguaje se diseñó ejercitación para estimular la denominación, el lenguaje en uso (pragmática del lenguaje) y relaciones lógico-semánticas.

Lev_Aphasia está destinado a la rehabilitación del lenguaje en personas afásicas mediante telepráctica, que es una metodología complementaria a la terapia cara a

cara, permitiendo que reciban tratamiento en su domicilio diariamente.

Este prototipo fue desarrollado en el contexto de un proyecto de I+D, en el que personas afásicas colaboraron desinteresadamente. Se prevé aplicar CA a las personas afásicas que constituyan la muestra del Protocolo de Ensayo Clínico avalado por la Comisión Conjunta de Investigación en Salud, Ministerio de Salud de la Provincia de Buenos Aires Expte. 1105.

El DCU se aplicará en lo inmediato, en la mejora de la IU, y usaremos CA para una adecuada recolección de las necesidades para luego obtener los requerimientos para el diseño.



Figura 1. Estructura de la telepráctica.

Actualmente, el usuario terapeuta se conecta mediante internet al servidor que aloja al software Lev_Aphasia junto con los correspondientes ejercicios que él mismo diseñó y asignó para los usuarios pacientes, quienes también se conectan a dicho servidor, mediante internet, para resolver sus respectivos ejercicios. Una vez resueltos, el terapeuta podrá revisar sus resultados. (Ver Figura 1).

Cuando el usuario terapeuta abre la aplicación, accede a la “Página de acceso del usuario terapeuta”, donde deberá ingresar su nombre de usuario y contraseña (ver Figura 2).

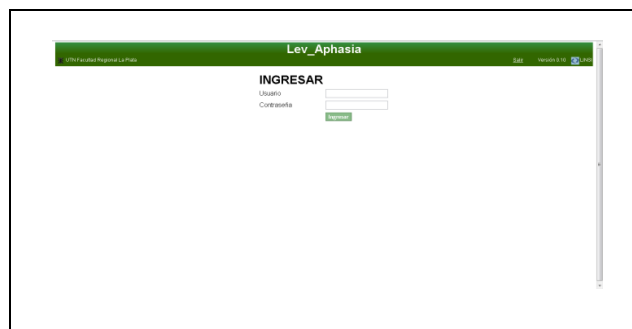


Figura 2. Página de acceso del usuario terapeuta.

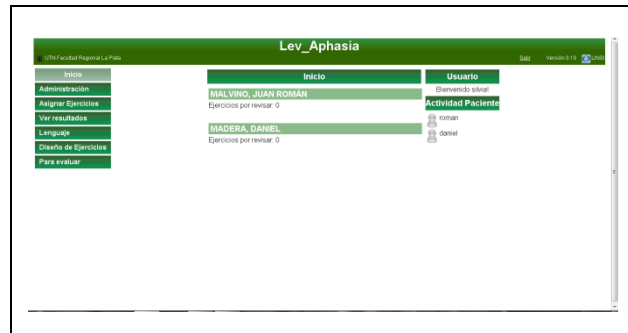


Figura 3. Página de inicio del usuario terapeuta.

Una vez que haya ingresado a su cuenta, y ya en la pantalla “Página de inicio del usuario terapeuta”, podrá ver la cantidad de ejercicios que tiene que revisar de cada uno de sus pacientes, ver cuáles son sus pacientes que han estado realizando ejercicios recientemente, y acceder a diferentes módulos: el de administración, el de asignación de ejercicios, el de visualización de resultados, el del lenguaje, el de diseño de ejercicios y el de evaluación (ver Figura 3).



Figura 4. Página de asignación de ejercicios.

Cuando el terapeuta accede al módulo de “Asignar ejercicios”, podrá seleccionar los pacientes a los que desea asignarles su correspondiente ejercitación. A su vez, para cada uno de ellos, podrá elegir los ejercicios que considere necesarios de las áreas a rehabilitar de la persona afásica (lenguaje, memoria, atención), además de poder ver los que ya le fueron asignados. Los ejercicios posibles, para el caso del módulo lenguaje, son crucigramas, antónimos, sinónimos, palabra aislada, sopas de letras, frases lacunares, frases sinónimos, palabra en contexto. Y podrán seleccionarse de acuerdo a diferentes criterios: el área a rehabilitar, el área de estimulación (ejemplo: denominación), la categoría gramatical, el tipo, los campos semánticos, el nivel de dificultad, etc.

