

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

USABILIDAD Y COMUNICABILIDAD EN GRID COMPUTING

ARTURO ALEJANDRO FIGUEROA RETAMAL

TESIS DE GRADO
MAGÍSTER EN INGENIERÍA INFORMÁTICA

DICIEMBRE 2011

Pontificia Universidad Católica de Valparaíso
Facultad de Ingeniería
Escuela de Ingeniería Informática

USABILIDAD Y COMUNICABILIDAD EN GRID COMPUTING

ARTURO ALEJANDRO FIGUEROA RETAMAL

Profesor Guía: **Cristian Rusu**

Programa: **Magíster en Ingeniería Informática**

Diciembre 2011

Resumen: Grid Computing es una tecnología emergente de computación distribuida, que permite conectar equipos geográficamente distantes en una red especial llamada “Grid”, proporcionando a estos equipos la capacidad de compartir recursos de distinto tipo y de manera transparente para los usuarios. Actualmente, la tecnología de Grid Computing es bastante compleja, por lo cual su acceso es casi exclusivo para expertos. Dada esta problemática, se hace necesario mejorar la Usabilidad en los productos del tipo Grid Computing para acercar este software a otros tipos de usuarios. La Comunicabilidad es un atributo que representa el entendimiento del mensaje entregado por el diseñador de software al usuario, por lo que se podría inferir que la Comunicabilidad puede tener incidencia en algún grado sobre la Usabilidad de un producto de software. Para abordar la problemática asociada a los productos de Grid Computing, se pretende plantear una metodología para evaluar productos de software Grid Computing, cosa que se complementa con un estudio de Comunicabilidad. Con este estudio, además, se analiza el impacto de la Comunicabilidad sobre la Usabilidad para estos productos de software.

Palabras clave: Heurísticas de Usabilidad, Comunicabilidad, Grid Computing.

Abstract: Grid computing is an emerging technology of distributed computing, which allows you to connect machines geographically distant in a special network called "Grid", providing this equipment the ability to share resources of different types and transparently to users. At present, Grid Computing technology is quite complex, so it is almost exclusive access to experts. Given these problems, it is necessary to improve the Usability of such products in Grid Computing to bring this software to other types of users. Communicability is an attribute that represents the understanding of the message delivered by the software designer to the user, so it could be inferred that Communicability could have an impact in some degree on the Usability of a software product. To address the problems associated with the products of Grid Computing, is intended to propose a methodology for evaluating software products Grid Computing, which is complemented by a study of Communicability. This study also analyzes the impact of Communicability on Usability for these software products.

Keywords: Usability Heuristics, Communicability, Grid Computing.

Índice de contenidos

1. Presentación del Tema	1
1.1 Introducción.....	1
1.2. Discusión Bibliográfica	2
1.3 Análisis de Objetivos y Metodología.....	2
1.3.1 Objetivo General	2
1.3.2 Objetivos Específicos	2
1.3.3 Metodología	3
1.4 Guía de Secciones.....	4
2. Marco Teórico: Estado del Arte	6
2.1 La Tecnología Grid Computing	6
2.1.1 Características de los Sistemas Grid Computing	7
2.1.2 Arquitectura de los Sistemas Grid Computing	9
2.1.3 Principales Usos de la Tecnología Grid Computing.....	10
2.1.4 Tipos de Grid.....	11
2.1.5 Tipos de Usuarios.....	13
2.2 Usabilidad en Productos de Software	14
2.2.1 Evaluaciones de Usabilidad.....	14
2.2.2 Heurísticas de Usabilidad en productos de Grid Computing	16
2.2.3 Checklist para la Evaluación de Usabilidad en Aplicaciones Grid Computing.....	17
2.2.4 Experimentos de Usabilidad en aplicaciones del tipo Grid Computing	20
2.3 La Ingeniería Semiótica y la Comunicabilidad.....	23
2.3.1 La Ingeniería Semiótica.....	23
2.3.2 La Comunicabilidad	23

2.3.3 Inspecciones Semióticas.....	24
2.3.4 Pruebas de Comunicabilidad	26
3. Solución Propuesta.....	28
3.1 Mejora Tentativa a Heurísticas Específicas para Productos del Tipo Grid Computing.....	28
3.1.1 Problemas Detectados Solamente con Heurísticas Genéricas	28
3.1.2 Análisis Inicial de Problemas No Detectados con las Heurísticas Específicas para Aplicaciones del Tipo Grid Computing.....	29
3.2 Análisis Inicial de Impacto de la Comunicabilidad Sobre la Usabilidad	30
3.2.1 Experimento Piloto: Inspección Semiótica del Producto GreenView	31
3.2.2 Experimento Piloto: Inspección Semiótica del producto GreenLand	38
3.2.3 Análisis de resultados y conclusiones en experimentos piloto de Comunicabilidad	47
4. Desarrollo de la Solución.....	49
4.1 Revisión de Heurísticas Específicas para Aplicaciones del Tipo Grid Computing	49
4.1.1 Propuesta de heurística: Diseño Estético	50
4.1.2 Propuesta de heurísticas	51
4.2 Experimentación y Pruebas	56
4.2.1 Evaluación de Comunicabilidad sobre el producto GreenView	59
4.2.2 Evaluación heurística sobre producto el Boinc haciendo uso de las heurísticas específicas para aplicaciones del tipo Grid Computing	62
4.2.3 Evaluación Heurística Sobre el Producto Boinc Haciendo Uso de las Heurísticas Genéricas de Nielsen ..	65
4.2.4 Inspección Semiótica Sobre el Producto Boinc	68
4.2.5 Evaluación Heurística Sobre el Producto GreenLand v2 Haciendo Uso de las Heurísticas Específicas para Aplicaciones del Tipo Grid Computing.....	79
4.2.6 Evaluación Heurística Sobre el Producto GreenLand v2 Haciendo Uso de las Heurísticas Genéricas de Nielsen.....	82
4.2.7 Prueba de Usabilidad Sobre el Producto GreenLand v2.....	85
4.2.8 Inspección Semiótica sobre producto el GreenLand v2	89

4.2.9 Evaluación de Comunicabilidad sobre el producto GreenLand v2	96
4.3 Análisis de Impacto de la Comunicabilidad Sobre la Usabilidad	99
4.3.1 La Relación Entre Quiebres Comunicacionales y Heurísticas de Usabilidad	100
4.3.2 La relación Comunicabilidad - Usabilidad en el producto GreenView	102
4.3.3 La Relación Comunicabilidad - Usabilidad en el Producto Boinc	104
4.3.4 La Relación Comunicabilidad - Usabilidad en el Producto GreenLand v2	106
4.3.5 Resumen de Heurísticas Asociadas a Problemas de Comunicabilidad en los Productos Analizados	109
5. Conclusiones	110
6. Bibliografía	112
7. Anexos	114
Anexo A: Acuerdo de Confidencialidad.....	114
Anexo B: Cuestionario de Prueba de Usabilidad producto GreenView	115
Anexo C: Planilla de Evaluación de Comunicabilidad.....	118
Anexo D: Plantilla completa de problemas y notas en evaluación heurística del cliente Boinc haciendo uso de las heurísticas específicas para aplicaciones del tipo Grid Computing	122
Anexo E: Plantilla completa de problemas y notas en evaluación heurística de GreenLand v2 haciendo uso de las heurísticas específicas para aplicaciones del tipo Grid Computing	125
Anexo F: Plantilla completa de problemas y notas en evaluación heurística del cliente Boinc haciendo uso de las heurísticas genéricas de Nielsen	127
Anexo G: Plantilla completa de problemas y notas en evaluación heurística de GreenLand v2 haciendo uso de las heurísticas genéricas de Nielsen	130
Anexo H: Plantilla de Prueba de Usabilidad sobre el producto GreenLand v2	132

Índice de Figuras

Figura 2.1: Representación de un sistema de Grid Computing.....	7
Figura 2.2: Arquitectura de un sistema de Grid Computing.....	9
Figura 3.1: Información explícita sobre ciertas funciones al colocar el cursor del mouse sobre ellas.....	31
Figura 3.2: Información explícita con información acerca del software en la pestaña About GreenView.....	32
Figura 3.3: Opciones de pestaña Coarse to fine.....	33
Figura 3.4: Información de la pestaña Fine to coarse.....	33
Figura 3.5: Información de la pestaña Processing status.....	34
Figura 3.6: Información de la pestaña Gpp computing.....	34
Figura 3.7: Botones y controles que manipulan la selección de área en el mapa.....	36
Figura 3.8: Acción del botón guardar.....	36
Figura 3.9: Símbolo meta-lingüístico de mensaje sobre botón.....	39
Figura 3.10: Opciones para seleccionar el área geográfica.....	39
Figura 3.11: Pantalla de los procesos.....	40
Figura 3.12: Distintos símbolos estáticos.....	41
Figura 3.13: Símbolos estáticos sección processing status.....	41
Figura 3.14: Símbolos estáticos sección input data section.....	42
Figura 3.15: En esta imagen correspondiente a la sección de procesos.....	42
Figura 3.16: Controles de manipulación directa.....	44
Figura 3.17: Controles de manipulación directa y mensajes de estado del sistema.....	44
Figura 3.18: Ventana de pop-up que se abre al hacer click sobre el ícono de diskette.....	45
Figura 4.1: Arquitectura del software de Grid Computing Boinc.....	57
Figura 4.2: Programa cliente ejecutando una serie de tareas enviadas por los proyectos base.....	58
Figura 4.3: GreenLand v2 después de haber terminado un proceso.....	59

Figura 4.4: Documentación y ayuda en línea.....	69
Figura 4.5: Mensajes de los administradores de proyectos.....	69
Figura 4.6: Ayuda del servidor.....	70
Figura 4.7: Vista básica del cliente Boinc.....	71
Figura 4.8: Mensaje desplegado al colocar el puntero del mouse encima de los botones.....	72
Figura 4.9: Vista avanzada, pestaña Proyectos.....	72
Figura 4.10: Vista avanzada, pestaña Tareas.....	73
Figura 4.11: Vista avanzada, pestaña Disco.....	73
Figura 4.12: Opciones del botón Avanzado.....	73
Figura 4.13: Modo avanzado, preferencias de computación.....	74
Figura 4.14: Opciones de conexión.....	74
Figura 4.15: Mensajes del servidor.....	75
Figura 4.16: Retroalimentación acerca del progreso constante de las tareas.....	76
Figura 4.17: Despliegue de mensaje de “Proyecto añadido”, ahora el equipo añadirá a sus procesos tareas de otro proyecto.....	76
Figura 4.18: Mensajes entregados por el servidor al enviar Jobs.....	77
Figura 4.19: Gráfico de cuestionario post-test, pregunta 1.....	87
Figura 4.20: Gráfico de cuestionario post-test, pregunta 2.....	87
Figura 4.21: Gráfico de cuestionario post-test, pregunta 3.....	88
Figura 4.22: Gráfico de cuestionario post-test, pregunta 4.....	88
Figura 4.23: Gráfico de cuestionario post-test, pregunta 5.....	89
Figura 4.24: Ayuda externa del producto GreenLand v2.....	90
Figura 4.25: Símbolo metalingüístico de descripción de un elemento.....	90
Figura 4.26: Símbolo metalingüístico de descripción del proyecto.....	91
Figura 4.27: Símbolo metalingüístico de descripción de funciones.....	91
Figura 4.28: Símbolos estáticos referentes a la estructura del layout y títulos de funciones.....	92

Figura 4.29: Símbolos estáticos referentes a títulos de funciones.....	93
Figura 4.30: Símbolos estáticos desplegados al terminar la ejecución de un proceso.....	93
Figura 4.31: Símbolo dinámico mostrando un mensaje al ejecutar el proceso.....	94
Figura 4.32: Mensajes de estado del sistema.....	94
Figura 4.33: Ventana de pop-up para descarga de resultado.....	95

Índice de Tablas

Tabla 1.1: Metodología de investigación a lo largo del trabajo.....	4
Tabla 2.1: Checklist para evaluaciones de Usabilidad en Grid Computing.....	17
Tabla 2.2: Mapeo para las Heurísticas Específicas para aplicaciones del tipo Grid Computing.....	19
Tabla 2.3: Total de problemas encontrados en GreenView.....	21
Tabla 2.4: Total de problemas encontrados en GreenLand.....	22
Tabla 3.1: Listado de problemas encontrados exclusivamente con las heurísticas genéricas de Nielsen. Caso de Estudio GreenView.....	28
Tabla 3.2: Listado de problemas encontrados exclusivamente con las heurísticas genéricas de Nielsen. Caso de Estudio GreenLand.....	29
Tabla 3.3: Ranking de problemas detectados exclusivamente con heurísticas genéricas de Nielsen para los productos GreenView y GreenLand.....	30
Tabla 4.1: Listado de problemas encontrados exclusivamente con las heurísticas genéricas de Nielsen reclasificados en las heurísticas específicas para aplicaciones del tipo Grid Computing. Caso de Estudio GreenView.....	49
Tabla 4.2: Listado de problemas encontrados exclusivamente con las heurísticas genéricas de Nielsen reclasificados en las heurísticas específicas para aplicaciones del tipo Grid Computing. Caso de Estudio GreenLand.....	50
Tabla 4.3: Nuevo listado general para heurísticas para aplicaciones del tipo Grid Computing.....	52
Tabla 4.4: Mapeo entre las la nueva lista de heurísticas específicas para aplicaciones del tipo Grid Computing y las heurísticas genéricas de Nielsen.....	53
Tabla 4.5: Checklist para heurísticas revisadas específicas para aplicaciones del tipo Grid Computing.....	54
Tabla 4.6: Perfiles de sujetos de prueba en evaluación de Comunicabilidad.....	59
Tabla 4.7: Quiebres Comunicacionales ocurridos en la prueba con usuarios.....	60
Tabla 4.8: Cuantificación de quiebres comunicacionales en GreenView tras la prueba con usuarios.....	61
Tabla 4.9: Perfiles de evaluadores para el cliente Boinc.....	63
Tabla 4.10: Incumplimiento de principios por los problemas detectados.....	63
Tabla 4.11: Ranking de problemas encontrados en el cliente Boinc según Criticidad.....	64

Tabla 4.12: Perfiles de evaluadores.....	66
Tabla 4.13: Incumplimiento de principios (Nielsen) por los problemas detectados.....	66
Tabla 4.14: Ranking de problemas encontrados en el cliente Boinc según Criticidad.....	67
Tabla 4.15: Perfiles de evaluadores para GreenLand v2.....	79
Tabla 4.16: Incumplimiento de principios por los problemas detectados en GreenLand v2.....	80
Tabla 4.17: Problemas encontrados en GreenLand v2 ordenados por promedio de Criticidad.....	80
Tabla 4.18: Perfiles de evaluadores (Nielsen).....	82
Tabla 4.19: Incumplimiento de principios (Nielsen) por los problemas detectados.....	83
Tabla 4.20: Problemas encontrados en GreenLand v2 ordenados por promedio de Criticidad.....	83
Tabla 4.21: Perfiles de usuario para prueba de Usabilidad en producto GreenLand v2.....	85
Tabla 4.22: Tareas realizadas por los sujetos de prueba.....	86
Tabla 4.23: Perfiles de sujetos de prueba en evaluación de Comunicabilidad.....	97
Tabla 4.24: Quiebres comunicacionales ocurridos en la prueba con usuarios.....	97
Tabla 4.25: Cuantificación de quiebres comunicacionales en GreenLand v2 tras la prueba con usuarios.....	98
Tabla 4.26: Relaciones entre quiebres comunicacionales y heurísticas de Usabilidad específicas.....	100
Tabla 4.27: Relaciones entre heurísticas de Usabilidad y quiebres comunicacionales.....	101
Tabla 4.28: Posibles quiebres comunicacionales asociados a problemas de Usabilidad en el producto GreenView.....	102
Tabla 4.29: Cuantificación de posibles quiebres comunicacionales en GreenView asociados a problemas de Usabilidad.....	103
Tabla 4.30: Posibles quiebres comunicacionales asociados a problemas de Usabilidad en el producto Boinc.....	104
Tabla 4.31: Cuantificación de posibles quiebres comunicacionales en el producto Boinc asociados a problemas de Usabilidad.....	106
Tabla 4.32: Posibles quiebres comunicacionales asociados a problemas de Usabilidad en el producto GreenLand v2.....	107
Tabla 4.33: Cuantificación de posibles quiebres comunicacionales en el producto GreenLand v2 asociados a problemas de Usabilidad.....	108

Tabla 4.34: Resumen de heurísticas de Usabilidad asociadas a problemas de Comunicabilidad más importantes.....109

1. Presentación del Tema

1.1 Introducción

Grid Computing es una tecnología emergente de computación distribuida, que permite conectar equipos geográficamente distantes en una red especial llamada *Grid*. Este tipo de *grids* o redes proporcionan la capacidad de compartir recursos de distinto tipo y de manera transparente [1] [5]. Esta nueva tecnología ataca la problemática relacionada con una necesidad creciente de realizar tareas que requieren de una gran capacidad de procesamiento, con lo que se hace necesario el poder repartir carga de distintos servicios y de manera transparente para el usuario.

Actualmente el uso de software del tipo *Grid Computing* se limita a usuarios experimentados, dada la complejidad técnica de su infraestructura. Dentro de este marco es que surge la necesidad de mejorar la Usabilidad de estos productos de software, permitiendo así que una mayor cantidad de usuarios no técnicos comprendan y utilicen de manera eficaz estos sistemas.

La Usabilidad se define como un atributo de calidad que hace referencia al entendimiento y al uso efectivo y eficiente del software [2]. En función de ello es que resulta un elemento determinante en el éxito o fracaso de un producto de software, pues si éste no es entendido por los usuarios no es utilizado de manera satisfactoria para así cumplir con sus necesidades. En torno a este atributo existen muchos elementos que por su naturaleza son subjetivos, por lo que es necesario usar metodologías específicas que permitan analizar y cuantificar la Usabilidad.

De manera complementaria, la Comunicabilidad es la “cualidad distintiva de sistemas interactivos basados en software de comunicarse de manera efectiva y eficiente su intención en el diseño y sus principios de interactividad subyacentes al sistema” [6]. De este modo, la Comunicabilidad implica que el usuario comprenda mensaje enviado al usuario por diseñador a través del propio diseño y las funciones interactivas que el software contiene, lográndose este propósito y que sea de la mejor manera posible [6]. Es decir, la intención es la comprensión del mensaje entregado por el diseñador hacia el usuario, de modo que este comprenda el propósito y la manera de uso de un software, a través de la simbología entregada por la interfaz de un determinado producto de software.

La Comunicabilidad puede ayudar a la comprensión del software de *Grid Computing*, de manera complementaria a la Usabilidad. Sin embargo, no existen estudios acabados sobre el real impacto que provoca la Comunicabilidad sobre la Usabilidad en el ámbito de *Grid Computing*

1.2. Discusión Bibliográfica

El primer concepto importante de este trabajo es la Usabilidad y en lo que a bibliografía se refiere, hay que nombrar a Jakob Nielsen, Doctor en Interacción Persona-Computador de la Universidad Técnica de Dinamarca, en Copenhague. A través de sus publicaciones ha evidenciado los problemas asociados a la Usabilidad en los productos de software.

El segundo concepto importante es la Comunicabilidad, tema para lo cual se puede citar a la Dra. Clarisse Sieckenius de Souza, de la Pontificia Universidad Católica de Río de Janeiro, considerada uno de los exponentes más importantes en torno a la Ingeniería Semiótica y la Comunicabilidad en sistemas de software interactivos. Su obra resume las definiciones y metodología para evaluar Comunicabilidad.

Como tercer tema, hay que hacer mención a la tecnología *Grid Computing*, donde su principal expositor es el Dr. Ian Foster, profesor distinguido en Ciencias de la Computación en el Argon Institute y la Universidad de Chicago, en sus publicaciones expone las características de las redes de *Grid Computing*.

Otros expositores han desarrollado nuevas publicaciones complementando lo expuesto por los autores anteriormente citados, generando así mayor investigación y conocimientos en torno a las temáticas en cuestión. Dentro de algunos, cabe destacar el trabajo de título “Usabilidad en Aplicaciones Grid Computing” [1], utilizado como base de esta investigación. Además hay que mencionar el trabajo de Usabilidad en el tema de *Grid Computing* con académicos de la Technical University of Cluj-Napoca, Rumania, liderado por el Dr. Dorian Gorgan, con quien se han generado una serie de publicaciones.

1.3 Análisis de Objetivos y Metodología

1.3.1 Objetivo General

Formalizar una propuesta de heurísticas para realizar evaluaciones de Usabilidad en aplicaciones del tipo *Grid Computing* y evaluar el impacto de la Comunicabilidad sobre la Usabilidad en torno a este tipo de aplicaciones.

1.3.2 Objetivos Específicos

- Evaluar y refinar las heurísticas de Usabilidad existentes para aplicaciones *Grid Computing*.

- Evaluar y refinar el checklist referente a las heurísticas de Usabilidad existentes para aplicaciones *Grid Computing*.
- Analizar el impacto de la Comunicabilidad sobre la Usabilidad en *Grid Computing*.

1.3.3 Metodología

Para esta investigación, se analizará una problemática en torno a la Usabilidad en productos *Grid Computing*. Para realizar esta investigación se efectúan una serie de hipótesis, para finalmente inferir ciertas conclusiones extraídas de los datos, los cuales en varias instancias son subjetivos.

Dados los antecedentes expuestos, se utiliza para realizar la investigación, un enfoque cualitativo, realizando las siguientes tareas [7]:

- Planteamiento del problema
 - Definición de la problemática actual en cuestión
 - Definición de objetivos generales y específicos
- Revisión de la literatura
 - Revisión de literatura específica sobre la problemática
 - Revisión de trabajos previos
- Recolección de los datos
 - Revisión de datos recolectados de trabajos previos
 - Desarrollo de nuevos experimentos
- Análisis de los datos
 - Análisis de datos de nuevos experimentos
 - Análisis de heurísticas previas y estudio de mejoras
 - Análisis de incidencia de Comunicabilidad sobre Usabilidad
- Reporte de resultados
 - Resultados de análisis de heurísticas de Usabilidad en *Grid Computing*
 - Resultados de análisis de Comunicabilidad en productos *Grid Computing*

Tabla 1.1: Metodología de investigación a lo largo del trabajo.

Etapa	Sección
Planteamiento del problema	
Definición de la problemática actual en cuestión	1.1
Definición de objetivos generales y específicos	1.3.1, 1.3.2
Revisión de la literatura	
Revisión de literatura específica sobre la problemática	1.2, 2.1, 2.2.1, 2.3.1, 2.3.2, 2.3.4, 2.3.5
Revisión de trabajos previos	2.2.2, 2.2.3, 2.2.4
Recolección de los datos	
Revisión de datos recolectados de trabajos previos	2.2.5
Desarrollo de nuevos experimentos	3.2.1, 3.2.2, 4.2
Análisis de los datos	
Análisis de datos de nuevos experimentos	3.2.3, 4.2.3, 4.2.5
Análisis de heurísticas previas y estudio de mejoras	3.1, 4.1
Análisis de incidencia de Comunicabilidad sobre Usabilidad	4.3
Reporte de resultados	
Resultados de análisis de heurísticas de Usabilidad en <i>Grid Computing</i>	4.2.2, 4.2.3, 4.2.4, 4.2.5, 4.2.6, 5
Resultados de análisis de Comunicabilidad en productos <i>Grid Computing</i>	3.2.1, 3.2.2, 3.2.3, 4.2.1, 4.2.7, 4.2.8, 5

1.4 Guía de Secciones

En este apartado se describe el contenido de las secciones del presente trabajo, con el propósito de brindar una visión global del proceso relacionado con el desarrollo de éste.

- 1. Presentación del tema:** Introducción, objetivos y metodología utilizada para desarrollar el tema.
- 2. Marco teórico, Estado del arte:** Describe qué es la tecnología *Grid Computing*, su arquitectura y clasificaciones. Explica los conceptos de Usabilidad y Comunicabilidad como atributos de calidad de software, además se describen los métodos actuales de experimentación en torno a éstos.
- 3. Solución propuesta:** Describe una solución tentativa a los temas delineados en los objetivos propuestos. Se revisan experimentos sobre productos de software del tipo *Grid Computing* realizados en una investigación anterior [1] y se realizan algunos nuevos en torno a la Comunicabilidad, con el objetivo de obtener un acercamiento inicial a la solución tentativa que se pretende llegar.
- 4. Desarrollo de la solución:** Se realizan nuevos experimentos tanto de Usabilidad como de Comunicabilidad sobre productos de software del tipo *Grid Computing*, tanto en productos analizados anteriormente como en

nuevos. Se analizan los datos resultantes de estos nuevos experimentos y se revisan las heurísticas específicas de Usabilidad para aplicaciones del tipo *Grid Computing* de un trabajo anterior [1] refinándose su checklist y creando una nueva heurística. Se revisan y analizan los resultados en los experimentos de Comunicabilidad y se explica la relación con las heurísticas de Usabilidad tanto en los mensajes meta-comunicacionales como en los quiebres, llegando a una serie de tablas donde se observa el calce entre estos conceptos.

5. Conclusiones: Observaciones finales, conclusiones de este trabajo tanto en relación con los objetivos de éste como en otros temas que salieron a la luz a lo largo del desarrollo.

6. Bibliografía

7. Anexos

2. Marco Teórico: Estado del Arte

2.1 La Tecnología Grid Computing

Con el pasar de los años, las tecnologías presentan un crecimiento exponencial lo que trae consigo nuevas posibilidades y retos. Uno de estos es el procesamiento de grandes volúmenes de información, que requiere de cálculos complejos, que con el uso de una arquitectura tradicional y centralizada son imposibles de resolver. Junto con esto, las organizaciones crecen y se expanden requiriendo diversos servicios computacionales los cuales deben ser transparentes para los usuarios. Para abarcar estas problemáticas, surgen diversas tecnologías, entre las cuales se destaca *Grid Computing*.

El término *Grid* o Red es un concepto propuesto por Ian Foster y Carl Kesselman haciendo similitud con una red eléctrica. En una red o malla eléctrica (*Power Grid*) el usuario se conecta y hace uso de la energía (recurso o servicio), sin importar de qué plantas de energía proviene ésta o cuál es la estructura de la red eléctrica [10].

La primera definición propuesta por Ian Foster y Carl Kesselman señala que una “Grid Computacional es una infraestructura de hardware y software que provee de un acceso confiable, consistente, generalizado y de bajo costo a las capacidades de cómputo de alto nivel” [11]. Posteriormente, los autores redefinieron y ampliaron el concepto, el cual puede descomponerse en tres partes o elementos clave. Estos se explican a continuación [11]:

- **Coordina los recursos distribuidos:** Una Grid integra y coordina los recursos y los usuarios que viven dentro de dominios de control diferentes, por ejemplo, el escritorio del usuario en comparación con un sistema computación central, diferentes unidades administrativas de la misma empresa y/o de diferentes empresas. De lo contrario, se trataría de un sistema de administración local.
- **Usando un estándar, abierto, haciendo uso de protocolos e interfaces de propósito general:** Una Grid es construida a partir de múltiples protocolos e interfaces que se ocupan de temas tan fundamentales como la autenticación, autorización, el descubrimiento de recursos y acceso a los recursos. Es importante que estos protocolos e interfaces sean abiertos y estándar. De lo contrario, se trata de un sistema de aplicación específica.
- **Para ofrecer calidad de servicio no trivial:** Una Grid permite a los recursos que la componen, ser utilizados en forma coordinada para ofrecer diferentes calidades de servicio-relativos. Por ejemplo, tiempo de respuesta, rendimiento, disponibilidad y la seguridad y/o la utilización de múltiples tipos de recursos para

satisfacer las distintas y complejas demandas del usuario, por lo que la utilidad del sistema combinado es significativamente mayor que la de la suma de sus partes.

Grid Computing, surge como una tecnología que permite conectar una gran cantidad de equipos geográficamente dispersos, los cuales de manera coordinada, procesan grandes cantidades de información y pueden ofrecer una amplia gama de recursos y servicios, sin estar sujetos a un control centralizado. De esta manera el sistema funciona como un gran supercomputador que puede proporcionar un poder de procesamiento enorme. Además permite entregar los diversos recursos y servicios a los usuarios que se conectan, de manera transparente, sin que ellos sepan que están bajo un sistema con una arquitectura especial.



Figura 2.1: Representación gráfica de un sistema de *Grid Computing* [1].

2.1.1 Características de los Sistemas Grid Computing

Los sistemas *Grid Computing* tienen una serie de características. Dentro de ellas, se pueden observar algunas que son propias de los sistemas de computación distribuida [1]:

- **Transparencia/abstracción:** Una Grid entrega una capa de abstracción, proporcionada por el middleware, para integrar los diversos componentes que forman parte del sistema, muchos de los cuales son heterogéneos. Para esto el sistema hace uso de métodos que permiten eliminar dependencias locales para compartir recursos.
- **Compartición de recursos:** Haciendo uso de la capacidad de abstracción se pueden compartir recursos en gran escala entre los diversos nodos del sistema. Estos nodos son distribuidos en distintas localizaciones

geográficas y pueden llegar a pertenecer a distintas organizaciones. De esta manera los sistemas *Grid Computing* pueden extender otros, los que pueden colaborar y coordinarse para resolver problemas entre los diversos nodos.

- **Flexibilidad:** La estructura de un ambiente de *Grid* no es rígida, en lugar de eso utiliza funcionalidades modulares que pueden ir variando en el tiempo de forma dinámica. Los ambientes *Grid*, además, son flexibles en la gestión y administración de los distintos recursos que dispone. Por ejemplo, en la medida en que realizan trabajos o tareas planificadas, se reparte la carga de manera dinámica y en tiempo real, coordinando el trabajo entre los distintos nodos.
- **Administración y control descentralizado:** Esta es una característica particular de las *Grid* que las diferencia de otros sistemas similares como los clústers. Esta característica hace mención a la capacidad de invocar a los distintos recursos y servicios del sistema de manera independiente a los procesos que requieran de ellos. Así, en conjunto con la característica de flexibilidad, se logra una administración y control de los recursos de manera descentralizada y dinámica.
- **Escalabilidad:** Una estructura *Grid* puede comenzar a implementarse de manera local, dentro de una empresa u organización, que puede estar distribuida en zonas geográficas distintas. Ayudándose de las características de compartición de recursos y transparencia/abstracción, se logra una plataforma que permite ir agregando recursos al sistema sin que los usuarios noten el cambio en la estructura. Gracias a las características de flexibilidad, y administración y control descentralizado, el sistema se adapta de manera dinámica a los nuevos elementos sin reducir su rendimiento.
- **Alto rendimiento:** Al existir una estructura, con una capacidad de escalabilidad teóricamente infinita, un sistema de *Grid Computing* puede albergar servicios que entregan un rendimiento y capacidad de cómputo extremadamente alto, con posibilidades teóricamente infinitas y en comparación a otros sistemas, con un bajo costo.
- **Generalización:** Dadas las características anteriormente explicadas, se puede tener un sistema que puede extenderse a una infinidad de ambientes y dispositivos.
- **Personalización:** En un sistema de *Grid Computing*, gracias a su capacidad de escalabilidad, se pueden ir agregando y modificando los recursos y servicios que el sistema provee, de acuerdo las necesidades de los usuarios. De esta manera se puede tener un sistema altamente específico y personalizado en razón de los distintos requerimientos.

- **Heterogeneidad:** Un sistema *Grid Computing*, puede tener una serie de recursos inmensamente variados, tanto de hardware como de software. Por ejemplo, distintos computadores, con hardware distinto, otros tipos de máquinas como instrumentación científica, distintos sistemas operativos, distintos administradores de bases de datos, etc.

2.1.2 Arquitectura de los Sistemas Grid Computing

Un sistema de *Grid Computing* tiene una arquitectura de capas, las cuales están basadas en los distintos protocolos que permiten a los usuarios hacer uso de los recursos y servicios entregados por el sistema. Esta arquitectura cuenta de cinco capas, las cuales se explican a continuación [12]:

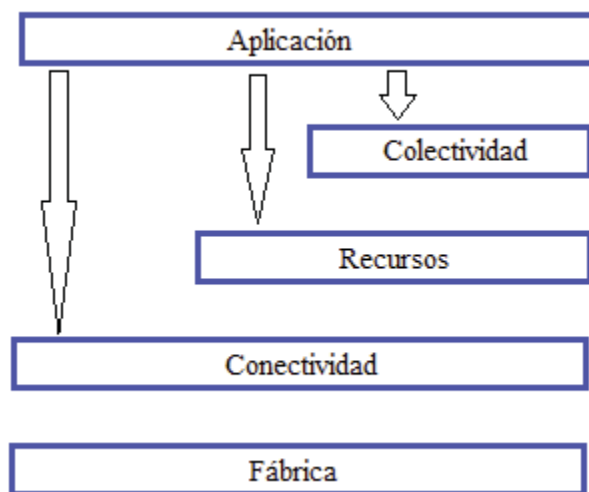


Figura 2.2: Arquitectura de un sistema de Grid Computing.

- **Capa de fábrica:** En esta capa se encuentra la infraestructura, los recursos computacionales base sobre los cuales se monta el sistema, como por ejemplo los equipos servidores, clústers, supercomputadores, maquinaria diversa perteneciente al sistema, medios de almacenamiento, la red, etc.
- **Capa de conectividad:** Aquí se encuentran los diversos protocolos de comunicación entre las máquinas, como por ejemplo TCP/IP, TCP, UDP, SSL, etc. En esta capa también se encuentran los sistemas de autenticación de usuarios.

- **Capa de recursos:** Esta capa es la encargada de reunir la información acerca de la estructura y el estado de recursos del sistema que están disponibles, como por ejemplo su configuración, carga, políticas de repartición de recursos, etc. Esta capa, además es la encargada de realizar la gestión y administración de estos recursos en torno a las peticiones de los usuarios.
- **Capa de colectividad:** Esta capa tiene una serie de servicios que apoyan la gestión y coordinación de los recursos, como son: servicios de directorio, servicios de descubrimiento de software, schedulers distribuidos, servicios de replicación de datos, monitorización y diagnóstico de la ejecución de las tareas, administración de accesos a datos distribuidos, etc.
- **Capa de aplicación:** En esta capa se encuentran los protocolos y el software que va a comunicar a las demás capas. En esta capa es importante hacer mención al middleware que va a realizar la función anteriormente explicada, otorgando la abstracción/transparencia entre el usuario y los distintos recursos y servicios que entrega el sistema.

2.1.3 Principales Usos de la Tecnología Grid Computing

Esta tecnología, además de solucionar problemas que requieren una gran capacidad de cómputo, permite unificar e integrar distintos recursos tecnológicos y así proveer soporte para una gran variedad de aplicaciones. Los usos pueden categorizarse en función de la problemática que abordan, según la siguiente clasificación [5]:

- **Soporte a la supercomputación distribuida:** Permite reducir los tiempos en la ejecución de tareas, además de poder realizar otras tareas altamente complejas que no serían posible en un sistema simple o de una sola máquina.
- **Soporte a la computación de alto rendimiento:** Permite hacer uso de ciclos de procesador que se encuentren sin uso en el sistema para realizar tareas independientes.
- **Soporte a la computación bajo demanda:** Permite usar la *Grid* para obtener recursos del sistema cuando no es rentable, en materia de costos de procesamiento, hacer uso de los recursos locales existentes.
- **Soporte a la computación intensiva en datos:** Permite a las aplicaciones utilizar la *Grid* para sintetizar nueva información de repositorios de datos distribuidos, bibliotecas digitales y bases de datos.
- **Soporte a la computación colaborativa:** Permite a las aplicaciones utilizar la *Grid* para posibilitar y mejorar las interacciones entre personas de una manera sincrónica o asincrónica a través de un espacio virtual.

- **Soporte a la computación multimedia:** Permite entregar contenidos multimedia entre distintos participantes, asegurando una calidad de servicio.

2.1.4 Tipos de Grid

Existen numerosos tipos de clasificaciones, según distintos autores. A continuación se detalla una clasificación según estructura organizacional [13]:

Grids Departamentales

Este tipo de Grid se ha desarrollado con el objetivo de resolver los problemas de un determinado grupo de personas dentro de una empresa. Los recursos no son compartidos por otros grupos dentro de la misma empresa. A continuación se muestra una subclasificación de este tipo de *Grids*:

- **Clúster grids:** Consiste en uno o más sistemas trabajando en conjunto para proporcionar un único punto de acceso a los usuarios. Es utilizado por un equipo relacionado a un sólo proyecto y se puede brindar apoyo en tareas que requieran una alta capacidad de cómputo.
- **Infra grids:** Corresponden a *Grids* que optimizan los recursos computacionales dentro de una empresa. Pueden pertenecer a alguna sección específica de ésta.

Grids de Empresa

Esta clase de Grid está compuesta por los recursos distribuidos en una empresa y provee servicio a todos los usuarios de ésta. A continuación se detalla una subclasificación de este tipo de redes:

- **Grids empresa:** Este tipo de Grid se implementa en grandes empresas que tienen una presencia global o que precisan acceso a los recursos fuera de una ubicación geográfica corporativa única. Las Grids Empresa corren detrás del firewall corporativo.
- **Grids intra:** La función de este tipo de Grid es el intercambio de recursos entre las distintas secciones o departamentos de una empresa. Una Grid intra puede ser local o recorrer la organización en distintas zonas.
- **Grids campus:** Una Grid de este tipo permite que múltiples departamentos puedan compartir los recursos informáticos de una manera cooperativa. Los elementos de este tipo de Grids pueden consistir en estaciones de trabajo y servidores, así pueden existir recursos centralizados con sede en múltiples dominios administrativos, en los departamentos, o a través de la empresa.

Grids Extra-empresas

Este tipo de Grid se implementa entre las empresas, sus socios y sus clientes. Los recursos de la red están disponibles generalmente a través de una red privada virtual. A continuación se presenta una subclasificación de este tipo de *Grids*:

- **Extra grids:** Habilitan el uso compartido de los recursos con los asociados externos. Esto supone que la conectividad entre dos o más empresas, es a través de algún servicio de confianza, como una red privada o una red privada virtual.
- **Grids de socios:** Son Grids entre organizaciones pertenecientes a mercados similares, que tienen la necesidad de colaborar en proyectos y compartir recursos como medio para alcanzar un objetivo en común.

