

# **ESCUELA POLITÉCNICA NACIONAL**

**FACULTAD DE INGENIERÍA EN SISTEMAS  
INFORMÁTICOS Y DE COMPUTACIÓN**

**DISEÑO DE EXPERIENCIA DE USUARIO DE UN EDITOR  
GRÁFICO WEB PARA LA GESTIÓN Y PREPARACIÓN DE  
INFRAESTRUCTURA**

**ANÁLISIS HEURÍSTICO DEL EDITOR VISUAL ARGON PARA EL  
APROVISIONAMIENTO DE INFRAESTRUCTURA**

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO  
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN  
SOFTWARE**

**CRISTHIAN DAVID MUÑOZ AULESTIA**

**[cristhian.munoz@epn.edu.ec](mailto:cristhian.munoz@epn.edu.ec)**

**DIRECTOR: CARLOS EFRIAN IÑIGUEZ JARRIN**

**[carlos.iniguez@epn.edu.ec](mailto:carlos.iniguez@epn.edu.ec)**

**QUITO, ENERO 2023**

## **CERTIFICACIONES**

Yo, Cristhian David Muñoz Aulestia declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

---

**CRISTHIAN DAVID MUÑOZ AULESTIA**

Certifico que el presente trabajo de integración curricular fue desarrollado por Cristhian David Muñoz Aulestia, bajo mi supervisión.

---

**CARLOS EFRAIN IÑIGUEZ JARRIN**  
**DIRECTOR**

## **DECLARACIÓN DE AUTORÍA**

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

CRISTHIAN DAVID MUÑOZ AULESTIA

CARLOS EFRAIN IÑIGUEZ JARRIN

## **DEDICATORIA**

Quiero dedicar especialmente a una persona que particularmente se ha esmerado a pesar de los obstáculos y que gracias al apoyo de personas extraordinarias y su resiliencia ha logrado alcanzar lo trazado; esto me lo dedico a mí. Aunque pueda sonar egocéntrico, este documento representa el esfuerzo que día a día he estado dispuesto afrontar a fin de completar la carrera que he elegido.



## **AGRADECIMIENTO**

En primer lugar, agradezco a Jehová, aunque sé que no intervino directamente estoy seguro de que me ha dado las fuerzas y la resiliencia necesaria para continuar y terminar este objetivo. A mis padres, que a pesar de todos los problemas y multitud de situaciones han sido un apoyo constante tanto económico como emocional para lograr terminar esta etapa de mi vida. A mi novia Sam, que a pesar de las circunstancias hemos logrado esperar para ver cumplidas nuestras metas, siendo mi gran apoyo constante. A mis amigos, que han logrado comprender las circunstancias por las que he pasado, y queda claro que sin los ánimos y la gran motivación que me han dado no hubiera logrado alcanzar esta meta.

# ÍNDICE DE CONTENIDO

CERTIFICACIONES .....	I
DECLARACIÓN DE AUTORÍA .....	II
DEDICATORIA.....	III
AGRADECIMIENTO .....	IV
ÍNDICE DE CONTENIDO.....	V
ÍNDICE DE FIGURAS .....	VII
ÍNDICE DE TABLAS .....	VIII
RESUMEN .....	IX
ABSTRACT .....	X
1 INTRODUCCIÓN .....	1
1.1 Objetivo general.....	2
1.2 Objetivos específicos .....	2
1.3 Alcance .....	2
1.4 Marco teórico .....	3
1.4.1 Balanceo de Carga .....	3
1.4.2 Aprovisionamiento de Infraestructura.....	5
1.4.3 ARGON.....	7
1.4.4 Interfaz de Usuario (UI).....	8
1.4.4.1 Interfaz por línea de Comandos (CLI) .....	9
1.4.4.2 Interfaz Gráfica (GUI).....	9
1.4.5 Experiencia de Usuario (UX).....	10
1.4.6 Técnica Personas .....	11
1.4.7 Heurísticas y Análisis Heurístico de Usabilidad .....	12
1.4.8 Evaluación Heurística .....	13
2 METODOLOGÍA .....	15
2.1 Preparación .....	15
2.1.1 Objetivos de ARGON .....	16
2.1.2 Perfil Usuario .....	18
2.1.3 Definición del Caso de Uso y sus Tareas Claves .....	21
2.1.4 Elección de Heurísticas a utilizar.....	24
2.1.5 Elaboración de la Lista de Verificación.....	24
2.2 Revisión del Producto.....	25
2.2.1 Entendimiento del producto.....	25

2.2.2 Descubrimiento de problemas.....	26
2.2.2.1 Al Aplicar/Comprender las tareas del caso de uso.....	26
2.2.2.2 Al Interactuar/Explorar con la IU .....	26
2.3 Opinión de Expertos .....	28
2.3.1 Protocolo de la Evaluación.....	28
2.3.1 Ejecución de la Evaluación .....	33
2.4 Preparación del Reporte .....	35
2.4.1 Recopilación de problemas de usabilidad .....	38
2.4.2 Puntaje de severidad y soluciones a problemas.....	39
2.4.3 Elaboración del reporte de hallazgos .....	39
3 RESULTADOS, CONCLUSIONES Y RECOMENDACIONES.....	39
3.1 Resultados.....	39
3.2 Conclusiones .....	41
3.3 Recomendaciones .....	42
4 REFERENCIAS BIBLIOGRÁFICAS.....	43
5 ANEXOS .....	46
ANEXO I. Formulario para levantar Perfil Usuario.....	46
ANEXO II. Resultados de la encuesta para Perfil Persona.....	51
ANEXO III. Detalle del Caso de Uso (Documento de Recorrido Cognitivo).....	55
ANEXO IV. Lista de Verificación de las Heurísticas de Nielsen.....	61
ANEXO V. Evaluación Interna .....	68
ANEXO VI. Análisis Heurístico para ARGON (enlace) .....	74
ANEXO VII. Evaluaciones Externas (enlace).....	74
ANEXO VIII. Resumen autogenerado sobre las respuestas de la Lista de Verificación .....	75
ANEXO IX. Recopilación de problemas de usabilidad (enlace).....	75
ANEXO X. Asignación de Severidad a los problemas de usabilidad .....	76
ANEXO XI. Infografía del reporte de hallazgos (enlace opcional) .....	81

## ÍNDICE DE FIGURAS

<b>Figura 1.</b> Diagrama ilustrativo del funcionamiento de un Balanceador de Carga. ....	4
<b>Figura 2.</b> Editor Gráfico de ARGON en Eclipse Modeling Framework. Fuente: Sandobalin, Insfran y Abrahao [12]......	8
<b>Figura 3.</b> Metodología DCU basado en la ISO9241, indicando el estado donde se puede realizar la evaluación heurística. ....	11
<b>Figura 4.</b> Curvas de proporción; problemas de usabilidad encontrados y relación beneficios a costos por el número de evaluadores. Fuente: Nielsen [23]. ....	13
<b>Figura 5.</b> Interfaz de usuario del editor de ARGON. Fuente: Herramienta ARGON. ....	17
<b>Figura 6.</b> Plantilla Persona “Francisco Sarzosa”. Fuente: Elaboración propia. ....	21
<b>Figura 7.</b> Pantalla del editor de ARGON con el caso de uso implementado. Fuente: Elaboración propia. ....	23
<b>Figura 8.</b> Captura de Pantalla de la interacción simultánea con el Caso de Uso implementado y la Lista de Verificación. Fuente: Elaboración propia. ....	27
<b>Figura 9.</b> Captura de Pantalla de la intervención de AnyDesk para las máquinas virtuales. Fuente: Elaboración propia. ....	29
<b>Figura 10.</b> Captura de Pantalla de la manipulación esperada por los expertos. Fuente: Elaboración propia. ....	29
<b>Figura 11.</b> Captura de la sección “Introducción” del protocolo a seguir. Fuente: Elaboración propia. ....	30
<b>Figura 12.</b> Captura de la sección “Aplicativos/Documentos” del protocolo a seguir. Fuente: Elaboración propia. ....	31
<b>Figura 13.</b> Captura de la sección “Aplicativos/Documentos” del protocolo a seguir. Fuente: Elaboración propia. ....	32
<b>Figura 14.</b> Resumen de respuestas de la Evaluación Interna. Fuente: Elaboración propia. ....	36
<b>Figura 15.</b> Resumen de respuestas de la experta MsC. Mayra Carrión. Fuente: Elaboración propia. ....	37
<b>Figura 16.</b> Resumen de respuestas del experto PhD. Julian Galindo. Fuente: Elaboración propia. ....	37
<b>Figura 17.</b> Resumen de respuestas del experto Dr. Marco Santorum. Fuente: Elaboración propia. ....	38
<b>Figura 18.</b> Resumen del recopilado de respuestas de las Evaluaciones Heurísticas. Fuente: Elaboración propia. ....	40

## ÍNDICE DE TABLAS

<b>Tabla 1.</b> Heurísticas de Usabilidad definidas por Jakob Nielsen. Fuente: Nielsen [22].....	12
<b>Tabla 2.</b> Apartados para llevar a cabo la Evaluación Heurística. ....	14
<b>Tabla 3.</b> Preguntas de la encuesta para definir la plantilla Persona. ....	18
<b>Tabla 4.</b> Secuencia de Tareas que definen el Caso de Uso para el aprovisionamiento de un Balanceador de Carga. ....	22
<b>Tabla 5.</b> Problemas y Aspectos relevantes de la experta MsC. Mayra Carrión. ....	33
<b>Tabla 6.</b> Problemas y Aspectos relevantes del experto PhD. Julián Galindo. ....	34
<b>Tabla 7.</b> Problemas y Aspectos relevantes del experto Dr. Marco Santórum.....	35

## RESUMEN

Un análisis heurístico es un mecanismo útil para evaluar la usabilidad en prototipos o herramientas ya implementadas. Este tipo de análisis recaba y obtiene el criterio de expertos en el campo de usabilidad, cuyas sugerencias son de gran aporte para solventar los problemas de usabilidad encontrados. En los editores gráficos, lograr una buena usabilidad implica que la interfaz de usuario del editor sea sencilla, intuitiva y amigable a la mayor cantidad de usuarios que lo usan. En este trabajo se realiza un Análisis Heurístico para encontrar los problemas de usabilidad en el editor gráfico de ARGON. ARGON es una herramienta cuyo propósito es permitir al usuario modelar gráficamente una infraestructura para aprovisionarla en la nube. La evaluación se realiza siguiendo un caso de uso específico “Aprovisionamiento de un Balanceador de Carga en Amazon Web Services (AWS)” donde el experto evalúa las 10 heurísticas de Jakob Nielsen en la IU de ARGON. El experto identifica problemas de usabilidad al interactuar con la herramienta, siguiendo su secuencia de acciones/pasos del caso de uso. Los problemas encontrados fueron priorizados considerando como base un puntaje de severidad. La lista de problemas encontrados se convierte en un insumo relevante para un futuro re-diseño de la IU de la herramienta ARGON.

**PALABRAS CLAVE:** Análisis Heurístico, Interfaz de Usuario (IU), Caso de Uso, Balanceador de Carga.

## **ABSTRACT**

A heuristic analysis is a useful mechanism to evaluate the usability of prototypes or already implemented tools. This type of analysis collects and obtains the criteria of usability experts, whose suggestions greatly contribute to solving the usability problems found. In graphic editors, achieving good usability involves a simple, intuitive, and friendly user interface to the largest number of users who use it. In this work, a Heuristic Analysis is carried out to find usability problems in the graphic editor of ARGON. ARGON is a tool whose purpose is to allow the user to graphically model an infrastructure to provision it in the cloud. The Heuristic Analysis is carried out following a specific use case, where the experts evaluate Jakob Nielsen's 10 heuristics in the ARGON UI. The experts identify usability problems when interacting with the tool, following the sequence of actions/steps of the use case. The problems found were prioritized based on a severity score. The list of problems found becomes a relevant input for a future redesign of the ARGON UI.

**KEYWORDS:** Heuristic Analysis, User Interface (UI), Use Case, Load Balancer.

# 1 INTRODUCCIÓN

ARGON [1] es una herramienta stand-alone cuyo objetivo es facilitar el aprovisionamiento (gestión y preparación) de infraestructura. Es decir, a través de ARGON es posible modelar y crear/eliminar elementos de infraestructura tales como red, almacenamiento, sistemas operativos, entre otros, requeridos para un proyecto de TI.

El desarrollo de ARGON está basado en el enfoque DSL. Un DSL se define como un lenguaje específico que tiene como propósito solventar problemas particulares de un dominio [2], que en el caso de ARGON es el aprovisionamiento de infraestructura. A pesar de los beneficios y ventajas que presenta un DSL para afrontar un problema específico, su manipulación es compleja. Utilizar un DSL, generalmente implica interactuar con la línea de comandos y escribir sintaxis compleja propia del lenguaje, que previamente tiene que ser aprendida y dominada por el usuario. Esto ha generado que el uso de los DSL esté limitado principalmente a usuarios especializados y expertos en el dominio [2]. Con la finalidad de mejorar la experiencia de uso y asegurar la eficiencia y eficacia de manipulación, algunos DSL como SQL y CSound han desarrollado editores visuales como interfaces de usuario (MySQL Workbench [3] y Cabbage [4]) que facilitan su interacción.

Para evitar la manipulación de ARGON a través de la línea de comandos, como la mayoría de los DSL, ARGON implementa un editor visual como Interfaz de Usuario (UI). Sin embargo, los aspectos visuales, estéticos e interactivos de la UI no fueron el enfoque prioritario en el desarrollo de ARGON. En este sentido, para mejorar la interacción de la UI, es necesario identificar problemas de interacción en la UI actual, que permitan plantear su rediseño. Una de las técnicas para encontrar los problemas de usabilidad se fundamenta en el Análisis Heurístico.

El análisis heurístico forma parte del diseño de la Experiencia de Usuario, cuya importancia radica en mejorar la satisfacción del usuario con respecto a la usabilidad, es decir en cuán fácil de usar es la interfaz de usuario [5]. El análisis heurístico consiste en la evaluación de las interfaces de usuario por parte de expertos en el campo de la UX, quienes con apoyo de directrices y principios de diseño identifican potenciales problemas de usabilidad y aportan soluciones a dichos problemas [5].

El desarrollo de este componente consiste en realizar el Análisis Heurístico de la interfaz de usuario actual de ARGON con el fin de identificar problemas de interacción. [6]. Los



resultados obtenidos del análisis servirán como punto de entrada para la toma de decisiones respecto a los aspectos de usabilidad a ser mejorados en la interfaz. En efecto, los resultados obtenidos podrán guiar el posterior rediseño una UI como editor gráfico en la web para el DSL “ARGON”, lo que permitirá eliminar la barrera impuesta de su funcionamiento stand-alone.

## **1.1 Objetivo general**

Realizar una evaluación heurística de la interfaz gráfica de la herramienta de aprovisionamiento de infraestructura ARGON, para el caso de uso de balanceo de carga.

## **1.2 Objetivos específicos**

- Detectar problemas de usabilidad en la interfaz de usuario de la herramienta de aprovisionamiento de infraestructura ARGON, considerando el caso de uso: balanceo de carga.
- Proponer soluciones de diseño tomando en cuenta los principios de usabilidad de Jacob Nielsen.
- Establecer niveles de prioridad a los problemas de usabilidad encontrados como aporte al diseño/rediseño de la interfaz de usuario.

## **1.3 Alcance**

Enfocado en el análisis heurístico de la interfaz de usuario (editor visual) de ARGON, considerando la tarea específica: balanceo de carga. Este estudio seguirá los cuatro pasos metodológicos de un estudio heurístico: preparación, revisión del producto, opinión de expertos y generación del reporte de hallazgos.

- Preparación:
  - Identificar objetivos de la herramienta Argón
  - Esbozar perfil de usuario
  - Definir tareas claves
  - Decidir heurísticas a utilizar
  - Derivar listas de verificación o directrices para el dominio de aplicación
- Revisión del Producto
  - Entender el modelo de diseño del producto
  - Aplicar tareas y anotar problemas de usabilidad
  - Explorar pantallas secundarias y anotar problemas de usabilidad
- Opinión de Expertos

- Informe de 3 expertos sobre los objetivos de la UI, los usuarios y tareas clave
  - Solicitar a cada experto trabajar a lo largo de los escenarios de tareas
- Preparación de Reporte
  - Reunir problemas de usabilidad sin duplicados
  - Asignar un puntaje de severidad y proveer una solución a cada problema encontrado
  - Escribir reporte de hallazgos.

## **1.4 Marco teórico**

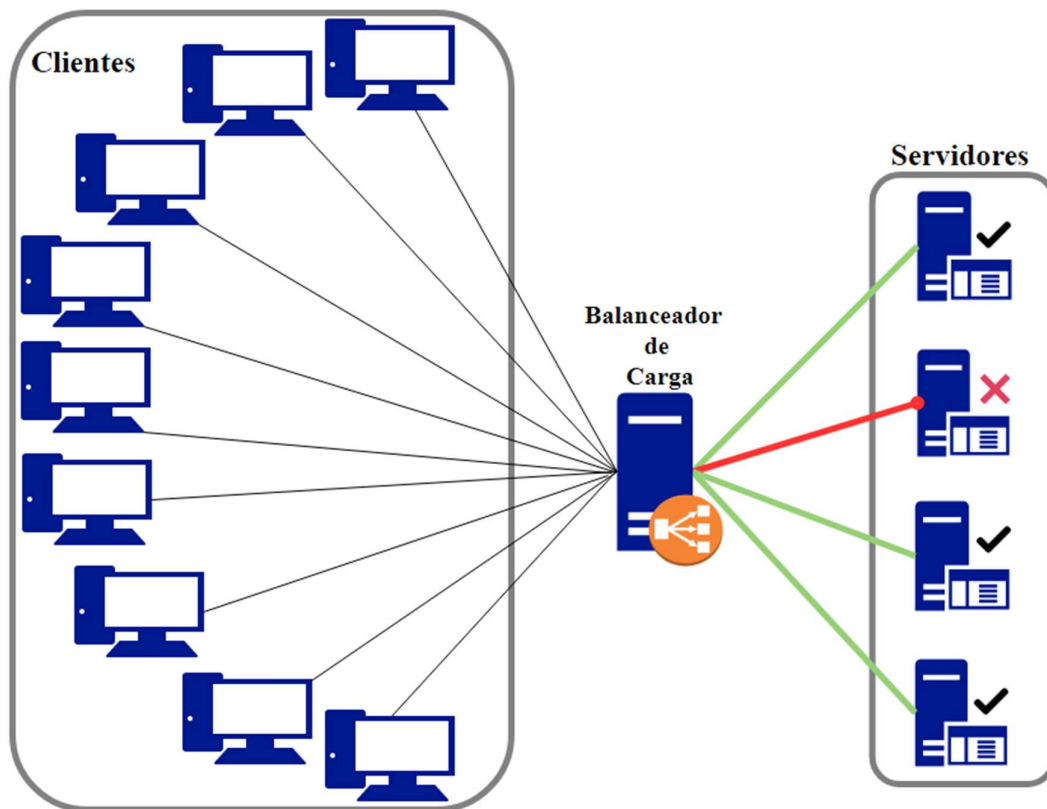
### **1.4.1 Balanceo de Carga**

Balanceo de Carga hace referencia a la distribución que se hace ante numerosas tareas, solicitudes y demás peticiones a cierto conjunto de recursos informáticos que realizarán este procesamiento. Con el fin de optimizar el procesamiento solicitado, un balanceador de carga logra evitar la sobrecarga desigual a los recursos de procesamiento [7].

En el contexto de la red, un balanceador de carga mejora la capacidad de respuesta de las aplicaciones evitando la sobre demanda de peticiones al servidor o granja de servidores; siendo la metodología más escalable para el manejo múltiple de peticiones. Esto se debe a que, un balanceador de carga se encuentra física o lógicamente entre la comunicación de los dispositivos clientes y el servidor, receptando y redirigiendo cada solicitud a un determinado servidor que se encuentre libre para procesar [8].

Los balanceadores de carga gestionan el flujo de peticiones o solicitudes evitando ciertos servidores que no logren llevar a cabo la tarea solicitada. Para esto, los balanceadores de carga toman en cuenta diversos factores como el estado de los servidores activo/inactivo, ubicación geográfica, número de conexiones activas, ancho de banda mínimo, conexión mínima, tráfico asignado, tiempo de respuesta mínimo, etc. Esto permite asegurar el rendimiento óptimo de los servicios, manteniendo en control entornos de TI complejos [8].

Los balanceadores están incorporados en los controladores de entrega de aplicaciones (ADC), y son capaces de aprovechar distintos algoritmos como Round Robin. Además, permiten que el tiempo de respuesta del servidor y el método de conexión sean los mínimos cumpliendo así las exigencias del mundo actual [8]. Además, su uso incrementa la disponibilidad de los servicios a los que da soporte, manteniendo la posibilidad de que estos sean escalable ante las necesidades que surjan a lo largo del tiempo.



**Figura 1.** Diagrama ilustrativo del funcionamiento de un Balanceador de Carga.

La figura 1 muestra el funcionamiento del balanceador de carga donde 9 clientes, dentro o fuera de la red, solicitan un mismo servicio que se dispone en 4 servidores. Estas solicitudes son procesadas por un único balanceador de carga que tiene en cuenta el estado de cada servidor, redirigiendo únicamente las peticiones a 3 servidores activos, de los 4 servidores existentes.

Existen al menos 2 tipos de balanceadores de carga: basado en Hardware y basado en Software. Los balanceadores basados en Hardware, se refiere a equipos físicos donde su rendimiento es de altas prestaciones logrando procesar cientos de gigas de tráfico para

diversas aplicaciones. Además, gracias a la virtualización se puede tener diversas instancias de balanceadores asignadas para una aplicación o varias aplicaciones en los servidores, esto permitiría mayor flexibilidad y aislamiento a más de una aplicación alojada en un mismo servidor [8]. Por otro lado, los basados en Software resultan ser más flexibles ya que permiten que su alojamiento pueda ser en un servidor que no tenga sobrecarga de procesamiento, ejecutándose en hipervisores compartidos. Esto permite mayor control en su configuración dependiendo del uso que se le haga, adaptándose a los requerimientos mínimos y necesarios; reduciendo espacio y costos [8]. Además, este tipo de balanceadores son predilectos en la nube, pues su configuración e instalación resulta viable al poder estar con otros servicios dentro de un solo equipo de altas prestaciones.

Este concepto ayudará a definir un determinado caso de uso, el cual guiará el objetivo para realizar un análisis específico a la herramienta de ARGON. A fin de concretar este concepto, el caso de uso se definirá en determinadas tareas a cumplir, logrando cubrir el objetivo del Balanceo de Carga para un modelo de Infraestructura de Tecnología de la Información.

### **1.4.2 Aprovisionamiento de Infraestructura**

Entender el dominio del aprovisionamiento de infraestructura involucra entender la Infraestructura de Tecnología de la Información (TI). La Infraestructura TI involucra todo lo concerniente a equipo físico (hardware), software, elementos e instalaciones de redes; que permitan su correcto funcionamiento. Esto con el fin de que los servicios informáticos alojados en estas estructuras puedan ser desarrollados, probados, entregados, controlados y hasta respaldados dentro o fuera de donde se aloje físicamente esta estructura [9].

Teniendo en cuenta el concepto de Infraestructura TI, su aprovisionamiento hará referencia a todo el proceso que involucra la instalación y configuración de esta estructura de elementos. Actualmente, las necesidades de infraestructura dependen en su mayoría del giro de la empresa, y la atención a dichas necesidades puede involucrar con frecuencia dos estrategias: tradicional y en la nube.

En la estrategia **tradicional**, la empresa o negocio provee todos los elementos necesarios para la formación de la Infraestructura tecnológica [10]. Estos elementos pueden verse en 3 categorías:

- **Hardware:** Computadores, servidores, clústeres, routers, etc.
- **Software:** Sistemas Operativos, Servicios web, sistemas de gestión de contenido (CMS), etc.
- **Redes:** Todo el cableado de red necesario para la conexión entre los distintos elementos, definiendo distintas redes locales como conexiones externas.

En la **nube**, los proveedores de servicios de infraestructura (p. ej., AWS, Azure) proveen accesos bajo demanda a distintos recursos informáticos que son altamente configurables, liberando y aprovisionando distintos recursos, eliminando el esfuerzo y costo de adquisición [10]. Esta estrategia busca principalmente reducir costos excesivos en la compra de servidores, routers y demás elementos informáticos, a fin de enfocar la importancia en el giro del negocio.

Existen modelos de prestación de servicios dentro de la nube, uno de ellos es la Infraestructura como servicio (Infrastructure-as-a-Service, IaaS). El IaaS es un modelo de entrega de infraestructura en la nube en la que los elementos de infraestructura (p. ej., servidores, redes) están disponibles de manera virtualizada. En el entorno virtualizado, los elementos de infraestructura deben seguir un proceso de instalación, así como el mantenimiento de los sistemas operativos necesarios, tales como el levantamiento de diversos protocolos para la instalación de la red [12].

El aprovisionamiento de infraestructura de TI se asocia en su mayoría con el modelo IaaS. En este sentido, el aprovisionamiento pretende el modelamiento de toda una infraestructura de TI diseñada a las necesidades de la empresa, buscando la mayor rentabilidad, escalabilidad y disponibilidad a lo largo del tiempo; evitando en el gasto de hardware posiblemente innecesario a los inicios de la empresa.

