UNIVERSIDAD TECNICA
FEDERICO SANTA MARIA

DEPARTAMENTO
DE INFORMÁTICA

Propuesta de un modelo de negocio basado en SaaS a partir de un Free Open Source Software: Caso AuditForge

José Manuel Llanos Gaete
jose.llanosg@usm.cl

Profesor Guía:
Mauricio Figueroa Colarte

Profesor Correferente:
Pedro Francisco Godoy Barrera

Resumen: El presente trabajo desarrolla una propuesta de modelo de negocio en modalidad Software como Servicio (SaaS) basado en software de código abierto, con el propósito de la generación automatizada de reportes para servicios de pentesting. Para la formulación del modelo, se emplearon metodologías como la estimación de mercado (TAM, SAM, SOM) y el marco Lean Canvas, complementadas con un análisis de flujo de caja para evaluar la viabilidad financiera. La validación mediante encuestas a profesionales del sector confirmó la viabilidad del proyecto: un 53 % mostró disposición a adquirir la solución, con una utilidad percibida de 3.9/5. Se identificaron necesidades clave, como la preferencia por pruebas gratuitas (58 %) y la priorización de seguridad e integraciones. Los resultados respaldan iterar el producto, enfocándose en pymes (72 % de interés vs. 17 % en grandes empresas). El plan de acción incluye fases de validación con usuarios, lanzamiento inicial, escalamiento técnico y crecimiento sostenido, con métricas claras que permitan identificar el éxito o fracaso de casa fase. Entre las principales limitaciones se identificó la escasez de datos específicos del nicho y el desafío que representa la adquisición de los primeros clientes. Se concluye que es factible desarrollar un modelo de negocio rentable a partir de software de código abierto bajo ciertas condiciones, y este trabajo sienta una base de referencia que combina áreas de negocio, software libre y ciberseguridad para futuros emprendimientos tecnológicos.

Palabras Clave: modelo emprendedor, *SaaS*, emprendimiento digital, *hacking* ético.

1. Introducción

1.1. Contexto, motivación y problemática

En el mundo actual, cada vez son más las empresas que integran tecnología digital a su negocio (proceso conocido como transformación digital), dependiendo a la vez en mayor manera de la seguridad de esta. Cada cierto tiempo sale a la luz algún escándalo relacionado: ataques de *ramsomware*, denegación de servicios, liberación de información privada de los usuarios (eventos conocidos como *leaks*), entre otros tipos de ataques que generan daños tanto como para los afectados directos (usuarios, clientes); como para las empresas que proveen los servicios tecnológicos. Esto puede significar desde el deterioro de la reputación de estos proveedores hasta el cierre de sus operaciones debido a juicios impagables o pérdida de clientes.

Un ejemplo de esto es lo ocurrido a *IFX Networks*, un proveedor estadounidense de servicios de tecnología, víctima de un ciberataque en 2023 [1] donde 762 empresas latinoamericanas fueron afectadas; comprometiendo la disponibilidad y confidencialidad de muchos servicios públicos en el suceso.

El *pentesting* es una práctica en la cual, de manera consensuada, se realizan pruebas de seguridad en un sistema informático con el fin de encontrar vulnerabilidades y reportarlas al administrador de dicho sistema, para que este pueda tomar conciencia de las mismas y realizar los arreglos necesarios. Esto permite minimizar la posibilidad de eventos como los descritos previamente, reforzando el sistema ante usos con fines maliciosos.

El proceso de *pentesting* (conocido también como *ethical hacking*) tiene varias etapas (ver figura 1), dependiendo de la metodología que se utilice, pero siempre concluye con la elaboración de un informe escrito el cual detalla todos los hallazgos de vulnerabilidades: su alcance, su severidad, la descripción de la misma,

CONSTANCIA DE VALIDACIÓN Y CONFIDENCIALIDAD DE MONOGRAFÍA A REPOSITORIO ACADÉMICO

1.- IDENTIFICACIÓN DEL TRABAJO ACADÉMICO

Tipo de monografía (marcar una opción): ☒ Memoria o trabajo de título; ☐ Tesis de Postgrado;

Título del trabajo: Propuesta de un modelo de negocio basado en SaaS a partir de un Free Open Source Software: Caso AuditForge

Nombre del candidato(a): José Manuel Llanos Gaete

Carrera / Grado: Ingeniería Civil Informática

Campus: Casa Central ; **Departamento**: Informática

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Yo, Mauricio Figueroa Colarte, en mi calidad de profesor(a) guía/director(a) del trabajo académico mencionado anteriormente **DEJO CONSTANCIA** que:

- He revisado esta versión del documento y corresponde a la versión final aprobada del trabajo.
- El trabajo cumple con los requisitos académicos y de formato establecidos por la institución

3.- EVALUACIÓN DE CONFIDENCIALIDAD POR PROPIEDAD INDUSTRIAL

☒ El trabajo **NO contiene información que amerite confidencialidad** y puede ser publicado de inmediato en repositorio con acceso abierto.

☐ El trabajo **CONTIENE** información con potenciales implicancias de propiedad industrial o intelectual y requiere un periodo de confidencialidad (embargo) por:

☐ 6 meses; ☐ 12 meses; ☐ 2 años; ☐ 3 años; ☐ 5 años; ☐ 10 años

Fundamentación de la necesidad de confidencialidad (obligatorio si se solicita embargo):

4.- FIRMAS

Profesor(a) guía o director(a) de memoria o tesis:

Fecha: 28/07/2025 ; **Firma:** _____

Estudiante o Candidato(a):

Fecha: 28/07/2025 ; **Firma:** _____

Este formulario debe ser insertado como página 2 de la memoria o tesis, completado y firmado por estudiante y profesor(a) antes de la entrega en portal PRISMA de Biblioteca USM.

el como explotarla y como remediarla, entre otros campos que deben ser redactados por el *pentester*. Este proceso puede resultar engorroso, tomando una buena parte del tiempo del servicio de *pentesting* en razón de su dificultad, además de ser tedioso para las organizaciones que se dediquen al rubro.

Figura 1: Etapas del ethical hacking. Tomado de Fynd Academy [2]

El equipo DevForge de la feria de software UTFSM edición XXXII (2024)[1] identificó este dolor en los auditores de ciberseguridad (o *ethical hackers*), quienes deben constantemente lidiar con la fase de reportes en cada servicio de auditoría.

Para dar solución al problema encontrado, se desarrolló **AuditForge**: una plataforma web de generación de reportes de *ethical hacking* de manera automatizada basada en plantillas. En la interfaz web, el usuario genera vulnerabilidades/hallazgos quien luego de relacionarlos a un servicio/auditoría y llenar algunos detalles, puede exportar un informe en un formato editable (.docx, .odt, etc) o en un formato de presentación (pdf); para su posterior edición o entrega al ente objetivo del servicio de ethical hacking. Los datos almacenados permiten generar gráficos que resultan beneficiosos tanto a los auditores como a los clientes de estos, ya que de aquí se obtiene información relevante respecto a las tendencias en los hallazgos y servicios, así como se integra una herramienta de recomendación mediante IA para aumentar la productividad de los usuarios.

La aplicación fue planteada como FOSS[2], siendo publicado de forma completamente libre en GitHub[3]. Tras completar las características planificadas para la feria de software, se recibieron diferentes *issues* de usuarios que estaban probando y desplegando el software, lo que fue indicativo de un buen recibimiento por parte del sector productivo. El desarrollo fue bastante valorado por la comunidad de pentesting, quienes validaron su funcionalidad y utilidad, participando en pruebas de UI/UX donde se obtuvieron buenos comentarios.

Con un sistema de donaciones como método de financiamiento, este modelo de negocio implica que los usuarios pueden acceder al software y a su código fuente de manera libre y gratuita, con la posibilidad de realizar una donación voluntaria para apoyar a los desarrolladores. Cómo es posible notar, este modelo no va orientado al lucro y difícilmente puede generar ganancias a la pre-empresa involucrada en su desarrollo.

La falta de ingresos planificada para la organización (ya que las donaciones son opcionales y por lo tanto esporádicas), desencadena en consecuencias negativas para su subsistencia: la baja motivación en el desarrollo

[1] Feria de Software UTFSM
[2] Free Open Source Software
[3] GitHub

	Dradis	Cyver Core	PwnDoc	**AuditForge**
Open Source	X	X	✓	✓
Facilidad de uso	X	X	X	✓
Offline	X	X	✓	✓
Costo	80-150 USD/usuario/mes	100-450+ EUR/anual	Costos por mantener infraestructura propia	Costos por mantener infraestructura propia

Cuadro 1: Comparativa preliminar de competidores de AuditForge.

es grave para la calidad del producto de software final, los desarrolladores pueden incurrir en malas prácticas para sacar adelante las *features* necesarias y cumplir con las metas requeridas, sin considerar el costo a largo plazo. El proyecto entra en alto riesgo de abandono, ya que al no haber incentivos económicos, el equipo pierde interés en el desarrollo posterior a la feria y opta por abandonarlo, perdiendo una posible fuente de ingresos, pero también de experiencia laboral y empresarial; siendo estas de mucho valor en proyectos emergentes con una solución con innovación validada (durante la feria de software) como lo es AuditForge.

Dados todos estos efectos, se puede concluir que el potencial del producto final de la asignatura se pierde casi totalmente, ya que el software dejaría de recibir actualizaciones y soporte; y el proyecto se convertiría en un repositorio abandonado.

1.2. Definición del problema

Durante el planteamiento de AuditForge, se realizó una comparación de los posibles competidores del producto, consolidada en la tabla 1. Se seleccionaron los más similares a las funcionalidades propuestas para el producto. En esta se puede evidenciar que el producto original tiene la capacidad de sobrepasar a sus competidores dada su facilidad de uso, su cualidad *open-source* y su modalidad *offline*, pero que puede verse restringida su implementación en caso de no contar con la infraestructura necesaria para llevarla a cabo.

Es por esto que se considera que el encontrar una manera de hacer rentable el proyecto, haciendo las modificaciones debidas, sería beneficioso tanto para la organización encargada del proyecto así como para los posibles clientes interesados en su contratación.

En esa linea de trabajo es donde se identifica que el problema principal es encontrar una forma de transicionar el modelo de negocios del software *open-source* a uno que permita su subsistencia económica en el tiempo, problema al cual ya se han enfrentado grandes ponentes del software open-source: MongoDB, GitLab, WordPress; quienes han transformado desarrollos *open-source* en el centro de su modelo de negocios. Como ellos, hay muchísimos ejemplos de éxito que serán utilizados como inspiración para lograr los objetivos que se plantean en la sub-sección a continuación.

1.3. Objetivos

1.3.1. Objetivo general

El objetivo general consiste en proponer un modelo de negocio en modalidad *SaaS* a partir del *Free Open Source Software* AuditForge.

1.3.2. Objetivos específicos

Los objetivos específicos son:

1. Analizar las características del software AuditForge y el contexto de mercado actual para fundamentar la viabilidad de un modelo SaaS.

2. Diseñar un modelo de negocio bajo la modalidad SaaS adaptado a las necesidades de los usuarios objetivos.

3. Documentar las características y componentes del modelo propuesto utilizando herramientas como Lean Canvas.

4. Elaborar un plan de acción estratégico para la implementación del modelo de negocio en una futura etapa comercial.

1.4. Contenido del informe

En la sección 2 se hace la revisión del marco teórico. Se exploran técnicas para analizar y documentar el modelo de negocio, así como definiciones sobre software y open source, cloud computing y ciberseguridad. A continuación, en la sección 3 del estado del arte se referencian algunos trabajos sobre la feria de software que se han acercado de cierta forma a los objetivos planteados, así como a las técnicas utilizadas por las empresas que han basado su operación en software open-source. En la sección 4, de desarrollo, se realiza el análisis TAM, SAM, SOM y el Lean Canvas, lo cual desemboca en el flujo de caja presente en los anexos. En la sección 5, se realiza la validación. Para esto se detalla la encuesta realizada para validar el modelo de negocio, y el plan de acción que se apoya en la retroalimentación dada por los encuestados. La sección 6 concluye el trabajo detallando el alcance del mismo, sus limitaciones y el posible trabajo futuro, así como el impacto de este para los futuros lectores y su autor.

2. Marco teórico

2.1. Modelo de negocios

Un modelo de negocio se puede definir como "una representación abstracta, sea conceptual, textual y/o gráfica, de todos los arreglos arquitecturales, co-operacionales y financieros diseñados y desarrollados por una organización en el actualmente y en el futuro, así como de los productos y/o servicios clave que la organización ofrece o va a ofrecer, basado en aquellos arreglos que son necesarios para alcanzar su meta estratégica y sus objetivos" [3].