Grids Globales

Las Grids establecidas a través de Internet público constituyen las Grids globales. Las organizaciones pueden utilizar este tipo de grid para facilitar sus negocios o comprar en parte, o en su totalidad, capacidad de cómputo a proveedores de estos servicios. A continuación se detalla una subclasificación de este tipo de Grids:

- **Grids globales:** Permiten a los usuarios aprovechar los recursos externos que las Grids globales proporcionan, entregando recursos distribuidos a los usuarios en cualquier parte del mundo.
- **Inter grids:** Proporcionan capacidad de cómputo, de compartir datos y recursos de almacenamiento a través de la Web pública. Esto puede implicar el intercambio de recursos con otras empresas y la compra o venta del exceso de capacidad.

Grids de Cálculo

Esta clase de Grid facilita el acceso a los recursos computacionales, pudiendo clasificarse de acuerdo al tipo de hardware computacional implementado.

- **Grids de escritorio:** Son Grids que aprovechan los recursos informáticos de las computadoras de escritorio.
- **Grids servidor:** Algunas empresas crean estas Grids para dar soporte a sus diversos departamentos, por lo general sobre un sistema operativo Unix/Linux.
- **Grids alto-rendimiento/clúster:** Estas Grids corresponden a sistemas de alto nivel de procesamiento, tales como supercomputadores o clústers.

Grids de Datos

Grids especialmente implementadas para el acceso y análisis de grandes volúmenes de datos. Estas Grids están optimizadas para operaciones relacionadas con el tratamiento de datos, requiriendo de una gran capacidad de almacenamiento.

Grids de Utilidad

Este tipo de Grid tiene una finalidad exclusivamente comercial, permitiendo la venta a otras organizaciones de capacidad de cómputo, distintos recursos y servicios.

2.1.5 Tipos de Usuarios

Los distintos perfiles se organizan de acuerdo a los conocimientos específicos y tareas de los usuarios. A continuación se mencionan los distintos tipos [14]:

- **Usuario final de servicios:** Tiene poco conocimiento técnico, sus tareas se limitan a ingresar información al sistema y ejecutar consultas.
- **Usuario final de servicios ejecutador:** Tiene un poco más de conocimiento técnico que el anterior, lo cual le permite ejecutar tareas en el sistema.
- **Usuario poderoso agnóstico de nodo de recursos:** Posee un alto conocimiento técnico el cual le permite realizar el desarrollo del sistema a nivel general de éste. Este tipo de usuario no tiene interés en los nodos específicos que proveen los recursos.
- **Usuario poderoso con requerimientos de recursos específicos en los nodos:** Es tipo de usuario es similar al anterior, se diferencia de este último en que sus tareas tienen relación con nodos específicos que entregan ciertos recursos.
- **Usuario poderoso desarrollador de servicios:** Este usuario tiene un alto conocimiento técnico y su tarea principal es el desarrollo de servicios en el sistema.
- **Proveedor de servicio:** Usuario de alto conocimiento técnico, sus tareas están relacionadas con la administración del acceso al sistema y gestión de las identidades.

- **Administrador de infraestructura del sistema:** Posee un alto conocimiento técnico y su tarea es administrar los nodos pertenecientes al sistema.

2.2 Usabilidad en Productos de Software

La Usabilidad es un atributo de calidad de software que en la actualidad tiene escaso desarrollo o bien éste es con poca profundidad. Este atributo se centra en el entendimiento, la eficiencia y eficacia de uso de un producto de software bajo ciertas condiciones específicas [2]. Cabe señalar que la Usabilidad no se considera como un atributo unidimensional, ya que integra los siguientes componentes o atributos [8]:

- **Facilidad de aprendizaje:** El sistema debe ser fácil de aprender, de modo que el usuario rápidamente pueda comenzar a trabajar con él.
- **Eficiencia:** El uso del sistema debe ser eficiente una vez que el usuario ha aprendido a utilizarlo, de esta manera el sistema debe ofrecer un alto nivel de productividad.
- **Facilidad de recordar:** El uso del sistema debe ser fácil de recordar, de modo que un usuario que ha dejado de utilizar el sistema durante un tiempo, al volver a hacerlo no deba aprender nuevamente su uso.
- **Errores:** El sistema debe tener una baja tasa de errores y en el caso de que ocurran, los usuarios deben poder recuperarse fácilmente de ellos. En el sistema tampoco deben ocurrir errores catastróficos.
- **Satisfacción:** El uso del sistema debe ser agradable para el usuario, es una satisfacción subjetiva.

2.2.1 Evaluaciones de Usabilidad

Los métodos de evaluación para Usabilidad se dividen en dos grupos: Los métodos de inspección, que detectan problemas de Usabilidad basándose en la experticia de profesionales; y los de pruebas, que reconocen problemas mientras se observa a usuarios utilizando el software [4].

J. Nielsen definió diez heurísticas de Usabilidad, ampliamente conocidas, para productos de software. A continuación se detalla cada una de ellas [9]:

- **(N1) Visibilidad del estado del sistema:** El sistema debe mantener informado al usuario sobre aquello que está realizando.
- **(N2) Similitud entre el sistema y el mundo real:** El sistema debe seguir convenciones del mundo real, utilizando un lenguaje que sea propio del usuario, es decir, basado en el uso de frases y conceptos familiares.
- **(N3) Control y libertad del usuario:** El usuario debe percibir que es él quien tiene el control del sistema y no vice-versa, para ello debe contar con “salidas de emergencia” que permitan “deshacer” o “rehacer” las tareas que se llevan a cabo.
- **(N4) Consistencia y estándares:** Debe mantener las convenciones y estándares durante toda la ejecución del sistema en torno a estilos, palabras, comportamientos.
- **(N5) Prevención de errores:** El diseño debe ser suficientemente cuidadoso como para evitar errores por parte del usuario. Por ejemplo, es mejor mostrar una lista de opciones o menú, en lugar de que el usuario deba escribir algo.
- **(N6) Reconocimiento más que memoria:** Se debe reducir la carga de memoria, facilitando la ubicación a través de controles y opciones visibles para el usuario, de esta manera el usuario no debe recordar cómo efectuar las acciones.
- **(N7) Flexibilidad y eficiencia de uso:** El sistema debe acomodarse a los distintos estilos de trabajo proporcionando aceleradores, adaptación de controles, etc.
- **(N8) Diseño estético y minimalista:** Se debe eliminar u ocultar información que no sea relevante para el usuario.
- **(N9) Ayuda al usuario para reconocer, diagnosticar y recuperarse de errores:** Los mensajes de error deben ser claros y entendibles por el usuario.
- **(N10) Ayuda y documentación:** Deben existir tutoriales y ayuda para la utilización del sistema.

Como ya se mencionó anteriormente, la evaluación heurística es un método que permite detectar problemas de Usabilidad. Un grupo de evaluadores expertos verifican el cumplimiento del listado de heurísticas. Basándose en el cumplimiento de éstas se identifican los problemas [8].

Para realizar una evaluación heurística se deben seguir los siguientes pasos:

1. **Identificación de problemas:** Cada evaluador, según una lista de heurísticas, busca problemas en el software para luego elaborar un listado y descripción de cada uno de ellos.

2. **Consenso de problemas:** Los evaluadores se reúnen y discuten los problemas para finalmente generar una lista en común.
3. **Evaluación:** Tras realizar una lista en común, cada evaluador asigna una nota de acuerdo a la severidad y frecuencia de los problemas listados.
4. **Conclusiones:** A partir de las notas asignadas, se analizan los problemas y se generan conclusiones y posibles soluciones a los problemas más críticos.

2.2.2 Heurísticas de Usabilidad en productos de Grid Computing

Las heurísticas de Nielsen, anteriormente descritas, podrían no abarcar problemas de clases o tipos específicos de software. Para solucionar esto, se diseñaron nuevas heurísticas específicas para los productos de *Grid Computing*, estas se agrupan en tres categorías [1]:

Heurísticas de Diseño y estética

- **(H1) Claridad:** La interfaz de una aplicación de *Grid Computing* debe ser fácil de entender, utilizando elementos gráficos, texto y lenguaje claro.
- **(H2) Uso de metáforas:** Una aplicación de *Grid Computing* debe usar metáforas apropiadas, de manera que el usuario fácilmente entienda las acciones posibles, a través del uso de imágenes y objetos familiares.
- **(H3) Simplicidad:** Una aplicación de *Grid Computing* debe proveer de la información necesaria para concretar una tarea de manera concisa (clara).
- **(H4) Retroalimentación:** Una aplicación de *Grid Computing* debe mantener al usuario informado acerca del progreso de los “Jobs”, indicando tanto el estado global como detallado del sistema. La aplicación debe entregar la retroalimentación necesaria acerca de las acciones del usuario.
- **(H5) Consistencia:** Una aplicación de *Grid Computing* debe ser consistente en el uso del lenguaje y los conceptos. Las formas de ingresar y visualizar los resultados deben ser consistentes.

Heurísticas de navegación

- **(H6) Atajos para funcionalidades:** Una aplicación de *Grid Computing* debe proveer de atajos y abreviaciones, teclas de acceso rápido o líneas de comando para usuarios expertos.

- **(H7) Reducir carga de memoria:** Una aplicación de *Grid Computing* debe mantener los comandos principales siempre disponibles. Se debe ofrecer facilidad para encontrar elementos, funciones y opciones.
- **(H8) Explorabilidad:** Una aplicación de *Grid Computing* debe minimizar la navegación y debe proveer maneras de realizar tareas fáciles, claras y naturales.
- **(H9) Control sobre acciones:** Una aplicación de *Grid Computing* debe proveer maneras de cancelar una tarea o proceso que se está ejecutando. Debe permitir deshacer y/o realizar cambios de acciones.

Heurísticas de errores y ayuda

- **(H10) Prevención de errores:** Una aplicación de *Grid Computing* debe prevenir que los usuarios realicen acciones que puedan producir errores, y debe evitar confusiones que puedan llevar a estos.
- **(H11) Ayuda para recuperación de errores:** Una aplicación de *Grid Computing* debe proveer de mensajes claros, indicando las causas de errores y sugiriendo algún tipo de solución ante el error.
- **(H12) Documentación de ayuda:** Una aplicación de *Grid Computing* debe tener una completa documentación en línea que sea fácil de encontrar y entender. Debe proveer de ayuda contextual y glosario de términos para usuarios novicios.

2.2.3 Checklist para la Evaluación de Usabilidad en Aplicaciones Grid Computing

El Checklist o lista de comprobación es una especificación complementaria a las heurísticas, que permite comprobar, en detalle, el cumplimiento de cada una de las heurísticas específicas en una evaluación.

En un trabajo investigativo anterior [1] se generaron las heurísticas específicas para *Grid Computing* y el respectivo checklist, este último se detalla en la Tabla 2.1 [1].

Tabla 2.1: Checklist para evaluaciones de Usabilidad en *Grid Computing* [1].

ID	Problema
H1	Claridad
H1.1	El propósito de la aplicación es claro para todos sus usuarios objetivo
H1.2	Los controles de interfaz y elementos gráficos son fáciles de entender y su propósito es claro

H1.3	El idioma de la interfaz es familiar a los usuarios objetivo
H1.4	El lenguaje de la interfaz es claro y fácil de entender
H2	Uso de Metáforas
H2.1	Se utilizan metáforas que ayudan a mejorar y agilizar la comprensión del usuario sobre la aplicación
H2.2	Se utilizan metáforas sólo para conceptos o tareas simples.
H2.3	Se evita usar metáforas para comprender conceptos complejos.
H2.4	Las metáforas son fáciles de entender.
H3	Simplicidad
H3.1	La interfaz es simple, no está sobrecargada de elementos.
H3.2	No existen íconos, controles, menús, gráficos, textos u otros elementos de interfaz redundantes
H3.3	No existen elementos de interfaz puramente decorativos que no cumplan una labor.
H4	Retroalimentación
H4.1	Existen indicadores claros del estado del sistema
H4.2	Existen indicadores claros del estado de las tareas.
H4.3	El sistema reacciona claramente ante acciones del usuario
H4.4	Los indicadores de procesamiento son dinámicos
H5	Consistencia
H5.1	El vocabulario, los controles y las imágenes se usan consistentemente a través de la interfaz de la aplicación
H5.2	Las formas de ingreso de datos mantienen consistencia
H5.3	Los resultados del procesamiento son presentados de manera consistente
H6	Atajos para Funcionalidades
H6.1	Existen atajos a módulos y funcionalidades básicas de la aplicación
H6.2	Existe la opción de usar líneas de comando
H7	Reducir Carga de Memoria
H7.1	Las funcionalidades principales se encuentran siempre visibles y accesibles
H7.2	Las funcionalidades y opciones son fáciles de encontrar
H7.3	Los formularios llenados por el usuario se mantienen en memoria
H8	Explorabilidad
H8.1	Existe una secuencia clara y explícita de pasos para la realización de cada una de las tarea
H8.2	La navegación es intuitiva, fácil de entender
H8.3	Los menús son consistentes y la selección de opciones conduce a efectos predecibles
H9	Control sobre Acciones
H9.1	Existe la posibilidad de cancelar acciones o tareas

H9.2	Existe la posibilidad de revertir acciones
H9.3	Las tareas canceladas se detienen de inmediato, entregando una retroalimentación apropiada
H10	Prevención de Errores
H10.1	Si se solicita la carga de archivos, se especifica claramente los tipos de archivo permitidos por el sistema
H10.2	Se especifican o ejemplifican claramente los valores permitidos
H10.3	Se valida todo el ingreso de datos
H10.4	Los mensajes de error son fáciles de entender.
H10.5	Existen asistentes (wizard) para realizar tareas comunes
H11	Ayuda para Recuperación de Errores
H11.1	Se muestran mensajes de error claros a partir de la ocurrencia de estos
H11.2	Los mensajes de error se orientan hacia la resolución de los problemas
H11.3	Existen asistentes (wizard) para solucionar problemas comunes de alta complejidad
H12	Documentación de Ayuda
H12.1	Existe documentación online
H12.2	Existe manual de usuario
H12.3	Existen instrucciones para tareas comunes o básicas
H12.4	La ayuda es fácil de encontrar
H12.5	Existe ayuda contextual que guíe al usuario respecto al uso de los distintos elementos existentes

Con el objetivo de clarificar el tipo de problemas que abarcan las heurísticas para aplicaciones del tipo *Grid Computing*, se diseñó un mapeo comparativo que muestra el cumplimiento de las heurísticas genéricas de Nielsen dentro de las heurísticas específicas para aplicaciones del tipo *Grid Computing* (Tabla 2.2) [1]:

Tabla 2.2: Mapeo para las Heurísticas Específicas para aplicaciones del tipo Grid Computing [1].

Heurísticas Grid Computing		Heurística de Nielsen asociada	
ID	Definición	ID	Definición
H1	Claridad	N2	Concordancia Entre el Sistema y el Mundo Real
H2	Uso de Metáforas		
H3	Simplicidad	N8	Diseño Estético y Minimalista
H4	Retroalimentación	N1	Visibilidad del Estado del Sistema
H5	Consistencia	N4	Consistencia y Estándares
H6	Atajos para Funcionalidades	N7	Flexibilidad y Eficiencia de Uso
H7	Reducir Carga de Memoria	N6	Reconocer Mejor que Recordar

H8	Explorabilidad	N3	Control y Libertad del Usuario
H9	Control sobre Acciones		
H10	Prevención de Errores	N5	Prevención de Errores
H11	Ayuda para Recuperación de Errores	N9	Ayuda al Usuario a Reconocer, Diagnosticar y Recuperarse de Errores
H12	Documentación de Ayuda	N10	Ayuda y Rocumentación

2.2.4 Experimentos de Usabilidad en aplicaciones del tipo Grid Computing

Se realizaron una serie de investigaciones basadas en experimentos sobre aplicaciones del tipo Grid Computing, específicamente GreenView y GreenLand, ambas relacionadas con el procesamiento de datos atmosféricos.

GreenView v3.1 es una de las aplicación ambiental del tipo *Grid Computing* que permite realizar estudios climáticos a través de mediciones satelitales de alta resolución. Permite, por ejemplo efectuar tareas de modelación de la contaminación y del impacto que los espacios urbanos ejercen sobre la vegetación.

GreenView v3.1 fue sondeado por dos grupos de evaluadores, cada grupo compuesto por cuatro examinadores con un nivel de experiencia similar (medio) en evaluaciones heurísticas. Valiéndose de una escala de severidad, con un máximo de 4 y mínimo de 0, el primer grupo evaluó el producto utilizando las heurísticas de Nielsen, mientras que el segundo grupo, las heurísticas específicas para aplicaciones del tipo *Grid Computing*. Cabe señalar que los examinadores no contaban con experiencia en Usabilidad de aplicaciones del tipo *Grid Computing*.

La totalidad de evaluadores identificó 29 problemas. Utilizando las heurísticas específicas para *Grid Computing*, se encontró una mayor cantidad de problemas en relación al uso de las heurísticas de Nielsen. A continuación se detallan los resultados obtenidos en este experimento de Usabilidad

- 11 problemas (38%) fueron identificados por ambos grupos de evaluadores.
- 12 problemas (41%) fueron identificados por el grupo de evaluadores que utilizó las heurísticas específicas para aplicaciones de Grid Computing.
- 6 problemas (21%) fueron identificados por el grupo que utilizó las heurísticas de Nielsen.

Tabla 2.3: Total de problemas encontrados en GreenView [4].

Grupo 1: Usando heurísticas de Grid Computing		Grupo 2: Usando Heurísticas de Nielsen	
<i>ID</i>	<i>Número de problemas</i>	<i>ID</i>	<i>Número de problemas</i>
H1	3	N2	1
H2	1		
H3	0	N8	3
H4	2	N1	2
H5	1	N4	1
H6	1	N7	0
H7	0	N6	0
H8	1	N3	1
H9	1		
H10	3	N5	2
H11	2	N9	1
H12	2	N10	1
Total:17		Total:12	

Los problemas identificados por el grupo que utilizó las heurísticas de Nielsen, fueron calificados con un puntaje de 2,5 dentro de la escala de severidad lo cual significa que están dentro del promedio. En tanto, los evaluadores que utilizaron las heurísticas específicas para aplicaciones del tipo *Grid Computing*, no detectaron 6 problemas menores que sí fueron encontrados con las heurísticas de Nielsen. Utilizando las heurísticas específicas para aplicaciones del tipo *Grid Computing*, la mayoría de los problemas fueron calificados como severos (puntajes sobre 2,5) o altamente severos (puntajes sobre 3). De esta manera cuando se utilizaron las heurísticas anteriormente señaladas, se encontraron problemas con mayor severidad que los que resultaron de la evaluación con las heurísticas de Nielsen.

GreenLand v1.2 es otro software que utiliza la tecnología *Grid Computing* para obtener mapas de elementos presentes en la naturaleza, como por ejemplo el terreno, el agua, el aire y la vegetación. Esta aplicación, al igual que GreenView v3.1, fue examinada por dos grupos de evaluadores. Cada grupo estaba integrado por tres examinadores con un nivel de experiencia similar (medio) en evaluaciones heurísticas, pero sin conocimientos acabados en Usabilidad de aplicaciones del tipo *Grid Computing*. El primer grupo evaluó el producto utilizando las heurísticas de Nielsen y el segundo, utilizando las heurísticas específicas para aplicaciones del tipo *Grid Computing*.

Los 6 evaluadores identificaron un total de 48 problemas. Utilizando las heurísticas específicas para *Grid Computing*, nuevamente se encontraron una mayor cantidad de problemas en relación al uso de las heurísticas de Nielsen:

- 14 problemas (29%) fueron identificados por ambos grupos de evaluadores.
- 22 problemas (46%) fueron identificados exclusivamente por el grupo de evaluadores que utilizo las heurísticas específicas para aplicaciones de Grid Computing.
- 12 problemas (25%) fueron identificados exclusivamente por el grupo que utilizó las heurísticas de Nielsen.

Tabla 2.4: Total de problemas encontrados en GreenLand [4].

Grupo 1: Usando heurísticas de Grid Computing		Grupo 2: Usando Heurísticas de Nielsen	
<i>ID</i>	<i>Número de problemas</i>	<i>ID</i>	<i>Número de problemas</i>
H1	4	N2	6
H2	2		
H3	2	N8	4
H4	5	N1	2
H5	5	N4	5
H6	0	N7	1
H7	0	N6	0
H8	2	N3	2
H9	3		
H10	0	N5	0
H11	1	N9	1
H12	2	N10	1
Total:26		Total:22	

De una totalidad de 12 problemas identificados exclusivamente con las heurísticas de Nielsen, la mayoría fueron considerados con puntaje dentro de la media (alrededor de 2) excepto 2 de los cuales fueron calificados con un puntaje muy alto (puntaje sobre 3.5). De manera contraria, de los 22 problemas encontrados exclusivamente haciendo uso de las heurísticas específicas para aplicaciones del tipo Grid Computing, la mayoría fueron identificados como severos (puntaje sobre 2,5) Incluso pudo identificarse 1 problema con severidad de 4 y 3 problemas con severidad de 3.67 (altamente severos dentro de una escala entre 0 y 4).

De las evaluaciones anteriormente descritas, se desprende que tanto para las aplicaciones GreenView v3.1 y GreenLand v1.2, las heurísticas específicas para aplicaciones del tipo *Grid Computing* muestran un mejor desempeño al detectar problemas, en comparación con las heurísticas de Nielsen.

2.3 La Ingeniería Semiótica y la Comunicabilidad

2.3.1 La Ingeniería Semiótica

La Ingeniería Semiótica es una nueva perspectiva o acercamiento orientado al desarrollo de interfaces de usuario. Esta considera la Interacción Persona-Computador como un proceso que posibilita el envío de mensajes entre diseñador y usuario a través del software. Tras la comprensión del propósito del producto de software, el diseñador formula una visión sobre éste, luego transmite un mensaje (visión del diseño) y que sea comprendido por el usuario. De esta manera, el usuario no sólo comprende el uso de controles, sino que también el propósito de estos y del diseño del producto de software.

La Ingeniería Semiótica se compone de dos elementos: El primero es la Semiótica, disciplina enfocada al estudio de los símbolos, el proceso de significado y como los signos y el significado toman parte de la comunicación. El segundo elemento es la Ingeniería, ciencia que entrega un uso práctico a la semiótica, ayudando a desarrollar las diversas metodologías enfocadas a la evaluación de productos de software. Tras estas aclaraciones se enuncia el objetivo de la Ingeniería Semiótica, el cual es la Comunicabilidad, para que finalmente su uso surta un efecto positivo en la calidad de los productos de software.

2.3.2 La Comunicabilidad

Formalmente, Comunicabilidad se define como la “cualidad distintiva de sistemas interactivos basados en software para comunicar de manera efectiva y eficiente su intención en el diseño y sus principios de interactividad subyacentes al sistema” [3]. Por tanto, la Comunicabilidad, en sistemas de software interactivos, debe permitir que se comprenda el mensaje que el diseñador envía al usuario a través del diseño del software y las funciones interactivas que éste contiene. De esta forma, a la Comunicabilidad se le atribuye como la “capacidad de un sistema de software interactivo para lograr de manera completa la Meta-Comunicación transmitiendo así la esencia del mensaje del diseñador”.

La Meta-Comunicación

La Meta-Comunicación es la “comunicación acerca de la comunicación”, la comunicación sobre el cómo, cuándo, dónde y el propósito del sistema. Para esto hay que tener en cuenta que el medio o canal por el cual el

diseñador comunica su mensaje es el sistema de software. Esta comunicación se realiza en medida de que el usuario pueda interpretar la simbología expuesta por el diseñador en el sistema, como pueden ser texto e íconos, los cuales pueden llegar a ser interpretado de manera distinta por los usuarios. Por esta razón es importante identificarlos previamente y utilizar una simbología propia para su tipo, incluyendo y teniendo en cuenta usuarios de distintas culturas.

El modelo de la Meta-Comunicación se describe con los siguientes elementos:

- Emisor: diseñador
- Receptor: usuario
- Mensaje: visión de diseño del software
- Canal de comunicación: sistema de software
- Código: interfaz del sistema
- Contexto: el diseñador y el usuario son interlocutores

Habiendo enunciado los conceptos anteriores, se puede comprender que es lo que se busca en un análisis de Comunicabilidad, específicamente y así analizar las distintas señales entregadas por el software las cuales pueden ser estáticas o dinámicas. A través de estas señales el diseñador comunica su visión de quién es el usuario, cual es la manera en que el usuario realiza sus tareas y por qué las realiza así. Además de la visión que el diseñador tiene sobre el sistema, de cómo tiene que utilizarlo el usuario.

2.3.3 Inspecciones Semióticas

Al hablar de Comunicabilidad surge la necesidad de métodos que permitan su evaluación. Las inspecciones semióticas cumplen con este objetivo, evaluando y clasificando los diferentes tipos de señales o símbolos. Para cada tipo de señal, las inspecciones entregan un mensaje meta-comunicacional sobre el producto de software. Estos mensajes son elaborado por el diseñador de acuerdo a una plantilla meta-comunicacional definida, [6].

Tipos de Señales o Símbolos

- **Señales meta-lingüísticas:** Son aquellas señales que comunican explícitamente a los usuarios el significado codificado del sistema y cómo éste puede ser utilizado (ayuda, documentación, advertencias, explicaciones, etc.).

- **Señales estáticas:** Corresponde a señales con un significado que es interpretado independiente de las relaciones causales y temporales (estructura de la pantalla, menús y opciones, imágenes, texto, cuadros de diálogo, códigos de colores, formas, etc.).
- **Señales dinámicas:** Son señales o símbolos que están relacionados con aspectos temporales y causales en la interfaz, aparecen como elementos emergentes dentro de la interacción (cambios en el estado del sistema, indicadores, retroalimentación, etc.).

Plantilla de Mensajes Meta-Comunicacionales

Las inspecciones semióticas, como método evaluativo de la Comunicabilidad, contemplan el uso de plantillas meta-comunicacionales. A continuación se analizan las siguientes visiones sobre el entendimiento del mensaje entre diseñador y usuario, esto de acuerdo al contexto del elemento de análisis:

- **Aquí está mi entendimiento de quién eres:** Trata del entendimiento del diseñador acerca de quién es el usuario.
- **Qué he aprendido, qué necesitas hacer, la forma que prefieres hacerlo y por qué:** Es el entendimiento que tiene el diseñador sobre las tareas del usuario.
- **Este es el sistema que yo he diseñado para ti, y ésta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** Es el entendimiento general sobre el sistema, de manera que el usuario pueda realizar sus tareas con el software.

Metodología

Esta metodología es utilizada para reconocer los mensajes meta-comunicacionales entregados por el diseñador del sistema de software. Consta de una serie de pasos donde se van analizando los distintos tipos de señales o símbolos metalingüísticos existentes en el producto.

1. **Analizar los símbolos meta-lingüísticos y reconstruir en base a estos el mensaje de meta-comunicación según la plantilla:** En esta fase, se revisa completamente el escenario de inspección buscando sólo signos metalingüísticos, con la finalidad de dar respuesta a los cuestionamientos que plantea la plantilla meta-comunicacional. Deben ser los primeros en ser buscados ya que el diseñador los comunica explícitamente.

2. **Analizar los símbolos estáticos y revisar o reconstruir en base a estos el mensaje de meta-comunicación según la plantilla:** En este paso, el investigador debe volver a visitar el escenario de inspección, ahora considerando solamente los signos estáticos de forma aislada, y nuevamente debe responder los cuestionamientos de la plantilla meta-comunicacional. Estos deben ser examinados estáticamente, es decir aislados de relaciones causales y temporales, para así registrar lo que el diseñador está comunicando al usuario por medio de esta clase de signos.
3. **Analizar los símbolos dinámicos y revisar o reconstruir en base a estos el mensaje de meta-comunicación según la plantilla:** En esta etapa, el investigador debe realizar una nueva iteración, examinando nuevamente el escenario anteriormente visitado, observando los elementos que permiten la comunicación dinámica, y volver a llenar la plantilla meta-comunicacional, como se realizó anteriormente con las otras señales.
4. **Comparación del mensaje meta-comunicacional de los pasos 1, 2 y 3:** En este paso corresponde cotejar y comparar los resultados obtenidos en las etapas anteriores. El objetivo es detectar inconsistencias, patrones y relaciones coherentes que pudiesen existir entre los elementos detectados. Por lo tanto, si al comparar los tres tipos de comunicación se encuentran diferencias sustanciales, el escenario en juicio tiene posiblemente problemas comunicacionales o podría inducir a posibles quiebres comunicacionales.
5. **Evaluación global final de la Comunicabilidad del sistema inspeccionado:** En este paso es donde el investigador finalmente evalúa la Comunicabilidad del sistema, reconstruyendo el mensaje meta-comunicacional unificado, juzgando los costos y beneficios de las estrategias comunicativas identificadas en los pasos previos.

En la sección 3.2 puede observarse la aplicación de la metodología recién descrita, en GreenView y GreenLand, ambos productos de software del tipo *Grid Computing* [1].

2.3.4 Pruebas de Comunicabilidad

En estas pruebas o experimentos realizados en laboratorio, una cantidad determinada de usuarios prueba un software siguiendo una serie de pasos o tareas [6]. Los evaluadores analizan las acciones de los usuarios para detectar los quiebres comunicacionales que puedan ocurrir. Estas situaciones, en que el usuario no logra entender el mensaje entregado por el diseñador del software, pueden dificultar o impedir la realización de las tareas. Los quiebres comunicacionales son de 3 tipos y se definen a continuación [6]:

- **Quiebres Transitorios:** El usuario no entiende bien el diseño, se detiene buscando dentro de las opciones la forma de realizar las tareas.

Q1. ¿Dónde está?: El usuario sabe lo que quiere hacer pero no encuentra el elemento de interfaz apropiado.

Q2. ¿Y ahora qué?: El usuario no sabe que hay que hacer a continuación.

Q3. ¿Qué es esto?: El usuario no entiende un elemento específico de la interfaz.

Q4. ¡Oops!: El usuario comete un error e inmediatamente se da cuenta de su equivocación.

Q5. ¿Dónde estoy?: El usuario está tomando acciones equivocadas, que serían apropiadas en otro contexto.

Q6. ¡No puedo hacerlo de esta manera!: El usuario abandona un camino de interacción porque repentinamente piensa que dicho camino no es el correcto.

Q7. ¿Por qué no?: El usuario repite los pasos de una previa y no exitosa comunicación para descubrir qué estuvo mal.

Q8. ¿Qué pasó?: El usuario no entiende la respuesta del sistema a su acción.

Q9. ¡Auxilio!: El usuario explícitamente solicita asistencia.

- **Quiebres Parciales:** El usuario no entiende la manera de realizar las tareas y busca caminos alternativos, o realiza las tareas de otro modo.

Q10. ¡Puedo hacerlo de otra manera!: El usuario no está enterado totalmente de los caminos de acción que ofrece el sistema para llevar a cabo una determinada tarea (el usuario no entiende el diseño).

Q11. Gracias, pero no: El usuario entiende el camino que el sistema ofrece, pero elige un camino alternativo (el usuario entiende el diseño).

- **Quiebres Totales:** El usuario no entiende en absoluto el software, de manera que no puede utilizarlo e incluso puede llegar a no ser consciente de su uso errado.

Q12. ¡Me parece que está bien!: El usuario está convencido de que ha alcanzado su meta, pero eso no ha ocurrido (El usuario no es consciente del quiebre).

Q13. ¡Me rindo!: El usuario explícitamente expresa su inhabilidad para alcanzar la meta (El usuario es consciente del quiebre).

Los quiebres comunicacionales descritos son analizados por los evaluadores, quienes a su vez elaboran un perfil semiótico. Este perfil permite inferir el tipo de usuario, el modo de uso del software y la visión de diseño de éste.

3. Solución Propuesta

Según los objetivos propuestos en las secciones 1.3.1 y 1.3.2 se describe una solución tentativa para la problemática expuesta. Se realizan ciertos experimentos sobre productos de software analizados en un trabajo investigativo anterior [1] de manera de poder dilucidar una primera aproximación a la solución del problema, lo cual se expone finalmente en la sección 4.

3.1 Mejora Tentativa a Heurísticas Específicas para Productos del Tipo Grid Computing

En un trabajo investigativo previo [1], se desarrollaron una serie de heurísticas específicas con el fin de analizar productos del tipo Grid Computing. Para validar éstas, se llevó a cabo un conjunto de experimentos sobre los casos de estudio GreenView y GreenLand. De acuerdo a los resultados obtenidos de esta investigación, las heurísticas específicas [1] reflejaron comparativamente un mejor desempeño en la detección de problemas [9]. Sin embargo, algunos de éstos no fueron detectados haciendo uso de heurísticas específicas para aplicaciones del tipo Grid Computing,. Considerando estos resultados, y de forma tentativa, podrían incluirse mejoras a las heurísticas propuestas con el objetivo de abarcar los problemas detectados solamente con las heurísticas de Nielsen [9].

3.1.1 Problemas Detectados Solamente con Heurísticas Genéricas

En las tablas 3.1 y 3.2 se muestran los problemas detectados en los productos GreenView y GreenLand, con las heurísticas genéricas de Nielsen. Los problemas fueron ordenados según un parámetro de Criticidad (en un escala de 0 a 8), obtenido a través de la suma de su severidad y frecuencia (cada atributo con valor de 0 a 4). Además, se puede observar en la Tabla 3.1, que de los 6 problemas detectados exclusivamente con las heurísticas de Nielsen en el producto GreenView, los relacionados con la Heurística N8 (Diseño estético y minimalista) tienen una mayor ocurrencia.

Tabla 3.1: Listado de problemas encontrados exclusivamente con las heurísticas genéricas de Nielsen. Caso de Estudio GreenView.

ID	Problema	Prom. Criticidad	Heurísticas asociadas
1	Exceso de elementos en ventanas	5.25	N8
9	No se previene el ingreso de valores no permitidos	5	N5
10	Parece que todos los botones fueran igual de	5	N8

	importantes		
12	Al buscar ubicaciones inexistentes sistema se queda indefinidamente buscando	4	N1
11	Sistema resulta poco intuitivo	3.75	N8
3	Ayuda oculta para novatos	3.25	N10

Si bien la mayoría de los problemas detectados son menores para el producto GreenLand, en la Tabla 3.2 nuevamente la Heurística N8 (Diseño estético y minimalista) aparece con una mayor ocurrencia, seguida por la Heurística N4 (Consistencia y estándares). Cabe señalar, la existencia de un problema con alta criticidad asociado a la Heurística N7 (Flexibilidad y eficiencia de uso).

Tabla 3.2: Listado de problemas encontrados exclusivamente con las heurísticas genéricas de Nielsen. Caso de Estudio GreenLand.

ID	Problema	Prom. Criticidad	Heurísticas asociadas
13	Para detener un proceso hay que buscarlo	6	N7
15	Nombre de usuario desaparece en ocasiones	3.67	N8
3	Borrar parámetros	2.67	N2
16	Mal contraste entre fondo y texto	2.67	N8
17	Flechas distintas para uso similar	2	N4
20	Mayúsculas y minúsculas usadas indistintamente	1.33	N4

3.1.2 Análisis Inicial de Problemas No Detectados con las Heurísticas Específicas para Aplicaciones del Tipo Grid Computing

A raíz de lo anteriormente expuesto (3.1.1), según el total de los problemas detectados exclusivamente con las Heurísticas de Nielsen, se observan:

- 5 Problemas relacionados con la Heurística N8 (Diseño estético y minimalista)
- 2 Problemas relacionados con la Heurística N4 (Consistencia y estándares)
- 1 Problema relacionado con la Heurística N1 (Visibilidad del estado del sistema)
- 1 Problema relacionado con la Heurística N2 (Correspondencia entre el sistema y el mundo real)

- 1 Problema relacionado con la Heurística N5 (Prevención de errores)
- 1 Problema relacionado con la Heurística N7 (Flexibilidad y eficiencia de uso)

Tras obtener estos resultados, se estableció la cantidad de ocurrencias y los promedios de Criticidad de éstas. Los problemas que fueron detectados exclusivamente por las heurísticas genéricas de Nielsen, se muestra en la Tabla 3.3 a continuación:

Tabla 3.3: Ranking de problemas detectados exclusivamente con heurísticas genéricas de Nielsen para los productos GreenView y GreenLand.

ID	Heurística	Cantidad de ocurrencias	Prom. Criticidad
N8	Diseño estético y minimalista	5	4.668
N7	Flexibilidad y eficiencia de uso	1	6
N5	Prevención de errores	1	5
N1	Visibilidad del estado del sistema	1	4
N4	Consistencia y estándares	2	1.665
N2	Correspondencia entre el sistema y el mundo real	1	2.67

Tras esta primera aproximación, se puede concluir tentativamente que las heurísticas específicas para aplicaciones del tipo Grid Computing, podrían arrojar resultados inexactos, principalmente en lo vinculados a las heurísticas N8 (Diseño estético y minimalista), en relación a la cantidad de ocurrencias para problemas asociados y N7 (Flexibilidad y eficiencia de uso) en la exclusión de un problema mayor relacionado con esta heurística.

3.2 Análisis Inicial de Impacto de la Comunicabilidad Sobre la Usabilidad

Uno de los objetivos de este trabajo es investigar el impacto o incidencia de la Comunicabilidad sobre la Usabilidad. Para cumplir este objetivo, se realizaron inspecciones sobre productos de software del tipo *Grid Computing* y posteriormente los resultados fueron comparados con las evaluaciones de Usabilidad.

Como parte de este proceso investigativo, además se desarrollaron experimentos pilotos de Comunicabilidad, específicamente Inspecciones Semióticas.

3.2.1 Experimento Piloto: Inspección Semiótica del Producto GreenView

El primer experimento de Comunicabilidad se realizó sobre el producto GreenView, para así tener una primera aproximación o idea sobre el estado de la Comunicabilidad en productos de software del tipo *Grid Computing*. Como parte de esta etapa, se realizó una Inspección Semiótica haciendo uso de la metodología explicada anteriormente (2.3.4).

A continuación se detalla la metodología de trabajo utilizada en el desarrollo del experimento piloto:

Paso 1: Analizar los símbolos meta-lingüísticos y reconstruir en base a estos el mensaje de meta-comunicación según la plantilla.