Un ejemplo de la pantalla correspondiente a cuando el usuario pulsa el botón “Asignar ejercicios” puede verse en (ver Figura 4).



Figura 5. Página de inicio del usuario persona afásica.

Una vez seleccionados, el paciente podrá ver de la “Página de inicio del usuario persona afásica” los ejercicios que tiene que realizar, cuyos resultados podrán ser revisados luego por el terapeuta (Ver Figura 5).

Para el desarrollo del software se ha usado como metodología el Proceso Unificado. Éste es un proceso ágil basado en casos de uso y define cuatro fases (inicio, elaboración, construcción y transición), las cuales se superponen en el tiempo. Cada una de ellas tiene hitos de terminación claramente definidos. Cada fase puede dividirse en iteraciones, y algunas de ellas admiten entregas parciales del producto.

El Proceso Unificado genera modelos visuales con un fuerte uso de UML, enfatiza el control de cambios y la administración de requerimientos. Emplea una arquitectura robusta basada en componentes desde el comienzo y verifica constantemente la calidad. [3].

Para el software Lev_Aphasia, se decidió usar herramientas de software libre y gratuito, fundamentalmente porque se lo está trabajando en una Universidad Pública. Dos de estas herramientas son: Trac y Java Enterprise Edition (JEE).

Trac es empleada para gestionar proyectos. Es minimalista y colaborativa. Reemplazó al Assembla, ya que ofrece los mismos servicios, pero gratuitamente. Es de fácil instalación, tiene pocos errores convirtiéndola así en una herramienta robusta, y puede integrarse con otras herramientas usadas en el desarrollo tal como el control de versiones Subversion. Hoy en día, se está usando la versión “estable”, la 0.11.7, provista por los repositorios oficiales del sistema operativo Debian 6.

Java Enterprise Edition (JEE) es empleada como herramienta de desarrollo. La plataforma Java EE reduce la complejidad del desarrollo de aplicaciones empresariales proveyendo un modelo de desarrollo, API y ambiente en tiempo de ejecución que permita a los desarrolladores concentrarse en la funcionalidad. Java EE application development se concentra en el nivel medio para hacer que la gestión de aplicaciones empresariales sea más fácil, más robusta y más segura.

Se empleó la librería innovadora Google Web Toolkit para implementar una aplicación Web cliente/servidor con RIA (Rich Internet Application). Transforma el código Java en JavaScript que implementa un modelo de eventos y se comunica con el servidor mediante comunicaciones livianas y asíncronas conocidas como AJAX (JavaScript asíncrono y XML). RIA es una aplicación de Internet enriquecida que funciona en cualquier navegador Web, pero tiene una interacción con el usuario similar a la de una aplicación de escritorio.

3. Describiendo la Metodología Diseño Centrado en el Usuario

El Diseño Centrado en el Usuario es una metodología que considera al diseño de interfaz de usuario como el proceso por el cual un ingeniero del software crea, analiza y construye mecanismos que proporcionen el mejor canal de comunicación entre el software y el usuario. Esta metodología permite capturar y compartir conocimientos acerca de la población de usuarios de destino para poder así conducir a diseños más adecuados. Sostuvieron que “Un diseño cuidadoso de la interfaz de usuario es fundamental para el diseño general del software. Si se quiere que el software alcance su máximo potencial, su interfaz debe diseñarse de manera que se ajuste a las habilidades, experiencia y expectativas de los usuarios previstos.

Dicho de otra manera, la importancia del diseño de la interfaz de usuario se basa en que éste será el que modele la interacción entre usuario y aplicación, y por tanto posibilita o no la *consecución* de los objetivos perseguidos por el usuario.” [4].

