**GUÍA PARA DESARROLLAR EL TEXTO GUÍA**

PROGRAMA:

PROCESOS AGILES DE SOFTWARE

(Texto Guía)

DEPARTAMENTO OFERENTE:

INGENIERÍA

AUTOR(A):

JORGE ALEJANDRO AGUIRRE GUTIERREZ

FORMACIÓN:

ESPECIALIZACION DE INGENIERÍA DE SOFTWARE CON ÉNFASIS EN PRUEBAS

Manizales, 2024

**Tabla de Contenidos**

**UNIDAD 1: TEORIA SOBRE METODOLOGIAS AGILES.**

1. Conceptos de procesos agiles de software.
2. Beneficios de los procesos agiles de software.
3. Diferencias entre metodologías agiles y tradicionales.
4. Conceptos del SCRUM.
5. Roles del SCRUM.
6. Organización Proyecto y épicas.
7. Backlog.
8. Sprint.
9. Historias de usuario.
10. Evento Planning.
11. Evento Daily.
12. Evento Review.
13. Evento Retrospective.
14. Entrega de proyectos.
15. Conformación de equipos y roles.
16. Creación de organización y proyecto en plataforma DevOps.
17. Creación de épicas e historias de usuario en plataforma DevOps.
18. Practica de realización de Planning.
19. Practica de realización de Daily.
20. Practica de realización de Review.
21. Practica de realización de Retrospectiva.

***Antes de iniciar el texto guía, por favor lea con atención estas recomendaciones generales para el escrito:***

*El texto guía es un escrito académico elaborado por el autor del curso y tiene como principal función “presentar de forma didáctica el conocimiento de las distintas disciplinas…Por su diseño, son textos mixtos que reúnen información textual con otra de tipo visual (uso de imágenes, gráficas, tablas, etc.). También contienen muchos tipos de ayudas didácticas, especialmente seleccionadas y diseñadas, con el propósito de favorecer el aprendizaje (Díaz & Hernández, 2010, p. 158), tales como: videos, simuladores, animaciones, cuestionarios de verificación de lectura, ejercicios interactivos (opción múltiple, falso y verdadero, emparejamiento, clasificar conceptos) o juegos interactivos de Educaplay (*[*Ver tipo de actividades*](https://es.educaplay.com/tipos-de-actividades/)*) con recursos multimedia (video, audio, imagen, texto)*

*Siguiendo a Díaz y Hernández (2010), los textos considerados accesibles se caracterizan por:*

* *Tienen un arreglo estructurado y sistemático de las ideas.*
* *Poseen un buen nivel de coherencia de las ideas expuestas*
* *Son concisos y contienen poca información distractora o irrelevante*
* *Toman en cuenta el conocimiento previo del lector*
* *Presentan al lector la estructura del texto de forma clara (con señalizaciones) para beneficiar su comprensión profunda*

*Por favor tenga en cuenta las siguientes recomendaciones:*

* *Utilizar Verdana tamaño 12, interlineado sencillo.*
* *Las fuentes consultadas para desarrollar el contenido no deben ser de más de cinco años de antigüedad. Si se requiere utilizar fuentes de más años, se debe justificar en las referencias al final de la unidad.*
* *Todas las gráficas, ilustraciones o figuras, deben estar enumeradas con la herramienta de Word REFERENCIAS: INSERTAR TÍTULO. Debe registrarse la FUENTE: Autor, fecha, o sitio web. Cuando la gráfica es de autoría del docente, se debe especificar al final del título de la gráfica: FUENTE: ELABORACIÓN PROPIA.*
* *En el desarrollo del contenido, cuando se trate de ideas fundamentadas en otros autores, se debe hacer la cita correspondiente. Del mismo modo diferenciar claramente las opiniones del docente autor del módulo.*
* *Las citas deben referenciarse con Normas VANCOUVER (PARA CIENCIAS DE LA SALUD) – APA (CIENCIAS SOCIALES E INGENIERÍAS)*

**TÍTULO DE LA UNIDAD:**

**TEORIA SOBRE METODOLOGIAS AGILES**

**INTRODUCCIÓN A LA UNIDAD**

**Indicador de desempeño**

Al finalizar el estudio de esta unidad y desarrollar las actividades propuestas, el estudiante deberá estar en capacidad de:

- Comprender los conceptos de metodologías agiles en la industria del software y las diferencias con respecto a las metodologías tradicionales.

- Conocer la implementación del SCRUM dentro de un equipo de desarrollo y la participación efectiva de cada uno de sus eventos.

**-----------**

**Desarrollar cada uno de los temas propuestos**

***INDICACIONES:***

*Presentación del contenido utilizando estrategias que cumplan con las siguientes funciones:*

* *Detección de la información principal;*
* *Conceptualización de contenidos;*
* *Delimitación de la organización, estructura interna e interrelaciones entre dichos contenidos;*
* *y mantenimiento de la atención y motivación.*

*A lo largo del escrito haga uso de estrategias tales como: ilustraciones, esquemas, gráficas o resúmenes (al iniciar la unidad, al finalizar la unidad, o al finalizar un tema), preguntas intercaladas, analogías, pistas tipográficas (resaltar ideas centrales, conceptos, mapas conceptuales), ejemplificaciones para aclarar conceptos o mostrar aplicaciones, ejercicios.*

*Utilice recursos multimedia como material complementario: audios, videos, animaciones, simuladores, imágenes, entre otros.*

*Recuerde hacer las citas correspondientes con normas VANCOUVER O APA.*

*Recuerde proponer ejercicios interactivos a lo largo del texto para reforzar conocimientos o de verificación de comprensión de la lectura.*

*Recuerde titular y enumerar cada figura, tabla, imagen, gráfica, ilustración y registrar la fuente. Especificar cuando se trate de una gráfica de su propia autoría.*

**TEMA 1: CONCEPTOS DE PROCESOS AGILES DE SOFTWARE**

Un proceso ágil de software es una metodología de desarrollo de software que se basa en la colaboración, la adaptación y la entrega continua de valor. En lugar de seguir un plan rígido y definido desde el principio, los procesos ágiles dividen el trabajo en ciclos cortos de desarrollo, conocidos como iteraciones. En cada iteración, el equipo se enfoca en entregar una funcionalidad específica del software, recibiendo retroalimentación del cliente y adaptándose a los cambios en los requisitos a medida que avanza el proyecto. (Fuente: *Salesforce Latam. 2021. Disponible en:* [*https://www.salesforce.com/mx/blog/que-son-metodologias-agiles/#:~:text=Es%20decir%2C%20una%20metodolog%C3%ADa%20%C3%A1gil,resuelve%20etapas%20en%20poco%20tiempo*](https://www.salesforce.com/mx/blog/que-son-metodologias-agiles/#:~:text=Es%20decir%2C%20una%20metodolog%C3%ADa%20%C3%A1gil,resuelve%20etapas%20en%20poco%20tiempo)*)*