2.1.1. Business Model Canvas

Yendo a un concepto más concreto, en [4] se define un método para analizar modelos de negocio existentes, o diseñar nuevos: "Business Model Canvas" (BMC). A partir de una plantilla constituida por 9 bloques, se describe el modelo de negocio: Actividades clave, Recursos clave, Socios clave, Oferta de valor, Segmentos de clientes, Canales y Relaciones con los clientes. Esta plantilla se rellena con la información en cada recuadro, revisando de manera iterativa su contenido. Permite tener una visión clara del modelo de negocio, con el fin de que cualquiera pueda revisarlo y entenderlo fácilmente.

2.1.2. Lean Canvas

A partir del BMC, llegamos al modelamiento que nos interesa: el "Lean Canvas" (ver figura 2). Este es propuesto como una siguiente iteración del business model canvas; pero enfocado en startups que aún no inician su operación y tienen una máxima incertidumbre [5]. Por lo tanto, esta forma de modelamiento puede ser muy útil para el caso de uso de este trabajo. Se centra en problemas, soluciones, métricas clave y ventaja competitiva para validar rápidamente la idea en un entorno de baja información.

La idea es poder visualizar el lienzo (canvas) de manera gráfica ya sea presencialmente en una pizarra o en alguna aplicación donde se pueda ir rellenando cada recuadro.

Figura 2: Ejemplo de Lean Canvas. Tomado de Innokabi [6]

A continuación se presenta una descripción breve de cada campo utilizado en el Lean Canvas. El orden no es casualidad, se recomienda rellenar el lienzo en la siguiente disposición:

- **Problema**: Es uno o más dolores que sufre nuestro segmento de clientes, y que nuestro producto/servicio busca resolver. En esta sección se deben analizar las alternativas existentes.
- **Segmento de clientes**: Representan a quienes queremos apuntar como servicio de manera general. Es importante **caracterizar** los "early adopters", segmento de clientes que es más probable que esté dispuesto a adoptar nuestro producto en una etapa temprana. Enfocarse en este tipo de clientes es de vital importancia ya que dirigirse al mercado de masas desde el principio puede ser muy peligroso.
- **Proposición única de valor**: Es lo que hace nuestro producto/servicio (descrito en una frase sencilla) para resolver el problema de nuestro cliente. Es la razón principal por la que un posible cliente debería comprarnos.
- **Solución**: Se listan las principales características de nuestro producto/servicio.
- **Canales**: Se detalla el como llegaremos a nuestro segmento de clientes, para que conozcan lo que ofrecemos. Puede ser publicidad, emails, contacto a través de RRSS, etc.
- **Flujo de ingresos**: El cómo se va a generar dinero. Se debe definir bien el como se cobrará a los clientes, que estrategia de cobro se usará, si se ofrecerá gratis en un principio, etc.
- **Estructura de costes**: Se deben listar todos los costos asociados a la operación de nuestra organización.
- **Métricas clave**: Se analizan posibles métricas que sirvan como indicadores sobre los cuales tomar decisiones y monitorear el rendimiento de nuestro producto/servicio.
- **Ventaja especial**: Se refleja en una sola frase el como nos vamos a diferenciar de la competencia. Es algo que dificilmente pueda ser alcanzado por los competidores; se suele referir a esto como "Algo que no se pueda comprar ni copiar".

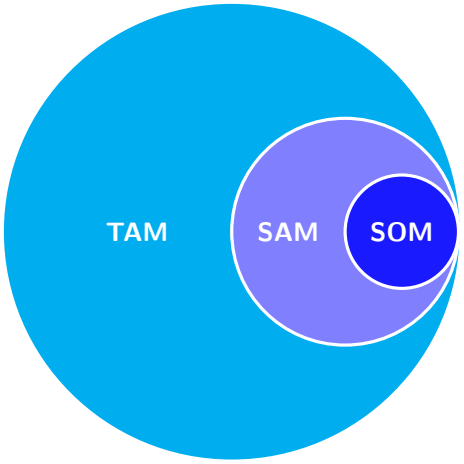

Figura 3: Diagrama TAM, SAM y SOM.

2.1.3. TAM, SAM y SOM

TAM (total adressable market), SAM (serviceable available market) y SOM (Serviceable obtainable market) [7], corresponden a los 3 elementos de una **técnica para estimar el mercado posible al que un producto o servicio puede apuntar a servir**.

- **TAM** Corresponde al mercado total al que deseamos ingresar, en otras palabras, el 100 % de los posibles clientes que se mueven por el mercado. Por ejemplo, si nuestro objetivo es vender autos, nuestro TAM es cualquier persona que pueda acceder a uno.

- **SAM** Corresponde al mercado que podemos servir **potencialmente** bajo nuestra disponibilidad de recursos y el modelo de producción actual, es decir, el máximo volumen de ingresos al que podemos aspirar. Siguiendo el ejemplo anterior, nuestro SAM puede ser todas las personas que pueden acceder a un vehículo dentro de una región geográfica al alcance de nuestro negocio.

- **SOM** Es el mercado al que nuestro producto o servicio puede acceder en un corto o mediano plazo. Esto permite estimar las ganancias potenciales que podríamos obtener en un periodo de tiempo, generalmente un año. En nuestro ejemplo de los autos, el SOM sería aquellas personas dispuestas a adquirir un vehículo próximamente. Es importante ser realistas con el SOM, ya que sobrestimar puede ser mucho más peligroso que subestimar.

En la figura 3 se hace énfasis en la relación entre TAM, SAM y SOM, siendo cada uno un subconjunto de su predecesor.

El análisis TAM, SAM y SOM presenta dos formas de ser realizado, bottom-up (de abajo a arriba) y top-down (de arriba a abajo) [8]. Bottom-up empieza por **datos** específicos del mercado **a los que tenemos acceso**, construyendo el SOM y posteriormente el SAM. Top-down empieza por el TAM de un modo más general y refina el SAM y el SOM a partir de segmentación de mercado y la toma de supuestos. La elección de una de estas dos metodologías depende de la cantidad de datos disponibles, la complejidad del mercado y el nivel de precisión requerida.

2.2. Software y Open Source

2.2.1. Software

La RAE[4] define el software como un "conjunto de programas, instrucciones y reglas para ejecutar ciertas tareas en una computadora u ordenador". El software es desarrollado utilizando comúnmente un lenguaje de programación, que forma el llamado código fuente y que consiste en un texto que puede ser leído, entendido y modificado de manera sencilla por el programador humano y a la vez es posible ejecutarlo en la computadora.

La propiedad intelectual del código fuente es materia de debate y gran controversia. Inicialmente el código fuente era tratado como un agregado mismo del hardware que las compañías vendían, para que los usuarios puedan hacer uso del mismo. Con el tiempo el software fue aumentando en complejidad y a la vez este era distribuido sin acceso al código fuente. A principios de los años ochenta casi todo el software era privativo, lo que significa que tiene un dueño que prohibe e impide la colaboración directa entre usuarios sobre el mismo [9]. Es en ese momento donde surgen movimientos como el proyecto GNU[5], a favor del "software libre".

2.2.2. Open-Source Software

El open-source es una clase de software que es desarrollado de forma comunitaria y con acceso libre a el código fuente, de manera en que cualquiera pueda verlo, modificarlo y distribuirlo de la forma en que le parezca conveniente [10]. El open-source es la contraparte del software propietario, el cual como se puede deducir solamente permite su utilización, más no la revisión del código fuente, modificación del mismo y distribución libre; todo esto amparado por licencias y ley.

El desarrollo del open-source ocurre comúnmente a partir de una necesidad de la comunidad, que es abordada de forma colaborativa mediante algún repositorio público al cual se puede acceder de forma libre y aportar según sea necesario. Además, dependiendo de la licencia que el proyecto posea, normalmente se puede "bifurcar" el repositorio, con el fin de llevar una linea de desarrollo aparte a la del proyecto original.

Open-source como concepto se origina del "software libre", pero posteriormente se propuso el término open-source ya que la acepción original podía generar confusión a los usuarios respecto al significado de "libre" (*free*), ya que esta palabra del idioma inglés significa tanto "libre" como "gratis", **y un proyecto open-source no significa necesariamente libre de costos**. La licencia GPL (General Public License, creada por la Free Software Foundation) permite cobrar por copias del software; siempre y cuando no se restrinjan las libertades del comprador [11].

2.3. Cloud Computing

2.3.1. Cloud Computing

El cloud computing se define como la disponibilidad de recursos computacionales como cómputo, almacenamiento y red bajo demanda, a través de internet, provistos por terceros. Esto ayuda a las organizaciones eliminando la carga que significa administrar la infraestructura física, pagando solamente por el uso de la misma [12]. Existen tres tipos principales de Cloud Computing (nubes):

- **Nube pública**: En una nube pública los recursos son ofrecidos por externos, para ser utilizados bajo demanda y satisfacer las necesidades empresariales de sus clientes.
- **Nube privada**: Las nubes privadas son totalmente administradas por una organización, proporcionando mayor control y seguridad sobre el uso de los recursos y los datos almacenados en este. También denominado on-premise.
- **Nube híbrida**: Combina los tipos anteriores a necesidad del usuario.

[4] RAE - software
[5] Philosophy of the GNU Project

En este contexto nacen los llamados "Cloud Service Providers" (CSP), proveedores de servicios en la nube. Entre los más utilizados se encuentran AWS de Amazon, Azure de Microsoft y Google Cloud Platform de Google.

A su vez, existen diferentes formas de disponer los recursos en la nube como servicio: La infraestructura como servicio (IaaS), la plataforma como servicio (PaaS) y el software como servicio (SaaS). A continuación se describirán brevemente cada una de estas presentaciones.

2.3.2. IaaS

En un IaaS, el proveedor otorga acceso a su infraestructura, la cual es administrada por el mismo. El usuario se puede olvidar de todas las tareas relacionadas a la gestión de la virtualización, los servidores físicos, el almacenamiento y la red. El consumidor de IaaS se debe hacer cargo del sistema operativo y cualquier entorno intermedio entre el mismo y la aplicación que este despliegue en la infraestructura.

Algunos ejemplos de IaaS son los proveedores de nube pública, como AWS, Microsoft Azure y Google Cloud. La infraestructura que uno puede utilizar como servicio en dichos proveedores normalmente son ofrecidos como una capa de abstracción para infraestructura real, por ejemplo en forma de redes virtuales privadas (VPC) que simulan una red privada de computadores, proveyendo conectividad entre servicios y aislamiento lógico. Otro ejemplo son los sistemas de almacenamiento en forma de "buckets", que abstraen el almacenamiento permitiendo la subida y descarga de archivos; o los servicios de computación, donde se entregan de forma rápida y simplificada máquinas virtuales listas para ser utilizadas [13].

2.3.3. PaaS

El PaaS es un servicio donde se entrega acceso a una plataforma de aplicaciones utilizada principalmente por programadores, ya que dichas aplicaciones forman parte o soportan a otras aplicaciones, pero también pueden corresponder a plataformas de sistemas de gestión personalizables. El usuario puede prescindir de gestionar la infraestructura subyacente a la plataforma y solamente dedicarse a configurar y administrar la plataforma.

Algunos ejemplos pueden ser bases de datos en la nube, funciones en la nube, sistemas de gestión como SAP o Moodle en la nube; así como sistemas de infraestructura como Kubernetes administrados por el proveedor, de manera en que el usuario de Kubernetes se dedica a desplegar cargas de trabajo y no debe preocuparse de administrar los nodos o el plano de control.

2.3.4. SaaS

SaaS o Software-as-a-Service es una manera de distribuir software en la cual se ofrece el acceso al mismo a través de internet. La organización proveedora se hace cargo del manejo de infraestructura de manera en que el cliente simplemente ocupe el software sin preocuparse de administrar su correcto funcionamiento. Un SaaS se apoya del cloud computing (y es visto como una forma de este), ya que puede acceder a recursos de manera rápida, elástica y a bajo costo.

A continuación se incluye un diagrama (ver figura 4) que contrasta las diferencias en responsabilidad de las diferentes presentaciones de computación en la nube. *On-site* hace referencia a infraestructura física, generalmente dispuesta en las instalaciones de la compañía.