La metodología del diseño centrado en el usuario consiste en las etapas de: descubrimiento, diseño, desarrollo. Esto difiere del presentado en el artículo “User Needs Analysis and requirements engineering: Theory and practice”, en el cual se sostiene que dentro de los principios del Diseño Centralizado en el Usuario, el proceso de diseño y el de desarrollo comprime 4 fases: gearing-up, diseño inicial (es donde se realiza el Análisis de las necesidades del usuario), desarrollo iterativo (se enfoca principalmente en el test iterativo de usabilidad) y la de instalación. El proceso del diseño progresa en los ciclos iterativos de análisis, diseño y evaluación. Luego, Shackel y Richardson (1991) proponen un modelo donde se agrega una fase de prototipo entre la de desarrollo y la de instalación para comenzar a alinear las actividades de HCI (interfaz hombre-computadora). [5].

4. Comprendiendo el Claims Analysis

Claims Analysis es una metodología sencilla que busca una recolección de la mayor cantidad de necesidades de los usuarios para obtener los requerimientos de diseño.

Un claim es una hipótesis sobre algo que debería considerarse importante en el diseño. No es un requerimiento porque no especifica cómo debería ser el diseño para una cierta necesidad: solo la documenta. Un claim es una declaración sobre lo que es cierto para el usuario, pero un requerimiento declara lo que hay que construir. En este punto dentro del proceso de diseño, es preferible capturar ideas en vez de identificar soluciones. [6].

Así, un Claim o "demanda", es una enunciación de las consecuencias de una característica específica de un diseño o artefacto en los usuarios y otras partes involucradas. Las demandas capturan el conocimiento de una manera que puede ser compartido, debatido, fortalecido, refutado y conectado. De esta manera, el "Análisis de Demanda" explicita las ventajas y desventajas de las características del diseño que se describen en los escenarios de uso actuales o futuros.

Por ejemplo: "AC demanda recibir en tiempo real información de las actividades de los pacientes cuando realizan las ejercitaciones de su terapia en sus domicilios". Este Claim no habla acerca de cómo eso se resolverá (requerimiento), que podría ser resuelto a través de un e-mail o con la plataforma web [7] y la base de datos MySQL de Lev_Aphasia, donde quedan registrados los resultados de la terapia.

4.1 Beneficios de claims

Un Claim coloca conocimiento/suposiciones en una forma tal que pueda ser compartida, debatida y rebatida (documento digital, escrito manual, grabación de voz, etc.) esto crea un artefacto reusable; captura el cambio continuo en el entendimiento a medida que los artefactos maduran. [8].

La naturaleza hipotética de las demandas es muy valiosa en situaciones en las que existen diferencias de opiniones sobre la manera de proceder en las tareas de diseño. En proyectos multidisciplinarios, cada disciplina participa con sus propias necesidades y preocupaciones, teniendo una perspectiva diferente de lo que se considera como importante para "un buen diseño", en relación con los usuarios a los que está destinado dicho diseño.

4.2 Ejemplo de software desarrollado aplicando Claim Analysis

El ejemplo de software que utilizaremos en este caso es la aplicación Bangaspeak, la cual está destinada al tratamiento de pacientes afásicos en cuanto a su habla y lenguaje. Para el desarrollo de la aplicación BangaSpeak, se empleó la metodología del diseño centrado en el usuario y la de Claims Analysis. Se tuvieron en cuenta una gran cantidad de claims. A partir de ellos, el equipo de desarrollo del software seleccionó los objetivos de diseño que consideraron más importantes para lograr la mayor satisfacción posible al usuario, debido a los recursos limitados existentes.

Los 3 objetivos de diseño más importantes seleccionados fueron:

- a_ Hacer que los terapeutas del lenguaje crean que vale la pena adoptar la tecnología
- b_ disponer de herramientas para crear y gestionar tratamiento de estímulos
- c_ construir una aplicación multifuncional.