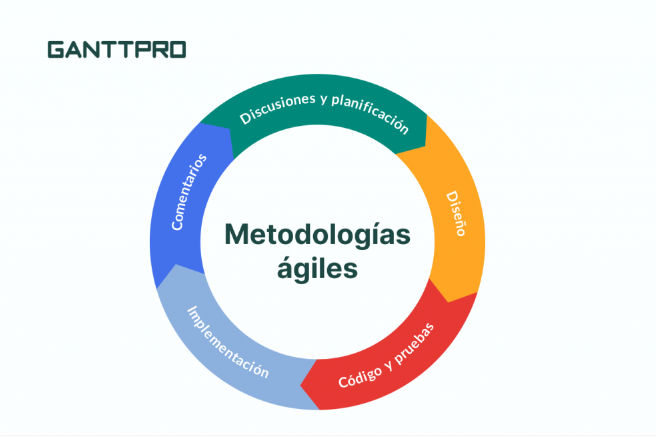
Características clave de los procesos ágiles de software:

* Enfoque en el cliente: La satisfacción del cliente es la máxima prioridad. Se busca involucrarlo activamente en el proceso y obtener su retroalimentación de manera constante.
* Adaptabilidad a los cambios: Los procesos ágiles son flexibles y permiten adaptarse a cambios en los requisitos o prioridades del proyecto sin afectar negativamente el desarrollo.
* Entrega incremental: Se entrega software funcional de manera regular, permitiendo al cliente probarlo y usarlo desde las primeras etapas del proyecto.
* Trabajo en equipo: Los equipos ágiles son multifuncionales y colaborativos, donde cada miembro aporta sus habilidades y conocimientos para el éxito del proyecto.
* Mejora continua: Se busca la mejora constante de los procesos y prácticas a través de la reflexión y la retrospectiva al finalizar cada iteración.



NOTA UAM VIRTUAL: Hacer uso de la anterior imagen y agregarle alrededor las 5 características claves de los procesos agiles de software.

EJEMPLO DE APLICACIÓN DE METODOLOGIA AGIL



NOTA UAM VIRTUAL: Agregar la siguiente imagen como ejemplo de metodología ágil.

**TEMA 2: BENEFICIOS DE LOS PROCESOS AGILES EN LA ORGANIZACIÓN**

Las metodologías ágiles ofrecen una serie de beneficios para las empresas que las implementan, tanto en el ámbito del desarrollo de software como en otros departamentos. Entre los principales beneficios podemos destacar:

1. Mayor satisfacción del cliente:

Al involucrar al cliente en el proceso de desarrollo, se asegura que el producto final cumpla con sus expectativas y necesidades.

La entrega incremental de software funcional permite al cliente probarlo y usarlo desde las primeras etapas del proyecto, proporcionando retroalimentación valiosa para mejorar el producto.

La comunicación constante y la transparencia entre el equipo de desarrollo y el cliente fomentan la confianza y la colaboración.

2. Mayor agilidad y flexibilidad:

Los procesos ágiles permiten adaptarse rápidamente a los cambios en los requisitos o prioridades del proyecto, sin afectar negativamente el desarrollo.

La división del trabajo en ciclos cortos facilita la identificación y corrección de problemas de manera temprana, evitando retrasos costosos al final del proyecto.

La cultura de mejora continua permite al equipo aprender de sus errores y optimizar sus procesos constantemente.

3. Mayor calidad del software:

Las pruebas unitarias continuas y la refactorización constante ayudan a mejorar la calidad del código y reducir la cantidad de errores.

La retroalimentación constante del cliente permite identificar y corregir problemas de usabilidad y funcionalidad antes de que el software llegue a manos del usuario final.

El enfoque en la entrega de valor en cada iteración asegura que el software final sea funcional y útil para el cliente.

4. Mayor motivación del equipo:

Los equipos ágiles suelen estar más motivados y comprometidos con el proyecto, ya que tienen mayor autonomía y responsabilidad sobre su trabajo.

La comunicación abierta y la colaboración entre los miembros del equipo fomentan un ambiente de trabajo positivo y productivo.

El reconocimiento y la celebración de los logros motivan al equipo a seguir mejorando su desempeño.

5. Mayor eficiencia y productividad:

La eliminación de tareas innecesarias y la optimización de los procesos permiten reducir el tiempo y los recursos necesarios para completar el proyecto.

La entrega incremental de software permite liberar valor al cliente de manera más rápida, lo que puede generar ingresos antes de que el proyecto esté finalizado.

La mejora continua de los procesos permite aumentar la eficiencia y la productividad a largo plazo.

*(Fuente: TalentBox, 2022. Disponible en:* [*https://blog.talentbox.la/7-beneficios-de-las-metodologias-agiles-para-tu-empresa/*](https://blog.talentbox.la/7-beneficios-de-las-metodologias-agiles-para-tu-empresa/)*)*



NOTA UAM VIRTUAL: Agregar la anterior imagen de forma interactiva para que el alumno de click en cada ventaja de las metodologías agiles. También agregar el siguiente video sobre los beneficios de las metodologías agiles en diferentes ámbitos empresariales: <https://www.youtube.com/watch?v=j8hNLrnhzFo>

**TEMA 3: DIFERENCIAS ENTRE METODOLOGIAS AGILES Y TRADICIONALES.**

|  |  |  |
| --- | --- | --- |
| **CARACTERÍSTICA** | **METODOLOGÍAS ÁGILES** | **METODOLOGÍAS TRADICIONALES** |
| Enfoque | Adaptativo e iterativo | Predictivo y lineal |
| Planificación | Planificación incremental y flexible | Planificación detallada y por adelantado |
| Entrega de valor | Entrega de valor en ciclos cortos y frecuentes | Entrega de valor al final del proyecto |
| Retroalimentación | Retroalimentación continua y adaptación | Retroalimentación al final de cada fase |
| Cambio | Se fomenta y se abraza el cambio | Se resiste el cambio |
| Colaboración | Se fomenta la colaboración entre equipos | El enfoque es más jerárquico |
| Documentación | Documentación ligera y enfocada | Documentación extensa y detallada |
| Ejemplos | Scrum, Kanban, Extreme Programming (XP) | Waterfall, Ciclo de Vida en Cascada (SDLC) |

*(Fuente: Propia. Jorge Aguirre. 2024)*

NOTA UAM VIRTUAL: Agregar la anterior tabla de forma interactiva y también poner de ejemplo las siguientes dos imágenes.

METODOLOGIAS AGILES

¿Cuál de las siguientes características NO es fundamental de las metodologías ágiles?

(a) Enfoque en la entrega de valor al cliente de manera iterativa e incremental.

(b) Adaptación a cambios y retroalimentación constante.

(c) Planificación y control centralizados.

(d) Colaboración entre equipos multifuncionales.

¿Cuál es el principal objetivo de las metodologías ágiles?

