2. Plataformas para Requisitos

2.1. Introducción y objetivos

En este tema se estudian las principales plataformas que nos ayudan a definir, documentar y gestionarlos requisitos en un proyecto de software.

Todo proyecto de software tiene un punto de entrada común: la captura de requisitos. Conocer las necesidades del cliente, documentarlas y transformarlas a requisitos de desarrollo, tanto funcionales como no funcionales, y la gestión de estos a lo largo del proyecto son tareas necesarias y fundamentales en cualquier desarrollo.

Las herramientas de requisitos no son nuevas y han ido evolucionando en paralelo a las metodologías, las herramientas, los lenguajes de desarrollo y las técnicas de ingeniería de software.

En este tema posicionaremos las diferentes herramientas de requisitos, detallaremos su relevancia actual y los criterios que podrían ser de utilidad para tomar la decisión de su adopción en una organización.

Con el estudio de este tema pretendemos alcanzar los siguientes objetivos:

* Comprender el valor y la utilidad que tienen las plataformas para la definición y gestión de los requisitos.
* Identificar los criterios principales que nos ayuden a seleccionar las plataformas en cada caso.
* Conocer las principales plataformas de definición de requisitos.
* Conocer las principales plataformas de gestión de requisitos.
* Saber utilizar algunas de las principales plataformas de requisitos

2.2. Contexto de plataformas de requisitos

El software que se produce es cada día más complejo. En las últimas décadas, se han introducido importantes mejoras en la forma en la que se lleva a cabo el desarrollo de software, especialmente en el ámbito de la gestión de proyectos, con la aparición de nuevas metodologías más eficientes, la codificación y las pruebas, mediante la aplicación de técnicas que permite la reutilización, la composición e incluso la automatización. Sin embargo, la principal fuente de defectos en cualquier desarrollo sigue siendo la actividad de captura y gestión de requisitos, puesto que es el punto en el que debe establecerse la comunicación entre negocio y tecnología.

Los requisitos son el preámbulo de cualquier proyecto de desarrollo de software. Ningún proyecto de software debería comenzar sin saber en qué va a consistir, qué es lo que se quiere desarrollar y cómo se podrá decir que se ha alcanzado con éxito su objetivo. Al ser el punto de partida de cualquier desarrollo, son considerados el principal criterio de éxito del equipo de proyecto, el fundamento de la definición de tareas del equipo y la base para el resto de las fases de desarrollo: diseño, codificación y pruebas.

Podemos encontrar diferentes definiciones de requisitos, desde la más básica en la que se establece que los requisitos son una especificación de lo que debería ser implementado (Sommerville y Sawyer, 1997), hasta otras más estandarizadas, como la de IEEE, que ofrece tres posibles definiciones de requisitos (IEEE, 2002):

* Una condición o capacidad que un usuario necesita para resolver un problema o alcanzar un objetivo.
* Una condición o capacidad que un sistema o componente de sistema debe poseer para satisfacer un contrato, estándar, especificación u otros documentos formalmente impuestos.
* Una representación documentada de una condición o una capacidad que cumple alguna de las dos definiciones anteriores.

En muchas ocasiones, requisitos se considera como sinónimo de requerimiento o de necesidad y, aunque tienen connotaciones algo diferentes, se puede considerar válida dicha consideración.

Aunque la industria del software suele utilizar estas definiciones como guía, es necesario destacar que hay aspectos y características propias de los requisitos que no quedan completamente descritas. Así, se tiene en cuenta las necesidades del usuario, ¿pero las de los desarrolladores?, ¿un cliente y un usuario son considerados lo mismo y tienen las mismas necesidades? o ¿tiene en cuenta también las restricciones? Estas son solo un pequeño ejemplo de lo que quedaría al margen de dicha definición. Por ello, Chemuturi (2013) establece:

«Un requisito es una necesidad, expectativa, restricción o interfaz establecida por cualquiera de los actores involucrados (*stakeholders*) en un desarrollo de software que debe ser satisfecha por el producto de software resultante en algún momento de su desarrollo» (Chemuturi, 2013).

De esta forma, los requisitos permiten alinear las diferentes visiones entre las personas involucradas en el proyecto para establecer una común y compartida, lo que define objetivos y expectativas.

Además de ser el inicio de un proyecto, los requisitos y las actividades relacionadas con ellos se extienden a lo largo del ciclo de vida de este, ya que deben ser elicitados, analizados, establecidos, documentados, modificados y gestionados a lo largo de todo el proyecto.

La integración de modelos de madurez de capacidades o CMMI (Chrissis et al., 2011), un marco de referencia fundamental en el ámbito de las buenas prácticas en el proceso de desarrollo del software, hace alusión a los requisitos en dos áreas de proceso diferenciadas: la primera, que pertenece al nivel dos, se denomina gestión de requisitos (*requirements management*, REQM) y su propósito consiste en gestionar los requisitos del producto y componentes del producto que tiene un proyecto, así como identificar las inconsistencias existentes entre dichos requisitos y los diferentes elementos de trabajo y planes dentro del proyecto. La segunda, denominada desarrollo de requisitos (*requirement development*, RD), pertenece al nivel tres y tiene como objetivo establecer y analizar todos los requisitos: de cliente, de producto y de componente de producto. Así se pone de manifiesto, por un lado, la importancia de los requisitos a la hora de llevar a cabo el desarrollo de un proyecto y, por otro, que los requisitos son elementos vivos a lo largo de este, por lo que hay establecer los mecanismos adecuados para su gestión de forma sistemática y disciplinada. En ocasiones, el conjunto de todas estas tareas es denominada como ingeniería de requisitos.

Las organizaciones de desarrollo tienen claro que los requisitos importan. Sin embargo, siguen siendo uno de los grandes retos para cualquier proyecto de software y uno de los procesos de negocio más desafiantes para llevarlos a cabo correctamente a pesar de los años en los que lleva siendo objeto de estudio (Lawrence et al., 2001). Unos requisitos mal definidos, inexistentes o cambiantes pueden hacer de un proyecto un problema, con un alto coste en dinero y tiempo. El éxito de un proyecto depende de una buena estructura para la ingeniería de requisitos puesto que la buena comunicación y entendimiento entre clientes, proveedores, actores implicados, organización interna y equipo de desarrollo hace que los requisitos sean algo más que un conjunto de documentos que recogen funcionalidades o diagramas del lenguaje unificado de modelado (UML).