A continuación, se exponen para cada uno de los objetivos y los claims a partir de los cuales se obtuvieron:

4.2.1. Hacer que los terapeutas del lenguaje crean que vale la pena adoptar la tecnología. Algunos claims obtenidos fueron: "Los terapeutas no tienen tiempo suficiente para preparar las sesiones", "las aplicaciones no hacen más que lo ya hecho en lápiz y papel", evidenciando que no se interesan en la tecnología en sí misma, por lo que debe ser justificable para que se comprometan con tiempo, esfuerzo y dinero. También se recolectaron claims que decían que los terapeutas tienen barreras para adaptarse a la tecnología (tiempo para aprender, costos, etc.) y a usar la tecnología (disponibilidad de materiales/aplicación, soporte IT, conocimiento). Sin embargo, de los claims extraídos, es evidente que quieren dar terapia de alta calidad. Basándonos en todo lo comentado, debemos aprovechar las capacidades que ofrece la tecnología tales como video, audio, micrófonos, acelerómetros, GPS, búsqueda online de imágenes, para crear tareas nuevas y totalmente diferentes y también la capacidad del reconocimiento del habla para puntuar automáticamente el habla del paciente y darle retroalimentación.

Este objetivo también es apoyado por claims que dicen que los terapeutas quieren comprometer a los pacientes a practicar en el hogar y que los pacientes necesitan oportunidades para practicar el lenguaje expresivo.

También se extrajeron claims que sugerían que los pacientes quieren tratamientos efectivos basados en la evidencia y adaptados para su tipo particular de afasia. Quieren tareas apropiadas para adultos, fáciles de usar e

incentivadoras. Por lo tanto, para que la persona afásica esté motivada, sería deseable que la ejercitación ofrezca juegos con puntajes o avances de niveles, aunque hay poca evidencia de rehabilitación de afasia a través de los juegos.

4.2.2 Disponer de herramientas para crear y gestionar terapias del lenguaje. Este objetivo surge a partir de claims que refieren el tiempo limitado de preparación que tienen los terapeutas para crear materiales que generen el compromiso de los pacientes a realizar la terapia periódicamente (adherencia a la terapia) y además, porque desean que ese material sea reusable. Se pide que los estímulos sean individualizados y personalizados para sus pacientes (en algunos artículos se ha discutido del valor potencial de materiales personalizados) Pero, basado en el claim de que los terapeutas tienen tiempo limitado para crear materiales (menos personalizados), se aceptó crear estímulos en la aplicación para asegurar eficiencia en tiempo y re-uso. De ahí, que, consideramos se desarrolle una aplicación/herramienta multifuncional para proveer así varias tareas, características, controles clínicos, configuración de tareas y ejercicios (ejercitación ad hoc).

4.2.3 Construir una aplicación multifuncional. Además de lo dicho anteriormente, (claims que expresan que los terapeutas no tienen tiempo para aprender y recordar el uso de varias aplicaciones y que necesitan protocolos terapéuticos), la aplicación multifuncional les da flexibilidad para desarrollar un plan de tratamiento para cada paciente; y/o mientras el paciente progresa, pueden actualizarse las tareas o asignarse nuevas.

El enfoque del diseño multifuncional propuesto contrasta con el de las aplicaciones terapéuticas empaquetadas. En las aplicaciones empaquetadas, los estímulos y tareas están juntas y la aplicación sigue un protocolo relativamente establecido, fijo. Algunas aplicaciones empaquetadas ajustan por sí solas el nivel de dificultad o avance según la performance del paciente (un paciente estándar o ideal), pero las tareas y estímulos son generalmente las mismas con un criterio de avance a niveles superiores algo cuestionable. Dada la heterogeneidad de la afasia, hace que sea menos probable que un paciente dado, se empareje bien al modelo de paciente ideal que propone el software empaquetado. No obstante, estos software, ofrecen estadísticas y cierta “evidencia”, así que los terapeutas se confortan en saber que la terapia tiene evidencia que la sustente (los pacientes mismos también quieren tratamientos efectivos basados en la evidencia). Esto posibilita que el empaquetamiento disminuya el tiempo de ajuste de la sesión. Sin embargo, la compra de aplicaciones empaquetadas para una práctica clínica puede ser difícil de justificar por el costo y expectativa de uso. [9].

4.3 Extrapolando los objetivos de diseño del BangaSpeak al software Lev_Aphasia

A continuación, analizaremos cómo aplicaríamos los objetivos de diseño mencionados en el BangaSpeak a Lev_Aphasia.