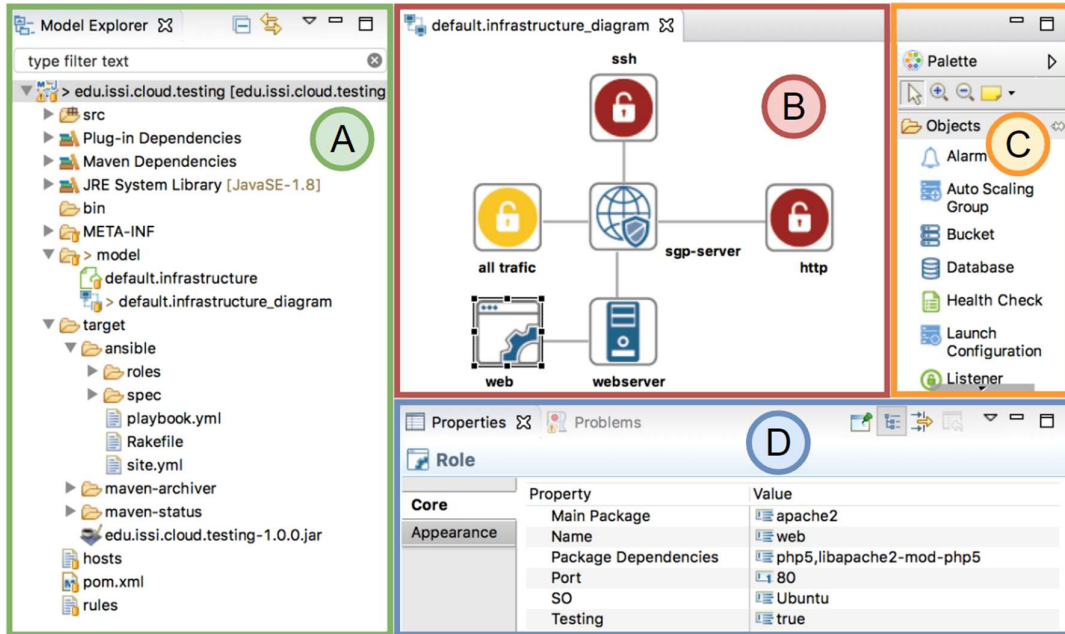
Dentro del modelo IaaS entra un enfoque para agilizar y automatizar el aprovisionamiento de infraestructura, denominado Infraestructura como Código (IaC). Este enfoque permite plasmar en scripts los comandos e instrucciones necesarias para la creación, actualización y ejecución de la infraestructura deseada, contemplando a las principales plataformas dedicadas a la prestación de infraestructura en la nube, como lo son Amazon AWS o Microsoft Azure [11]. Existen diferentes herramientas de IaC, cada una con un

lenguaje de scripting propietario, lo cual dificulta definir una infraestructura en específico, ya que no solo se tiene distintos tipos de scripting dependiendo de la herramienta, sino que cada proveedor también ofrece un tipo de infraestructura en específico. Este abanico de posibilidades hace que la administración de estos scripts de IaC se torne en una actividad lenta y vulnerable a errores [11].

### **1.4.3 ARGON**

ARGON es una herramienta dirigida al modelado de infraestructura, permitiendo el aprovisionamiento de recursos en la nube. ARGON se basa en el enfoque de IaC permitiendo generar los scripts necesarios para el aprovisionamiento a los distintos proveedores en la nube [11]. ARGON puede definir cierta infraestructura haciendo uso de su lenguaje de modelado; un Lenguaje Especifico de Dominio (DSL). Interactuar con el DSL generalmente implica interactuar con la línea de comandos. El DSL generaliza la escritura de scripts: el usuario utiliza un lenguaje general para definir la infraestructura para luego, a partir de la definición general, generar scripts concretos/específicos para cada proveedor en la nube. Es decir, ARGON es una herramienta que permite dejar de lado la complejidad de conocer y escribir en diferentes lenguajes de scripting, proveyendo los recursos necesarios para implementar una infraestructura para los distintos proveedores en la nube [11].

La construcción del editor gráfico de ARGON se fundamentó en una arquitectura basada en componentes y se encuentra implementada para Eclipse Modeling Framework o dentro de un determinado servidor de integración [13]. Cada uno de estos componentes están empaquetados en el tipo de archivo JAR que corresponden a los archivos de JAVA. La figura 2 muestra la interfaz de usuario de ARGON donde se puede identificar los componentes implementados en el editor de Eclipse. Existen cuatro componentes esenciales para su funcionamiento. La sección “A” de la figura 2, indica el explorador de archivos que se genera para un solo proyecto de modelamiento, que además es parte fundamental de la herramienta de Eclipse. La visualización del modelo gráfico se encuentra en la parte “B” de la figura 2, es en este apartado es donde el usuario interactúa con los componentes previamente escogidos de la paleta ubicada en la parte “C”. Una vez escogido cierto componente se puede modificar las propiedades que caracterizan a dicho elemento en la sección “D”.



**Figura 2.** Editor Gráfico de ARGON en Eclipse Modeling Framework. Fuente: Sandobalin, Insfran y Abrahao [12].

Desde su creación, el editor gráfico estuvo enfocado en facilitar, a los usuarios, el uso del DSL subyacente de ARGON. No obstante, a pesar de que el objetivo fue “facilitar” la interacción al “usuario”, precisamente no se consideraron, en su desarrollo, aspectos tanto interactivos como estéticos que son primordiales para que la interacción usuario-herramienta sea fácil, sin complicaciones. Por lo tanto, este trabajo está enfocado en analizar el editor gráfico, a fin de encontrar y proponer soluciones específicas que le ayuden al usuario a interactuar de mejor manera. A partir de ahora, se hará alusión al editor gráfico de ARGON como la interfaz de usuario correspondiente a la herramienta.

### 1.4.4 Interfaz de Usuario (UI)

Una interfaz de usuario se define como todo componente de hardware o software que permite a cualquier tipo de usuario interactuar con esta. Dicha interacción puede ser dada de forma física, por ejemplo: tocar, palpar, oír, ver, hablar [13]. Para que se lleve a cabo esta interacción, la UI al menos tiene 2 componentes principales: la entrada (input) y salida (output) de información. Algunos ejemplos de entrada son componentes bastante comunes como el ratón, teclado o paneles táctiles, mientras que las salidas frecuentes son pantallas led o proyectores.

Dentro de las UI encontramos distintos tipos siendo clasificadas por sus componentes hasta por su forma de interacción. Dentro de las últimas tenemos las Interfaces por la línea de comandos (Command-Line Interface, CLI), Interfaces Gráficas (Graphic User Interface, GUI) y por último las Interfaces Naturales (Natural User Interface, NUI). Para este proyecto será esencial la aclaración de las 2 primeras interfaces: CLI y GUI.

#### **1.4.4.1 Interfaz por línea de Comandos (CLI)**

Este tipo de Interfaz hace referencia a la entrada de instrucciones mediante el tipeo única y exclusivamente de texto, permitiendo al usuario crear sentencias que serán ejecutadas por el programa escucha. Usualmente, este tipo de interfaz está ligada a la ejecución de programas especializados como los DSL o instrucciones específicas al sistema operativo [14]. Su uso se limita a usuarios especializados ya que es necesario cierto aprendizaje sobre la sintaxis que entiende el programa.

#### **1.4.4.2 Interfaz Gráfica (GUI)**

Para este caso, la Interfaz Gráfica se puede definir por su forma de interacción con el usuario. Es decir, un programa que tenga una UI reaccionará a las acciones de un determinado usuario, dichas acciones pueden ser el movimiento del ratón, el aplastamiento de teclas o botones con el objetivo de hacer una acción que se muestra en pantalla [15]. Dependiendo de su diseño, composición y orden en sus elementos, la UI puede o no ser intuitiva para el usuario. Por lo general, una GUI bien diseñada logra disminuir gran parte de la carga cognitiva en el usuario, pues brinda la información necesaria para su siguiente acción. Además, existe gran interés por diseños de UI que provoquen respuestas emocionales, atrayendo la atención del usuario de forma lúdica con distintas animaciones [16].

La interacción con ARGON, al ser una herramienta basada en DSL, se realiza por línea de comandos. Reemplazar la interfaz CLI de ARGON por una GUI, permitiría reducir la complejidad de aprender la sintaxis empleada en el CLI. Es por eso por lo que, implementar un editor gráfico, como interfaz de usuario GUI, contribuye a disminuir la complejidad de interacción con ARGON.



### **1.4.5 Experiencia de Usuario (UX)**

En el contexto más general, la UX hace alusión a cómo se siente el usuario al interactuar con un producto o servicio. La “sensación” (lo que siente o percibe el usuario) está influenciada por la formación y sincronización de diversos componentes que afectan para bien o mal la experiencia que el usuario experimenta al usar o interactuar con un producto o servicio. Es decir, la UX considera los sentimientos y la satisfacción que un individuo experimenta al usar y palpar cierto producto/servicio [17].

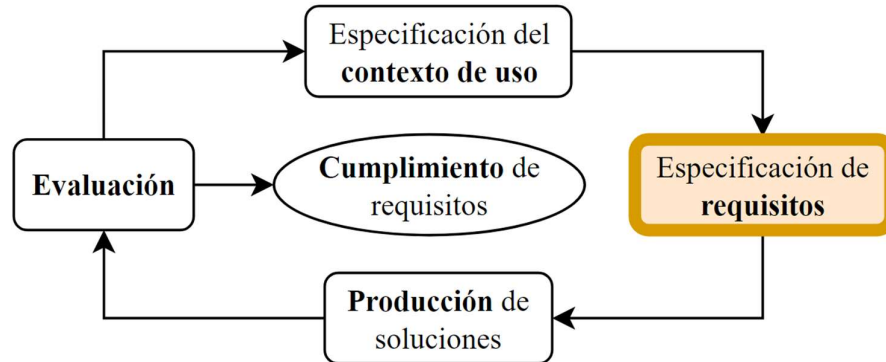
En la arena digital, la UX tiene que ver con la sensación que experimenta el usuario al interactuar con productos digitales. En el caso de este proyecto, la UX tiene que ver con los componentes que afectan la experiencia de un usuario al interactuar con ARGON. Para determinar dichos componentes, es importante considerar los objetivos de ARGON (la herramienta a evaluar), las necesidades de los usuarios, así como también las distintas limitaciones de la herramienta.

Considerando el concepto de UX, el “diseño de UX” es entonces una práctica profesional que persigue influenciar el comportamiento o percepción del usuario al interactuar con productos/servicios [18].

Existen 3 diferentes metodologías para diseñar la UX. La primera es el Diseño Centrado en el Humano (HCD), cuya perspectiva es el ser humano como género, a fin de buscar problemas en diseño existentes proponiendo soluciones. La segunda es el Diseño Centrado en el Usuario (DCU) compartiendo muchas similitudes con HCD pero haciendo una clara distinción sobre el usuario objetivo entre todos los seres humanos. Por último, se tiene el Diseño Centrado en la Experiencia (XCD), cuyo enfoque son palabras claves siendo sus diseños resultantes basados en los conceptos que se tienen de estas palabras.

El diseño de la UX puede ser aplicable desde la creación de un producto nuevo, hasta el mejoramiento de un producto ya existente. En el caso de mejorar la UX de un producto, es necesario investigar los problemas existentes que afectan su uso, con el fin de proponer posibles soluciones a ser implementadas. Para ello, la técnica “Evaluación Heurística”, que es aplicada en el DCU, servirá en la investigación de estos problemas.

La figura 3 muestra el proceso de cuatro actividades para diseñar un software interactivo, que el estándar ISO9241 vincula al DCU. La evaluación heurística es aplicada en la actividad “Especificación de requisitos” (considerando que el producto existe). En efecto, el análisis heurístico, tema de este trabajo, es una excelente técnica que contribuye al levantamiento y detalle de los requerimientos del usuario objetivo.



**Figura 3.** Metodología DCU basado en la ISO9241, indicando el estado donde se puede realizar la evaluación heurística.

Para realizar la evaluación heurística de la IU de ARGON se debe tener claro el **contexto de uso**, es decir, identificar a las personas a las se dirige el producto (el usuario objetivo), para qué lo usarán y en qué condiciones [19] La técnica Personas es técnica comúnmente utilizada para definir al usuario objetivo. Dicha técnica es una de las diversas técnicas aplicadas en DCU para estructurar la información del usuario objetivo y que generalmente es aplicada en la actividad “Especificación de requisitos”.

### 1.4.6 Técnica Personas

Esta técnica permite modelar de uno o varios usuarios objetivos a partir de la descripción de un grupo numeroso de personas [20]. Este modelo de usuario debe contener parámetros que describan en su mayoría las destrezas, habilidades, debilidades destrezas y demás aspectos que sean comunes en todo el grupo de personas seleccionadas [21]. Esta técnica permitirá definir las necesidades y requerimientos más importantes que el grupo de personas espera que ARGON cumpla. Una vez contemplado el modelo de usuario, para fines prácticos se le denominará como “Perfil Persona”.

## 1.4.7 Heurísticas y Análisis Heurístico de Usabilidad

La usabilidad es parte de la UX y se define como un atributo de calidad de un producto/servicio. Dicho atributo permite evaluar la facilidad que tiene un usuario al hacer uso de un servicio o producto, como por ejemplo la interfaz de usuario [22]. Lograr que un producto sea “usable” involucra aplicar diversos principios que dependen tanto de la tecnología como de la plataforma en la que se desarrolla el producto.

En el caso de la interfaz de usuario, existen principios para interfaces de aplicativos móviles, web y hasta dispositivos inteligentes como un smartwatch [16]. Estos principios son usados para diseñar e implementación las interfaces de usuario. No obstante, cuando estos principios son utilizados para evaluar la interfaz de usuario, estos toman el nombre de **Heurísticas**. Evaluar la interfaz de usuario mediante las heurísticas es una práctica conocida como Análisis o Evaluación Heurística. La tabla 1, detalla las heurísticas de Jakob Nielsen las cuales son empleadas para evaluar interfaces de usuario, incluyendo aquellas implementadas en plataforma Web.

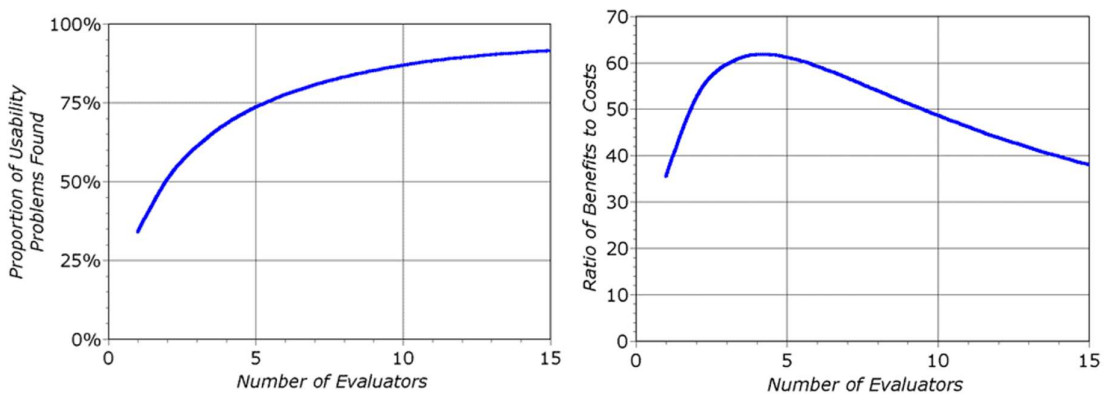
**Tabla 1.** Heurísticas de Usabilidad definidas por Jakob Nielsen. Fuente: Nielsen [23].

Heurística	Definición
1. Visibilidad del estado del sistema	Mantener al usuario consciente de lo que está sucediendo por medio de comentarios lo más rápido posible.
2. Coincidencia entre el sistema y el mundo real	Utilizar el mayor léxico familiar para el usuario, asegurando un correcto entendimiento de conceptos y significados tanto de palabras, imágenes e iconos que se observan en el mundo físico.
3. Control y libertad del usuario	Permitir al usuario deshacer acciones o procesos que erróneamente empezó, mostrando de manera fácil e intuitiva la salida.
4. Consistencia y estándares	Mantener las convenciones establecidas por productos ya estandarizados. Es decir, el usuario no debe aprender nuevos símbolos, mecánicas de interacción o demás elementos visuales que son comunes observarlos de cierta forma en otros productos.
5. Diagnóstico y recuperación de errores.	Informar de manera sencilla sobre los errores producidos indicando al usuario las maneras de poder solucionarlo.
6. Prevención de errores	Eliminar o prevenir cualquier situaciones o condición que pueda resultar ambigua al usuario causándole acciones inesperadas.
7. Reconocimiento en lugar de recordar	Permitir al usuario que reconozca la información de la interfaz en lugar de tener que memorizar elementos,

	acciones y opciones.
8. Eficiencia de uso y Flexibilidad	Proporcionar distintos accesos directos u ocultos que le permita a cualquier usuario personalizar o agilizar ciertas acciones frecuentes o repetitivas.
9. Diseño estético y minimalista	Mantener interfaces de usuario lo menos cargada de información irrelevante, que dificulte al usuario identificar la información relevante.
10. Ayuda y documentación	Proporcionar al usuario documentación externa que permita comprender como ejecutar o corregir tareas en específico.

### 1.4.8 Evaluación Heurística

Este tipo evaluación resulta conveniente para agilizar tiempos de entrega y resolver conflictos, pues de manera rápida se puede obtener retroalimentación sobre diseños, prototipos e interfaces de usuario ya implementadas [18]. Además, es realizado por expertos quienes a partir de su amplio conocimiento en el área de la interacción y usabilidad pueden ofrecer retroalimentación apropiada, permitiendo realizar este análisis en cualquier etapa del ciclo de un proyecto de software. Esto a menudo logra estructurar ciertas pruebas de usuario, ya que la usabilidad empelada depende del tipo de usuario final al que esté enfocado el proyecto. [17]. Sin embargo, el número de evaluadores dependerá del costo de contratar evaluadores y el beneficio que se obtiene.



**Figura 4.** Curvas de proporción; problemas de usabilidad encontrados y relación beneficios a costos por el número de evaluadores. Fuente: Nielsen [24].

La figura 4 muestra 2 curvas donde según Jacob Nielsen en 1993, se relaciona al número de evaluadores con respecto a los problemas encontrados y a los beneficios

obtenidos por el costo (respectivamente ilustrados). Estos resultados muestran que un único evaluador logra encontrar tan solo un 35% de los problemas de usabilidad, incrementando este porcentaje mientras se añadan más evaluadores. Por otra parte, Nielsen también indica que el beneficio obtenido por el costo a pagar a cada evaluador disminuye a partir de 4 evaluadores, mostrando así que las evaluaciones del quinto hasta el décimo quinto evaluador serán redundantes o a su vez no identifique problemas de gran relevancia. Por ello, el número óptimo de evaluadores estará entre 3 a 5, proporcionando y abarcando en su gran mayoría los problemas de usabilidad más relevantes.

En un nivel macro, una evaluación heurística debe tener al menos 5 apartados [18].

1. **Reunir conocimiento previo de la herramienta a analizar:** Identificar el propósito, objetivos y los usuarios específicos para los que fue creada la interfaz de usuario.
2. **Elección de las heurísticas a evaluar:** Para evaluar la IU de ARGON se tendrá en cuenta la implementación a futuro será como servicio web, por lo que las heurísticas empleadas serán parte de las 10 heurísticas propuestas por Jacob Nielsen.
3. **Identificar las áreas prioritarias de la herramienta a evaluar:** Se identifica las áreas que necesitan ser evaluadas, las cuales serán el foco del estudio para comprobar su implementación o falta de usabilidad.
4. **Generación de retroalimentación por parte de expertos:** Se solicita a expertos que intenten realizar tareas, las cuales definen escenarios que comprenden las áreas priorizadas a evaluar.
5. **Recopilar y presentar errores o problemas encontrados:** Se reúne los errores de usabilidad encontrados en las áreas evaluadas de la interfaz de usuario y se los presenta de manera sintetizada.

Para realizar la Evaluación Heurística de ARGON, se ha adaptado los 5 apartados macro a 4 apartados principales. Esto se ha realizado para asociar las tareas requeridas cumpliendo así el objetivo macro de los 5 apartados. En este sentido, el apartado 1 y 2 fueron combinados en uno solo. La tabla 2 describe cada apartado y las técnicas necesarias a emplear.

**Tabla 2.** Apartados para llevar a cabo la Evaluación Heurística.

Apartado	Propósito	Tareas
----------	-----------	--------

1. Preparación	Identificar los objetivos y propósito de la herramienta a analizar, conociendo el perfil usuario y el caso de uso a probar. Además, se define las heurísticas a evaluar haciendo un listado de apartados a tener en consideración.	<ul style="list-style-type: none"> <li>• Conocer la GUI</li> <li>• Perfil de usuario con la técnica de PERSONA</li> <li>• Tareas Clave, dependiendo del caso de uso a analizar</li> <li>• Determinar las Heurísticas a identificar y comprobar</li> <li>• Elaborar listado de verificación para la evaluación</li> </ul>
2. Revisión del Producto	Se determinará una exploración por la GUI, conociendo y familiarizando con su funcionamiento. Además, se debe identificar las áreas importantes a ser analizadas como reconocer los posibles errores a comprobar.	<ul style="list-style-type: none"> <li>• Exploración minuciosa de la GUI</li> <li>• Evaluación interna con el listado de verificación</li> </ul>
3. Opinión de Expertos	Se pretende obtener la retroalimentación necesaria dada luego de la evaluación por los expertos.	<ul style="list-style-type: none"> <li>• Generación de plantilla del informe generado por los expertos</li> <li>• Evaluación individual por parte de los expertos</li> </ul>
4. Preparación de Reporte	Se enfoca en recopilar toda la retroalimentación dada por los expertos como por la generada internamente, estructurando e identificando los problemas encontrados y sus posibles recomendaciones.	<ul style="list-style-type: none"> <li>• Recopilación de Hallazgos</li> <li>• Asignación de severidad a los problemas hallados</li> <li>• Generar presentación</li> </ul>

## 2 METODOLOGÍA

El desarrollo de este trabajo de integración curricular sigue cada apartado indicado en la tabla 2. A continuación, se describe cada apartado con las correspondientes decisiones llevadas a cabo.

### 2.1 Preparación

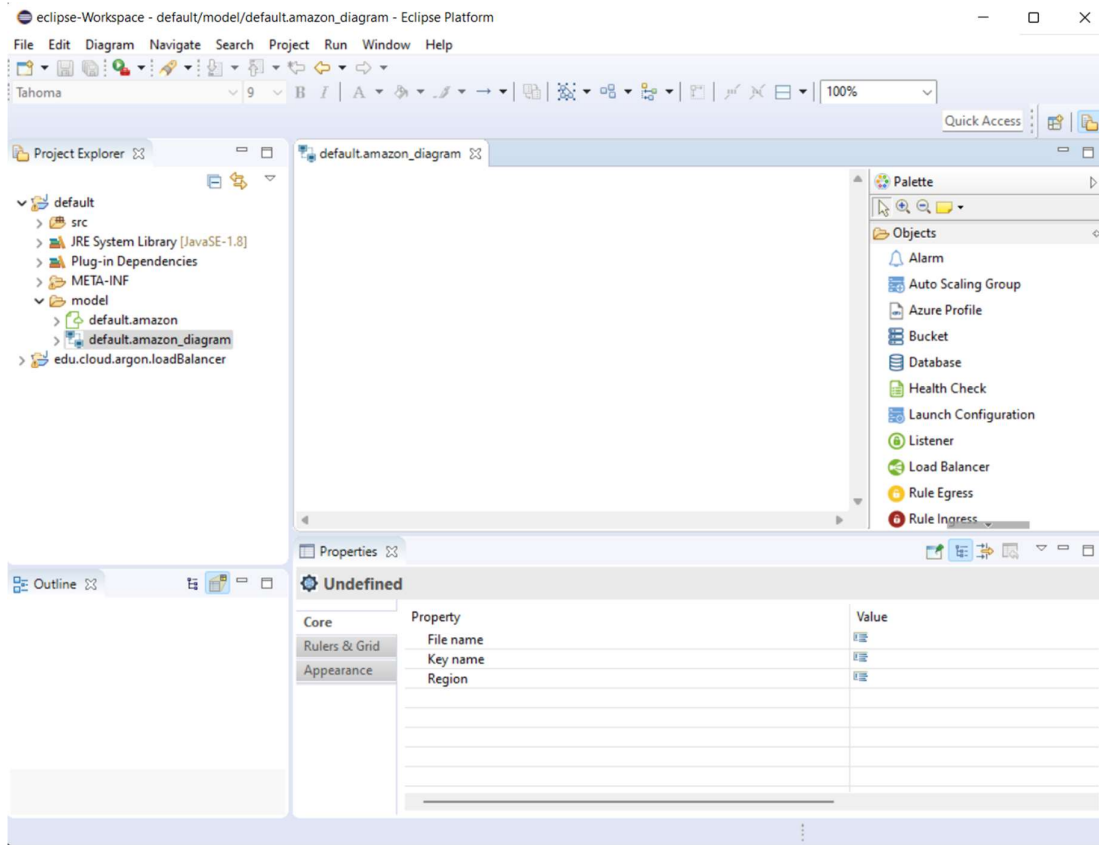
La importancia de este apartado radica en definir: 1) el funcionamiento de la herramienta de ARGON, 2) el perfil del usuario que hace uso de ARGON, también conocido como “usuario tipo” y 3) las heurísticas que serán empleadas para evaluar la IU de ARGON. El resultado de este apartado generará la información de contexto necesaria para que los expertos en usabilidad puedan evaluar la IU de ARGON.

### **2.1.1 Objetivos de ARGON**

Los objetivos de ARGON fueron definidos a partir de la sinergia de dos perspectivas: 1) investigar la literatura relacionada a la creación y propuesta de la herramienta y 2) exploración directamente la herramienta.

Respecto a la investigación de la literatura, se consideró como seminal al artículo de Sandobalín et. al [11] donde se hace referencia a la propuesta de creación de ARGON y se describe un ejercicio como prueba de concepto de la herramienta. El PhD. Julio Sandobalín es uno de los creadores de la herramienta y actualmente es docente de la Facultad de Sistemas.

Respecto a la exploración directa, ARGON fue instalado en su software base con el fin de interactuar con su interfaz de usuario, el editor gráfico. Ya que ARGON fue desarrollado como un componente acoplable al IDE Eclipse, la instalación consideró primero instalar Eclipse Modeling Framework como software base para luego instalar ARGON como componente acoplado al IDE Eclipse. La figura 5 muestra la interfaz de usuario (IU) del editor gráfico de ARGON. Con ARGON instalado, se exploró la IU del editor de manera práctica, esto es, interactuando con la IU para intentar recrear el ejercicio “balanceo de carga” descrito como prueba de concepto de la herramienta en el artículo científico de los autores [11].



**Figura 5.** Interfaz de usuario del editor de ARGON. Fuente: Herramienta ARGON.

Considerando tanto la investigación de literatura como la exploración directa, se han definido los siguientes objetivos que ARGON con su editor grafico pretende conseguir:

- Modelar infraestructura de manera visual, haciendo uso de íconos representativos a los elementos físicos (servidores, routers, switches, etc).
- Configurar cada elemento de la infraestructura, definiendo sus propiedades (las propiedades se definen en el panel “*Properties*” ubicado en la parte inferior de la Figura 5). Además, se infiere que estas propiedades pretenden ser mostradas de forma clara y simple, a fin de no añadir mensajes informativos o de ayuda al usuario.
- Visualizar de manera global el modelo, indicando la sección del modelo que se encuentra maximizado.
- Mostrar los elementos de infraestructura de manera genérica (sin peculiaridades de los fabricantes, como se muestra en la sección “*Palette*” de la Figura 5).