(a) Reducir costos de desarrollo de software.

(b) Aumentar la satisfacción del cliente.

(c) Simplificar la gestión de proyectos.

(d) Todas las anteriores.

¿Cuáles son algunos de los beneficios de utilizar metodologías ágiles?

(a) Mayor flexibilidad y capacidad de adaptación a cambios.

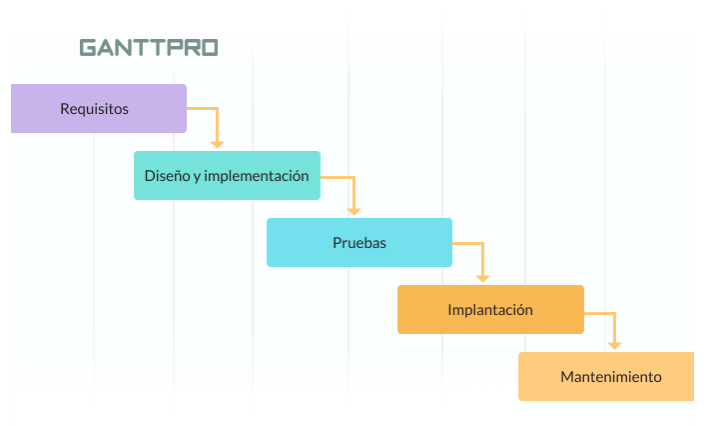
(b) Mejor comunicación y colaboración entre equipos.

(c) Entrega más rápida de valor al cliente.

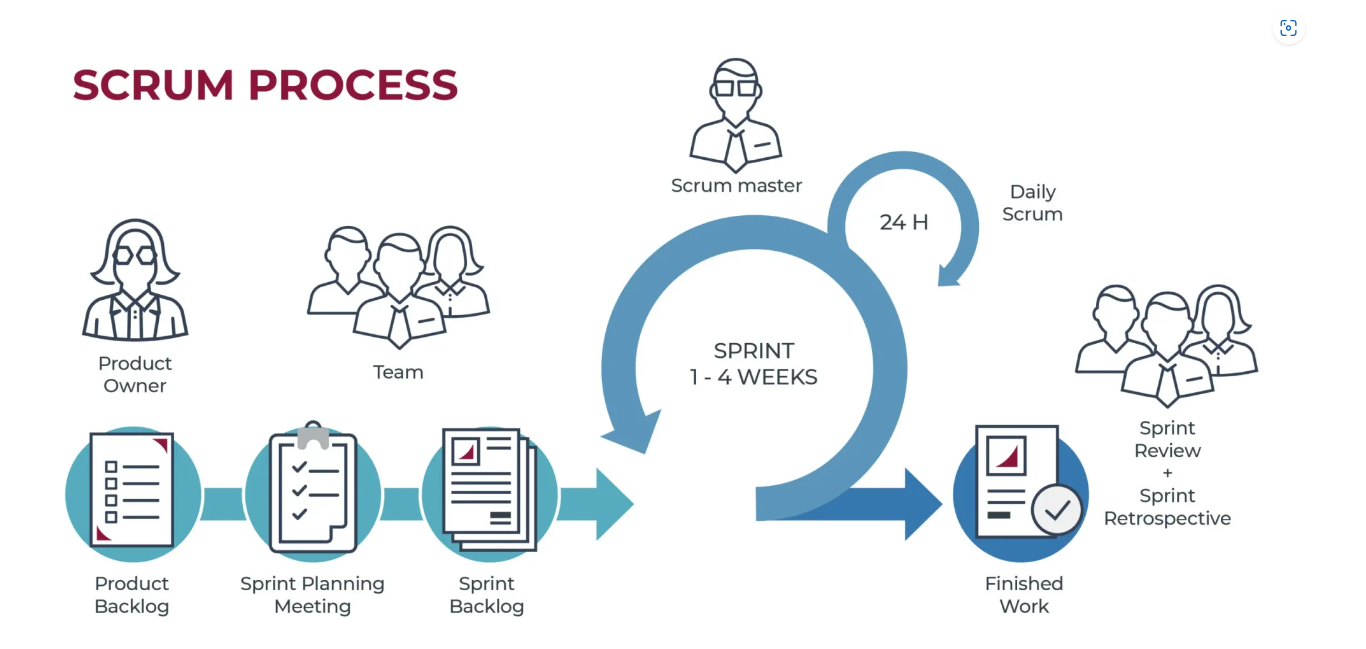
(d) Todos los anteriores

NOTA UAM VIRTUAL: Agregar las anteriores preguntas en formato de opción múltiple con única respuesta.

**ESQUEMA METODOLOGIA TRADICIONAL**



**ESQUEMA METODOLOGIA AGIL**



*(Fuente: Anastasia Stsepanets. GANTTPRO, 2024. Disponible en:* [*https://blog.ganttpro.com/es/metodologia-de-cascada/*](https://blog.ganttpro.com/es/metodologia-de-cascada/)*)*

ACTIVIDAD:

Evaluar la comprensión de los estudiantes sobre las diferencias entre las metodologías ágiles y tradicionales.

Instrucciones:

Dividir la clase en dos grupos: "Metodologías Ágiles" y "Metodologías Tradicionales".

Asigne a cada grupo una lista de características que corresponden a su metodología asignada. Las características pueden incluir:

* Enfoque iterativo y adaptativo (Metodología Agil)
* Priorización del feedback del cliente (Metodología agil)
* Entregas finales (Metodología tradicional)

Rigidez ante cambios (Metodología tradicional)

* Trabajo en equipo colaborativo (Metodología agil)
* Entregas incrementales (Metodología agil)
* Flexibilidad ante cambios (Metodología agil)
* Enfoque lineal y secuencial (Metodología tradicional)
* Planificación detallada por adelantado (Metodología tradicional)
* Estructura jerárquica (Metodología tradicional)

NOTA UAM VIRTUAL: Agregar la anterior actividad para que el estudiante seleccione a que hace referencia cada item, si a una metodología ágil o una tradicional.

**TEMA 4: CONCEPTOS DEL SCRUM**

Scrum es una metodología de gestión de proyectos ágil que se basa en ciclos cortos de trabajo llamados sprints. En cada sprint, el equipo se enfoca en entregar una parte valiosa del producto final.

Caso de uso comparativo: Equipo de futbol.

Imaginemos un equipo deportivo entrenando para un partido importante:

* El entrenador: Es como el Scrum Master, guía al equipo y facilita el proceso.
* Los jugadores: Son como el Equipo de Desarrollo, trabajan juntos para crear el producto final.
* Los entrenamientos: Son como los Sprints, cada uno con un objetivo específico.
* El plan de juego: Es como el Backlog del Producto, una lista de todas las tareas que se deben completar.
* Las reuniones: Son como las Ceremonias Scrum, momentos para planificar, revisar y celebrar el progreso.