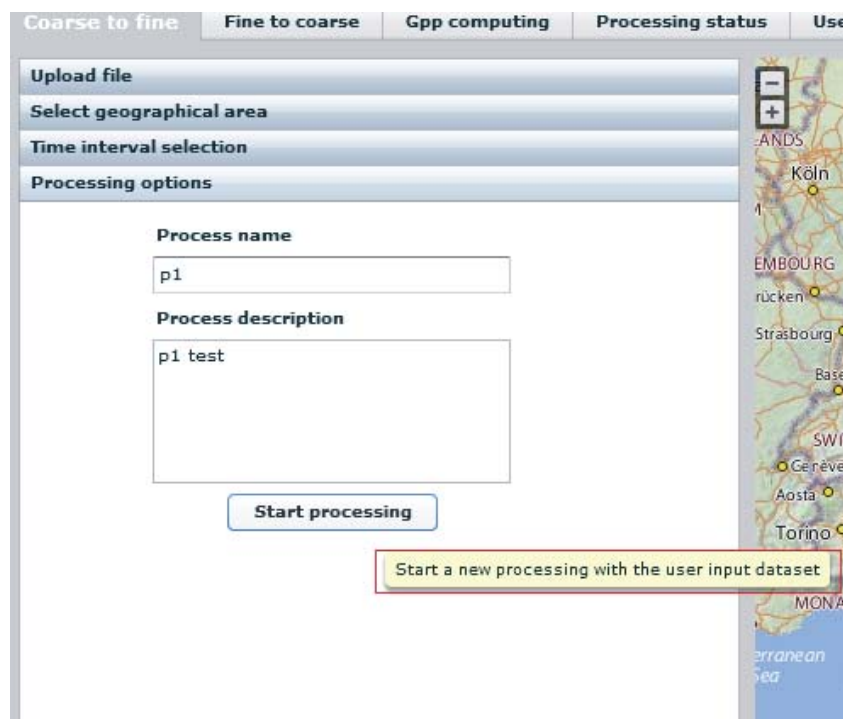


Figura 3.1: Información explícita sobre ciertas funciones al colocar el cursor del mouse sobre ellas.

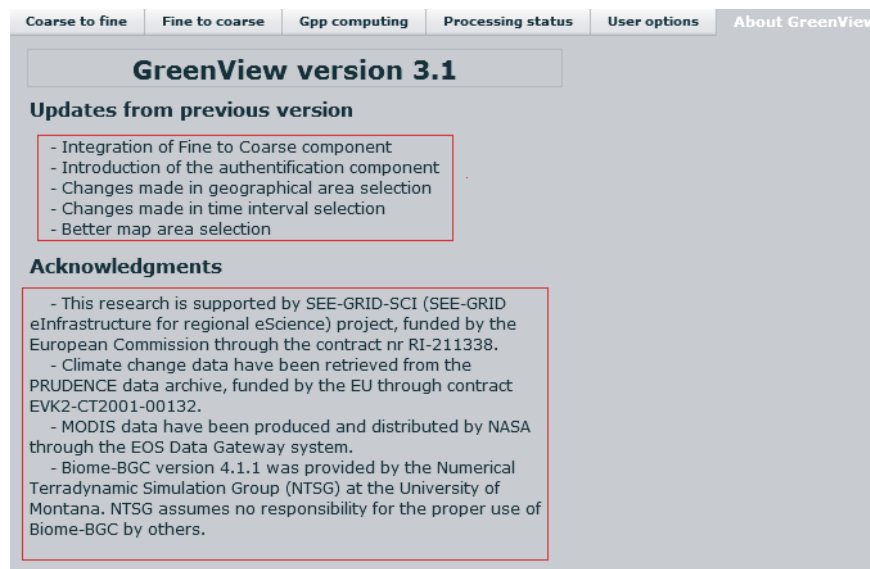


Figura 3.2: Información explícita con información acerca del software en la pestaña About GreenView.

GreenView casi no posee símbolos meta-lingüísticos, lo que hace suponer que la aplicación está dirigida a usuarios expertos. Cabe señalar que la versión de esta aplicación no cuenta con un tutorial o ayuda.

Mensajes meta-comunicacionales

- **Aquí está mi entendimiento de quien eres:** Eres un usuario experto, puesto que no requieres ningún tipo de ayuda para utilizar el producto GreenView.
- **Qué he aprendido que necesitas hacer, la forma que prefieres hacerlo y por qué:** No sé mucho acerca de lo que necesitas hacer, puesto que no lo indicas explícitamente, tengo muy poca información. Sé que procesas grupos de datos climáticos sin conocer el propósito real de esto. Todavía no sé el porqué o bien es algo encriptado o de difícil comprensión. Además sé que en esta última versión realizas más tareas pues se agregaron funciones.
- **Éste es el sistema que yo he diseñado para ti, y ésta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** El sistema ha sido diseñado para un usuario experto, que no necesita ningún tipo de ayuda, por lo mismo no te indico las tareas que debes realizar con el software, por lo que tu visión puede ser confusa.

Paso 2: Analizar los símbolos estáticos y revisar o reconstruir en base a estos el mensaje de meta-comunicación

según la plantilla.

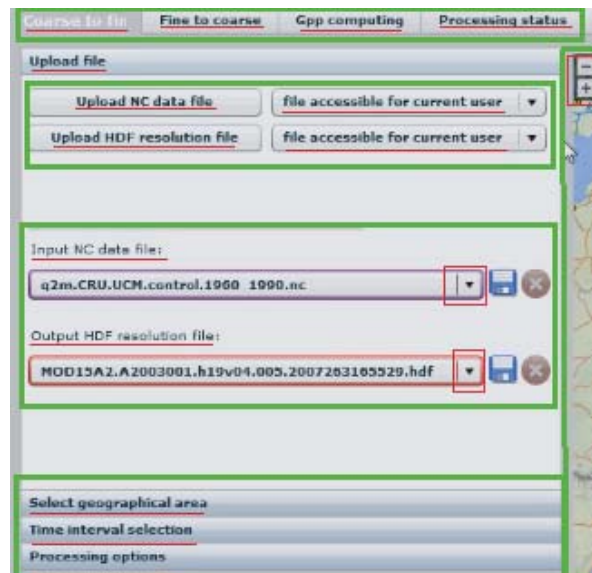


Figura. 3.3: Opciones de pestaña *Coarse to fine*. Pueden observarse distintos símbolos meta-lingüísticos como estructura y agrupación de controles (en recuadro verde), íconos de controles (en recuadros rojos) y mensajes como títulos de funciones (subrayados en rojo).

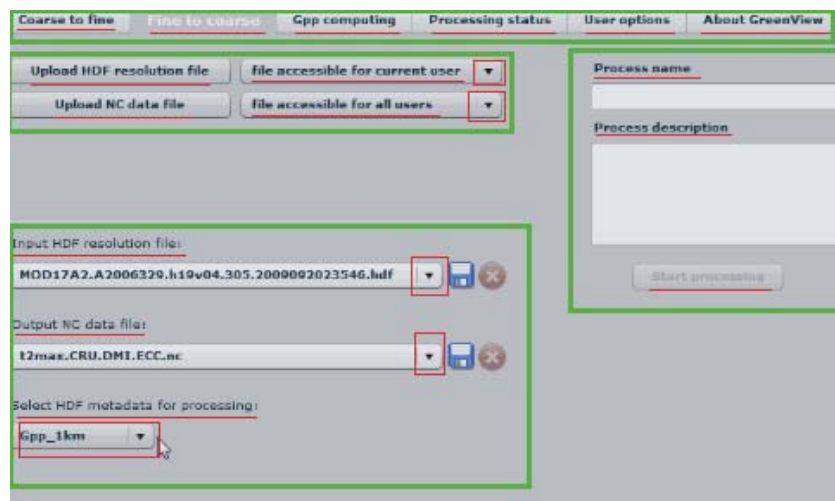


Figura 3.4: Información de la pestaña *Fine to coarse*. Nuevamente se observan títulos implícitos de distintas funciones del software, además de una estructura y diversos controles que corresponden a símbolos meta-lingüísticos.

Time info	Node name	Start server time	End server time	Status	Options
1989, August	70_CoarseToFine	2010-04-29 13:50:10	2010-04-29 14:01:13	DONE	[Icons]
1989, July	76_CoarseToFine	2010-04-29 13:50:10	2010-04-29 14:01:13	DONE	[Icons]
1989, June	72_CoarseToFine	2010-04-29 13:50:10		RUNNING	
1989, May	68_CoarseToFine	2010-04-29 13:50:10		RUNNING	
1989, April	64_CoarseToFine	2010-04-29 13:50:10		RUNNING	
1989, March	60_CoarseToFine	2010-04-29 13:50:10	2010-04-29 14:05:54	DONE	[Icons]
1989, February	56_CoarseToFine	2010-04-29 13:50:11	2010-04-29 14:06:03	DONE	[Icons]
1989, January	52_CoarseToFine	2010-04-29 13:50:11		RUNNING	
1988, December	48_CoarseToFine	2010-04-29 13:50:11		RUNNING	
1988, November	44_CoarseToFine	2010-04-29 13:50:11		RUNNING	
1988, October	40_CoarseToFine	2010-04-29 13:50:11		SCHEDULED	
1988, September	36_CoarseToFine	2010-04-29 13:50:11		SUBMITTED	
1988, August	32_CoarseToFine	2010-04-29 13:50:12		SCHEDULED	
1988, July	28_CoarseToFine	2010-04-29 13:50:12		SUBMITTED	
1988, June	24_CoarseToFine	2010-04-29 13:50:12		SUBMITTED	

Figura 3.5: Información de la pestaña *Processing status*. Puede observarse la información acerca de los distintos procesos que están corriendo en el sistema, información que no es explícita y corresponde a símbolos meta-lingüísticos.

Processing and monitoring

No. of model simulation: 100000

Process name: Calibration

Process description: EGC model calibration

[Start processing]

Start a new processing with the user input dataset

Parameters for normal ini file:

- soil_depth_n: /* (m) effective soil depth (corrected for rock fraction) */
- sand_n: /* (%) sand percentage by volume in rock-free soil */
- silt_n: /* (%) silt percentage by volume in rock-free soil */
- clay_n: /* (%) clay percentage by volume in rock-free soil */
- ndep_n: /* (kgN/m2/yr) wet+dry atmospheric deposition of N */
- nfix_n: /* (kgN/m2/yr) symbiotic+asymbiotic fixation of N */

Parameters for spinup ini file:

- soil_depth_s: /* (m) effective soil depth (corrected for rock fraction) */
- sand_s: /* (%) sand percentage by volume in rock-free soil */
- silt_s: /* (%) silt percentage by volume in rock-free soil */
- clay_s: /* (%) clay percentage by volume in rock-free soil */
- ndep_s: /* (kgN/m2/yr) wet+dry atmospheric deposition of N */
- nfix_s: /* (kgN/m2/yr) symbiotic+asymbiotic fixation of N */

Conditions:

- sand_n + silt_n + clay_n = 100
- sand_s + silt_s + clay_s = 100
- soil_depth_s = soil_depth_n
- sand_s = sand_n
- silt_s = silt_n
- clay_s = clay_n
- nfix_s = nfix_n

Figura 3.6: Información de la pestaña *Gpp computing*. El texto incluido en los botones y su estructura, corresponde a símbolos meta-lingüísticos, mientras que la ejecución de la acción o cambio de estado del sistema, corresponde a un símbolo dinámico.

GreenView tiene una gran cantidad de símbolos estáticos, principalmente referidos a títulos de tareas o funciones. Se observa además, una estructura ordenada donde se agrupan las funciones por pestañas y un sub-agrupamiento entre éstas.

Mensajes meta-comunicacionales

- **Aquí está mi entendimiento de quien eres:** Eres un usuario que tiene un amplio entendimiento acerca del propósito del software, además se puede suponer que eres un usuario frecuente del software.
- **Qué he aprendido que necesitas hacer, la forma que prefieres hacerlo y por qué:** Necesitas tener agrupadas todas las funciones de un mismo tipo para accederlas rápidamente y así solicitar al sistema que realice las tareas de manera ordenada. Conozco las tareas que realizas, por ejemplo procesar archivos de imágenes con distintas resoluciones pertenecientes a zonas geográficas y poseer información del estado de los procesos realizados por el sistema. Pero aún no sé por qué necesitas realizar estas tareas. Lo que entiendo es que necesitas tener un agrupamiento correcto de las funciones para acceder de manera expedita a ellas.
- **Éste es el sistema que yo he diseñado para ti, y ésta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** El sistema ha sido diseñado para un usuario experto, que sabe con exactitud lo que tiene que hacer y dónde encontrar las distintas funciones en el software. En este caso el software debería cumplir con tu visión, pues tú sabes cómo utilizarlo y éste te da todos los títulos de las funciones de manera ordenada.

Paso 3: Analizar los símbolos dinámicos y revisar o reconstruir en base a estos el mensaje de meta-comunicación según la plantilla.

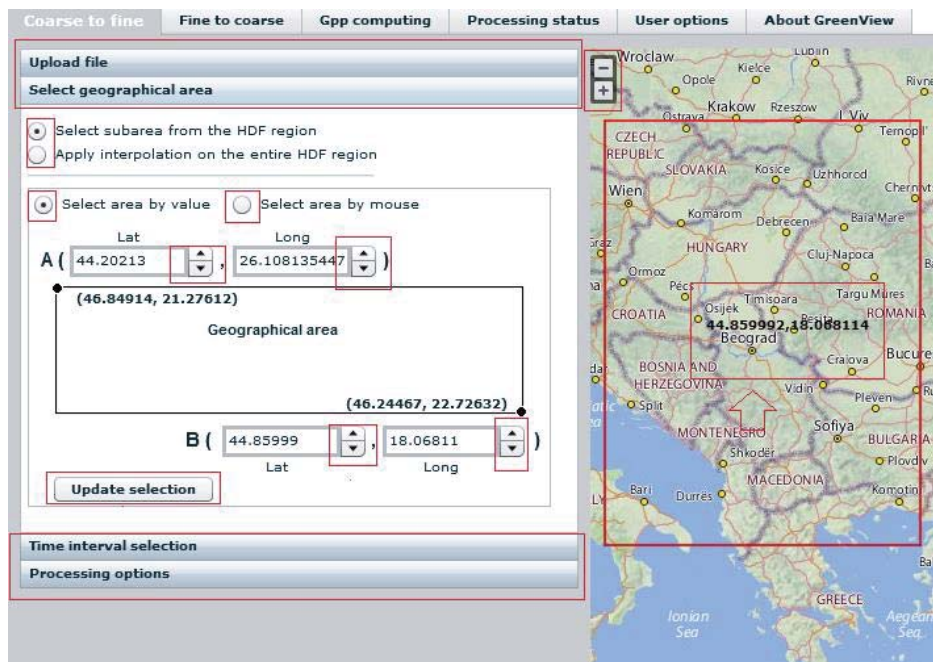


Figura 3.7: Botones y controles que manipulan la selección de área en el mapa. También puede manipularse directamente sobre el mapa moviendo el mouse y haciendo click en él, tal como indica la flecha.

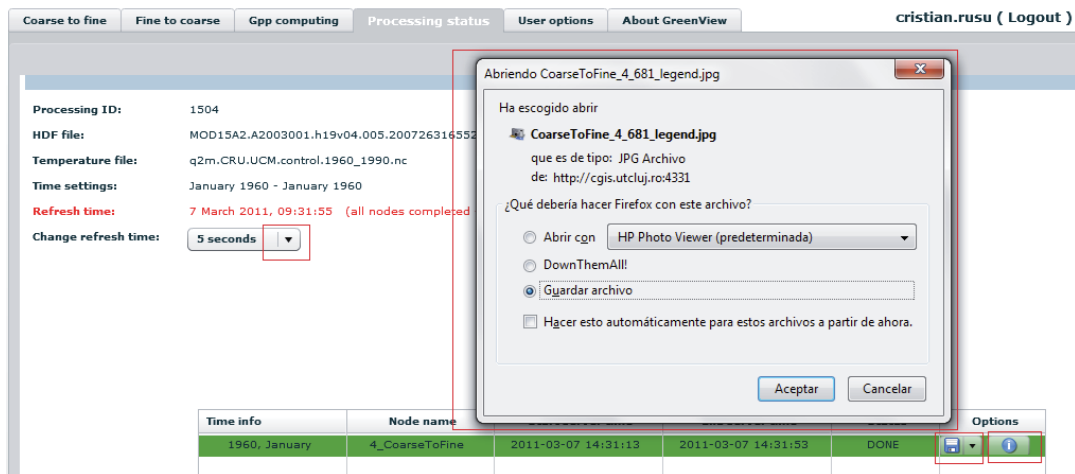


Figura 3.8: Acción del botón *guardar*. Al presionar este botón, se abre una ventana para llevar a cabo esta acción. Pueden apreciarse, además, otros controles que producen un efecto sobre el estado del sistema.

GreenView cuenta con una serie de símbolos dinámicos, relacionados con controles para manipular el mapa, botones y menús desplegables.

Mensajes meta-comunicacionales

- **Aquí está mi entendimiento de quien eres:** Eres un usuario que tiene un amplio entendimiento acerca del propósito del software. Puedes ser, además, un usuario frecuente del software.
- **Qué he aprendido que necesitas hacer, la forma que prefieres hacerlo y por qué:** Necesitas tener agrupadas todas las funciones de un mismo tipo para accederlas rápidamente, además necesitas controles como botones, menús desplegables y otros controles más naturales para manipular el mapa. Esto te facilita la interacción con el software., pero con esto no me queda claro el por qué.
- **Éste es el sistema que yo he diseñado para ti, y esta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** El sistema ha sido diseñado para un usuario experto, quien necesita relacionarse directamente con una serie de controles interactivos que le permitan hacer uso del software realizando las tareas de manera expedita, siendo esta la visión entendida para la simbología dinámica.

Paso 4: Comparación del mensaje meta-comunicacional de los pasos 1, 2 y 3.

En general, las señales meta-comunicacionales más presentes en GreenView son las estáticas y las dinámicas. Las señales meta-lingüísticas son extremadamente escasas, limitándose a algunas tareas someramente explicadas.

Debido a la falta de señales meta-lingüísticas, se puede inferir que esta aplicación está dirigida a usuarios expertos, con necesidades muy específicas y que además, están habituados al uso del software, pues no se muestran ayudas ni tutoriales. Como consecuencia, no existe información que explique las funciones del software ni su propósito. Esto coincide con lo que se observa en las señales estáticas, son títulos de funciones estructuradas y separadas principalmente por pestañas y las señales dinámicas, que son los controles que permiten manipular el software, para ambos tipos de señales no hay mayor explicación de sus funciones.

De los resultados obtenidos se desprende la existencia de una sobrecarga de señales estáticas y dinámicas, en comparación con las señales meta-lingüísticas encontradas. Esto permite suponer que el producto podría no estar terminado o bien, correspondería a un prototipo.

Paso 5: Evaluación final y global de la Comunicabilidad en el software

Mensajes meta-comunicacionales

- **Aquí está mi entendimiento de quien eres:** Eres un usuario experto y frecuente del software. Sabes utilizar el software y manipular sus controles. No necesitas ayuda, puede que te confundas si eres un usuario primerizo del software pues no te muestro nada acerca del propósito ni de las tareas que se pueden realizar.
- **Qué he aprendido que necesitas hacer, la forma que prefieres hacerlo y por qué:** He aprendido que tienes necesidades muy específicas y tareas como procesar imágenes de distintas resoluciones para analizarlas y sacar distinta información de éstas, entiendo que procesas datos climáticos. Sé que necesitas tener las distintas funciones estructuradas y agrupadas, además de poder interactuar con ellas a través de diversos controles, porque para ti esta es la manera rápida y expedita para utilizar el software. Si eres primerizo utilizando el software navegas por él y buscas las distintas funciones que realiza pues no te digo en un principio qué hacer y cómo hacerlo. El por qué de las funciones que realizas no lo tengo claro, en casi ningún momento recibo información explícita de lo que haces, lo que entiendo en torno a las funciones es que necesitas una agrupación correcta de las funciones para accederlas de manera expedita.
- **Éste es el sistema que yo he diseñado para ti, y esta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** He diseñado un software que está estructurado según las principales tareas que se pueden realizar con él y debes utilizarlo haciendo uso de los diversos controles existentes. Sabes que cosas tienes que hacer y eres un experto en el uso del software, en el caso contrario puedes confundirte y tener problemas al utilizar el software, e incluso sin llegar a entenderlo.

3.2.2 Experimento Piloto: Inspección Semiótica del producto GreenLand

La investigación contempló un segundo experimento, en este caso sobre el producto GreenLand. Los resultados obtenidos de este pilotaje, permitieron elaborar comparativas con el producto GreenView y conclusiones en torno a los mensajes meta-comunicacionales.

A continuación se detalla la metodología de trabajo utilizada en el desarrollo del experimento piloto:

Paso 1: Analizar los símbolos meta-lingüísticos, y reconstruir en base a estos, el mensaje de meta-comunicación según la plantilla.

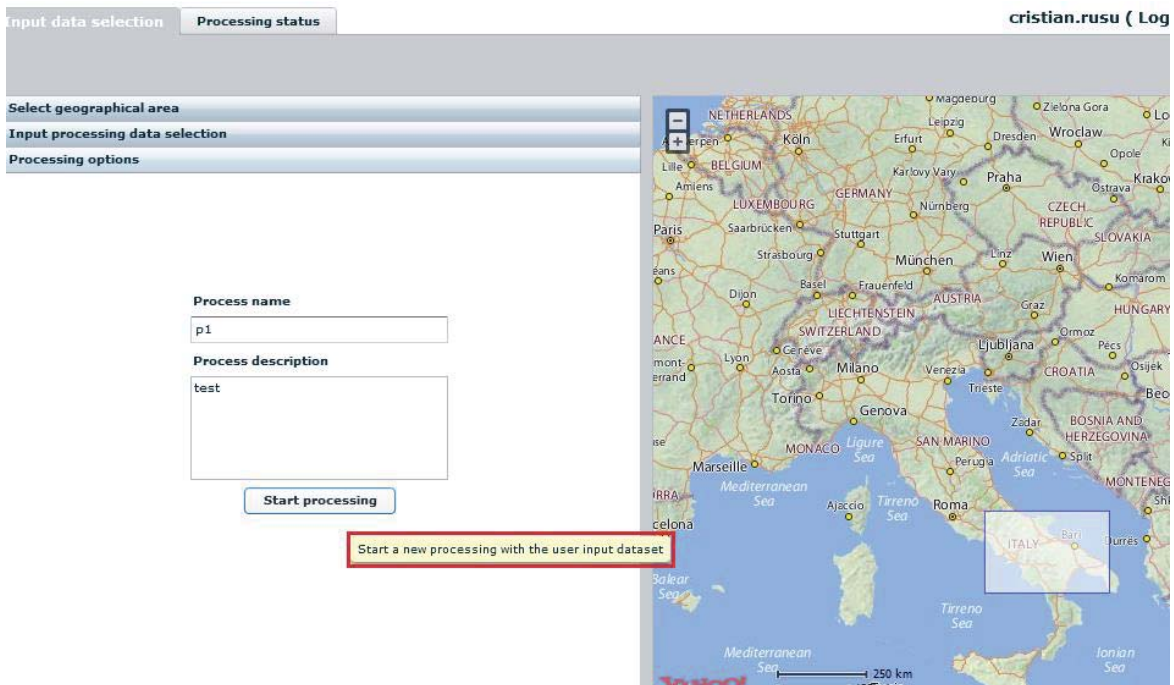


Figura 3.9: Símbolo meta-lingüístico de mensaje sobre botón.

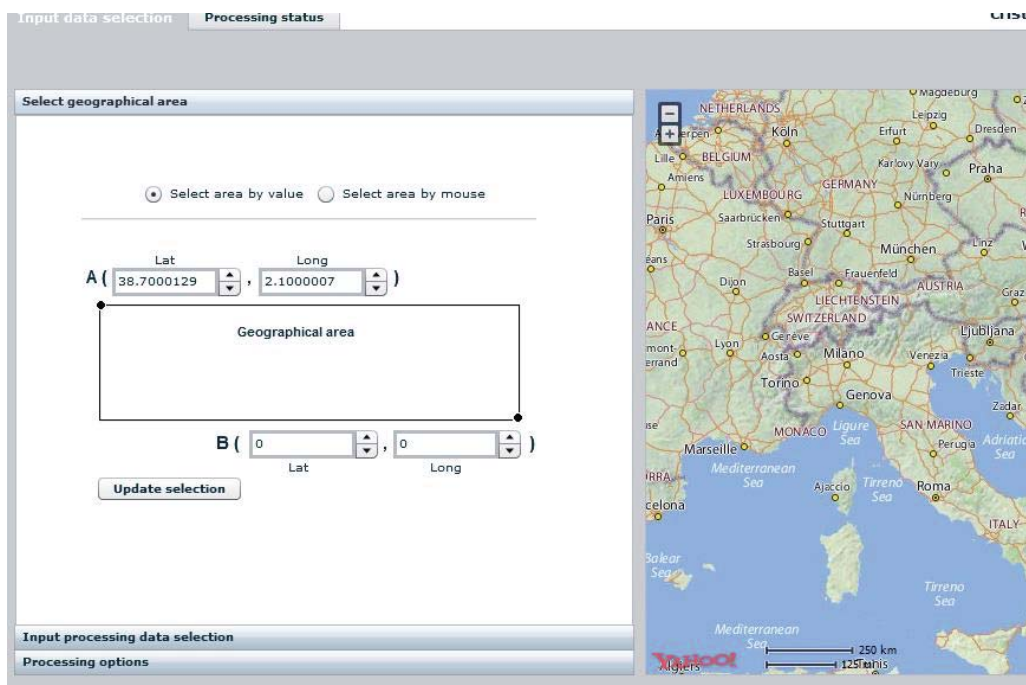


Figura 3.10: No se muestran símbolo meta-lingüísticos en las opciones que permiten seleccionar el área geográfica.

Input data selection Processing stati cristian.rusu (Logout)

Refresh time: 18 July 2011, 03:31:02 (refresh every 5 seconds)

Change refresh time: 5 seconds ▼

Current process status information

description	Node name	Start server time	End server time	Status ▲	Options
	1322_24_SubFloat	2011-07-18 10:29:00	2011-07-18 10:30:14	COMPLETED	
	1442_12_MultFloat	2011-07-18 10:28:58	2011-07-18 10:30:19	COMPLETED	
	1442_24_SubFloat	2011-07-18 10:28:56	2011-07-18 10:30:21	DONE	
	1322_9_MultFloat	2011-07-18 10:29:02	2011-07-18 10:30:23	DONE	
	1442_9_MultFloat	2011-07-18 10:28:58	2011-07-18 10:30:23	DONE	
Red * Red	1442_4_Mult	2011-07-18 10:28:58	2011-07-18 10:30:23	DONE	
	1202_12_MultFloat	2011-07-18 10:29:06	2011-07-18 10:30:33	DONE	
Red * Red	1322_4_Mult	2011-07-18 10:29:03	2011-07-18 10:30:13	DONE	
NIR * NIR	1202_3_Mult	2011-07-18 10:29:07		DONE	
	1082_14_Add	2011-07-18 10:29:10		DONE	
	1082_24_SubFloat	2011-07-18 10:29:09		RUNNING	
	1082_26_MultFloat	2011-07-18 10:29:08		RUNNING	
	1082_12_MultFloat	2011-07-18 10:29:10		SCHEDULED	
	1082_9_MultFloat	2011-07-18 10:29:10		SCHEDULED	
	962_26_MultFloat	2011-07-18 10:29:12		SCHEDULED	

Figura 3.11: Pantalla de los proceso, se observa un aviso como símbolo meta-lingüístico.

A partir de este experimento, se observó que la aplicación GreenLand incluye escasos símbolos meta-lingüísticos. Además, la versión revisada del producto no cuenta con “Ayudas” ni una sección “Acerca de”, por lo tanto se puede inferir que la aplicación está dirigida a usuarios expertos.

Mensajes meta-comunicacionales

- **Aquí está mi entendimiento de quien eres:** No requieres ningún tipo de ayuda, por lo que puedo concluir que eres un usuario experto.
- **Qué he aprendido que necesitas hacer, la forma que prefieres hacerlo y por qué:** No entiendo lo que necesitas hacer ni el propósito, tengo información casi nula. Entiendo que procesas datos de un área geográfica, pero sin saber la razón ni el propósito.
- **Éste es el sistema que yo he diseñado para ti, y ésta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** El sistema ha sido diseñado para un usuario que no necesita ayuda al realizar las tareas, un usuario experto. Tampoco te doy indicaciones de qué tareas realizas.

Paso 2: Analizar los símbolos estáticos, y revisar o reconstruir en base a estos, el mensaje de meta-comunicación según la plantilla.

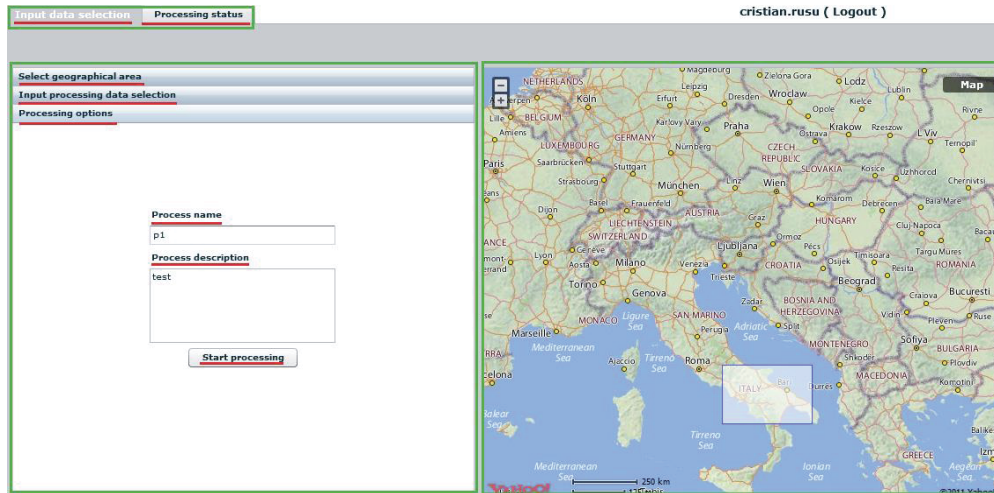


Figura 3.12: Se observan distintos símbolos estáticos entre los cuales hay una estructura (en color verde) con dos secciones principales bien diferenciadas, además del uso de pestañas. También se observan textos correspondientes a las funciones del software (en rojo).

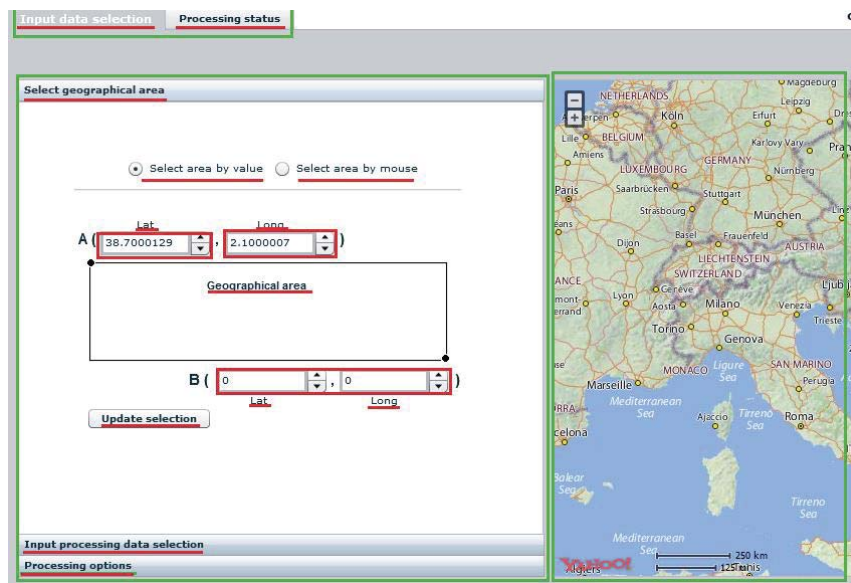


Figura 3.13: Al igual que en la imagen anterior, acá se observa la serie de símbolos estáticos presentes en esta sección.

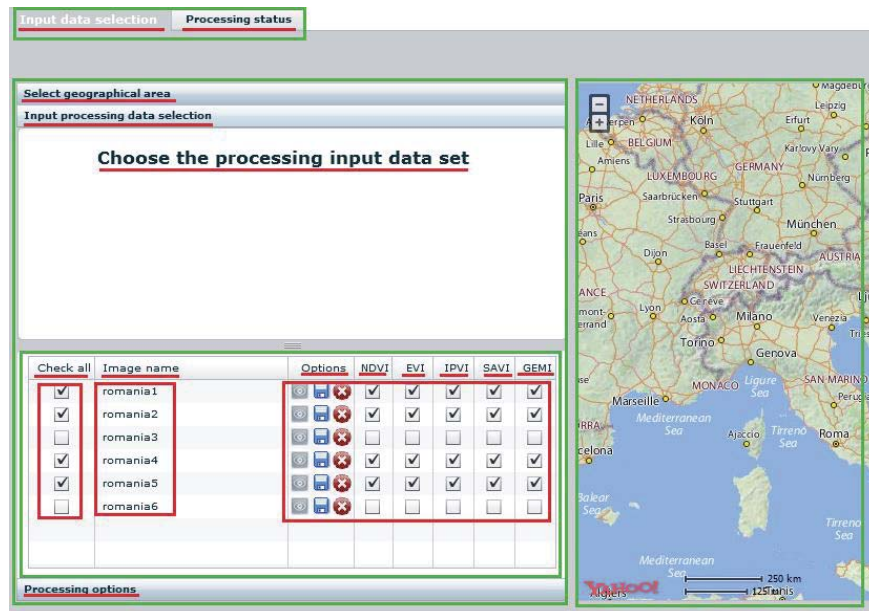


Figura 3.14: En esta imagen, además de los símbolos estáticos descritos anteriormente, se observan íconos y controles de checkbox, los cuales de por sí representan símbolos estáticos como simbología de funciones del producto de software.

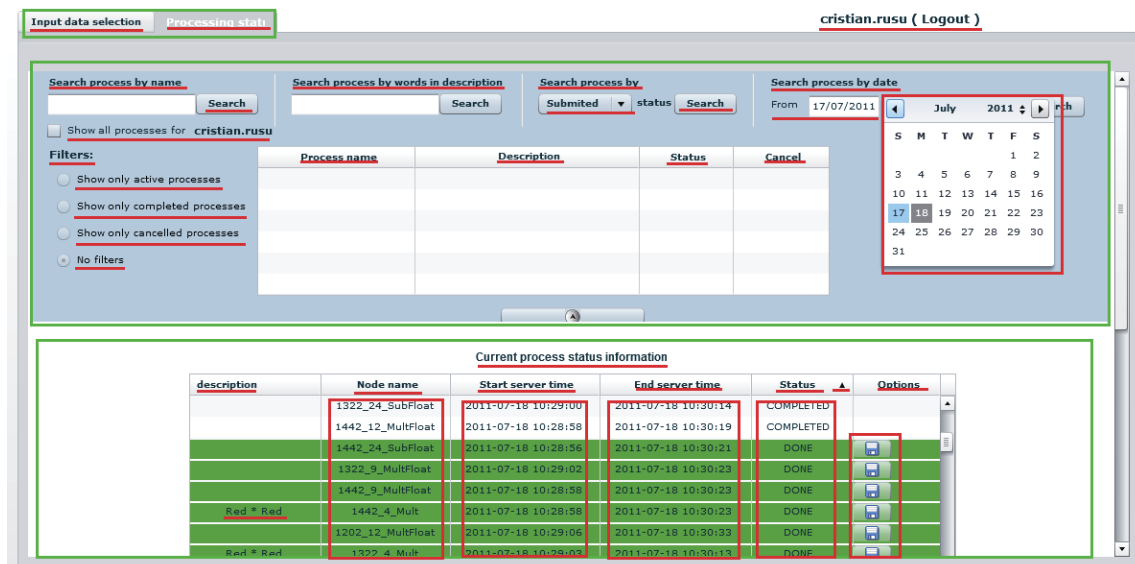


Figura 3.15: En esta imagen correspondiente a la sección de procesos, puede observarse un buen número de símbolos estáticos, mostrándose una combinación de distintos tipos, que incluyen estructuras, nombres de funciones y mensajes entregados por el producto GreenLand.

GreenLand, cuenta con un gran número de símbolos dinámicos en torno a sus funcionalidades. Además se observan controles y mensajes entregados por el software. Es un caso muy similar a lo descrito para el producto GreenView.

Mensajes meta-comunicacionales

- **Aquí está mi entendimiento de quien eres:** Eres un usuario que entiende en su totalidad el funcionamiento del software y estás familiarizado con él.
- **Qué he aprendido que necesitas hacer, la forma que prefieres hacerlo y por qué:** Necesitas tener las funciones ordenadas y agrupadas para así hacer uso del producto de manera expedita, utilizando una serie de controles (botones y menús desplegables). Prefieres tener las funciones ordenadas de manera concisa pues el producto de software cumple una función muy específica y así te es más cómodo utilizarlo para realizar tus tareas.
- **Éste es el sistema que yo he diseñado para ti, y ésta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** El sistema ha sido diseñado para un usuario experto, quien necesita interactuar directamente con una serie de controles. Estos te permiten realizar las tareas de manera expedita, siendo esta la visión entendida en torno a la simbología estática.

Paso 3: Analizar los símbolos dinámicos, y revisar o reconstruir en base a estos, el mensaje de meta-comunicación según la plantilla.

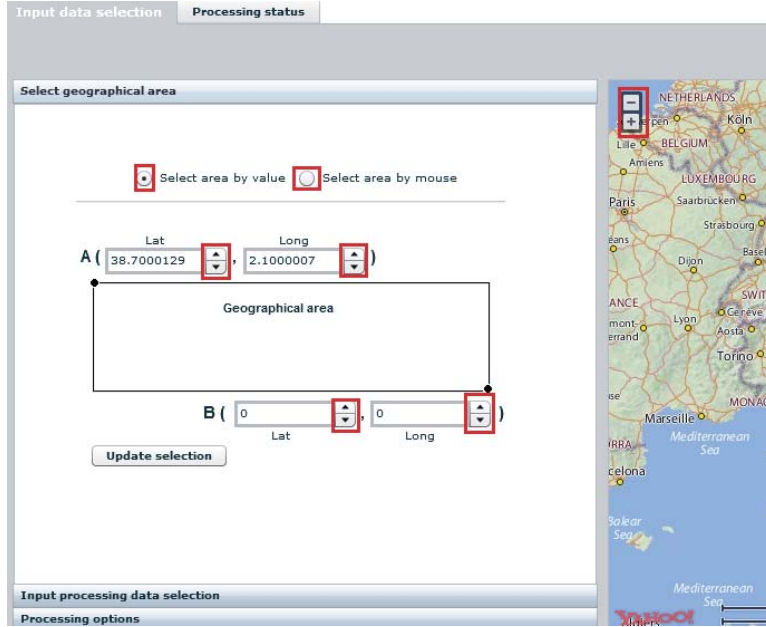


Figura 3.16: Controles de manipulación directa.