- Generar, de manera automática, un script cuyo código representa la infraestructura modelada. Es decir, el script es el resultado de transformar el modelo gráfico de la infraestructura a código del DSL de ARGON.

Los objetivos aquí definidos son fundamentales para idear cómo llevar a cabo la evaluación heurística, pues dichos objetivos resaltan las limitantes, funcionalidades y alcance de la herramienta y deben ser considerados por los expertos en la evaluación. Previo a idear la evaluación heurística, es importante también, además de definir los objetivos de ARGON, definir las habilidades, capacidades y limitaciones de la persona que hará uso de ARGON. En la siguiente sección se define el perfil usuario que hará uso de ARGON.

### 2.1.2 Perfil Usuario

Al tener un acercamiento con el PhD. Julio Sandobalín, uno de los autores de la herramienta ARGON, se constató que, aunque la herramienta fue pensada para profesionales involucrados en la gestión y mantenimiento de equipos tecnológicos, hubo cierto rechazo/desinterés por parte de este público. Por ello, el autor pretende que el uso de ARGON sea en el ámbito académico, proporcionando a los estudiantes una visión intuitiva de cómo modelar e implementar una Infraestructura TI.

En consecuencia, para comprender al usuario objetivo, se realizó una encuesta a los estudiantes que harían uso de ARGON. La encuesta, como técnica de recolección de información, consistió en 19 preguntas agrupadas en 4 secciones, como se detalla en la tabla 3. Los resultados de la encuesta son el insumo para posteriormente sintetizarlos, a través de la técnica Persona, en una plantilla que describe el perfil del “usuario tipo” que usa la herramienta ARGON.

**Tabla 3.** Preguntas de la encuesta para definir la plantilla Persona.

<ul style="list-style-type: none"> <li>• <b>Información Personal</b> <ol style="list-style-type: none"> <li>1. ¿Cuál es su género?</li> <li>2. Indique su rango de edad</li> <li>3. Indique la carrera que actualmente está cursando</li> <li>4. ¿Cuál es el semestre que está cursando actualmente?</li> <li>5. ¿Qué área(s) de la informática le interesa más?</li> </ol> </li> </ul>
---

<ul style="list-style-type: none"> <li>• <b>Infraestructura de (Tecnologías de la Información) TI</b> <ol style="list-style-type: none"> <li>6. ¿Cree usted que modelar la Infraestructura TI de una empresa es complejo?</li> <li>7. Describa en un párrafo corto, ¿por qué cree que es complejo modelar la Infraestructura TI?</li> <li>8. ¿Ha diseñado/modelado una Infraestructura TI? (incluye proyectos pequeños o medianos hechos en clases de Redes)</li> <li>9. ¿Qué tipo de casos ha modelado?</li> <li>10. ¿Con qué aspecto cree usted que tiene o tendría problemas al diseñar e implementar una Infraestructura TI?</li> </ol> </li> </ul>
<ul style="list-style-type: none"> <li>• <b>Infraestructura como Servicio</b> <ol style="list-style-type: none"> <li>11. ¿Conoce el concepto de "Infraestructura como Servicio" (IaaS)?</li> <li>12. ¿Conoce o ha escuchado acerca del término "aprovisionamiento de infraestructura" en el contexto de la informática?</li> <li>13. ¿De qué Proveedor de Infraestructura tiene mayor conocimiento?</li> <li>14. ¿Qué problemas ha encontrado al utilizar los servicios de los Proveedores de Infraestructura?</li> <li>15. ¿Estaría dispuesto a aprender a gestionar la Infraestructura de TI utilizando scripts de código?</li> </ol> </li> </ul>
<ul style="list-style-type: none"> <li>• <b>Editores e Interfaz de Usuario</b> <ol style="list-style-type: none"> <li>16. ¿Qué editores gráficos ha utilizado para diagramar y/o entender visualmente una infraestructura TI?</li> <li>17. Describa la utilidad que le aporta usar el o los editores gráficos seleccionados</li> <li>18. ¿Qué tipo de interfaz de usuario le haría sentir más cómodo para gestionar la infraestructura de TI en la nube?</li> <li>19. ¿Qué dispositivo(s) electrónico(s) usted utilizaría para diseñar e implementar una infraestructura TI?</li> </ol> </li> </ul>

Las 4 secciones de la tabla 3 están enfocadas en conocer, paso a paso, al usuario. En la primera sección, información personal, se identifica al estudiante conociendo su edad, carrera de pre-grado y hasta las áreas de su carrera que más le interesa. La segunda sección, infraestructura de TI, se enfoca en conocer el nivel de conocimiento y la destreza en asimilar el concepto de infraestructura de TI y sus aplicaciones. La tercera sección, infraestructura como servicio, se enfoca en conocer lo que el usuario entiende y conoce sobre el servicio ofrecido por proveedores para levantar/aprovisionar infraestructura en la nube. Por último, la cuarta sección, editores e interfaz de usuario, se enfoca en conocer los editores que el usuario conoce, utiliza y cuán fácil es para él utilizarlos. El lector puede ver en el ANEXO I. Formulario para levantar Perfil Usuario, la implementación de la encuesta con las preguntas de la tabla 3.

Para ejecutar la encuesta se propuso, en primer lugar, tomar en cuenta a estudiantes que cursaron la asignatura de tercer semestre del pregrado e Ingeniería en Software: "Fundamentos de Redes y Conectividad ICCD353". Este alcance se realizó debido a

que se espera que estos estudiantes estén familiarizados con distintos elementos de infraestructura.

No obstante, pruebas previas de concepto realizadas de la encuesta reportaron un número considerable de estudiantes de niveles superiores con escasos conocimientos en el aprovisionamiento de infraestructura en la nube, a pesar de haber realizado proyectos con infraestructura. Esta connotación es importante ya que, ARGON podría considerarse como una interesante herramienta para la instrucción académica en esta área.

Dado el número significativo de estudiantes que desconocen sobre el aprovisionamiento en la nube, se consideró ampliar el rango de estudiantes a aquellos que han cursado la asignatura de redes, esta vez, incluyendo a estudiantes del pregrado de Ingeniería en Computación. Con estas consideraciones, se difundió la encuesta a la mayor cantidad de personas logrando obtener respuestas de 54 participantes. El ANEXO II. Resultados de la encuesta para Perfil Persona muestra los resultados obtenidos expresados en porcentajes.

Tomando en cuenta los resultados se procedió a definir la plantilla Persona, como lo muestra la figura 6. La plantilla muestra los aspectos claves de las personas que llenaron la encuesta. La plantilla muestra estos aspectos clasificados en 5 apartados. El primero, "DESCRIPCION", describe a la persona por las actividades que realiza. El segundo, "OBJETIVO", detalla el objetivo que pretende conseguir la persona. El tercero, "MOTIVACIONES", describe las motivaciones que tiene la persona para usar el editor gráfico. La cuarta, "FRUSTRACIONES", muestra las frustraciones y complicaciones que tiene la persona cuando intenta modelar e implementar infraestructura en la nube. El último, "TECNOLOGIAS", muestra las tecnologías que la persona utiliza, definiendo así las habilidades y debilidades que posee.

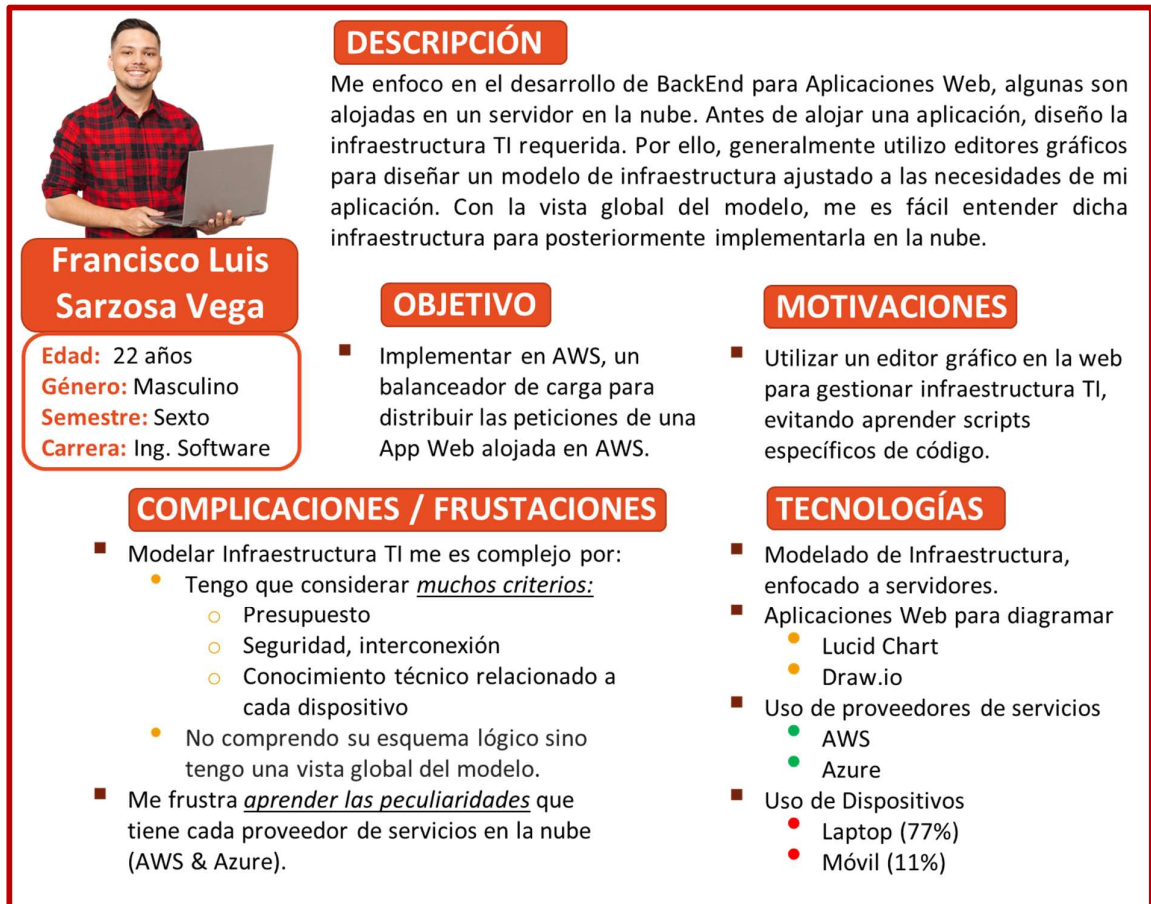


Figura 6. Plantilla Persona “Francisco Sarzosa”. Fuente: Elaboración propia.

### 2.1.3 Definición del Caso de Uso y sus Tareas Claves

Ya que el editor gráfico de ARGON consta únicamente de una sola IU, como lo muestra la figura 5, se propuso utilizar un caso de uso cuyas tareas permitan experimentar los distintos cambios que existirían en la IU. Para lo cual, se empleó el caso de uso “Load Balancer Provisioning in Amazon Web Services” que ha sido implementado y documentado con éxito en video por el Phd. Julio Sandobalin [25].

Una vez revisado y replicado el video con sus 10 pasos descritos, se infirió que únicamente los 8 primeros pasos pertenecen a la interacción que se tiene con la IU de ARGON. Las 2 primeras tareas han sido catalogadas como “tareas propias de Eclipse”, por lo que no forman parte de ARGON explícitamente. La tabla 4 describe el caso de uso y la secuencia de tareas (pasos) necesarias para llevar a cabo el modelamiento en ARGON.

**Tabla 4.** Secuencia de Tareas que definen el Caso de Uso para el aprovisionamiento de un Balanceador de Carga.

## DESCRIPCIÓN DE LA TAREA

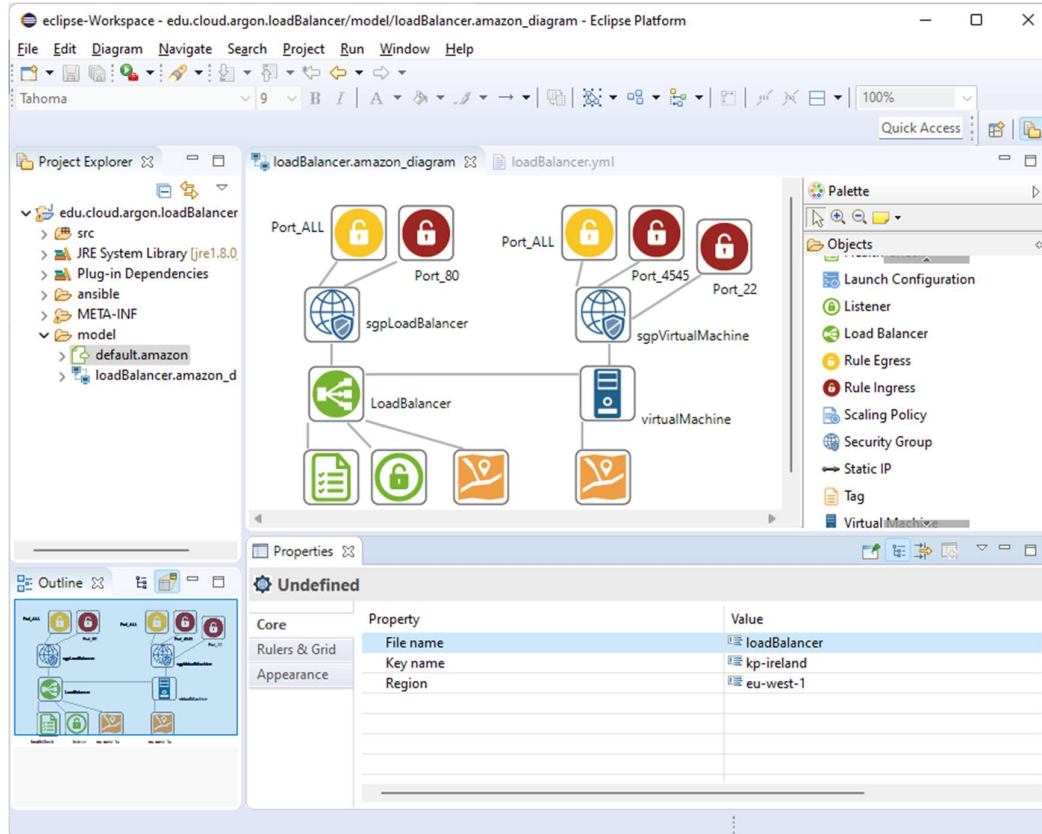
### **Aprovisionamiento de un Balanceador de Carga en Amazon Web Services (AWS)**

Balancear la carga es una actividad frecuente en el aprovisionamiento de infraestructura. Esta consiste en alivianar la cantidad de peticiones realizadas a un servidor, distribuyendo dichas peticiones entre varios servidores.

## SECUENCIA DE TAREAS

Para lograr el objetivo planteado, Francisco realiza 8 tareas usando ARGON como herramienta. Estas tareas se describen a continuación:

1. **Crear un diagrama de infraestructura para AWS (propio de Eclipse).**
2. **Completar las propiedades para el diagrama de infraestructura (propio de Eclipse).**
3. Modelar un balanceador de carga y sus respectivas máquinas virtuales.
4. Modelar un chequeo de salud (Health Check) para el balanceador de carga.
5. Modelar un oyente (Listener) para el balanceador de carga.
6. Modelar y delimitar zonas para:
  - Balanceador de Carga.
  - Máquinas Virtuales
7. Modelar grupos de seguridad con sus respectivas reglas de entradas y salidas para:
  - Balanceador de Carga.
  - Máquinas Virtuales
8. Generar el Script de la Infraestructura para Ansible.



**Figura 7.** Pantalla del editor de ARGON con el caso de uso implementado. Fuente: Elaboración propia.

La realización de las tareas contempladas permitirá observar los cambios posibles en la IU de ARGON, lo que permitirá obtener un mayor conocimiento del típico uso y los elementos que son desplegados en cada tarea. La figura 6 muestra el caso de uso implementado, el cual aporta más información visual que la figura 5, ya que, es posible visualizar un modelo típico con sus correspondientes elementos, relaciones y propiedades. Por ello, se propuso realizar parcialmente un documento denominado *Recorrido Cognitivo*, cuyo propósito será el de explicar al experto el Caso de Uso que se pretende analizar. Además, este documento también contemplaría a “Francisco”, el estudiante promedio (plantilla Persona), para que el experto considere las limitantes y circunstancias que Francisco tendría al intentar alcanzar su objetivo en ARGON. Se debe resaltar que, el documento será únicamente informativo al experto, ayudándole a comprender todos los cambios resultantes de cada tarea y a tener una mejor perspectiva sobre la herramienta a evaluar. El ANEXO III. Detalle del Caso de Uso (Documento de Recorrido Cognitivo)muestra el documento de Recorrido Cognitivo que para fines prácticos será nombrado como “Detalle del Caso de Uso”.

## **2.1.4 Elección de Heurísticas a utilizar**

Para elegir las heurísticas a evaluar, se debe tomar en cuenta las características y el estado de la herramienta a evaluar, es decir, si la herramienta es un prototipo o ya es una herramienta implementada. En este sentido, se ha considerado que:

1. ARGON es una herramienta implementada como ejecutable funcional para escritorio. Sin embargo, esta carece de ciertas características comunes como: recursos sonoros, documentación, manejo óptimo de errores y demás.
2. En el futuro, el autor desea implementar ARGON como un aplicativo web.

Dadas estas consideraciones, se ha decidido elegir las 10 heurísticas de Nielsen, de entre otras, debido a que estas han sido utilizadas con éxito comúnmente para el diseño de páginas y aplicativos webs. A pesar de que la carencia de características descritas en el punto 1 dejaría fuera de análisis a 3 de las 10 heurísticas de Nielsen (“3. Control y libertad del usuario”, “6. Prevención de Errores” y “10. Ayuda y Documentación”), se ha considerado que el análisis podría ser más enriquecedor si se abordan las 10 heurísticas en total. Así, se pretende constatar, con la opinión de los expertos, el nulo o bajo cumplimiento de ciertas heurísticas, lo que se traducirá en una cantidad significativa de sugerencias realizadas a dichas heurísticas. A continuación, se detalla la elaboración de la Lista de Verificación que incluye las 10 heurísticas.

## **2.1.5 Elaboración de la Lista de Verificación**

Para evaluar cada heurística, se definió un banco de preguntas. Para definir las preguntas, se consideraron trabajos en la literatura que presentan preguntas para lograr profundizar en cada heurística. Principalmente, se consideró en gran parte el trabajo realizado por Deniese Pierotti [26], cuyo recurso abierto a su uso define una “Lista de Verificación” que detalla un total de 254 preguntas relacionadas a cada una de las heurísticas de Nielsen. Estas preguntas fueron depuradas, pasando por diversas revisiones, eligiendo y modificando las preguntas hasta adaptarlas al contexto de la IU de ARGON. Además, se incluyeron ciertas preguntas (no estaban consideradas en el documento de Pierotti) para abordar aspectos específicos de ARGON. El ANEXO IV. Lista de Verificación de las Heurísticas de Nielsen muestra el resultado de la depuración realizada, el cual considera cierto diseño visual para hacer fácil la

diferenciación de las heurísticas y la elección de las posibles opciones de respuesta. Esta lista de verificación consta de 3 partes en forma de columnas: preguntas, opciones de respuesta y “Recomendaciones & Sugerencias”.

## **2.2 Revisión del Producto**

Una vez completada el apartado de preparación de la evaluación heurística con sus correspondientes documentos, se realiza una “evaluación interna” que implica revisar el material de preparación para verificar qué tan fácil sería para un experto usar el material para realizar la evaluación heurística. Para esta revisión, mi persona, Cristhian Muñoz, asumo el papel de un experto y simulo una prueba que haría un experto, tratando de identificar posibles problemas de usabilidad presentes en la IU de ARGON. La razón de asumir el papel de experto está justificada tanto por el entrenamiento en realizar evaluaciones heurísticas para propósitos de aprendizaje que he recibido en la asignatura de Usabilidad y Accesibilidad ISWD732. No obstante, es importante destacar que la “evaluación interna” será enriquecida en gran manera por las diversas opiniones de los expertos en sus correspondientes evaluaciones, las cuales se denominan “Evaluaciones Externas”. Los siguientes subapartados describen cómo se llevó a cabo la evaluación interna.

### **2.2.1 Entendimiento del producto**

Entender cómo funciona la herramienta ARGON implica verificar en la IU de ARGON el cumplimiento de los objetivos descritos en el apartado 2.1.1. No obstante, ARGON al contar únicamente con una sola pantalla, como lo muestra la figura 5, no es posible evidenciar fácilmente el cumplimiento de los objetivos de ARGON, ni los cambios o elementos que son parte de la IU. En la figura 5, por ejemplo, muestra un proyecto vacío lo cual no permite constatar el objetivo de “visualizar un modelo de infraestructura con sus elementos gráficos representativos”, tampoco es posible generar ni visualizar el script resultado de convertir el modelo de infraestructura a código.

Para evitar este problema y verificar si la IU de ARGON cumple con los objetivos del apartado 2.1.1, se ejecutó la secuencia de tareas del caso de uso “balanceo de carga”, con lo cual se logra visualizar el modelo de infraestructura de manera global y detallada, así como las correspondientes conexiones entre los elementos; siendo



posible generar el script del modelo como muestra la figura 6. Así pues, la aplicación de la secuencia de pasos del caso uso ayudó en gran manera a entender y obtener una perspectiva más amplia del uso y el alcance de la herramienta ARGON.

## **2.2.2 Descubrimiento de problemas**

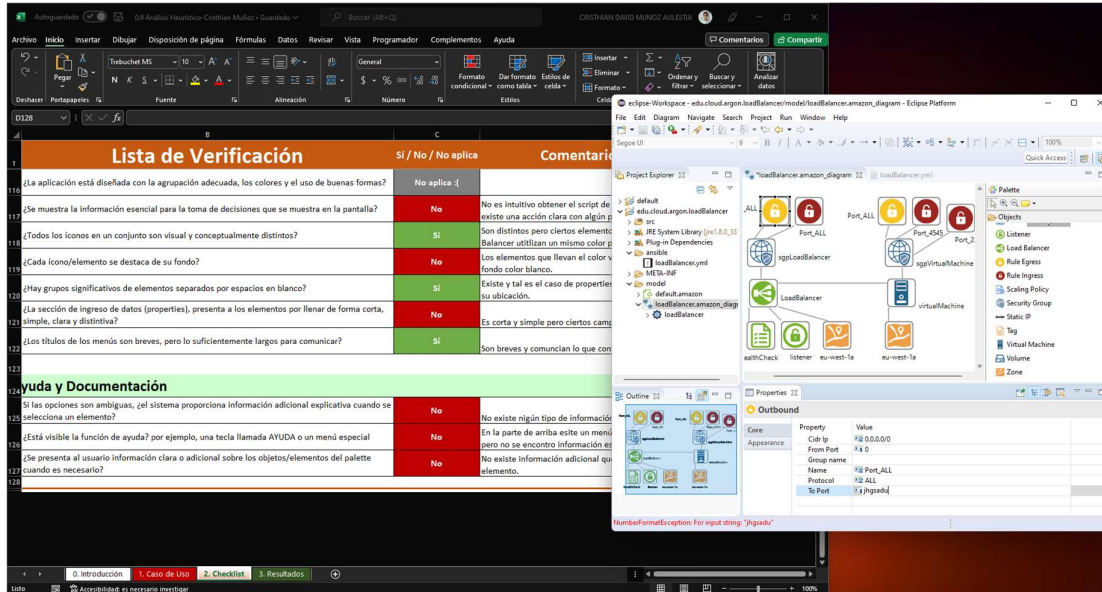
Una vez entendido cómo funciona ARGON, se procede a identificar/descubrir problemas de usabilidad en la IU. Para esto, se analiza las pantallas resultantes de la ejecución de cada tarea del caso de uso. El descubrimiento de problemas está guiado y soportado por las Lista de Verificación (ANEXO IV. Lista de Verificación de las Heurísticas de Nielsen) que contempla cada heurística. Es decir, el descubrimiento de los problemas se dará por las siguientes actividades.

### **2.2.2.1 Al Aplicar/Comprender las tareas del caso de uso**

Se hace uso del documento de “Detalle de Caso de Uso” para analizar el estado de la IU después de realizar la secuencia de 8 tareas propuestas del caso de uso. El documento “Detalle de Caso de Uso” describe el objetivo del caso de uso, las tareas a realizar para conseguirlo y cada tarea está acompañada de una captura de pantalla de la IU resultante. Comprendido el objetivo del caso de uso y el proceso hasta llegar a la tarea 8, se evaluó, apoyado de la Lista de Verificación, las IU de cada tarea. Visualizar la captura de pantalla de la IU en cada tarea no fue suficiente para evaluar las heurísticas, por lo que, de manera simultánea a visualizar la captura de pantalla, se interactuó directamente con la IU en la herramienta ARGON.

### **2.2.2.2 Al Interactuar/Explorar con la IU**

La interacción con la IU permitió responder a las preguntas de la lista de verificación. La interacción con la IU fue simultánea a la lectura de la lista de verificación. Una vez leída una pregunta, se intenta responderla a partir del resultado de los hallazgos encontrados durante la interacción con la IU, tal como lo indica la figura 7. La duración de la evaluación duró aproximadamente 1 hora y 20 minutos.



**Figura 8.** Captura de Pantalla de la interacción simultánea con el Caso de Uso implementado y la Lista de Verificación. Fuente: Elaboración propia.

Una vez terminada la Evaluación Interna, se obtuvo la lista de respuestas y comentarios que se encuentra en el ANEXO V. Evaluación Interna. El detalle de todos los problemas de usabilidad encontrados en la Evaluación Interna se especificará en el apartado 2.4 concerniente a los hallazgos de los problemas de usabilidad. No obstante, se puede mencionar de manera temprana que no existe cumplimiento de la heurística 10 relacionada con la ayuda y documentación, ya que, en las 3 preguntas relacionadas a esta heurística no se reportó ningún hallazgo de su cumplimiento. Además, las heurísticas 2, 5, 7, 8 y 9 reportan un cumplimiento de entre un 30% y 50%. Este breve análisis se deriva del conteo de las respuestas a la "Lista De Verificación", cuyas opciones válidas al conteo fueron "Sí" o "No". La opción "Sí" indica el cumplimiento de la heurística, la opción "No" indica el bajo o no cumplimiento.

Con este resultado tentativo, se espera que en las evaluaciones externas se encuentre un comportamiento similar en las respuestas para estas heurísticas, cuyo cumplimiento sea poco evidente en la IU de ARGON. A continuación, se preparará las evaluaciones externas a fin de obtener la opinión de los expertos en Usabilidad.