2.3. Valor y utilidad de las plataformas de requisitos

Una vez establecida la necesidad de documentar y mantener los requisitos de los proyectos, las diferentes organizaciones comienzan a plantearse la necesidad de tener herramientas que les permitan hacer dichas actividades de forma más eficaz. En un momento inicial, la primera decisión pasa por tener algo de fácil acceso y posible utilización por cualquier persona. Por ello, las herramientas más utilizadas en la fase de requisitos suelen ser los editores de texto, como MS Word, o las hojas de cálculo, como MS Excel. En el ámbito de la ingeniería de requisitos, esto se conoce como actividades manuales que, a pesar de tener buena voluntad, acaban siendo rechazadas puesto que no son efectivas ni eficientes, por lo que no suelen ser suficientes.

Diferentes herramientas y plataformas de gestión de requisitos han ido surgiendo a lo largo de los últimos años, más o menos complejas, que permiten llevar a cabo acciones como la administración de las relaciones entre los requisitos, la gestión de los cambios o análisis del impacto a un cambio de requisitos. Además, gran parte de las herramientas de desarrollo de software ofrecen funcionalidades para la gestión de requisitos. Esto, que puede ser considerado como una gran ventaja por los equipos de desarrollo, ya que se obtiene por defecto la interoperabilidad con el resto de las fases de desarrollo, no siempre dispone de toda la potencia que una buena definición y gestión de requisitos necesita.

A la hora de seleccionar la plataforma de requisitos que más se ajusta a una organización, podemos hacer uso de la ISO/IEC TR 24766:2009, un marco que sirve como guía para la evaluación de las capacidades de las herramientas de ingeniería de requisitos. Para ello divide las capacidades de una herramienta en seis categorías principales (International Organization for Standarization, 2009):

* Elicitación de requisitos, es decir, si la plataforma de requisitos es capaz de soportar la identificación de los *stakeholders* (actores asociados), la captura y traza con los requerimientos de negocio y usuario, los requerimientos funcionales y los requerimientos no funcionales durante las tareas de descubrimiento de requisitos.
* Análisis de requisitos, es decir, si la plataforma de requisitos es capaz de soportar la descomposición de requerimientos en requisitos funcionales y no funcionales y de analizar la viabilidad y el riesgo de los requisitos.
* Especificación de requisitos, es decir, si la plataforma es capaz de soportar la especificación de los requisitos de una forma consistente, accesible y revisable.
* Verificación y validación de requisitos, es decir, si la herramienta es capaz de soportar diferentes formas evaluar y testear la validación y verificación de los requisitos.
* Gestión de requisitos, es decir, si la herramienta es capaz de soportar la monitorización de los cambios y el mantenimiento de los requisitos para asegurar que están reflejados en el producto resultante.
* Otras capacidades, entre las que podemos incluir la trazabilidad de los requisitos con los resultados de otras partes del proyecto, información más administrativa de la plataforma, integración con otras herramientas o el uso de interfaces de usuarios más gráficos.

Durante los últimos años, se han realizado diferentes estudios y revisiones que intentan recoger los requisitos que deben cumplir las herramientas de gestión de requisitos. Entre los más importantes destacamos los trabajos del INCOSE Tool Survey y el informe de Business Industry Reports.

En esta misma línea, Carrillo de Gea et al. (2021), en su trabajo, han actualizado una lista de criterios para evaluar las herramientas de ingeniería de requisitos, al incorporar nuevos criterios aparecidos debido a la evolución de las diferentes metodologías y tecnologías junto aquellos a los que hacía referencia la ISO/IEC TR 24766:2009 y que siguen siendo aspectos clave de cualquier herramienta (International Organization for Standarization, 2009).

La lista, en la que se destacan los diferentes criterios, su descripción y su identificación, es la siguiente, en la que se detalla la forma de evaluar como (+), (++) o (+++) el grado de cobertura de cada herramienta respecto a cada criterio:

* Organización de los requerimientos con metadatos, atributos y reutilización (OR): la herramienta permite al usuario agregar metadatos a los requerimientos en diferentes formatos (+), incluye atributos de requisitos predefinidos y atributos definidos por el usuario (++) y proporciona requisitos definidos por el usuario y plantillas de documentos de requisitos (+++).
* Informes, consultas a bases de datos y lenguaje de interfaz abierta (RDOIL):la herramienta es capaz de generar informes estructurados, sofisticados y/o estandarizados a partir de los datos (+), su funcionalidad se puede ampliar fácilmente (++) y ofrece un lenguaje de extensión, API y/o API REST (+++).
* Controles internos, es decir, consistencia, dependencias e historia (IC):la herramienta ofrece mecanismos para la detección y resolución de conflictos entre requisitos y/o verificaciones de consistencia (+), admite análisis de dependencia (++) e incluye funcionalidad de control de versión de requisitos y gestión de cambios de requisitos (+++).
* Apoyo a la trazabilidad, es decir, arrastrar y soltar (horizontal y verticales) (TS)*:*la herramienta ofrece soporte a trazabilidad visual y/o diferentes tipos de trazas (+), soportes a la trazabilidad de los requisitos con otros artefactos (++) y ofrece una trazabilidad completa para gestionar los requisitos a lo largo del ciclo de vida del desarrollo, incluida la trazabilidad externa a otras herramientas (+++).
* Gestión de variantes (VM):la herramienta incluye soporte explícito para reutilización (+), creación y gestión de variantes (++) y líneas de productos de software y/o catálogos de requisitos reutilizables (+++).
* Trabajo remoto, solo en la nube (RWC):la herramienta puede funcionar sin conectividad (+), su funcionalidad cuando no está conectada es completa (++) y permite al usuario trabajar sin conexión y sincronizar cuando la conectividad está disponible (+++).
* Múltiples vistas (MV):la herramienta incluye bases de datos y vistas orientadas a documentos de los requisitos (es decir, Word especificación, tabla, etc.) (+), ofrece diferentes vistas de las relaciones de trazabilidad (textual, árbol, matriz) (++), y permite al usuario visualizar las dependencias de requisitos y su evolución (+++).
* Rendimiento (Perf):la herramienta puede gestionar grandes proyectos con grandes cantidades de datos (+), está optimizada para cargar grandes modelos (++) y ofrece un rendimiento adecuado y es receptivo, incluso si hay trabajo remoto sin conectividad (+++).
* Colaboración, gestión del *workflow* (CWM):la herramienta brinda soporte para el trabajo colaborativo y/o distribuido, ofrece funcionalidad de revisión, discusión y/o voto —así como comunicación asíncrona y/o síncrona—, incluye control de acceso y seguridad granular (+), proporciona flujos de trabajo (*workflows*) predefinidos, flexibles y personalizables (++) y admite múltiples dispositivos (por ejemplo, navegador web, aplicación móvil) (+++).
* Fácilmente adaptable e integrable en los procesos de negocio (IBP):la herramienta es personalizable y flexible para adaptarse a las condiciones del proyecto, los *stakeholders* o la naturaleza. de los requisitos (+), puede adaptarse fácilmente a cualquier tipo de proceso de negocio o industria (++) y puede integrarse o adaptarse fácilmente a herramientas y procesos preexistentes dentro del organización (+++).
* Federación y notificación con herramientas *application and product lifecycle management* (ALM/PLM) (FN):la herramienta ofrece funcionalidad ALM/PLM (+), ofrece la posibilidad de integrarse son soluciones ALM/PLM de terceros (++) y tiene soporte Open Services for Lifecycle Collaboration (+++).
* Exportación/Importación con formatos estándar (EI):la herramienta ofrece una funcionalidad completa de importación/exportación y soporte para formatos estándar: CSV (+), otros (excepto el *requirements interchange format*[ReqIF] estándar) (++) y ReqIF estándar (+++).
* Macros para repetir comandos (MaC):la herramienta incluye la funcionalidad de creación y gestión de macros (+), proporciona un lenguaje de scripting (++) y/u otros mecanismos de automatización (+++).
* Esfuerzo de formación y curva de aprendizaje (TLC):la herramienta incluye documentación adecuada, tutoriales, materiales de capacitación y recursos (+) y cursos y/o certificaciones (++) y es fácil de usar y/o aprender (+++).
* *Agile*, integración y distribución continuas (CI/CD) y desarrollo y operaciones (DevOps):la herramienta está concebida o es fácilmente adaptable a metodologías ágiles y *pipelines* DevOps (+), permite la gestión de requisitos en marcos ágiles, *lean* o *scaled agile* (++) e integración continua (CI)/ despliegue continuo (CD) y DevOps (+++).
* Soporte inteligente (IS):la herramienta el procesado, agrupado o clasificación de requisitos mediante el uso de procesamiento de lenguaje u otros métodos automáticos basados en inteligencia artificial y analíticas de datos (+), ofrece funcionalidad de generación de código (++) y proporciona recomendaciones sobre los activos del proyecto (por ejemplo, *stakeholders* y requisitos) para facilitar el seguimiento y la toma de decisiones (+++).
* Escalabilidad (Scal):la herramienta ofrece soporte básico a grandes proyectos (+), puede escalar adecuadamente con el tamaño del proyecto y del equipo (++) y se ofrece dentro de un modelo de servicio en la nube (SaaS) (+++).