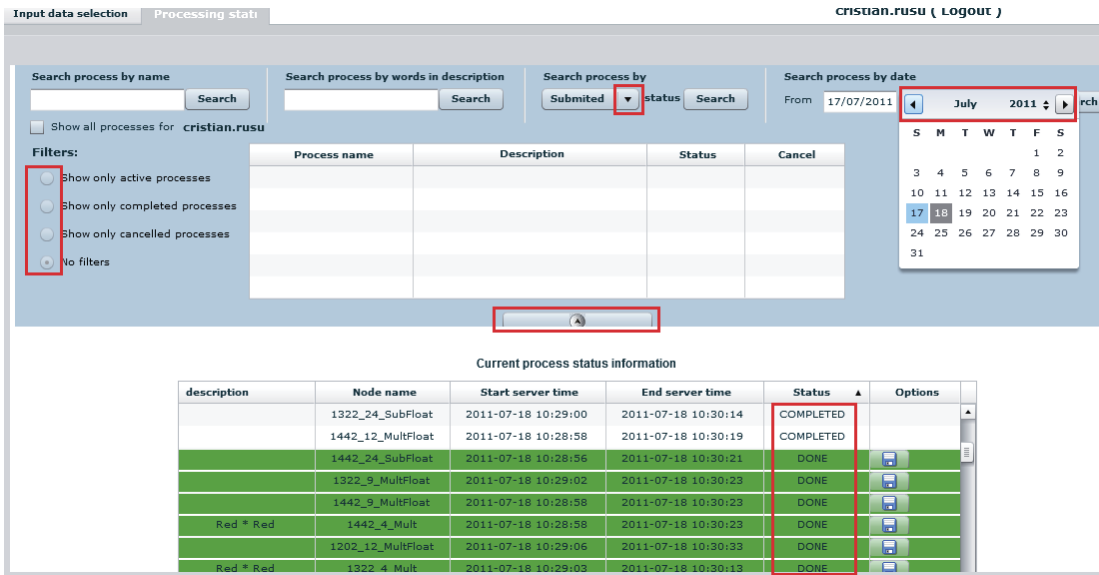


Figura 3.17: Controles de manipulación directa y mensajes de estado del sistema.

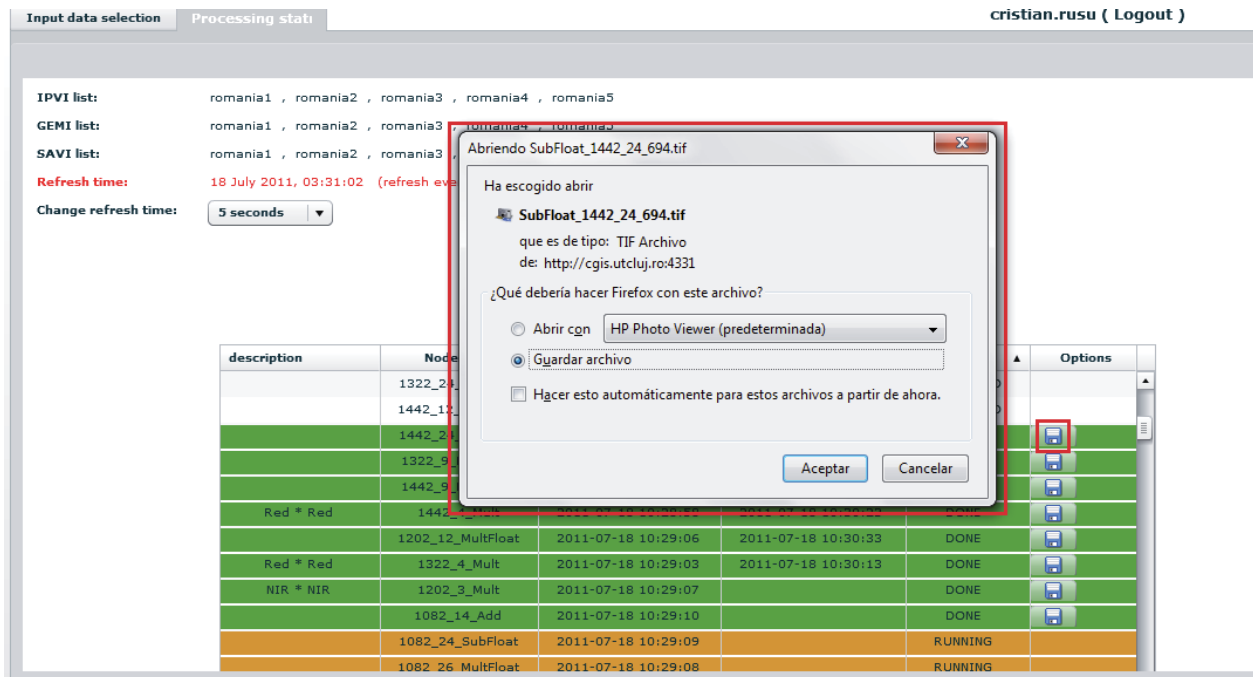


Figura 3.18: Ventana de pop-up que se abre al hacer click sobre el ícono de diskette.

Los símbolos dinámicos con los que cuenta GreenLand, están centrados principalmente en la manipulación del producto de software, tales como los diversos controles en forma de botones y menús desplegables.

Mensajes meta-comunicacionales

- **Aquí está mi entendimiento de quien eres:** Eres un usuario que entiende en su totalidad el funcionamiento del software y estás familiarizado con él.
- **Qué he aprendido que necesitas hacer, la forma que prefieres hacerlo y por qué:** Necesitas tener las funciones ordenadas y agrupadas para así hacer uso del producto de software de manera expedita, utilizando una serie de controles tales como botones y menús desplegables. He aprendido que como resultado del uso del programa, recibes como salida una imagen de alta resolución, pero no me queda claro por qué y para qué.
- **Éste es el sistema que yo he diseñado para ti, y ésta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** El sistema ha sido diseñado para un usuario experto, quien necesita interactuar directamente con una serie de controles interactivos que te

permiten hacer uso del software realizando las tareas de manera expedita. Este sistema tiene una función muy específica, siendo esta la visión entendida para la simbología dinámica.

Paso 4: Comparación del mensaje meta-comunicacional de los pasos 1, 2 y 3.

Las señales meta-comunicacionales estáticas están mayoritariamente presentes en GreenLand, en comparación con aquellas dinámicas.. Por su parte, las señales meta-lingüísticas son escasas, casi completamente ausentes, limitándose a algunas tareas pobremente explicadas al pasar el cursor del mouse sobre los controles.

Debido a la falta de señales meta-lingüísticas, se puede inferir que el usuario al que está dirigida la aplicación, debe ser un usuario experto, con dominio total del sistema. Esto debido a la ausencia de ayudas y/o tutoriales que guíen al usuario en la resolución de problemas o que explique las funciones y propósitos del software. Esto coincide con lo que puede observarse en las señales estáticas que son títulos de funciones estructuradas y separadas por grupos y en las señales dinámicas que son los controles que permiten manipular software.

La sobrecarga de señales estáticas y dinámicas, en comparación con las pocas señales meta-lingüísticas, permite suponer que el producto no esté terminado, es un prototipo del producto, o bien es parte de un sistema mayor.

Paso 5: Evaluación final y global de la Comunicabilidad en el software

Mensajes meta-comunicacionales

- **Aquí está mi entendimiento de quien eres:** Eres un usuario que entiende en su totalidad el funcionamiento del software y estás familiarizado con él. Sabes usar el software y manipular sus controles. No necesitas ayuda de ningún tipo, ni siquiera soporte para el software. Puede que te confundas si eres un usuario primerizo, pues no te muestro nada acerca del propósito, ni de las tareas que se pueden realizar.
- **Qué he aprendido que necesitas hacer, la forma que prefieres hacerlo y por qué:** He aprendido que tienes una necesidad muy específica donde se procesa información de terrenos, para luego obtener una serie de imágenes que pueden ser analizadas. Sé que necesitas tener los controles estructurados y agrupados, porque para ti esta es la manera ordenada, rápida y expedita para utilizar el software. El porqué de las tareas que realizas no lo tengo claro, en casi ningún momento recibo información explícita de lo que haces.
- **Éste es el sistema que yo he diseñado para ti, y ésta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** He diseñado un software que está

estructurado en torno a los controles para configurar la tarea que realizas y la información entregada finalmente por el sistema. Este software debes utilizarlo haciendo uso de los diversos controles existentes para que finalmente, después de haber configurado la tarea, recibir una salida en forma de imágenes. Sabes que cosas tienes que hacer y eres un experto en el uso del software, en el caso contrario puedes confundirte y tener problemas al utilizar el software, e incluso sin llegar a entenderlo.

3.2.3 Análisis de resultados y conclusiones en experimentos piloto de Comunicabilidad

Los mensajes meta-comunicacionales son muy similares para ambos productos de software (GreenView y GreenLand), puesto tienen un sesgo a partir de que están implementados, al parecer, por el mismo equipo de desarrollo dada la similitud en sus interfaces. Además, cabe la posibilidad de que los dos productos cumplan funciones complementarias para el tipo de usuario que los manipula, quien es un experto en el ámbito de la meteorología y la geología.

A continuación se expone lo que puede extraerse de los mensajes meta-comunicacionales finales de los productos de software analizados, en relación a las Heurísticas Específicas para productos del tipo Grid Computing. Cabe señalar la similitud de los mensajes meta-comunicacionales finales en ambos productos de software y la relación que existe entre ellos, por lo que el siguiente análisis es de carácter global para ambos productos de software analizados (GreenView y GreenLand).

- **Aquí está mi entendimiento de quien eres:** En los mensajes meta-comunicacionales de ambos productos se hace notar a un usuario experto, quien no necesita ninguna ayuda. Esto tiene asociado la gran cantidad de problemas asociados a la Heurística H1 (Claridad), como se observa en las tablas 2.3 y 2.4 de la sección 2.2.5. Además, esto puede ser contraproducente para los nuevos usuarios del producto de software, los cuales necesitan un tiempo de entrenamiento mayor, problema que tiene asociada la Heurística H12 (Ayuda y documentación).
- **Qué he aprendido que necesitas hacer, la forma que prefieres hacerlo y por qué:** En estos mensajes meta-comunicacionales, se destaca que los productos de software GreenView y GreenLand están bien estructurados en torno a las opciones concernientes a la configuración de las tareas y la ejecución de las mismas. Esto ayuda a la utilización efectiva de los productos de software GreenView y GreenLand. De esta manera se encuentran pocos problemas asociados al grupo de Heurísticas específicas para productos del tipo Grid Computing relacionadas a la navegación (tablas 2.3 y 2.4 de la sección 2.2.5), las cuales son H6 (Atajos para funcionalidades), H7 (Reducir carga de memoria), H8 (Explorabilidad) y H9 (Control sobre acciones).

- **Éste es el sistema que yo he diseñado para ti, y ésta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** En estos mensajes meta-comunicacionales finales para los productos GreenView y GreenLand, se vuelve a destacar la estructuración de las funciones de éstos. También se señala que el propósito no se encuentra bien claro, especialmente para usuarios primerizos, que podrían no llegar a entender el uso de estos productos de software, lo cual se podría asociar a problemas relacionados con la Heurística H12 (Ayuda y documentación).

Finalmente, tras estos análisis, se puede evidenciar una relación y un impacto de la Comunicabilidad sobre la Usabilidad. De esta manera tras analizar los mensajes meta-comunicacionales finales, se pueden encontrar a nivel general problemas relacionados con las heurísticas, con lo cual se podría llegar a inferir que antes de realizar evaluaciones heurísticas sería bueno realizar inspecciones semióticas y así conocer la generalidad de los principales problemas que se podrían encontrar con las evaluaciones heurísticas.

4. Desarrollo de la Solución

4.1 Revisión de Heurísticas Específicas para Aplicaciones del Tipo Grid Computing

El análisis de los resultados entregado en 3.1.1 y 3.1.2 muestra que claramente hay un problema relacionado con la heurística genérica de Nielsen N8 (Diseño estético y minimalista), en donde hubo una alta incidencia de problemas relacionados a esta heurística que no fueron detectados con las heurísticas específicas para aplicaciones del tipo *Grid Computing*. De esta manera hubo 5 problemas con un promedio de Criticidad de 4,668 entre los cuales se encuentran 2 problemas graves, con una Criticidad de 5 y 5,25 (en una escala comprendida entre 0 y 8 para la Criticidad).

Existen otros problemas no detectados y en muy pequeño número, los cuales pueden clasificarse dentro de las heurísticas específicas para aplicaciones del tipo *Grid Computing*, los cuales pudieron haber sido detectados utilizando estas heurísticas.

Los problemas no detectados pueden ser reclasificados en heurísticas específicas para aplicaciones del tipo *Grid Computing* (2.2.2) para así visualizar de mejor manera fallas que puedan tener estas heurísticas. La reclasificación general de los problemas no detectados se puede observar en las siguientes tablas:

Tabla 4.1: Listado de problemas encontrados exclusivamente con las heurísticas genéricas de Nielsen reclasificados en las heurísticas específicas para aplicaciones del tipo *Grid Computing*. Caso de Estudio GreenView.

ID	Problema	Prom. Criticidad	Heurísticas específicas asociadas
1	Exceso de elementos en ventanas	5.25	H3
9	No se previene el ingreso de valores no permitidos	5	H10
10	Parece que todos los botones fueran igual de importantes	5	No clasificable
12	Al buscar ubicaciones inexistentes sistema se queda indefinidamente buscando	4	H4
11	Sistema resulta poco intuitivo	3.75	No clasificable
3	Ayuda oculta para novatos	3.25	H12

Tabla 4.2: Listado de problemas encontrados exclusivamente con las heurísticas genéricas de Nielsen reclasificados en las heurísticas específicas para aplicaciones del tipo *Grid Computing*. Caso de Estudio GreenLand.

ID	Problema	Prom. Criticidad	Heurísticas específicas asociadas
13	Para detener un proceso hay que buscarlo	6	H6
15	Nombre de usuario desaparece en ocasiones	3.67	H4
3	Borrar parámetros	2.67	H5
16	Mal contraste entre fondo y texto	2.67	No clasificable
17	Flechas distintas para uso similar	2	H5
20	Mayúsculas y minúsculas usadas indistintamente	1.33	H5

Se revisaron las heurísticas específicas para aplicaciones del tipo *Grid Computing* [1] y se añadieron los siguientes elementos a su checklist (2.2.3):

- **H1.5:** Los elementos de la interfaz están correctamente etiquetados utilizando un lenguaje claro y representativo.
- **H3.4:** Los elementos dentro de la interfaz están correctamente ordenados y jerarquizados.
- **H4.5:** Los mensajes son entregados de manera completa y correcta (H5.5 en la nueva propuesta de heurísticas, sección 4.1.2).

4.1.1 Propuesta de heurística: Diseño Estético

Un elemento importante en los productos de software actuales es su diseño gráfico pues es lo primero que el usuario percibe al interactuar con el producto. Los elementos del diseño, en una primera instancia, deben ser claros, sin que su percepción sea impedida o dificultosa lo cual puede traer consigo problemas de Usabilidad. Junto con esto, otro aspecto importante del diseño es su orden en la pantalla o “layout”, puesto que esto tiene relación con la forma de que las personas perciben y entienden las relaciones entre los elementos [8] por lo que deben ser correctamente desplegados en la interfaz.

En la propuesta de heurísticas actual (2.2.2) no existe una heurística específica para aplicaciones del tipo *Grid Computing* que abarque la problemática anteriormente descrita. Esto puede observarse en el análisis de los problemas encontrados con las heurísticas genéricas de Nielsen donde se encontraron algunos problemas que no son clasificables dentro de las heurísticas específicas para aplicaciones del tipo *Grid Computing* (Tabla 4.1, Tabla 4.2). De esta manera se observa una incapacidad de estas heurísticas de abarcar problemas producidos por errores de visualización. Este tipo de problemas sí puede ser abarcado por la heurística genérica de Nielsen N8 (Diseño estético y minimalista), no así como las heurísticas genéricas para aplicaciones del tipo *Grid Computing*.

Dada la problemática anteriormente explicada, se propone añadir una heurística que pueda abarcarla. Esta propuesta de una nueva heurística tiene por nombre “Diseño Estético” y se define de la siguiente manera: “En una aplicación de *Grid Computing*, la información y los elementos presentados por la interfaz deben ser desplegados de manera correcta y visualmente afable.”

Para apoyar la nueva heurística definida, se genera un checklist que pueda representar de manera simple y concisa lo que la nueva heurística quiere representar. Los puntos a considerar para el checklist de la nueva heurística “Diseño Estético” son:

1. El contraste de colores dentro de la interfaz, entre los mensajes y el fondo es el adecuado para una lectura cómoda.
2. El tamaño de las fuentes es correcto para una lectura adecuada de los mensajes .
3. El diseño de los íconos es el adecuado para una visualización correcta .
4. Los elementos dentro del layout de la aplicación están correctamente alineados y tienen el tamaño adecuado para sus contenedores.
5. El layout se adapta correctamente a distintas resoluciones/tamaños de pantalla de modo que todos los mensajes y controles sigan siendo accesibles y desplegados correctamente.

4.1.2 Propuesta de heurísticas

Dada la nueva heurística, el listado general para aplicaciones del tipo *Grid Computing*, se presenta en la Tabla 4.3:

Tabla 4.3: Nuevo listado general para heurísticas para aplicaciones del tipo *Grid Computing*.

ID	Heurística	Definición
<u>Heurísticas de diseño y estética</u>		
H1	Claridad	La interfaz de una aplicación de <i>Grid Computing</i> debe ser fácil de entender, utilizando elementos gráficos, texto y lenguaje claro.
H2	Uso de metáforas	Una aplicación de <i>Grid Computing</i> debe usar metáforas apropiadas, haciendo de las posibles acciones, fáciles de entender, a través de imágenes y objetos familiares.
H3	Simplicidad	Una aplicación de <i>Grid Computing</i> debe proveer de la información necesaria para concretar una tarea de manera concisa (clara y sin elementos que puedan ser distractivos o que sobrecarguen con información).
H4	Diseño estético	En una aplicación de <i>Grid Computing</i> , la información y los elementos presentados por la interfaz deben ser desplegados de manera correcta y visualmente agradable.
H5	Retroalimentación	Una aplicación de <i>Grid Computing</i> debe mantener al usuario informado acerca del progreso de los “Jobs”, indicando tanto el estado global como detallado del sistema. La aplicación debe entregar la retroalimentación necesaria acerca de las acciones del usuario.
H6	Consistencia	Una aplicación de <i>Grid Computing</i> debe ser consistente en el uso del lenguaje y los conceptos. Las formas ingresar y visualizar los resultados debe ser consistente.
<u>Heurísticas de navegación</u>		
H7	Atajos para funcionalidades	Una aplicación de <i>Grid Computing</i> debe proveer de atajos y abreviaciones, teclas de acceso rápido o líneas de comando para usuarios expertos.
H8	Reducir carga de memoria	Una aplicación de <i>Grid Computing</i> debe mantener los comandos principales siempre disponibles. Se debe ofrecer facilidad para encontrar elementos, funciones y opciones.
H9	Explorabilidad	Una aplicación de <i>Grid Computing</i> debe minimizar la navegación y debe proveer maneras de realizar tareas fáciles, claras y naturales.
H10	Control sobre acciones	Una aplicación de <i>Grid Computing</i> debe proveer maneras de cancelar una tarea o proceso que se está ejecutando. Debe permitir deshacer y/o realizar cambios de acciones.
<u>Heurísticas de errores y ayuda</u>		
H11	Prevención de errores	Una aplicación de <i>Grid Computing</i> debe prevenir que los usuarios realicen acciones que puedan producir errores, y debe evitar confusiones que puedan llevar a estos.
H12	Ayuda para Recuperación de Errores	Una aplicación de <i>Grid Computing</i> debe proveer de mensajes claros, indicando las causas de errores y sugiriendo algún tipo de solución ante el error.
H13	Documentación de Ayuda	Una aplicación de <i>Grid Computing</i> debe tener una completa documentación en línea que sea fácil de encontrar y entender. Debe proveer de ayuda contextual y glosario de términos para usuarios novicios.

De la misma forma, cabe agregar el mapeo entre las heurísticas específicas para aplicaciones del tipo *Grid Computing* y las heurísticas genéricas de Nielsen. Basándose en el mapeo anterior, puede añadirse la nueva heurística relacionada con la heurística de Nielsen N8 (Diseño estético y minimalista), la cual puede abarcar aspectos de la nueva heurística propuesta. En este mapeo se puede observar el cumplimiento de las heurísticas genéricas de Nielsen dentro de las heurísticas específicas para aplicaciones del tipo *Grid Computing*. Si ahora se compara con el mapeo anterior [1] en la Tabla 2.2, se puede observar la heurística H3 (Simplicidad) en completa relación con la heurística genérica de Nielsen N8 (Diseño estético y minimalista) donde la heurística específica H3 (Simplicidad) no contiene ningún elemento que pueda incluir aspectos de diseño con los cuales se pueda incurrir a problemas de Usabilidad como si ocurre con la propuesta de la nueva heurística “Diseño estético”.

De esta manera, para clarificar el tipo de problemas que abarcan las heurísticas específicas para aplicaciones del tipo *Grid Computing*, incluyendo la nueva propuesta, en comparación a las heurísticas genéricas de Nielsen, se presenta el mapeo en la siguiente tabla:

Tabla 4.4: Mapeo entre la nueva lista de heurísticas específicas para aplicaciones del tipo *Grid Computing* y las heurísticas genéricas de Nielsen.

Heurísticas Grid Computing		Heurística de Nielsen asociada	
ID	Definición	ID	Definición
H1	Claridad	N2	Concordancia entre el sistema y el mundo real
H2	Uso de Metáforas		
H3	Simplicidad	N8	Diseño estético y minimalista
H4	Diseño Estético		
H5	Retroalimentación	N1	Visibilidad del estado del sistema
H6	Consistencia	N4	Consistencia y estándares
H7	Atajos para Funcionalidades	N7	Flexibilidad y eficiencia de uso
H8	Reducir Carga de Memoria	N6	Reconocer mejor que recordar
H9	Explorabilidad	N3	Control y libertad del usuario
H10	Control sobre Acciones		
H11	Prevención de Errores	N5	Prevención de errores
H12	Ayuda para Recuperación de Errores	N9	Ayuda al usuario a reconocer, diagnosticar y recuperarse de errores
H13	Documentación de Ayuda	N10	Ayuda y documentación

Dados los cambios introducidos al checklist anterior y la nueva propuesta de heurística, se presenta un nuevo checklist enfocado a las heurísticas específicas para aplicaciones del tipo *Grid Computing*, el cual se muestra a continuación en la Tabla 4.5.

Tabla 4.5: Checklist para heurísticas revisadas específicas para aplicaciones del tipo *Grid Computing*.

ID	Heurística/Definición checklist
Heurísticas de diseño y estética	
H1	Claridad
H1.1	El propósito de la aplicación es claro para todos sus usuarios objetivo
H1.2	Los controles de interfaz y elementos gráficos son fáciles de entender y su propósito es claro
H1.3	El idioma de la interfaz es familiar a los usuarios objetivo
H1.4	El lenguaje de la interfaz es claro y fácil de entender
H1.5	Los elementos de la interfaz están correctamente etiquetados utilizando un lenguaje claro y representativo
H2	Uso de metáforas
H2.1	Se utilizan metáforas que ayudan a mejorar y agilizar la comprensión del usuario sobre la aplicación
H2.2	Se utilizan metáforas sólo para conceptos o tareas simples
H2.3	Se evita usar metáforas para comprender conceptos complejos
H2.4	Las metáforas son fáciles de entender
H3	Simplicidad
H3.1	La interfaz es simple, no está sobrecargada de elementos
H3.2	No existen íconos, controles, menús, gráficos, textos u otros elementos de interfaz redundantes
H3.3	No existen elementos de interfaz puramente decorativos que no cumplan una labor
H3.4	Los elementos dentro de la interfaz están correctamente ordenados y jerarquizados
H4	Diseño estético
H4.1	El contraste de colores dentro de la interfaz, entre los mensajes y el fondo es el adecuado para una lectura cómoda
H4.2	El tamaño de las fuentes es correcto para una lectura adecuada de los mensajes
H4.3	El diseño de los íconos es el adecuado para una visualización correcta
H4.4	Los elementos dentro del layout de la aplicación están correctamente alineados y tienen el tamaño adecuado para sus contenedores
H4.5	El layout se adapta correctamente a distintas resoluciones/tamaños de pantalla de modo que todos los mensajes y controles sigan siendo accesibles y desplegados correctamente

H5	Retroalimentación
H5.1	Existen indicadores claros del estado del sistema
H5.2	Existen indicadores claros del estado de las tareas.
H5.3	El sistema reacciona claramente ante acciones del usuario
H5.4	Los indicadores de procesamiento son dinámicos
H5.5	Los mensajes son entregados de manera completa y correcta
H6	Consistencia
H6.1	El vocabulario, los controles y las imágenes se usan consistentemente a través de la interfaz de la aplicación
H6.2	Las formas de ingreso de datos mantienen consistencia
H6.3	Los resultados del procesamiento son presentados de manera consistente
<u>Heurísticas de navegación</u>	
H7	Atajos para funcionalidades
H7.1	Existen atajos a módulos y funcionalidades básicas de la aplicación
H7.2	Existe la opción de usar líneas de comando
H8	Reducir carga de memoria
H8.1	Las funcionalidades principales se encuentran siempre visibles y accesibles
H8.2	Las funcionalidades y opciones son fáciles de encontrar
H8.3	Los formularios llenados por el usuario se mantienen en memoria
H9	Explorabilidad
H9.1	Existe una secuencia clara y explícita de pasos para la realización de cada una de las tarea
H9.2	La navegación es intuitiva, fácil de entender
H9.3	Los menús son consistentes y la selección de opciones conduce a efectos predecibles
H10	Control sobre acciones
H10.1	Existe la posibilidad de cancelar acciones o tareas
H10.2	Existe la posibilidad de revertir acciones
H10.3	Las tareas canceladas se detienen de inmediato, entregando una retroalimentación apropiada

Heurísticas de errores y ayuda	
H11	Prevención de errores
H11.1	Si se solicita la carga de archivos, se especifica claramente los tipos de archivo permitidos por el sistema
H11.2	Se especifican o ejemplifican claramente los valores permitidos
H11.3	Se valida todo el ingreso de datos
H11.4	Los mensajes de error son fáciles de entender.
H11.5	Existen asistentes (wizard) para realizar tareas comunes
H12	Ayuda para Recuperación de Errores
H12.1	Se muestran mensajes de error claros a partir de la ocurrencia de estos
H12.2	Los mensajes de error se orientan hacia la resolución de los problemas
H12.3	Existen asistentes (wizard) para solucionar problemas comunes de alta complejidad
H13	Documentación de Ayuda
H13.1	Existe documentación online
H13.2	Existe manual de usuario
H13.3	Existen instrucciones para tareas comunes o básicas
H13.4	La ayuda es fácil de encontrar
H13.5	Existe ayuda contextual que guíe al usuario respecto al uso de los distintos elementos existentes

4.2 Experimentación y Pruebas

Se efectúan una nueva serie de pruebas sobre diversos casos de estudio sobre una serie de casos de prueba, algunos son productos analizados anteriormente como GreenView y GreenLand y se añaden dos nuevos, el proyecto Boinc y GreenLand_v2 (nueva versión de GreenLand, producto analizado en un trabajo anterior [1]), los cuales se describen a continuación:

Nuevo Caso de Prueba 1: El Proyecto Boinc

Este proyecto es una iniciativa de investigadores de la Universidad Berkley de California, el cual es un producto de software del tipo *Grid Computing*, bajo el concepto de computación voluntaria, donde cualquier persona puede prestar capacidad de cómputo de su computador para los distintos proyectos que se soportan.

La gran ventaja de este producto de software es la facilidad de agregar equipos a los procesos de computación de los proyectos que son mantenidos por el software Boinc.

La arquitectura y funcionamiento de este producto de software se explica de la siguiente manera:

- **Capa de proyectos:** Son los proyectos individuales e independientes que utilizan un servidor para enviar las tareas (jobs) hacia el área de la Capa de Computación. Estos proyectos son por lo general de índole científica donde es necesario realizar cálculos extremadamente complejos y demandantes de capacidad de procesamiento, como por ejemplo análisis de fenómenos climáticos y análisis de datos astronómicos.
- **Capa de coordinación:** Está compuesta por un middleware que funciona en los servidores de la Capa de Proyectos, el cual permite que se determinen las instancias y la ubicación de los diversos proyectos de computación.
- **Capa de computación:** Es quien realiza el trabajo relacionado con las tareas enviadas por la Capa de Proyectos. Los clientes se conectan a través de un software a la capa de Coordinación, donde de manera transparente, se listan y se localizan los distintos proyectos individuales. El cliente recibe esta lista y decide para que proyectos se va a utilizar su capacidad de computación. Finalmente, el cliente reparte el tiempo en desuso del procesador para repartirlo en las tareas (o parte de ellas) asignadas por los proyectos base.

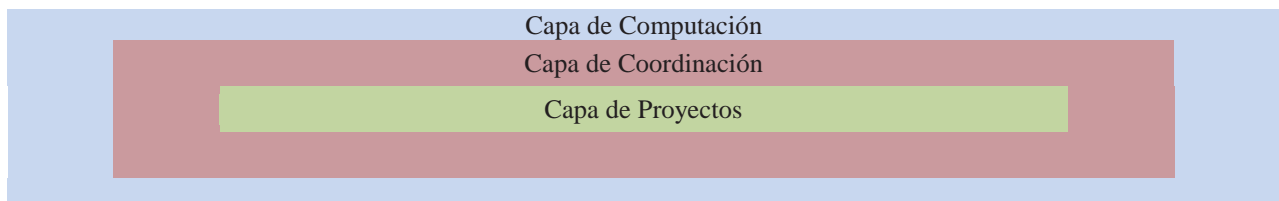


Figura 4.1: Arquitectura del software de *Grid Computing* Boinc.

De esta manera y en resumen, existe una serie de proyectos a los cuales los clientes se conectan y procesan las tareas de los proyectos que se quiere agregar capacidad de procesamiento los cuales son seleccionables de una lista.

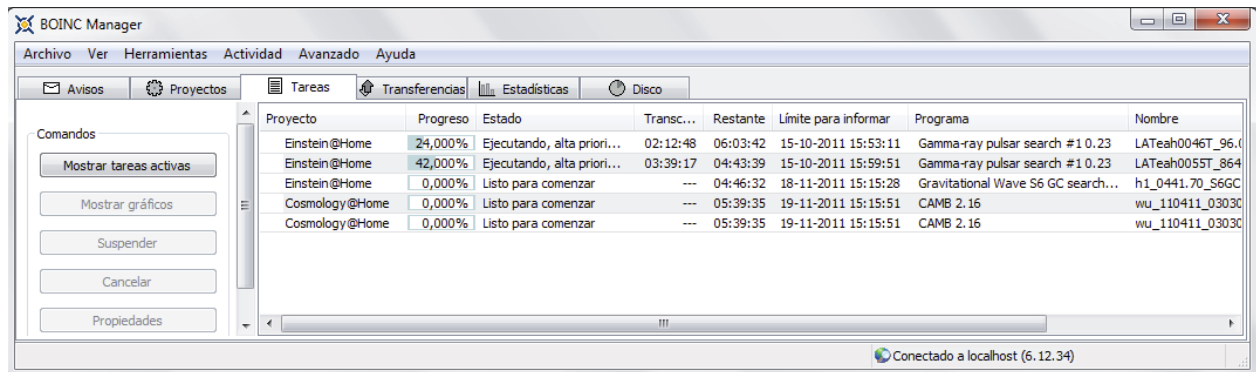


Figura 4.2: Programa cliente ejecutando una serie de tareas enviadas por los proyectos base.

Nuevo Caso de Prueba 2: GreenLand v2

Es una nueva versión del producto de software GreenLand para el manejo y el análisis de datos geoespaciales, procesamiento de imágenes/mapas, modelado espacial y visualización. Este producto ofrece soporte para diversos operadores de análisis geográfico.

Este producto de software utiliza workflows con las distintas tareas de análisis a realizar, las cuales, de manera transparente, reparte en los nodos distribuidos del sistema.

GreenLand v2 está específicamente enfocado a analizar datos pertenecientes al área geográfica de Estambul y del río Rioni, situado en Georgia [15].

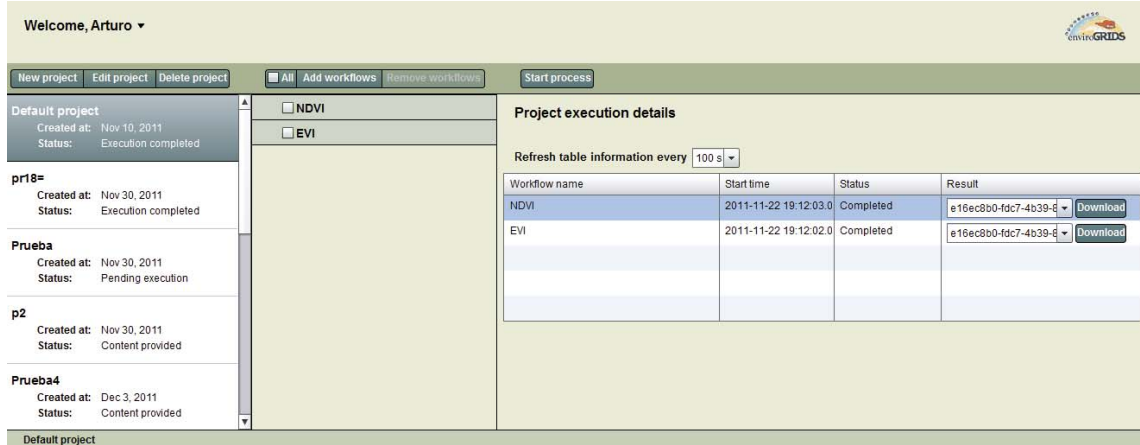


Figura 4.3: GreenLand v2 después de haber terminado un proceso.

4.2.1 Evaluación de Comunicabilidad sobre el producto GreenView

Ésta fue una prueba con usuarios tomada en laboratorio con el fin de detectar diversos quiebres comunicacionales entre el diseñador y el usuario para así, finalmente, elaborar un perfil semiótico. Los usuarios involucrados en esta prueba tienen los perfiles indicados en la Tabla 4.6.

Tabla 4.6: Perfiles de sujetos de prueba en evaluación de Comunicabilidad.

Sujeto de prueba	Ocupación	Experiencia previa/Conocimientos de computación distribuida	Experiencia previa/Conocimientos de Grid Computing
S1	Docente universitario y estudiante de doctorado en ingeniería informática	Conocimientos básicos	Ninguno
S2	Estudiante de doctorado en ingeniería informática	Conocimientos avanzados	Conocimientos básicos
S3	Egresado de ingeniería civil informática	Conocimientos básicos	Ninguno
S4	Estudiante de magister en ingeniería informática	Conocimientos básicos	Ninguno

Quiebres Comunicacionales Encontrados en la Evaluación de GreenView

A los sujetos de prueba se les entregó una serie de tareas las cuales han tenido que completar, durante su desarrollo ocurrieron una serie de quiebres comunicacionales de los 3 tipos: quiebres transitorios, parciales y totales.

Tabla 4.7: Quiebres Comunicacionales ocurridos en la prueba con usuarios.

Sujeto de prueba	Tipo de Quiebre comunicacional	Observaciones
S1, S2, S3, S4	Q. Transitorio - ¿Dónde está?	Todos los sujetos de prueba han tenido problemas en encontrar la ayuda/documentación, en distintos grados
S3	Q. Transitorio - ¿Dónde está?	Dificultad para encontrar la opción para selección de área del mapa y utilizarla
S3	Q. Transitorio - ¿Y ahora qué?	Después de ejecutar un proceso, el sujeto de prueba tiende a no realizar ninguna acción específica, sino a revisar los controles.
S3	Q. Transitorio - ¿Qué es esto?	No se entiende recuadro marcado en el mapa
S2, S4	Q. Transitorio - ¿Qué es esto?	Al revisar la información de los “Jobs” que están en ejecución/se ejecutaron
S3	Q. Transitorio - ¿Qué es esto?	Al revisar la ayuda/documentación, el sujeto de prueba la encuentra complicada y confusa
S1	Q. Transitorio - ¿Qué es esto?	Dificultad para entender la interacción con el mapa
S3	Q. Transitorio - ¡Oops!	Buscando seleccionar áreas fuera del cuadro que delimita el mapa
S3, S4	Q. Transitorio - ¿Qué pasó?	Al interactuar con el mapa tratando de seleccionar un área, les parece no funcionar (sin haber encontrado aún la opción para seleccionar un área del mapa)
S3	Q. Transitorio - ¡Auxilio!	Al tratar de seleccionar un área del mapa (sin haber encontrado aún la opción para seleccionar un área del mapa)
S1	Q. Parcial - ¡Puedo hacerlo de otra manera!	El sujeto de prueba busca otras maneras de observar la información de los “Jobs” y de rescatar más información de ellos
S3, S4	Q. Total - ¡Me rindo!	Los sujetos de prueba no pueden dar con la opción para seleccionar un área del mapa y utilizarla adecuadamente

Interpretación de Quiebres Comunicacionales

En total se detectaron 18 quiebres comunicacionales los cuales se detallan en la Tabla 4.7 y de 11 tipos, repartidos entre todos los sujetos de prueba, de esta manera, la cuantificación de los problemas se puede observar en la Tabla 4.8.