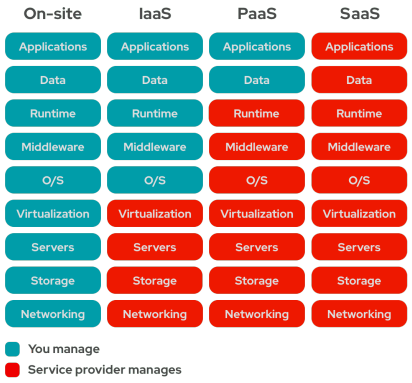

Figura 4: Diferencias en administración On-site, IaaS, PaaS y SaaS. Tomado de RedHat [14]

2.4. Ciberseguridad

2.4.1. Definición general

La ciberseguridad, según IBM [15], se "refiere a todas las tecnologías, prácticas y políticas para prevenir los ciberataques o mitigar su impacto. La ciberseguridad tiene como objetivo proteger los sistemas informáticos, las aplicaciones, los dispositivos, los datos, los activos financieros y las personas contra el ransomware y otros malware, las estafas de phishing, el robo de datos y otras ciberamenazas".

Matt Bishop [16] define la seguridad de un sistema en la siguiente proposición: "Si el sistema permanece en estados que son permitidos, y los usuarios solo pueden realizar acciones que son permitidas, entonces se puede afirmar que el sistema es seguro. Si el sistema puede pasar a un estado no permitido, o el usuario puede ejecutar una acción no permitida, entonces el sistema es no seguro".

El principio fundamental de la ciberseguridad es la tríada CIA: confidencialidad, integridad y disponibilidad (availability) [17]. La confidencialidad limita el acceso a información sensible, la integridad asegura que los datos de la organización permanezcan fidedignos y precisos, y la disponibilidad se encarga de proveer acceso oportuno al sistema y sus datos.

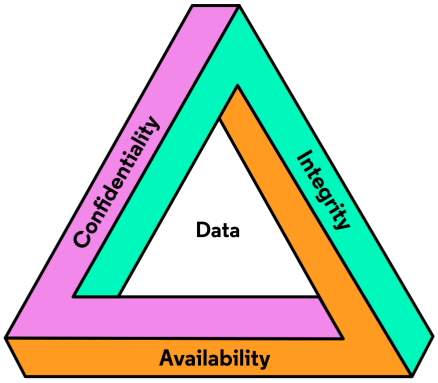

Figura 5: CIA triad. Tomado de codecademy [18]

2.4.2. Hacking

El hacking, en el contexto de la ciberseguridad, corresponde a la actividad de acceder de manera no autorizada a sistemas informáticos utilizando medios no convencionales, a menudo ilegales [19].

Esta acción es independiente de su propósito, y esto es clave en el contexto de este trabajo. Una persona que realiza hacking, llámese hacker, puede realizarlo con fines maliciosos: acceder a los datos de los usuarios, manipularlos en su almacenamiento o tránsito, o denegar el acceso a los usuarios al sistema; afectando como se puede notar a aspectos de la tríada CIA mencionada previamente. Este tipo de hacker es considerado como uno de "sombrero negro". En el otro extremo se encuentran los hackers llamados "sombrero blanco", aquellos que se mueven dentro de lo legal para ayudar a organizaciones a protegerse ante amenazas. Estos son conocidos comúnmente como "hackers éticos". Normalmente trabajan para compañías del mundo de la seguridad informática pero también pueden desarrollarse independientemente como consultores. **Uno de los servicios más comunes que pueden ofrecer los hackers éticos es el *pentesting*.**

Existen otras categorias en las cuales se pueden encasillar a los hackers, como puede ser el de "sombrero gris", que si bien realizan hacking sin permiso, su intención normalmente es ayudar a su objetivo; ya sea para obtener una compensación económica o un puesto de trabajo. También existen los "hacktivistas", que son aquellos que realizan hacking bajo una motivación ideológica, normalmente para generar una reacción en el público general. Un ejemplo de hacktivistas son la organización Anonymous[6].

2.5. Pentesting

El pentesting es "una prueba de seguridad que lanza un ciberataque simulado para encontrar vulnerabilidades en un sistema informático"[20]. Un ethical hacker que realiza pentesting se arma tanto de herramientas automatizadas como de métodos "manuales" para llevar al máximo la "penetración" en el sistema; con el fin de ponerse en la posición de un hacker malicioso y, además de documentar las vulnerabilidades, medir la robustez del sistema ante amenazas y el impacto que estas pueden generar. Un pentesting es más exhaustivo que una evaluación de vulnerabilidades automatizada.

El pentesting corresponde a una práctica habitual de quienes se desenvuelven en el hacking ético. Muchas veces se mezclan ambos conceptos, pero el pentesting en su definición más purista corresponde a un subconjunto de las labores de un ethical hacker. Esta práctica es cada vez más valorada, ya que las regulaciones de seguridad de los datos exigen ciertos controles de seguridad, incluso algunas explicitando la necesidad de pentesting como en PCI-DSS[7] v4.0.1 (norma para seguridad de datos de tarjetas de pago), requerimiento 11.4: "Se realizan regularmente pruebas de penetración externas e internas, y las vulnerabilidades explotables y debilidades de seguridad son corregidas". El gobierno de Estados Unidos recomienda tomar medidas como el pentesting para evitar ataques de ramsomware [21].

Un pentesting puede ser realizado como servicio por una compañía dedicada a la ciberseguridad, así como por un particular (freelancer) o puede ser concebido en un evento de "Bug Bounty", donde una organización permite a pentesters atacar a un ambiente controlado de la misma para recompensar (solamente) a aquellos que encuentren vulnerabilidades.

2.6. Tipos de Pentesting

2.6.1. Pentest de Aplicaciones

El pentest de aplicaciones se encarga de encontrar vulnerabilidades en aplicaciones de uso común, como lo son servicios web, API's, IOT, aplicaciones móviles, entre otros.

Una buena referencia, utilizada a menudo por pentesters, es el OWASP Top 10[8], que documenta periódicamente el top de vulnerabilidades más críticas encontradas comúnmente en aplicaciones.

[6] Anonymous (Wikipedia)
[7] Web PCI
[8] OWASP Top 10

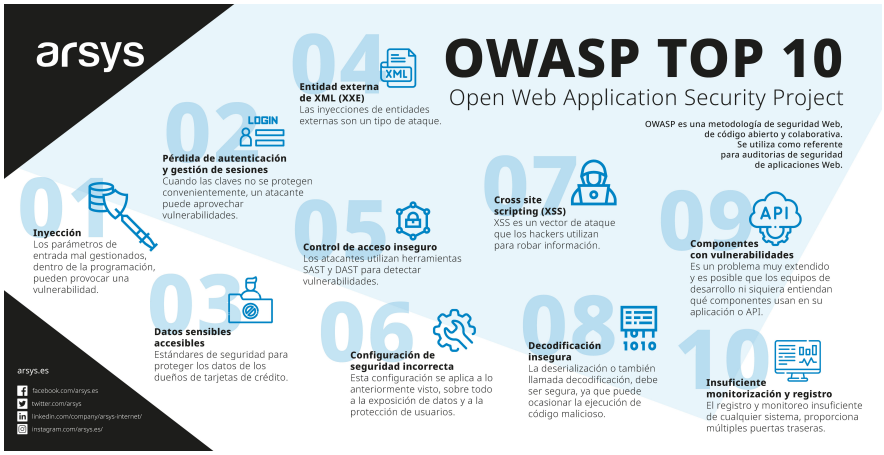

Figura 6: Infografía OWASP Top 10. Tomado de arsys [22]

2.6.2. Pentest de Red

El pentest de red pone a prueba, ya sea de manera externa o interna, a la red informática de la organización. A diferencia del pentest de aplicaciones, en este caso no se pone a prueba un activo en específico, si no que se abarca a toda la red en cuestión.

2.6.3. Pentest de Hardware

El pentest de hardware involucra aprovechar las interfaces de hardware y los canales de comunicación para encontrar vulnerabilidades **a nivel de dispositivo**. Un atacante en este escenario puede por ejemplo aprovechar vulnerabilidades en la configuración de la infraestructura utilizada por la organización para acceder sin autorización a datos de la misma.

2.6.4. Pentest de Personal

Los pentest de personal aprovechan la mala higiene de la ciberseguridad de los empleados de una organización para dejar en evidencia posibles vectores de ataques maliciosos. Comúnmente se utiliza el **phising**, que consiste en engañar mediante correos electrónicos, SMS u otros medios para que algún empleado exponga datos confidenciales de la organización, sin darse cuenta de que está siendo engañado.

También existen otros métodos como el "tailgating" [23], utilizado para engañar en persona a un miembro de la organización y hacer ingreso a las instalaciones de la misma sin la autorización debida, potencialmente entregando acceso al atacante a información privada dentro de la oficina.

2.7. Etapas del Pentesting

Existen diferentes estándares que definen etapas para realizar un pentesting con éxito. PTES[9] es una guía estándar para la ejecución de pruebas de penetración, diseñado para ser utilizado por profesionales del ethical hacking que ofrece una metodología clara y organizada para realizar servicios de pruebas de penetración [24]. Un pentesting que se apoya en PTES cuenta con las siguientes etapas:

- Pre-ataque: En esta fase se planifica con el cliente el proyecto: el alcance del pentest, los objetivos, y se resuelve cualquier duda que este pueda tener acerca del servicio.

[9] Web PTES

- Recolección de información: Se utilizan diferentes fuentes de información para acumularla y así tener una base por la cual poder empezar a identificar posibles vulnerabilidades.
- Modelado de Amenazas: En esta etapa se lleva a cabo un análisis sobre la información recolectada para así encontrar las amenazas que el sistema pueda tener.
- Análisis de Vulnerabilidades: En esta etapa se utilizan herramientas automatizadas o manuales para detectar vulnerabilidades activas (explotables) en el sistema.
- Explotación: Se lleva a cabo la explotación de las vulnerabilidades en caso de ser posible.
- Post-Explotación: Se analiza el impacto de la explotación realizada, analizando posibles datos sensibles y/o movimiento lateral, lo que corresponde a "escapar" del contexto actual hacia otros activos de similar "jerarquía". Por ejemplo, al lograr acceder a un computador de la organización sin autorización, un movimiento lateral sería acceder a otros computadores similares dentro de la red desde el computador original.
- Reportería: Esta fase final considera la documentación de los hallazgos en el servicio, elaborando un **informe presentable al cliente** que permita tomar conocimiento de lo que se encontró durante el pentesting. Según PTES, debe contar con dos secciones principales: el resumen ejecutivo, donde se describe en un alto nivel los resultados del pentesting, haciendo énfasis en los riesgos asociados a las vulnerabilidades encontradas; y el reporte técnico, donde se va al detalle en el alcance, la información, el método de ataque, su impacto y las medidas recomendadas para remediar cada vulnerabilidad.

3. Estado del arte

3.1. FESW y Emprendimiento

La Feria de Software (FESW) ha servido como un espacio propicio para el surgimiento de iniciativas de emprendimiento tecnológico. Diversos trabajos de titulación han surgido a partir de esta instancia, enfocados en soluciones innovadoras y comercializables.

En "Guía metodológica para desarrollar emprendimientos tic por estudiantes universitarios: Caso de estudio feria de software USM" [25] se trabaja una plataforma web para enseñar sobre emprendimiento a alumnos de la carrera de ingeniería civil informática. En dicho trabajo, una encuesta realizada a estudiantes de primer año reveló que **un 98 % manifestó algún interés en emprender**.

Acercándose un poco a los objetivos de este trabajo, "Reingeniería y modelo de negocios para un proyecto de la feria de software UTFSM - Caso de estudio: Smatch" [26], además de realizar una refactorización al software desarrollado en su feria de software, replantea el modelo de negocios utilizando business model canvas, culminando con un flujo de caja para estimar su desarrollo financiero.

El trabajo realizado para Smatch es muy relevante, ya que es uno de los pocos trabajos (si no el único) que toma un proyecto de feria y lo refactoriza haciendo un gran énfasis en su modelo de negocio.

3.2. Monetización de Open-source

Como se mencionó previamente, el open-source no significa libre de costos. Esto ha sido aprovechado por múltiples empresas que han basado su operación en su software open-source, quizás librando el acceso al mismo pero abordando otras formas de monetización para obtener rentabilidad.