Siguiendo estos criterios, Carillo de Gea et al. (2021) han llevado a cabo una evaluación de trece herramientas seleccionadas entre más de las 200 disponibles en el mercado. En la Figura 1 se observa el resumen del estado del uso de las trece herramientas de ingeniería de requisitos y en la Tabla 1, la evaluación de dichas herramientas en cuanto a los criterios definidos.

Figura 1. Herramientas de ingeniería de requisitos y estado de su práctica. Fuente: Carrillo de Gea et al., 2021.

A table with numbers and symbols

Description automatically generated

Tabla 1. Evaluación de herramientas de Ingeniería de Requisitos. Fuente: Carrillo de Gea et al., 2021.

2.4. Principales plataformas de requisitos

Algunas de las herramientas comerciales mejor valoradas por los diferentes estudios sobre soluciones de gestión e ingeniería de requisitos son las siguientes:

Tabla 2. Principales plataformas de requisitos. Fuente: elaboración propia.

IBM DOORS Family

IBM ofrece dos soluciones de gestión de requisitos con muchas capacidades similares: IBM Requirements Management DOORS e IBM Requirements Management DOORS Next. Sin embargo, los productos ofrecen diferentes implementaciones y oportunidades estratégicas para los analistas de requisitos y los equipos de diseño, desarrollo y prueba. Ambos productos continúan siendo mejorados y respaldados por sus respectivas bases de clientes.

DOORS continúa siendo, después de años, la solución de gestión de requisitos líder del mercado y ya se encuentra en la versión 9. Proporciona una amplia gama de capacidades de gestión de requisitos muy maduras, como módulos de especificación de requisitos estructurados, importación y exportación de datos de ida y vuelta, firmas electrónicas, líneas de base y vistas de requisitos personalizables con trazabilidad de varios niveles y la gestión de cambios de los cambios a los requisitos. Da soporte a grandes proyectos de desarrollo con equipos en programas de ingeniería de sistemas complejos y de alto cumplimiento, que dispongan de miles de objetos o cientos de usuarios simultáneos.

DOORS ofrece una API (interfaz de programación de aplicaciones) extensible programable por el usuario llamada DOORS eXtension Language (DXL). DXL proporciona una forma de personalizar estas actividades:

* Automatización de la administración
* Creación de vistas de trazabilidad multinivel
* Cálculo de métricas
* Implementación de la lógica de negocios
* Ampliación de la interfaz de usuario
* Integración con otras herramientas

Figura 2. Vista de Project Window en DOORS. Fuente: Software informer, 2022.

DOORS Next se ha desarrollado como una plataforma colaborativa de gestión de requisitos que dispone de un cliente web. Este producto incluye capacidades tales como definición de requisitos visuales, elementos de trabajo e integración de pruebas y planificación. Utiliza las capacidades de Jazz Team Server para respaldar la colaboración en equipo a través de paneles, revisiones y comentarios. Los usuarios, proyectos, tipos de datos, tipos de artefactos, atributos y etiquetas tienen una administración común. Estos servicios comunes y sistemas de tipos ayudan a los equipos de proyecto a definir y gestionar los requisitos de forma coherente. Su principal característica es su interoperabilidad con otras herramientas, incluida la gestión del ciclo de vida, la colaboración en equipo y la ingeniería de sistemas/software.

Figura 3. Vista de la lista de artefactos en DOORS Next. Fuente: ReqPro, s.f.

DOORS y DOORS Next pueden utilizarse conjuntamente. Por ejemplo, puede vincular requisitos y artefactos relacionados entre los dos productos. Los módulos y objetos que se exportan desde DOORS se pueden importar a DOORS Next.

El cliente web de DOORS Next proporciona herramientas para definir requisitos en documentos de texto enriquecido y representaciones visuales en diagramas de procesos comerciales, diagramas de casos de uso, guiones gráficos, bocetos de interfaz de usuario y flujos de pantalla. Los artefactos de requisitos se pueden organizar y reutilizar en vistas, colecciones y módulos. El cliente web admite enlaces de trazabilidad a artefactos de requisitos relacionados y en aplicaciones Elm a planes de desarrollo, elementos de trabajo, planes de prueba, casos de prueba, diseños y modelos.