Tabla 4.8: Cuantificación de quiebres comunicacionales en GreenView tras la prueba con usuarios.

Quiebres Transitorios	
Tipo	Cantidad
¿Dónde está?	5 (27,77%)
¿Y ahora qué?	1 (5,55%)
¿Qué es esto?	5 (27,77%)
¡Oops!	1 (5,55%)
¿Qué pasó?	2 (11,11%)
¡Auxilio!	1 (5,55%)
Quiebres Parciales	
Tipo	Cantidad
¡Puedo hacerlo de otra manera!	1 (5,55%)
Quiebres Totales	
Tipo	Cantidad
¡Me rindo!	2 (11,11%)
Total:	18

Tras estos datos aportados se puede observar en la Tabla 4.8 que principalmente los usuarios tienen problemas relacionados al entendimiento de la interfaz, pues hay en gran medida quiebres de los tipos “¿Dónde está?” y “¿Qué es esto?”, pasando por quiebres del tipo “¿Qué pasó?” hasta llegar a quiebres comunicacionales totales del tipo “¡Me rindo!”. También se encontraron quiebres comunicacionales relacionados a la búsqueda y utilización de la ayuda/documentación, que para los sujetos de prueba les fue complicado hacer uso de ésta.

De esta manera, para usuarios con perfiles similares a los que tomaron la prueba (profesionales del área informática, no expertos en el área de estudio relacionadas con el producto) les es complicado y confuso el uso de la aplicación, sin llegar a entender bien los diversos controles y opciones. Estos usuarios quedan sin comprender correctamente y/o completamente el mensaje enviado por el diseñador del software, de esta manera se observa que dicho mensaje es para usuarios expertos del área de estudio comprendida del producto GreenView.

Definición del Perfil Semiótico

Basándose en las conclusiones realizadas en el análisis de los quiebres comunicacionales, se realiza un perfil semiótico para el producto GreenView.

- **¿Quiénes son los usuarios de mi producto?** Los usuarios de mi producto son expertos del área de la meteorología, que conocen bien el uso del software, quizás recibirán algún tipo de capacitación especial dada la complejidad del software.
- **¿Qué he aprendido de las necesidades (deseos) de mis usuarios?** He aprendido que mis usuarios requieren tener de una serie de pasos para obtener información del software teniendo todos los controles agrupados de manera de poder interactuar con todos ellos, pues todos son importantes.
- **¿Qué prefieren los usuarios y por qué?** Los usuarios requieren de pasos secuenciales para generar sus reportes y prefieren tener agrupadas las funciones para saber ubicarse dentro de la interfaz, porque estás acostumbrado a que tus tareas sean secuenciales, por pasos, para encontrar resultados.
- **¿Qué sistema he diseñado y cómo podrían/debieran utilizarlo?** He diseñado un sistema para usuarios expertos, que sólo requieren de ayuda muy específica, pues conocen la utilización del sistema, el cual utilizas para generar información haciendo uso de pasos específicos y secuenciales.
- **¿Cuál es mi visión de diseño?** En mi diseño tengo elementos agrupados y términos que son entendibles por el usuario experto del área. He diseñado una serie de controles para que puedas interactuar con el software.

4.2.2 Evaluación heurística sobre producto el Boinc haciendo uso de las heurísticas específicas para aplicaciones del tipo Grid Computing

Se realizó una Evaluación heurística sobre el nuevo caso de prueba, el producto Boinc, para lo cual se utilizó un cliente que tiene como propósito agregar equipos para sumar capacidad de cómputo a tareas de los distintos proyectos asociados a Boinc, los cuales suelen ser de cálculos de índole científica. Para esta evaluación se utilizaron las heurísticas refinadas para aplicaciones del tipo *Grid Computing* (4.1.2), dentro de las cuales se incluye la nueva heurística definida. Para realizar este experimento fue necesario hacer uso de un grupo de evaluadores los cuales tienen los perfiles indicados en la Tabla 4.9.

Tabla 4.9: Perfiles de evaluadores para el cliente del Proyecto Boinc.

Evaluador	Experiencia en evaluac. heurísticas	Experiencia general
E1	Media (ha participado entre 8-12 ev. heurísticas)	<ul style="list-style-type: none"> • Docente del área informática • Usuario habitual de internet • Sin conocimiento previo del software a evaluar
E2	Media (ha participado entre 8-12 ev. heurísticas)	<ul style="list-style-type: none"> • Desarrollador de software • Usuario habitual de internet • Sin conocimiento previo del software a evaluar
E3	Media (ha participado entre 6-12 ev. heurísticas)	<ul style="list-style-type: none"> • Desarrollador de páginas web • Administrador de sistema • Usuario habitual de internet • Sin conocimiento previo del software a evaluar

Se encontraron un total de 28 problemas, con un promedio de criticidad de 4,23 (en una escala de 0 a 8). En la Tabla 6.10 se puede observar el detalle del incumplimiento de las heurísticas por los problemas detectados.

Tabla 4.10: Incumplimiento de principios por los problemas detectados.

ID	Heurística/Principio de Usabilidad	Problemas que incumplen el principio	Total problemas por principio
H1	Claridad	1, 3, 4, 8, 18, 19	6 (13,33%)
H2	Uso de metáforas	1	1 (2,22%)
H3	Simplicidad	3, 4, 11	3 (6,66%)
H4	Diseño estético	3, 4, 8, 9, 11, 28	6 (13,33%)
H5	Retroalimentación	15, 29	2 (4,44%)
H6	Consistencia	11, 13, 16, 18, 19, 26, 27	7 (15,55%)
H7	Atajos para funcionalidades	14, 22	2(4,44%)
H8	Reducir carga de memoria	2, 6	2 (4,44%)
H9	Explorabilidad	21, 27	2 (4,44%)
H10	Control sobre acciones	2, 20	2 (4,44%)
H11	Prevención de errores	6, 10,21, 23, 24, 25	6 (13,33%)
H12	Ayuda para recuperación de errores	7	1 (2,22%)
H13	Documentación de ayuda	5, 15, 16, 17, 19	5 (11,1%)

En la tabla anterior (Tabla 4.10), puede observarse el funcionamiento de la nueva heurística (Diseño Estético - H4), la cual sirvió para detectar 6 problemas, lo cual valida el considerar esta nueva heurística.

Ranking de Problemas Según Criticidad

Tabla 4.11: Ranking de problemas encontrados en el cliente del Proyecto Boinc según Criticidad.

ID	Problema	Promedios		
		S	F	C
27	Archivos de configuración no disponibles	3,667	3,333	7
17	Al presionar la ayuda esta solo muestra un icono	3,333	3	6,333
20	No existe Control sobre acciones	2,667	3	5,667
19	Mensajes poco explicativos	2,667	2,667	5,333
2	Comenzar una tarea que esta lista para empezar. (Vista avanzada)	3	2	5
16	Ayuda en diferente idioma	3	2	5
25	No valida campos	2,333	2,667	5
5	Link de ayuda mal configurado. (Vista avanzada)	2	2,667	4,667
7	Hay acciones que no producen retroalimentación. (Vista avanzada)	2,333	2,333	4,667
13	Diferentes idiomas en una misma pantalla	2,667	2	4,667
1	Metáfora no representativa de proyecto. (Vista avanzada)	2,667	1,667	4,333
12	Interfaz de usuario sobrecargada. (Vista simple)	1,667	2,667	4,333
14	No existen atajos para todas las funcionalidades	2,667	1,667	4,333
15	No existe retroalimentación de lo que se está ejecutando	2,667	1,667	4,333
18	Idiomas distintos en secciones de control	2	2,333	4,333
3	Descripción larga y difícil de leer. (Vista avanzada)	2	2	4
11	Error al mostrar opciones de configuración. (Vista simple)	2	2	4
26	Eje Y promedio de uso negativo	2	2	4
4	Superposición de elementos. (Vista avanzada)	1,333	2,333	3,667
6	No existe la opción de guardar el log de eventos. (Vista avanzada)	1,667	2	3,667
21	Sección de noticias aparece cuando se presiona en mensajes	1,667	2	3,667
22	En pantalla básica, no existe atajos para funcionalidades	2	1,667	3,667
24	No prevee el error simple	2	1,333	3,333
28	Adaptación inadecuada de layout en distintos idiomas	1,333	2	3,333

8	Gráfico con colores similares. (Vista avanzada)	1,333	1,667	3
10	Gráfico con fecha errónea. (Vista avanzada)	1,667	1	2,667
23	Al pausar la aplicación el mensaje es inconsistente con el botón a presionar	1,333	1	2,333
9	Eje sin rotular en gráfico de estadísticas. (Vista avanzada)	1	1	2
Total: 28		Prom. Criticidad: 4,23		

Aspectos Positivos

- Interfaz de usuario en modo simple y avanzado, fáciles de interpretar.
- Ofrece ayuda en forma clara y precisa. En ella se explica casi todas las funciones del programa.

Aspectos Negativos

- Falta más retroalimentación sobre el estado del programa.
- Faltan algunas funciones sencillas, por ejemplo guardar el log de eventos, un panel más gráfico donde poder comenzar, eliminar, pausar, cancelar las tareas.
- Si bien el programa es para usuarios específicos, falta en la ayuda explicar aspectos técnicos y términos utilizados.

4.2.3 Evaluación Heurística Sobre el Producto Boinc Haciendo Uso de las Heurísticas Genéricas de Nielsen

Se realizó esta evaluación haciendo uso de las heurísticas genéricas de Nielsen (2.2.1) para analizar el producto de software Boinc y así para comparar los resultados con la nueva lista de heurísticas específicas para aplicaciones del tipo *Grid Computing* y de esta forma, validar su efectividad. Para esta actividad fue necesario hacer uso de un grupo de evaluadores, los cuales tienen los perfiles indicados en la Tabla 4.12.

Tabla 4.12: Perfiles de evaluadores.

Evaluador	Experiencia en evaluac. heurísticas	Experiencia general
E1	Media-Baja (ha participado entre 4-8 ev. heurísticas)	<ul style="list-style-type: none"> • Estudiante de pregrado • Usuario habitual de internet • Sin conocimiento previo del software a evaluar
E2	Media (ha participado entre 8-12 ev. heurísticas)	<ul style="list-style-type: none"> • Estudiante de Doctorado • Usuario habitual de internet • Docente • Sin conocimiento previo del software a evaluar
E3	Media (ha participado entre 6-12 ev. heurísticas)	<ul style="list-style-type: none"> • Estudiante de Doctorado • Usuario habitual de internet • Sin conocimiento previo del software a evaluar

Se encontraron un total de 23 problemas, con un promedio de criticidad de 3,6 (en una escala de 0 a 8). En la Tabla 4.13 puede observarse el detalle del incumplimiento de las heurísticas por los problemas detectados.

Tabla 4.13: Incumplimiento de principios (Nielsen) por los problemas detectados.

ID	Heurística/Principio de Usabilidad	Problemas que incumplen el principio	Total problemas por principio
N1	Visibilidad del estado del sistema	7,11,22	3 (8,33%)
N2	Similitud entre el sistema y el mundo real	6,8,15,17,21	5 (13,88)
N3	Control y libertad del usuario	1, 13,21	3 (8,33%)
N4	Consistencia y estándares	2,5,9,12,18,19	6 (16,66%)
N5	Prevención de errores	3,14,17,22	4 (11,11%)
N6	Reconocimiento más que memoria	4,20	2 (5,55%)
N7	Flexibilidad y eficiencia de uso	4,10,13,16	4 (11,11%)
N8	Diseño estético y minimalista	1,6,9,12	4 (11,11%)
N9	Ayuda al usuario para reconocer, diagnosticar y recuperarse de errores	14	1 (2,77%)
N10	Ayuda y documentación	3,4,6,10	4 (11,11%)

Ranking de problemas según Criticidad

Tabla 4.14: Ranking de problemas encontrados en el cliente del Proyecto Boinc según Criticidad.

ID	Problema	Promedios		
		S	F	C
3	No se especifica claramente que representan las pestañas superiores	2,667	2	4,667
13	Aviso que redirecciona a un sitio externo a la aplicación	2,667	2	4,667
22	Confusión al pausar proyecto	2	2,667	4,667
10	Distintas opciones para un mismo objetivo: Manager de cuentas	2,333	2	4,333
16	Mucha información se encuentra fuera del manager y que además redirecciona a una página web externa	2,667	1,667	4,333
7	No existe feedback correspondiente al comando "Actualizar"	2,333	1,667	4
8	El nombre de las actividades no genera información útil	2	2	4
12	Los avisos se encuentran en una sola pestaña para todos los proyectos	2	2	4
17	Existe inconsistencia en opciones de uso del espacio en disco	2,333	1,667	4
18	No se encuentra el botón "Reanudar"	2,333	1,667	4
19	Se presenta una opción y aparece otra cosa	2,333	1,667	4
15	La casilla "No verificar archivos de imagen" no posee explicación	2,333	1,333	3,667
2	Inconsistencia de idioma	1,667	1,667	3,333
4	Las opciones de proyectos no mantienen un formato fijo	1,667	1,667	3,333
6	Información dentro de "Propiedades del proyecto" es poco clara	2,333	1	3,333
11	La pestaña de "transferencias" no posee información descriptiva	2	1,333	3,333
21	Mensajes inconsistentes	2	1,333	3,333
9	Existe reincidencia de opciones para cambio de vistas	1,333	1,667	3
1	Solo existe una opción para cambio de skins	1,333	1,333	2,667
14	Los formatos de horario no son suficientemente claros, en inactividad por día.	1,333	1,333	2,667
20	Botones pequeños	1	1,667	2,667
23	Acción clic equivocada	1,333	1,333	2,667
5	Es poco claro el link de "Graphics Available"	1,333	1	2,333
Total: 23		Promedio Criticidad: 3,6		

Comparación de Resultados

Comparando los resultados de esta evaluación heurística y la realizada con las heurísticas específicas para aplicaciones del tipo *Grid Computing* (4.2.2) sobre el producto Boinc (tablas 4.11 y 4.14), se destacan las siguientes observaciones:

- Haciendo uso de las heurísticas específicas para aplicaciones del tipo *Grid Computing* se encontraron 5 problemas más (21,73%) que con las heurísticas genéricas de Nielsen (28 vs. 23 problemas).
- El promedio de Criticidad entregado por la evaluación realizada con las heurísticas específicas para aplicaciones del tipo *Grid Computing* es un 17,5 % mayor (Criticidad 4,23 vs. 3,6) que el promedio de Criticidad de los problemas encontrados con las heurísticas genéricas de Nielsen.
- En la evaluación realizada con las heurísticas específicas para aplicaciones del tipo *Grid Computing* se encontraron 4 problemas críticos (con una Criticidad mayor a 5, en una escala de 0 a 8) los cuales no fueron encontrados en la evaluación realizada con las heurísticas genéricas de Nielsen.
- Se encontraron 11 problemas (43,13%) en común entre la evaluación realizada con las heurísticas específicas para aplicaciones del tipo *Grid Computing* y la evaluación realizada con heurísticas genéricas de Nielsen.
- Se encontraron 17 problemas (33,33%) exclusivos en la evaluación realizada con las heurísticas específicas para aplicaciones del tipo *Grid Computing*.
- Se encontraron 10 problemas (19,6%) exclusivos en la evaluación realizada con las heurísticas genéricas de Nielsen.

Finalmente, dadas estas observaciones, se puede dilucidar que las heurísticas específicas para aplicaciones del tipo *Grid Computing* son mejores que las de Nielsen a la hora de evaluar estos productos de software específicos, pues se encontraron más problemas y además que se mostraron más importantes dada una Criticidad mayor.

4.2.4 Inspección Semiótica Sobre el Producto Boinc

Paso 1: Analizar los símbolos meta-lingüísticos y reconstruir en base a estos el mensaje de meta-comunicación según la plantilla.

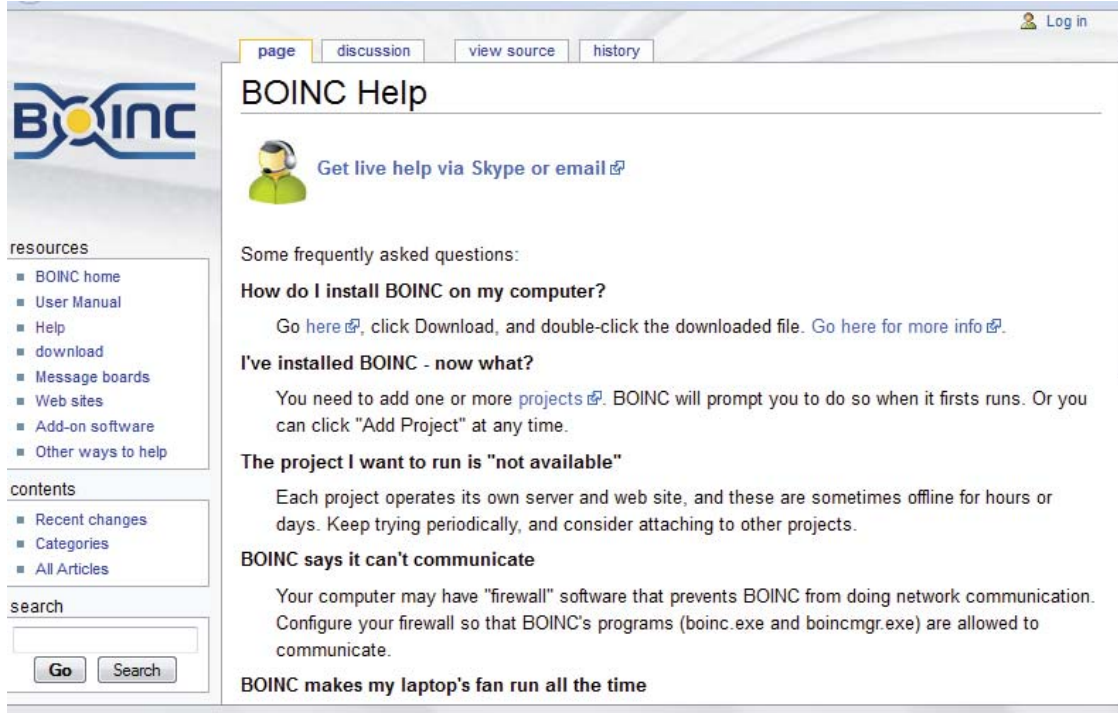


Figura 4.4: Documentación y ayuda en línea.

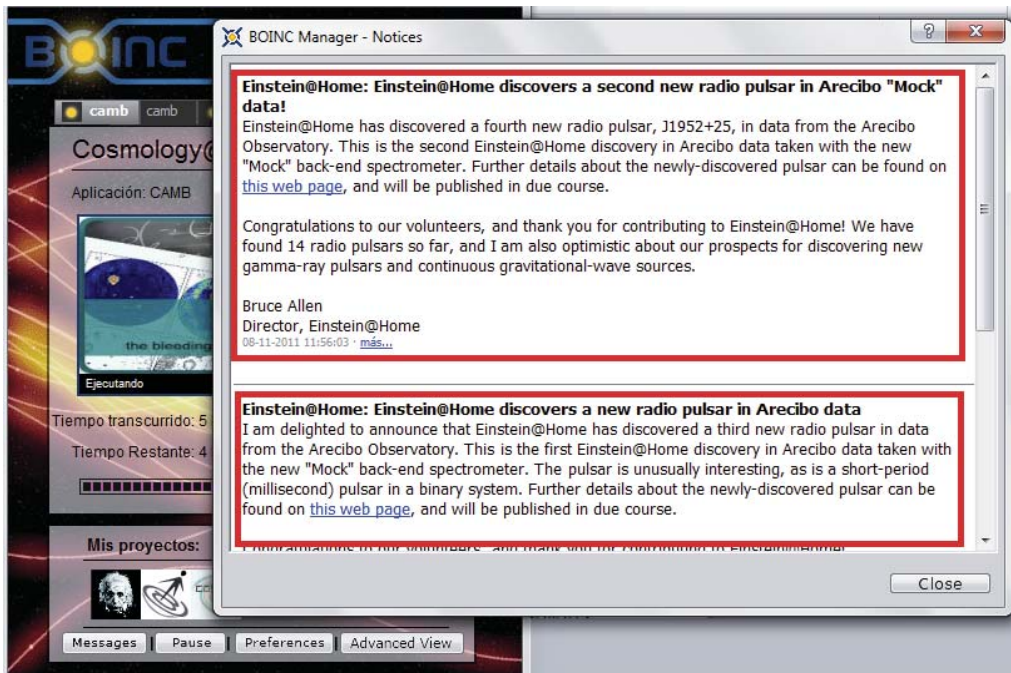


Figura 4.5: Mensajes de los administradores de proyectos.

```

source filename [arguments]          suspend [-f]
test [expr]                          time [-p] PIPELINE
times                                trap [-lp] [arg signal_spec ...]
true                                  type [-afptP] name [name ...]
typeset [-afirtx] [-p] name[=valu    ulimit [-SHacdflmnpqstuvx] [limit
umask [-p] [-S] [mode]              unalias [-a] name [name ...]
unset [-f] [-v] [name ...]          until COMMANDS; do COMMANDS; done
variables - Some variable names an   wait [n]
while COMMANDS; do COMMANDS; done   { COMMANDS ; }
debian:/home/boincadm/boinc/tools# help times
times: times
    Print the accumulated user and system times for processes run from
    the shell.
debian:/home/boincadm/boinc/tools# help jobs
jobs: jobs [-lnprs] [jobspec ...] or jobs -x command [args]
    Lists the active jobs. The -l option lists process id's in addition
    to the normal information; the -p option lists process id's only.
    If -n is given, only processes that have changed status since the last
    notification are printed. JOBSPEC restricts output to that job. The
    -r and -s options restrict output to running and stopped jobs only,
    respectively. Without options, the status of all active jobs is
    printed. If -x is given, COMMAND is run after all job specifications
    that appear in ARGS have been replaced with the process ID of that job's
    process group leader.
debian:/home/boincadm/boinc/tools#

```

Figura 4.6: Ayuda del servidor.

En el proyecto Boinc, las señales metalingüísticas están orientadas a la documentación y envío de mensajes por parte de los administradores. Según lo expuesto, en la documentación se explica bien el propósito del proyecto.

Mensajes meta-comunicacionales

- **Aquí está mi entendimiento de quien eres:** Más bien es un “ustedes”, son 2 tipos de usuarios, un usuario básico con pocos conocimientos y un usuario extremadamente avanzado.
- **Que he aprendido que necesitas hacer, la forma que prefieres hacerlo y porqué:** Según la documentación aprendí que necesitas añadir capacidad de computación diversos proyectos, ya sea porque eres un administrador de éstos o quieres hacerlo de manera voluntaria y así apoyar la investigación científica dentro de diversas áreas.
- **Éste es el sistema que yo he diseñado para ti, y esta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** He desarrollado un sistema en dos partes, la primera para usuarios con poca experiencia, donde puedes cumplir tu propósito de agregar capacidad de computación muy fácilmente y una segunda parte para usuarios expertos quienes son administradores de proyectos.

Paso 2: Analizar los símbolos estáticos y revisar o reconstruir en base a estos el mensaje de meta-comunicación según la plantilla.

Desde la primera pantalla del software cliente se pueden observar una serie de símbolos estáticos, que van desde estructura de los diversos módulos hasta mensajes de retroalimentación.



Figura 4.7: Vista básica del cliente Boinc.



Figura 4.8: Mensaje desplegado al colocar el puntero del mouse encima de los botones.

El software tiene una vista básica y otra avanzada. En la vista avanzada también se encuentran una gran cantidad de señales estáticas, referentes a su estructura, información y etiquetado de los diversos controles.

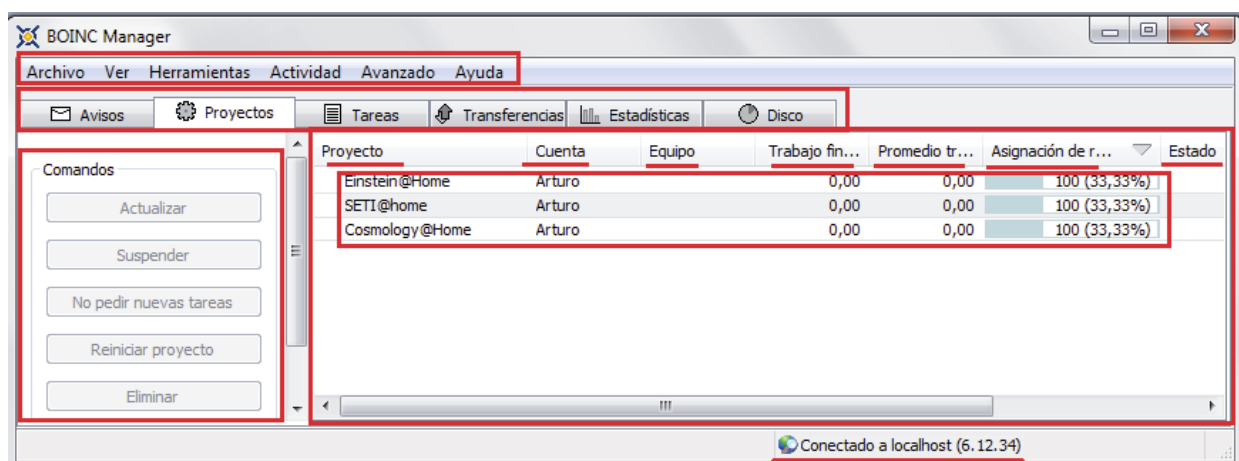


Figura 4.9: Vista avanzada, pestaña *Proyectos*.

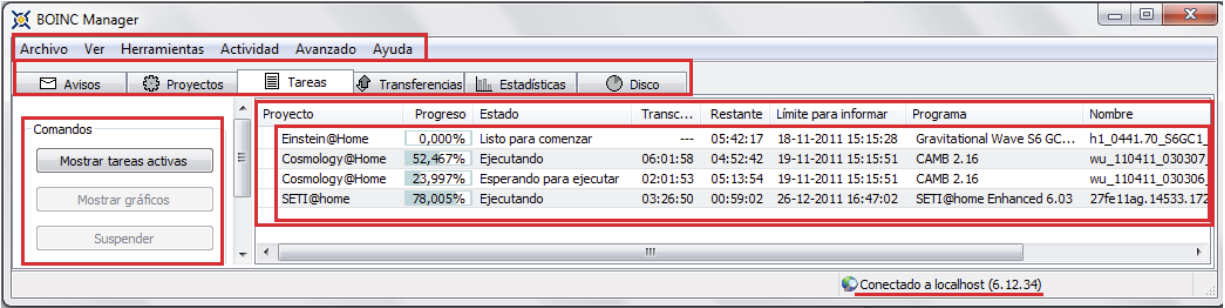


Figura 4.10: Vista avanzada, pestaña *Tareas*.

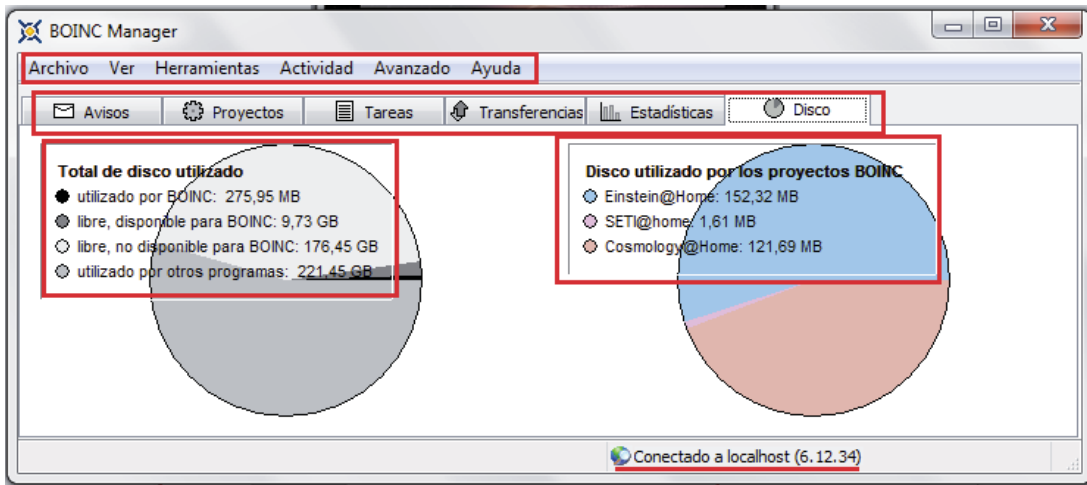


Figura 4.11: Vista avanzada, pestaña *Disco*.

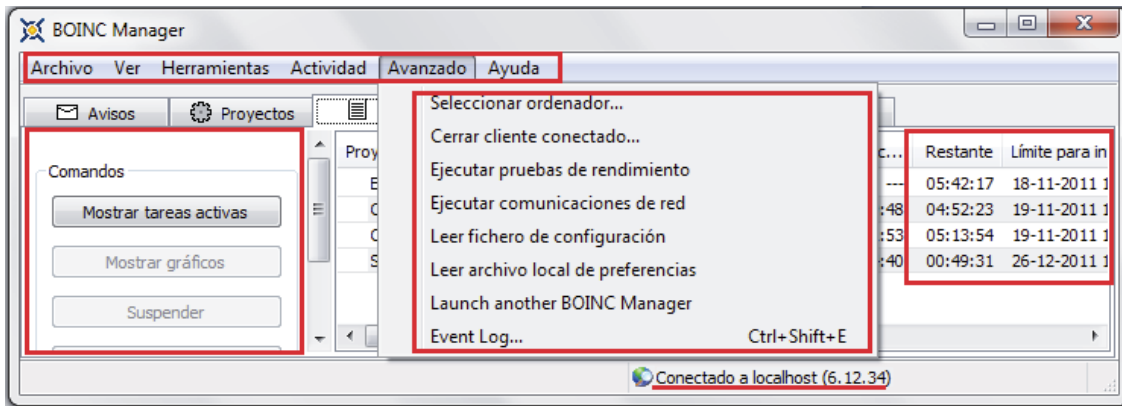


Figura 4.12: Opciones del botón *Avanzado*.

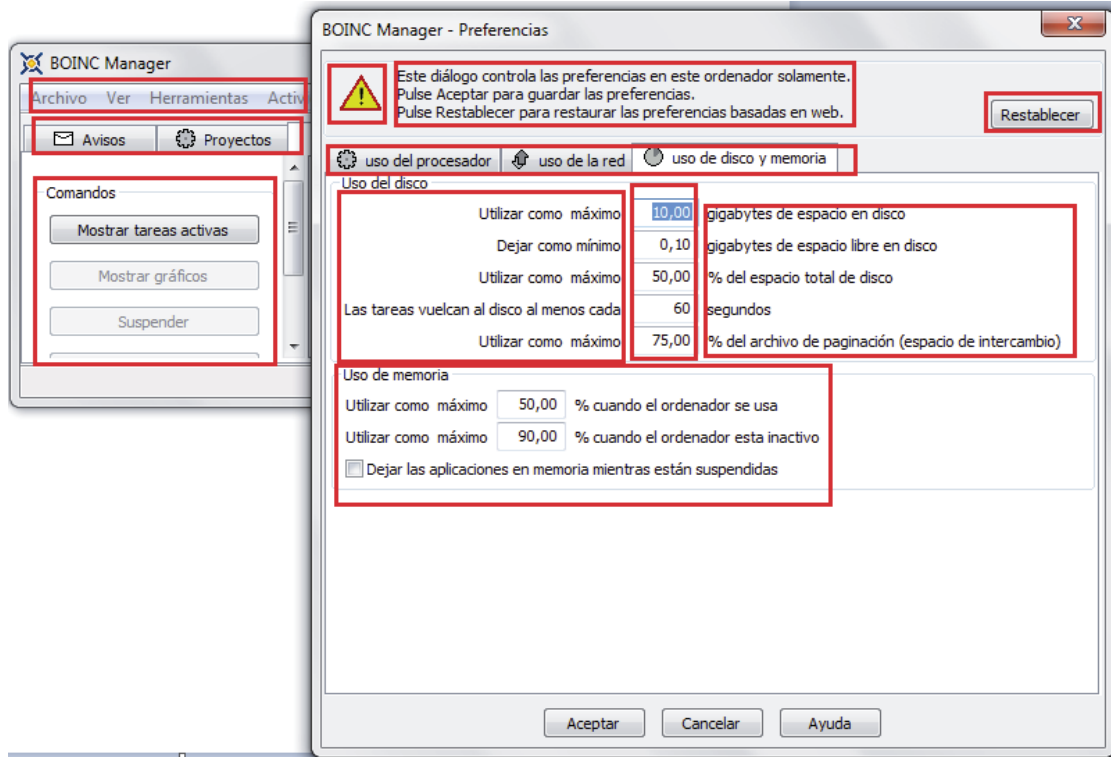


Figura 4.13: Modo avanzado, preferencias de computación.

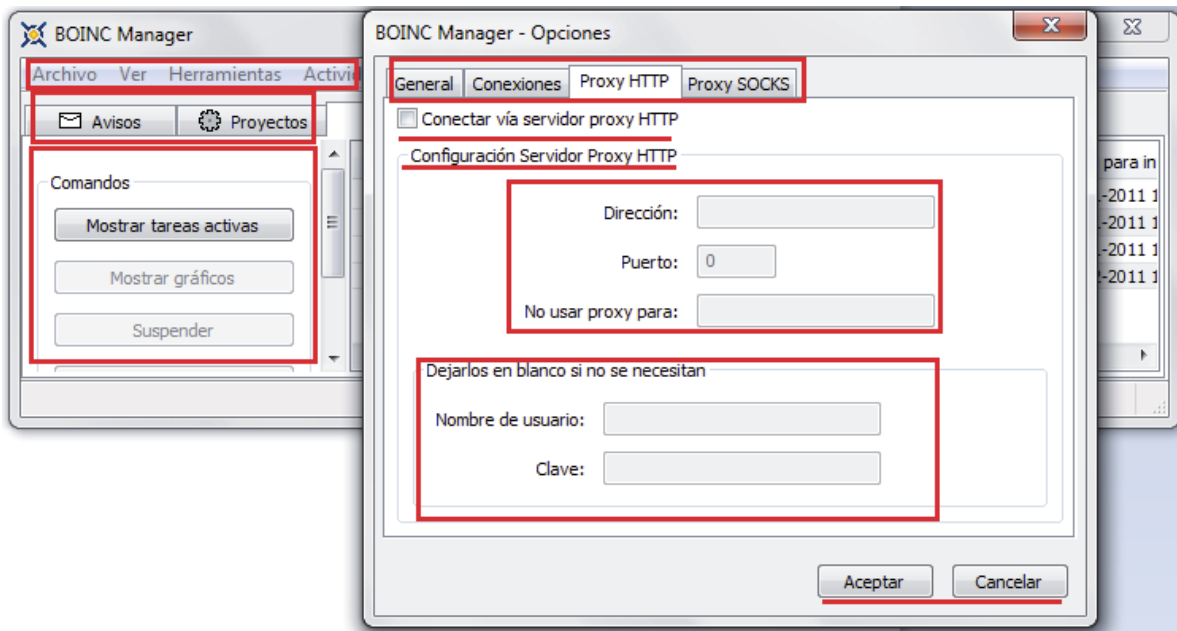


Figura 4.14: Opciones de conexión.

```

11 results
Einstein@Home albert 461 r1_1369_0_282_S4R2a_2 Uploading 1:46:47
Einstein@Home albert 461 r1_1369_0_280_S4R2a_2 Uploading 1:46:44
Einstein@Home albert 461 r1_1369_0_278_S4R2a_1 Uploading 1:46:40
Einstein@Home albert 461 r1_1369_0_276_S4R2a_0 Uploading 1:46:41
Einstein@Home albert 461 r1_1369_0_275_S4R2a_0 Uploading 1:46:34
boincsimap simap 507 60501005,001345_0 Ready to report 1:39:55
boincsimap simap 507 60501005,001737_3 Uploading 1:39:17
boincsimap simap 507 60501005,001993_3 Running 1:45 1,722% 1:38:10
boincsimap simap 507 60501005,002260_0 Running 15 0,232% 1:39:39
boincsimap simap 507 60501005,002497_2 Ready to run 1:39:51
boincsimap simap 507 60501005,002825_3 Ready to run 1:33:51

6 file transfers
Einstein@Home r1_1369_0_282_S4R2a_2_0 up 147 kB 42 fails, next try in 37:44
Einstein@Home r1_1369_0_280_S4R2a_2_0 up 148 kB 41 fails, next try in 28:14
Einstein@Home r1_1369_0_278_S4R2a_1_0 up 147 kB 44 fails, next try in 1:44:27
Einstein@Home r1_1369_0_276_S4R2a_0_0 up 146 kB 38 fails, next try in 8:12
Einstein@Home r1_1369_0_275_S4R2a_0_0 up 147 kB 37 fails, next try in 6:12
boincsimap 60501005,001737_3_0 up 1046 kB 229 kB

Recent messages
17:52:36 Started upload of file 60501005,001737_3_0
17:52:34 Starting task 60501005,002260_0 using simap version 507
17:52:34 Computation for task 60501005,001737_3 finished
17:52:34 Rescheduling CPU; application exited
17:52:30 Throughput 56704 bytes/sec
17:52:30 Finished upload of file 60501005,001345_0_0
17:52:08 Started upload of file 60501005,001345_0_0
17:51:09 Access to reference site succeeded - project servers may be temporarily down.
17:51:08 Backing off 1 minutes and 0 seconds on upload of file 60501005,001345_0_0
17:51:08 Temporarily failed upload of 60501005,001345_0_0; http error
17:51:07 Project communication failed: attempting access to reference site
17:51:06 Started upload of file 60501005,001345_0_0
17:51:04 Starting task 60501005,001993_3 using simap version 507
[1 - update boincsimap] r task 60501005,001345_0 finished

```

Figura 4.15: Mensajes del servidor.

El producto de *Grid Computing* Boinc es extremadamente rico en señales estáticas, donde se muestran los controles bien etiquetados y ordenados. El servidor, de igual forma entrega mensajes al administrador.