## **2.3 Opinión de Expertos**

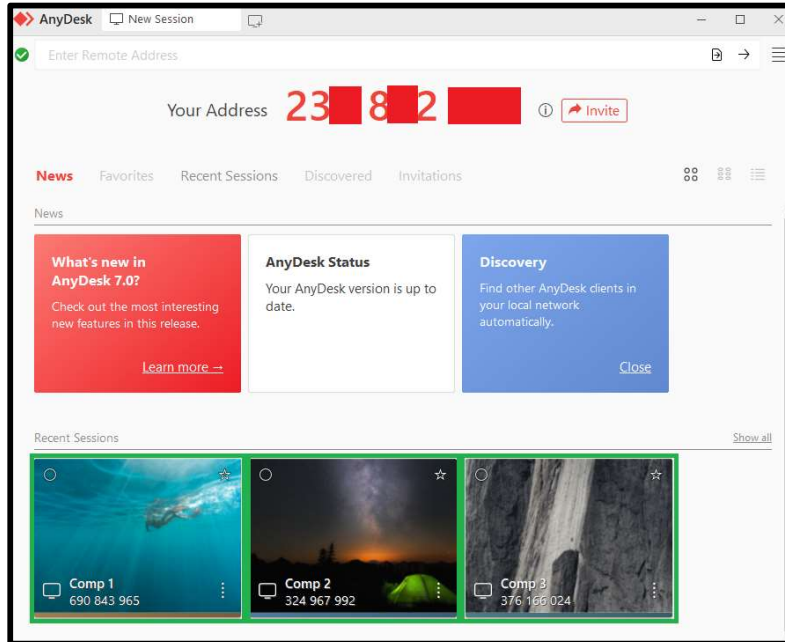
Para este apartado se han elegido 3 expertos que son docentes titulares de la Facultad de Sistemas: “Msc. Mayra Carrión Toro”, “PhD. Julián Galindo” y “Dr. Marco Santórum”. Los docentes fueron escogidos debido a su experiencia en dictar asignaturas relacionadas con la Usabilidad para distintos aplicativos, como por ejemplo: “Usabilidad y Accesibilidad ISWD732”, “Interacción Humano Computador ISWD723” y “Desarrollo de Juegos Interactivos ISWD823”.

Una vez elegidos los expertos, se debe proponer la forma o ambiente en el que se deberá desarrollar las evaluaciones externas. Considerando el tiempo limitado y variable de los expertos para llevar a cabo la evaluación, se propuso el uso de 3 máquinas virtuales para que se pongan a disposición de los expertos. En estas máquinas, se levantó el ambiente de evaluación que incluye el material y aplicaciones necesarias para la ejecución de la evaluación.

Con el fin de simplificar la cantidad de documentos entregados al experto, se ha unificado los documentos “Detalle del Caso de Uso” y “Lista de Verificación de las Heurísticas de Nielsen” en un solo fichero de Excel. Este fichero, denominado “Análisis Heurístico para ARGON” se muestra en el ANEXO VI. Análisis Heurístico para ARGON ([enlace](#)). Cabe mencionar que, en este mismo fichero Excel se ha agregado la pestaña “0. Introducción”, que corresponde al protocolo a seguir por el experto.

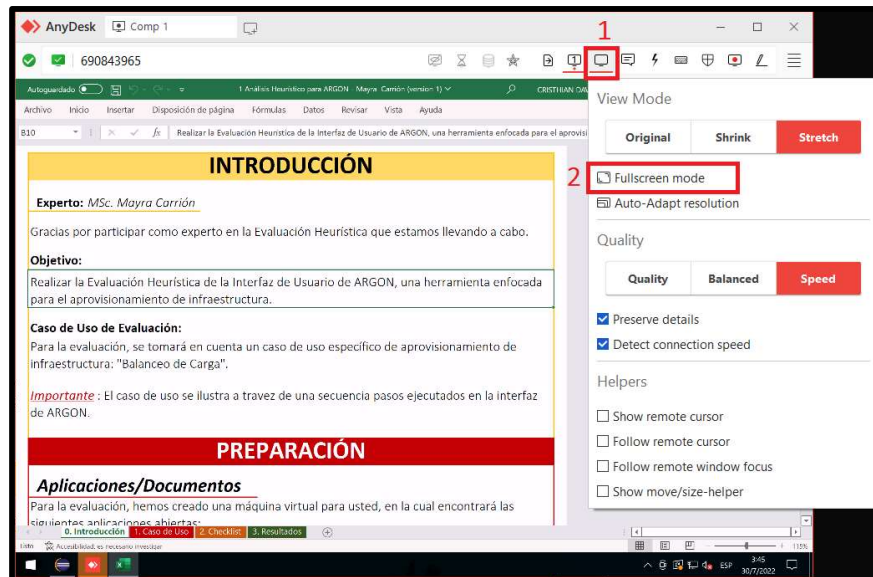
### **2.3.1 Protocolo de la Evaluación**

Al tener presente que el experto usará una máquina virtual, se envió a los expertos un correo informativo con las credenciales necesarias para acceder a la máquina virtual por medio del aplicativo AnyDesk. Este aplicativo de uso libre, permite tomar control de cualquier otra máquina conociendo su dirección IP virtual y la contraseña definida para su acceso con o sin privilegios. La figura 8 muestra el acceso a las 3 máquinas virtuales (encerradas en el contorno verde). En el campo “Enter Remote Address” (en la parte superior), el experto debe ingresar la IP virtual de su máquina virtual asignada y posteriormente su clave de acceso.



**Figura 9.** Captura de Pantalla de la intervención de AnyDesk para las máquinas virtuales.  
Fuente: Elaboración propia.

Una vez dentro de la máquina virtual, se espera que el experto visualice el espacio de trabajo como lo muestra la figura 9, donde se le recomienda seleccionar el ícono que señala el recuadro rojo #1 y posteriormente el número #2. Esto permitirá que el experto visualice en toda la pantalla de su computador la máquina virtual asignada, esperando así una interacción agradable y fácil de visualizar.



**Figura 10.** Captura de Pantalla de la manipulación esperada por los expertos. Fuente: Elaboración propia.

A continuación, se tiene en cuenta que el experto comience con la lectura de la pestaña abierta "0. Introducción". Esta pestaña contiene, a nivel general, 3 importantes secciones. La primera sección (figura 11) presenta el objetivo de la evaluación e indica el caso de uso a utilizar para la comprensión de la herramienta de ARGON.

## INTRODUCCIÓN

**Experto:** *Dr...*

---

Gracias por participar como experto en la Evaluación Heurística que estamos llevando a cabo.

**Objetivo:**  
Realizar la Evaluación Heurística de la Interfaz de Usuario de ARGON, una herramienta enfocada para el aprovisionamiento de infraestructura.

**Caso de Uso de Evaluación:**  
Para la evaluación, se tomará en cuenta un caso de uso específico de aprovisionamiento de infraestructura: "Balanceo de Carga".

***Importante:*** El caso de uso permitirá visualizar todos los cambios posibles presentados en la única pantalla que contiene la Interfaz de Usuario de ARGON.

**Figura 11.** Captura de la sección "Introducción" del protocolo a seguir. Fuente: Elaboración propia.

La segunda sección (figura 12), explica las aplicaciones y documentos que el experto deberá tomar en cuenta durante la evaluación. Esta explicación se limita a describir el contenido que va a encontrar el experto en la máquina virtual. El primer programa corresponde a OBS Studio, el cual permitirá grabar la interacción del experto durante toda la evaluación. El segundo, un documento Excel, que tendrá 4 pestañas que abarcará el protocolo a seguir, documento del Detalle del Caso de Uso (ANEXO III. Detalle del Caso de Uso (Documento de Recorrido Cognitivo)), Lista de verificación de las Heurísticas e Nielsen (ANEXO IV. Lista de Verificación de las Heurísticas de Nielsen) y, por último, el dashboard autogenerado que se explicará en la sección 2.4. El tercer programa es la herramienta de ARGON y su IU ejecutada por el aplicativo de Eclipse Modeling Framework.

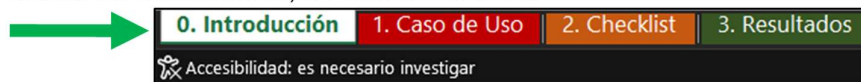
# PREPARACIÓN

## Aplicaciones/Documentos

Para la evaluación, hemos creado una máquina virtual para usted, en la cual encontrará las siguientes aplicaciones abiertas:

**1. OBS Studio:** Software que permitirá grabar su interacción con el computador mientras realiza la evaluación.

**2. Documento de Excel "Análisis Heurístico para ARGON":** Documento en que se encuentra actualmente, en este encontrará:

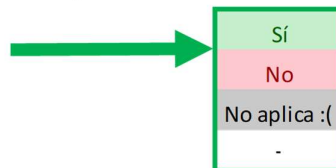


**2.0 Introducción:** Hoja informativa y en la que está actualmente.

**2.1 Caso de Uso:** En esta hoja encontrará a detalle el caso de uso y la ficha Persona de "Francisco", el usuario objetivo para el futuro aplicativo en la web de ARGON.

**Importante:** Las 2 primeras tareas se encuentran en rojo ya que **NO** son pasos propios de ARGON.

**2.2 Checklist:** Lista de Verificación basada en 9 de las 10 heurísticas propuestas por Jakob Nielsen. Se recomienda añadir todos los comentarios o sugerencias posibles a fin de obtener mayor retroalimentación. Además, la lista contiene la columna para marcar su respuesta, los únicos valores a seleccionar son:



**2.3 Resultados:** Resumen autogenerado del checklist.

### 3. Eclipse Workspace con el proyecto "edu.cloud.argon.loadBalancer"

Este Software contiene la herramienta de ARGON en forma de componentes. El caso de uso se encuentra completo en la herramienta para su correspondiente

**Figura 12.** Captura de la sección "Aplicativos/Documentos" del protocolo a seguir. Fuente: Elaboración propia.

La tercera sección (figura 13), corresponde a los pasos que deberá seguir el experto para llevar a cabo la Evaluación Heurística con éxito, tal como lo indica la figura 13. Sus cuatro pasos definen: 1) comenzar con la grabación de la evaluación, 2) comprender al perfil Persona y entender el caso de uso por las capturas de pantallas correspondientes a su secuencia de pasos, 3) interactuar con la IU de ARGON que implementa con éxito con el

caso de uso, 4) evaluar con la IU con la Lista de Verificación y 5) detener la grabación de la evaluación.

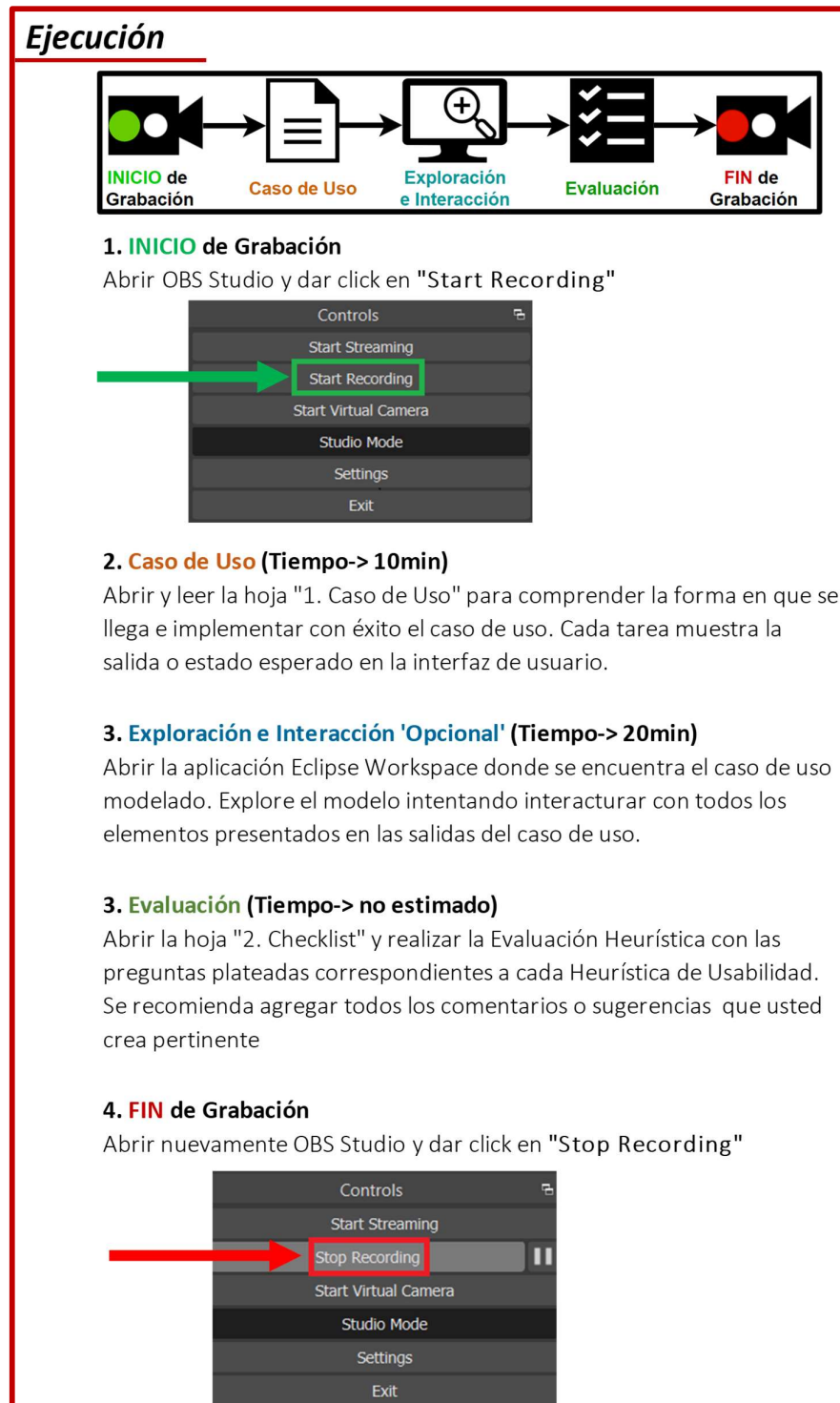


Figura 13. Captura de la sección "Aplicativos/Documents" del protocolo a seguir. Fuente: Elaboración propia.

### 2.3.1 Ejecución de la Evaluación

Una vez contemplado el protocolo a seguir y la preparación de las máquinas virtuales, se procedió coordinar el día y hora que el evaluador pueda disponer para la Evaluación Heurística. Sin embargo, se contempló que la grabación de la pantalla correspondiente a la máquina virtual OBS podría de poca ayuda pues no se consideraba las reacciones habladas ni las expresiones faciales del experto. Por ello, se propuso reunirse con cada experto en una sesión de Zoom a la hora y día que indicaron su disponibilidad, en la que se animaba al experto a prender su cámara y proyectar su pantalla. Esto debido a que, mientras el experto realiza la evaluación puede ir comentando en voz alta los problemas y o sugerencias que se le van presentado durante la ejecución de la evaluación; una técnica conocida como “Thinking Aloud” [27].

Además, se solicitó el consentimiento del experto para grabar la sesión. Esto con el fin de que la interacción (comentarios en voz alta, expresiones fáciles y sugerencias) quede registrada en video para posterior análisis profundo. Además, cabe mencionar que las 3 evaluaciones tuvieron que ser guiadas por mi persona, Cristhian Muñoz, cumpliendo el papel de facilitador para el experto.

El primer experto en realizar la Evaluación Heurística fue la MsC. Mayra Carrión, cuya evaluación fue llevada a cabo a las 6:50 pm, el viernes 22 de julio del 2022. La evaluación tomó una 1 hora y 27 minutos. Algunos problemas o aspectos relevantes durante la evaluación se describen en la tabla 5.

**Tabla 5.** Problemas y Aspectos relevantes de la experta MsC. Mayra Carrión.

<b>Aspectos Técnicos</b>
<ul style="list-style-type: none"><li>• Desconocimiento del uso del aplicativo AnyDesk para el sistema operativo MacOS (computador propio del experto), lo que causo molestias en la visibilidad del experto al no poder escalar la pantalla de la máquina virtual.</li><li>• Poco control del mouse que tenía el experto, lo que imposibilito el poder seleccionar fácilmente las opciones de respuesta para la Lista de Verificación.</li><li>• Visibilidad pobre del experto hacia el teclado, pues se manifestó que el experto al encontrarse en un lugar a oscuras no podía teclear muy bien.</li></ul>
<b>Desenvolvimiento del Experto</b>
<ul style="list-style-type: none"><li>• El tiempo resulto ser un factor de gran relevancia para el experto, por lo que, intento avanzar con la evaluación lo más rápido que podía, dejando de lado algunos aspectos y criterios de protocolo propuesto.</li></ul>



- Al tener cierta frustración al usar su teclado, prefirió dejar expuesto en la grabación su criterio correspondiente a los comentarios y sugerencias que eran parte de la Lista de Verificación.
- Poco o baja importancia a la pestaña correspondiente al Detalle del Caso de Uso (Anexo 3), dejando completamente de lado al perfil Persona.

El segundo experto en realizar la Evaluación Heurística fue el PhD. Julián Galindo, cuya evaluación fue llevada a cabo a las 5:15 pm, el martes 26 de julio del 2022. La evaluación tomó 1 hora y 54 minutos. Los problemas o aspectos relevantes durante la evaluación se describen en la tabla 6.

**Tabla 6.** Problemas y Aspectos relevantes del experto PhD. Julián Galindo.

<b>Aspectos Técnicos</b>
Ninguno
<b>Desenvolvimiento del Experto</b>
<ul style="list-style-type: none"> <li>• Para una interacción más agradable con la IU de ARGON y el documento de la Lista de Verificación, el experto pidió tener en su propio computador el documento Excel en el que fue colocado en un segundo monitor. Esto le permitió al experto desenvolverse de mejor manera durante la evaluación.</li> <li>• El experto no estaba conforme con tan solo evaluar las capturas de pantalla correspondientes a cada paso del caso de uso. Esto le motivo a realizar por su propia cuenta cada paso descrito del caso de uso.</li> <li>• Tuvo una excelente participación al cumplir la técnica “Thinking Aloud”, ya que, no sé limito a mencionar lo esencial al cumplir los pasos del caso de uso, sino que mencionada las razones que motivaban a su criterio de evaluación.</li> <li>• El uso de un bloc de notas adicional, le facilito el plasmar sus criterios de evaluación mientras iba encontrando distintos problemas en la IU.</li> <li>• Mientras iba realizando la secuencia de tareas, fue notorio el profundo conocimiento que tenia de la Heurísticas de Nielsen, pues sus criterios eran manifestados después de mencionar alguna heurística en particular.</li> <li>• Preferencia en dejar expresada su criterios, comentarios y sugerencias de forma verbal ante la grabación, en vez de escribir en el apartado destinado en la Lista de Verificación.</li> </ul>

Por último, el tercer experto en realizar la Evaluación Heurística fue el Dr. Marco Santórum, cuya evaluación fue llevada a cabo a las 3:15 pm, el miércoles 27 de julio del 2022. La evaluación tomó 51 minutos. Los problemas o aspectos relevantes durante la evaluación se describen en la tabla 7.

**Tabla 7.** Problemas y Aspectos relevantes del experto Ph.D. Marco Santórum.

<b>Aspectos Técnicos</b>
<ul style="list-style-type: none"> <li>• A pesar de tener un computador con sistema operativo MacOS, pudo fácilmente escalar la pantalla de la máquina virtual, permitiéndole así una visibilidad adecuada y óptima para el experto.</li> </ul>
<b>Desenvolvimiento del Experto</b>
<ul style="list-style-type: none"> <li>• Único experto en tomarse en tiempo estipulado para leer detenidamente las instrucciones del protocolo, caso de uso y la lista de verificación.</li> <li>• Adecuada interacción con la herramienta teniendo un enfoque en evaluar los componentes de la UI de ARGON, más no al editor de Eclipse Modeling Framework.</li> <li>• Preferencia en dejar expresada su criterios, comentarios y sugerencias de forma verbal ante la grabación, en vez de escribir en el apartado destinado en la Lista de Verificación.</li> <li>• Excelente desempeño en el tiempo empleado para cada apartado, sin dejar de lado los criterios propuestos para la evaluación (protocolo).</li> <li>• No empleo tiempo adicional en seguir interactuando la herramienta, sino que se basó en el alcance propuesto con el caso de uso.</li> </ul>

Considerando la información de ejecución de cada experto, se puede concluir que los expertos tuvieron distinto desempeño al realizar la Evaluación Para algunos fue primordial el tiempo que se empleó en la evaluación, mientras que, para otros fue complicado entender el alcance de la herramienta a evaluar. El ANEXO VII. Evaluaciones Externas (enlace) contiene el enlace a los recursos recuperados de las evaluaciones, tanto la grabación como el documento Excel propuesto y llenado por los expertos. Completadas las evaluaciones externas, el siguiente paso fue recopilar los comentarios y sugerencias a fin de preparar el reporte de usabilidad esperado para la IU de ARGON.

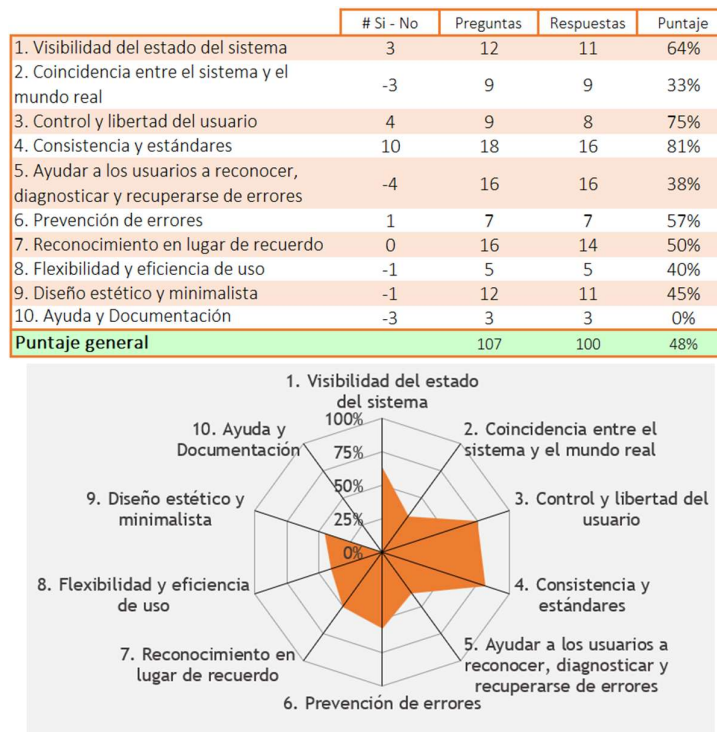
## 2.4 Preparación del Reporte

Un primer acercamiento a las respuestas obtenidas por los expertos fue definir un resumen de cumplimiento de cada heurística. Es decir, por cada experto se tiene un resumen que indica si se cumplieron o no la o las heurísticas analizadas. Cabe señalar que al final de cada evaluación externa, el resumen de cumplimiento es presentado al experto.

El resumen consiste en un gráfico autogenerado tipo radar que muestra la asignación de un porcentaje de cumplimiento a cada heurística como resultado de contar tanto las preguntas respondidas con Sí o No por el experto. Es decir, si una heurística cuenta con más respuestas positivas que negativas obtendría cierto porcentaje de

cumplimiento. El ANEXO VIII. Resumen autogenerado sobre las respuestas de la Lista de Verificación muestra la plantilla del resumen autogenerado para la Evaluación Heurística. A continuación, se detalla los resúmenes arrojados por la evaluación interna y las evaluaciones externas.

La figura 14 corresponde al resumen de las respuestas para la evaluación interna. Este resumen presenta un bajo o muy poco cumplimiento para las heurísticas 2, 5, 8, 9 y 10. Es decir que, hubo muy pocas preguntas que fueron respondidas de forma afirmativa.



**Figura 14.** Resumen de respuestas de la Evaluación Interna. Fuente: Elaboración propia.

Las figuras 15, 16 y 17 corresponden al resumen de las respuestas para las evaluaciones externas. En estos resúmenes se puede evidenciar que, los criterios de cada experto fueron bastante diferentes. No obstante, se puede evidenciar que en estas evaluaciones hubo nuevamente poco cumplimiento para las heurísticas de 5, 8 y 10. Se espera por su puesto, que este comportamiento sea el mismo después de unificar en un solo reporte (reporte final) todos los problemas y criterios obtenidos de estas evaluaciones.

	# Si - No	Preguntas	Respuestas	Puntaje
1. Visibilidad del estado del sistema	0	12	10	50%
2. Coincidencia entre el sistema y el mundo real	3	9	7	71%
3. Control y libertad del usuario	4	9	6	83%
4. Consistencia y estándares	4	18	16	63%
5. Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de errores	-3	16	3	0%
6. Prevención de errores	2	7	2	100%
7. Reconocimiento en lugar de recuerdo	1	16	9	56%
8. Flexibilidad y eficiencia de uso	-1	5	1	0%
9. Diseño estético y minimalista	3	12	9	67%
10. Ayuda y Documentación	-1	3	3	33%
<b>Puntaje general</b>		<b>107</b>	<b>66</b>	<b>52%</b>



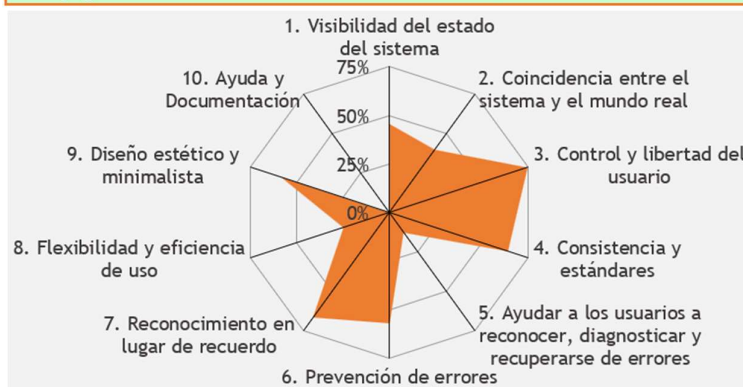
Figura 15. Resumen de respuestas de la experta MsC. Mayra Carrión. Fuente: Elaboración propia.