Jama Software

Jama Connect es una plataforma de desarrollo de productos que ayuda a los equipos de hardware y software a conectar actividades aisladas de desarrollo, pruebas y gestión de riesgos para proporcionar cumplimiento integral, mitigación de riesgos y mejora de procesos. La plataforma permite a los usuarios definir, alinear y ejecutar proyectos, lo que reduce los tiempos de ciclos prolongados, el esfuerzo dedicado a demostrar el cumplimiento y el desperdicio de trabajo.

Jama Connect admite múltiples metodologías de desarrollo concurrentes e incluye trazabilidad que permite la gestión de procesos complejos, una interfaz intuitiva y fácil de usar. Esto, a su vez, alinea a las partes interesadas en el desarrollo de hardware, firmware y software. Tiene características como la trazabilidad de datos y recursos de extremo a extremo, análisis de impacto en tiempo real, revisiones y aprobaciones que pueden ayudar a impulsar la colaboración en equipo. Además, puede integrar el software con aplicaciones de terceros como Jira, Rally, GitHub y otras herramientas de gestión de requisitos o ALM, como Micro Focus y Perforce para asegurar la trazabilidad, visibilidad y colaboración a lo largo de todo el ciclo de vida.

La herramienta pone a los equipos en funcionamiento rápidamente con marcos específicos de la industria que se alinean con los estándares, lo que genera confianza en el cumplimiento de las normativas, y permite que los equipos muestren fácilmente a los auditores cómo se tomaron las decisiones con informes.

Figura 4. Vista de trazabilidad de requisitos en Jama Software. Fuente: Jama Software, s. f.

Modern Requirements4DevOps

Es la solución de gestión de requisitos mejor valorada por los principales analistas. Está integrada en Azure DevOps, lo que lo convierte en una potente solución de gestión de requisitos. Facilita la integración de los equipos bajo una plataforma para facilitar un modelo de fuente única de la verdad. Sus herramientas están diseñadas para acelerar la velocidad de los proyectos, brindar valor comercial, satisfacer las necesidades de calidad y cumplimiento para todas las metodologías e industrias (Requirements Management, 2019).

Da soporte a documentación textual, casos de uso, revisión en línea, integración con firma electrónica, trazabilidad extremo a extremo, creación de línea de base y control de versiones, conexión de requisitos con diagramas y *mockups*, así como la generación automática de informes y auditorías personalizados.

Algunas de sus características o funcionales principales son:

* Smart Docs: una herramienta de creación similar a Word que está disponible en línea. Permite crear elementos de trabajo de Azure DevOps en una vista de documento con texto enriquecido, tablas e imágenes en línea, utilizando un conjunto completo de controles de texto enriquecido.
* Baseline: proporciona una instantánea de un punto en el tiempo de cualquiera o todos los elementos de trabajo de Azure DevOps dentro de un proyecto. Esta herramienta brinda a los usuarios la capacidad de ver y comparar líneas base y comparar entre reversiones más antiguas y recientes de elementos de trabajo. Tener una línea de base en cada proyecto permite a los usuarios monitorear el desempeño actual del proyecto, administrar el alcance de la versión, crear notas de la versión, retroceder a una versión anterior y mejorar la precisión de las estimaciones futuras. Un uso secundario de la función de copia de elementos de trabajo dentro de la herramienta es para la reutilización de elementos de trabajo. La herramienta permite a los usuarios crear una biblioteca de paquetes reutilizables de requisitos —como requisitos no funcionales (NFR)— y luego aplicarlos según sea necesario a nuevos proyectos, junto con capacidades de informes de reutilización para realizar un seguimiento de las eficiencias.
* Análisis de trazas: la herramienta proporciona enlaces completos de extremo a extremo entre hasta diez niveles de diferentes tipos de elementos de trabajo en una matriz horizontal, incluidos los resultados de las pruebas y los defectos. También le permite personalizar las propiedades que se muestran. La matriz de intersección permite a los usuarios agregar o eliminar relaciones entre elementos de trabajo de manera eficiente, lo que ayuda a los usuarios a planificar, rastrear y analizar la calidad, la cobertura y la integridad del proyecto.
* Gestión de la revisión: permite administrar la función de revisión y aprobación de requisitos o cualquier tipo de elementos de trabajo y solicitar firmas electrónicas. Los usuarios pueden programar una revisión y solicitar una respuesta de uno o más revisores sobre múltiples requisitos. Las partes interesadas pueden ver la lista de requisitos en un paquete de revisión y también ver todas las respuestas proporcionadas en relación con los requisitos individuales. Los revisores pueden responder en «Aceptar/Rechazar» y «Aceptar todo/Rechazar todo».
* Alice: es la primera de una serie de herramientas inspiradas en inteligencia artificial (IA) en la familia de requisitos modernos. Es un asistente de analista de negocios cognitivo que le permite obtener requisitos automáticamente a través de una jerarquía de preguntas estructurada lógicamente.
* Informes inteligentes: permite crear automáticamente informes HTML detallados directamente desde la vista de trabajo pendiente o consulta. Filtra, agrupa, ordena y exporta la información como archivo Word o PDF. Utilice informes para compartir información con partes interesadas externas y generar documentos de requisitos del sistema o del negocio. Con informes avanzados, los usuarios pueden crear y almacenar plantillas de Word definibles por el usuario.
* Diagramas: proporciona una forma visual para que los usuarios creen y gestionen el flujo de eventos o el funcionamiento interno de la aplicación, de forma gráfica. Utilizando modelos UML y otros diagramas profesionales, los usuarios pueden visualizar los requisitos del proyecto en un diseño de modelado. También permite a los usuarios conectar un diagrama con elementos de trabajo de Azure DevOps, así como también admite la discusión sobre elementos de trabajo en medio de un grupo de usuarios en el proyecto del equipo conectado. La automatización de análisis adicional en este módulo proporciona la creación de historias de usuario y el escenario de prueba y la generación de pasos de prueba.
* Editor de casos de uso: los usuarios pueden crear diagramas de actividad automáticamente simplemente ingresando un caso de uso textual, lo que proporciona una verificación visual de la narrativa creada. Los usuarios pueden definir los casos de uso usando varias herramientas dentro del editor de casos de uso para crear una estructura consistente, escenarios correctos y enlaces dentro y hacia casos de uso externos; el diagrama de actividad se actualizará en función de los cambios realizados en la narrativa del caso de uso. Los usuarios pueden codificar con colores los diferentes escenarios dentro de un caso de uso para visualizar diferentes aspectos a fin de comprender y validar los escenarios. Los usuarios pueden ver todos los casos de uso junto con sus representaciones visuales (diagramas) en un proyecto de equipo.
* Analizador de impacto: permite resaltar cualquier elemento de trabajo y ver todos sus elementos de trabajo vinculados visualmente en una vista de mapa mental. La vista se puede ampliar para incluir elementos de trabajo dentro de un equipo o para incluir también dependencias entre equipos. Si los cambios son inminentes, los usuarios pueden capturar el impacto de los cambios, estimar el trabajo involucrado y crear un presupuesto por correo electrónico para enviar a las partes interesadas correspondientes. Los usuarios también pueden crear tareas de elementos de trabajo mientras están en esta vista para abordar los cambios.
* Simulación y *wireframes*: ayuda a los usuarios a organizar sus pensamientos visualmente, creando un flujo de pantallas o *wireframe*. Luego, cuando la información debe crearse en Azure DevOps como requisitos, historias de usuario y tareas, la aplicación le permite crear todos los elementos de trabajo directamente desde la simulación al proyecto TFS (*team fundation server*). En cualquier momento, los usuarios pueden enviar un archivo de maqueta a las partes interesadas del proyecto, quienes a su vez pueden ver la maqueta, anotar las pantallas, proporcionar comentarios de texto o de voz al autor. La aplicación admite la obtención, validación y claridad de requisitos en un entorno visual.