4.3.1. Hacer que los terapeutas del lenguaje creen que vale la pena adoptar la tecnología. Lev_Aphasia permite configurar y testear la cantidad de intentos y el tiempo de ejecución de cada ejercicio, además de la fecha en que fue realizado.

Las características de diseño de los ejercicios, organizados en módulos de estimulación de Memoria de Trabajo, Atención Dividida y Lenguaje, facilitan la elección de los mismos y el diseño de la terapia por parte del terapeuta.

Respecto del usuario afásico, estimamos que las características de los ejercicios diseñados posibilitaran su adherencia a la terapia. Esto estaría dado también, en la situación de que las personas afásicas puedan realizar la ejercitación en el momento del día en que se sientan más aptos para realizarlos.

Creemos que esta oferta motivará a la persona afásica para la utilización de la aplicación y así a su adherencia a la terapia.

El poseer una base de datos con información suficiente sobre la frecuencia de uso de la aplicación por parte del afásico, alertara al usuario terapeuta, en caso de que el usuario afásico no realice la ejercitación en el tiempo y frecuencia pactados.

Lev_Aphasia, posee ejercicios diseñados que permiten almacenar variables psicolingüísticas para dar evidencia respecto de la terapia (ya sea de las elecciones de los terapeutas como de las personas afásicas) tales como grado de frecuencia, categoría semántica, categoría gramatical, número de sílabas, etc. Creemos que un punto muy atractivo es la modalidad de telepráctica, que posibilita a las personas afásicas puedan realizar la terapia diseñada, en el hogar y los terapeutas puedan acceder a información de sus actividades de forma remota. Todo esto, sobre todo la telepráctica, hace que los terapeutas y personas afásicas se comprometan con tiempo, esfuerzo y ahorro de dinero. Elementos muy atractivos para la adopción de la aplicación.

En varios de los claims obtenidos en el desarrollo del BangaSpeak, vemos que los terapeutas y usuarios aún hablan mucho de lo que piensan o sienten reflejando básicamente la falta de adopción de la tecnología, requiriendo así, que la aplicación sea fácil de usar por parte de los usuarios.

4.3.2. Disponer de herramientas para crear y gestionar terapias del lenguaje. Lev_Aphasia permite crear materiales reutilizables para disminuir el tiempo de preparación de ejercitación por parte de los terapeutas. El prototipo Lev_Aphasia, está diseñado para incorporar una gran cantidad de ejercitación balanceada en tres niveles (bajo, medio y alto) en la base de datos. Como así también agregar nuevos diseños de ejercicios de estimulación. Es una aplicación/herramienta multifuncional para proveer así varias tareas, características, controles clínicos, configuración de tareas y estímulos.

4.3.3. Construir una aplicación multifuncional. Lev_Aphasia permite desarrollar un plan de tratamiento para cada paciente; mientras la persona afásica la utiliza, el usuario terapeuta puede supervisar la realización de los ejercicios y decidir cuándo progresar en el nivel de complejidad de los mismos; pueden actualizar, re-diseñar y re-utilizar tareas para otros pacientes.

5. Diseño Centrado en el Usuario para Rehabilitación de Afasias

El diseño orientado al usuario “se caracteriza por asumir que todo el proceso de diseño y desarrollo del software debe estar conducido por el usuario, sus necesidades, características y objetivos”, los dos autores mencionados de dicha cita, D’Uva y Koleff, sostuvieron que “Siguiendo esta línea y puntualmente dentro del ámbito de la telerehabilitación, el diseño orientado al paciente (PCD) es la metodología que enfoca e implementa el diseño de interfaz sobre la personalización de herramientas informáticas como herramientas de terapia”.

Dentro de este modelo, el paciente es visto como el actor dentro de un contexto clínico; es decir, para el caso de lesiones cerebrales, el contexto clínico son las sesiones de terapia. Recordemos que éstas son espacios de tiempo cortos y centrados en una actividad puntual. Por lo tanto, la interfaz del software debe diseñarse de tal manera que permita ser aprendida por el paciente en poco tiempo, así como también, ser fácil de configurar.” [10]. El tiempo estimado de sesión terapéutica no debe superar los 45 minutos.