Scrum es como un juego de equipo que tiene las siguientes ventajas:

* Todos colaboran: Cada miembro del equipo tiene un rol importante que jugar.
* Comunicación constante: La comunicación abierta y fluida es esencial.
* Adaptación a los cambios: Los planes pueden cambiar a medida que se avanza, el equipo se adapta a los nuevos desafíos.
* Enfoque en el valor: Se prioriza la entrega de valor al cliente en cada sprint.
* Mejora continua: El equipo busca constantemente formas de mejorar su proceso y producto.

*(Fuente: Propia, Jorge Aguirre. 2024)*

NOTA UAM VIRTUAL: Hacer uso de la anterior imagen para que el estudiante haga click mostrando la información del caso comparativo con respecto a un equipo de futbol y las ventajas.

**TEMA 5: ROLES DEL SCRUM**

1. Scrum Master: El guía sabio

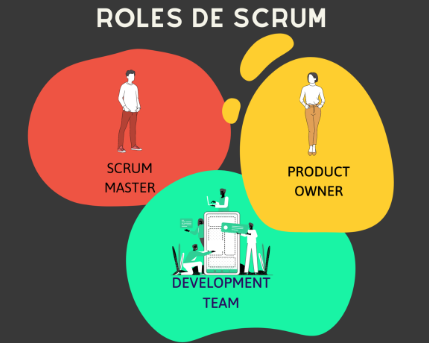
* Misión: Guiar y facilitar el proceso Scrum, eliminando obstáculos y asegurando que el equipo siga los principios Scrum.
* Habilidades: Liderazgo, comunicación, resolución de problemas y conocimiento de Scrum.
* Acciones: Convocar y facilitar las ceremonias Scrum, proteger el tiempo del equipo, resolver conflictos y motivar al equipo.

2. Product Owner: El visionario

* Misión: Representar las necesidades del cliente y definir la visión del producto.
* Habilidades: Comprensión del negocio, comunicación, priorización y visión estratégica.
* Acciones: Crear y mantener el backlog del producto, priorizar las tareas, colaborar con el equipo y recopilar feedback de las partes interesadas.

3. Equipo de Desarrollo: Los héroes valientes

* Misión: Desarrollar el producto de acuerdo con el backlog del producto.
* Habilidades: Habilidades técnicas, trabajo en equipo, autoorganización y resolución de problemas.
* Acciones: Estimar tareas, diseñar, desarrollar, probar y entregar el producto.



*(Fuente: CABSA. Digital Solutions 2024. Disponible en:* [*https://www.cabsa.es/blog/3-roles-de-scrum-y-sus-caractersticas*](https://www.cabsa.es/blog/3-roles-de-scrum-y-sus-caractersticas)*)*

NOTA UAM VIRTUAL: Hacer uso de la anterior imagen para que el estudiante pueda dar click a cada uno y muestre su misión, habilidades y acciones.

PREGUNTAS

¿Cuál es el rol responsable de la gestión del backlog del producto en Scrum?

(a) Scrum Master

(b) Dueño del Producto

(c) Equipo de Desarrollo

(d) Ninguna de las anteriores

¿Cuál es el rol responsable de facilitar el proceso Scrum y eliminar los impedimentos para el equipo?

(a) Scrum Master

(b) Dueño del Producto

(c) Equipo de Desarrollo

(d) Ninguna de las anteriores

¿Cuál es el rol responsable de desarrollar el software de acuerdo con los requisitos del producto?

(a) Scrum Master

(b) Dueño del Producto

(c) Equipo de Desarrollo

(d) Ninguna de las anteriores

¿Cuál es el rol responsable de priorizar los elementos del backlog del producto?

(a) Scrum Master

(b) Dueño del Producto

(c) Equipo de Desarrollo

(d) Ninguna de las anteriores

NOTA UAM VIRTUAL: Agregar las anteriores preguntas en formato de opción múltiple con única respuesta.

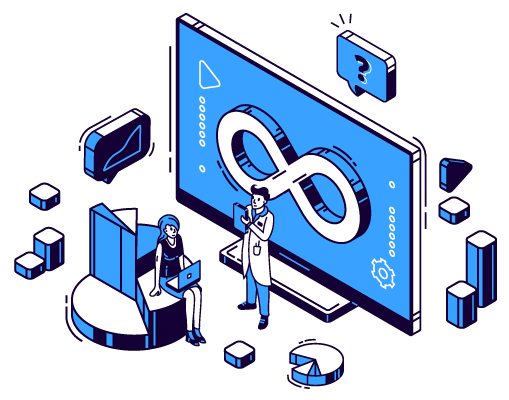
**TEMA 6: LA ORGANIZACIÓN, LOS PROYECTOS Y LAS EPICAS**

En el mundo de Scrum, la organización y las épicas son dos piezas fundamentales que permiten a los equipos abordar proyectos complejos de manera eficiente y efectiva.

**Organización:**

Imagina una biblioteca gigante con miles de libros. Sin un sistema de organización, encontrar el libro que buscas sería una tarea titánica. Lo mismo sucede en Scrum. La organización tiene las siguientes características:

* Estructurar el trabajo: El backlog del producto es la biblioteca de Scrum, donde se encuentran todas las historias de usuario, funcionalidades y mejoras que se deben implementar. Una buena organización del backlog permite priorizar las tareas, facilitar la planificación y garantizar que el equipo se enfoque en lo que realmente importa.
* Desglosar grandes objetivos: Las épicas son como capítulos dentro de la historia del producto. Representan grandes objetivos que se descomponen en historias de usuario más pequeñas y manejables. Esto facilita la comprensión del proyecto, la estimación del esfuerzo y la gestión de las dependencias.
* Visualizar el progreso: El tablero Kanban es como un mapa del tesoro para el equipo Scrum. Permite visualizar el flujo de trabajo, identificar cuellos de botella y celebrar los avances. Una buena organización del tablero Kanban proporciona transparencia y facilita la colaboración entre los miembros del equipo.



NOTA UAM VIRTUAL: Hacer uso de la anterior imagen para que de forma dinámica se muestren las características de una organización.

**Proyectos:**

Es un espacio de trabajo centralizado para planificar, desarrollar y entregar software. Contiene las siguientes caracteristicas:

Administrar tu código fuente:

* Almacena tu código en un repositorio Git seguro y accesible.
* Realiza un seguimiento de los cambios con un historial de versiones completo.
* Colabora con tu equipo en el código de forma eficiente.

Planificar y realizar un seguimiento del trabajo:

* Crea tableros de backlogs para visualizar tu flujo de trabajo.
* Asigna tareas a los miembros del equipo y realiza un seguimiento de su progreso.
* Define sprints y ciclos de lanzamiento para organizar tu trabajo.

Realizar pruebas continuas:

* Automatiza la ejecución de pruebas unitarias, de integración y funcionales.
* Obtén información inmediata sobre los resultados de las pruebas.
* Identifica y soluciona errores rápidamente.