Figura 5. Vista de trazabilidad de requisitos en Modern Requirements. Fuente: Visual Studio, 2019.

Visure Requirements

Es una solución de gestión de requisitos que ayuda a las empresas a optimizar los procesos relacionados con las pruebas de aceptación, la gestión de la configuración, la colaboración de tareas y más en una plataforma centralizada. Los administradores pueden configurar permisos basados en roles, comparar versiones y configurar flujos de trabajo de procesos. Permite a los analistas de datos crear relaciones, generar jerarquías, administrar trazabilidades y capturar automáticamente los requisitos de MS Excel, Outlook y MS Word. La herramienta de gestión de proyectos permite a los supervisores crear representaciones gráficas de dominios, navegar a través de especificaciones y crear hipervínculos de proyectos. Además, los gerentes pueden generar informes definidos por el usuario, corporativos, interactivos y predefinidos en múltiples formatos de archivo, como HTML, PDF, XML y CSV. Facilita la integración con varias soluciones de terceros, como Accompa, Jira, MS Sharepoint y Salesforce.

Algunas de sus características o funcionales principales son:

* Ingeniería de requisitos impulsada por procesos: la herramienta orquesta el proceso de desarrollo, gestionando toda la información relacionada con los requisitos (como requisitos, pruebas, solicitudes de cambio, riesgos, etc.), sus relaciones e interacciones con los usuarios. Ayuda a estandarizar y hacer cumplir los procesos definidos en toda la organización, formalizar una estructura de especificación de requisitos comunes y manejar los cambios a lo largo del ciclo de vida.
* Cobertura del ciclo de vida de los requisitos: brinda un soporte integral al proceso completo de requisitos, incluida la captura, el análisis, la especificación, la validación y verificación, la trazabilidad, la gestión y la reutilización de requisitos.
* Orientación al usuario: es una herramienta totalmente personalizable que permite a los administradores proporcionar a los usuarios vistas para realizar cada una de sus tareas específicas como: trazabilidad, análisis de impacto, creación de requisitos, validación, especificación, etc. El usuario accederá solo a la información que necesita para trabajar y con las opciones que necesitan.
* Colaboración: está destinada a ser una herramienta multiusuario que permite a varios usuarios trabajar con el mismo conjunto de requisitos al mismo tiempo, manteniendo un seguimiento e informes sobre cada cambio con un sistema completo de control de versiones.
* Integraciones: proporciona una plataforma de integración versátil que se puede utilizar para integrarse con herramientas de terceros, comerciales o propietarias, para extender las funciones de análisis de impacto de cambio a elementos fuera del alcance de la herramienta.
* Reutilización: permite compartir requisitos y otros artefactos como pruebas o casos de uso. Los usuarios pueden crear fácilmente familias de productos —que comparten un núcleo de requisitos comunes—, un conjunto de requisitos de estándares o incluso funciones completas junto con sus pruebas y casos de uso. Además, la reutilización va mucho más allá de copiar y pegar, ya que es posible mantener una referencia al proyecto original y actualizarse a través de los cambios realizados en los elementos originales (si es necesario).

Figura 6. Vista del *dashboard* de Visure Requirements. Fuente: Capterra, s. f.

ReqSuite RM

Es un software de gestión de requisitos totalmente personalizable, intuitivo y alimentado por IA para lograr la máxima eficiencia y facilitación en los proyectos. Con ReqSuite RM, se minimiza el riesgo de demoras, errores, reelaboración costosa y esfuerzo manual adicional como resultado de la falta de soporte de herramientas o de un soporte inadecuado.

Figura 7. Vista del entorno para especificar requisitos con ReqSuite RM. Fuente: Osseno, 2022.

Algunas de sus características o funcionales principales son:

* Transparencia y visión general: permite tener una visión general completa, transparencia y trazabilidad con respecto a sus requisitos. Por lo tanto, siempre conoce el estado actual y mantiene el control sobre su proyecto.
* Eficiencia y simplicidad: automatiza tareas que requieren mucho tiempo, aumenta la reutilización y simplifica el intercambio con otras herramientas. Esto hace que todo el trabajo de su proyecto sea más simple, más eficiente y enfocado.
* Calidad y conformidad: mejora la calidad de sus requisitos y la conformidad estándar de sus proyectos. Como resultado, logra menos riesgo, reelaboración y necesidad de aclaraciones en sus proyectos.
* Estructura y uniformidad: crea estructuras y procesos claros para gestionar sus requisitos de forma centralizada y uniforme. Esto mejora la cooperación y reduce al mínimo las pérdidas de información.

2.5. Plataformas para historias de usuario y prototipos de interfaz gráfico

El mapeo de historias de usuario (*user story mapping,*USM) es una técnica utilizada en el descubrimiento de productos para esbozar un nuevo producto o una nueva característica para un producto existente (Patton, 2014). Esta técnica nos permite tener una visión compartida del desarrollo entre los diversos *stakeholders*, a la vez que tenemos un mapa global de la aplicación ordenado por versiones. Permite definir conjunto de historias que conformarán el mínimo producto viable (MPV) a sacar al mercado, que permita a los usuarios llevar a cabo todo el flujo de una manera básica, así como una primera priorización de nuevas funcionalidades ordenadas por versiones y alineada a la funcionalidad que percibe el usuario.

Las principales plataformas para historias de usuario son:

Tabla 3. Principales plataformas que soportan historias de usuario. Fuente: elaboración propia.