Es interesante explorar cómo lo hacen otras empresas para obtener financiamiento a partir de sus propios proyectos open source. Diversos estudios han identificado al menos cuatro estrategias principales de monetización aplicables a proyectos de software libre o de código abierto (FOSS) [27]:

- **Open core**: El producto base (core) es accesible de forma libre para cualquiera que desee utilizarlo. Para obtener ganancias, existe una versión "premium" o "enterprise" (de pago) que ofrece diferentes ventajas (mayor soporte, features extra, etc) que son deseadas por las organizaciones.
- **Hosting/Soluciones Cloud**: En este modelo la organización encargada del proyecto ofrece una versión hosteada de pago, de manera que el usuario final no tenga que preocuparse de la infraestructura necesaria para su funcionamiento, ya que esta es administrada por la organización. Es necesario en algunos casos cambiar la licencia a una que impida comercializar libremente el producto, ya que otros podrían ofrecer el mismo servicio quitando clientes y ganancias a la organización encargada del software original.
- **Servicios profesionales**: Consiste en vender soporte técnico extendido, capacitación y consultoría, más no el acceso al software. RedHat es un famoso ejemplo del uso de este modelo de negocio.
- **Marketplace (mercado)**: Consiste en hacer de intermediario entre clientes y proveedores, ofreciendo estos últimos extensiones del producto original (el cual es open source). Un ejemplo de esto es el proyecto Android, con su tienda de aplicaciones que obtiene ganancias al intermediar la venta de software de terceros con los usuarios del sistema operativo.

En la práctica, muchas compañías combinan elementos de varios modelos; es importante elegir el más adecuado a la naturaleza del software [28]. Para este proyecto se opta por el enfoque "Hosting", dado que resulta el más compatible con las capacidades actuales de AuditForge y no requiere modificaciones estructurales profundas.

4. Desarrollo

4.1. Análisis TAM, SAM, SOM

Dado el limitado acceso a datos específicos del mercado objetivo, se empleará una estrategia de análisis de mercado top-down, comenzando con estimaciones globales y refinando progresivamente el alcance hacia segmentos más específicos.

4.1.1. Costo de AuditForge

Para hacer el análisis primero se requiere estimar el cobro que se realizará anualmente por cliente. El cálculo para TAM, SAM y SOM se realizará de la siguiente manera:

$$\text{cantidad de clientes} * \text{cobro anual} \tag{1}$$

El costo estimado mínimo en infraestructura es de 52 USD, para crear una instancia de VM de 2 cores y 8GB de RAM, con 100GB en disco de boot y 100GB extras para base de datos en el proveedor de nube Digital Ocean[10]. Esta instancia permite desplegar una **versión básica** y funcional de nuestro software para un equipo pequeño (hasta 10 personas).

Se tiene en cuenta además las horas hombre utilizadas en el despliegue de la infraestructura y configuración del software, estimadas en 4 horas máximo. Para este cálculo se tomará como base el sueldo de un ingeniero de sistemas junior, publicado por IT Hunters[11]. Con dicho valor nos da un costo de despliegue de 31 USD (aproximadamente).

Si consideramos además 4 horas de soporte gratuito mensual para nuestro cliente (los mismos 31 USD), nos queda una inversión inicial de $114,22 USD el primer mes y un gasto recurrente de $83,11 USD. En la tabla 2 se resume el análisis de costos realizado.

[10] DigitalOcean - Basic Droplets
[11] Guía Salarial IT HUNTER

Item	Descripción	Costo (USD)
Infraestructura	VM en DigitalOcean 2vCPU 8GB RAM y SSD de 100GB	52
HH Despliegue	4 horas, tiempo fijo (una sola vez)	31
HH Soporte	4 horas, tiempo gratuito mensual para el cliente	31

Cuadro 2: Tabla de costos para atender a un cliente.

Finalmente, podemos cobrar 200 USD por un plan básico de hasta 10 usuarios, que queda en **20 USD mensuales por usuario**, y en **2400 USD anuales por cliente**. Se escogió este valor tan bajo ya que para realizar el análisis es recomendado subestimar antes que sobre-estimar, sobre todo cuando se tiene poca información acerca del mercado, lo cual ocurre en nuestro caso de estudio.

Estos costos son descritos en la sección de "Estructura de Costos" del Lean Canvas, y formalizados en un flujo de caja.

4.1.2. Metodología

Existe poca información acerca de compañías de pentesting a nivel mundial. Si bien en algunas fuentes podemos encontrar por ejemplo un listado de las compañías certificadas por CREST[12] (una organización internacional de certificación en ciberseguridad), la cantidad de compañías es bastante reducida y especializada, lo que se aleja de nuestros clientes potenciales. Por otro lado, podríamos tomar como base para el TAM la cantidad de investigadores de ciberseguridad reportada en el landing page de HackerOne[13] (reconocida plataforma de bug-bounty), que corresponde a 2 millones; pero faltaría información para refinar el SAM y SOM.

Para realizar el análisis entonces se utilizará el motor de búsqueda de empresas de LinkedIn [29], que nos permite utilizar diferentes filtros para ir refinando la búsqueda, obteniendo la cantidad de empresas que cumplen con dichos filtros. Cabe aclarar que el buscador de LinkedIn no es una base de datos abierta, por lo que los resultados pueden fluctuar en el tiempo, pero para efectos de este análisis es suficiente para tener una aproximación.

4.1.3. TAM

Para el TAM se utilizará el valor obtenido en LinkedIn al buscar empresas, filtrando por categoría "Seguridad de redes y sistemas informáticos", lo que nos da un resultado de 60.000 empresas:

$$\text{TAM} = 60{,}000\ [\text{clientes}] * 2400\ [\text{USD}] = 144{,}000{,}000\ [\text{USD}] \tag{2}$$

4.1.4. SAM

Para el SAM, queremos refinar nuestro alcance de posibles empresas, y para ello reduciremos el espacio de búsqueda a empresas a las cuales podamos dar soporte de manera fluida, es decir, aquellas que se encuentren en un país donde se hable español (por simplicidad Chile, Argentina, Perú, Colombia y España); y que además cuenten con 1 a 10 empleados, ya que esto significaría que serían candidatos para nuestro plan básico. Aplicando estos filtros, la cantidad se reduce a 888 resultados de empresas:

$$\text{SAM} = 888\ [\text{clientes}] * 2400\ [\text{USD}] = 2{,}131{,}200\ [\text{USD}] \tag{3}$$

[12] Members - CREST
[13] HackerOne

4.1.5. SOM

Bajo nuestro modelo actual, podríamos considerar 180 horas de soporte mensuales gratuitas para el cliente, en caso de contar con un ingeniero de plataforma full-time. Por lo tanto, sin necesidad de escalar el tamaño del equipo y manteniendo nuestro costo definido previamente, podríamos servir a un máximo de 45 clientes sin escapar de las horas designadas para nuestro trabajador.

$$\text{SOM} = 45\ [\text{clientes}] * 2400\ [\text{USD}] = 108{,}000\ [\text{USD}] \tag{4}$$

Dado este análisis, AuditForge podría obtener ingresos de alrededor de 108.000 USD anuales en el corto-mediano plazo con 1 solo empleado (ingeniero de plataforma) y **cobrando un valor bajo** por usuario.

4.2. Lean Canvas

Para desarrollar el Lean Canvas se tomó una versión preliminar realizada durante la asignatura de Feria de Software. En su versión inicial, el modelo de negocio se sustentaba únicamente en donaciones voluntarias. Aunque esta estrategia era funcional en el contexto de desarrollo estudiantil, no aseguraba ingresos sostenibles. La transición al modelo SaaS representa un avance estratégico hacia la viabilidad económica de la solución.

A continuación se detalla cada elemento del Lean Canvas actualizado, que modela AuditForge como un SaaS que persiste como open-source.

4.2.1. Problema

- **Tiempos de procesos de ofimática altos y engorrosos.** La auditoría, en sus etapas explicada previamente, finaliza con la elaboración de un reporte entregable para el cliente. Dicho reporte, de no contar con una herramienta especializada (como AuditForge), debe ser elaborado utilizando algún editor de texto ya sea online (Google Drive - Documentos) o local (LibreOffice). Esto convierte esta última etapa en algo muy engorroso y que puede tomar un buen pedazo del tiempo del servicio de pentesting.

- **Falta de insights.** Siguiendo el punto anterior, al escribirlo directamente en un documento (.docx u otro formato), no se generan datos asociados (vulnerabilidades, métricas, etc) más allá del documento mismo. Al entregar el documento estos datos se "pierden" y no se puede tener un histórico de vulnerabilidades de un cliente, por ejemplo.

- **Carencia de colaboratividad.** El usar una plataforma "manual" de escritura de documentos dificulta o incluso impide la colaboratividad, por lo que si dos o más Pentesters deben redactar un informe en conjunto, se vuelve aún más dificultoso.

En la tabla 3 se incluye un análisis de las features y el costo de los competidores **(alternativas existentes)** más similares AuditForge. Es necesario aclarar que varios de estos proyectos tienen un formato open-core. La versión base no es considerada en esta comparativa, en la feature "open-source", ya que AuditForge ofrece **todas** sus características como open-source.

4.2.2. Segmento de clientes

- **Compañías de pentesting**: Corresponde a compañías dedicadas al rubro del pentesting, las cuales tienen uno o más pentesters que realizan servicios de hacking ético bajo contrato a otras compañías.

- **Bug bounty hunters**: Son un tipo de pentester que participa en programas de bug bounty, donde deben realizar pentesting a compañías y son recompensados monetariamente solo aquellos que hacen hallazgos de vulnerabilidades dentro de un scope definido.

	Dradis	Cyver core	PlexTrac	Faraday	AuditForge
Open-source	X	X	X	X	✓
Colaborativdad	✓	✓	✓	✓	✓
Dashboards	✓	✓	X	X	✓
Recomendación de CWE y CVSS	X	X	X	X	✓
Costo mensual por usuario (Dólares)	80	99	Oculto. Es necesario contactar a ventas.	134	A establecer

Cuadro 3: Comparativa de features y costos de AuditForge y sus competidores.

- **Freelancers**: Son pentesters que trabajan como freelancers realizando pentestings a compañías que lo requieran.

Los early adopters serían, dentro de mismo segmento de clientes, aquellos pentesters (o compañías de pentesting) emergentes, ya que estarían dispuestos a contratar un servicio que está en sus primeras fases con tal de tener una ventaja ante su propia competencia y a bajo costo.

4.2.3. Proposición única de valor

El servicio permitirá mejorar los tiempos de generación de informes, sin la preocupación de utilizar herramientas de terceros imposibles de auditar (al ser open source) y aumentando la facilidad de uso, con herramientas desarrolladas para agilizar aún más la velocidad de generación de reportes.

4.2.4. Solución

- **Generación de informes** de auditorías de forma automatizada: Esto permitira a los auditores concentrarse en el pentesting y no en redactar un documento, al simplemente tener que rellenar campos en la plataforma web cada vez que encuentren alguna vulnerabilidad.
- Creación de un **dashboard** con información esencial para el cliente y auditor: Este dashboard permite tener insights a partir del histórico de vulnerabilidades de un cliente. Con esta informactión útil el pentester (auditor) y el cliente auditado, ellos pueden reconocer patrones de vulnerabilidades, áreas de mejora, entre otros.
- **Sistema de recomendación utilizando IA**: Este sistema está integrado en la interfaz web de AuditForge y permite rellenar campos repetitivos (CWE[14] y CVSS[15]) mediante la recomendación de los mismos a partir de la descripción de una vulnerabilidad.

4.2.5. Canales

- **Contacto directo a través de LinkedIn**: Esta plataforma es muy útil ya que la gente expone sus habilidades, lo que nos permite reconocer a nuestros objetivos fácilmente. Las empresas también suelen exponer su rubro en esta red social, lo que facilita distinguir a quienes se debe contactar.
- **Anuncios en internet**: Se debe invertir en anuncios mediante alguna plataforma de ads.
- **Landing page**: Se debe utilizar la landing page como método de acercamiento de los usuarios hacia nuestro producto.

[14] What is CWE?
[15] CVSS

4.2.6. Flujo de ingresos

- **Administración de la infraestructura (SaaS)**: Corresponde al cobro por uso de la plataforma. Se debe idear una estrategia de cobro, ya sea por usuario, por plataforma, capacidad del despliegue, cantidad de reportes, etc.
- **Soporte**: El SaaS se entrega como está, pero si un cliente desea ayuda extra para configurar, debe ser cobrado de todas formas.

Los ingresos posibles fueron aproximados en la sección previa del análisis TAM, SAM, SOM.

4.2.7. Estructura de costes

- **Hosting** Incluye los costos asociados a la contratación de servicios en la nube para alojar AuditForge.
- **Administración de infraestructura**: Incluye los costos de las HH (horas hombre) dedicadas a administrar la infraestructura.
- **Soporte**: Incluye los costos de las HH (horas hombre) dedicadas a dar soporte a los clientes que lo requieran.
- **Marketing**: Son los gastos asociados a contratación de servicios para llegar a nuestros futuros clientes (anuncios, LinkedIn premium, entre otros).