	# Si - No	Preguntas	Respuestas	Puntaje
1. Visibilidad del estado del sistema	-1	12	11	45%
2. Coincidencia entre el sistema y el mundo real	-5	9	7	14%
3. Control y libertad del usuario	-3	9	7	29%
4. Consistencia y estándares	7	18	11	82%
5. Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de errores	-9	16	15	20%
6. Prevención de errores	-6	7	6	0%
7. Reconocimiento en lugar de recuerdo	1	16	15	53%
8. Flexibilidad y eficiencia de uso	-5	5	5	0%
9. Diseño estético y minimalista	4	12	12	67%
10. Ayuda y Documentación	-3	3	3	0%
<b>Puntaje general</b>		<b>107</b>	<b>92</b>	<b>31%</b>



Figura 16. Resumen de respuestas del experto PhD. Julian Galindo. Fuente: Elaboración propia.

	# Si - No	Preguntas	Respuestas	Puntaje
1. Visibilidad del estado del sistema	-1	12	11	45%
2. Coincidencia entre el sistema y el mundo real	-1	9	5	40%
3. Control y libertad del usuario	4	9	8	75%
4. Consistencia y estándares	4	18	14	64%
5. Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de errores	-12	16	16	13%
6. Prevención de errores	1	7	7	57%
7. Reconocimiento en lugar de recuerdo	5	16	15	67%
8. Flexibilidad y eficiencia de uso	-2	5	4	25%
9. Diseño estético y minimalista	2	12	12	58%
10. Ayuda y Documentación	-3	3	3	0%
<b>Puntaje general</b>		<b>107</b>	<b>95</b>	<b>44%</b>



**Figura 17.** Resumen de respuestas del experto Dr. Marco Santorum. Fuente: Elaboración propia.

Después de haber analizado los resúmenes, se puede esperar que de las heurísticas con poco cumplimiento se obtenga mayor cantidad de problemas. A fin de descubrir estos problemas, se analiza detenidamente las grabaciones y se recopilan los criterios y sugerencias realizadas por los expertos.

### 2.4.1 Recopilación de problemas de usabilidad

Para la recopilación de los problemas, se reúne, en un solo documento Excel, las respuestas obtenidas de las evaluaciones, añadiendo en la parte de comentarios y sugerencias los criterios expuestos por cada experto en las grabaciones. El ANEXO IX. Recopilación de problemas de usabilidad (enlace) muestra los problemas de usabilidad encontrados en las evaluaciones realizadas. A continuación, se asigna un puntaje de severidad a cada problema encontrado.

## **2.4.2 Puntaje de severidad y soluciones a problemas**

Para la asignación de puntaje de severidad, se tomó en cuenta la escala de puntuación propuesta por Jakob Nielsen [28]. La escala de puntuación consiste en asignar un número de 0 a 4 cuyos significados son los siguientes:

- 0:** No se considera un problema de usabilidad.
- 1:** Problema superficial, si hay tiempo adicional es posible solucionarlo.
- 2:** Problema menor, su solución tiene baja prioridad.
- 3:** Problema mayor, es importante corregirlo y tiene alta prioridad.
- 4:** Problema catastrófico, se debe corregir lo más antes posible.

La asignación de severidad no pudo ser realizada por cada experto debido a su limitado tiempo para la evaluación heurística. Es por eso por lo que, la asignación de severidad se realizó considerando los resultados de cada evaluación y observando los comentarios registrados en las grabaciones. El ANEXO X. Asignación de Severidad a los problemas de usabilidad muestra la severidad asignada a los problemas de usabilidad encontrados.

## **2.4.3 Elaboración del reporte de hallazgos**

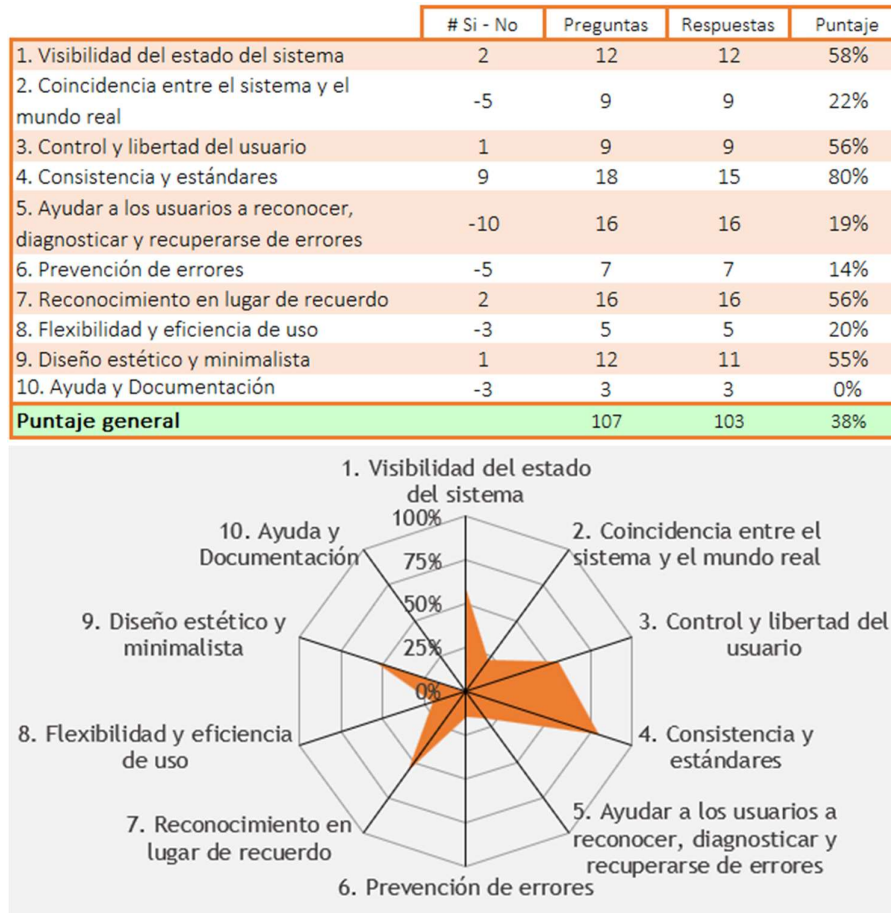
El reporte de hallazgos se presenta en forma de infografía que plasma el resultado de problemas de usabilidad encontrados. La infografía se encuentra en el ANEXO XI. Infografía del reporte de hallazgos (enlace opcional). Y enfatiza, con capturas de pantalla, los problemas de usabilidad que registran un alto puntaje de severidad. Dichos problemas son los que tienen mayor puntaje de severidad y resultan evidentes a simple vista de los expertos.

# **3 RESULTADOS, CONCLUSIONES Y RECOMENDACIONES**

## **3.1 Resultados**

Muchos de los resultados de cada apartado de la metodología se han ido mencionando y adjuntando en los anexos, tablas y figuras correspondientes. Sin embargo, lo relevante de este análisis heurístico es comprender si la IU del editor gráfico de

ARGON cumple con las heurísticas elegidas. Por ello, el resumen de las evaluaciones realizadas se convierte en la forma cuantitativa y visual para entender si la IU de ARGON tiene o no problemas en su usabilidad. La figura 17 indica el resultado obtenido de las evaluaciones (externas e interna).



**Figura 18.** Resumen del recopilado de respuestas de las Evaluaciones Heurísticas. Fuente: Elaboración propia.

A diferencia de las figuras 15, 16 y 17, la figura 18 indica mayor cantidad de respuestas contempladas en la evaluación. Logrando comprobar que, al tener una mayor cantidad de evaluadores se consigue un amplio hallazgo de los problemas de usabilidad presentes en la IU de ARGON. Esto permitió tener una gráfica más acertada sobre el cumplimiento de las Heurística de Usabilidad propuestas por Jakob Nielsen. En dicha grafica se aprecia que no existe cumplimiento alguno de la heurística #10 “Ayuda y Documentación”. Además, existe muy poco cumplimiento de las siguientes heurísticas:

- “2. Consistencia entre el sistema y el mundo real”
- “5. Ayuda a reconocer, diagnosticar y recuperarse de errores”
- “6. Prevención de errores”

- “8. Flexibilidad y eficiencia de uso”

Con ello, se puede deducir que, cualquier tipo de usuario que cumpla con el perfil de Francisco (Perfil Usuario) está propenso a cometer errores en la IU, sin tener la posibilidad de recuperarse de ellos. Esto potencialmente frustrará al usuario limitando el uso que este tenga con la herramienta ARGON. Ya que, según los resultados arrojados, el usuario no será capaz de entender el funcionamiento ni el significado de los elementos de un modelo. Esto convertiría a la tarea de modelado de infraestructura en una tarea frustrante que requiera demasiada concentración por parte del usuario. Es decir, la IU de ARGON no resulta amigable al usuario y más bien, crea impedimentos al usuario para lograr concretar el objetivo de modelar un balanceador de carga para AWS.

### **3.2 Conclusiones**

En esta sección se presentan las conclusiones del proyecto las cuales se describen en función de cada uno de los objetivos perseguidos en el proyecto:

- Respecto al objetivo 1: “Detectar problemas de usabilidad en la interfaz de usuario de la herramienta de aprovisionamiento de infraestructura ARGON, considerando el caso de uso: balanceo de carga”, se realizó el análisis de la secuencia de tareas de un caso de uso previamente preparado: “Balance de carga”. El caso de uso animó al experto a interactuar directamente con la IU de ARGON, permitiéndole realizar hallazgos de diversos problemas de usabilidad. 63 problemas de usabilidad en la interfaz de usuario de la herramienta de ARGON fueron reportados.
- Respecto al objetivo 2: “Proponer soluciones de diseño tomando en cuenta los principios de usabilidad de Jacob Nielsen”, se recopiló los comentarios y sugerencias que dieron los expertos durante las evaluaciones; lo cual permitió proponer posibles soluciones ante los problemas encontrados. Para ello, los expertos prefirieron registrar sus comentarios y sugerencias en forma verbal (Thinkig Aloud), esto gracias a que se realizó la grabación de cada sesión. A partir del análisis de las grabaciones, se obtuvo todo tipo de criterio y sugerencias para la solución a los problemas encontrados.
- Respecto al objetivo 3: “Establecer niveles de prioridad a los problemas de usabilidad encontrados como aporte al diseño/rediseño de la interfaz de usuario”,



se asignó puntajes de severidad a los problemas de usabilidad encontrados; a mayor nivel de severidad, mayor impacto en la Experiencia de Uso. Esto ayudará a enfatizar la solución de los problemas, en caso de existir futuros diseños a implementar sobre la IU de ARGON como aplicativo web.

El logro de los objetivos específicos descritos conlleva al logro del objetivo principal del proyecto: “Realizar una evaluación heurística de la interfaz gráfica de la herramienta de aprovisionamiento de infraestructura ARGON, para el caso de uso de balanceo de carga”. Es decir, en este Trabajo de Integración Curricular se realizó una evaluación heurística tomando en cuenta el criterio de 3 expertos en Usabilidad. Para ello, se definió un caso de uso, Balanceo de Carga, que a partir de su secuencia de tareas se generaron 2 tipos de evaluaciones: la primera, la evaluación interna, fue realizada por mi persona, Cristhian Muñoz y la segunda, la evaluación externa, fue realizada por los expertos en Usabilidad. Una vez realizadas las evaluaciones, se recopilaron las respuestas en un solo reporte, logrando plasmar en una infografía 11 de los 63 problemas de usabilidad encontrados que presenta la Herramienta de ARGON.

### **3.3 Recomendaciones**

- Para llevar a cabo las evaluaciones, se recomienda conocer a profundidad los aplicativos con los que la evaluación se llevará a cabo, que para este trabajo fue AnyDesk. Al desconocer el funcionamiento de este aplicativo para el sistema operativo MacOS, se produjo inconvenientes con un experto durante la evaluación, limitando la visibilidad que tenía el experto hacia la máquina virtual.
- En este trabajo se observó que, a determinados expertos les resultó esencial intentar realizar la secuencia de tareas del caso de uso. Por esto, se recomienda permitir al experto llevar a cabo el caso de uso en la herramienta a evaluar, siempre y cuando exista cierta flexibilidad en el tiempo propuesto para la evaluación.
- En caso de que existan herramientas cuyo funcionamiento dependa de otras, que en el caso de ARGON depende de Eclipse Modeling Framework, y se desee un análisis heurístico, se recomienda dejar claro qué parte de todo lo que se evaluará corresponde al aplicativo a evaluar. Esto debido a que, durante la evaluación con los expertos se manifestó la duda sobre el punto de vista que se debía tener para la IU de ARGON, es decir, los expertos tenían dudas sobre evaluar características propias del IDE Eclipse.

- En caso de que los futuros diseños de ARGON contemplen algún aplicativo como referencia, se recomienda tomar en cuenta las IU de aplicativos como Neatbeans o .Net. Estos aplicativos han sido elegidos por muchos usuarios por su eficiencia y comodidad de uso, constatando dicho criterio por uno de los expertos que formó parte de las evaluaciones.
- Se evidenció que, el papel que jugaban los evaluadores como expertos en usabilidad, hizo que estos se confiaran por su conocimiento del área. Esto provocó poco interés en los documentos informativos, llegando en algunos casos hasta contestar las preguntas de evaluación sin tener presente a Francisco (ficha persona) para sus criterios. En este sentido, se recomienda que el facilitador de la evaluación responda inmediatamente las preguntas que del experto emerjan durante la evaluación.

## 4 REFERENCIAS BIBLIOGRÁFICAS

- [1] J. Sandobalin, E. Insfran y S. Abrahao, «An Infrastructure Modelling Tool for Cloud Provisioning,» 14 Septiembre 2017. [En línea]. Available: [https://www.researchgate.net/publication/316701858\\_An\\_Infrastructure\\_Modelling\\_Tool\\_for\\_Cloud\\_Provisioning](https://www.researchgate.net/publication/316701858_An_Infrastructure_Modelling_Tool_for_Cloud_Provisioning). [Último acceso: 12 Diciembre 2021].
- [2] M. Marjan, J. Heering y A. Sloane, When and how to develop domain-specific languages., vol. 37, 2005, pp. 316-344.
- [3] «MySQL Workbench,» [En línea]. Available: <https://www.mysql.com/products/workbench/>. [Último acceso: 13 Enero 2022].
- [4] R. Walsh, «cabbage,» [En línea]. Available: <https://cabbageaudio.com/>. [Último acceso: 1 Febrero 2020].
- [5] R. Unger y C. Chander, «What is User Experience Design,» de *A project guide to UX Design for user experience designer in the field or in the making*, Bekeley, New Riders, 2021, p. 23.
- [6] E. Foundation, «Eclipse,» [En línea]. Available: <https://www.eclipse.org/ide/>. [Último acceso: 26 Enero 2020].
- [7] F5, «What Is Load Balancing?,» NGINX, 2022. [En línea]. Available: <https://www.nginx.com/resources/glossary/load-balancing/>. [Último acceso: 20

Mayo 2022].

- [8] Citrix Systems, «What is load balancing?,» Citrix, 2022. [En línea]. Available: <https://www.citrix.com/solutions/app-delivery-and-security/load-balancing/what-is-load-balancing.html>. [Último acceso: 20 Mayo 2022].
- [9] AXELOS, «ITIL 4 edition, Glossary,» Enero 2019. [En línea]. Available: <https://purplegriffon.com/downloads/resources/itil4-foundation-glossary-january-2019.pdf>. [Último acceso: 25 Mayo 2022].
- [10] IBM, «¿Qué es infraestructura de TI?,» IBM, 2022. [En línea]. Available: <https://www.ibm.com/mx-es/topics/infrastructure>. [Último acceso: 25 Mayo 2022].
- [11] J. Sandobalin, E. Insfran y S. Abrahao, «Argon: A model-driven infrastructure provisioning tool,» *ACM/IEEE 22nd International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*, pp. 738-742, 2019.
- [12] J. Sandobalin, E. Insfran y S. Abrahao, *End-to-end automation in cloud infrastructure provisioning*, 2017.
- [13] G. Wilbert, *The Essential Guide to User Interface Design. An Introduction to GUI Design Principles and Techniques*, Indianapolis: Wiley Publishing, Inc., 2007.
- [14] W3Schools, «What is Command Line Interface (CLI)?,» [En línea]. Available: [https://www.w3schools.com/whatis/whatis\\_cli.asp](https://www.w3schools.com/whatis/whatis_cli.asp). [Último acceso: 2 Junio 2022].
- [15] P. Fischer, *An Introduction Graphical User Interfaces with Java Swing*, England: Pearson Education Limited, 2005.
- [16] B. Shneiderman, C. Plaisant, M. Cohen, S. Jacobs, N. Elmqvist y N. Diakopoulos, *DESIGNING THE USER INTERFACE*, EdinBurgh: Pearson Education Limited, 2018, p. 172.
- [17] H. Sharp, Y. Rogers y J. Preece, *Interaction Design: beyond human-computer interaction*, Fifth Edition, Indianapolis: John Wiley & Sons, Inc., 2019.
- [18] R. Unger y C. Candler, *A Project Guide to UX Design*, Second Edition, Berkeley: New Riders, 2012.
- [19] No Solo Usabilidad, «3. Diseño Centrado en el Usuario (DCU),» 2020. [En línea]. Available: <https://www.nosolousabilidad.com/manual/3.htm>. [Último acceso: 5 Agosto 8].

- [20] A. Cooper, R. Reimann, D. Cronin, C. Noessel, J. Csizmadi y D. LeMoine, *About Face, The Essentials of Interaction Design*, Indianapolis: John Wiley & Sons, 2014, pp. 61-72.
- [21] K. O'Connor, «Personas: The Foundation of a Great User Experience,» 25 Marzo 2011. [En línea]. Available: <https://uxmag.com/articles/personas-the-foundation-of-a-great-user-experience>. [Último acceso: 6 Junio 2022].
- [22] J. Nielsen, «Usability 101: Introduction to Usability,» Nielsen Norman Group, 2012 Enero 3. [En línea]. Available: <https://www.nngroup.com/articles/usability-101-introduction-to-usability/>. [Último acceso: 4 Junio 2022].
- [23] J. Nielsen, «10 Usability Heuristics for User Interface Design,» 15 Noviembre 2020. [En línea]. Available: <https://www.nngroup.com/articles/ten-usability-heuristics/>. [Último acceso: 19 Junio 2022].
- [24] J. Nielsen, «How to Conduct a Heuristic Evaluation,» 1 Noviembre 1994. [En línea]. Available: <https://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/>. [Último acceso: 19 Junio 2022].
- [25] J. Sandobalin, «Load Balancer Provisioning in Amazon Web Services,» 26 Junio 2019. [En línea]. Available: <https://vimeo.com/344519833>. [Último acceso: 1 Julio 2022].
- [26] D. Pierotti, «Heuristic evaluation-a system checklist,» Xerox Corporation, 1995.
- [27] J. Nielsen, «Thinking Aloud: The #1 Usability Tool,» 15 Enero 2012. [En línea]. Available: <https://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool/>. [Último acceso: 29 Julio 2022].
- [28] J. Nielsen, «Severity Ratings for Usability Problems,» 1 Noviembre 1994. [En línea]. Available: <https://www.nngroup.com/articles/how-to-rate-the-severity-of-usability-problems/>. [Último acceso: 30 Julio 2022].
- [29] Red Hat, «¿Qué es la infraestructura hiperconvergente?,» 27 Febrero 2020. [En línea]. Available: <https://www.redhat.com/es/topics/hyperconverged-infraestructure/what-is-hyperconverged-infraestructure>. [Último acceso: 25 Mayo 2020].
- [30] Amazon Web Services, «Types of Cloud Computing,» 2022. [En línea]. Available: <https://aws.amazon.com/es/types-of-cloud-computing/>. [Último acceso: 25 Mayo 2022].

## 5 ANEXOS

### ANEXO I. Formulario para levantar Perfil Usuario

# Modelamiento de Infraestructura en la Nube

Como estudiante de la carrera de Software/Computación solicitamos de su ayuda para el levantamiento e identificación de usuarios objetivos, en el uso de cierta herramienta enfocada al modelamiento de Infraestructura en la nube.

Section 1 ...

## Información Personal

**1** ¿Cuál es su género? \*

Masculino

Femenino

**2** Indique su rango de edad \*

menor a 21 años

21 - 23 años

23 - 25 años

mayor a 25 años

**3** Indique la carrera que actualmente está cursando \*

Software

Computación

4

¿Cuál es el semestre que está cursando actualmente? \*

- Cuarto
- Quinto
- Sexto
- Séptimo
- Octavo
- Noveno

5

¿Qué área(s) de la informática le interesa más? \*

- Desarrollo FrontEnd
- Desarrollo BackEnd
- Infraestructura Tecnológica
- Seguridad
- Diseño UX
- Other

Section 2

...

### Infraestructura de (Tecnologías de la Información) TI



6

¿Cree usted que modelar la Infraestructura TI de una empresa es complejo? \*

- Si
- No

7

Describa en un párrafo corto, ¿por qué cree que es complejo modelar la Infraestructura TI? \*

Enter your answer

8

¿Ha diseñado/modelado una Infraestructura TI? (incluye proyectos pequeños o medianos hechos en clases de Redes) \*

- Si
- No

9

¿Qué tipo de casos ha modelado? \*

- Servidores
- Balanceo de Carga
- Escalamiento
- Other

10

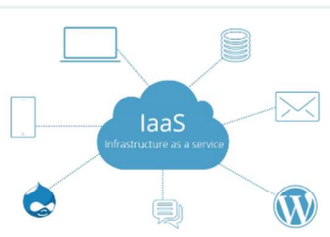
¿Con qué aspecto cree usted que tiene o tendría problemas al diseñar e implementar una Infraestructura TI? \*

- Identificar elementos de infraestructura
- Esquematización lógica
- Configuración
- Levantamiento de Servicios
- Other

Section 3

...

Infraestructura como Servicio



11

¿Conoce el concepto de "Infraestructura como Servicio" (IaaS)? \*

- Si
- No

12

¿Conoce o ha escuchado acerca del término "aprovisionamiento de infraestructura" en el contexto de la informática ? \*

- Si
- No

13

¿De qué Proveedor de Infraestructura tiene mayor conocimiento? \*

- AWS
- Azure
- Google Cloud Platform
- Otro
- Ninguno

14

¿Qué problemas ha encontrado al utilizar los servicios de los Proveedores de Infraestructura? \*

- Dependencia del proveedor
- Diferentes formas para realizar la misma actividad en cada proveedor
- Necesidad de acceso online
- Other

15

¿Estaría dispuesto a aprender a gestionar la Infraestructura de TI utilizando scripts de código? \*

- Si
- No

Section 4

...

Editores e Interfaz de Usuario





16

¿Qué editores gráficos ha utilizado para diagramar y/o entender visualmente una infraestructura TI? \*

Lucid Chart

Visio

Draw.io

Other

17

Describa la utilidad que le aporta usar el o los editores gráficos seleccionados \*

Enter your answer

18

¿Qué tipo de interfaz de usuario le haría sentir más cómodo para gestionar la infraestructura de TI en la nube? \*

Consola de Línea de Comandos (CLI)

Editor visual gráfico como aplicación de escritorio

Editor visual gráfico como aplicación Web

Other

19

¿Qué dispositivo(s) electrónico(s) usted utilizaría para diseñar e implementar una infraestructura TI? \*

Laptop

Teléfono Móvil

Tableta

+ Add new

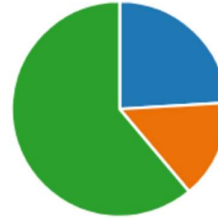
## ANEXO II. Resultados de la encuesta para Perfil Persona

### 1. ¿Cuál es su género?

[More Details](#)

[Insights](#)

<span style="color: blue;">●</span> Masculino	13
<span style="color: orange;">●</span> Femenino	8
<span style="color: green;">●</span> Other	33



### 2. Indique su rango de edad

[More Details](#)

[Insights](#)

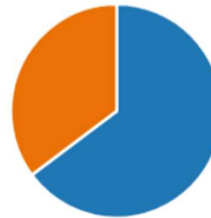
<span style="color: blue;">●</span> menor a 21 años	0
<span style="color: orange;">●</span> 21 - 23 años	32
<span style="color: green;">●</span> 23 - 25 años	17
<span style="color: red;">●</span> mayor a 25 años	5



### 3. Indique la carrera que actualmente está cursando

[More Details](#)

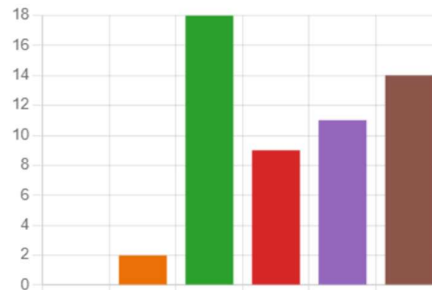
<span style="color: blue;">●</span> Software	35
<span style="color: orange;">●</span> Computación	19



### 4. ¿Cuál es el semestre que está cursando actualmente?

[More Details](#)

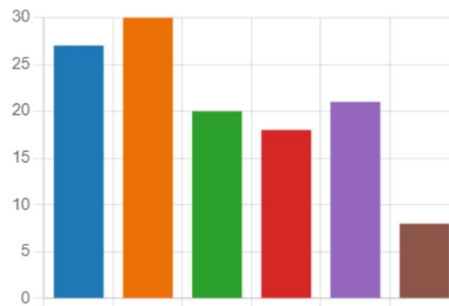
<span style="color: blue;">●</span> Cuarto	0
<span style="color: orange;">●</span> Quinto	2
<span style="color: green;">●</span> Sexto	18
<span style="color: red;">●</span> Séptimo	9
<span style="color: purple;">●</span> Octavo	11
<span style="color: brown;">●</span> Noveno	14



5. ¿Qué área(s) de la informática le interesa más?

[More Details](#)

● Desarrollo FrontEnd	27
● Desarrollo BackEnd	30
● Infraestructura Tecnológica	20
● Seguridad	18
● Diseño UX	21
● Other	8



6. ¿Cree usted que modelar la Infraestructura TI de una empresa es complejo?

[More Details](#)

[Insights](#)

● Si	34
● No	20



7. Describa en un párrafo corto, ¿por qué cree que es complejo modelar la Infraestructura TI?

[More Details](#)

[Insights](#)

34  
Responses

Latest Responses

"Se necesita tener un conocimiento amplio del tema y una organización inte..."  
 "Debo tener muchos conocimientos para implementar una infraestructura"

8 respondents (24%) answered **cuenta** for this question.



8. ¿Ha diseñado/modelado una Infraestructura TI? (incluye proyectos pequeños o medianos hechos en clases de Redes)

[More Details](#)

[Insights](#)

● Si	29
● No	25



9. ¿Qué tipo de casos ha modelado?