Implementar software de forma continua:

* Crea builds de compilación y lanzamiento automatizadas.
* Implementa tu software en entornos de producción de forma segura y confiable.
* Monitoriza el rendimiento de tu software en producción.

Colaborar con tu equipo:

* Comunícate con tu equipo a través de chats, foros y notificaciones.
* Comparte código, documentos y otros recursos.
* Obtén información sobre el trabajo de tu equipo en tiempo real.

*(Fuente: Microsoft. Azure DevOps, 2024. Disponible en:* [*https://learn.microsoft.com/es-es/azure/devops/organizations/projects/about-projects?view=azure-devops*](https://learn.microsoft.com/es-es/azure/devops/organizations/projects/about-projects?view=azure-devops)*)*



NOTA UAM: Hacer uso de la anterior imagen para que de forma dinámica se muestren las características de una organización.

**Épicas:**

Las épicas son como las grandes batallas en una guerra. Son objetivos a gran escala que abarcan varias historias de usuario y funcionalidades relacionadas. Dividir el proyecto en épicas ofrece múltiples beneficios:

* Agrupar historias relacionadas: Las épicas permiten agrupar historias de usuario que comparten un objetivo común. Esto facilita la comprensión del proyecto, la estimación del esfuerzo y la planificación de las entregas.
* Gestión de dependencias: Las épicas pueden tener dependencias entre sí, lo que significa que algunas deben completarse antes que otras. Identificar y gestionar estas dependencias es crucial para evitar retrasos y garantizar el éxito del proyecto.
* Comunicación efectiva: Las épicas son una excelente herramienta de comunicación para las partes interesadas. Permiten comunicar los objetivos generales del proyecto y el progreso general.7

*(Fuente: Microsoft. Azure DevOps, 2024. Disponible en:* [*https://learn.microsoft.com/es-es/azure/devops/boards/backlogs/define-features-epics?view=azure-devops&tabs=agile-process*](https://learn.microsoft.com/es-es/azure/devops/boards/backlogs/define-features-epics?view=azure-devops&tabs=agile-process)*)*

**TEMA 7: EL BACKLOG**

En Scrum, el backlog es como un mapa del tesoro que guía al equipo hacia el éxito.

¿Qué es el backlog?

El backlog es una lista priorizada de todas las tareas, funcionalidades y mejoras que se deben implementar en un producto. Es como una lista de compras para un chef: contiene todos los ingredientes necesarios para crear una deliciosa comida.

¿Quién crea el backlog?

El Product Owner es el responsable de crear y mantener el backlog. Es como el chef de la aventura, quien decide qué ingredientes se necesitan y en qué orden se deben cocinar.

¿Cómo se organiza el backlog?

El backlog se organiza en historias de usuario, que son descripciones breves y claras de las necesidades o deseos de los usuarios. Cada historia de usuario debe tener un valor claro y ser estimable en términos de tiempo y esfuerzo.

¿Cómo se prioriza el backlog?

El Product Owner prioriza el backlog en función del valor que cada historia de usuario aporta al producto. Las historias de usuario con mayor valor se colocan en la parte superior del backlog, mientras que las menos importantes se colocan en la parte inferior.

¿Cómo se utiliza el backlog?

El backlog se utiliza durante la planificación del sprint para seleccionar las historias de usuario que se completarán en el siguiente sprint. El equipo de desarrollo estima el tiempo que llevará completar cada historia de usuario y se compromete a entregarlas al final del sprint.

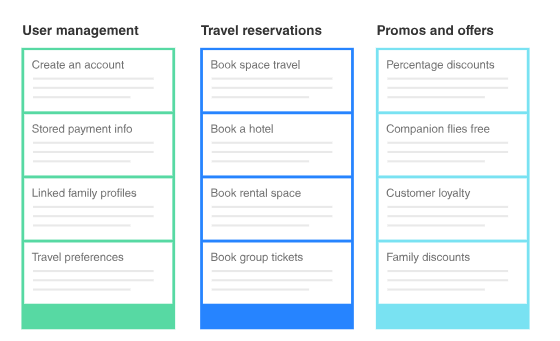
*(Fuente: Propia, Jorge Aguirre. 2024)*

NOTA UAM VIRTUAL: Realizar un acordeón con cada una de las preguntas anteriores para que el estudiante vea las respuestas

Video: Que es el Backlog para la empresa Atlassian dueño de la plataforma Jira <https://youtu.be/zkZJxvKzILY>

NOTA UAM VIRTUA: Agregar el anterior video sobre el backlog.

Ejemplo de Backlog



(Fuente: De Dan Radigan, Atlassian. 2024. Disponible en: <https://www.atlassian.com/es/agile/scrum/backlogs>)

PREGUNTAS

¿Qué es el backlog?

a) Una lista de historias de usuario pendientes para un proyecto.

b) Una herramienta de gestión de tiempo para organizar las actividades diarias.

c) Un software para el seguimiento de errores y defectos en un proyecto.

d) Una lista priorizada de funcionalidades y mejoras para un producto.

¿Quién es responsable de crear y priorizar el backlog?

a) El equipo de desarrollo.

b) El Scrum Master.

c) El Product Owner.

d) Las partes interesadas del proyecto.

¿Cuáles son algunos de los beneficios de utilizar el backlog para la gestión de proyectos ágiles?

a) Mejora la planificación y la visibilidad del trabajo.

b) Fomenta la colaboración y la comunicación entre el equipo.

c) Permite una mayor flexibilidad y adaptación a cambios.

d) Todas las opciones anteriores son correctas.

NOTA UAM VIRTUAL: Agregar las anteriores preguntas en formato de opción múltiple con única respuesta.

**TEMA 8: SPRINT**

Un sprint es un período corto de tiempo (generalmente de 1 a 4 semanas) durante el cual el equipo de Scrum se enfoca en completar un conjunto específico de historias de usuario. Es como una carrera corta con un objetivo definido: entregar valor al cliente al final del sprint.

¿Qué sucede en un sprint?