Easy Agile

Es un software de historias de usuario que permite a los desarrolladores de productos crear mapas de historias simples, pero colaborativos en Jira. El software permite a los usuarios mapear los viajes de los usuarios junto con tableros ágiles en Jira. Los desarrolladores de productos pueden agregar historias nuevas o existentes al *story map* y priorizarlas según el valor de sus usuarios. Es ideal para dividir su historia de usuario en secciones más pequeñas y manejables porque tiene una función de planificación de sprint. Le da al propietario del producto o al maestro de scrum una idea más clara de la línea de tiempo del producto para evitar demoras. También tiene una función de planificación de versiones que brinda a los desarrolladores de productos la libertad de experimentar con nuevas funciones sin preocuparse por dañar la versión o el modelo activo (Easy Agile, 2022).

Agile User Story Map para Jira

Es una herramienta de mapeo de historias de usuarios que admite el refinamiento de la cartera de productos, lo que permite evaluar rápidamente las tareas que deben completarse para que pueda realizar un seguimiento de los miembros del equipo. También permite a los usuarios ver las dependencias para priorizar tareas y promover la responsabilidad. El software utiliza una línea de tiempo similar a un diagrama de Gantt, lo que le brinda una vista panorámica de su proyecto, para que sepa si necesita más recursos humanos o si ajusta su presupuesto para cumplir con los plazos (Atlassian Marketplace, 2022).

Figura 8. Vista del Agile User Story Map para Jira. Fuente: Atlassian Marketplace, s. f.

Avion

Es una herramienta de mapeo de historias de usuario para el desarrollo ágil de software que le permite visualizar su producto mediante la creación de recorridos de usuario detallados. Los usuarios adjuntan archivos y documentos a las historias de usuario, lo que asegura que otros miembros del equipo tengan acceso a documentos críticos. Brinda al equipo visibilidad no solo de la hoja de ruta de la historia del usuario, sino también de todos los cambios en los criterios, lo que garantiza que todos estén en la misma página durante todo el proceso de desarrollo del producto. Si maneja con frecuencia proyectos de desarrollo de productos confidenciales, esta plataforma incorpora un control de acceso basado en roles. Esta función permite a los administradores definir el nivel de acceso que tienen los miembros del equipo según sus funciones. Con la función de control de acceso, puede traer talento externo o autónomos para proyectos específicos sin preocuparse de que accedan a información confidencial que no es esencial para sus funciones. Se integra con otras herramientas de desarrollo de productos y software como Jira Cloud, Jira Server, Trello, GitHub, Zenhub, etc. (Avion, s. f.).

Figura 9. Vista de flujos en USM con Avion. Fuente: Avion, s. f.

Cardboard

Es una pizarra en línea que su equipo de desarrollo puede usar para crear historias de usuarios durante el desarrollo del producto. Permite invitar a miembros remotos del equipo a colaborar en tiempo real, lo que garantiza que escribir historias de usuarios se convierta en un esfuerzo de equipo. Tiene una funcionalidad de arrastrar y soltar, lo que lo hace fácil de usar y reduce la curva de aprendizaje para los nuevos usuarios. Promueve la responsabilidad porque tiene una función de carriles y divisores que pueden ayudar a los usuarios a distinguir visualmente quién es responsable de cada paso del proyecto, dándoles una mejor idea de con quién colaborar para los pasos o tareas interconectados. Esta característica también puede ayudar a promover la responsabilidad porque los usuarios pueden visualizar cómo los retrasos en ciertos pasos pueden afectar el progreso general de un proyecto. Se integra con otras herramientas de desarrollo de software como Trello, Azure DevOps, VersionOne y Confluence (CardBoard, s. f.).

Figura 10. Vista de USM con CardBoard. Fuente: CardBoard, s. f.

Craft.io

Es un software de administración de productos en donde los mismos responsables (*product owners*) pueden usarlo para administrar todas las etapas del desarrollo del producto. Esta ágil plataforma de gestión de proyectos permite a los usuarios descubrir nuevas ideas, colaborar y crear prototipos de productos. Se sincroniza automáticamente con Google Drive y Jira para que los desarrolladores puedan concentrarse en sus tareas asignadas en lugar de actualizar documentos constantemente. Incorpora una función de planificación de capacidad que permite evaluar la disponibilidad de recursos y la asignación de miembros del equipo, lo que le brinda una mejor idea sobre cómo redistribuir los recursos para nuevos proyectos.

Puede usar el software para simular cómo el ajuste de los recursos afectará la entrega del proyecto para que pueda adaptarse rápidamente a los cambios inesperados. Se integra con otras herramientas de desarrollo como GitLab, Google Workspace, Dropbox, Jira y Microsoft Active Directory (Craft.io, s. f.)*.*

Figura 11. Vista del editor de especificaciones de Craft.io. Fuente: Craft.io, s. f.

Featmap

Es un software de historia de usuario de código abierto para ayudar a imaginar qué acciones podría tomar el usuario ideal al interactuar con su producto. Permite asignar personajes de usuario a los miembros del equipo para capturar los comportamientos y las necesidades de sus usuarios finales, lo que le permite crear productos centrados en el usuario. También permite compartir la visión de su producto con los miembros del equipo y las partes interesadas con cuentas de invitados ilimitadas (solo lectura). Como software de código abierto, puede agregar o eliminar funciones según sus necesidades para tener un software de historia de usuario con todas las funciones críticas y ninguna de las adicionales que no necesita. Puede alojar el software en su propio servidor para ahorrar dinero y aumentar el nivel de seguridad de sus productos (Featmap, s. f.).

Figura 12. Vista de USM en Featmap. Fuente: Featmap, s. f.

FeatureMap

Es un software de mapeo de historias de usuario para estructurar el contenido y establecer prioridades.Puede usar el software para planificar varios pasos por adelantado, lo que le permite anticipar problemas y planificar contingencias. También permite dividir tareas grandes en tareas más pequeñas para que sean más fáciles de completar y evitar abrumar a los miembros del equipo. Es la mejor opción para los usuarios que prefieren una herramienta local que funcione bien incluso en áreas donde la conectividad a Internet es inestable. Ser una herramienta local también significa que la falta de conexión a internet no retrasará el desarrollo del producto porque los usuarios aún pueden acceder a archivos y documentos cruciales (FeatureMap, s. f.).

Figura 13. Vista de USM en FeatureMap. Fuente: Feature Map, s. f.

En este vídeo, *Gestionando mapas de historias de usuario con FeatureMap*, aprenderemos a utilizar FeatureMap para crear mapas de historias de usuario y poder priorizar funcionalidades dentro del *roadmap*de un producto.