[More Details](#)

[Insights](#)

● Servidores	25
● Balanceo de Carga	3
● Escalamiento	1
● Other	0



10. ¿Con qué aspecto cree usted que tiene o tendría problemas al diseñar e implementar una Infraestructura TI?

[More Details](#)

● Identificar elementos de infraest...	21
● Esquematización lógica	35
● Configuración	22
● Levantamiento de Servicios	24
● Other	0

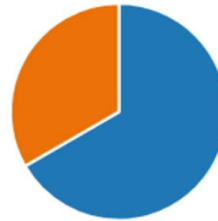


11. ¿Conoce el concepto de "Infraestructura como Servicio" (IaaS)?

[More Details](#)

[Insights](#)

● Si	36
● No	18



12. ¿Conoce o ha escuchado acerca del término "aprovisionamiento de infraestructura" en el contexto de la informática ?

[More Details](#)

[Insights](#)

● Si	21
● No	33



13. ¿De qué Proveedor de Infraestructura tiene mayor conocimiento?

[More Details](#)

● AWS	28
● Azure	21
● Google Cloud Plataform	16
● Otro	1
● Ninguno	11



14. ¿Qué problemas ha encontrado al utilizar los servicios de los Proveedores de Infraestructura?

[More Details](#)

● Dependencia del proveedor	28
● Diferentes formas para realizar l...	16
● Necesidad de acceso online	17
● Other	4



15. ¿Estaría dispuesto a aprender a gestionar la Infraestructura de TI utilizando scripts de código?

[More Details](#)

● Si	48
● No	6



16. ¿Qué editores gráficos ha utilizado para diagramar y/o entender visualmente una infraestructura TI?

[More Details](#)

● Lucid Chart	39
● Visio	18
● Draw .io	29
● Other	8



17. Describa la utilidad que le aporta usar el o los editores gráficos seleccionados

[More Details](#)

[Insights](#)

54  
Responses

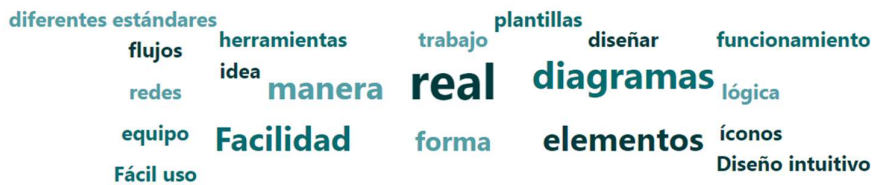
Latest Responses

"Todo lo gráfico es más sencillo de explicar que palabras."

"Facilita el uso y entendimiento del usuario"

"Ayuda a entender la infraestructura"

9 respondents (17%) answered **real** for this question.



## <<--DESCRIPCIÓN DE LA TAREA-->>

### **Aprovisionamiento de un Balanceador de Carga en Amazon Web Services (AWS)**

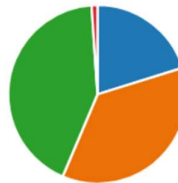
Balancear la carga es una actividad frecuente en el aprovisionamiento de infraestructura. Esta consiste en alivianar la cantidad de peticiones realizadas a un servidor, distribuyendo dichas peticiones entre varios servidores.

A continuación, se muestra la ficha de una persona que persigue como objetivo balancear la carga de un servidor.

18. ¿Qué tipo de interfaz de usuario le haría sentir más cómodo para gestionar la infraestructura de TI en la nube?

[More Details](#)

● Consola de Línea de Comandos ...	19
● Editor visual gráfico como aplica...	34
● Editor visual gráfico como aplica...	40
● Other	1



19. ¿Qué dispositivo(s) electrónico(s) usted utilizaría para diseñar e implementar una infraestructura TI?

[More Details](#)

● Laptop	54
● Teléfono Móvil	8
● Tableta	8



## **ANEXO III. Detalle del Caso de Uso (Documento de Recorrido Cognitivo)**

## FICHA PERSONA



**Francisco Luis Sarzosa Vega**

**Edad:** 22 años

**Género:** Masculino

**Semestre:** Sexto

**Carrera:** Ing. Software

### DESCRIPCIÓN

Me enfoco en el desarrollo de BackEnd para Aplicaciones Web, algunas son alojadas en un servidor en la nube. Antes de alojar una aplicación, diseño la infraestructura TI requerida. Por ello, generalmente utilizo editores gráficos para diseñar un modelo de infraestructura ajustado a las necesidades de mi aplicación. Con la vista global del modelo, me es fácil entender dicha infraestructura para posteriormente implementarla en la nube.

### OBJETIVO

- Implementar en AWS, un balanceador de carga para distribuir las peticiones de una App Web alojada en AWS.

### COMPLICACIONES / FRUSTACIONES

- Modelar Infraestructura TI me es complejo por:
  - Tengo que considerar muchos criterios:
    - Presupuesto
    - Seguridad, interconexión
    - Conocimiento técnico relacionado a cada dispositivo
  - No comprendo su esquema lógico sino tengo una vista global del modelo.
- Me frustra aprender las peculiaridades que tiene cada proveedor de servicios en la nube (AWS & Azure).

### MOTIVACIONES

- Utilizar un editor gráfico en la web para gestionar infraestructura TI, evitando aprender scripts específicos de código.

### TECNOLOGÍAS

- Modelado de Infraestructura, enfocado a servidores.
- Aplicaciones Web para diagramar
  - Lucid Chart
  - Draw.io
- Uso de proveedores de servicios
  - AWS
  - Azure
- Uso de Dispositivos
  - Laptop (77%)
  - Móvil (11%)

## >> SECUENCIA DE TAREAS

Para lograr el objetivo planteado, Francisco realiza 8 tareas usando ARGON como herramienta. Estas tareas se describen a continuación:

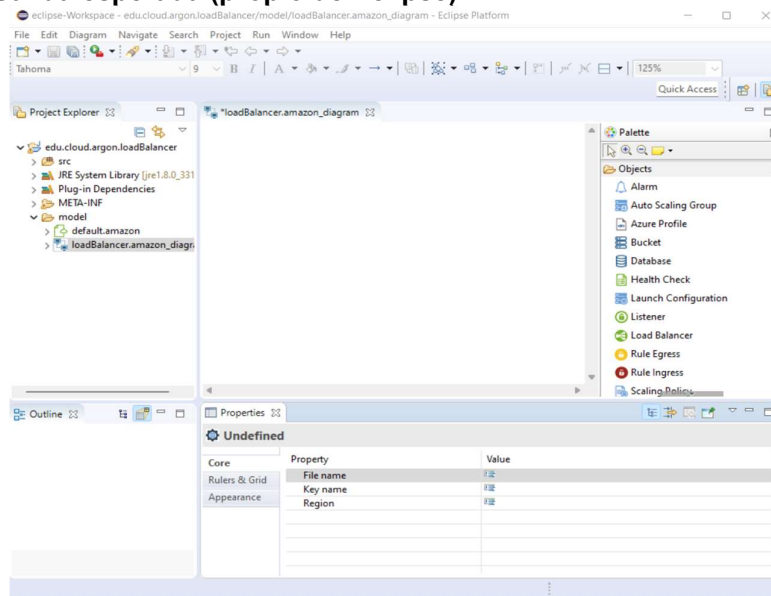
1. **Crear un diagrama de infraestructura para AWS (propio de Eclipse).**
2. **Completar las propiedades para el diagrama de infraestructura (propio de Eclipse).**
3. Modelar un balanceador de carga y sus respectivas máquinas virtuales.
4. Modelar un chequeo de salud (health check) para el balanceador de carga.
5. Modelar un oyente (listener) para el balanceador de carga.
6. Modelar y delimitar zonas para:
  - Balanceador de Carga.
  - Máquinas Virtuales
7. Modelar grupos de seguridad con sus respectivas reglas de entradas y salidas para:
  - Balanceador de Carga.
  - Máquinas Virtuales
8. Generar el Script de la Infraestructura para Ansible.

## EJECUCIÓN → TAREAS

### Paso 1

#### X Crear un diagrama de infraestructura para AWS

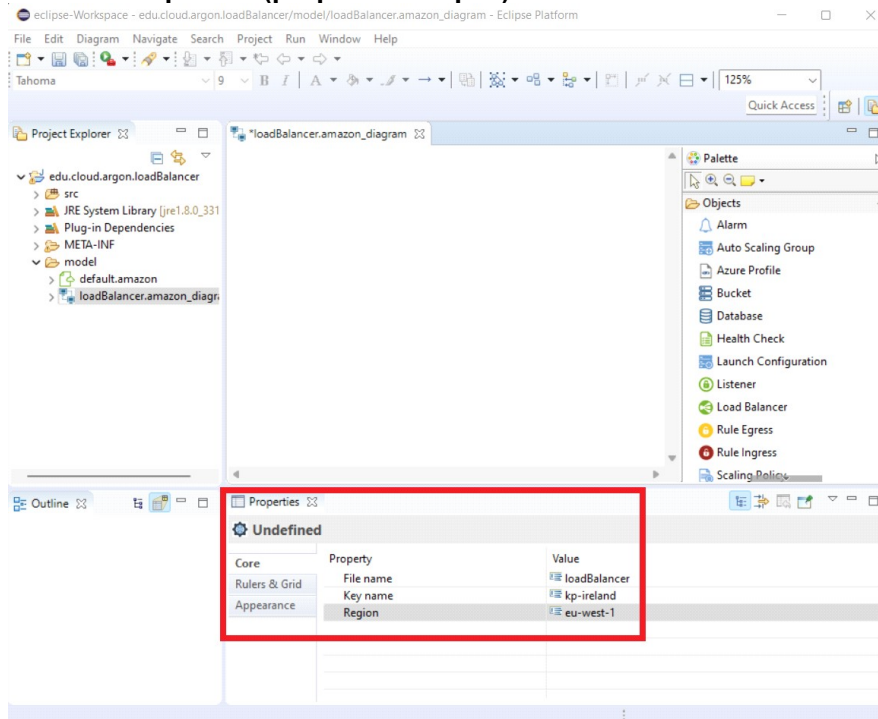
- Salida esperada (propio de Eclipse)



### Paso 2

#### X Completar las propiedades para el diagrama de infraestructura

- Salida esperada (propio de Eclipse)

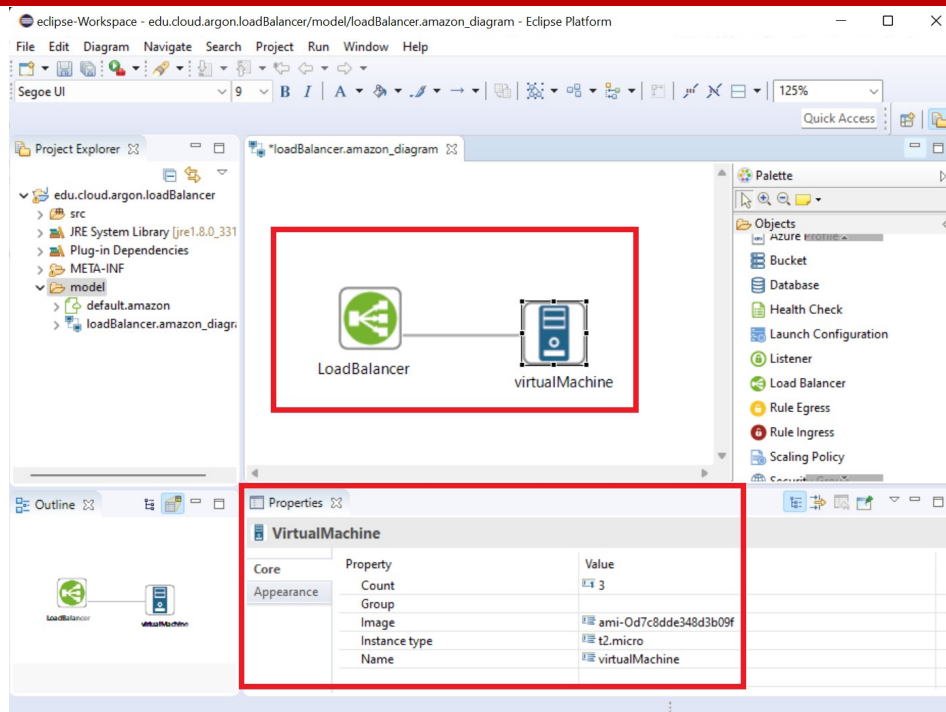


### Paso 3

#### ✓ Modelar un balanceador de carga y sus respectivas máquinas virtuales

- Salida esperada

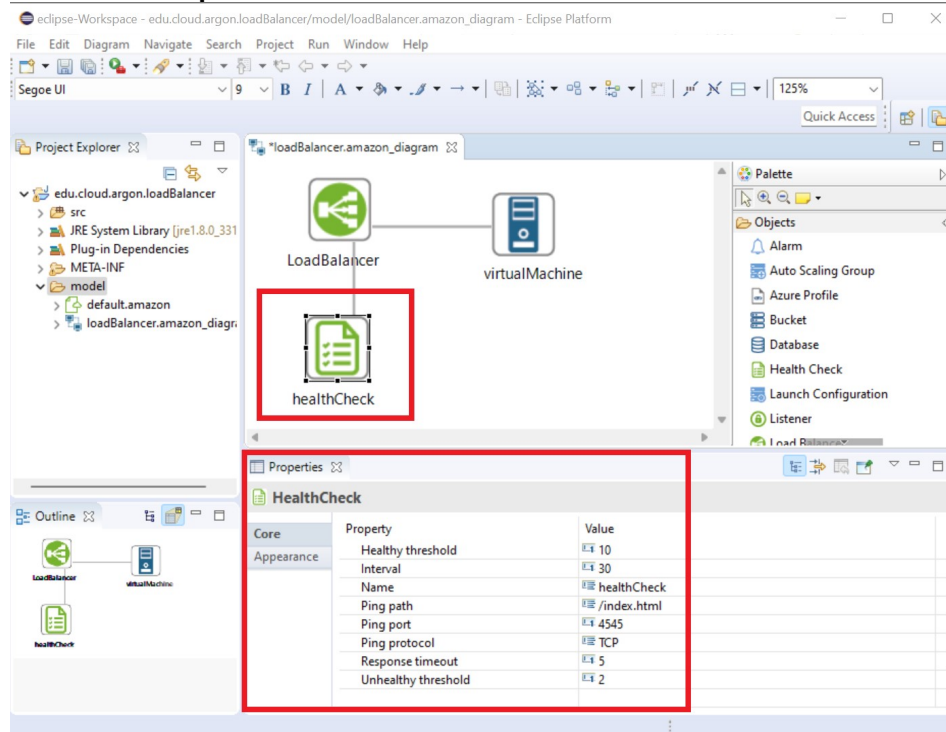




#### Paso 4

✓ Modelar un chequeo de salud (health check) para el balanceador de carga

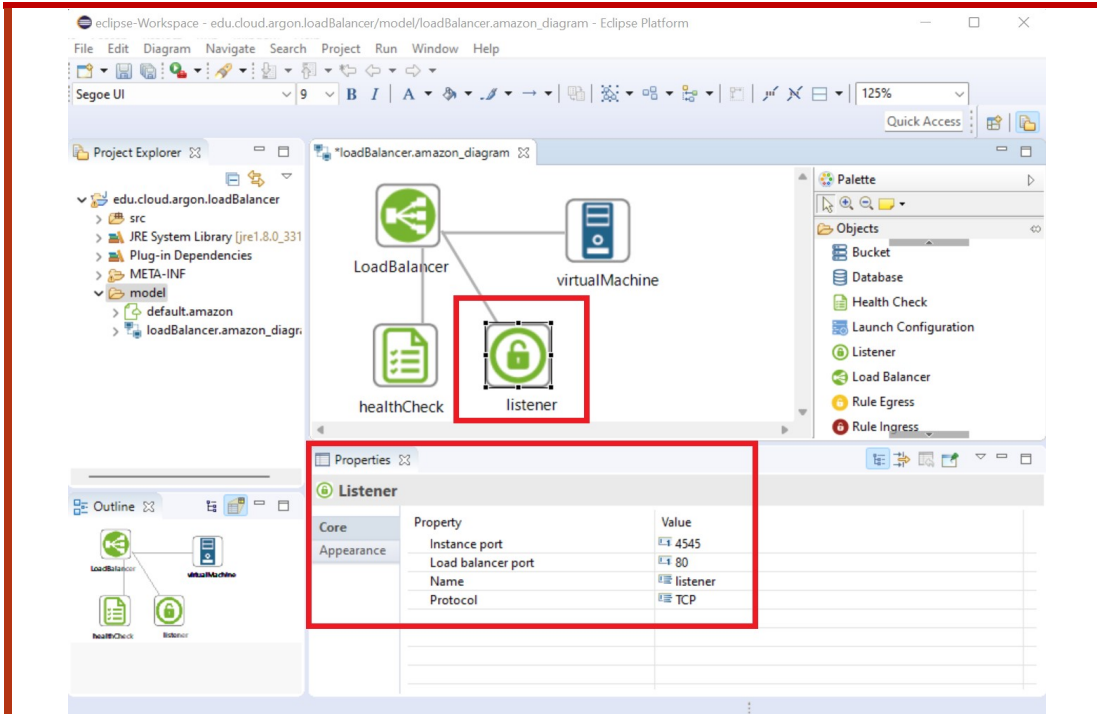
- Salida esperada



#### Paso 5

✓ Modelar un oyente (listener) para el balanceador de carga

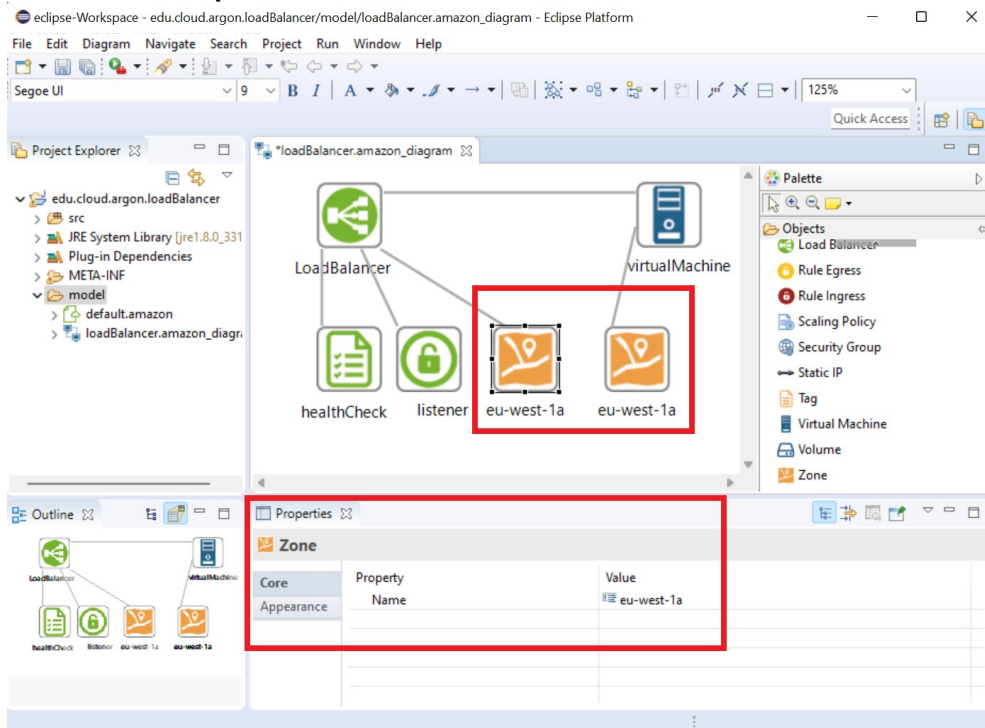
- Salida esperada



### Paso 6

✓ Modelar y delimitar zonas para: **Balancedador de Carga & Máquinas Virtuales**

- Salida esperada



### Paso 7

✓ Modelar grupos de seguridad con sus respectivas reglas de entradas y salidas para:

## Balancedador de Carga & Máquinas Virtuales

- Salida esperada

The screenshot shows the Eclipse IDE interface with a project named 'edu.cloud.argon.loadBalancer'. The main diagram area displays a 'LoadBalancer' component connected to a 'virtualMachine' component. Two security groups are associated with the load balancer: 'sgpLoadBalancer' and 'sgpVirtualMachine'. The 'sgpLoadBalancer' security group is highlighted with a red box, and its properties are shown in the Properties view below. The 'sgpVirtualMachine' security group is also highlighted with a red box. The Properties view for 'sgpLoadBalancer' is as follows:

Core	Property	Value
Appearance	Description	Security group for load balancer
	Name	sgpLoadBalancer

The detailed view shows two security group rules for the 'sgpLoadBalancer' security group. The 'Outbound' rule is highlighted with a red box, and its properties are shown in the Properties view below. The 'Inbound' rule is also highlighted with a red box, and its properties are shown in the Properties view below.

Core	Property	Value
Appearance	Cidr Ip	0.0.0.0/0
	From Port	0
	Group name	Port_ALL
	Name	Port_ALL
	Protocol	ALL
	To Port	0

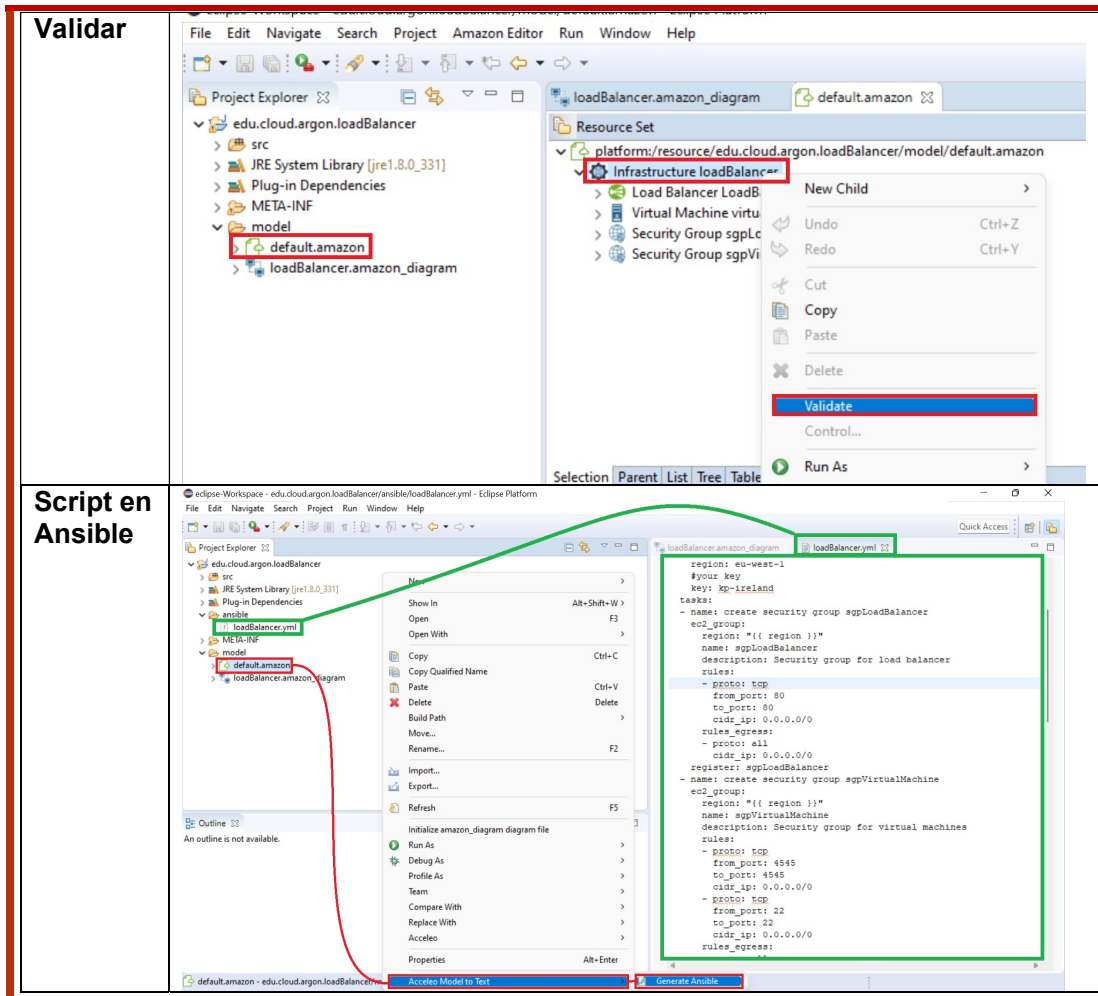
  

Core	Property	Value
Appearance	Cidr Ip	0.0.0.0/0
	From Port	80
	Group name	Port_80
	Name	Port_80
	Protocol	TCP
	To Port	80

### Paso 8

#### ✓ Generar el Script de la Infraestructura para Ansible

- Salida esperada



## ANEXO IV. Lista de Verificación de las Heurísticas de Nielsen

Lista de Verificación		Sí / No / No aplica	Comentarios & Sugerencias
<b>1. Visibilidad del estado del sistema</b>			
1,1	Quando se selecciona un icono / elemento, ¿se resalta de otros elementos que no están seleccionados?	Sí	
1,2	¿Todas las pantallas comienzan con un título o encabezado que describe el contenido de la pantalla?	No	
1,3	¿No son necesarios altos niveles de concentración o se requiere recordar información?	No aplica :(	

1,4	¿Hay señales visuales sobresalientes para identificar la selección activa?	-	
1,5	Cuando se hace clic en un botón de acción o se completa un proceso, ¿el sistema proporciona una retroalimentación al usuario?	-	
1,6	¿Se notifica al usuario cuando un grupo de acciones se completó con éxito e indica el siguiente conjunto de acciones a realizar para completar la tarea/lograr el objetivo?	-	
1,7	¿Los elementos muestran claramente en cuáles se puede hacer clic y en cuáles no?	-	
1,8	Si existen menús, ¿hay información visual sobre qué opciones ya están seleccionadas?	-	
1,9	¿Hay retroalimentación visual (cambia la forma o el color) cuando se seleccionan o se mueven los íconos/elementos?	-	
1,10	¿Se indica claramente el estado actual de un ícono/elemento?	-	
1,11	¿El sistema proporciona una retroalimentación cuando se hace clic en las acciones principales?	-	
1,12	Al mirar, ¿el usuario puede saber el estado del sistema y las alternativas de acción?	-	
<b>2. Coincidencia entre el sistema y el mundo real</b>			
2,1	¿Son los íconos concretos y familiares?	Sí	
2,2	¿Los íconos/elementos se explican solos y son fáciles de entender?	No	
2,3	¿Aparecen en la misma zona elementos relacionados e interdependientes?	No aplica :(	
2,4	¿Los íconos/elementos siguen las convenciones del mundo real?	-	
2,5	¿Los colores seleccionados corresponden a las expectativas comunes sobre los códigos de color?	-	
2,6	En las secciones de ingreso de datos, ¿los campos a llenar son descritos con terminología familiar para el usuario?	-	
2,7	¿El lenguaje de la interfaz emplea la jerga del usuario y evita la jerga informática?	-	
2,8	¿El sistema/aplicación habla el idioma de los usuarios?	-	