En la tabla 2, del análisis TAM, SAM, SOM, se incluyen los costos estimados básicos de hosting, administración de infraestructura y soporte para AuditForge para el caso de un cliente.

Para los gastos de marketing se considerará empezar por un budget mensual de 300 USD en una campaña de Google Ads, e ir aumentando en razón de los resultados. Se puede utilizar LinkedIn de forma gratuita para contactar posibles clientes e ir generando contenido y actualizaciones acerca de la plataforma. La landing page puede ser hosteada en Vercel usando el plan PRO de 20 USD[16] y con el dominio auditforge.sh[17] que tiene un costo de 35 USD anuales.

Todos los costes e ingresos serán detallados en un **flujo de caja en los anexos** que además incluye impuestos, inversión inicial y capital de trabajo, con desglose de cada elemento, así como el VAN y el TIR. Esto nos permitirá aterrizar el valor final del plan para que la inversión sea rentable.

4.2.8. Métricas clave

- Reducción de tiempo en la generación de informes: Si se mide el tiempo destinado a la elaboración de informes, se podría observar una mejoría al utilizar AuditForge.
- Satisfacción del pentester: Es útil conocer la satisfacción del usuario final. Puede ser obtenida a través de encuestas.
- Satisfacción del cliente auditado: Lo mismo que para el punto anterior, en este caso es de gran utilidad ya que si el cliente auditado está satisfecho entonces se está agregando valor para el cliente que contrata AuditForge como SaaS.
- Cantidad de estrellas/forks en GitHub: Esto nos permite cuantificar la aceptación por parte de la comunidad open-source a AuditForge.

[16] Vercel - Pricing
[17] Namecheap - auditforge.sh

4.2.9. Ventaja especial

- Un equipo compuesto por pentesters interesados en el avance del proyecto: El equipo DevForge, encargado de AuditForge, está compuesto de pentesters y otros interesados en el desarrollo y la ciberseguridad, así como de especialistas en IA.

- La mejora de AuditForge gracias a las memorias articuladas con FESW: La nueva modalidad de memoria integrada con FESW[18] permitirá entregar nuevas versiones mejoradas del producto, aumentando sus capacidades y madurando su proceso de desarrollo. Se está trabajando en mejoras a los modelos de IA, mejoras a su arquitectura (como software), se implementaran mejoras a las pruebas del mismo y al ciclo de desarrollo seguro.

5. Validación del Modelo y Plan de acción

5.1. Metodología

Para validar nuestro modelo negocio se realizará una encuesta **(detallada en la sección a continuación)**. Esta encuesta está pensada tanto para pentesters dependientes (empleados), y para cargos de liderazgo que tengan peso en la toma de decisiones, como sería contratar AuditForge. Se utilizará Google Forms para recopilar los datos, y se contactará a personas que se dediquen profesionalmente al pentesting mediante LinkedIn, en su modalidad premium. Se escogió esta plataforma para realizar el contacto ya que facilita la búsqueda de profesionales a partir de palabras claves como "Pentester" o "Ethical Hacker".

El objetivo es validar tanto desde un punto de vista funcional (problema y solución entregada), así como de uno comercial (precios, factibilidad, fricciones con la implementación de la solución, etc). Las respuestas permitirán generar el plan de acción.

Para hacer el análisis de los resultados, se seguirá la siguiente metodología:

1. Se revisará entre los resultados, cuál es la tendencia general en torno al producto, es decir, el resultado que "satura" entre todas las respuestas; una síntesis de las principales coincidencias o hallazgos. Se segmentará en base a las primeras tres preguntas, a fin de **detectar si existen tendencias** entre los diferentes sectores.

2. Se observará cuales son las opiniones más relevantes o que destacan de entre todas las respuestas. Las preguntas 4, 7 y 9 nos permitirán conocer a detalle las necesidades de los encuestados, y así **conocer sobre que aspectos de la solución planteada se debe pivotar o iterar**.

3. Se hará un análisis general obtenido a partir de las respuestas obtenidas. Para validar desde un punto de vista funcional, se tomará el promedio de la pregunta 8: "Del 1 al 5, ¿Que tan útil encuentras una solución como AuditForge?" y se espera un valor mayor a 3. Para validar desde el punto de vista comercial el promedio de respuestas "Si" a la pregunta 11: "En caso de desear AuditForge, ¿Estarías dispuesto a pagar 20 USD mensuales por usuario?" debe ser mayor al 50 %. Para iterar, ambos valores esperados deben cumplirse, de lo contrario se debe pivotar.

Si la conclusión es **iterar**, se debe definir los aspectos a mejorar y una forma de medir dichas mejoras una vez implementadas. Si la decisión es **pivotar**, se debe declarar cuales elementos se cambiarán del enfoque original del producto, cual será el nuevo camino y por qué este será efectivo.

[18] Informática formaliza modalidad de titulación utilizando proyectos de Feria de Software

5.1.1. Preguntas

1. **¿Cual es tu rol laboral actual en ciberseguridad?** Pentester independiente (freelancer, bug bounty hunter, etc); Pentester dependiente (empleado); C-Level (CEO, CTO, etc); Otro (especifique).
2. **¿Cual es el tamaño de tu organización?**: 1 (soy independiente); 2-20 empleados; 21-50 empleados; 51-100 empleados; Más de 100 empleados.
3. **¿Utilizas actualmente alguna herramienta de pago para aumentar la productividad en tu negocio?** Si; No.
4. **Si tuvieras que contratar una herramienta (software) de productividad, ¿Cuáles son los 2 aspectos que más valoras?** (Elegir 2): Seguridad; Facilidad de uso; Obtención de métricas; Integraciones; Colaboratividad.
5. **En tu proceso de servicio de pentesting, ¿Que software utilizas para elaborar un reporte con tus hallazgos para tu cliente?** Microsoft Word/Google Docs; LibreOffice; Software de reportería (de terceros); Software propio/plataforma interna; Otro (especifique).
6. **Del 1 al 5, ¿Que tan satisfecho estás con tu proceso actual de elaboración de reportes de pentesting?**
7. **¿Que dificultad encuentras al tener que presentar los hallazgos de tus servicios? Ya sea en la elaboración del informe, entrega del mismo, etc.** Responder brevemente.

AuditForge es un software open source que automatiza la generación de informes a partir de plantillas. Los auditores (pentesters) simplemente deben detallar los hallazgos en la interfaz web junto con una descripción general del servicio de pentesting y exportar un informe en formato PDF o editable para hacer entrega del mismo. Incluye inteligencia artificial (no LLM, completamente privado) para recomendar tanto el CWE y CVSS correspondiente a partir de la descripción de la vulnerabilidad; y tomando como base los datos de cada informe genera dashboards informativos que permiten detectar tendencias en cada cliente y servicio.

8. **Del 1 al 5, ¿Que tan útil encuentras una solución como AuditForge?**
9. **Del 1 al 5, ¿Que tanto valoras cada feature de AuditForge?** Generación de informes; Dashboard informativo; Sistema de recomendación de CWE y CVSS; Trabajo colaborativo; Modelo Open-Source.
10. **¿En que rango de presupuesto (USD, mensual) para herramientas de este tipo se encuentra tu organización? Si eres independiente, contestar de todas formas.** Menos de 100 USD; 101 a 500 USD; Más de 500 USD; No hay presupuesto/no está definido;
11. **En caso de desear AuditForge, ¿Estarías dispuesto a pagar 20 USD mensuales por usuario?** Si; No (justifique).
12. **¿Qué te impediría probar Auditforge?** Costo; Curva de aprendizaje; Limitaciones de seguridad; Mejores alternativas; Otro (especifique).
13. **¿Qué necesitarías para convencerte de adoptar AuditForge?** Demo; Prueba gratuita; Testimonios; Garantía de servicio.

5.2. Resultados

Luego de buscar en LinkedIn a usuarios mediante las palabras claves previamente mencionadas, se seleccionó a aquellos que contasen con una cantidad razonable de contactos en LinkedIn (al menos 100), que pertenezcan a un país hispanohablante, y que en su perfil declarasen al menos 1 año de experiencia en algún cargo relacionado al pentesting. Se contactó en total a 30 personas, 17 de ellas respondieron el formulario.

Los resultados, los cuales se encuentran detallados en los anexos, reflejan una tendencia mayoritariamente positiva en la percepción del producto. Los encuestados pertenecen a todos los grupos que definimos en la pregunta 1, por lo que se obtuvieron diversas opiniones. Los tamaños de organización también son diversos.

El resultado predominante entre las respuestas recopiladas es una visión positiva hacia el producto. En el gráfico 7 podemos visualizar el promedio de valoración a cada feature de AuditForge, donde todas obtuvieron una valoración elevada (entre 3 y 5), y AuditForge obtuvo una puntuación de utilidad de 3.9 (pregunta 8); lo que supera el valor establecido para iterar de 3.0.

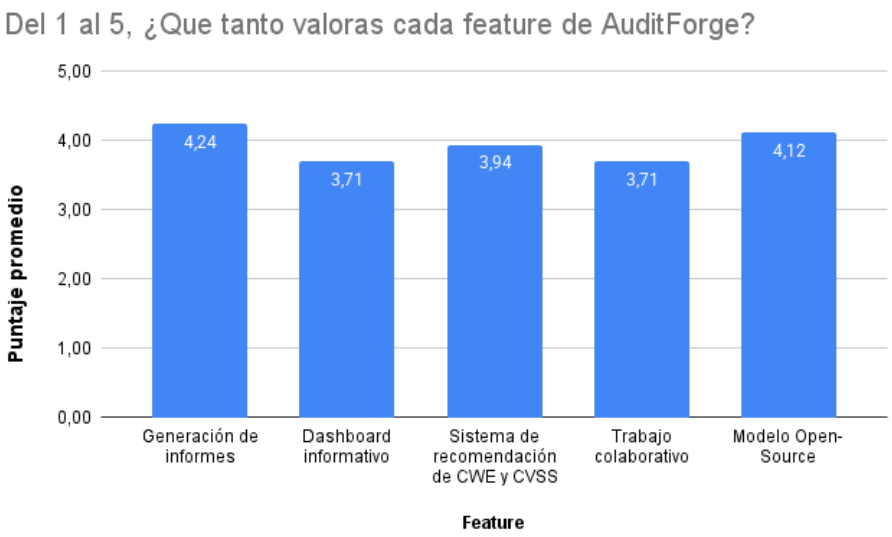

Figura 7: Puntaje promedio de valoración de cada feature de AuditForge.

Un 53 % estuvo dispuesto a adquirir AuditForge con el precio y las condiciones propuestas. En nuestro Lean Canvas apuntamos al segmento de clientes más pequeño o "emergente", ya que planteamos que ellos estarían dispuestos a adoptar una solución más barata con tal de obtener una ventaja comptetitiva. **Esto se refleja en las respuestas**; como se puede visualizar en el gráfico 8, al segmentar por tamaño de empresa "Más de 100 empleados" solo un 17 % (una sola persona) estaría dispuesto a adquirir AuditForge, mientras que el segmento contrario manifestó un 72 % de disposición a adquirir AuditForge.

Por otro lado, podemos observar una gran adopción de herramientas de productividad (en general). La mayoría (67 %) valora la seguridad de dicha herramienta, como era de esperar, pero además se puede ver cierta tendencia hacia la capacidad de integraciones (41 %), lo que va de la mano con algunas de las respuestas a la pregunta 7, en la cual dos de los encuestados manifestaron la necesidad de **obtener métricas de manera sencilla**:

- "Métricas estadísticas de evaluación debo integrarlo con Power BI u/o Tableau"
- "La obtención de métricas de forma automática, sin necesidad de registrar manualmente los valores para obtener las métricas."

Además destaca el hecho de que el 47 % utiliza actualmente herramientas manuales para la generación de sus reportes, lo que los perfila como potenciales usuarios de AuditForge, ya que una de las características principales es la generación automatizada de reportes de pentesting. El otro 53 % utiliza ya sea herramientas internas o externas para automatizar este proceso, lo que indica que este problema ya fue identificado previamente y ha dado lugar a la competencia de AuditForge.

Revisando la congruencia de las respuestas, destaca que aquellos que valoran la colaboratividad de un software de productividad (pregunta 4), también valoran el trabajo colaborativo de auditforge (pregunta 9) con un puntaje promedio de 4.4, 0.69 más de puntaje que el total de los encuestados. Otro aspecto relevante a destacar es que al comparar la satisfacción de su proceso de pentesting, aquellos que utilizan herramientas manuales promediaron 1.1 puntos más bajo (3.1) que aquellos que utilizan herramientas automatizadas, ya sea plataformas internas o herramientas de terceros (4.2), como se puede visualizar en el gráfico 9. Esta diferencia es otra evidencia de que existe una necesidad entre el público objetivo de AuditForge.