Mensajes meta-comunicacionales

- **Aquí está mi entendimiento de quien eres:** Más bien es un “ustedes”, son 2 tipos de usuarios, un usuario básico con pocos conocimientos y un usuario extremadamente avanzado.
- **Que he aprendido que necesitas hacer, la forma que prefieres hacerlo y porqué:** He aprendido que necesitas manejar el software de manera fácil, especialmente si eres un cliente conectándote a la red. Necesitas tener los controles a la mano y bien ordenados pues necesitas que sea así si eres un cliente. Si eres un servidor, también necesitas ver los mensajes de manera ordenada.
- **Éste es el sistema que yo he diseñado para ti, y esta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** Este es un sistema que he

desarrollado, que sea fácil de utilizar si es que quieres añadir capacidad de computación a un proyecto. Además el sistema es altamente modificable para que puedas añadir tus desarrollos propios y así beneficiarte de la capacidad de computación entregada.

Paso 3: Analizar los símbolos dinámicos y revisar o reconstruir en base a estos el mensaje de meta-comunicación según la plantilla.

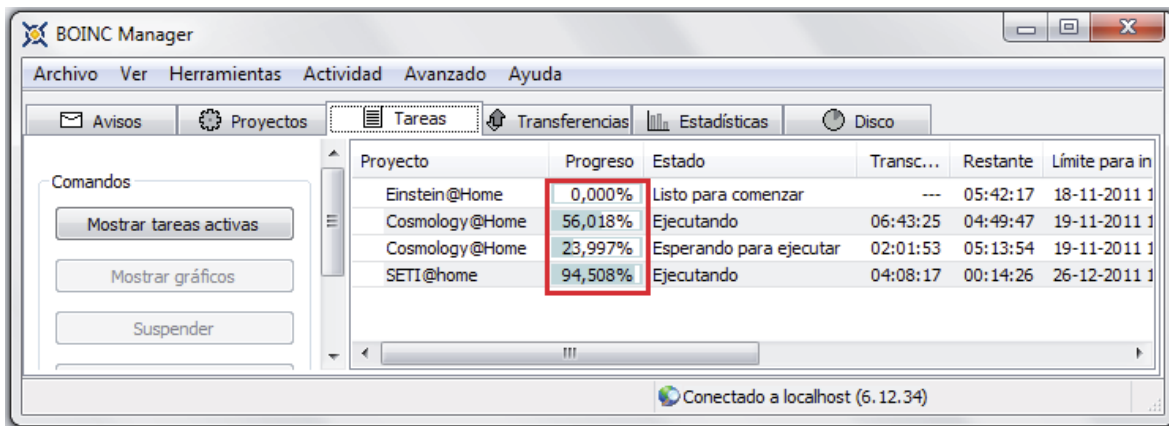


Figura 4.16: Retroalimentación acerca del progreso constante de las tareas.

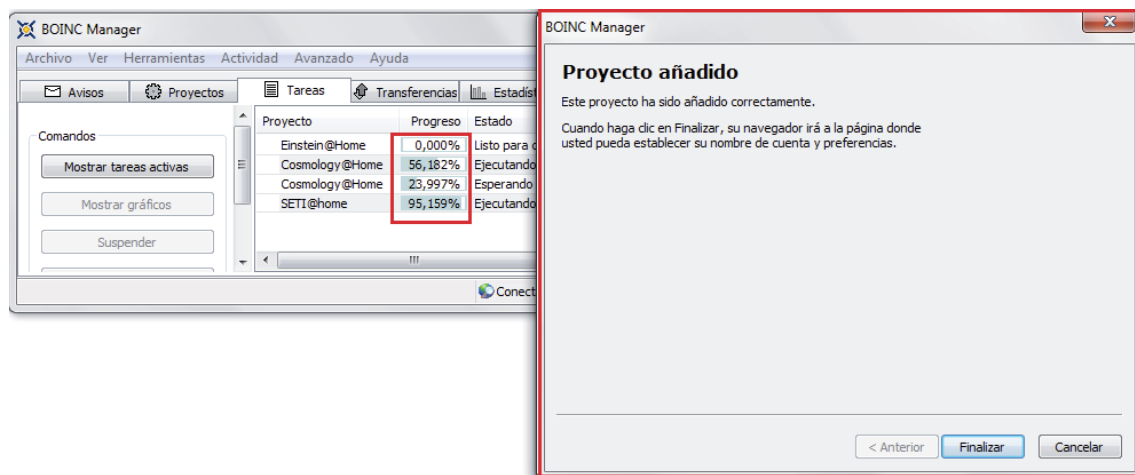


Figura 4.17: Despliegue de mensaje de “Proyecto añadido”, ahora el equipo añadirá a sus procesos tareas de otro proyecto.


```

8d3-546781a001b0 )
[Wed Aug 31 21:19:16 2011] [info] Got storeJobOutputDir
[Wed Aug 31 21:19:16 2011] [info] Starting uploading output for job (ID: 1cb7c407-1852-4e14-b8d3-546781a001b0)
[Wed Aug 31 21:19:16 2011] [info] Uploaded /tmp/agentWorkDir/1cb7c407-1852-4e14-b8d3-546781a001b0/1cb7c407-1852-4e14-b8d3-546781a001b0.tgz for the job.
[Wed Aug 31 21:19:16 2011] [info] *****
[Wed Aug 31 21:19:16 2011] [info] Reporting to jobmanager (jmboinc-lb@cernvm-messaging.cern.ch/boincalphajm-01) that the job (ID: 1cb7c407-1852-4e14-b8d3-546781a001b0) has been completed
[Wed Aug 31 21:19:16 2011] [info] *****

[Wed Aug 31 21:19:16 2011] [info] Scheduling new job request
[Wed Aug 31 21:19:16 2011] [info] Asking jmboinc-lb@cernvm-messaging.cern.ch for an adress of the job manager
[Wed Aug 31 21:19:17 2011] [info] Got have_getJob from jmboinc-lb@cernvm-messaging.cern.ch/boincalphajm-01
[Wed Aug 31 21:19:17 2011] [info] Job manager (jmboinc-lb@cernvm-messaging.cern.ch/boincalphajm-01) requires /cvmfs/sft.cern.ch/lcg/external/Boost/1.44.0_python2.6/i686-slc5-gcc43-opt/include/boost-1.44/boost/type_traits/remove_cv.hpp to be present. Doing the check.
[Wed Aug 31 21:19:17 2011] [info] The file is present. Proceeding with the job request
[Wed Aug 31 21:19:17 2011] [info] Asking jmboinc-lb@cernvm-messaging.cern.ch/boincalphajm-01 for a job
[Wed Aug 31 21:19:17 2011] [info] Got runJob
[Wed Aug 31 21:19:17 2011] [info] Got job to run with ID bc149a65-39e1-4117-a953-631d72123bd8
[Wed Aug 31 21:19:18 2011] [info] Starting fetching input files for job (ID: bc149a65-39e1-4117-a953-631d72123bd8)
[Wed Aug 31 21:19:18 2011] [info] Fetching input files for job (ID: bc149a65-39e1-4117-a953-631d72123bd8) is finished
[Wed Aug 31 21:19:18 2011] [info] Creating working directory /tmp/agentWorkDir/bc149a65-39e1-4117-a953-631d72123bd8 for the job.
[Wed Aug 31 21:19:18 2011] [info] Create wrapper script for job (ID: bc149a65-39e1-4117-a953-631d72123bd8)
[Wed Aug 31 21:19:18 2011] [info] Starting child process (PID: 2418) to execute the command for job (Job ID: bc149a65-39e1-4117-a953-631d72123bd8)

```

Figura 4.18: Mensajes entregados por el servidor al enviar Jobs.

En el producto Boinc, pueden encontrarse un escaso número de señales dinámicas, esto es porque el cliente no permite hacer cosas más allá de agregar proyectos a computar. Algo parecido puede suceder con el servidor, pero este es el que tiene el mayor potencial, siendo de código libre, un desarrollador puede realizar proyectos sobre el producto Boinc y llegar a integrar diversas interfaces para su uso. Más allá va a depender del desarrollador de proyectos.

Mensajes meta-comunicacionales

- **Aquí está mi entendimiento de quien eres:** Más bien es un “ustedes”, son 2 tipos de usuarios, un usuario básico con pocos conocimientos y un usuario extremadamente avanzado, eso lo sigo entendiendo y afirmando.
- **Que he aprendido que necesitas hacer, la forma que prefieres hacerlo y porqué:** He aprendido que como cliente no necesitas hacer mucho que sea realmente significativo para ti, tus opciones de afectar el sistema son extremadamente limitadas y es así porque tu propósito es el de solamente agregar capacidad de

cómputo. Si eres un servidor, un desarrollador, tus posibilidades son casi infinitas, puesto que he desarrollado este sistema de manera que lo modifiques y crees cosas sobre él porque así te vas a beneficiar de sus capacidades.

- **Éste es el sistema que yo he diseñado para ti, y esta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** Este es un sistema que he desarrollado, que sea fácil de utilizar si es que quieres añadir capacidad de computación a un proyecto.

Paso 4: Comparación del mensaje meta-comunicacional de los pasos 1, 2 y 3.

Las señales meta-comunicacionales más presentes en Boinc son las dinámicas, seguidas de las meta-lingüísticas. De esta manera se observa un software que está bien armado, con un propósito fijo y definido, además de tener la documentación de apoyo.

La falta de señales dinámicas se debe a que, en primer lugar, el cliente no puede introducir cambios mayores al sistema, puesto que no es su propósito. El caso del servidor, es porque es un sistema abierto para desarrolladores, que estos puedan generar sus propios proyectos, por lo que el inducir cambios al sistema y a su estado va a depender mayormente del desarrollador.

En los tres tipos de mensajes meta-comunicacionales, se observan claramente 2 perfiles de usuarios, uno sin experiencia quien es el que añade capacidad de computación al sistema y un usuario experto, quien es el desarrollador del proyecto que está haciendo uso de Boinc para computar.

Paso 5: Evaluación final y global de la Comunicabilidad en el software

Mensajes meta-comunicacionales

- **Aquí está mi entendimiento de quien eres:** Son 2 perfiles de usuarios, uno es un usuario común sin mayores conocimientos, el cual incluso dona voluntariamente su capacidad de cómputo. El segundo usuario es un desarrollador experto que saca provecho de las grandes posibilidades que tiene el sistema para realizar grandes cálculos. Junto con esto, tú desarrollas e implementas proyectos sobre Boinc.
- **Que he aprendido que necesitas hacer, la forma que prefieres hacerlo y porqué:** He aprendido que tienes tareas y propósitos bien específicos, dependiendo del tipo de usuario que eres. Si eres un cliente sólo necesitas añadir equipos al sistema y controlar la manera en que el o los equipos que añadas sean utilizados sus recursos de computación, por eso que he diseñado un software sencillo, para que puedas utilizarlo sin

dificultad. Si eres un desarrollador de proyectos, he aprendido la forma en que aprovechas los recursos del sistema Boinc, en donde tú ejecutas tus tareas y el resto las ejecuta, lo haces así porque quizás no tengas los recursos para hacer uso de la capacidad de procesamiento que necesitas.

- **Éste es el sistema que yo he diseñado para ti, y esta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** He diseñado un sistema que permita muy fácilmente agregar capacidad de procesamiento y que puede ser aprovechada por usuarios expertos cumpliendo el propósito de tus tareas.

4.2.5 Evaluación Heurística Sobre el Producto GreenLand v2 Haciendo Uso de las Heurísticas Específicas para Aplicaciones del Tipo Grid Computing

Se realizó una Evaluación heurística sobre el otro nuevo caso de prueba, el producto GreenLand v2. Para esta evaluación se utilizaron las heurísticas refinadas para aplicaciones del tipo *Grid Computing* (4.1.2), dentro de las cuales se incluye la nueva heurística definida. Ésta actividad fue realizada por un grupo de evaluadores los cuales tienen los perfiles indicados en la Tabla 4.15.

Tabla 4.15: Perfiles de evaluadores para GreenLand v2.

Evaluador	Experiencia en evaluac. heurísticas	Experiencia general
E1	Media (ha participado entre 8-12 ev. heurísticas)	<ul style="list-style-type: none"> • Docente del área informática • Usuario habitual de internet • Sin conocimiento previo del software a evaluar
E2	Media (ha participado entre 8-12 ev. heurísticas)	<ul style="list-style-type: none"> • Desarrollador de software • Usuario habitual de internet • Sin conocimiento previo del software a evaluar
E3	Media (ha participado entre 6-12 ev. heurísticas)	<ul style="list-style-type: none"> • Desarrollador de páginas web • Administrador de sistema • Usuario habitual de internet • Sin conocimiento previo del software a evaluar

Se encontraron un total de 26 problemas, con un promedio de criticidad de 4,78 (en una escala de 0 a 8). En la siguiente tabla puede observarse el detalle del incumplimiento de las heurísticas por los problemas detectados.

Tabla 4.16: Incumplimiento de principios por los problemas detectados en GreenLand v2.

ID	Heurística/Principio de Usabilidad	Problemas que incumplen el principio	Total problemas por principio
H1	Claridad	5, 6, 7, 8, 17, 18, 22, 24, 25	9 (24,32%)
H2	Uso de metáforas	0	0 (0%)
H3	Simplicidad	13	1 (2,7%)
H4	Diseño estético	9, 19	2 (5,4%)
H5	Retroalimentación	10, 11, 15, 22, 21, 23, 24	7 (18,91%)
H6	Consistencia	3, 4, 19, 23, 24, 25	6 (16,21%)
H7	Atajos para funcionalidades	16, 22	2 (5,4%)
H8	Reducir carga de memoria	0	0 (0%)
H9	Explorabilidad	12	1 (2,7%)
H10	Control sobre acciones	24	1 (2,7%)
H11	Prevención de errores	21, 24, 26	3 (8,1%)
H12	Ayuda para recuperación de errores	2, 24, 26	3 (8,1%)
H13	Documentación de ayuda	1, 21	2 (5,4%)

En la tabla anterior, nuevamente puede observarse el funcionamiento de la nueva heurística (Diseño Estético - H4), la cual sirvió para detectar 2 problemas, lo cual complementa la validación de esta nueva heurística.

Ranking de problemas según Criticidad

Tabla 4.17: Problemas encontrados en GreenLand v2 ordenados por promedio de Criticidad.

ID	Problema	Promedios		
		S	F	C
1	Ayuda no disponible	3,333	3,667	7
13	Resultados poco claros	3,333	3,333	6,667
8	Información provista incompleta	3,667	2	5,667
19	No se ejecutan todos los procesos	3,667	2	5,667
26	Tareas enviadas y no procesadas no se pueden cancelar	4	2,333	5,667

2	Recuperar contraseña no disponible	3	2,333	5,333
4	Aplicación sólo en un idioma	2,333	3	5,333
9	Aplicación no se adapta a la resolución	3	2,333	5,333
20	No hay respuesta del sistema	3,333	2	5,333
21	Ejecución de proyectos sin carga de archivo	3,333	2	5,333
22	No es posible configurar el sistema	3	2,333	5,333
12	No hay posibilidad de volver al estado anterior	2,333	2,667	5
14	No deja claro si el botón está activo o no	2	3	5
6	Mensaje de error "ofensivo"	2,667	2	4,667
10	Elementos sin descripción	2,333	2,333	4,667
11	Valores sin descripción o con nombres poco significativos	2,333	2,333	4,667
24	Mensaje confuso y error de sistema	2,333	2,333	4,667
15	Retroalimentación poco clara	2,333	2	4,333
16	No posee atajos	1,333	3	4,333
3	Enlace de contacto me obliga a instalar un cliente	2,667	1,333	4
5	Mensaje de error que no es claro	2	2	4
23	No se actualiza la descripción	2,333	1,667	4
7	Campo de contacto no da toda la información	2,667	1	3,667
18	Falta descripción	2	1,333	3,333
17	Logo poco claro	1	2	3
25	Caracteres no soportados	1,333	1	2,333
Total: 26		Promedio Criticidad: 4,78		

Aspectos positivos

- Buena retroalimentación e información en pantalla sobre funciones
- Diseño simple y minimalista, lo cual deja bien claro los pasos a seguir para utilizar el software

Aspectos negativos

- No existe ayuda interna, dentro del software

- Sistema para usuario experto, demasiado técnico para un novato

4.2.6 Evaluación Heurística Sobre el Producto GreenLand v2 Haciendo Uso de las Heurísticas Genéricas de Nielsen

Esta evaluación se realizó haciendo uso de las heurísticas genéricas de Nielsen (2.2.1) para analizar el producto de software GreenLand v2. El objetivo de esto es lograr diversas observaciones sobre la comparación de las dos evaluaciones, la realizada con las heurísticas específicas para aplicaciones del tipo Grid Computing y la realizada con las heurísticas genéricas de Nielsen. Para realizar esta actividad fue necesario hacer uso de un grupo de evaluadores, los cuales tienen los perfiles indicados en la Tabla 4.18.

Tabla 4.18: Perfiles de evaluadores (Nielsen).

Evaluador	Experiencia en evaluac. heurísticas	Experiencia general
E1	Media-Baja (ha participado entre 4-8 ev. heurísticas)	<ul style="list-style-type: none"> • Estudiante de pregrado • Usuario habitual de internet • Sin conocimiento previo del software a evaluar
E2	Media (ha participado entre 8-12 ev. heurísticas)	<ul style="list-style-type: none"> • Estudiante de Doctorado • Usuario habitual de internet • Docente • Sin conocimiento previo del software a evaluar
E3	Media (ha participado entre 6-12 ev. heurísticas)	<ul style="list-style-type: none"> • Estudiante de Doctorado • Usuario habitual de internet • Sin conocimiento previo del software a evaluar

Se encontraron un total de 19 problemas, con un promedio de criticidad de 3,96 (en una escala de 0 a 8). En la Tabla 4.19 puede observarse el detalle del incumplimiento de las heurísticas por los problemas detectados.

Tabla 4.19: Incumplimiento de principios (Nielsen) por los problemas detectados.

ID	Heurística/Principio de Usabilidad	Problemas que incumplen el principio	Total problemas por principio
N1	Visibilidad del estado del sistema	1,13,14,15,16,18	6
N2	Similitud entre el sistema y el mundo real	4,6,10,12	4
N3	Control y libertad del usuario	5,8	2
N4	Consistencia y estándares	4,5	2
N5	Prevención de errores	1,2,6,16,19	5
N6	Reconocimiento más que memoria	0	0
N7	Flexibilidad y eficiencia de uso	6	1
N8	Diseño estético y minimalista	7,9,11	3
N9	Ayuda al usuario para reconocer, diagnosticar y recuperarse de errores	3,13,17	3
N10	Ayuda y documentación	10,13	2

Ranking de Problemas Según Criticidad

Tabla 4.20: Problemas encontrados en GreenLand v2 ordenados por promedio de Criticidad.

ID	Problema	Promedios		
		S	F	C
1	Inicialización lenta de la aplicación sin retroalimentación adecuada	3,667	3,333	7
13	No se genera una lista de errores, o faltantes para hacer correr el proyecto	3,333	2,333	5,667
2	Permite agregar workflows, aun no estando cargada la lista de proyectos.	2,667	2,333	5
6	La restricción de archivos al subir al servidor, es poco clara	3	2	5
10	Los inputs del workflow no posee explicación ni diferencias entre ellos	2,667	2	4,667
5	Imágenes perdidas, botones sin nombre	3	1,333	4,333
14	Elementos de la interfaz no se cargan	2	2,333	4,333
16	Eliminación no funciona	2,333	2	4,333
19	Problemas al iniciar proyecto	2,333	1,667	4
3	El mensaje de error es poco claro, ni otorga solución	2	1,667	3,667
17	Aplicación no responde	2,667	1,333	3,667
18	Botones parecen no funcionar	2,667	1,667	3,667
12	El nombre del "Result" no describe nada	2,667	1,667	3,333

4	El nombre del botón es incomprensible	1,667	1,333	3
8	No permite el libre scroll si se tiene seleccionado un proyecto	1,667	1,333	3
11	La tabla de Project Execution Result se encuentra cortada por la columna	1,667	1,333	3
15	Botones no ejecutan ninguna acción	1,333	1,667	3
7	El cuadro de "perfil" se descuadra con la información	1,333	1,333	2,333
9	El ícono del mouse no cambia al pasar por los distintos botones	1	1,333	2,333
Total: 19		Promedio Criticidad: 3,96		

Comparación de Resultados

Comparando los resultados de esta evaluación heurística y la realizada con las heurísticas específicas para aplicaciones del tipo *Grid Computing* (4.1.2) sobre el producto GreenLand v2 (tablas 4.17 y 4.20), se destacan las siguientes observaciones:

- Haciendo uso de las heurísticas específicas para aplicaciones del tipo *Grid Computing* se encontraron 7 problemas más (36,84%) que con las heurísticas genéricas de Nielsen (26 vs. 19 problemas).
- El promedio de Criticidad entregado por la evaluación realizada con las heurísticas específicas para aplicaciones del tipo *Grid Computing* es un 20,7 % mayor (Criticidad 4,78 vs. 3,96, en una escala de 0 a 8) que el promedio de Criticidad de los problemas encontrados con las heurísticas genéricas de Nielsen.
- En la evaluación realizada con las heurísticas específicas para aplicaciones del tipo *Grid Computing* se encontraron 7 problemas críticos (con una Criticidad mayor a 5, en una escala de 0 a 8) los cuales no fueron encontrados en la evaluación realizada con las heurísticas genéricas de Nielsen.
- En la evaluación realizada con las heurísticas genéricas de Nielsen se encontraron 2 problemas críticos (con una Criticidad mayor a 5, en una escala de 0 a 8) los cuales no fueron encontrados en la evaluación realizada con las heurísticas específicas para aplicaciones del tipo *Grid Computing*.
- Se encontraron 10 problemas (37,77%) en común entre la evaluación realizada con las heurísticas específicas para aplicaciones del tipo *Grid Computing* y la evaluación realizada con heurísticas genéricas de Nielsen.
- Se encontraron 18 problemas (40%) exclusivos en la evaluación realizada con las heurísticas específicas para aplicaciones del tipo *Grid Computing*.

- Se encontraron 9 problemas (20%) exclusivos en la evaluación realizada con las heurísticas genéricas de Nielsen.

Finalmente, dadas estas observaciones, se puede dilucidar que las heurísticas específicas para aplicaciones del tipo *Grid Computing* siguen siendo mejores que las de Nielsen (complementando esta conclusión con lo expuesto al final de la sección 4.2.3, en la página 68) a la hora de evaluar estos productos de software específicos, pues se encontraron más problemas y además que se mostraron más importantes dada una Criticidad mayor.

4.2.7 Prueba de Usabilidad Sobre el Producto GreenLand v2

El producto GreenLand v2 fue sometido a una prueba de Usabilidad con el propósito de detectar nuevos problemas y corroborar los detectados anteriormente. Para esta actividad se hizo uso de un grupo de sujetos de prueba, los cuales tienen los perfiles mencionados en la Tabla 4.21.

Tabla 4.21: Perfiles de usuario para prueba de Usabilidad en producto GreenLand v2.

Sujeto de prueba	Ocupación	Experiencia previa/Conocimientos de computación distribuida	Experiencia previa/Conocimientos de Grid Computing
S1	Estudiante de ingeniería informática, trabajador a medio tiempo	Sin conocimientos	Ninguno
S2	Estudiante de ingeniería informática	Sin conocimientos	Ninguno
S3	Estudiante de magister en ingeniería informática	Conocimientos básicos	Ninguno
S4	Estudiante de magister en ingeniería informática	Conocimientos básicos	Ninguno

Los sujetos de pruebas participantes, tienen un perfil en común, todos son estudiantes del área informática y cuentan con pocos conocimientos del área de computación distribuida, lo que para el uso de GreenLand v2, puede ser tomado como un grupo de usuarios novatos.

Tareas

Las tareas a realizar en GreenLand v2 por los sujetos de prueba están enfocadas al entendimiento del software y su uso, por lo que las tareas son de búsqueda de información y la ejecución de un proceso en el software.

Tabla 4.22: Tareas realizadas por los sujetos de prueba.

ID	Tarea
1	Buscar y ejecutar video de presentación
2	Buscar la documentación y describir lo que hace el software
3	Hacer login en la aplicación, navegar y familiarizarse con los controles
4	Ejecutar un proceso

Resultados de la Prueba

En la primera tarea, todos los usuarios la completaron dentro del tiempo establecido, menos uno el cual tuvo un poco de dificultad para encontrarlo. Un problema detectado, es la necesidad de codecs de video instalados en el equipo, pues un usuario tuvo problemas al reproducir el video dado la falta del codec.

La segunda tarea fue completada por todos los usuarios en un tiempo promedio de 6 minutos y 18 segundos. Encontraron la documentación y la revisaron, pero ningún usuario pudo describir correctamente la función y el propósito del software, no pudo entenderse.

La tercera tarea fue completada por los usuarios, los cuales en poco tiempo se familiarizaron con los controles del software.

La cuarta tarea de la prueba de Usabilidad fue completada también exitosamente por todos los sujetos de prueba, se observó en dos de ellos alguna dificultad para guardar los parámetros de los procesos.

Cuestionario Post-Test

Al completar la prueba, a los usuarios se les entregó una encuesta acerca de su experiencia con el uso de GreenLand v2, los resultados se expresan en los siguientes gráficos:

1. ¿Pudo completar las tareas?

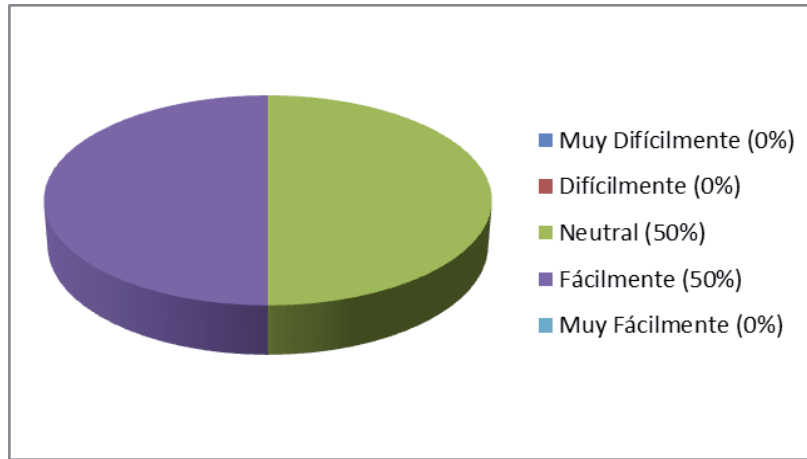


Figura 4.19: Gráfico de cuestionario post-test, pregunta 1.

2. ¿Considera que la información requerida en la prueba ha sido fácil de encontrar?

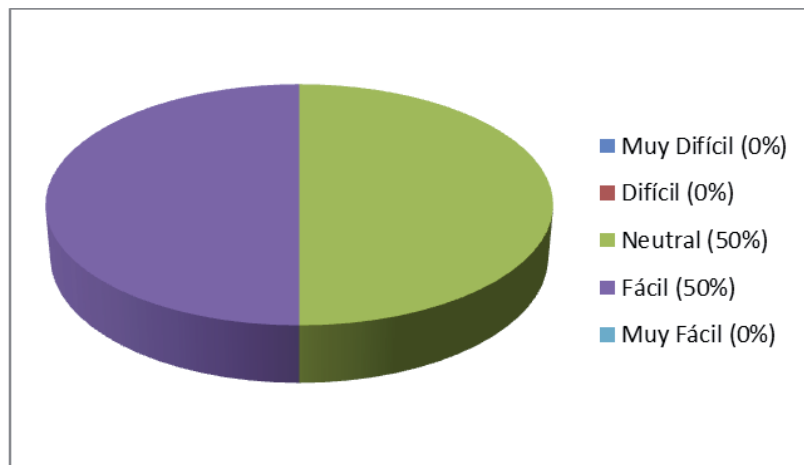


Figura 4.20: Gráfico de cuestionario post-test, pregunta 2.

3. ¿Considera que la información ha sido útil?

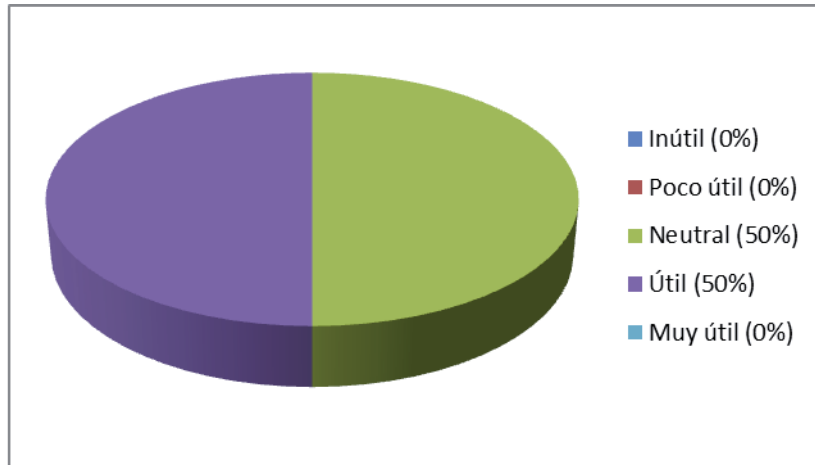


Figura 4.21: Gráfico de cuestionario post-test, pregunta 3.

4. ¿Considera que el software es fácil de entender?

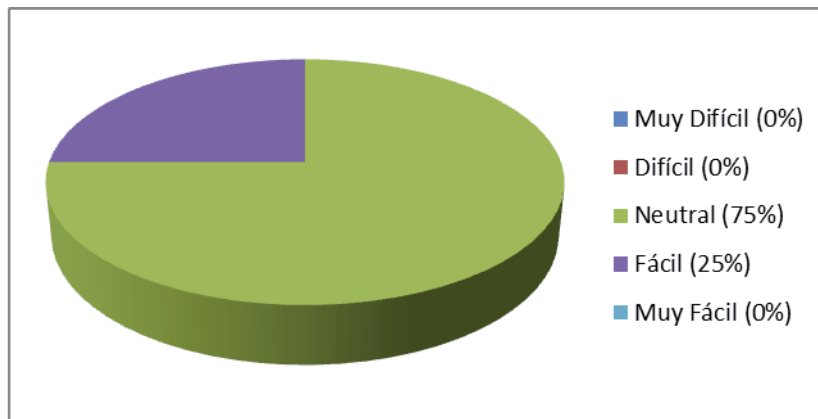


Figura 4.22: Gráfico de cuestionario post-test, pregunta 4.

5. ¿Considera que el software es fácil de utilizar?

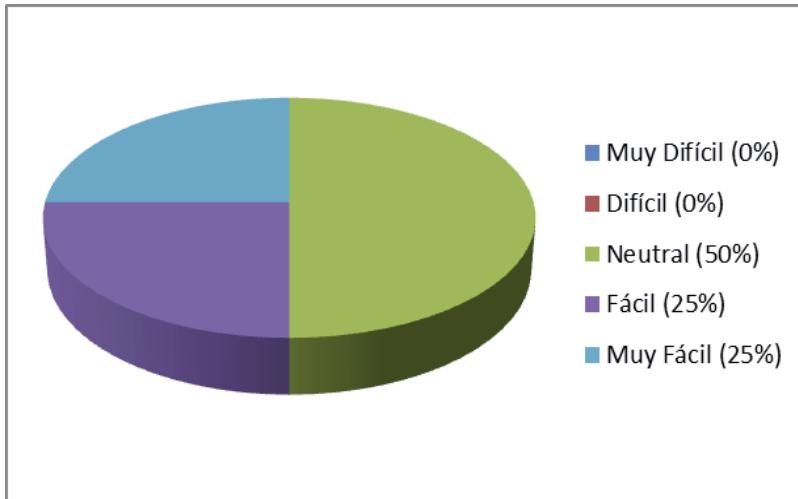


Figura 4.23: Gráfico de cuestionario post-test, pregunta 5.

Resumiendo los resultados del cuestionario post-test, según lo expresado por los sujetos de prueba, su experiencia fue neutral, con algunos puntos positivos. De esta manera los sujetos de prueba expresaron que el software no es difícil de utilizar y es relativamente entendible para ellos.

Junto con esto, los sujetos de prueba expresaron encontraron sencilla y ordenada la manera que GreenView v2 trabaja con workflows, pero a su vez mostraron algo de disgusto que la aplicación estuviese solamente en inglés.

4.2.8 Inspección Semiótica sobre producto el GreenLand v2

Paso 1: Analizar los símbolos meta-lingüísticos y reconstruir en base a estos el mensaje de meta-comunicación según la plantilla.

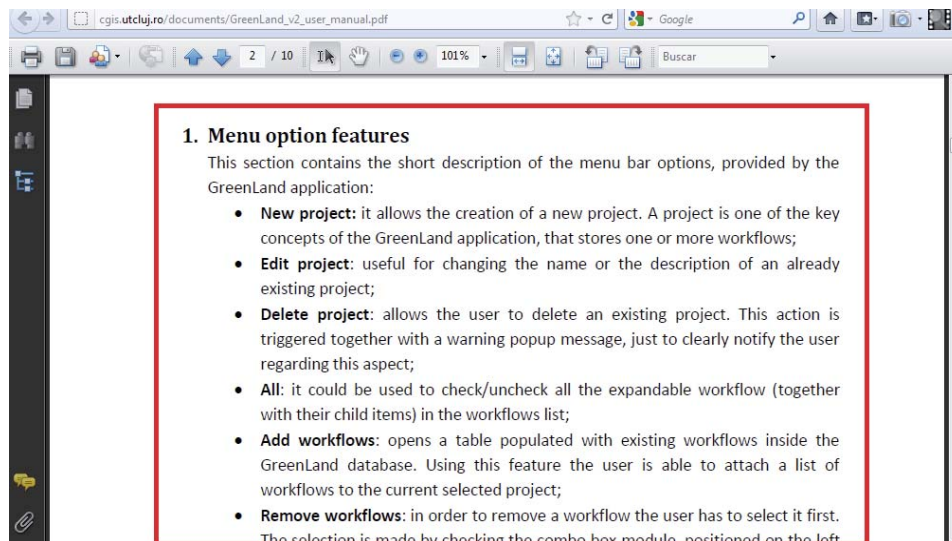


Figura 4.24: Ayuda externa del producto GreenLand v2.

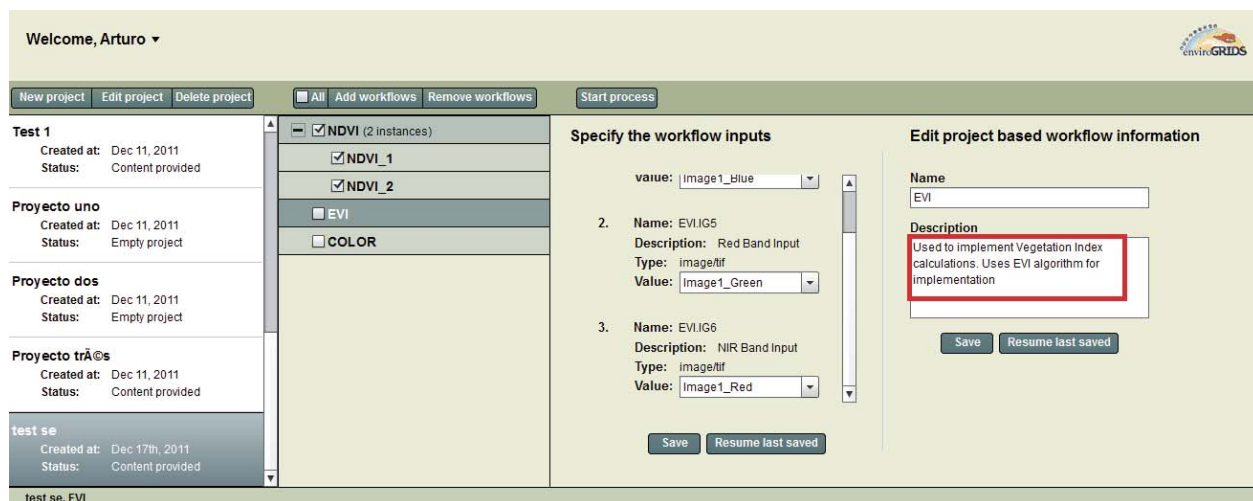


Figura 4.25: Símbolo metalingüístico de descripción de un elemento.

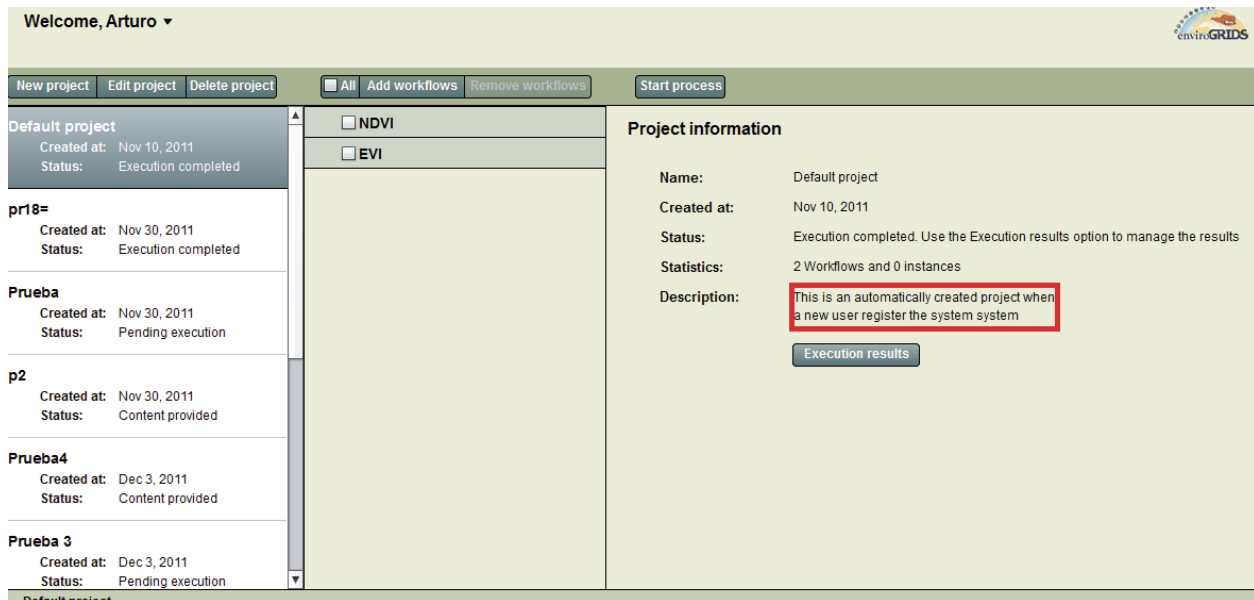


Figura 4.26: Símbolo metalingüístico de descripción del proyecto.