2,9	¿Se han evitado las secuencias de letras poco comunes siempre que ha sido posible?	-	
<b>3. Control y libertad del usuario</b>			
3,1	¿Puede el usuario editar los datos de entrada antes de procesarlos?	Sí	
3,2	¿Pueden los usuarios reducir el tiempo de entrada de datos copiando y modificando los datos existentes?	No	
3,3	¿Pueden los usuarios revertir fácilmente sus acciones?	No aplica :(	
3,4	¿Existe una función de "deshacer" a nivel de una sola acción, una entrada de datos y un grupo completo de múltiples acciones?	-	
3,5	¿Se permiten ediciones de caracteres en los campos de entrada de datos?	-	
3,6	¿Pueden los usuarios establecer sus propios valores predeterminados a los archivos y elementos del editor?	-	
3,7	¿Permite la aplicación que los usuarios se recuperen fácilmente de los errores (ingreso de datos o símbolos inesperados)?	-	
3,8	¿La aplicación guía al usuario sobre las funciones o restringe la libertad del usuario en la aplicación?	-	
3,9	Si las listas de elementos son largas (más de siete elementos), ¿pueden los usuarios seleccionar un elemento moviendo el cursor o escribiendo un código mnemotécnico?	-	
<b>4. Consistencia y estándares</b>			
4,1	¿La interfaz gráfica sigue los estándares de formato de la industria?	Sí	
4,2	Los íconos/elementos de la interfaz de usuario, ¿siguen los estándares y los usos e implementaciones esperados?	No	
4,3	¿Las abreviaturas no incluyen puntuación?	No aplica :(	
4,4	¿Los números enteros están justificados a la derecha?	-	
4,5	¿Los íconos/elementos están etiquetados?	-	
4,6	¿No hay más de doce a veinte tipos de iconos?	-	

4,7	¿Se utilizan colores cromáticos de alto valor para atraer la atención?	-	
4,8	¿Cada página (archivo en el apartado de visualización) tiene un título?	-	
4,9	¿La aplicación contiene palabras con mayúsculas completas en toda la aplicación?	-	
4,10	¿Aparecen las notificaciones en la misma posición de la aplicación cada vez?	-	
4,11	¿Las etiquetas, entradas de datos y elementos de acción son coherentes en toda la aplicación?	-	
4,12	¿Las técnicas para llamar la atención se usan solo para condiciones excepcionales?	-	
4,13	¿Es posible el desplazamiento vertical y horizontal en cada ventana?	-	
4,14	¿Los títulos de los menús están centrados o justificados a la izquierda?	-	
4,15	¿Las convenciones de nombres son gramaticalmente correctas y coherentes en toda la aplicación?	-	
4,16	¿El lenguaje que se usa en los botones de acción es fácil de entender y común, lo que hace que sea fácil de recordar?	-	
4,17	Los elementos del menú (palette), ¿están justificados a la izquierda, con el número de elemento o mnemotécnico antes del nombre?	-	
4,18	¿Existe una estructura y convenciones coherentes en interfaz de usuario que reduzcan la carga cognitiva?	-	
<b>5. Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de errores</b>			
5,1	¿Se utiliza el sonido para señalar un error (ingreso de caracteres erróneos)?	Sí	
5,2	¿Los mensajes de error están redactados de modo que el sistema, no el usuario, cargue con la culpa?	No	
5,3	¿Los mensajes de error son breves y sin ambigüedades?	No aplica :(	
5,4	¿Los mensajes de error son gramaticalmente correctos?	-	
5,5	¿Los mensajes de error evitan el uso de signos de exclamación?	-	



5,6	¿Los mensajes de error informan al usuario de la gravedad del error?	-	
5,7	¿Los mensajes de error sugieren la causa del problema?	-	
5,8	¿Los mensajes de error proporcionan información en una semántica adecuada?	-	
5,9	¿Los mensajes de error proporcionan información en una sintáctica apropiada?	-	
5,10	¿Los mensajes de error indican qué acción debe realizar el usuario para corregir el error?	-	
5,11	¿Todos los mensajes de error del sistema utilizan un estilo gramatical, forma, terminología y abreviaturas coherentes?	-	
5,12	¿Los mensajes colocan a los usuarios en control del sistema?	-	
5,13	¿el sistema es compatible con usuarios novatos y expertos teniendo varios niveles de detalles de mensajes de error disponibles?	-	
5,14	¿La aplicación proporciona la función de deshacer después de iniciar la acción principal?	-	
5,15	¿El sistema advierte a los usuarios si están a punto de cometer un error potencialmente grave?	-	
5,16	¿La aplicación proporciona buenos valores predeterminados en las entradas de la aplicación?	-	
<b>6. Prevención de errores</b>			
6,1	¿Se han utilizado puntos o guiones bajos para el número de espacios de caracteres disponibles de las entradas de datos (properties)?	Sí	
6,2	¿En las entradas de datos contienen valores predeterminados cuando corresponde?	No	
6,3	¿Las entradas de datos son mayúsculas y minúsculas siempre que sea posible?	No aplica :(	
6,4	¿la navegación en el editor es simple y visible?	-	
6,5	¿El sistema evita que los usuarios cometan errores siempre que sea posible?	-	
6,6	¿El sistema advierte a los usuarios si están a punto de cometer un error potencialmente grave?	-	
6,7	¿Existen restricciones útiles y visuales que evitan que el usuario cometa errores?	-	



## 7. Reconocimiento en lugar de recuerdo

7,1	¿La visualización de datos comienza en la esquina superior izquierda de la pantalla?	Sí	
7,2	¿Los elementos del formulario (properties) se distribuyen siempre en un formato similar para cualquier elemento?	No	
7,3	¿Se colocan indicaciones, pistas y mensajes donde es probable que el ojo mire en la pantalla?	No aplica :(	
7,4	¿Las áreas de texto tienen "espacio para respirar" a su alrededor?	-	
7,5	¿Se utiliza el espacio en blanco para crear simetría y dirigir la vista en la dirección adecuada?	-	
7,6	¿Se han agrupado los elementos en zonas lógicas y se han utilizado encabezados para distinguir entre zonas?	-	
7,7	¿Están las etiquetas cerca de los elementos, pero separadas por al menos un espacio?	-	
7,8	¿Están claramente marcados las entradas de datos opcionales?	-	
7,9	¿Están claramente marcados las entradas de datos obligatorios?	-	
7,10	¿Se utiliza el tamaño, la negrita, el subrayado, el color, el sombreado o la tipografía para definir importancia en los diferentes elementos de la pantalla?	-	
7,11	¿Se utilizan bordes, espacios, líneas, colores, letras, títulos en negrita, líneas de reglas o áreas sombreadas para identificar zonas significativas?	-	
7,12	¿Se ha utilizado el mismo color para agrupar elementos relacionados?	-	
7,13	¿Tiene una paleta de colores consistente en toda la interfaz?	-	
7,14	¿Hay un buen contraste de color y brillo entre los elementos y el fondo?	-	
7,15	¿Se han usado colores claros, brillantes y saturados para enfatizar los datos y se han usado colores más oscuros, apagados y desaturados para no enfatizar los datos?	-	
7,16	¿Los elementos de la interfaz ofrecen posibilidades? Es decir, hacen evidente dónde existe una posible acción.	-	

## 8. Flexibilidad y eficiencia de uso

8,1	¿Se le presentan al usuario atajos para alcanzar los objetivos finales?	Sí	
8,2	¿Puede el usuario personalizar acciones frecuentes?	No	
8,3	¿Se le presenta al usuario información ambiental para acciones rápidas?	No aplica :(	
8,4	¿los usuarios pueden guardar entradas de datos (properties) parcialmente llena?	-	
8,5	En las entradas de datos (properties), ¿los usuarios tienen la opción de hacer clic directamente en la entrada o usar un atajo de teclado?	-	
<b>9. Diseño estético y minimalista</b>			
9,1	¿El diseño de la interfaz de usuario es simple y fácil de entender?	Sí	
9,2	¿Está abrumado el usuario con demasiada información y acciones principales que tomar?	No	
9,3	¿El usuario tiene claro el significado de todos los íconos y por qué están incluidos en el diseño?	No aplica :(	
9,4	¿Cada pantalla de entrada de datos (properties) tiene un título corto, sencillo, claro y distintivo?	-	
9,5	¿El sistema permite que el usuario se concentre en los elementos clave sin información irrelevante?	-	
9,6	¿La aplicación está diseñada con la agrupación adecuada, los colores y el uso de buenas formas?	-	
9,7	¿Se muestra la información esencial para la toma de decisiones que se muestra en la pantalla?	-	
9,8	¿Todos los íconos en un conjunto son visual y conceptualmente distintos?	-	
9,9	¿Cada ícono/elemento se destaca de su fondo?	-	
9,10	¿Hay grupos significativos de elementos separados por espacios en blanco?	-	
9,11	¿La sección de ingreso de datos (properties), presenta a los elementos por llenar de forma corta, simple, clara y distintiva?	-	
9,12	¿Los títulos de los menús son breves, pero lo suficientemente largos para comunicar?	-	
<b>10. Ayuda y Documentación</b>			

10,1	Si las opciones son ambiguas, ¿el sistema proporciona información adicional explicativa cuando se selecciona un elemento?	Sí	
10,2	¿Está visible la función de ayuda? por ejemplo, una tecla llamada AYUDA o un menú especial	No	
10,3	¿Se presenta al usuario información clara o adicional sobre los objetos/elementos del palette cuando es necesario?	No aplica :(	

## ANEXO V. Evaluación Interna

Lista de Verificación		Sí / No / No aplica	Comentarios & Sugerencias
<b>1. Visibilidad del estado del sistema</b>			
1,1	Cuando se selecciona un icono / elemento, ¿se resalta de otros elementos que no están seleccionados?	Sí	Existe un borde negro que resalta al elemento seleccionado en el modelo.
1,2	¿Todas las pantallas comienzan con un título o encabezado que describe el contenido de la pantalla?	Sí	En la parte del modelo existe una sección que detalla el nombre del archivo abierto
1,3	¿No son necesarios altos niveles de concentración o se requiere recordar información?	Sí	Al ejecutar o tener en cuenta las tareas puede parecer complejo por llenar diversas propiedades
1,4	¿Hay señales visuales sobresalientes para identificar la selección activa?	Sí	Existen bordes o subrayado en azul para el texto
1,5	Cuando se hace clic en un botón de acción o se completa un proceso, ¿el sistema proporciona una retroalimentación al usuario?	No	No se evidencia una retroalimentación.
1,6	¿Se notifica al usuario cuando un grupo de acciones se completó con éxito e indica el siguiente conjunto de acciones a realizar para completar la tarea/lograr el objetivo?	No aplica :(	
1,7	¿Los elementos muestran claramente en cuáles se puede hacer clic y en cuáles no?	No	Todo parece que se puede hacer clic, y no cambia la forma del mouse al pasar por encima de un elemento.
1,8	Si existen menús, ¿hay información visual sobre qué opciones ya están seleccionadas?	Sí	En la parte izquierda existe un menú del desglose de archivos que se marca en azul el elemento que se selecciona.
1,9	¿Hay retroalimentación visual (cambia la forma o el color) cuando se seleccionan o se mueven los íconos/elementos?	Sí	Existe una sombra azul cuando se intenta mover un elemento, más no se mueve el elemento hasta que se deja de soltar el mouse.
1,10	¿Se indica claramente el estado actual de un ícono/elemento?	Sí	Solo cuando se lo selección.
1,11	¿El sistema proporciona una retroalimentación cuando se hace clic en las acciones principales?	No	Solo se da el nombre del campo que se tiene que llenar.
1,12	Al mirar, ¿el usuario puede saber el estado del sistema y las alternativas de acción?	No	Existe solo el modelo, y a menos que se sepa la tarea a realizar no se sabe el paso a realizar.

2. Coincidencia entre el sistema y el mundo real			
2,1	¿Son los iconos concretos y familiares?	No	Algunos son ambiguos como el icono para definir zonas, parece como un punto en Google Maps por el icono usado.
2,2	¿Los íconos/elementos se explican solos y son fáciles de entender?	No	Iconos como el correspondiente de Static Ip, solo son entendidos al leer su nombre en el palette.
2,3	¿Aparecen en la misma zona elementos relacionados e interdependientes?	Sí	se puede identificar en la UI al menos 5 zonas donde se encuentran sus elementos, campos e iconos.
2,4	¿Los íconos/elementos siguen las convenciones del mundo real?	No	Ciertos íconos si no siguen el estándar como los íconos correspondientes a load Balancer, Zone, Tag, Static IP
2,5	¿Los colores seleccionados corresponden a las expectativas comunes sobre los códigos de color?	No	Hay abundancia de colores primarios, pero son usados más de una vez en distintos elementos.
2,6	En las secciones de ingreso de datos, ¿los campos a llenar son descritos con terminología familiar para el usuario?	No	Algunos elementos como Health Check, Bucket y Launch Configuration no son del todo comprendidos ni sus usos
2,7	¿El lenguaje de la interfaz emplea la jerga del usuario y evita la jerga informática?	No	Aplica mucha jerga Informática y emplea nombres con ciertas convenciones como "eu-west-1a" o Bucket.
2,8	¿El sistema/aplicación habla el idioma de los usuarios?	Sí	Parcialmente ya que, en propiedades abunda tecnicismos.
2,9	¿Se han evitado las secuencias de letras poco comunes siempre que ha sido posible?	Sí	No existen caracteres poco usados
3. Control y libertad del usuario			
3,1	¿Puede el usuario editar los datos de entrada antes de procesarlos?	Sí	Cuando se pone datos en las propiedades, son verificados antes en el mismo momento
3,2	¿Pueden los usuarios reducir el tiempo de entrada de datos copiando y modificando los datos existentes?	Sí	Si se pueden copiar y pegar elementos ingresados.
3,3	¿Pueden los usuarios revertir fácilmente sus acciones?	Sí	Con control+z revierte cierta acción.
3,4	¿Existe una función de "deshacer" a nivel de una sola acción, una entrada de datos y un grupo completo de múltiples acciones?	No	
3,5	¿Se permiten ediciones de caracteres en los campos de entrada de datos?	Sí	
3,6	¿Pueden los usuarios establecer sus propios valores predeterminados a los archivos y elementos del editor?	No aplica :(	
3,7	¿Permite la aplicación que los usuarios se recuperen fácilmente de los errores (ingreso de datos o símbolos inesperados)?	Sí	Cuando se ingresa caracteres no esperados solo vuelve a su estado anterior el campo de entrada.
3,8	¿La aplicación guía al usuario sobre las funciones o restringe la libertad del usuario en la aplicación?	No	No se ve alguna restricción.
3,9	Si las listas de elementos son largas (más de siete elementos), ¿pueden los usuarios seleccionar un elemento moviendo el	Sí	En el palette existe algunos elementos que se muestran más y más con scroll down

	cursor o escribiendo un código mnemotécnico?		
<b>4. Consistencia y estándares</b>			
4,1	¿La interfaz gráfica sigue los estándares de formato de la industria?	No	Los elementos para seleccionar usualmente se encuentran en el lado izquierdo y muchas propiedades a llenar se lo hacen con pop-ups.
4,2	Los íconos/elementos de la interfaz de usuario, ¿siguen los estándares y los usos e implementaciones esperados?	Sí	
4,3	¿Las abreviaturas no incluyen puntuación?	Sí	
4,4	¿Los números enteros están justificados a la derecha?	No	En propiedades están alineados a la izquierda
4,5	¿Los íconos/elementos están etiquetados?	Sí	cada elemento tiene su nombre a su lado
4,6	¿No hay más de doce a veinte tipos de íconos?	Sí	Existen 18 íconos en el palette
4,7	¿Se utilizan colores cromáticos de alto valor para atraer la atención?	Sí	Son colores primarios
4,8	¿Cada página (archivo en el apartado de visualización) tiene un título?	Sí	
4,9	¿La aplicación contiene palabras con mayúsculas completas en toda la aplicación?	No	
4,10	¿Aparecen las notificaciones en la misma posición de la aplicación cada vez?	Sí	Cuando se ingresa mal un dato en las propiedades.
4,11	¿Las etiquetas, entradas de datos y elementos de acción son coherentes en toda la aplicación?	Sí	Las entradas de datos en propiedades son coherentes en toda la aplicación.
4,12	¿Las técnicas para llamar la atención se usan solo para condiciones excepcionales?	No aplica :)	
4,13	¿Es posible el desplazamiento vertical y horizontal en cada ventana?	Sí	
4,14	¿Los títulos de los menús están centrados o justificados a la izquierda?	Sí	Justificados a la izquierda
4,15	¿Las convenciones de nombres son gramaticalmente correctas y coherentes en toda la aplicación?	Sí	No se ha encontrado una excepción
4,16	¿El lenguaje que se usa en los botones de acción es fácil de entender y común, lo que hace que sea fácil de recordar?	Sí	
4,17	Los elementos del menú (palette), ¿están justificados a la izquierda, con el número de elemento o mnemotécnico antes del nombre?	Sí	
4,18	¿Existe una estructura y convenciones coherentes en interfaz de usuario que reduzcan la carga cognitiva?	-	
<b>5. Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de errores</b>			
5,1	¿Se utiliza el sonido para señalar un error (ingreso de caracteres erróneos)?	No	
5,2	¿Los mensajes de error están redactados de modo que el sistema, no el usuario, cargue con la culpa?	Sí	Estos mensajes reportan de forma técnica el error cometido
5,3	¿Los mensajes de error son breves y sin ambigüedades?	No	Los mensajes son como "NumberFormatException: For input

			string: 'XCVB'", que para usuarios Francisco puede ser confuso.
5,4	¿Los mensajes de error son gramaticalmente correctos?	Sí	
5,5	¿Los mensajes de error evitan el uso de signos de exclamación?	Sí	
5,6	¿Los mensajes de error informan al usuario de la gravedad del error?	No	Al tener mensajes de error muy técnicos no se
5,7	¿Los mensajes de error sugieren la causa del problema?	No	Menciona que causa el error, pero no sugiere que es correcto como: "NumberFormatException: For input string: 'XCVB'"
5,8	¿Los mensajes de error proporcionan información en una semántica adecuada?	No	El detalle del error podría asemejarse a lenguaje de bajo nivel.
5,9	¿Los mensajes de error proporcionan información en una sintáctica apropiada?	No	Su sintaxis es adecuada pero anda intuitiva para el usuario
5,10	¿Los mensajes de error indican qué acción debe realizar el usuario para corregir el error?	No	
5,11	¿Todos los mensajes de error del sistema utilizan un estilo gramatical, forma, terminología y abreviaturas coherentes?	Sí	
5,12	¿Los mensajes colocan a los usuarios en control del sistema?	No	No se sugiere ningún tipo de ayuda.
5,13	¿el sistema es compatible con usuarios novatos y expertos teniendo varios niveles de detalles de mensajes de error disponibles?	No	
5,14	¿La aplicación proporciona la función de deshacer después de iniciar la acción principal?	No	No existe ningún componente sugiera o indique para deshacer acciones.
5,15	¿El sistema advierte a los usuarios si están a punto de cometer un error potencialmente grave?	Sí	El mensaje se indica en la parte inferior desde el ingreso de datos inesperados.
5,16	¿La aplicación proporciona buenos valores predeterminados en las entradas de la aplicación?	Sí	En ciertos campos de propiedades se tiene valores predeterminados, por ejemplo, para "Protocol" y sus opciones "TCP, UDP, ICMP, ALL"
<b>6. Prevención de errores</b>			
6,1	¿Se han utilizado puntos o guiones bajos para el número de espacios de caracteres disponibles de las entradas de datos (properties)?	No	Los campos de entrada en propiedades no especifican el número de caracteres d ninguna forma.
6,2	¿En las entradas de datos contienen valores predeterminados cuando corresponde?	Sí	Parcialmente como para definir protocolos.
6,3	¿Las entradas de datos son mayúsculas y minúsculas siempre que sea posible?	No	No hacen existe ninguna limitación para caracteres en mayúscula o minúscula.
6,4	¿la navegación en el editor es simple y visible?	Sí	Solo compone de una pantalla o UI
6,5	¿El sistema evita que los usuarios cometan errores siempre que sea posible?	Sí	Parcialmente, como para campos que solo se aceptan números detecta letras y deshace lo que hizo el usuario.
6,6	¿El sistema advierte a los usuarios si están a punto de cometer un error potencialmente	Sí	Mientras se está ingresando datos erróneos.

	grave?		
6,7	¿Existen restricciones útiles y visuales que evitan que el usuario cometa errores?	No	NO existe ningún tipo de advertencia.
<b>7. Reconocimiento en lugar de recuerdo</b>			
7,1	¿La visualización de datos comienza en la esquina superior izquierda de la pantalla?	No aplica :)	
7,2	¿Los elementos del formulario (properties) se distribuyen siempre en un formato similar para cualquier elemento?	Sí	Para cada componente la opción de properties se despliega siempre de la misma forma.
7,3	¿Se colocan indicaciones, pistas y mensajes donde es probable que el ojo mire en la pantalla?	No	NO existe indicaciones visualmente notorias
7,4	¿Las áreas de texto tienen "espacio para respirar" a su alrededor?	Sí	Parcialmente, en la opción de properties no existe espacio para respirar en su altura, pero si en su anchura.
7,5	¿Se utiliza el espacio en blanco para crear simetría y dirigir la vista en la dirección adecuada?	-	
7,6	¿Se han agrupado los elementos en zonas lógicas y se han utilizado encabezados para distinguir entre zonas?	Sí	En 5 zonas distinguibles que tienen archivos, campos de entrada, elementos y diagrama.
7,7	¿Están las etiquetas cerca de los elementos, pero separadas por al menos un espacio?	Sí	En los elementos del palette si cumple.
7,8	¿Están claramente marcados las entradas de datos opcionales?	No	No existe ninguna forma que indique ciertos datos son opcionales.
7,9	¿Están claramente marcados las entradas de datos obligatorios?	No	No existe forma visual que indique los datos obligatorios a llenar.
7,10	¿Se utiliza el tamaño, la negrita, el subrayado, el color, el sombreado o la tipografía para definir importancia en los diferentes elementos de la pantalla?	No	No hace uso de esos elementos, pues todo el texto mostrado tiene un mismo tamaño, color y tipografía.
7,11	¿Se utilizan bordes, espacios, líneas, colores, letras, títulos en negrita, líneas de reglas o áreas sombreadas para identificar zonas significativas?	Sí	Las 5 zonas significativas están marcadas por colores como celeste y gris para definir el interior de la zona.
7,12	¿Se ha utilizado el mismo color para agrupar elementos relacionados?	No	No es notoria una agrupación de elementos por color.
7,13	¿Tiene una paleta de colores consistente en toda la interfaz?	Sí	Se usa los mismos colores primarios y también celeste y gris para delimitar sus zonas.
7,14	¿Hay un buen contraste de color y brillo entre los elementos y el fondo?	Sí	
7,15	¿Se han usado colores claros, brillantes y saturados para enfatizar los datos y se han usado colores más oscuros, apagados y desaturados para no enfatizar los datos?	No	No hay una diferenciación de énfasis en los colores y ni menos para los datos.
7,16	¿Los elementos de la interfaz ofrecen posibilidades? Es decir, hacen evidente dónde existe una posible acción.	No	Resulta confuso para las posibles acciones a seguir si es Francisco. Además, los archivos de la parte izquierda no contienen información adicional.
<b>8. Flexibilidad y eficiencia de uso</b>			
8,1	¿Se le presentan al usuario atajos para alcanzar los objetivos finales?	No	No ha aparecido ningún tipo de atajo al pasar el mouse.



8,2	¿Puede el usuario personalizar acciones frecuentes?	No	No se ha mostrado tal opción en la UI.
8,3	¿Se le presenta al usuario información ambiental para acciones rápidas?	No	No está presente tal información.
8,4	¿Los usuarios pueden guardar entradas de datos (properties) parcialmente llenas?	Sí	No existe indicaciones sobre campos obligatorios a llenar.
8,5	En las entradas de datos (properties), ¿los usuarios tienen la opción de hacer clic directamente en la entrada o usar un atajo de teclado?	Sí	Se puede editar tal entrada al hacer clic directamente.
<b>9. Diseño estético y minimalista</b>			
9,1	¿El diseño de la interfaz de usuario es simple y fácil de entender?	No	Es simple, pero existe información muy técnica que necesita información adicional para entender su significado.
9,2	¿Está abrumado el usuario con demasiada información y acciones principales que tomar?	No	No existe demasiada información al ser una sola UI, pero su jerga Informática complica su entendimiento.
9,3	¿El usuario tiene claro el significado de todos los íconos y por qué están incluidos en el diseño?	No	Ciertos iconos no tienen un estándar de uso por lo que su uso queda en duda para usuarios novatos.
9,4	¿Cada pantalla de entrada de datos (properties) tiene un título corto, sencillo, claro y distintivo?	Sí	Tiene, pero su nombre se limita al nombre del elemento.
9,5	¿El sistema permite que el usuario se concentre en los elementos clave sin información irrelevante?	Sí	El foco del interés se dirige al modelo y sus elementos maximizados en la parte central
9,6	¿La aplicación está diseñada con la agrupación adecuada, los colores y el uso de buenas formas?	No aplica :)	
9,7	¿Se muestra la información esencial para la toma de decisiones que se muestra en la pantalla?	No	No es intuitivo obtener el script de Ansible para un usuario novato, pues no existe una acción clara con algún para ello.
9,8	¿Todos los íconos en un conjunto son visual y conceptualmente distintos?	Sí	Son distintos pero ciertos elementos como Health Check, Listener y Load Balancer utilizan un mismo color para sus íconos.
9,9	¿Cada ícono/elemento se destaca de su fondo?	No	Los elementos que llevan el color verde lima no se destacan del fondo color blanco.
9,10	¿Hay grupos significativos de elementos separados por espacios en blanco?	Sí	Existe y tal es el caso de properties que tiene espacio en blanco que delimita su ubicación.
9,11	¿La sección de ingreso de datos (properties), presenta a los elementos por llenar de forma corta, simple, clara y distintiva?	No	Es corta y simple pero ciertos campos tienden a confusión por su significado.
9,12	¿Los títulos de los menús son breves, pero lo suficientemente largos para comunicar?	Sí	Son breves y comunican lo que contienen.
<b>10. Ayuda y Documentación</b>			
10,1	Si las opciones son ambiguas, ¿el sistema proporciona información adicional explicativa cuando se selecciona un elemento?	No	No existe ningún tipo de información adicional que ayude a un usuario novato.
10,2	¿Está visible la función de ayuda? por	No	En la parte de arriba este un menú



	ejemplo, una tecla llamada AYUDA o un menú especial		que hace alusión con su nombre "Help", pero no se encontró información específica para modelos con ARGON.
10,3	¿Se presenta al usuario información clara o adicional sobre los objetos/elementos del palette cuando es necesario?	No	No existe información adicional que ayude a entender el uso de cada elemento.