El diseño centrado en el paciente es una metodología de diseño que incorpora factores técnicos y clínicos. Además, aprovecha la habilidad del paciente para rediseñar y refinar la interfaz de usuario, y lograr una muy buena adaptación entre el sistema y el usuario. Puede incrementar la función cognitiva de un paciente como un

efecto tecnológico. Adicionalmente, algunos individuos han mostrado un aumento en las habilidades cognitivas subyacentes. [10].

6. Aplicaciones del Claim Analysis y Diseño Centrado en el Usuario en Lev_Aphasia

Hemos logrado una aplicación web mediante "diseño evolutivo" para obtener rápidamente un prototipo funcional y mínimo. De esta manera, todo cambio de una funcionalidad o de la interfaz, puede ser captado y analizado en una etapa temprana del proyecto, minimizando la incertidumbre y optimizando los tiempos de desarrollo. Esto permite evaluar si la idea del diseñador satisface las necesidades correspondientes. Por tanto, se prioriza el diseño simple y eficaz por sobre el diseño óptimo, en pos de obtener un producto más fácil de entender y mantener.

En la etapa de descubrimiento del DCU, se puede usar CA para identificar claims informalmente permitiendo que ideas, creencias, percepciones puedan derivarse de muchas fuentes diferentes, a la vez de centrarse más en una amplia variedad de usuarios. El CA también permite documentar las necesidades para los miembros del equipo que tenían poca o nada de experiencia con afásicos o con pacientes con patología en el habla del lenguaje. [8].

7. Trabajo Futuro: Mejoras a Lev_Aphasia Uniendo Diseño Centrado en el Usuario y Claims Analysis

El Claims Analysis, en el marco del enfoque de DCU, es apropiado para llevar adelante las mejoras al prototipo Lev_Aphasia, teniendo en cuenta el Diseño Evolutivo ya que así podremos saber exactamente las necesidades de los usuarios, satisfacerlas y estimular a que se comience a adoptar esta tecnología.

Hay otros métodos para capturar las necesidades, como por ejemplo “diseño participativo y entrevistas”, que confían fuertemente en la retroalimentación verbal del usuario, siendo inadecuado para afásicos. Otros, como el de testeo de prototipos, requiere que los usuarios interactúen con el prototipo pero en esta etapa aún hay que implementarlo. [10].

8. Discusión

Mientras algunas metodologías de DCU proponen explícitamente el refinamiento iterativo de los requerimientos, la iteración por sí misma es generalmente confinada a una etapa particular, complicando volver, recopilar, actualizar e integrar los datos adicionales del Análisis de las necesidades del usuario dentro del diseño

de la interfaz una vez que se haya completado la fase de captura de requerimientos. También, a partir de dicho artículo, algunos autores dicen que los requerimientos siguen apareciendo y son refinados durante el ciclo de vida, pero no está claro cómo es esto o cómo puede ser realizado en la práctica. Algunas metodologías se enfocan en la recolección de datos del Análisis de las necesidades del Usuario pero poco dicen sobre la representación y comunicación de las salidas de este Análisis, otras lo contemplan pero no dicen claramente cómo actualizar los hallazgos a medida que surgen los nuevos requerimientos. Hay por tanto escasez de metodologías apropiadas que guíen la iterativa recolección de datos, análisis, presentación y actualización de los hallazgos del Análisis de las Necesidades del Usuario en los actuales marcos de tiempo de desarrollo. [11].

Algunos autores consideraban que dado que las herramientas rápidas de prototipado empezaban a estar disponibles, y aplicando métodos económicos tales como estudios en lapicera y papel, el prototipado podría y debería estar integrado en todas las fases en puntos importantes en lugar de una sola fase. Esto es consistente con el modelo espiral de Boehm en el cual el desarrollo del sistema progresa rígidamente de una fase a la siguiente como en el modelo tradicional de cascada. Pero a la hora de construir el prototipo nos encontramos con el problema expuesto en el artículo “User Needs Analysis and requirements engineering: Theory and practice”: no está claro cuándo el prototipado debería comenzar y terminar, qué y cómo debería ser prototipado y con qué nivel de fidelidad, cuándo se debería terminar con el análisis de las necesidades del usuario, cuándo la retroalimentación de los usuarios en los prototipos no podría estar más acomodado, o cómo integrar nuevos requerimientos dinámicamente. [5].