Un sprint se divide en varias etapas:

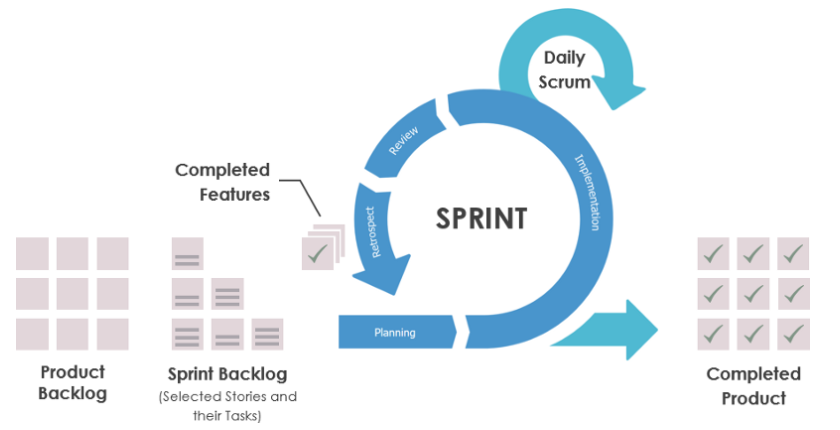
1. Planificación del sprint: Al inicio del sprint, el equipo selecciona las historias de usuario del backlog del producto que se completarán en el sprint.
2. Desarrollo: Durante el sprint, el equipo de desarrollo trabaja en las historias de usuario seleccionadas, utilizando sus habilidades y experiencia para crear un producto que funcione y satisfaga las necesidades del cliente.
3. Revisión diaria: Cada día, el equipo se reúne para discutir su progreso, identificar problemas y ajustar su plan según sea necesario.
4. Revisión del sprint: Al final del sprint, el equipo presenta lo que ha completado a las partes interesadas.
5. Retrospectiva del sprint: Después de la revisión del sprint, el equipo reflexiona sobre lo que salió bien y lo que no, y busca formas de mejorar su proceso para el próximo sprint.

¿Por qué son importantes los sprints?

Los sprints son importantes por varias razones:

* Entrega de valor frecuente: Los sprints permiten al equipo entregar valor al cliente de forma regular, lo que ayuda a obtener feedback temprano y a mejorar el producto de forma continua.
* Enfoque y motivación: Los sprints ayudan al equipo a mantenerse enfocado en un objetivo específico durante un período corto de tiempo, lo que aumenta la motivación y la productividad.
* Adaptación a los cambios: Los sprints permiten al equipo adaptarse a los cambios en las necesidades del cliente o del mercado de forma rápida y flexible.
* Mejora continua: Los sprints promueven una cultura de mejora continua, donde el equipo aprende de sus experiencias y busca formas de mejorar su proceso en cada sprint.

(Fuente: De Max Rehkopf. Atlassian, 2024. Disponible en: <https://www.atlassian.com/es/agile/scrum/sprints>)



NOTA UAM VIRTUAL: Realizar un acordeón con cada una de las preguntas anteriores para que el estudiante vea las respuestas.

¿Qué es un sprint?

a) Una fase de planificación del proyecto que define los objetivos generales.

b) Un período de tiempo fijo en el que se desarrolla un conjunto específico de funcionalidades.

c) Una reunión donde se revisan los avances del proyecto y se toman decisiones.

d) Una herramienta de gestión de tareas para organizar el trabajo del equipo.

¿Cuál es la duración típica de un sprint?

a) Una semana o menos.

b) Una a cuatro semanas.

c) Un mes o más.

d) No hay una duración estándar, depende del proyecto.

¿Qué actividades se llevan a cabo durante un sprint?

a) Planificación, desarrollo, revisión y retrospectiva.

b) Diseño, implementación, pruebas y despliegue.

c) Investigación, análisis, documentación y capacitación.

d) Todas las opciones anteriores pueden ocurrir en un sprint.

¿Quién es responsable de definir las funcionalidades que se desarrollarán en un sprint?

a) El Scrum Master.

b) El Product Owner.

c) El equipo de desarrollo.

d) Todas las partes interesadas en el proyecto.

¿Cuáles son algunos de los beneficios de utilizar sprints en la gestión de proyectos ágiles?

a) Promueven la entrega rápida y frecuente de valor al cliente.

b) Facilitan la adaptación a cambios en los requisitos o prioridades.

c) Mejoran la comunicación y la colaboración entre el equipo.

d) Todas las opciones anteriores son correctas.

NOTA UAM VIRTUAL: Agregar las anteriores preguntas en formato de opción múltiple con única respuesta.

**TEMA 9: HISTORIAS DE USUARIO**

Las Historias de usuario son ¡La voz del cliente en Scrum!.

En el mundo de Scrum, las historias de usuario son como las voces de los clientes que guían al equipo hacia la creación de un producto que realmente satisfaga sus necesidades.

¿Qué son las historias de usuario?

Las historias de usuario son descripciones breves y claras de las necesidades o deseos de los usuarios. Son como pequeñas historias que describen cómo un usuario específico quiere interactuar con el producto.

¿Cómo se escriben las historias de usuario?

Las historias de usuario se escriben siguiendo una estructura simple:

Como: Esta frase describe al usuario que está realizando la acción.

Yo quiero: Esta frase describe la necesidad o el deseo del usuario.

Para que: Esta frase explica el beneficio o el objetivo que el usuario quiere lograr.

Ejemplo de una historia de usuario:

Como cliente, yo quiero poder buscar productos por categoría para que pueda encontrar fácilmente lo que busco.

¿Por qué son importantes las historias de usuario?

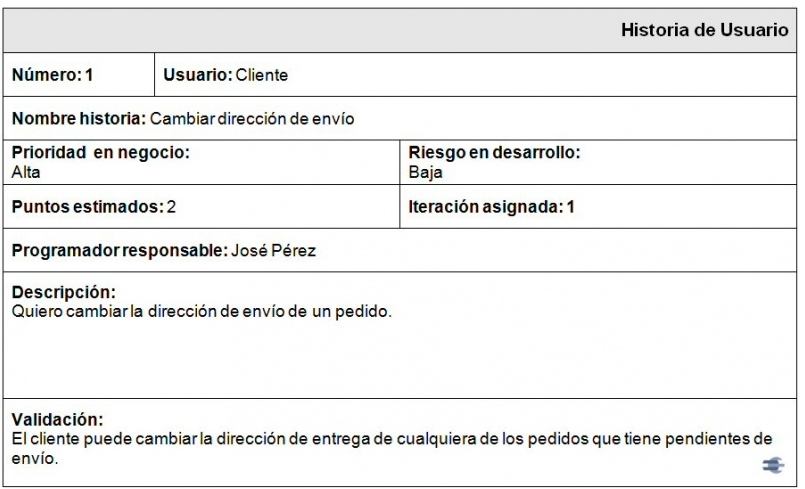
Las historias de usuario son importantes por varias razones:

* Capturan las necesidades del cliente: Las historias de usuario se basan en las necesidades y deseos reales de los clientes, lo que garantiza que el producto esté diseñado para satisfacer sus necesidades.
* Promueven la colaboración: Las historias de usuario facilitan la colaboración entre el Product Owner y el equipo de desarrollo, asegurando que todos estén en la misma página.
* Facilitan la estimación: Las historias de usuario son fáciles de entender y estimar, lo que ayuda al equipo a planificar los sprints de manera efectiva.
* Guían el desarrollo: Las historias de usuario sirven como guía para el equipo de desarrollo, asegurando que el producto se desarrolle de acuerdo con las necesidades del cliente.

(Fuente: Propia, Jorge Aguirre. 2024)

NOTA UAM VIRTUAL: Realizar acordeón de información con las anteriores preguntas.