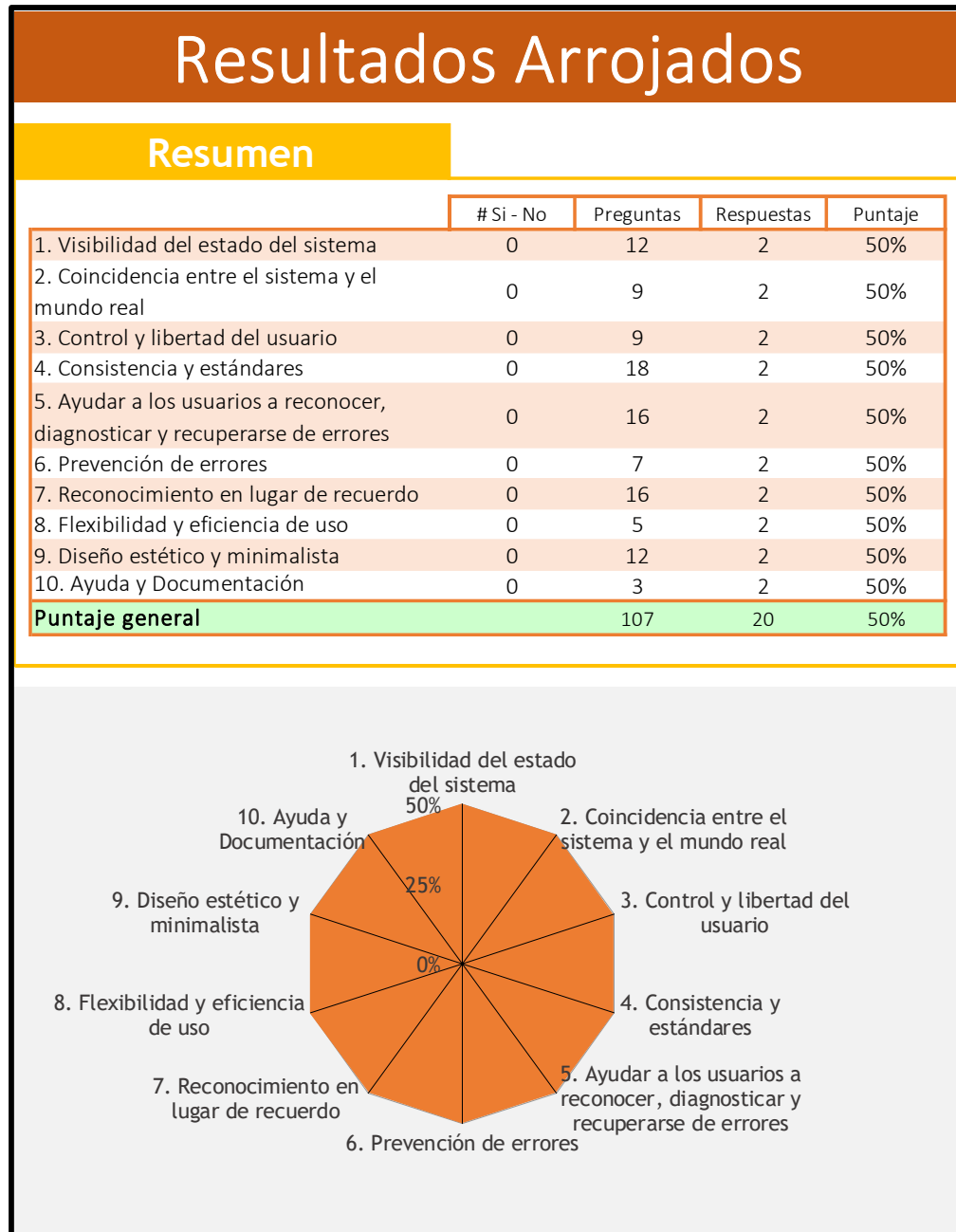
## ANEXO VI. Análisis Heurístico para ARGON (enlace)

[https://epnecuador-my.sharepoint.com/:x/g/personal/cristhian\\_munoz\\_epn\\_edu\\_ec/EUPKR9UW7\\_JMiOfSvr9Y21MBpdBlmzvCFEXatstEjp31RA?e=gq4vmb](https://epnecuador-my.sharepoint.com/:x/g/personal/cristhian_munoz_epn_edu_ec/EUPKR9UW7_JMiOfSvr9Y21MBpdBlmzvCFEXatstEjp31RA?e=gq4vmb)

## ANEXO VII. Evaluaciones Externas (enlace)

[https://epnecuador-my.sharepoint.com/:f/g/personal/cristhian\\_munoz\\_epn\\_edu\\_ec/ErMWU1qXoXtNrHB3Nxxd0HgBOA4UNfr4m4sO2XBB5z5pig?e=fynYmc](https://epnecuador-my.sharepoint.com/:f/g/personal/cristhian_munoz_epn_edu_ec/ErMWU1qXoXtNrHB3Nxxd0HgBOA4UNfr4m4sO2XBB5z5pig?e=fynYmc)

## ANEXO VIII. Resumen autogenerated sobre las respuestas de la Lista de Verificación



## ANEXO IX. Recopilación de problemas de usabilidad (enlace)

[https://epnecuador-my.sharepoint.com/:x/g/personal/cristhian\\_munoz\\_epn\\_edu\\_ec/EdovMtszkEdBp1aoBwKcYksBF\\_2q38ZzQzb9IDa9t4KmAg?e=h0EDr3](https://epnecuador-my.sharepoint.com/:x/g/personal/cristhian_munoz_epn_edu_ec/EdovMtszkEdBp1aoBwKcYksBF_2q38ZzQzb9IDa9t4KmAg?e=h0EDr3)

## ANEXO X. Asignación de Severidad a los problemas de usabilidad

Problemas		Severidad	Recomendaciones & Sugerencias
<b>1. Visibilidad del estado del sistema</b>			
1,1	Cuando se hace clic en un botón de acción o se completa un proceso, el sistema <b>NO</b> proporciona una retroalimentación al usuario.	2	Definir pop ups emergentes indicando el éxito o fracaso de ciertas acciones.
1,2	<b>NO</b> se notifica al usuario cuando un grupo de acciones se completó con éxito e indica el siguiente conjunto de acciones a realizar para completar la tarea/lograr el objetivo.	2	Definir claramente los procesos para generar con éxito un modelo, saltando pop ups emergentes indicando el éxito o fracaso de ciertos grupos de acciones.
1,3	Los elementos <b>NO</b> muestran claramente en cuáles se puede hacer clic y en cuáles no. Todo parece que se puede hacer clic, y no cambia la forma del mouse al pasar por encima de un elemento.	3	Definir un color característico a los elementos que puedan ser elegidos por el mouse o que se muevan y cambien de color cuando se pase el cursor por encima de los elementos.
1,4	<b>NO</b> hay retroalimentación visual (cambia la forma o el color) cuando se seleccionan o se mueven los íconos/elementos. Existe una sombra azul cuando se intenta mover un elemento, más no se mueve el elemento hasta que se deja de soltar el mouse.	4	Definir correctamente la función de Drag and Drop, haciendo su respectiva animación cuando se esté moviendo de un lado a otro el elemento elegido.
1,5	El sistema <b>NO</b> proporciona una retroalimentación cuando se hace clic en las acciones principales. Por ejemplo, en propiedades solo se da el nombre del campo que se tiene que llenar.	1	Que se muestre con pop ups, posibles ejemplos de datos de entrada que se esperaría. Podrían ser animaciones que se activen automáticamente.
1,6	Son necesarios altos niveles de concentración o se requiere recordar información	3	Disminuir la carga cognitiva, guiando al usuario en ciertas acciones que resulten complejas, con el obtener el script para Ansible.
1,7	Al mirar, el usuario <b>NO</b> puede saber el estado del sistema y las alternativas de acción. Pues, existe solo el modelo, y a menos que se sepa la tarea a realizar no se sabe el paso a realizar.	3	Definir durante los primeros usos del editor que ciertos botones parpadeen a fin de indicar una posible acción después de terminar con alguna fase del modelado de infraestructura. Por ejemplo, que el botón para generar script se active una vez que el modelo no tenga errores y sea verificado.
<b>2. Coincidencia entre el sistema y el mundo real</b>			
2,1	<b>NO</b> son los íconos concretos y familiares. Algunos son ambiguos como el icono para definir zonas, parece como un punto en Google Maps por el ícono usado.	4	Redefinir los íconos con menos carga visual o pequeños detalles que puedan ser confusos.

2,2	Los íconos/elementos <b>NO</b> se explican solos y son fáciles de entender. Iconos como el correspondiente de Static Ip, solo son entendidos al leer su nombre en el palette.	4	
2,3	Los íconos/elementos <b>NO</b> siguen las convenciones del mundo real. Ya que, ciertos íconos si no siguen el estándar como los íconos correspondientes a load Balancer, Zone, Tag y Static IP.	4	Redefinir los íconos con el uso de íconos semejantes que sean utilizados comúnmente por otras empresas, tal es el ejemplo del editor de Packet Tracer.
2,4	Los colores seleccionados <b>NO</b> corresponden a las expectativas comunes sobre los códigos de color. Aunque los colores usados son colores primarios, se usan más de una vez en distintos elementos y eso causa cierta confusión.	2	Elegir una paleta de colores distinto y características de la herramienta, como Amazon y su color amarillo característico y sus variantes.
2,5	En las secciones de ingreso de datos, los campos a llenar <b>NO</b> son descritos con terminología familiar para el usuario. Y esto se observa cuando se intenta llenar las propiedades de elementos como Health Check, Bucket y Launch Configuration, los cuales no son del todo comprendidos ni sus usos.	3	Mantener los nombres de ciertos componentes tal como otros autores o empresas hacen uso de estos. Se puede proveer ejemplos gráficos del uso y significado de los elementos.
2,6	El lenguaje de la interfaz <b>NO</b> emplea la jerga del usuario. Ya que, aplica mucha jerga Informática y emplea nombres con ciertas convenciones como "eu-west-1a" o Bucket, cuyos términos resultan confusos.	3	Para definir términos con cierta estructura proveer información al usuario sobre el uso y explicar la importancia de aquellas convenciones. Pop ups de ayuda emergentes.
2,7	El sistema/aplicación <b>NO</b> habla el idioma de los usuarios, ya que, en propiedades abunda tecnicismos.	4	añadir toda la información necesaria para conocer el significado de los elementos.
<b>3. Control y libertad del usuario</b>			
3,1	<b>NO</b> pueden los usuarios revertir fácilmente sus acciones. Pues no existen botones específicos para deshacer, obligando al usuario a usar acciones como control+z para revertir.	3	Definir uno o botones para deshacer acciones en conjunto o una sola acción y en su contraparte permite rehacer dichas acciones.
3,2	<b>NO</b> existe una función de "deshacer" a nivel de una sola acción, una entrada de datos y un grupo completo de múltiples acciones.	3	
3,3	<b>NO</b> pueden los usuarios establecer sus propios valores predeterminados a los archivos y elementos del editor.	1	En las propiedades de cada elemento o ícono, se recomendaría habilitar una opción que permita al usuario poner su configuración por defecto.
3,4	La aplicación <b>NO</b> guía al usuario sobre las funciones. Dejando al usuario que intente deducir su próxima acción.	4	Definir un tutorial por defecto siempre que un usuario novato desee conocer el funcionamiento del editor.
<b>4. Consistencia y estándares</b>			
4,1	La interfaz gráfica <b>NO</b> sigue los	3	Los elementos a seleccionar usualmente se

	estándares de formato de la industria.		encuentran en el lado izquierdo y muchas propiedades a llenar se lo hacen con pop-ups.
4,2	Los números <b>NO</b> enteros están justificados a la derecha, ya que, en propiedades están alineados a la izquierda.	4	se recomienda dejar claro el tipo de dato y más si es un entero justificando en propiedades a la parte derecha.
4,3	La aplicación <b>NO</b> contiene palabras con mayúsculas completas en toda la aplicación.	0	En caso de ser necesario para distintos elementos, se lo recomienda el uso de mayúsculas.
4,4	Los elementos (textos y líneas del modelo) al sobreponerse no se identifican muy bien.	4	Restricción para que las líneas del modelo se pongan encima de los nombres que identifican los elementos/iconos modelados.
<b>5. Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de errores</b>			
5,1	<b>NO</b> Se utiliza el sonido para señalar un error (ingreso de caracteres erróneos).	1	Se recomienda el uso especialmente cuando se comete errores, estos llamarán la atención del usuario para corregir rápidamente.
5,2	Los mensajes de error son breves, pero con ambigüedades. Los mensajes son como "NumberFormatException: For input string: 'XCVB'", que para usuarios Francisco puede ser confuso.	2	Adaptarse al lenguaje (Español, English, Francés) de Francisco.
5,3	Los mensajes de error <b>NO</b> informan al usuario de la gravedad del error. Al ser los mensajes muy técnicos no dan información de la gravedad.	3	Definir con colores la gravedad del error que se muestra como un pop up.
5,4	Los mensajes de error <b>NO</b> sugieren la causa del problema. Menciona que causa el error, pero no sugiere que es correcto como: "NumberFormatException: For input string: 'XCVB'"	3	
5,5	Los mensajes de error <b>NO</b> proporcionan información en una semántica adecuada. El detalle del error podría asemejarse a lenguaje de bajo nivel.	3	Definir los mensajes en un lenguaje de alto nivel, el que use el usuario.
5,6	Los mensajes de error proporcionan información en una sintáctica apropiada, pero <b>NO</b> es nada intuitiva para el usuario	3	
5,7	<b>NO</b> existe coherencia entre la fuente y la interfaz de usuario, es decir, una vez generado el script no es posible su modificación por medio del modelo gráfico. Lo que puede causar problemas al no es relacionados estos documentos.	2	permitir que le script sea generado en paralelo con el modelo, siendo visible los cambios que se hacen en el modelo para el script y viceversa.
5,8	Los mensajes <b>NO</b> colocan a los usuarios en control del sistema, pues no sugieren ninguna posibilidad para solucionar.	4	Mostrar los mensajes de error alternativas de acción claras y cortas hacia el usuario.
5,9	El sistema <b>NO</b> es compatible con usuarios novatos y expertos.	3	Tener varios niveles de detalles de mensajes de error disponibles, tanto para usuarios novatos como expertos.

5,10	La aplicación <b>NO</b> proporciona la función de deshacer después de iniciar la acción principal. No existe ningún componente sugiera o indique para deshacer acciones.	3	Incluir un botón que sugiera deshacer acciones.
5,11	El sistema <b>NO</b> advierte a los usuarios si están a punto de cometer un error potencialmente grave. Aunque, el mensaje se indica en la parte inferior para el ingreso de datos inesperados. Pero no advierte de la gravedad.	4	Mostrar con pop ups el error que comete el usuario y, lo que se esperaría que haga para corregir.
<b>6. Prevención de errores</b>			
6,1	<b>NO</b> se han utilizado puntos o guiones bajos para el número de espacios de caracteres disponibles de las entradas de datos (properties).	1	Definir con el tamaño del campo los caracteres que permite o la utilización de guiones " _ ".
6,2	La aplicación proporciona parcialmente valores predeterminados en las entradas de la aplicación. Pero, solo en ciertos campos de propiedades se tiene valores predeterminados, por ejemplo, para "Protocolo" y sus opciones "TCP, UDP, ICMP, ALL". Pero, no tiene en cuenta quizás a valores con cierta estructura como "imagen".	3	Colocar valores de entrada predeterminados siempre que sean posibles o que usuario pueda seleccionar alguna alternativa posible.
6,3	<b>No</b> existe ninguna limitación para escribir caracteres en mayúscula o minúscula.	4	Indicar al usuario el tipo de datos que se espera.
6,4	El sistema evita que los usuarios cometan errores siempre que sea posible. Parcialmente, como para campos que solo se aceptan números detecta letras y deshace lo que hizo el usuario.	3	Aunque deshace automáticamente la acción del usuario, se necesita que el usuario sea notificado de forma simple y entendible.
6,5	El sistema advierte a los usuarios si están a punto de cometer un error potencialmente grave. Pero, solo en ciertas situaciones como cuando se está ingresando datos erróneos y no es claro lo que se espera.	4	Para el ingreso de datos (properties), definir sobre el tipo de datos que son esperados.
6,6	NO existe restricción para nombrar los elementos del modelo, teniendo en muchos casos nombres duplicados.	2	Definir restricciones visuales con pop ups indicando que no se puede nombrar a los elementos de la misma forma.
6,7	<b>NO</b> existen restricciones útiles y visuales que evitan que el usuario cometa errores.	3	No mostrar todos los campos que sean innecesarios. Además, se puede colocar datos de ejemplo que sirvan de guía al usuario para definir formatos.
<b>7. Reconocimiento en lugar de recuerdo</b>			
7,1	<b>NO</b> se colocan indicaciones, pistas y mensajes donde es probable que el ojo mire en la pantalla.	2	Enfatizar botones de acción que deba tener presente el usuario.
7,2	<b>NO</b> existe ninguna forma que indique ciertos datos son opcionales.	2	No obligar a llenar esos campos y poner estos campos con colores muy bajos.

			También se podrían ocultar estos elementos a fin de dar prioridad a los esenciales.
7,3	<b>NO</b> existe forma visual que indique los datos obligatorios a llenar	3	Colocar color rojo o un asterisco dejando claro que campos son obligatorios a llenar.
7,4	<b>NO</b> Se utiliza el tamaño, la negrita, el subrayado, el color, el sombreado o la tipografía para definir importancia en los diferentes elementos de la pantalla.	1	Hacer uso de tamaño, la negrita, el subrayado, el color, el sombreado o la tipografía, a fin de mostrar un énfasis en determinadas acciones o elementos.
7,5	<b>NO</b> Se ha utilizado el mismo color para agrupar elementos relacionados.	1	Agrupar por colores elementos del palette que tengan gran relación entre sí.
7,6	<b>NO</b> Se han usado colores claros, brillantes y saturados para enfatizar los datos y se han usado colores más oscuros, apagados y desaturados para no enfatizar los datos.	2	Cuando se seleccione un elemento bajar la intensidad de los colores pertenecientes a los otros componentes modelados. Para luego, volver a la misma intensidad de color al dejar de seleccionar el elemento.
7,7	Los elementos de la interfaz <b>NO</b> hacen evidente dónde existe una posible acción.	2	Resulta confuso para las posibles acciones a seguir si es Francisco. Además, para los archivos en la parte izquierda agregar información adicional que ayude a comprender el su uso e importancia.
7,8	<b>NO</b> hay opciones de avance hacia atrás, el usuario debe forzar un atajo como ctrl + z	2	Colocar un botón visible que sugiera deshacer acciones.
<b>8. Flexibilidad y eficiencia de uso</b>			
8,1	<b>NO</b> se le presentan al usuario atajos para alcanzar los objetivos finales. Ya que, No aparece ningún tipo de atajo al pasar el mouse.	2	Mostrar claramente los botones de acción y sus posibles atajos para agilizar el trabajo del usuario. (crear un nuevo modelo, verificar, generar script)
8,2	<b>NO</b> puede el usuario personalizar acciones frecuentes.	1	Permitir al usuario definir parámetros por defecto de acuerdo a las necesidades.
8,3	<b>NO</b> se le presenta al usuario información ambiental para acciones rápidas.	2	Sugerir con pop up un siguiente paso al usuario, teniendo en cuenta la última acción realizada.
8,4	Los usuarios <b>NO</b> pueden guardar entradas de datos (properties) parcialmente llena, ya que, <b>NO</b> existe indicaciones sobre campos obligatorios a llenar.	3	Obligar al usuario a llenar ciertos campos completamente necesarios y, no dejarle pasar a otra tarea en caso de estar incompleto las propiedades de un elemento.
8,5	<b>NO</b> existen shortcuts para proporcionar información adicional para ayuda al usuario.	2	Añadir información sobre los shortcuts posibles para cuando el cursor esté sobre el elemento o botón de acción.
<b>9. Diseño estético y minimalista</b>			
9,1	El diseño de la interfaz de usuario es simple, pero <b>NO</b> fácil de entender porque existe información muy técnica que necesita información adicional para entender su significado.	4	Si es necesario mantener el tecnicismo, se recomienda añadir toda la información adicional para que sea entendible.
9,2	El usuario <b>NO</b> tiene claro el significado de todos los íconos y por qué están incluidos en el diseño. Ciertos iconos no tienen un estándar de uso por lo que su uso queda en duda para usuarios novatos.	4	Rediseño de los elementos/iconos.

9,3	<b>NO</b> es intuitivo obtener el script de Ansible para un usuario novato, pues no existe una acción clara con algún para ello.	4	Añadir una acción en forma de botón que siempre esté visible, activándose una vez terminado el modelo de infraestructura completo y verificado.
9,4	La sección de ingreso de datos (properties), presenta a los elementos por llenar de forma corta, simple pero ciertos campos tienden a confusión por su significado.	4	Añadir información adicional cuando el cursor esté sobre el elemento.
9,5	Objetos visuales (palette) demasiado pequeños, sin búsqueda de elementos. Esto obliga al usuario a buscar de arriba abajo.	3	Modificar el tamaño de los íconos para que puedan verse desde un metro de distancia sin complicaciones.
9,6	<b>NO</b> hay definición de todas las propiedades para todos los elementos (por ejemplo, elemento de enlace)	2	Añadir información adicional cuando el cursor esté sobre el elemento.
<b>10. Ayuda y Documentación</b>			
10,1	<b>NO</b> existe ningún tipo de información adicional que ayude a un usuario novato.	1	Incluir una sección plegable que muestre parte de la documentación cuando se selecciona un componente, enseñando a un usuario novato sobre los elementos que va a encontrar en la IU.
10,2	<b>NO</b> está visible la función de ayuda. En la parte de arriba este un menú que hace alusión con su nombre "Help", pero no se encontró información específica para modelos con ARGON.	4	Añadir un botón o caja para buscar ayudar sobre un tema o componente en específico para ARGON.
10,3	<b>NO</b> Se presenta al usuario información clara o adicional sobre los objetos/elementos del palette cuando es necesario.	3	Añadir información adicional cuando el cursor esté sobre el elemento.
10,4	Cuando los usuarios cometen errores (enlazar Listener - LoadBalancer), <b>NO</b> existen documentación de ayuda.	4	Hacer clic derecho o para mostrar un icono de ayuda. por lo que el usuario se ve obligado a encontrar accesos directos como fn + F1

## ANEXO XI. Infografía del reporte de hallazgos (enlace opcional)

[https://www.canva.com/design/DAFIOE8a\\_Xo/ISEmc-41NgFmYSt0jER2wA/view?utm\\_content=DAFIOE8a\\_Xo&utm\\_campaign=designshare&utm\\_medium=link2&utm\\_source=sharebutton](https://www.canva.com/design/DAFIOE8a_Xo/ISEmc-41NgFmYSt0jER2wA/view?utm_content=DAFIOE8a_Xo&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton)



# Interfaz de Usuario ARGON

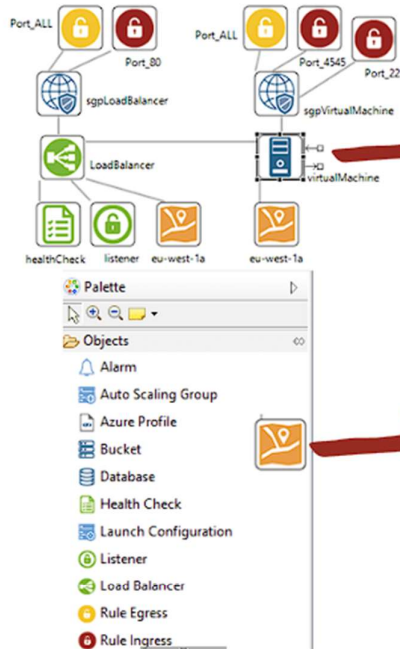
The screenshot displays the Eclipse IDE interface for the ARGON project. The main workspace shows a diagram of the system architecture. The diagram includes a LoadBalancer component connected to a sgpVirtualMachine component, which is associated with two eu-west-1 instances. The diagram also shows a healthCheck component connected to the LoadBalancer. The Properties panel is open, showing the details for a VirtualMachine component. The Properties panel has two tabs: Core and Appearance. The Core tab is selected, and the following table shows the properties and their values:

Property	Value
Count	1 uewfuib
Group	Security Group sgpVirtualMachine
Image	ami-Od7c8dde348d3b09f
Instance type	t2.micro
Name	virtualMachine

At the bottom of the IDE, a red error message is visible: "NumberFormatException: For input string: \"uewfuib\"".

# Principales Problemas de Usabilidad

(11 de 63 Problemas Encontrados)



## MODELADO

- 1.-NO PREVIENE EL ERROR:** Al enlazar elementos con las 2 flechas.
- 2.-VISIBILIDAD ERRONEA:** Sobreposición de texto y flechas.
- 3.-SIN ESTADO:** No se indica un estado bueno ni malo a un elemento mal configurado.

## ÍCONOS

- 4.-POCA VISIBILIDAD:** No son visibles a un metro de distancia.
- 5.-CONFUSIÓN:** No son intuitivos, ni dejan claro su funcionamiento.
- 6.-SIN DISEÑO:** Los colores no siguen una paleta en específico y se usan aleatoriamente.



**7.-NO EXITE DOCUMENTACIÓN NI INFORMACIÓN ADICIONAL QUE AYUDE A SOLVENTAR LAS DUDAS DEL USUARIO**

## MENSAJES DE ERROR

- 8.-CONFUSIÓN:** NO hablan el lenguaje del usuario, nada amigable al usuario.
- 9.-SIN AYUDA:** NO indican una posible solución.

## ENTRADA DE DATOS

- 10.-CONFUSIÓN:** NO indican el tipo de dato esperado.
- 11.-SIN RESTRICCIÓN:** NO indican los campos que son opcionales u obligatorios.