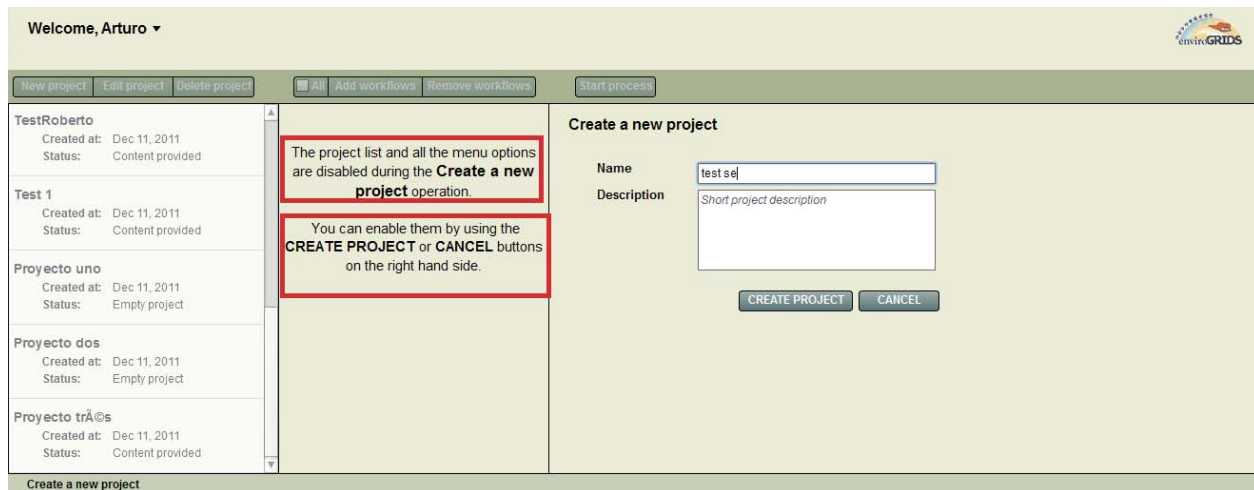


Figura 4.27: Símbolo metalingüístico de descripción de funciones.

En el producto GreenLand v2, las señales metalingüísticas están orientadas a la documentación externa y a ciertas pequeñas descripciones de elementos del software. La documentación explica las tareas que puede realizar el software.

Mensajes meta-comunicacionales

- **Aquí está mi entendimiento de quien eres:** Eres un usuario avanzado, con amplios conocimientos de tu área de estudio.
- **Qué he aprendido que necesitas hacer, la forma que prefieres hacerlo y por qué:** Según la documentación aprendí que necesitas procesar imágenes geográficas de cierta zona muy específica y prefieres hacer las cosas en una secuencia bien definida. Quieres tener una gran cantidad de funciones, todas agrupadas y ordenadas.
- **Éste es el sistema que yo he diseñado para ti, y ésta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** He desarrollado un sistema ordenado que te permite realizar tus tareas en un orden bien definido para tu propósito de generar y analizar imágenes satelitales.

Paso 2: Analizar los símbolos estáticos y revisar o reconstruir en base a estos el mensaje de meta-comunicación según la plantilla.

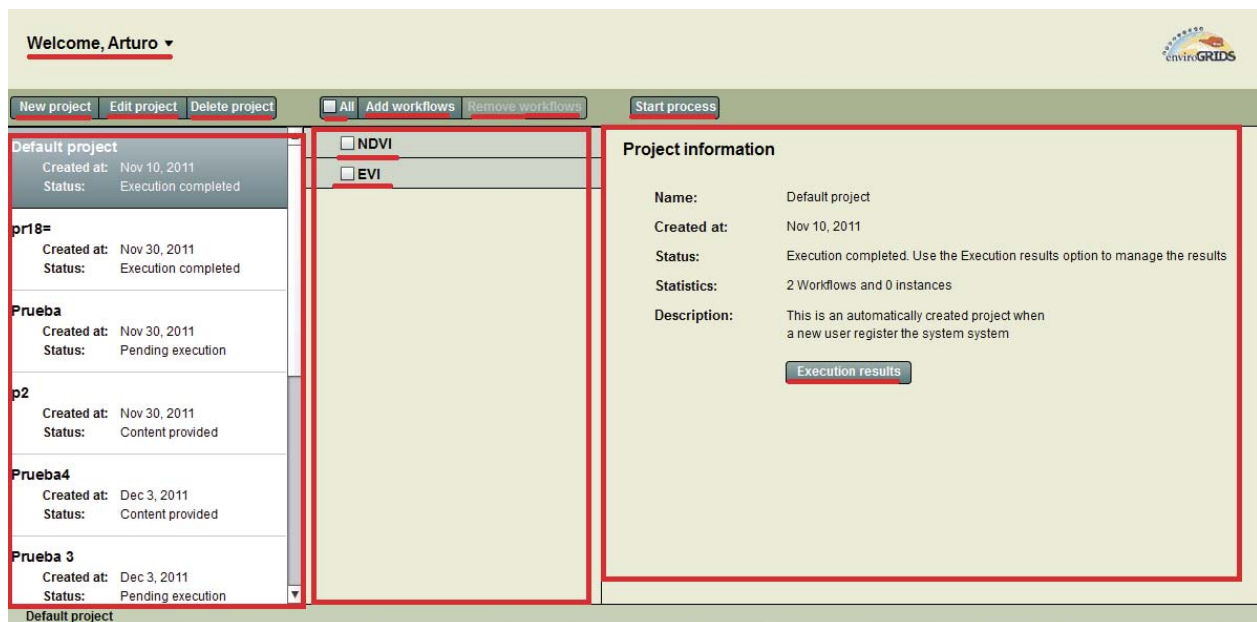


Figura 4.28: Símbolos estáticos referentes a la estructura del layout y títulos de funciones.

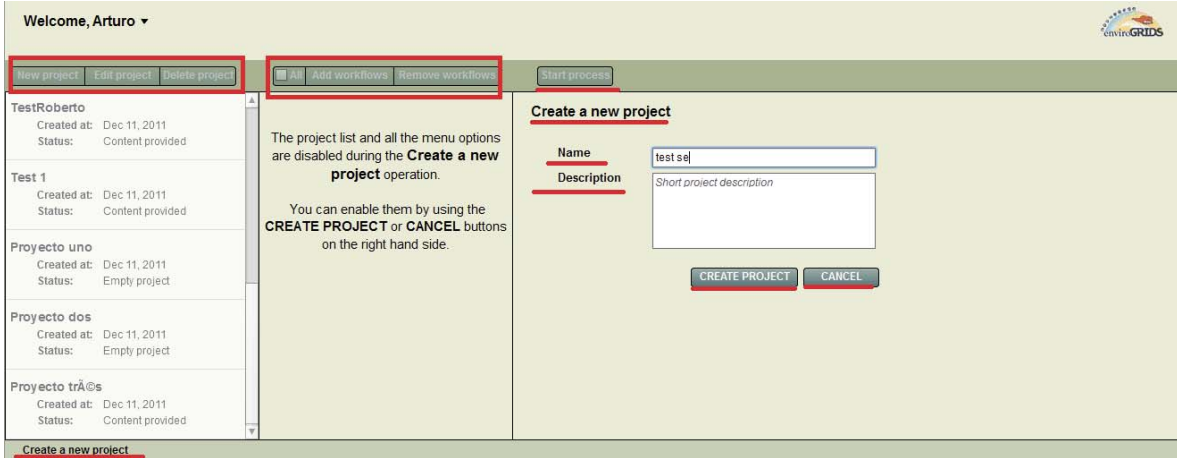


Figura 4.29: Símbolos estáticos referentes a títulos de funciones.

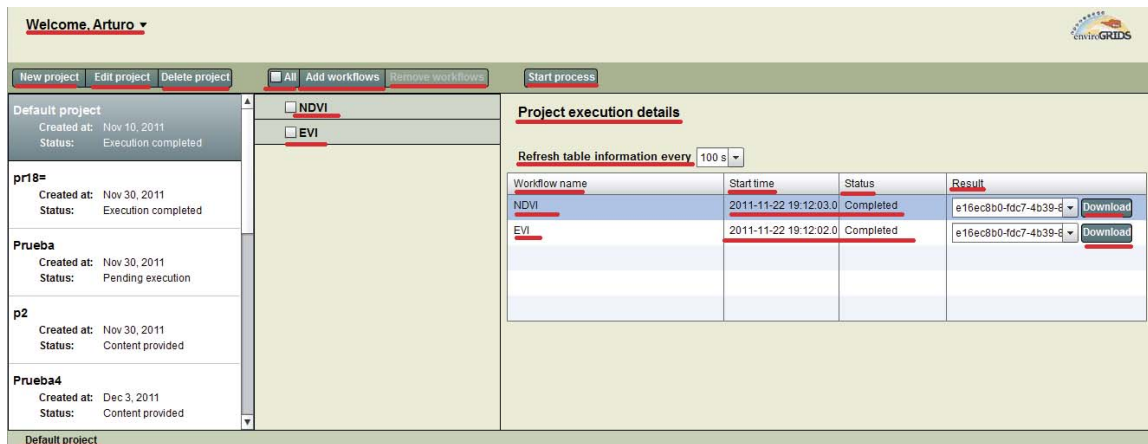


Figura 4.30: Símbolos estáticos desplegados al terminar la ejecución de un proceso.

Greenview v2, muestra una gran cantidad de símbolos estáticos, los que son su mayoría, con los cuales nombra las distintas funciones y controles de la interfaz.

Mensajes meta-comunicacionales

- **Aquí está mi entendimiento de quien eres:** Eres un usuario que podría ser novato, dado el orden de los elementos, pero en total conocimiento de tu área de estudio.

- **Que he aprendido que necesitas hacer, la forma que prefieres hacerlo y porqué:** He aprendido que necesitas realizar tareas haciendo uso de un orden bien establecido, manejar diversos parámetros de configuración para tus tareas. Prefieres hacerlo de esta manera pues tus tareas son secuenciales y necesitas generar una serie de resultados.
- **Éste es el sistema que yo he diseñado para ti, y esta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** He desarrollado un sistema ordenado que te permite realizar tus tareas en un orden bien estructurado y definido. No entiendo bien tu propósito.

Paso 3: Analizar los símbolos dinámicos y revisar o reconstruir en base a estos el mensaje de meta-comunicación según la plantilla.

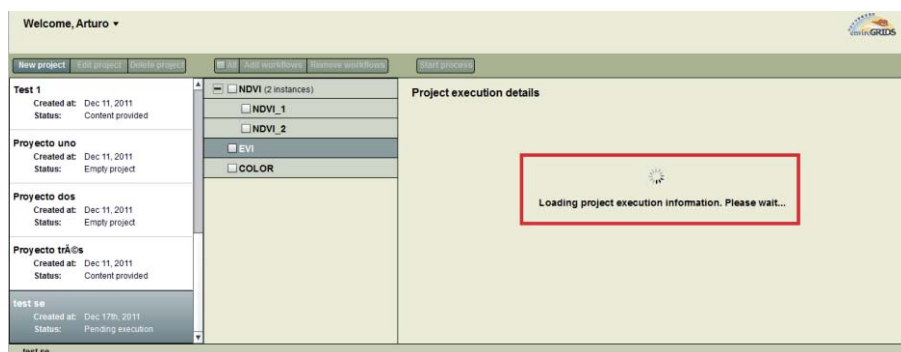


Figura 4.31: Símbolo dinámico mostrando un mensaje al ejecutar el proceso.

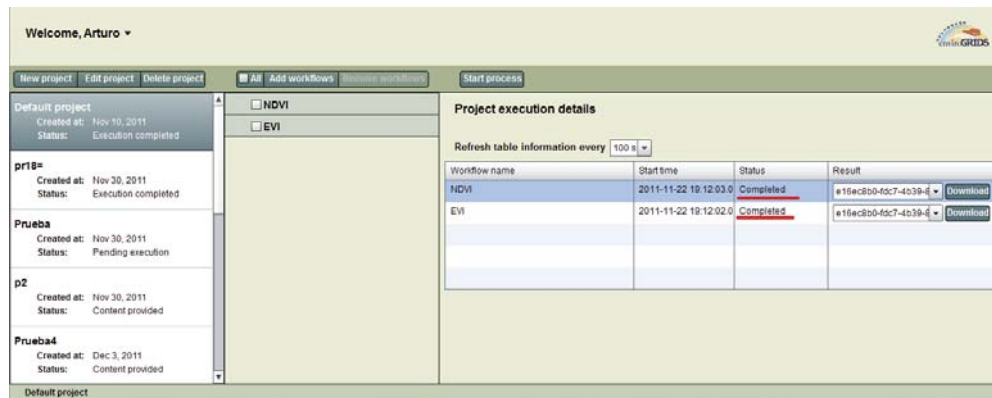


Figura 4.32: Mensajes de estado del sistema.

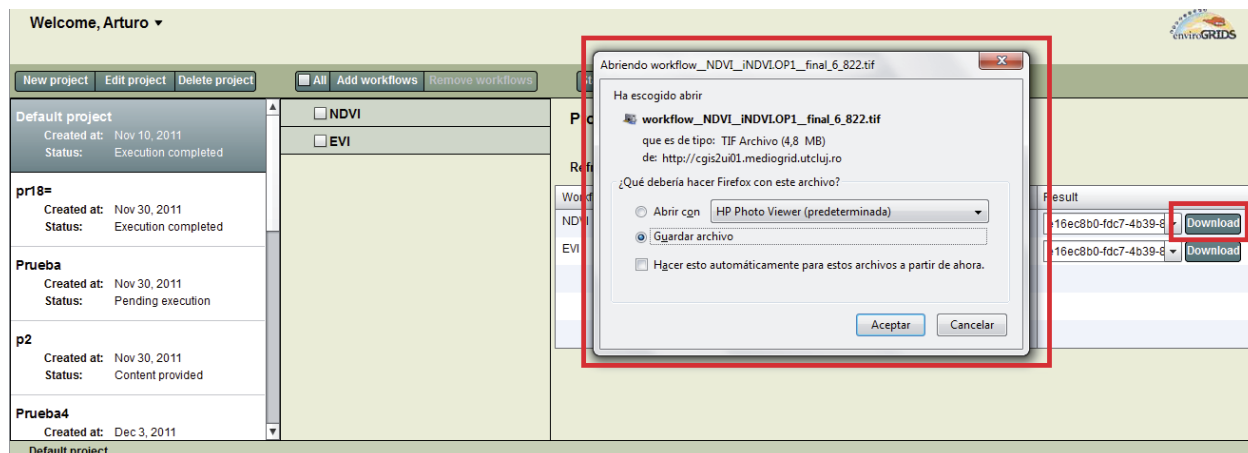


Figura 4.33: Ventana de pop-up para descarga de resultado.

En GreenLand v2 las señales dinámicas están casi ausentes, sólo se limitan a mostrar el estado de los procesos, alguna retroalimentación sobre lo que está haciendo el sistema y ventanas para descargar los resultados.

Mensajes meta-comunicacionales

- **Aquí está mi entendimiento de quien eres:** Eres un usuario avanzado, con tareas bastante específicas.
- **Que he aprendido que necesitas hacer, la forma que prefieres hacerlo y porqué:** He aprendido que necesitas información sobre los estados de las tareas y que necesitas acceso a los resultados de una manera fácil y expedita. Con estos símbolos dinámicos no me queda claro el por qué.
- **Éste es el sistema que yo he diseñado para ti, y esta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** He desarrollado un sistema ordenado que te permite realizar tus tareas y revisar su estado para luego obtener los resultados de manera sencilla.

Paso 4: Comparación del mensaje meta-comunicacional de los pasos 1, 2 y 3.

Las señales meta-comunicacionales más presentes en GreenLand v2 son las estáticas, seguidas de las dinámicas. Las señales meta-lingüísticas son las más escasas limitándose a algunas descripciones de elementos del software y a un manual externo.

Estas diferencias entre los diversos tipos de señales en Greenland v2, se debe a que el software cumple con una función bastante específica y delimitada, probablemente porque no sea un producto terminado o quizás se trate de un prototipo.

De los tres tipos de mensajes meta-comunicacionales, se observa 1 perfil de usuario, el cual, al parecer no es experto como usuario general, pero si es muy específico y experto en las tareas en que se desempeña, pues no requiere descripciones especialmente detalladas.

Las señales estáticas, se muestran bien ordenadas y parecen expresar un software bien estructurado el cual tiene un orden claro para realizar las tareas.

Paso 5: Evaluación final y global de la Comunicabilidad en el software

Mensajes meta-comunicacionales

- **Aquí está mi entendimiento de quien eres:** Eres un usuario que no es un experto informático, pero si eres experto dentro de tu área de trabajo, la geología, pues no necesitas descripciones mayores en las tareas que realizas.
- **Que he aprendido que necesitas hacer, la forma que prefieres hacerlo y porqué:** He aprendido que realizas tareas de análisis de datos geográficos. Tienes todas tus funciones ordenadas y agrupadas donde tu eliges que tareas añadir. Tus tareas son bastante acotadas a una zona geográfica en especial. Todo esto lo haces así porque tus tareas son secuenciales.
- **Éste es el sistema que yo he diseñado para ti, y esta es la manera en que puedes o debes usarlo de manera que cumplas el rango de propósito que caen en esta visión:** He diseñado un sistema que te permita ejecutar tus tareas de manera sencilla y rápida pues requieren de alto un nivel de procesamiento para obtener los resultados de manera expedita. Por esto debes utilizarlo manipulando sus controles y sus variables para los procesos secuenciales que ejecutas para finalmente obtener un resultado del sistema.

4.2.9 Evaluación de Comunicabilidad sobre el producto GreenLand v2

Ésta fue una prueba con usuarios tomada en laboratorio con el fin de detectar diversos quiebres comunicacionales entre el diseñador y el usuario para así, finalmente, elaborar un perfil semiótico. Para realizar esta

actividad fue necesario hacer uso de un grupo de sujetos de prueba, los cuales tienen los perfiles indicados en la Tabla 4.2.3.

Tabla 4.23: Perfiles de sujetos de prueba en evaluación de Comunicabilidad.

Sujeto de prueba	Ocupación	Experiencia previa/Conocimientos de computación distribuida	Experiencia previa/Conocimientos de Grid Computing
S1	Estudiante de ingeniería civil informática, trabajador a medio tiempo	Sin conocimientos	Ninguno
S2	Estudiante de ingeniería civil informática	Sin conocimientos	Ninguno
S3	Estudiante de magister en ingeniería informática	Conocimientos básicos	Ninguno
S4	Estudiante de magister en ingeniería informática	Conocimientos básicos	Ninguno

Quiebres Comunicacionales Encontrados en la Evaluación de GreenLand v2

A los sujetos de prueba se les entregó una serie de tareas las cuales han tenido que completar, durante su desarrollo ocurrieron una serie de quiebres comunicacionales de los 3 tipos: quiebres transitorios, parciales y totales.

Tabla 4.24: Quiebres comunicacionales ocurridos en la prueba con usuarios.

Sujeto de prueba	Tipo de Quiebre comunicacional	Observaciones
S1, S3	Q. Transitorio - ¿Dónde está?	Problema al tratar de añadir otra instancia de mismo workflow (o tarea) dentro de un proceso
S1, S2	Q. Transitorio - ¿Y ahora qué?	Después de añadir los workflows al proceso, el usuario se queda unos 30 segundos sin saber con qué seguir para ejecutar el proceso
S1, S2, S3, S4	Q. Transitorio - ¿Qué es esto?	Los valores o parámetros de los workflows parecen no ser entendidos por ningún sujeto de prueba
S3	Q. Transitorio - ¡No puedo hacerlo de esta manera!	Al tratar de ejecutar al video de presentación/tutorial, este no funciona por falta de los codecs en el equipo, por lo que el usuario busca otras maneras de

		reproducirlo
S2	Q. Transitorio - ¿Qué pasó?	Al presionar “Save” para guardar los cambios en los parámetros de un workflow, el usuario no recibe retroalimentación por lo que no sabe si pasa algo
S1, S3	Q. Total - ¡Me rindo!	Al tratar de añadir otras instancias de un mismo workflow al proceso el usuario no puede encontrar como

Interpretación de Quiebres Comunicacionales

En total se detectaron 12 quiebres comunicacionales y de 6 tipos, repartidos entre todos los sujetos de prueba, en lo que se puede observar:

Tabla 4.25: Cuantificación de quiebres comunicacionales en GreenLand v2 tras la prueba con usuarios.

Quiebres Transitorios	
Tipo	Cantidad
¿Dónde está?	2 (16,66%)
¿Y ahora qué?	2 (16,66%)
¿Qué es esto?	4 (33,33%)
¿Qué pasó?	1 (8,33%)
Quiebres Parciales	
Tipo	Cantidad
¡Puedo hacerlo de otra manera!	1 (8,33%)
Quiebres Totales	
Tipo	Cantidad
¡Me rindo!	2 (16,66%)
Total:	12

Tras estos datos aportados se puede observar principalmente que los usuarios tienen problemas relacionados con el entendimiento de cierto elemento de la interfaz, específicamente con los valores o parámetros de los workflows (Tabla 4.24), lo cual puede verse reflejado en la Tabla 4.25 donde se muestra una mayoría de quiebres comunicacionales del tipo “¿Qué es esto?”. También se ha encontrado un problema con una función que al parecer está algo oculta para los usuarios. Además puede notarse otro problema en donde el sistema no entrega una retroalimentación apropiada.

Si bien la aplicación es bastante acotada, dio algunos problemas para los sujetos de pruebas, como se muestra anteriormente, donde principalmente existe un elemento que es de difícil entendimiento (los parámetros de los workflow, Tabla 4.24) para los usuarios por lo que el diseñador, en este caso, ha fallado al tratar de entregar su visión del software.

Definición del perfil semiótico

Basándose en las conclusiones realizadas en el análisis de los quiebres comunicacionales, se realiza un perfil semiótico para el producto GreenLand v2.

- **¿Quiénes son los usuarios de mi producto?** Los usuarios de mi producto son expertos del área de la geología, que conocen bien toda la terminología del negocio. No son expertos informáticos, pero si son especialistas en el área de la geología.
- **¿Qué he aprendido de las necesidades (deseos) de mis usuarios?** He aprendido que mis usuarios requieren de pasos muy específicos y acotados para realizar sus tareas. El usuario requiere de una aplicación sencilla de usar (para ellos) y ordenada.
- **¿Qué prefieren los usuarios y por qué?** Los usuarios requieren de pasos secuenciales para realizar sus tareas, pues de esta forma evita o minimiza los errores que puedan surgir.
- **¿Qué sistema he diseñado y cómo podrían/debieran utilizarlo?** He diseñado un sistema para usuarios expertos, que conocen toda la terminología y no necesitan mucha ayuda. Este sistema permite realizar tareas de análisis para imágenes satelitales de una región bien específica.
- **¿Cuál es mi visión de diseño?** En mi diseño tengo elementos agrupados y términos que son entendibles por el usuario experto del área de la geología. He diseñado una serie de controles para que puedas interactuar con el software de manera ordenada y secuencial para así evitar posibles errores.

4.3 Análisis de Impacto de la Comunicabilidad Sobre la Usabilidad

En las pruebas piloto desarrolladas en la sección 3.2, existe una relación entre la Comunicabilidad y la Usabilidad, ahora con la ejecución de más experimentos se quiere llegar a conclusiones más acabadas al respecto [16].

4.3.1 La Relación Entre Quiebres Comunicacionales y Heurísticas de Usabilidad

Se puede establecer que ciertos quiebres comunicacionales [6] pueden tener relación con heurísticas y problemas de Usabilidad, en este caso las correspondientes a aplicaciones del tipo *Grid Computing* (4.1.3), como se puede observar en la Tabla 4.26.

Tabla 4.26: Relaciones entre quiebres comunicacionales y heurísticas de Usabilidad específicas.

Quiebres Transitorios	
Quiebre comunicacional	Heurísticas de Usabilidad asociadas
Q1. ¿Dónde está?	H4 (Diseño estético) H8 (Reducir carga de memoria) H9 (Explorabilidad)
Q2. ¿Y ahora qué?	H3 (Simplicidad) H9 (Explorabilidad)
Q3. ¿Qué es esto?	H1 (Claridad) H2 (Uso de metáforas) H4 (Diseño estético) H6 (Consistencia)
Q4. ¡Oops!	H10 (Control sobre acciones) H11 (Prevención de errores) H12 (Ayuda para Recuperación de Errores)
Q5. ¿Dónde estoy?	H9 (Explorabilidad)
Q6. ¡No puedo hacerlo de esta manera!	H9 (Explorabilidad)
Q7. ¿Por qué no?	H1 (Claridad) H12 (Ayuda para Recuperación de Errores)
Q8. ¿Qué pasó?	H1 (Claridad) H5 (Retroalimentación)
Q9. ¡Auxilio!	H13 (Documentación de Ayuda)
Quiebres Parciales	
Quiebre comunicacional	Heurísticas de Usabilidad asociadas
Q10. ¡Puedo hacerlo de otra manera!	H1 (Claridad) H2 (Uso de metáforas) H3 (Simplicidad) H7 (Atajos para Funcionalidades) H9 (Explorabilidad)
Q11. Gracias, pero no	H9 (Explorabilidad)
Quiebres Totales	
Quiebre comunicacional	Heurísticas de Usabilidad asociadas
Q12. ¡Me parece que está bien!	H1 (Claridad) H11 (Prevención de errores)
Q13. ¡Me rindo!	H3 (Simplicidad)

Utilizando otro enfoque, en la Tabla 4.27 se muestra la asociación de las heurísticas específicas para aplicaciones del tipo *Grid Computing* y las heurísticas genéricas de Nielsen con los diversos quiebres comunicacionales.

Tabla 4.27: Relaciones entre heurísticas de Usabilidad y quiebres comunicacionales.

Heurísticas de Nielsen		Heurísticas Grid Computing		Quiebres Comunicacionales	
ID	Definición	ID	Definición	ID	Definición
N2	Concordancia entre el sistema y el mundo real	H1	Claridad	Q3 Q7 Q8 Q10 Q12	¿Qué es esto? ¿Por qué no? ¿Qué pasó? ¿Puedo hacerlo de otra manera! ¿Me parece que está bien!
		H2	Uso de Metáforas	Q3 Q10	¿Qué es esto? ¿Puedo hacerlo de otra manera!
N8	Estética y diseño minimalista	H3	Simplicidad	Q2 Q10 Q13	¿Y ahora qué? ¿Puedo hacerlo de otra manera! ¿Me rindo!
		H4	Diseño Estético	Q1 Q3	¿Dónde está? ¿Qué es esto?
N1	Visibilidad del estado del sistema	H5	Retroalimentación	Q8	¿Qué pasó?
N4	Consistencia y estándares	H6	Consistencia	Q3	¿Qué es esto?
N7	Flexibilidad y eficiencia de uso	H7	Atajos para Funcionalidades	Q10	¿Puedo hacerlo de otra manera!
N6	Reconocer mejor que recordar	H8	Reducir Carga de Memoria	Q1	¿Dónde está?
N3	Control y libertad del usuario	H9	Explorabilidad	Q1 Q2 Q5 Q6	¿Dónde está? ¿Y ahora qué? ¿Dónde estoy? ¿No puedo hacerlo de esta manera!
		H10	Control sobre Acciones	Q4	¿Oops!
N5	Prevención de errores	H11	Prevención de Errores	Q4 Q12	¿Oops! ¿Me parece que está bien!
N9	Ayuda al usuario a reconocer, diagnosticar y recuperarse de errores	H12	Ayuda para Recuperación de Errores	Q4 Q7	¿Oops! ¿Por qué no?
N10	Ayuda y documentación	H13	Documentación de Ayuda	Q9	¿Auxilio!

4.3.2 La relación Comunicabilidad - Usabilidad en el producto GreenView

Como se puede observar en la Tabla 4.8, la gran mayoría de los quiebres comunicacionales detectados durante la prueba (4.2.1) fueron transitorios del tipo “¿Dónde está?” (27,77%) y del tipo “¿Qué es esto?” (27,77%), los cuales fueron relacionados a problemas de búsqueda y uso de la documentación y ayuda, junto con problemas asociados al uso y el entendimiento del software. Estos problemas pueden estar asociados a las heurísticas específicas para aplicaciones del tipo *Grid Computing* H13 (Documentación de ayuda), H1 (Claridad) y H9 (Explorabilidad) como se puede observar en la Tabla 4.26.

En un trabajo anterior [1], como se explica en 2.2.4, se efectuó una evaluación heurística sobre el mismo producto, en donde la mayor cantidad de problemas detectados, están asociados con la heurísticas H1 (Claridad), H11 (Prevención de errores), H12 (Ayuda para recuperación de errores) y H13 (Documentación de ayuda). En la Tabla 4.28, se muestran los eventuales quiebres comunicacionales asociados a los problemas de Usabilidad encontrados en el trabajo anterior [1] citado para el producto GreenView.

Tabla 4.28: Posibles quiebres comunicacionales asociados a problemas de Usabilidad en el producto GreenView.

ID	Problema de Usabilidad	Criticidad	Posibles quiebres comunicacionales asociados
9	No hay una secuencia explícita para que el usuario realice una tarea, no hay guía procedimientos	6	¿Dónde está? ¿Y ahora qué? ¡Puedo hacerlo de otra manera!
6	Mensajes de error poco explicativos	5,75	¡Oops!
16	Mensajes de error no ofrecen solución	5,75	¡Oops!
17	Ayuda contextual insuficiente	5,75	¡Auxilio!
1	Manual demasiado confuso	5,25	¿Qué es esto?
10	No hay atajos	5	¡Puedo hacerlo de otra manera!
2	Aplicación sólo en ingles	4,75	¿Qué es esto?
15	No se indica el formato que deben tener los archivos de GPP Computing	4,75	¡Me parece que está bien!
3	No hay ayuda	4,25	¿Dónde está? ¡Auxilio!
4	No hay botones deshacer o cancelar	4,25	¿Dónde está? ¡Oops!
11	Faltan metáforas en GPP Computing	4,25	¿Qué es esto?
5	No se sabe si el sistema está procesando	4	¿Qué pasó?
8	Sistema permite subir imágenes no aceptadas	4	¡Me parece que está bien!
14	No se especifican o ejemplifican valores válidos para “Coarse image resolution”	4	¡Me parece que está bien!
7	No se entiende objetivo de GPP	3,75	¿Qué es esto?

13	El reloj que aparece durante el procesamiento no es animado, dando la apariencia de que el sistema se ha pegado	3,5	¿Qué es esto? ¿Qué pasó?
12	La barra de progreso no indica gráficamente el progreso	3	¿Qué pasó?

Análisis de Posibles Quiebres Comunicacionales

En total se pueden observar 21 posibles quiebres comunicacionales y de 8 tipos, asociados a problemas de Usabilidad, los cuales se muestran a en la Tabla 4.29.

Tabla 4.29: Cuantificación de posibles quiebres comunicacionales en GreenView asociados a problemas de Usabilidad.

Quiebres Transitorios	
Tipo	Cantidad
¿Dónde está?	3 (14,28%)
¿Y ahora qué?	1 (4,76%)
¿Qué es esto?	5 (23,8%)
¡Oops!	3 (14,28%)
¿Qué pasó?	2 (9,52%)
¡Auxilio!	2 (9,52%)
Quiebres Parciales	
Tipo	Cantidad
¡Puedo hacerlo de otra manera!	2 (9,52%)
Quiebres Totales	
Tipo	Cantidad
¡Me parece que está bien!	3 (14,28%)
Total:	21

Según lo que se expone en la Tabla 4.29, se puede observar que la mayoría de los quiebres comunicacionales asociados son del tipo “¿Qué es esto?” (23,8%), seguidos con los relacionados con los tipos “¿Dónde está?” (14,28%), ¡Oops! (14,28%) y “¡Me parece que está bien!” (14,28%). Como se puede dilucidar, según estos datos, esto se puede relacionar directamente con los resultados de la evaluación de Comunicabilidad realizada sobre el producto GreenView (4.2.1), como se muestra en la Tabla 4.7.

4.3.3 La Relación Comunicabilidad - Usabilidad en el Producto Boinc

Acerca de los problemas de Usabilidad presentados por el producto Boinc, se puede observar en la Tabla 4.11 que los problemas principalmente se encuentran asociados a las heurísticas específicas para aplicaciones del tipo Grid Computing H6 (Consistencia), H1 (Claridad), H11(Prevención de errores) y H4 (Diseño estético).

En los mensajes meta-comunicacionales (4.2.5), se puede observar claramente que el software está diseñado para 2 tipos de usuarios, en donde las diferencias entre ellos son abismantes, incluso solamente uno de ellos tiene un software “usable”, el otro no. El cliente tiene un software con interfaz gráfica y menús de opciones, el servidor solamente tiene una consola de comandos, lo cual transgrede claramente las heurísticas H1 (Claridad), H4 (Diseño estético) y H6 (Consistencia). El servidor claramente debería tener una interfaz del tipo front-end, con un instalador sencillo (como lo tiene el cliente), quizás del tipo rpm o deb.

Utilizando otra perspectiva, pueden analizarse los problemas de Usabilidad encontrados en la evaluación heurística (usando heurísticas específicas para aplicaciones del tipo *Grid Computing*) realizada sobre el producto Boinc (4.2.2), en relación a posibles quiebres comunicacionales que podrían asociarse. Esto puede observarse en la Tabla 6.30.

Tabla 4.30: Posibles quiebres comunicacionales asociados a problemas de Usabilidad en el producto Boinc.

ID	Problema de Usabilidad	Criticidad	Posibles quiebres comunicacionales asociados
27	Archivos de configuración no disponibles	7	¿Dónde está?
17	Al presionar la ayuda esta solo muestra un icono	6,333	¿Qué es esto? ¿Qué pasó? ¡Auxilio!
20	No existe Control sobre acciones	5,667	¿Dónde está? ¡No puedo hacerlo de esta manera!
19	Mensajes poco explicativos	5,333	¿Qué es esto?
2	Comenzar una tarea que esta lista para empezar. (Vista avanzada)	5	¿Qué pasó? ¿Y ahora qué?
16	Ayuda en diferente idioma	5	¿Qué es esto?
25	No valida campos	5	¡Me parece que está bien!
5	Link de ayuda mal configurado. (Vista avanzada)	4,667	¿Qué pasó? ¿Dónde estoy?
7	Hay acciones que no producen retroalimentación. (Vista avanzada)	4,667	¿Qué pasó?
13	Diferentes idiomas en una misma pantalla	4,667	¿Qué es esto?
1	Metáfora no representativa de proyecto. (Vista avanzada)	4,333	¿Qué es esto?

12	Interfaz de usuario sobrecargada. (Vista simple)	4,333	¿Dónde está?
14	No existen atajos para todas las funcionalidades	4,333	¡No puedo hacerlo de esta manera!
15	No existe retroalimentación de lo que se está ejecutando	4,333	¿Qué pasó?
18	Idiomas distintos en secciones de control	4,333	¿Qué es esto?
3	Descripción larga y difícil de leer. (Vista avanzada)	4	¿Qué es esto?
11	Error al mostrar opciones de configuración. (Vista simple)	4	¿Qué pasó?
26	Eje Y promedio de uso negativo	4	¿Qué es esto?
4	Superposición de elementos. (Vista avanzada)	3,667	¿Qué es esto?
6	No existe la opción de guardar el log de eventos. (Vista avanzada)	3,667	¿Dónde está?
21	Sección de noticias aparece cuando se presiona en mensajes	3,667	¿Qué es esto? ¿Dónde estoy?
22	En pantalla básica, no existe atajos para funcionalidades	3,667	¡No puedo hacerlo de esta manera!
24	No prevee el error simple	3,333	¿Qué pasó? ¡Me parece que está bien!
28	Adaptación inadecuada de layout en distintos idiomas	3,333	¿Dónde está? ¿Dónde estoy?
8	Gráfico con colores similares. (Vista avanzada)	3	¿Qué es esto?
10	Gráfico con fecha errónea. (Vista avanzada)	2,667	¡Me parece que está bien!
23	Al pausar la aplicación el mensaje es inconsistente con el botón a presionar	2,333	¿Qué es esto? ¡Me parece que está bien!
9	Eje sin rotular en gráfico de estadísticas. (Vista avanzada)	2	¿Qué es esto?

Análisis de Posibles Quiebres Comunicacionales

En total se pueden observar 35 posibles quiebres comunicacionales y de 8 tipos, asociados a problemas de Usabilidad, los cuales se muestran a en la Tabla 4.31.

Tabla 4.31: Cuantificación de posibles quiebres comunicacionales en el producto Boinc asociados a problemas de Usabilidad.

Quiebres Transitorios	
Tipo	Cantidad
¿Dónde está?	5 (14,28%)
¿Y ahora qué?	1 (2,85%)
¿Qué es esto?	13 (37,14%)
¿Dónde estoy?	2 (5,71%)

¿Qué pasó?	6 (17,14%)
¡Auxilio!	1 (2,85%)
¡No puedo hacerlo de esta manera	3 (8,57%)
Quiebres Totales	
Tipo	Cantidad
¡Me parece que está bien!	4 (11,42%)
Total:	35

Según estos datos, se puede observar que el producto Boinc, podría presentar quiebres comunicacionales asociados principalmente a los del tipo “¿Qué es esto?” (37,14%), “¿Qué pasó?” (17,14%) y “¿Dónde está?” (14,28%). De este modo, en torno a la Comunicabilidad, el producto puede presentar problemas donde el usuario no entienda ciertos controles o funciones del producto de software Boinc o las indicaciones que se muestran en los diversos mensajes que el producto despliega.

4.3.4 La Relación Comunicabilidad - Usabilidad en el Producto GreenLand v2

El producto GreenLand v2 presenta una mayor cantidad de problemas de Usabilidad concentrados en las heurísticas H1 (Claridad), H5 (Retroalimentación) y H6 (Consistencia), según lo mostrado en la Tabla 4.17.

Una prueba con usuarios encontró que estos no comprendían el propósito del software, además de algo de dificultad en el uso de ciertos controles (4.2.6).

Una inspección semiótica sobre el producto GreenLand v2 (4.2.7) arrojó que éste está bien armado y acotado para ciertos usuarios expertos del área de la geología, que es bien específico para realizar tareas de análisis secuenciales.