Miro

Es una pizarra digital que podemos usar en varios proyectos, como lluvia de ideas, seminarios web, reuniones remotas y flujos de trabajo ágiles. Tiene funciones de videoconferencia y uso compartido de pantalla para que las reuniones en línea sean más atractivas e informativas. También tiene muchas plantillas prediseñadas que puede usar en casi cualquier tipo de proyecto, lo que ayuda a lanzar proyectos rápidamente. Es una de las principales plataformas de historias de usuario para trabajar en múltiples proyectos porque permite una cantidad ilimitada de miembros del equipo. Se integra con otras herramientas de productividad y gestión de productos como Jira Cards, Google Drive, Slack, Google Calendar y Microsoft Active Directory(Miro, 2022).

Figura 14. Vista de la plantilla de historias de usuario de Miro. Fuente: Miro, s. f.

Productboard

Es una plataforma de gestión de productos para ayudar a los *product owners*a comprender las necesidades de sus clientes para su consideración durante el desarrollo del producto. Tiene hojas de ruta personalizables con opciones de filtrado que permiten a los usuarios identificar fácilmente los objetivos y logros del proyecto. También permite a los usuarios adaptar hojas de ruta a diferentes audiencias, como usuarios finales y partes interesadas, lo que las hace más fáciles de entender. Permite a los usuarios finales asignar puntajes a las características del producto, para que los desarrolladores sepan en qué características concentrarse y cuáles necesitan desarrollar más.

También puede usarse para identificar características clave que los usuarios finales consideran críticas para garantizar que las incluya en productos futuros. Se integra con otras herramientas de colaboración y desarrollo de productos como Slack, Microsoft Teams, Trello, Salesforce y Zendesk (Productboard, s. f.).

StoriesOnBoard

Otra herramienta para identificar y registrar las necesidades de sus visitantes mediante historias de usuario con multitud de plantillas personalizables. Permite asignar roles, comportamientos y necesidades a los miembros del equipo, lo que le brinda una mejor idea de cómo los futuros usuarios usarán y se beneficiarán del producto. Permite la integración de herramientas como Google Chrome, Zapier, Trello, GitHub, G Suite y otras (StoriesOnBoard, s. f.).

Trello

Llena de listas y tarjetas, esta herramienta parece simple, pero tiene todas las funciones que necesita para el mapeo de historias. Puede publicar comentarios para brindar retroalimentación instantánea al equipo, agregar fechas de vencimiento, listas de verificación, etiquetas y cargar archivos usando Google Drive, OneDrive, Box y Dropbox. Tiene un sistema de notificación sólido, que avisa a través de notificaciones de escritorio, notificaciones automáticas móviles y correo electrónico. Trello también tiene aplicaciones para iPhone, iPad, teléfonos Android, tabletas, relojes, tabletas Kindle Fire y Windows, lo que garantiza que su equipo se mantenga actualizado incluso mientras está en movimiento (Trello, 2022).

Figura 15. Vista de tareas e historias de usuario en Trello. Fuente: Trello, s. f.

En este vídeo, *Gestionando historias de usuario y tareas con Trello*, aprenderemos a utilizar Trello para gestionar tareas o requisitos de un proyecto de forma visual y colaborativa.

*Wireframes*

Por otro lado, existen plataformas que ayudan a generar maquetas o prototipos de interfaz de usuario. Aunque muchas veces se incluyen dentro de la parte de diseño de la experiencia del usuario, en realidad son también una gran fuente de información para poder obtener requisitos en los que nadie piensa de otra manera y que salen a la luz cuando se muestra un ejemplo rudimentario de la interacción.

Un *wireframe* es un modelo o marco esquelético que describe el diseño básico y las funciones de una interfaz de usuario (como un sitio web o una aplicación). El objetivo de un *wireframe* es comunicar rápida y fácilmente el contenido de la página, su estructura y diseño y las funciones de la aplicación. En ocasiones se confunde con las maquetas (*mockups*), que son estructuras estáticas que incluyen detalles más estilísticos y visuales de la interfaz de usuario para presentar un modelo realista de cómo se verá la página o aplicación final.

Las principales plataformas de apoyo a la creación de prototipos que pueden servirnos como base para extraer requisitos de nuestro sistema son:

Tabla 4. Principales plataformas para *wireframes*. Fuente: elaboración propia.

Adobe XD

Esta aplicación destaca como una opción fácil de usar para proyectos de diseño de interfaces. Todo, desde la creación de esquemas hasta la creación de prototipos básicos, puede suceder dentro de XD. Y, en comparación con otras herramientas de Adobe ricas en funciones, la interfaz mínima de XD es una bocanada de aire fresco. En el momento en que se abre la aplicación, un conjunto de cajas de luz de incorporación interactivas lo ayuda a orientarse rápidamente con las herramientas de diseño al alcance de su mano. La interfaz limpia y la incorporación rápida son invaluables para los principiantes que desean usar una herramienta profesional de estructura alámbrica sin perder demasiado tiempo tratando de aprender el software. Las herramientas básicas de diseño de vectores de *wireframe* se encuentran fácilmente en la barra de navegación de la izquierda. Los elementos de la interfaz de usuario no están incluidos, pero puede encontrar una variedad de opciones gratuitas con una búsqueda rápida en Google (Adobe, s. f.).

Balsamiq

Es una herramienta rápida de creación de tramas de interfaz de usuario de baja fidelidad que reproduce la experiencia de dibujar en un bloc de notas o pizarra, pero usando una computadora. Realmente obliga a concentrarte en la estructura y el contenido, lo que evita largas discusiones sobre colores y detalles que deberían venir más adelante en el proceso. Puede usarlo para generar bocetos digitales de su idea o concepto para una aplicación o sitio web, para facilitar la discusión y la comprensión antes de escribir cualquier código.

Figura 16. Vista de un *wireframe* creado con Balsamiq. Fuente: Balsamiq, s. f.