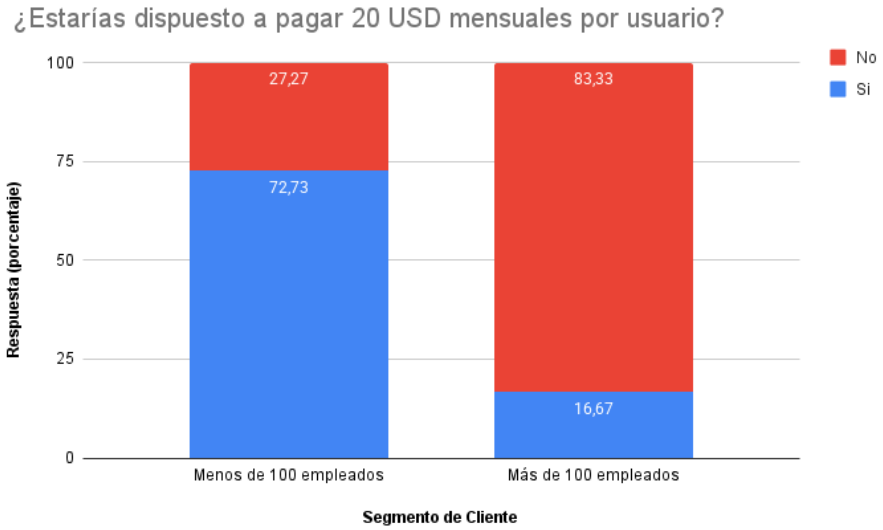

Figura 8: Porcentaje de adopción de AuditForge al realizar segmentación por tamaño de empresa.

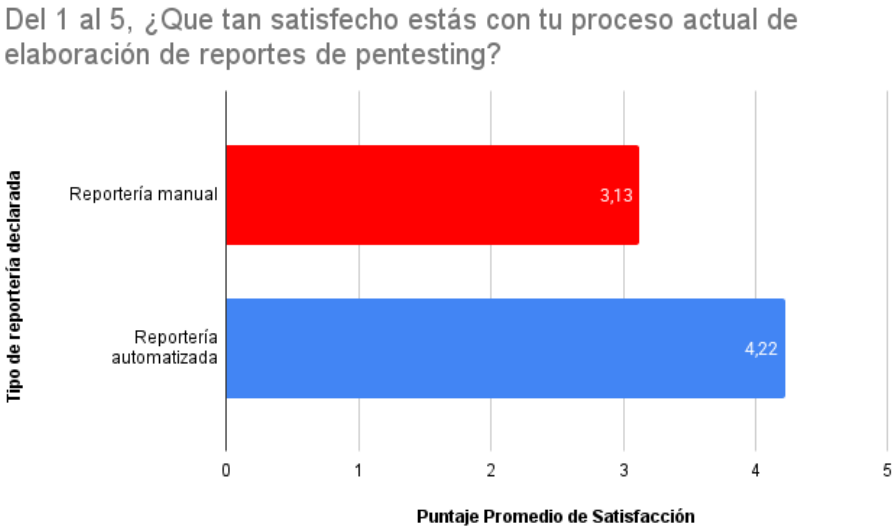

Figura 9: Puntaje promedio de satisfacción en el proceso de elaboración de reportes de pentesting, manual v/s automatizado.

Fase	Objetivo Principal	Hallazgo que motiva	Indicador de éxito
1	Validación con usuarios reales	58 % de los encuestados necesitarían una demo o una prueba gratuita.	80 % de los clientes de prueba están satisfechos con la solución
2	Atraer primeros usuarios	53 % de los encuestados está dispuesto a pagar	Se mantienen al menos 5 clientes de pago por más de un mes
3	Escalamiento	73 % de los pentesters pertenecientes a organizaciones pequeñas está dispuesto a pagar	1. Se logra implementar la nueva infraestructura 2. Se mantienen al menos 10 clientes de pago por más de un mes
4	Crecimiento sostenido	Un 67 % valora la seguridad y un 41 % valora la capacidad de integraciones	1. Encuestas sobre satisfacción de funcionalidad de integración puntúa sobre 80 % de aprobación 2. Se mantienen al menos 15 clientes de pago por más de un mes

Cuadro 4: Resumen del plan de acción.

Un 58 % de los encuestados manifestó que para convencerse de adoptar la solución necesitarían una demo o una prueba gratuita. Por lo tanto se identifica que el contacto directo con las funcionalidades de la plataforma es la mejor estrategia para la adopción de la misma por parte de los clientes.

En resumen, con estas respuestas podemos identificar 3 aspectos principales:

- El producto ha sido valorado positivamente y **se evidencia una necesidad concreta en el mercado objetivo**.
- Para fomentar la adopción de la solución, se sugiere ofrecer demostraciones en vivo y pruebas gratuitas.
- Se valora bastante la seguridad de la herramienta y sus integraciones, por lo que **se debe trabajar en mejorar la misma y desarrollar integraciones de la aplicación** para, por ejemplo, obtener métricas.

Dado que los resultados cumplen los criterios tanto para la validación "funcional" como para la comercial, **se decide iterar**. La iteración se centrará en el desarrollo de integraciones para obtener métricas, inicialmente. Estas mejoras serán evaluadas mediante nuevas encuestas centradas en la satisfacción de los usuarios con las nuevas funcionalidades.

5.3. Plan de acción

Para el plan de acción nos basaremos en las respuestas a la encuesta realizada. Este plan consta de una estructura de fases de implementación incremental. Esto permitirá tener los objetivos claros en cualquier momento y saber que acciones tomar en las situaciones a las que nos podamos enfrentar. La tabla 4 resume el plan de acción detallado a continuación.

5.3.1. Fase 1

Esta fase consta de la validación final del software con usuarios reales. Se debe contactar a posibles usuarios con los que agendar reuniones y explicarles el proceso, ofreciéndoles tanto la posibilidad de una

demo (en vivo) como de una prueba gratuita de 1 mes. Luego de la demo, o de completar su prueba gratuita, se les aplicará una encuesta de satisfacción; la cual debe puntuar con un 80 % de aprobación para pasar a la siguiente fase.

En la misma encuesta se debe consultar a los usuarios sobre que problemas tuvieron con la plataforma, potencialmente las preguntas 9, 11 y 12 de la encuesta inicial; para en caso de lograr el 80 % de aprobación, saber que elementos de la solución no fueron satisfactorias. Se debe remediar los elementos que fallaron y volver a aplicar la encuesta a los usuarios tras otro proceso de prueba o demo.

5.3.2. Fase 2

La fase 2 consta del lanzamiento inicial de la aplicación. En esta etapa debemos comenzar a atraer clientes mediante los canales definidos: campaña de Google Ads, LinkedIn y landing page. Es vital conseguir los primeros clientes, ya que su permanencia nos permitirá validar comercialmente el producto y obtener feedback de clientes comprometidos económicamente con el proyecto, teniendo en cuenta que los usuarios de la fase 1 pueden estar sesgados por el hecho de no tener que pagar.

El hallazgo clave obtenido del análisis es que el 53 % de los encuestados está dispuesto a pagar por la solución, bajo las condiciones descritas previamente. Se fija como meta obtener 5 clientes "persistentes", es decir, que se mantengan suscritos por más de 1 mes.

La retroalimentación constante es clave, se debe tomar encuestas a aquellos clientes que abandonen la plataforma, así como medir la satisfacción de aquellos que se mantengan. Si tras más de un semestre no se logra obtener ni mantener a los clientes, se deben analizar los resultados de las encuestas realizadas para reconocer los aspectos a mejorar o modificar.

5.3.3. Fase 3

En la tercera fase, debemos planificar el **escalamiento** de nuestro servicio. El hallazo relevante que se obtuvo del análisis es que aquellos clientes de compañías de menor tamaño (100 o menos empleados) tienen una disposición mucho mayor a adquirir el producto (+55 %), por lo que esta fase hará énfasis en dichos clientes.

Debemos implementar alternativas al despliegue en máquinas virtuales tradicionales, automatizando el mismo y utilizando un entorno de contenedores como Kubernetes para optimizar costos, el cual puede no ser rentable al contar con pocos clientes. Al utilizar una máquina virtual por cliente, estamos dedicando recursos solamente al sistema operativo (tanto CPU como RAM), esta cantidad "desperdiciada" crece linealmente con la cantidad de clientes. Un nodo de Kubernetes también reserva recursos para gestionar los servicios, pero al haber separación lógica se puede simplemente escalar verticalmente el nodo para ir agregando más clientes; evitando así ir "acumulando" recursos que son dedicados exclusivamente al sistema operativo de las máquinas virtuales tradicionales. **Utilizar este orquestador de contenedores nos permite atender a un segmento de clientes más pequeños** (menor a 10 usuarios), ya que simplemente podríamos desplegar AuditForge reduciendo los recursos destinados a cada contenedor. **El costo de la infraestructura es equivalente**, por lo que no se refleja un cambio en el flujo de caja.

Se fijará como meta para esta fase obtener 10 clientes persistentes, y completar la transición del despliegue de infraestructura. Un plazo tentativo para lograr esto es en un semestre.

En caso de no lograr las metas, se puede tener como alternativa el optimizar la infraestructura: seguir con VM's pero optimizar costos ya sea reduciendo la capacidad de las VM's o buscando otros proveedores. Si no se logra mantener 10 clientes, se puede enfocar en mantener a los actuales 5 y aprender de sus necesidades a partir de la retroalimentación que estos entreguen a través de las encuestas mencionadas en fases anteriores. Otra opción es realizar ofertas temporales a clientes de menor tamaño para atraerlos a la solución.

5.3.4. Fase 4

Esta fase consiste en el **crecimiento sostenido** de nuestro servicio. Al consolidarnos en el mercado debemos empezar a crecer como organización. Como logramos mantener una cantidad notable de clientes, podemos enfocarnos en las **necesidades manifestadas por los mismos en la encuesta de validación**. Como manifestaron principalmente su enfoque en la seguridad y en las integraciones, se hará una encuesta a los clientes actuales sobre las posibles mejoras que podría tener AuditForge en esa linea de trabajo. Posteriormente, se implementarán al proyecto open-source para que todos puedan disfrutar de las mejoras.

Se le pedirá a los clientes que evalúen la mejora, será considerará como exitosa si obtiene al menos un 80 % de aprobación. Además, se espera alcanzar una meta de 15 clientes; todo esto en un plazo de medio año.

6. Conclusiones

El trabajo realizado obtuvo un acercamiento inicial al modelo de negocios para un SaaS a partir de software open-source, en una industria de "nicho" como lo es la generación de reportes de servicios de pentesting. Se revisaron las estrategias utilizadas por compañías que basan su modelo de negocio en software de código abierto, destacando que, bajo ciertas condiciones, es posible obtener beneficios económicos a partir de esta modalidad.

Para la formulación del modelo de negocio, se aplicaron dos metodologías fundamentales: la estimación de mercado (TAM, SAM, SOM) y el marco Lean Canvas, que permitieron delimitar el alcance comercial y el valor diferencial de la propuesta. Además, se formalizaron los ingresos y costos de forma granular en un flujo de caja, lo que otorga una visión más clara desde el punto de vista financiero si se desea llevar el emprendimiento a la realidad. El análisis TAM, SAM, SOM y la comparación con las soluciones existentes permitieron cumplir los objetivos 1 y 2 del trabajo, y el desarrollo del Lean Canvas en su totalidad permite cumplir el objetivo numero 3.

Para validar la solución planteada se llevó a cabo una encuesta dirigida a posibles clientes y usuarios, con el fin de encontrar puntos claves que nos permitan decidir si iterar o pivotar, y elaborar un plan de acción. Los resultados de la encuesta aplicada a profesionales de pentesting validan tanto la viabilidad funcional como comercial de AuditForge, evidenciando una necesidad concreta en el mercado. Un 53 % de los encuestados mostró disposición a adquirir la solución bajo las condiciones propuestas, superando el umbral definido de 50 %; obteniendo además una valoración promedio de 3.9/5 en utilidad, superando el umbral mínimo de 3.0. La superación de ambos umbrales indica que la decisión a tomar para el plan de acción es iterar. Destaca especialmente la **receptividad del segmento de empresas emergentes** (menos de 100 empleados), donde un 72 % se mostró interesado, frente a solo un 17 % en empresas grandes (más de 100 empleados). La ejecución del plan de acción permitirá reducir las diferencias entre la solución planteada y las necesidades del mercado, con métricas claras para cada etapa.