9. Conclusión

Consideramos que CA es una metodología apropiada para aplicar a nuestro software porque permite, usando un método más bien informal, obtener mucho conocimiento de los usuarios desde varias fuentes distintas. Es una muy buena técnica que presenta conocimiento/suposiciones en una forma tal que pueda ser compartida, debatida y rebatida por todos los integrantes del equipo: ingenieros en Sistemas de Información, Terapeutas del Lenguaje y Psicólogos. Esto es beneficioso porque cada disciplina participa con sus propias necesidades y preocupaciones, teniendo una perspectiva diferente de lo que se considera como importante para “un buen diseño”.

El DCU, junto con el CA, nos permitiría identificar las necesidades que hagan una interfaz más útil por parte de los usuarios, propiciando la adherencia a la terapia por parte del usuario paciente y propiciando la adopción de la

herramienta por parte del usuario terapeuta, generando un circuito virtuoso.

10. Trabajos Futuros

Identificar las primeras barreras para la adherencia a la terapia.

11. Referencias

- [1] Vredenburg, K., Mao, J. Y., Smith, P. W., & Carey, T., “A survey of user-centered design practice”, In Proceedings of the SIGCHI conference on human factors in computing systems, New York, NY: ACM. Abril 2002, pp. 2.
- [2] Sommerville, I. “Ingeniería del Software”. Pearson. 2005, p. 332
- [3] Fontela, Carlos. “UML: modelado de software para profesionales”. Alfaomega Grupo Editor. 2011, p. 16
- [4] Koleff, M. D’Uva, I., “Aplicación Web para tratamiento de la Afasia Leve: Criterios de Usabilidad”, CONAIIISI. 2014, p. 1
- [5] Lindgaard, G., Dillona, R., Trbovicha, P., Whitea, R., Fernandes, G., Lundahlb, S., Pinnamanenib, A., “User Needs Analysis and requirements engineering: Theory and practice”, aHuman Oriented Technology Lab (HOTLab), Carleton University, 1125 Colonel By Drive, Ottawa, Ont., Canada K1S 5B6 bAmita Corporation, 1420 Blair Place, Gloucester, Ont., Canada K1J 9L8. Agosto 2005, p. 48
- [6] Messamer, P., Ramsberger, G., Atkins, A., “BangaSpeak: an example of app design for aphasia clients and SLP users”, Aphasiology, 2016, pp. 165 – 167
- [7] Messamer, P., Ramsberger, G., Atkins, A., “BangaSpeak: an example of app design for aphasia clients and SLP users”, Aphasiology, 2016, p. 167
- [8] Messamer, P., Ramsberger, G., Atkins, A., “BangaSpeak: an example of app design for aphasia clients and SLP users”, Aphasiology, 2016, p. 165
- [9] Messamer, P., Ramsberger, G., Atkins, A., “BangaSpeak: an example of app design for aphasia clients and SLP users”, Aphasiology, 2016, pp. 166-169
- [10] Koleff, M. D’Uva, I., “Aplicación Web para tratamiento de la Afasia Leve: Criterios de Usabilidad”, CONAIIISI. 2014, p. 4
- [11] Lindgaard, G., Dillona, R., Trbovicha, P., Whitea, R., Fernandes, G., Lundahlb, S., Pinnamanenib, A., “User Needs Analysis and requirements engineering: Theory and practice”, aHuman Oriented Technology Lab (HOTLab), Carleton University, 1125 Colonel By Drive, Ottawa, Ont., Canada K1S 5B6 bAmita Corporation, 1420 Blair Place, Gloucester, Ont., Canada K1J 9L8, Agosto 2005, p. 48 – 49