EJEMPLO DE HISTORIA DE USUARIO:



*(Fuente: SCRUM Manager, 2024. Disponible en:* [*https://www.scrummanager.com/bok/index.php/Historia\_de\_usuario*](https://www.scrummanager.com/bok/index.php/Historia_de_usuario)*)*

NOTA UAM VIRTUAL: Agregar la anterior imagen como ejemplo de historia de usuario.

PREGUNTAS

¿Qué es una historia de usuario?

a) Una descripción de un requisito funcional o no funcional de un sistema.

b) Una conversación informal entre un usuario y un desarrollador sobre una necesidad.

c) Una tarjeta física que se utiliza para registrar y priorizar tareas en un tablero Kanban.

d) Una forma de expresar el valor que un producto o servicio aporta a un usuario.

¿Cuál es la estructura típica de una historia de usuario?

a) Como [rol], quiero [acción] para [objetivo].

b) [Requisito funcional] debe ser implementado para que [usuario] pueda [beneficio].

c) Necesito [funcionalidad] para poder [completar tarea].

d) [Usuario] debe ser capaz de [realizar acción] en [contexto].

¿Quién es responsable de escribir las historias de usuario?

a) Los desarrolladores de software.

b) El Product Owner.

c) Los usuarios finales del producto.

d) Todas las partes interesadas en el proyecto.

¿Qué características debe tener una historia de usuario bien escrita?

a) Completa, detallada y específica.

b) Independiente, negociable y valorable.

c) Grande, compleja y difícil de estimar.

d) Abierta, ambigua y sujeta a interpretación.

¿Cuáles son algunos de los beneficios de utilizar historias de usuario para la gestión de proyectos ágiles?

a) Mejoran la comunicación y la colaboración entre el equipo y los usuarios.

b) Facilitan la priorización y la estimación del esfuerzo de desarrollo.

c) Promueven un enfoque centrado en el usuario y en la entrega de valor.

d) Todas las opciones anteriores son correctas.

NOTA UAM VIRTUAL: Agregar las anteriores preguntas en formato de opción múltiple con única respuesta.

**TEMA 10: EVENTO PLANNING**

El Planning, también conocido como planificación del sprint, es una ceremonia esencial en Scrum que se realiza al inicio de cada sprint. Es como una reunión crucial donde el equipo se prepara para la aventura que les espera.

¿Qué sucede en el Planning?

En el Planning, el equipo se reúne con el Product Owner para:

1. Revisar el backlog del producto: Se analizan las historias de usuario priorizadas y se seleccionan las que se completarán en el sprint.
2. Estimar el esfuerzo: El equipo estima el tiempo y esfuerzo que requerirá completar cada historia de usuario seleccionada.
3. Crear el backlog del sprint: Se crea una lista de las historias de usuario seleccionadas para el sprint, junto con su estimación de esfuerzo.
4. Definir el objetivo del sprint: Se establece un objetivo claro y conciso para el sprint, que describe lo que el equipo se compromete a entregar al final de este.

¿Por qué es importante el Planning?

* Alineación del equipo: Asegura que todos los miembros del equipo estén en la misma página y comprendan el objetivo del sprint.
* Enfoque y priorización: Ayuda al equipo a enfocarse en las historias de usuario más importantes y priorizadas.
* Planificación realista: Permite al equipo planificar el sprint de manera realista, teniendo en cuenta el tiempo y esfuerzo disponibles.
* Mejora continua: Brinda la oportunidad de reflexionar sobre los sprints anteriores y buscar formas de mejorar el proceso de planificación.

*(Fuente: Dave West, Atlassian. 2024. Disponible en:* [*https://www.atlassian.com/es/agile/scrum/sprint-planning*](https://www.atlassian.com/es/agile/scrum/sprint-planning)*)*



*(Fuente: GeeksForGeeks, 2024. Disponible en:* [*https://www.geeksforgeeks.org/features-importance-and-limitations-of-planning/*](https://www.geeksforgeeks.org/features-importance-and-limitations-of-planning/)*)*

NOTA UAM VIRTUAL: Agregar las anteriores preguntas en forma de acordeón con sus respuestas, incluyendo la imagen.

PREGUNTAS

¿Cuál es el objetivo principal del Sprint Planning?

a) Definir las funcionalidades que se desarrollarán en el siguiente sprint.

b) Estimar el tiempo necesario para completar cada funcionalidad.

c) Priorizar las funcionalidades del product backlog.

d) Todas las opciones anteriores son correctas.

¿Quién participa en el Sprint Planning?

a) El Scrum Master, el Product Owner y el equipo de desarrollo.

b) El Product Owner y las partes interesadas del proyecto.

c) El Scrum Master y el equipo de desarrollo.

d) Todas las partes interesadas en el proyecto.

¿Qué es el Sprint Backlog?

a) Una lista de todas las funcionalidades que se desarrollarán en el proyecto.

b) Un subconjunto del product backlog que contiene las funcionalidades para el sprint.

c) Un plan detallado de cómo se implementará cada funcionalidad.

d) Un documento que describe los objetivos y riesgos del sprint.

NOTA UAM VIRTUAL: Agregar las anteriores preguntas en formato de opción múltiple con única respuesta.

**TEMA 11: EVENTO DAILY**

El Daily Scrum es una reunión breve (máximo 15 minutos) que se realiza todos los días durante un sprint. Es como una breve parada dentro de la cueva para que los exploradores se comuniquen y planifiquen su siguiente paso.

¿Qué sucede en el Daily Scrum?

En el Daily Scrum, cada miembro del equipo responde tres preguntas:

* ¿Qué hice ayer para completar el objetivo del sprint?
* ¿Qué haré hoy para completar el objetivo del sprint?
* ¿Hay algún impedimento que me impida alcanzar el objetivo del sprint?
* ¿Por qué es importante el Daily Scrum?

¿Por qué es importante el Daily Scrum?

* Mejora la comunicación: Fomenta la comunicación diaria entre los miembros del equipo, lo que ayuda a evitar malentendidos y mantener a todos en la misma página.
* Identifica y resuelve problemas: Permite al equipo identificar y resolver problemas de manera temprana, evitando que se conviertan en obstáculos mayores.
* Promueve la colaboración: Fomenta la colaboración entre los miembros del equipo, permitiéndoles compartir ideas y ayudarse mutuamente.
* Adaptación al cambio: Permite al equipo adaptarse a los cambios en las prioridades o en el entorno del proyecto de manera rápida y flexible.
* Mejora continua: Brinda la oportunidad de reflexionar sobre el progreso diario y buscar formas de mejorar el proceso en el siguiente sprint.

*(Fuente: Propia, Jorge Aguirre. 2024)*



NOTA UAM VIRTUAL: Agregar las anteriores preguntas en forma de acordeón con sus respuestas, incluyendo la imagen.