Adicionalmente, se identificaron tres hallazgos clave:

1. Necesidad de automatización: El 47 % de los encuestados aún utiliza métodos manuales para generar reportes, y su satisfacción con su proceso de reportería es 1.1 puntos menor que quienes emplean herramientas automatizadas.

2. Estrategias de adopción: Un 58 % de los potenciales usuarios requiere demostraciones o pruebas gratuitas para adoptar la solución, lo que sugiere la importancia de un enfoque práctico en la comercialización.

3. Prioridades de desarrollo: La seguridad y las integraciones son aspectos críticos para los usuarios, lo que orienta la iteración inicial del producto.

El plan de acción se estructura en cuatro fases:

1. Validación final con usuarios reales mediante demos y pruebas gratuitas (meta: 80 % de satisfacción).

2. Lanzamiento inicial con captación de clientes a través de canales digitales (meta: 5 clientes persistentes).

3. Escalamiento técnico mediante migración a Kubernetes para optimizar costos y atender a empresas pequeñas (meta: 10 clientes persistentes).

4. Crecimiento sostenido, enfocado en mejorar integraciones y seguridad basado en feedback (meta: 15 clientes y 80 % de aprobación en nuevas funcionalidades).

Los resultados respaldan la iteración del producto, centrada en desarrollar funcionalidades para la obtención de métricas e integraciones, mientras se consolida un modelo comercial orientado a pymes del sector. La validación realizada y el desarrollo del plan de acción correspondiente permiten cumplir el objetivo específico número 4.

Una de las principales limitaciones detectadas fue la escasa disponibilidad de datos específicos del nicho de mercado abordado, lo que dificultó la estimación precisa de variables financieras y la obtención de comparables con soluciones similares. El modelo de negocio contempla además usuarios muy específicos (y de una demografía reducida), por lo que un limitante clave para la implementación del mismo puede ser la obtención de los primeros clientes.

Otro aspecto que se abordó de forma limitada en este trabajo fueron las técnicas de marketing, que son cruciales para hacerse un nombre como marca y acceder a los posibles clientes mediante los canales definidos en el Lean Canvas. Si bien se hace mención sobre la estrategia de marketing a muy grandes rasgos, se reconoce que podría ser mucho más detallado. Un estudio futuro podría abordar en profundidad tanto la parte teórica del marketing digital aplicado a un SaaS, así como una posible ejecución inicial del mismo. Otro estudio posterior podría enfocarse además en la parte legal del emprendimiento digital: la formación de la empresa, la contratación de personal o servicios, el proceso de venta, las necesidades del área tributaria, entre otros aspectos.

Esta investigación aborda áreas poco tratadas en trabajos de titulación previos en la carrera de Ingeniería Civil Informática, combinando elementos de negocio, software libre y plataformas SaaS aplicadas a soluciones técnicas del ámbito de ciberseguridad. Las técnicas utilizadas y sus resultados podrían ser utilizadas como referencia por los alumnos de feria de software. La elaboración de este proyecto permitió aplicar conocimientos técnicos y estratégicos en un contexto emprendedor, lo que constituye una base valiosa para futuros egresados interesados en iniciativas tecnológicas de base open-source. A nivel personal, el desarrollo de este tema fue de muchísimo valor, ya que particularmente no deseo ser empleado/dependiente toda mi vida; ya sea dentro de la informática o en un negocio cualquiera, mi intención es emprender. Por lo tanto, al desarrollar esta memoria pude acercarme a conceptos y técnicas que me serán útiles en el futuro para poder comenzar mi propio negocio.

7. Agradecimientos

Este trabajo significa la conclusión de una etapa, de años de esfuerzo, de buenos y malos momentos. Por lo mismo, me gustaría agradecer a todas las personas que estuvieron ahí, ya que de alguna manera se han vuelto una parte de mi. A mi familia, por darme el apoyo y la libertad para seguir adelante. A mis amigos, tanto aquellos que conozco hace muchos años (los de Villa Alemana, Coyhaique y los *geos*), así como aquellos que he conocido en el camino (amigos de la universidad, entre ellos *el club de los cojos*), ya que han significado un verdadero pilar en mi vida, y sin ellos jamás hubiera logrado llegar a este punto. A mis compañeros y profesores, muchos de ellos quienes han propiciado un ambiente que me motiva a dar lo mejor de mí. A las personas y empresas que he conocido en mis prácticas y trabajos, por darme la oportunidad de lucir mis virtudes y enfrentar mis defectos. A a mis gatos: Quesito, Jamona, Peluche y Manguito, además a Milaneso que me cuida desde el más allá; por servir como un gran apoyo emocional, su compañía ha sido invaluable hasta el día de hoy. Quisiera además agradecer a la música, por siempre manifestarse en mi vida en el momento adecuado.

Referencias

[1] J. Miranda, "Masivo ciberataque secuestra los datos de cientos de portales en Chile, Colombia y Panamá." https://www.biobiochile.cl/noticias/internacional/america-latina/2023/09/15/masivo-ciberataque-secuestra-los-datos-de-cientos-de-portales-en-chile-colombia-y-panama.shtml. [Última consulta el 2025-05-17].

[2] Fynd Academy, "Introduction to ethical hacking." https://www.fynd.academy/blog/introduction-to-ethical-hacking. [Última consulta el 2025-06-07].

[3] M. Al-Debei, R. El-Haddadeh, and D. Avison, "Defining the business model in the new world of digital business.," 14th Americas Conference on Information Systems, AMCIS 2008, vol. 3, p. 300, 01 2008.

[4] A. Osterwalder, Y. Pigneur, and T. Clark, Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers. Strategyzer series, Wiley, 2010.

[5] A. Maurya, Running Lean: Iterate from Plan A to a Plan That Works. Lean (O'Reilly), O'Reilly Media, Incorporated, 2012.

[6] Innokabi, "Lienzo lean canvas explicado paso a paso y con ejemplos." https://innokabi.com/lienzo-lean-canvas-el-lienzo-de-los-emprendedores/. [Última consulta el 2025-06-07].

[7] S. Blank, S. Blank, and B. Dorf, The Startup Owner's Manual: The Step-by-step Guide for Building a Great Company. No. v. 1, K&S Ranch, Incorporated, 2012.

[8] StartUpNV, "TAM SAM SOM Examples." https://startupnv.org/tam-sam-som-example/. [Última consulta el 2025-06-07].

[9] GNU, "Visión general del sistema gnu." https://www.gnu.org/gnu/gnu-history.es.html. [Última consulta el 2025-05-18].

[10] Red Hat, "Qué es el open source?." https://www.redhat.com/es/topics/open-source/what-is-open-source. [Última consulta el 2025-05-18].

[11] GNU, "Frequently asked questions about the gnu licenses." https://www.gnu.org/licenses/gpl-faq.en.html. [Última consulta el 2025-05-21].

[12] G. Cloud, "Qué es el cloud computing?." https://cloud.google.com/learn/what-is-cloud-computing?hl=es. [Última consulta el 2025-05-21].

[13] AWS, "Qué es la infraestructura como servicio (iaas)?." https://aws.amazon.com/es/what-is/iaas/. [Última consulta el 2025-05-31].

[14] Red Hat, "What is PaaS?." https://www.redhat.com/es/topics/cloud-computing/what-is-paas#ecosistemas-de-plataformas. [Última consulta el 2025-06-07].

[15] IBM, "Qué es la ciberseguridad?." https://www.ibm.com/es-es/topics/cybersecurity. [Última consulta el 2025-05-17].

[16] M. Bishop, "What is computer security?," IEEE Security & Privacy, vol. 1, no. 1, pp. 67–69, 2003.

[17] P. Toth, "Cybersecurity – a critical component of industry 4.0 implementation." https://www.nist.gov/blogs/manufacturing-innovation-blog/cybersecurity-critical-component-industry-40-implementation. [Última consulta el 2025-05-17].

[18] codecademy, "What is cybersecurity?." https://www.codecademy.com/learn/introduction-to-cybersecurity/modules/intro-cybersecurity-what-is-cybersecurity/cheatsheet. [Última consulta el 2025-06-07].

[19] IBM, "Qué es el hacking?." https://www.ibm.com/mx-es/topics/cyber-hacking. [Última consulta el 2025-05-17].

[20] IBM, "Qué es el pentesting?." https://www.ibm.com/mx-es/topics/penetration-testing. [Última consulta el 2025-05-17].

[21] A. Macias, "Business leaders must take urgent action to counter ransomware threat, white house warns in memo." https://www.cnbc.com/2021/06/03/ransomware-attacks-white-house-memo-urges-immediate-action-by-business.html. [Última consulta el 2025-05-18].

[22] arsys, "Owasp: ¿qué es y cómo usar esta metodología?." https://www.arsys.es/blog/owasp. [Última consulta el 2025-06-07].

[23] J. Santos, "Tailgating: ¿qué es y cómo puedes evitar que afecte a tu empresa?." https://www.deltaprotect.com/blog/tailgating-que-es. [Última consulta el 2025-05-18].

[24] Minery Report, "Penetration testing execution standard (ptes)." https://mineryreport.com/ciberseguridad/glosario/conceptos-generales/termino/penetration-testing-execution-standard-ptes/. [Última consulta el 2025-05-18].

[25] F. A. Da Silva Retamales, "GuÍa metodolÓgica para desarrollar emprendimientos tic por estudiantes universitarios: Caso de estudio feria de software usm," 2020.

[26] C. F. V. PEREIRA, "ReingenierÍa y modelo de negocios para un proyecto de la feria de software utfsm - caso de estudio: Smatch," 2019.

[27] O. Zinovyev, "How companies make millions on open source." https://blog.palark.com/open-source-business-models/. [Última consulta el 2025-05-21].

[28] I. Ghory, "The secrets of successful open source business models." https://insights.blossomcap.com/successful-open-source-business-models-2709e831e38a. [Última consulta el 2025-05-21].

[29] LinkedIn, "Resultados del buscador de empresas para el sector "seguridad de redes y sistemas informáticos"." https://www.linkedin.com/search/results/companies/?industryCompanyVertical=["118"], 2025. [Última consulta el 2025-06-07].

Anexos

Flujo de caja

VALOR PLAN BÁSICO	200
VALOR ANUAL	2400

Rentabilidad Exigida al Capital	12%

	AÑO	0	1	2	3	4	5
CANT. CLIENTES			5	10	15	20	45
INGRESO (USD)							
	Suscripción plan básico		$12.000,00	$24.000,00	$36.000,00	$48.000,00	$108.000,00
	INGRESOS		$12.000,00	$24.000,00	$36.000,00	$48.000,00	$108.000,00
EGRESO (USD)							
	Servidores plan básico		$3.120,00	$6.240,00	$9.360,00	$12.480,00	$28.080,00
	Marketing		$4.475,00	$4.475,00	$4.475,00	$4.475,00	$4.475,00
	Remuneraciones		$2.022,22	$4.044,44	$6.066,67	$8.088,89	$18.200,00
	Impuesto por venta		$3.000,00	$6.000,00	$9.000,00	$12.000,00	$27.000,00
	Adquisición PC	$1.500,00				$1.500,00	
	Inversión Capital de Trabajo	$6.308,61					
FLUJO AÑO (USD)		-$6.308,61	-$617,22	$3.240,56	$7.098,33	$9.456,11	$30.245,00
FLUJO ACUMULADO (USD)		-$6.308,61	-$8.425,83	$3.240,56	$7.098,33	$9.456,11	$28.745,00

VAN	$23.947,46
TIR	64%

Costos mensuales			
Ítem	**Descripción**	**Costo (USD)**	**Fuente**
Infraestructura	VM en DigitalOcean 2vCPU 8GB RAM y SSD de 100GB	$52,00	Droplet Pricing | DigitalOcean Volume Pricing | DigitalOcean
HH Soporte	4 horas, tiempo gratuito mensual para el cliente	$31,11	Cálculo valor HH
HH Despliegue	4 horas, tiempo fijo (una sola vez)	$31,11	Cálculo valor HH

Consideraciones

El valor hora se considera en dólares y aproximado por simplicidad

El primer año de un cliente es ligeramente más costoso por las HH de despliegue que se utilizan en su primer mes.