Figma

Destaca como una poderosa plataforma de *wireframes* basada en la nube con un proceso sencillo y rápido. Tendrá que diseñar sus propios componentes de interfaz de usuario (o agregarlos desde un kit prediseñado por separado), pero es fácil crear mesas de trabajo, agregar formas y texto e incluso agregar algunos prototipos para que pueda tener una mejor idea del flujo. El panel de la izquierda ayuda a mantener todo organizado, desde capas hasta mesas de trabajo y páginas separadas dentro del mismo documento de diseño. Para un diseño receptivo, se puede aplicar una superposición de columnas (si se prefiere, el sistema de cuadrícula Bootstrap) o se puede usar las restricciones de Figma, que le indican a cada elemento cómo debe responder cuando se cambia el tamaño del diseño para imitar varios tamaños de pantalla. Otra área en la que destaca es en sus oportunidades de colaboración en equipo. Debido a que es una aplicación basada en la web, varios miembros del equipo pueden iniciar sesión y acceder al archivo de diseño simultáneamente, ya sea para modificar el diseño o agregar contenido. La característica más atractiva es la capacidad de un equipo de mantener una conversación completa dentro del archivo de diseño, dejando comentarios similares a notas adhesivas que otros compañeros de equipo pueden responder o marcar como completos. Cuando llega el momento de que su equipo de desarrollo se haga cargo, los desarrolladores pueden tomar el código CSS del interior del archivo de diseño y exportar elementos individuales para usarlos según sea necesario (Figma, s. f.).

InVision Freehand

Es una aplicación abierta para crear o marcar *wireframes* y maquetas en una pizarra donde todos pueden contribuir y revisar. Invite a su equipo a una colaboración de *wireframes*, donde puede agregar imágenes y formas básicas o generar la maqueta con el cursor del mouse. Dado que es una aplicación en la nube, varios diseñadores y partes interesadas pueden iniciar sesión y escribir al mismo tiempo. Por sí mismo, es una buena herramienta de *wireframes*, aunque simplista, que se parece a cualquier otra aplicación de pizarra. Pero dado que ofrece sincronización en tiempo real con Sketch y Photoshop, su equipo puede optar por dibujar y anotar directamente en prototipos pulidos, lo que facilita mucho la etapa de diseño de la iteración. Y, dado que es parte de la *suite* InVision, sus bocetos de colaboración se almacenan junto con prototipos de trabajo y tableros de ideas para una referencia rápida a medida que avanza en su proyecto de diseño. Funciona bien para los diseñadores que desean colaborar con los miembros del equipo en un boceto de *wireframes* y/o comentar sobre prototipos más refinados (InVision, 2021).

Figura 17. Vista de *wireframe* de la aplicación Freehand. Fuente: InVisión, s. f.

Justinmind

No solo es fácil de aprender y divertido de usar, también mantiene un enfoque en la capacitación para crear un *wireframe* que pueda probarse como un prototipo funcional desde el principio. Lo hace al incluir algo que otras aplicaciones simplemente no se han molestado en desarrollar: elementos prototipo interactivos (entradas de texto, botones de radio, menús desplegables, etc.). Incluso a un nivel básico de *wireframe*, un menú desplegable funcional requeriría tres o más pantallas para configurarse en una herramienta como Sketch o Figma. Justinmind te permite agregarlo a tu *wireframe* con un solo clic. De hecho, la facilidad con la que puede crear y compartir *wireframes* realistas puede ahorrarle horas o días de trabajo en cualquier proyecto. Esto convierte a Justinmind en una gran herramienta para cualquier persona que quiera obtener comentarios reales y detallados sobre los diseños mucho antes en la etapa de estructura metálica (con un esfuerzo mínimo). Aparte del contenido de los elementos en sí, el diseño de la herramienta es muy fácil de entender: todos los elementos de diseño están en el lado izquierdo; los elementos de organización (como carpetas, lista de pantallas, etc.) están a la derecha (Justinmind, 2019).

Figura 18. Vista del editor de pantallas de Justinmind. Fuente: Justinmind, s. f.

Mockplus

Además de poder modelar un *wireframe* y agregar notas y documentación a cada elemento de su diseño, el área de proyecto también tiene espacio para que crear y guardar documentación en un editor de texto, guardar la guía de estilo (para que todos estén en la misma página) e incluso crear y asignar tareas al equipo. Dado que las tareas pueden vincularse a un proyecto (e incluso reducirse a un diseño individual dentro del proyecto), realmente puede marcar su proceso de retroalimentación de revisión a la vez que evoluciona con el desarrollo (Mockplus, 2022).

Sketch

Desde su lanzamiento en 2010, se ha mantenido en un lugar destacado como una herramienta de diseño vectorial poderosa pero liviana para usuarios de macOS. Por sí mismo, se puede usar para cualquier cosa,desde *wireframes* hasta una interfaz de usuario moderna y diseño de vectores de iconos y algunos diseños de interacción. Gracias a esta simplicidad, Sketch se puede usar para crear *wireframes* rápidamente con una combinación de mesas de trabajo y formas de diseño vectorial. Cuando se descarga la aplicación Sketch en una Mac, se puede notar que no hay componentes de interfaz de usuario integrados. Si bien podría diseñar sus propios componentes para usarlos como parte de su proceso de estructura alámbrica, existe una gran comunidad en línea de diseñadores que han creado y compartido muchos kits de diseño de *wireframe* gratuitos. Con un solo clic para descargar, tendrá una gran cantidad de botones, íconos y otros elementos de diseño que se pueden usar dentro de su archivo de Sketch. Como la mayoría de las aplicaciones de escritorio, Sketch hace que la colaboración sea un poco complicada, ya que no hay forma de permitir que varios diseñadores trabajen simultáneamente dentro de un diseño (Sketch, 2022).

Figura 19. Vista del editor de pantallas de Sketch. Fuente: Sketch, s. f.

UXPin

Es una de las primeras herramientas recomendadas para cualquier persona que esté aprendiendo a hacer *wireframes*. Si bien su rico conjunto de funciones puede ser un poco difícil de aprender para un nuevo diseñador, el esfuerzo dedicado a aprender la herramienta definitivamente vale la pena. Con UXPin, se pueden comenzar los esquemas con una biblioteca integrada de elementos de la interfaz de usuario que puede arrastrar y soltar directamente en su lienzo. El beneficio de esta *wireframe* de fidelidad ligeramente superior es que puede realizar un ajuste más fino del flujo y la funcionalidad sin perder mucho tiempo rediseñando los componentes de la pantalla. Y, dado que es posible leer archivos de Sketch y Photoshop, siempre se puede usar esas herramientas para convertir sus esquemas básicos en prototipos de alta fidelidad antes de importar el diseño pulido nuevamente a UXPin para aprovechar las otras características de la herramienta: agregar interacciones, presentar a un equipo y entregando las especificaciones de diseño a un desarrollador.

A pesar de las excelentes características de diseño de interfaz y *wireframes*, son las capacidades de presentación y documentación en vivo las que hacen que destaque. Si bien muchas aplicaciones le permiten obtener CSS, HTML o incluso JSON de su diseño, UXPin facilita presentar un prototipo funcional de su diseño, recopilar comentarios e incluir especificaciones/documentación, todo a la vez a través del modo de vista previa (UXPin, 2022).