Una prueba de Comunicabilidad en laboratorio (4.2.8) con usuarios arrojó una serie de quiebres comunicacionales principalmente del tipo “¿Qué es esto?”, como se muestra en la Tabla 4.25, donde los usuarios no entendían los conceptos entregados por el diseñador del software GreenLand v2.

Dados estos antecedentes, se puede observar claramente que existen problemas asociados a las heurísticas H1 (Claridad) y H5 (Retroalimentación) con las inferencias extraídas de los experimentos de Comunicabilidad. De esta manera, lo altamente específico de los usuarios hacia los que está enfocado el software GreenLand v2 hace que este sea poco entendible para otros lo cual puede generar problemas de Usabilidad asociados a las heurísticas anteriormente descritas. También puede añadirse la relación con la heurística H9 (Explorabilidad) al encontrarse en la prueba de Comunicabilidad con una gran cantidad de problemas del tipo “¿Qué es esto?”.

Utilizando otra perspectiva donde se analizan los problemas de Usabilidad encontrados en la evaluación heurística en relación con posibles quiebres comunicacionales asociados, se muestra en la Tabla 4.32.

Tabla 4.32: Posibles quiebres comunicacionales asociados a problemas de Usabilidad en el producto GreenLand v2.

ID	Problema de Usabilidad	Criticidad	Posibles quiebres comunicacionales asociados
1	Ayuda no disponible	7	¿Dónde está? ¡Auxilio!
13	Resultados poco claros	6,667	¿Qué es esto? ¿Qué pasó?
8	Información provista incompleta	5,667	¿Qué es esto?
19	No se ejecutan todos los procesos	5,667	¿Qué pasó?
26	Tareas enviadas y no procesadas no se pueden cancelar	5,667	¡No puedo hacerlo de esta manera! ¡Me rindo!
2	Recuperar contraseña no disponible	5,333	¿Dónde está? ¡Me rindo!
4	Aplicación sólo en un idioma	5,333	¿Qué es esto?
9	Aplicación no se adapta a la resolución	5,333	¿Dónde está? ¿Dónde estoy?
20	No hay respuesta del sistema	5,333	¿Qué pasó? ¿Y ahora qué? ¡Me rindo!
21	Ejecución de proyectos sin carga de archivo	5,333	¿Dónde está? ¿Qué pasó?
22	No es posible configurar el sistema	5,333	¿Dónde está?
12	No hay posibilidad de volver al estado anterior	5	¿Dónde está? ¡No puedo hacerlo de esta manera! ¡Me rindo!
14	No deja claro si el botón está activo o no	5	¿Qué es esto? ¿Qué pasó?
6	Mensaje de error "ofensivo"	4,667	¿Qué es esto?
10	Elementos sin descripción	4,667	¿Qué es esto?
11	Valores sin descripción o con nombres poco significativos	4,667	¿Qué es esto?
24	Mensaje confuso y error de sistema	4,667	¿Qué es esto? ¿Qué pasó?
15	Retroalimentación poco clara	4,333	¿Qué es esto? ¿Qué pasó?
16	No posee atajos	4,333	¡No puedo hacerlo de esta manera!
3	Enlace de contacto me obliga a instalar un cliente	4	¿Y ahora qué?
5	Mensaje de error que no es claro	4	¿Qué es esto? ¡Oops!
23	No se actualiza la descripción	4	¿Y ahora qué?

			¿Qué es esto?
7	Campo de contacto no da toda la información	3,667	¿Qué es esto?
18	Falta descripción	3,333	¿Qué es esto?
17	Logo poco claro	3	¿Qué es esto?
25	Caracteres no soportados	2,333	¿Qué es esto?

Interpretación de Quiebres Comunicacionales

En total se pueden observar 38 posibles quiebres comunicacionales y de 9 tipos, asociados a problemas de Usabilidad, los cuales se muestran en la Tabla 4.33.

Tabla 4.33: Cuantificación de posibles quiebres comunicacionales en el producto GreenLand v2 asociados a problemas de Usabilidad.

Quiebres Transitorios	
Tipo	Cantidad
¿Dónde está?	6 (15,78%)
¿Y ahora qué?	3 (7,89%)
¿Qué es esto?	15 (39,47%)
¡Oops!	1 (2,63%)
¿Dónde estoy?	1 (2,63%)
¿Qué pasó?	6 (15,78%)
¡Auxilio!	1 (2,63%)
¡No puedo hacerlo de esta manera	2 (5,26%)
Quiebres Totales	
Tipo	Cantidad
¡Me rindo!	3 (6,25%)
Total:	38

Según lo expuesto en la Tabla 4.33, puede observarse que los problemas, tienen principal relación con el quiebre comunicacional “¿Qué es esto?” (39,47%), el quiebre “¿Dónde está?” (15,78%) y el quiebre “¿Qué pasó?” (15,78%), lo cual tiene directa relación con los resultados de la prueba de Comunicabilidad realizada sobre el producto GreenLand v2. Si se comparan estos datos con los entregados en la prueba de Comunicabilidad de la Tabla 4.25, se observa una directa relación con las cantidades de quiebres comunicacionales del tipo “¿Qué es esto?”, los cuales fueron los más importantes según la prueba (33,33%).

4.3.5 Resumen de Heurísticas Asociadas a Problemas de Comunicabilidad en los Productos Analizados

En la Tabla 4.34 se muestra un resumen con las heurísticas asociadas encontradas relacionadas a los diversos análisis de Comunicabilidad y quiebres comunicacionales más destacados asociados a los problemas de Usabilidad encontrados en los productos de software analizados.

Tabla 4.34: Resumen de heurísticas de Usabilidad asociadas a problemas de Comunicabilidad más importantes.

Producto	Heurísticas asociadas a problemas de Comunicabilidad
GreenLand v1 (5.2.3)	H1 (Claridad) H13 (Documentación de ayuda)
GreenView	H1 (Claridad) H5 (Retroalimentación) H9 (Explorabilidad) H13 (Documentación de ayuda)
Boinc	H1 (Claridad) H4 (Diseño estético) H6 (Consistencia)
GreenLand v2	H1 (Claridad) H5 (Retroalimentación) H9 (Explorabilidad) H13 (Documentación de ayuda)

De esta manera, según las metodologías con que se cuenta para realizar los diversos experimentos, se encontraron heurísticas relacionadas directamente con la Comunicabilidad todas estas relacionadas con la entrega de mensajes entre el diseñador y el usuario.

Cabe destacar la posible asociación de quiebres comunicacionales a los diversos problemas de Usabilidad encontrados en los experimentos realizados sobre los productos de software del tipo *Grid Computing* analizados en este trabajo (4.3.1). De esta manera, se puede apreciar una relación directa con los quiebres comunicacionales en donde se pueden entender con mayor profundidad los problemas relacionados con el entendimiento del producto de software por parte del usuario en relación con posibles problemas de Usabilidad.

5. Conclusiones

Ciertamente *Grid Computing* es la tecnología candidata del futuro para hacerse cargo de problemas que requieren una alta capacidad de procesamiento, pues es altamente escalable y de costos relativamente bajos. Gracias a la tecnología de Grid Computing, se van a poder abordar los problemas del futuro, principalmente en lo relacionado con el análisis de grandes volúmenes de información. Los sistemas de información actuales generan una gran cantidad de datos y a lo largo del tiempo, de manera exponencial, incluso sistemas que no tienen relación directa con sistemas comerciales absolutos, como por ejemplo las redes sociales. Estas redes han tenido un crecimiento exponencial, generando inmensas cantidades de información, la cual no se está aprovechando completamente, qué pasará con esta información en 10 años más, qué sistema va a tener la capacidad de analizar tan inmensa cantidad de datos. De esta manera la tecnología *Grid Computing* aparece como la solución a esta problemática y de otras que irán apareciendo en el tiempo.

Como se observa tras los diversos experimentos de Usabilidad realizados en un trabajo anterior, los productos analizados presentan una serie de problemas de Usabilidad, lo cual dificulta el acercamiento del software a usuarios no expertos. La obra mencionada es un trabajo de titulación perteneciente a alumnos de la Pontificia Universidad Católica de Valparaíso, los cuales han sido miembros del grupo de investigación de Usabilidad “UseCV”, en la misma universidad anteriormente nombrada [1].

Uno de los resultados importantes del trabajo previamente nombrado, es que el uso de heurísticas específicas para una aplicación de *Grid Computing*, es bastante mejor que hacer uso de heurísticas genéricas para buscar problemas de Usabilidad. Se muestra que hay una gran cantidad de problemas que no son identificados por medio de las heurísticas genéricas y que además están rankeadas con un alto puntaje en severidad para estos problemas. También se muestra una pequeña cantidad de problemas encontrados con las heurísticas genéricas, pero estas tienen unos índices de severidad bastante bajos. De esta manera se sustenta lo dicho, acerca de que es mejor utilizar un grupo de heurísticas específicas para aplicaciones de *Grid Computing*. Esta experiencia podría repetirse con otros dominios de aplicaciones.

Al comparar los experimentos de Usabilidad con los experimentos de Comunicabilidad, se observa en ambos, problemas relacionados con la heurística relacionada con Ayuda y Documentación, lo cual se ve reflejado en la gran cantidad de problemas encontrados relacionados a esta heurística. De esta forma, en esta primera instancia se puede comenzar a observar la existencia de una relación entre Usabilidad y Comunicabilidad, llegando a conclusiones tan importantes como por ejemplo de que sería bueno realizar una inspección semiótica antes de una evaluación heurística o que ambos experimentos pueden ser complementarios.

Se han efectuado nuevas pruebas, en donde se puede inferir de manera más certera la existencia de una relación y/o impacto de la Comunicabilidad sobre la Usabilidad. De igual modo los experimentos de Comunicabilidad han servido para tener una visión global de un software y los problemas de Usabilidad que puedan presentarse en él para así tener especial circunspección al encontrarse con ellos en un experimento de Usabilidad. Los experimentos han mostrado especial relación entre heurísticas dentro del contexto de la comprensión y/o entendimiento de las interfaces.

En lo que se refiere a los objetivos de esta investigación, se observa una mejora en la metodología de evaluación basada en heurísticas específicas de *Grid Computing*, proveniente de un trabajo previo. Además, con el nuevo set de experimentos de Usabilidad y Comunicabilidad se puede observar de mejor forma la existencia de una relación entre la Usabilidad y la Comunicabilidad, aspectos que son objetivos específicos de esta investigación.

Tras el trabajo práctico hecho a través de los diversos experimentos de Usabilidad realizados sobre los productos de software analizados, se pudo dilucidar que las evaluaciones heurísticas pueden tener asociada cierta subjetividad dada por los evaluadores, pues cada uno puede hacer uso de un proceso cognitivo distinto a la hora de buscar problemas de Usabilidad. De esta manera se sugiere que la metodología para realizar evaluaciones heurísticas pueda formalizarse en un grado mayor, estableciendo alguna guía para buscar problemas de Usabilidad. Esta guía, podría acompañarse de una lista de errores comunes para cada heurística lo que permitiría ayudar al evaluador otorgando una orientación a la hora de buscar problemas de Usabilidad. Al mismo tiempo, se sugiere armar y capacitar equipos de evaluadores con miembros pertenecientes a otras áreas del conocimiento, que incluso podrían no ser informáticos para así complementar procesos cognitivos distintos.

En lo que se refiere al trabajo con los experimentos de Comunicabilidad, en lo relacionado a las pruebas con usuarios, estas están sujetas a la subjetividad, pues la metodología carece de un elemento que pueda cuantificar de alguna manera la criticidad de los quiebres comunicacionales encontrados. Lo único que se puede hacer es analizar la cantidad de los quiebres encontrados, pero no se puede observar el grado o la gravedad en el quiebre comunicacional. Esto puede ser tema para un trabajo futuro, al igual que lo planteado en torno a la metodología de evaluación heurística.

6. Bibliografía

1. Hayvar D., Tapia G., “*Usabilidad en Aplicaciones Grid Computing*”, Memoria de Título, Pontificia Universidad Católica de Valparaíso, 2010.
2. ISO/IEC, *ISO/IEC 9126, “Software Engineering Product Quality*”, 2001.
3. Rusu C., Roncagliolo S., Hayvar D., Tapia G., Rusu V., Gorgan D., “*Usability Heuristics for Grid Computing Applications*”, ACHI 2011, The Fourth International Conference on Advances in Computer-Human Interactions, 2011.
4. Roncagliolo S., Rusu V., Rusu, C., Hayvar D., Tapia G., Gorgan D., “*Grid Computing Heuristics in Practice*”, ITNG IEEE Computer Society, 2011.
5. Bote-Lorenzo, M., Dimitriadis, Y., Gomez-Sanchez, E., “*Grid Characteristics and Uses: a Grid Definition*”, University of Valladolid, España, 2004.
6. Sieckenius de Souza C., Faria Leitão C., “*Semiotic Engineering Methods for Scientific Research in HCI*”, 2009
7. Hernández R., Fernández C., “*Metodología de la Investigación*”, 5ta Edición, McGraw-Hill, 2010.
8. Nielsen J., “*Usability Engineering*”, Morgan Kaufmann Publishers, 1993.
9. Nielsen J., “*Ten Usability Heuristics*”, Disponible en www.useit.com/papers/heuristic/heuristic_list.html, [Revisado el 15/09/2012].
10. Foster I., Kesselman C., “*The Grid 2: Blueprint for a New Computing Infrastructure*”, 2nd Edition, 2004.
11. Foster I., “*What is the Grid? A Three Point Checklist*”, Argonne National Laboratory & University of Chicago, 2002.

12. Foster I., Kesselman C., "*The Anatomy of the Grid: Enabling Scalable Virtual Organizations*", International J. Supercomputer Applications, vol. 15(3), 2001
13. Abbas A., "*Grid Computing: A Practical Guide to Technology and Applications*", Charles River Media, 2004
14. Norman, M., "*Types of grid users and the Customer-Service Provider relationship: a future picture of grid use*", University of Oxford, 2006.
15. Technical University of Cluj-Napoca Romania, "*(CGIS) Computer Graphics and Interactive Systems Laboratory*", Disponible en <http://cgis.utcluj.ro/applications/greenland> [Revisado el 15/09/2012].
16. Rusu C., Roncagliolo S., Figueroa A., Rusu V., Gorgan D., "*Evaluating the Usability and the Communicability of Grid Computing Applications*", ACHI 2012, The Fifth International Conference on Advances in Computer-Human Interactions.

7. Anexos

Anexo A: Acuerdo de Confidencialidad

Acuerdo de confidencialidad

YO _____ ACEPTO a participar en una prueba de Usabilidad supervisada por _____, el día ___/___/____, en el *Laboratorio de Usabilidad de la Escuela de Ingeniería Informática de la Pontificia Universidad Católica de Valparaíso*. Entiendo y estoy de acuerdo con las condiciones mencionadas en adelante.

Entiendo que el experimento se hace sólo para evaluar un sistema de software, NO mis capacidades/habilidades/conocimientos.

Entiendo que los resultados del experimento se utilizarán sólo para propósitos académicos y/o de investigación, sin que mi identidad sea revelada.

Entiendo que puedo comunicar al supervisor del experimento, en cualquier momento, mi malestar, molestia o inconformidad. Entiendo que puedo abandonar el experimento y el laboratorio en cualquier momento.

Firma

Anexo B: Cuestionario de Prueba de Usabilidad producto GreenView

(1) Cuestionario pre-test

Conteste el siguiente cuestionario.

I. DATOS PERSONALES

1. Sexo Femenino Masculino
2. Edad _____
3. Nivel más alto de educación completado o en proceso.
Enseñanza media Completa En Proceso
Técnico Completa En Proceso
Universitario Completa En Proceso
Otro ¿Cuál? _____
4. ¿Cuál es su ocupación? _____

II. INFORMACIÓN DE EXPERIENCIA PREVIA

5. De acuerdo a sus conocimientos en la computación distribuida. ¿En qué categoría se identifica?
 Sin conocimientos
 Conocimientos básicos
 Conocimientos avanzados

6. De acuerdo a sus conocimientos en la tecnología Grid Computing. ¿En qué categoría se identifica?
 Sin conocimientos
 Conocimientos básicos
 Conocimientos avanzados

6. ¿Usted ha utilizado previamente alguna aplicación de computación distribuida o de Grid Computing?
 Sí, cuál(es): _____
 No

(2) Lista de tareas

Considere el siguiente escenario:

Usted trabaja en una oficina de meteorología y tiene que generar imágenes satelitales para entregarlas a un investigador experto para que las analice y las interprete.

Tarea 1

1. Abra el navegador web y coloque la dirección `cgis.utcluj.ro:4331/interpolation3.1/`
2. Ingrese con los datos: user: cristian.rusu password: test

Tarea 2

1. Busque en la ayuda la descripción de las principales funcionalidades del software

Tarea 3

1. Navegue libremente por la aplicación y revise sus distintas opciones

Tarea 4

1. Seleccione una pequeña área del mapa del sur de Italia utilizando el mouse
2. Seleccione un intervalo entre noviembre y diciembre de 1990
3. Ejecute el proceso utilizando el nombre “proceso de prueba” (sin comillas) y como descripción “prueba de Comunicabilidad” (sin comillas)

Tarea 5

1. Busque información sobre los procesos que se están ejecutando
 2. Busque los procesos terminados
 3. Escriba el nombre de un nodo donde se ejecutó un proceso terminado
-

(3) Cuestionario post-test

Encierre en un círculo la nota más apropiada para cada una de las siguientes preguntas.

1. ¿Pudo completar las tareas?

1	2	3	4	5
Muy Difícilmente	Difícilmente	Neutral	Fácilmente	Muy Fácilmente

2. ¿Considera que la información disponible en la ayuda es completa (suficiente)?

1	2	3	4	5
Muy en desacuerdo	En desacuerdo	Neutral	De acuerdo	Completamente de acuerdo

4. ¿Considera que la información encontrada en la ayuda ha sido útil?

1	2	3	4	5
Inútil	Poco útil	Neutral	Útil	Muy útil

3. ¿Considera que el software es fácil de utilizar?

1	2	3	4	5
Muy difícil	Difícil	Neutral	Fácil	Muy fácil

4. ¿Considera que la ha sido fácil entender el propósito del software?

1	2	3	4	5
Muy difícil	Difícil	Neutral	Fácil	Muy fácil

5. ¿Qué fue lo que más le llamó la atención o le agradó del software?

6. ¿Qué fue lo que más le disgustó del software?

7. ¿Qué cosas de la interfaz mejoraría en el software?

Anexo C: Planilla de Evaluación de Comunicabilidad

Evaluación de Comunicabilidad

Software/producto: _____

Fecha: ___/___/___

Usuario: _____

Evaluador: _____

1) Categorización de Quiebres comunicacionales

Categorización: Quiebres transitorios				
Tipo de quiebre (tag)	Característica distintiva	Ejemplo de ocurrencia	Cantidad	Observaciones
1) ¿Dónde está?	El usuario sabe lo que quiere hacer pero no encuentra el elemento de interfaz apropiado	El usuario revisa elementos, buscando el adecuado		
2) ¿Y ahora qué?	El usuario no sabe que hay que hacer a continuación	El usuario está buscando de manera aleatoria o secuencial indicaciones sobre la continuación de la secuencia de interacción		
3) ¿Qué es esto?	El usuario no entiende un elemento específico de interfaz	El usuario se posiciona sobre algún elemento, esperando encontrar explicaciones		
4) ¡Oops!	El usuario comete un error e	El usuario inmediatamente		

	inmediatamente se da cuenta de su equivocación	intenta volver un paso atrás, con la función “deshacer” o cualquier operación que le permita anular los efectos de su error		
5) ¿Dónde estoy?	El usuario está tomando acciones equivocadas, que serían apropiadas en otro contexto	El usuario trata ejecutar operaciones o búsquedas de símbolos correctos, pero en un contexto equivocado		
6) ¿No puedo hacerlo de esta manera!	El usuario abandona un camino de interacción porque repentinamente piensa que dicho camino no es el correcto	El usuario interrumpe una actividad que estaba realizando y toma una dirección completamente diferente		
7) ¿Por qué no?	El usuario repite los pasos de una previa y no exitosa comunicación para descubrir qué estuvo mal	El usuario trata mejorar su entendimiento a través de la repetición de los mismos pasos, buscando potenciales errores		

8) ¿Qué pasó?	El usuario no entiende la respuesta del sistema a su acción	El usuario repite la acción porque la respuesta no existe o se entiende mal		
9) ¡Auxilio!	El usuario explícitamente solicita asistencia	El usuario deliberadamente llama una función determinada (por ejemplo presionando la tecla F1)		
Categorización: Quebres parciales				
Tipo de quiebre (tag)	Característica distintiva	Ejemplo de ocurrencia	Cantidad	Observaciones
10) ¡Puedo hacerlo de otra manera!	El usuario no está enterado totalmente de los caminos de acción que ofrece el sistema para llevar a cabo alguna determinada tarea. <u>El usuario NO entiende el diseño</u>	El usuario alcanza su objetivo por un camino no óptimo, como efecto secundario de una acción que tiene, en realidad, otro propósito		
11) Gracias, pero no.	El usuario entiende el camino que el sistema ofrece, pero elige un	El usuario elige conscientemente expresarse de una manera distinta a la propuesta		

	camino alternativo. <u>El usuario entiende el diseño</u>			
<u>Categorización: Quiebres totales</u>				
Tipo de quiebre (tag)	Característica distintiva	Ejemplo de ocurrencia	Cantidad	Observaciones
12) ¡Me parece que está bien!	El usuario está convencido de que ha alcanzado su meta, pero de hecho eso no ha sucedido. <u>El usuario NO es consiente del quiebre</u>	El usuario termina la interacción, pero el sistema está en un estado que no es consistente con la intención y meta del usuario		
13) ¡Me rindo!	El usuario explícitamente expresa su inhabilidad para alcanzar la meta. <u>El usuario es consiente del quiebre</u>	El usuario termina la interacción sin haber alcanzado el éxito		

Totales

Cantidad de quiebres transitorios:

Cantidad de quiebres parciales:

Cantidad de quiebres totales:

Observaciones

Anexo D: Plantilla completa de problemas y notas en evaluación heurística del cliente Boinc haciendo uso de las heurísticas específicas para aplicaciones del tipo Grid Computing

Id	Problema	Eval 1			Eval 2			Eval 3			Promedios			Desviación Estándar		
		S	F	C	S	F	C	S	F	C	S	F	C	S	F	C
1	Metáfora no representativa de proyecto. (Vista avanzada)	4	2	6	1	1	2	3	2	5	2,667	1,667	4,333	1,528	0,577	2,082
2	Comenzar una tarea que esta lista para empezar. (Vista avanzada)	4	2	6	2	2	4	3	2	5	3	2	5	1	0	1
3	Descripción larga y difícil de leer. (Vista avanzada)	2	2	4	2	2	4	2	2	4	2	2	4	0	0	0
4	Superposición de elementos. (Vista avanzada)	1	3	4	1	2	3	2	2	4	1,333	2,333	3,667	0,577	0,577	0,577
5	Link de ayuda mal configurado. (Vista avanzada)	2	3	5	2	2	4	2	3	5	2	2,667	4,667	0	0,577	0,577
6	No existe la opción de guardar el log de eventos. (Vista avanzada)	1	2	3	2	2	4	2	2	4	1,667	2	3,667	0,577	0	0,577
7	Hay acciones que no producen retroalimentación. (Vista avanzada)	3	2	5	2	2	4	2	3	5	2,333	2,333	4,667	0,577	0,577	0,577
8	Gráfico con colores similares. (Vista avanzada)	2	1	3	1	2	3	1	2	3	1,333	1,667	3	0,577	0,577	0
9	Eje sin rotular en gráfico de estadísticas. (Vista avanzada)	1	1	2	1	1	2	1	1	2	1	1	2	0	0	0

10	Gráfico con fecha errónea. (Vista avanzada)	2	1	3	1	1	2	2	1	3	1,667	1	2,667	0,577	0	0,577
11	Error al mostrar opciones de configuración. (Vista simple)	2	2	4	2	2	4	2	2	4	2	2	4	0	0	0
12	Interfaz de usuario sobre cargada. (Vista simple)	2	2	4	1	3	4	2	3	5	1,667	2,667	4,333	0,577	0,577	0,577
13	Diferentes idiomas en una misma pantalla	3	2	5	2	2	4	3	2	5	2,667	2	4,667	0,577	0	0,577
14	No existen atajos para todas las funcionalidades	4	2	6	2	1	3	2	2	4	2,667	1,667	4,333	1,155	0,577	1,528
15	No existe retroalimentación de lo que se está ejecutando	3	2	5	2	1	3	3	2	5	2,667	1,667	4,333	0,577	0,577	1,155
16	Ayuda en diferente idioma	4	2	6	2	2	4	3	2	5	3	2	5	1	0	1
17	Al presionar la ayuda esta solo muestra un icono	4	4	8	3	2	5	3	3	6	3,333	3	6,333	0,577	1	1,528
18	Idiomas distintos en secciones de control	2	3	5	2	2	4	2	2	4	2	2,333	4,333	0	0,577	0,577
19	Mensajes poco explicativos	3	3	6	2	3	5	3	2	5	2,667	2,667	5,333	0,577	0,577	0,577
20	No existe Control sobre acciones	3	3	6	2	3	5	3	3	6	2,667	3	5,667	0,577	0	0,577
21	Sección de noticias aparece cuando se presiona mensajes	1	2	3	2	2	4	2	2	4	1,667	2	3,667	0,577	0	0,577
22	En pantalla básica, no existe atajos para funcionalidades	2	2	4	2	1	3	2	2	4	2	1,667	3,667	0	0,577	0,577

23	Al pausar la aplicación el mensaje es inconsistente con el botón a presionar	1	1	2	2	1	3	1	1	2	1,333	1	2,333	0,577	0	0,577
24	No prevee el error simple	2	2	4	2	1	3	2	1	3	2	1,333	3,333	0	0,577	0,577
25	No valida campos	2	4	6	2	2	4	3	2	5	2,333	2,667	5	0,577	1,155	1
26	Eje Y promedio de uso negativo	2	2	4	2	2	4	2	2	4	2	2	4	0	0	0
27	Archivos de configuración no disponibles	4	4	8	3	2	5	4	4	8	3,667	3,333	7	0,577	1,155	1,732
28	Adaptación inadecuada de layout en distintos idiomas	2	2	4	1	2	3	1	2	3	1,333	2	3,333	0,577	0	0,577

Anexo E: Plantilla completa de problemas y notas en evaluación heurística de GreenLand v2 haciendo uso de las heurísticas específicas para aplicaciones del tipo Grid Computing

Id	Problema	Eval 1			Eval 2			Eval 3			Promedios			Desviación Estándar		
		S	F	C	S	F	C	S	F	C	S	F	C	S	F	C
1	Ayuda no disponible	4	4	8	3	3	6	3	4	7	3,333	3,667	7	0,577	0,577	1
2	Recuperar contraseña no disponible	3	3	6	3	2	5	3	2	5	3	2,333	5,333	0	0,577	0,577
3	Enlace de contacto me obliga a instalar un cliente	4	2	6	2	1	3	2	1	3	2,667	1,333	4	1,155	0,577	1,732
4	Aplicación sólo en un idioma	3	4	7	1	1	2	3	4	7	2,333	3	5,333	1,155	1,732	2,887
5	Mensaje de error que no es claro	2	2	4	2	2	4	2	2	4	2	2	4	0	0	0
6	Mensaje de error "ofensivo"	3	2	5	2	2	4	3	2	5	2,667	2	4,667	0,577	0	0,577
7	Campo de contacto no da toda la información	3	1	4	2	1	3	3	1	4	2,667	1	3,667	0,577	0	0,577
8	Información provista incompleta	4	2	6	3	2	5	4	2	6	3,667	2	5,667	0,577	0	0,577
9	Aplicación no se adapta a la resolución	4	3	7	2	1	3	3	3	6	3	2,333	5,333	1	1,155	2,082
10	Elementos sin descripción	3	3	6	2	2	4	2	2	4	2,333	2,333	4,667	0,577	0,577	1,155
11	Valores sin descripción o con nombres poco significativos	3	3	6	2	2	4	2	2	4	2,333	2,333	4,667	0,577	0,577	1,155
12	No hay posibilidad de volver al estado anterior	2	4	6	2	1	3	3	3	6	2,333	2,667	5	0,577	1,528	1,732

13	Resultados poco claros	4	4	8	3	3	6	3	3	6	3,333	3,333	6,667	0,577	0,577	1,155
14	No deja claro si el botón está activo o no	2	4	6	2	2	4	2	3	5	2	3	5	0	1	1
15	Retroalimentación poco clara	2	2	4	3	2	5	2	2	4	2,333	2	4,333	0,577	0	0,577
16	No posee atajos	1	4	5	2	1	3	1	4	5	1,333	3	4,333	0,577	1,732	1,155
17	Logo poco claro	1	4	5	1	1	2	1	1	2	1	2	3	0	1,732	1,732
18	Falta descripción	2	2	4	2	1	3	2	1	3	2	1,333	3,333	0	0,577	0,577
19	No se ejecutan todos los procesos	4	2	6	3	2	5	4	2	6	3,667	2	5,667	0,577	0	0,577
20	No hay respuesta del sistema	4	2	6	3	2	5	3	2	5	3,333	2	5,333	0,577	0	0,577
21	Ejecución de proyectos sin carga de archivo	4	2	6	3	2	5	3	2	5	3,333	2	5,333	0,577	0	0,577
22	No es posible configurar el sistema	3	3	6	3	2	5	3	2	5	3	2,333	5,333	0	0,577	0,577
23	No se actualiza la descripción	3	3	6	2	1	3	2	1	3	2,333	1,667	4	0,577	1,155	1,732
24	Mensaje confuso y error de sistema	3	3	6	2	2	4	2	2	4	2,333	2,333	4,667	0,577	0,577	1,155
25	Caracteres no soportados	2	1	3	1	1	2	1	1	2	1,333	1	2,333	0,577	0	0,577
26	Tareas enviadas y no procesadas no se pueden cancelar	4	4	8	4	1	3	4	2	6	4	2,333	5,667	0	1,528	2,517

Anexo F: Plantilla completa de problemas y notas en evaluación heurística del cliente Boinc haciendo uso de las heurísticas genéricas de Nielsen

Id	Problema	Eval 1			Eval 2			Eval 3			Promedios			Desviación Estándar		
		S	F	C	S	F	C	S	F	C	S	F	C	S	F	C
1	Solo existe una opción para cambio de skins	2	3	5	1	1	2	1	0	1	1,333	1,333	2,667	0,577	1,528	2,082
2	Inconsistencia de idioma	2	3	5	1	1	2	2	1	3	1,667	1,667	3,333	0,577	1,155	1,528
3	No se especifica claramente que representan las pestañas superiores	3	3	6	3	2	5	2	1	3	2,667	2	4,667	0,577	1	1,528
4	Las opciones de proyectos no mantienen un formato fijo	3	3	6	1	1	2	1	1	2	1,667	1,667	3,333	1,155	1,155	2,309
5	Es poco claro el link de "Graphics Available"	2	2	4	1	1	2	1	0	1	1,333	1	2,333	0,577	1	1,528
6	Información dentro de "Propiedades del proyecto" es poco clara	3	2	5	3	1	4	1	0	1	2,333	1	3,333	1,155	1	2,082
7	No existe feedback correspondiente al comando "Actualizar"	4	3	7	2	1	3	1	1	2	2,333	1,667	4	1,528	1,155	2,646
8	El nombre de las actividades no genera información útil	3	4	7	2	2	4	1	0	1	2	2	4	1	2	3
9	Existe reincidencia de opciones para cambio de vistas	2	2	4	1	3	4	1	0	1	1,333	1,667	3	0,577	1,528	1,732
10	Distintas opciones para un mismo objetivo: Manager de cuentas	2	3	5	3	2	5	2	1	3	2,333	2	4,333	0,577	1	1,155

11	La pestaña de "transferencias" no posee información descriptiva	3	3	6	1	1	2	2	0	2	2	1,333	3,333	1	1,528	2,309
12	Los avisos se encuentran en una sola pestaña, de todos los proyectos.	3	3	6	2	2	4	1	1	2	2	2	4	1	1	2
13	El aviso redirecciona a un sitio externo a la aplicación.	3	3	6	2	2	4	3	1	4	2,667	2	4,667	0,577	1	1,155
14	Los formatos de horario no son suficientemente claros, en inactividad por día.	2	2	4	1	1	2	1	1	2	1,333	1,333	2,667	0,577	0,577	1,155
15	La casilla "No verificar archivos de imagen" no posee explicación	3	3	6	2	1	3	2	0	2	2,333	1,333	3,667	0,577	1,528	2,082
16	Mucha información se encuentra fuera del manager y redirecciona a una página web	3	3	6	2	1	3	3	1	4	2,667	1,667	4,333	0,577	1,155	1,528
17	Existe inconsistencia en opciones de uso del espacio en disco	3	3	6	2	1	3	2	1	3	2,333	1,667	4	0,577	1,155	1,732
18	No se encuentra el botón "Reanudar"	3	3	6	2	1	3	2	1	3	2,333	1,667	4	0,577	1,155	1,732
19	Se presenta una opción y aparece otra cosa	3	3	6	1	1	2	3	1	4	2,333	1,667	4	1,155	1,155	2
20	Botones pequeños	1	3	4	1	1	2	1	1	2	1	1,667	2,667	0	1,155	1,155
21	Mensajes inconsistentes	3	2	5	1	1	2	2	1	3	2	1,333	3,333	1	0,577	1,528

22	confusión al pausar proyecto	4	3	7	1	3	4	1	2	3	2	2,667	4,667	1,732	0,577	2,082
23	Acción clic equivocada	2	2	4	1	2	3	1	0	1	1,333	1,333	2,667	0,577	1,155	1,528

Anexo G: Plantilla completa de problemas y notas en evaluación heurística de GreenLand v2 haciendo uso de las heurísticas genéricas de Nielsen

Id	Problema	Eval 1			Eval 2			Eval 3			Promedios			Desviación Estándar		
		S	F	C	S	F	C	S	F	C	S	F	C	S	F	C
1	Inicialización lenta de la aplicación sin retroalimentación adecuada	4	4	8	3	3	6	3	3	6	3,333	3,333	6,667	0,577	0,577	1,155
2	Permite agregar workflows, aun no estando cargada la lista de proyectos.	3	3	6	2	1	3	3	3	6	2,667	2,333	5	0,577	1,155	1,732
3	El mensaje de error es poco claro, ni otorga solución	3	2	5	1	1	2	2	2	4	2	1,667	3,667	1	0,577	1,528
4	El nombre del botón es incomprensible	2	2	4	1	1	2	2	1	3	1,667	1,333	3	0,577	0,577	1
5	Imágenes perdidas, botones sin nombre	4	2	6	2	1	3	3	1	4	3	1,333	4,333	1	0,577	1,528
6	La restricción de archivos al subir al servidor, es poco clara	3	3	6	2	1	3	4	2	6	3	2	5	1	1	1,732
7	El cuadro de "perfil" se descuadra con la información	2	2	4	1	1	1	1	1	2	1,333	1,333	2,333	0,577	0,577	1,528
8	No permite el libre scroll si se tiene seleccionado un proyecto	2	2	4	2	1	3	1	1	2	1,667	1,333	3	0,577	0,577	1
9	El ícono del mouse no cambia al pasar por los distintos botones	1	2	3	1	1	2	1	1	2	1	1,333	2,333	0	0,577	0,577
10	Los inputs del workflow no posee explicación ni diferencias entre	3	3	6	1	1	2	4	2	6	2,667	2	4,667	1,528	1	2,309

	ellos																
11	La tabla de Project Execution Result se encuentra cortada por la columna	2	2	4	2	1	3	1	1	2	1,667	1,333	3	0,577	0,577	1	
12	El nombre del "Result" no describe nada	3	2	5	2	2	1	3	1	4	2,667	1,667	3,333	0,577	0,577	2,082	
13	No se genera una lista de errores, o faltantes para hacer correr el proyecto	4	4	8	2	1	3	4	2	6	3,333	2,333	5,667	1,155	1,528	2,517	
14	elementos de la interfaz no se cargan	4	4	8	1	1	2	1	2	3	2	2,333	4,333	1,732	1,528	3,215	
15	botones no ejecutan ninguna acción	2	3	5	1	1	2	1	1	2	1,333	1,667	3	0,577	1,155	1,732	
16	eliminación no funciona	3	3	6	2	1	3	2	2	4	2,333	2	4,333	0,577	1	1,528	
17	aplicación no responde	3	2	5	3	1	3	2	1	3	2,667	1,333	3,667	0,577	0,577	1,155	
18	Botones parecen no funcionar	3	3	6	2	1	2	3	1	3	2,667	1,667	3,667	0,577	1,155	2,082	
19	Problemas al iniciar proyecto	3	3	6	2	1	3	2	1	3	2,333	1,667	4	0,577	1,155	1,732	

Anexo H: Plantilla de Prueba de Usabilidad sobre el producto GreenLand v2

Prueba de Usabilidad

Software/producto: _____

Fecha: ___/___/___

Usuario: _____

Evaluador: _____

Tarea	Descripción	Criterios de éxito	Tiempo máximo	Cumpl. de tarea (éxito o fracaso)	Observaciones
1	1. Vaya a http://cgis.utcluj.ro/applications/greenland 2. Busque el video de presentación de “GreenLand v2” 3. Reproduzca el video	Logra reproducir el video	3 min		
2	1. Busque el manual de usuario del software “GreenLand v2” 2. Escriba una descripción de lo que hace el software (muy resumido)	Encuentra la información y logra describir el producto	10 min		
3	1. Haga click en el link “Free use of the GreenLand v2 application” 2. Haga login con los siguientes datos: User: arturo Password: testpass	Realiza el login y navega por las opciones	6 min		

	3. Navegue y revise los controles y opciones del software libremente. No se tome más de 5-6 minutos				
4	<p>1. Cree un proyecto llamado “Prueba” y en la descripción coloque “prueba de Usabilidad”</p> <p>2. Añada 4 operaciones (workflow) de la lista</p> <p>3. Modifique los parámetros de entrada de los workflow (libremente)</p> <p>4. Elimine 1 workflow</p> <p>5. Ejecute el proceso</p>	Logra ejecutar un proceso			