Costo anual infraestructura	**$624,00**
Costo anual remuneraciones (primer año cliente)	**$404,44**
Costo anual remuneraciones	**$373,33**

Cálculo valor HH			
Sueldo ingeniero de sistemas	**Horas mensuales jornada full time**	**Valor Hora**	**Fuente**
1400	180	7,777777778	Guía Salarial IT Hunter

Costos anuales			
Ítem	**Descripción**	**Costo (USD)**	**Fuente**
Vercel PRO	Hosting para landing page	$240,00	Vercel Pricing
auditforge.sh	Dominio para la landing page	$35,00	Namecheap | AuditForge
Anuncios de google	Presupuesto fijo mensual de 350 USD	$4.200,00	Google Ads | Bid & Budget

Costo total anual	**$4.475,00**

Resultados encuesta (hacer zoom)

¿Cual es tu rol laboral actual en ciberseguridad?	¿Cual es el tamaño de tu organización?	¿Utilizas actualmente alguna herramienta de pago para aumentar la productividad en tu negocio? Puede ser Jira, trello, etc...	Si tuvieras que contratar una herramienta (software) de productividad, ¿Cuáles son los 2 aspectos que más valoras?	En tu proceso de servicio de pentesting, ¿Que software utilizas para elaborar un reporte con tus hallazgos para tu cliente?	Del 1 al 5, ¿Que tan satisfecho estás con tu proceso actual de elaboración de reportes de pentesting?	¿Que dificultad encuentras al tener que presentar los hallazgos de tus servicios? Ya sea en la elaboración del informe, entrega del mismo, etc.	Del 1 al 5, ¿Que tan útil encuentras una solución como AuditForge?	Del 1 al 5, ¿Que tanto valoras cada feature de AuditForge? 1: Nada valorado 5: Muy valorado [Generación de informes]	Del 1 al 5, ¿Que tanto valoras cada feature de AuditForge? 1: Nada valorado 5: Muy valorado [Dashboard informativo]	Del 1 al 5, ¿Que tanto valoras cada feature de AuditForge? 1: Nada valorado 5: Muy valorado [Sistema de recomendación de CWE y CVSS]	Del 1 al 5, ¿Que tanto valoras cada feature de AuditForge? 1: Nada valorado 5: Muy valorado [Trabajo colaborativo]	Del 1 al 5, ¿Que tanto valoras cada feature de AuditForge? 1: Nada valorado 5: Muy valorado [Modelo Open-Source]	¿En que rango de presupuesto (USD, mensual) para herramientas de este tipo se encuentra tu organización? Si eres independiente, contestar de todas formas.	En caso de desear AuditForge, ¿Estarías dispuesto a pagar 20 USD mensuales por usuario? Esto incluye 4 horas de soporte mensual.	¿Por qué no estarías dispuesto a pagar 20 USD?	¿Qué te impediría adoptar Auditforge?	¿Qué necesitarías para convencerte de adoptar AuditForge?
Pentester independiente (freelancer, bug bounty hunter, etc)	Más de 100 empleados	Sí	Seguridad, Colaborativiad	Software propio/plataforma interna	3	En ocasiones se emplea demasiado tiempo en la elaboración de la reportería en lugar de utilizar esos esfuerzos en entregar valor en el producto final	4	5	5	5	5	5	No hay presupuesto/no está definido	Si		Limitaciones de seguridad	Prueba gratuita
Pentester dependiente (empleado)	Más de 100 empleados	Sí	Facilidad de uso, Integraciones	Software propio/plataforma interna	5	Por ahora ninguna realmente	4	5	3	4	5	5	No hay presupuesto/no está definido	No	20 USD sí, por usuario no	nada	Prueba gratuita
Pentester dependiente (empleado)	2-20 empleados	No	Seguridad, Colaborativiad	LibreOffice	2	Al tener que presentar los hallazgos, ninguno. El problema del proceso actual diría yo que se debe al proceso altamente manual de reportería, lo cual se asocia al software y metodología de la empresa.	5	5	3	5	5	5	No hay presupuesto/no está definido	Si		Curva de aprendizaje	Demo
Pentester dependiente (empleado)	2-20 empleados	Sí	Facilidad de uso, Integraciones	Word, LibreOffice, OnlyOffice, herramientas propias para complementar.	3	En la comunicación con el cliente a la hora de presentar ciertas vulnerabilidades, aunque se puede relacionar más a las habilidades blandas. Es encontrar la forma correcta de explicar vulnerabilidades complejas, o una cadena de vulnerabilidades.	4	3	4	3	5	5	No sé	Si		Que la generación de informes tenga errores visuales, tener que recurrir a editar cosas de forma manual	Prueba gratuita
C-Level (CEO, CTO, jefatura, etc)	Más de 100 empleados	Sí	Obtención de métricas, Integraciones	Software propio/plataforma interna	5	La plataforma Integrates de Fluid Attacks se creo para eso, para minimizar al hacker la tarea de documentación. Pero dado que nosotros vendemos la plataforma o la ofrecemos OSS, el problema no es solo la presentación de informes, sino brindar un servicio de hacking continuo. Hay multiples herramientas pagas y OSS solo para Informes de Pentest	1	1	1	1	1	1	Más de 500 USD	No	La razon es que nosotros hemos desarrollado nuestra propia herramienta para ello, y la evolucionamos constantemente basado en lo que piden los clientes. Nosotros en realidad el PenTest de App, lo hacemos de forma continua y no es un Pentest de infra que genere un PDF o que genere una entrega especifica puntual. Entonces a nosotros no nos aplica en realidad dicha herramienta. Empresas que se dedican a esto, tipicamente compran AttackForge, Dradis, Faraday, Cobalt, PlexTrac,Vulcan, Kenna, Nucleus,	Mejores alternativas	Como nuestro foco, es basicamente controlar el backlog de nuestro propio producto, es dificil o casi imposible adoptar la herramienta mencionada. Creo que nosotros no somos el ICP (Ideal Customer Profile) para tu herramienta, ustedes tienen que hacer encuestas es a empresas pequenas de pentesting puntual. Aqui alguna lista de empresa que hace MPT, para que entrevisten. Por que las que hacen PTaaS tiene siempre su plataforma. https://help.fluidattacks.com/portal/en/kb/tags/mpt
Pentester independiente (freelancer, bug bounty hunter, etc)	1 (soy independiente)	Sí	Seguridad, Obtención de métricas	Software propio/plataforma interna	4	Métricas estadísticas de evaluación debo integrarlo con power bi u/o tablu	4	3	2	3	3	3	Menos de 100 USD	Si		Limitaciones de seguridad	Demo
Pentester dependiente (empleado)	Más de 100 empleados	Sí	Obtención de métricas, Integraciones	Overleaf	3	El informe toma tiempo, pero agrega formalidad el formato que entrega.	4	3	5	4	2	5	No hay presupuesto/no está definido	No	El proceso de generación de informe ya lo tengo estandarizado y automatizado, sin embargo considero que para lo que entrega la herramienta los 20USD estan OK.	Mejores alternativas	Garantía de servicio
Pentester dependiente (empleado)	2-20 empleados	Sí	Seguridad, Obtención de métricas	Software de reportería (de terceros)	4	La obtención de métricas de forma automática, sin necesidad de registrar manualmente los valores para obtener las métricas.	4	4	3	5	4	5	Menos de 100 USD	Si		Mejores alternativas	Demo
Pentester dependiente (empleado)	Más de 100 empleados	No	Seguridad, Colaborativiad	Software propio/plataforma interna	5	Ninguno	3	5	5	3	3	4	No hay presupuesto/no está definido	No	No lo necesito	Costo	Testimonios
Pentester dependiente (empleado)	21-50 empleados	Sí	Seguridad, Colaborativiad	Software propio/plataforma interna	3	El software que utilizamos rara vez genera el informe con el formato adecuado	4	5	3	4	5	3	No sé	Si		Mejores alternativas	Garantía de servicio
Pentester independiente (freelancer, bug bounty hunter, etc)	1 (soy independiente)	Sí	Seguridad, Facilidad de uso	LibreOffice	2	Actualmente trabajo como pentester independiente y utilizo LibreOffice para generar los informes, pero me resulta un proceso lento y poco profesional. Organizar los hallazgos manualmente es tedioso, el formato final carece de presentación profesional y además tengo dudas sobre la seguridad al compartir los documentos con clientes.	3	5	3	5	2	5	No hay presupuesto/no está definido	No	Aunque valoro mucho las funcionalidades de AuditForge (especialmente la generación automatizada de informes, el sistema de recomendación de CWE/CVSS y que sea open-source), no estaría dispuesto a pagar los 20 USD mensuales en mi situación actual. La razón principal es que, como pentester independiente, mi presupuesto para herramientas es limitado o inexistente ("no hay presupuesto definido"), y priorizo invertir en recursos que tengan un ROI inmediato y tangible (como plataformas de bug bounty o herramientas técnicas). Si bien las 4 horas de soporte son un plus, no justifican el costo para mi flujo de trabajo actual, donde ya gestiono informes de manera básica (aunque ineficiente). Sin embargo, si en el futuro mi volumen de clientes creciera y el ahorro de tiempo fuera significativo, reconsideraría una versión más económica o un modelo de pago único. Por ahora, optaría por usar la versión open-source y contribuir al proyecto (ej: reportando bugs o mejoras) en lugar de pagar la suscripción.	Costo	Prueba gratuita
Pentester dependiente (empleado)	2-20 empleados	Sí	Seguridad, Integraciones	Software propio/plataforma interna	4	A veces me cuesta estructurar bien la información. Tengo claro lo que quiero decir, pero al momento de armar el informe, no sé por dónde empezar o cómo dejarlo claro para quienes lo van a leer. Además, siento que muchas veces los informes se entregan, pero no hay una retroalimentación real, entonces no sé si realmente sirvieron o si entendieron lo que quise comunicar.	5	5	5	5	5	5	101 a 500 USD	Si		Limitaciones de seguridad	Demo
Pentester independiente (freelancer, bug bounty hunter, etc)	1 (soy independiente)	No	Obtención de métricas, Integraciones	Markdown/Pandoc	5	Mi contraparte (cliente) al recibir mi informe no lo hace llegar a TI/desarrolladores.	4	5	5	4	3	4	Menos de 100 USD	No	Para mi negocio (a lo más 1 pentesting cada 3-6 meses) no se justifica.	Costo	Testimonios
Pentester dependiente (empleado)	2-20 empleados	No	Seguridad, Integraciones	Microsoft Word/Google Docs	5	No tengo problemas en la redacción de informes, diseñe una metodología propia para diseñar los informes, principalmente, enfocados en las necesidades del cliente. Los formatos son para las certificaciones, pero en la vida real, los reportes son más personalizados y prefiero desarrollarlos para cada caso.	3	3	3	3	3	3	No sé	No	Porque no la necesito.	Tengo mi propia metodología	No necesito que me convenzan
Pentester dependiente (empleado)	2-20 empleados	No	Seguridad, Facilidad de uso	LibreOffice	4	La dificultad suele recaer en hacerle entender al cliente la gravedad de los hallazgos encontrados. Sin embargo, en términos del informe en sí mismo, no lo consideraría una dificultad pero es bastante tedioso tener que repetir varias cosas al documentar, sería ideal automatizar	5	5	5	5	5	5	101 a 500 USD	Si		Personalmente no tengo inconvenientes con la generación de informes en sí misma, mi objetivo es automatizar explicaciones, es decir, poder entregar contexto de una vulnerabilidad y que el sistema me genere los párrafos con sus explicaciones. Por esa razón, no adoptaría AuditForge, pero me parece una excelente solución	Tendría que tener la necesidad que se logra satisfacer adquiriendo AuditForge
Pentester dependiente (empleado)	Más de 100 empleados	Sí	Seguridad, Facilidad de uso	Microsoft Word/Google Docs	1	El tiempo que conlleva efectuar un informe detallado y claro con sus evidencias respectivas puede ser excesivo y más aun cuando se pueden tener 10, 20 o más vulnerabilidades en uno solo.	4	5	3	3	3	2	Menos de 100 USD	No	Si bien es una solución al problema inicial, igual hay formas o metodos para simplificar la redacción de un informe, además de utilizar plantillas y prompts de llm locales que te simplifican el trabajo a la hora de rellenar secciones. Y en lo particular, porque la empresa en la que trabajo ya esta en proceso de diseño de un metodo mejorado más a la medida...	Mejores alternativas	quizas una Demo y saber la capacidad de integración con más sistemas para sacarle más jugo
Pentester independiente (freelancer, bug bounty hunter, etc)	1 (soy independiente)	No	Facilidad de uso, Colaborativiad	Software de reportería (de terceros)	5	Mas que nada plantillas estandar o personalizadas para cada cliente	5	5	5	5	4	5	No hay presupuesto/no está definido	Si		Costo	Prueba gratuita