PREGUNTAS

¿Cuál es el propósito del Daily?

a) Sincronizar el trabajo del equipo y discutir los avances del sprint.

b) Resolver problemas y eliminar obstáculos que impiden el progreso.

c) Ajustar el Sprint Backlog en caso de cambios en las prioridades.

d) Todas las opciones anteriores son correctas.

¿Quién participa en el Daily Scrum?

a) El Scrum Master, el Product Owner y el equipo de desarrollo.

b) El Product Owner y las partes interesadas del proyecto.

c) El Scrum Master y el equipo de desarrollo.

d) Solo el equipo de desarrollo.

¿Cuánto tiempo debe durar el Daily Scrum?

a) Menos de 15 minutos.

b) Entre 15 y 30 minutos.

c) Entre 30 y 45 minutos.

d) No hay un tiempo límite definido.

NOTA UAM VIRTUAL: Agregar las anteriores preguntas en formato de opción múltiple con única respuesta.

**TEMA 12: EVENTO REVIEW**

La review es una ceremonia esencial en Scrum que se realiza al final de cada sprint. Es como una reunión final donde los aventureros comparten sus logros y aprenden de sus experiencias.

¿Qué sucede en la review?

En la review, el equipo:

* Presenta lo que ha completado durante el sprint: Demostrará las funcionalidades desarrolladas, las historias de usuario finalizadas y cualquier otro avance significativo.
* Recibe feedback de las partes interesadas: Las partes interesadas brindarán sus comentarios sobre el trabajo realizado, identificando aspectos positivos y áreas de mejora.
* Responde preguntas y aclara dudas: El equipo responderá a las preguntas de las partes interesadas y aclarará cualquier duda que puedan tener.
* Reflexiona sobre el sprint: El equipo analizará el proceso de desarrollo, identificando lo que funcionó bien y lo que se puede mejorar en el futuro.

¿Por qué es importante la review?

La review es importante por varias razones:

* Transparencia: Permite a las partes interesadas ver el progreso del proyecto y brindar feedback valioso.
* Inspección y adaptación: Brinda la oportunidad de inspeccionar el producto y adaptar el proceso según sea necesario.
* Celebración de los logros: Es un momento para celebrar los logros del equipo y reconocer su esfuerzo.
* Mejora continua: Promueve la mejora continua del proceso de desarrollo, identificando áreas de mejora y buscando nuevas formas de trabajar de manera más efectiva.

*(Fuente: Dan Radigan, Atlassian. 2024. Disponible en:* [*https://www.atlassian.com/agile/scrum/sprint-reviews*](https://www.atlassian.com/agile/scrum/sprint-reviews)*)*



NOTA UAM VIRTUAL: Agregar las anteriores preguntas en forma de acordeón con sus respuestas, incluyendo la imagen.

PREGUNTAS

¿Cuál es el objetivo principal del Sprint Review?

a) Demostrar el trabajo completado en el sprint a las partes interesadas.

b) Obtener retroalimentación sobre las funcionalidades desarrolladas.

c) Ajustar el product backlog en base a las necesidades del cliente.

d) Todas las opciones anteriores son correctas.

¿Quién participa en el Sprint Review?

a) El Scrum Master, el Product Owner, el equipo de desarrollo y las partes interesadas.

b) El Product Owner y las partes interesadas del proyecto.

c) El Scrum Master y el equipo de desarrollo.

d) Solo el equipo de desarrollo y las partes interesadas.

¿Qué tipo de preguntas se suelen hacer en el Sprint Review?

a) ¿Qué se ha completado?

b) ¿Qué problemas se encontraron?

c) ¿Qué se puede mejorar?

d) Todas las opciones anteriores son correctas.

NOTA UAM VIRTUAL: Agregar las anteriores preguntas en formato de opción múltiple con única respuesta.

**TEMA 13: EVENTO RETROSPECTIVA**

En el mundo del desarrollo ágil, una retrospectiva es una reunión o taller que se lleva a cabo al final de cada ciclo de trabajo (sprint o iteración) para reflexionar sobre lo que salió bien, lo que no funcionó y cómo se puede mejorar el proceso en el futuro. Es un espacio para que el equipo aprenda de sus experiencias y trabaje en conjunto para identificar áreas de mejora.

¿Qué sucede en una retrospectiva?

1. Establecer el escenario: El facilitador de la retrospectiva comienza la reunión dando la bienvenida a todos y estableciendo el propósito de la reunión.
2. Recopilar datos: El equipo luego trabaja en conjunto para recopilar datos sobre su trabajo reciente. Esto se puede hacer utilizando una variedad de técnicas, como lluvia de ideas, encuestas o simplemente tener una discusión abierta.
3. Analizar los datos: Una vez que se hayan recopilado los datos, el equipo los analiza para identificar patrones y tendencias.
4. Identificar áreas de mejora: El equipo luego identifica áreas específicas donde pueden mejorar su trabajo.
5. Generar acciones: Finalmente, el equipo genera acciones que tomarán para mejorar su trabajo en el futuro.

¿Por qué es importante la retrospectiva?

* Ayudan a los equipos a mejorar continuamente: Al reflexionar sobre su trabajo, los equipos pueden identificar áreas donde pueden mejorar y luego tomar medidas para realizar esas mejoras. Esto puede conducir a una mayor productividad, calidad y satisfacción del cliente.
* Fomentan el trabajo en equipo: Las retrospectivas proporcionan una oportunidad para que los miembros del equipo se reúnan y discutan su trabajo. Esto puede ayudar a fomentar la confianza y la comprensión entre los miembros del equipo, lo que puede conducir a una mejor colaboración.
* Ayudan a los equipos a adaptarse al cambio: El mundo de los negocios está en constante cambio, y los equipos deben poder adaptarse para seguir siendo competitivos. Las retrospectivas pueden ayudar a los equipos a identificar y adaptarse a los cambios en su entorno.

*(Fuente: Atlassian, 2024. Disponible en:* [*https://www.atlassian.com/team-playbook/plays/retrospective*](https://www.atlassian.com/team-playbook/plays/retrospective)*)*



NOTA UAM VIRTUAL: Agregar las anteriores preguntas en forma de acordeón con sus respuestas, incluyendo la imagen.

PREGUNTAS

¿Cuál es el objetivo principal del Sprint Retrospectiva?

a) Reflexionar sobre el sprint pasado e identificar oportunidades de mejora.

b) Analizar las causas de los problemas y definir acciones correctivas.

c) Celebrar los éxitos del equipo y agradecer su esfuerzo.

d) Todas las opciones anteriores son correctas.

¿Quién participa en el Sprint Retrospectiva?

a) El Scrum Master, el Product Owner, el equipo de desarrollo y las partes interesadas.

b) El Product Owner y las partes interesadas del proyecto.

c) El Scrum Master y el equipo de desarrollo.

d) Solo el equipo de desarrollo.

¿Qué tipo de preguntas se suelen hacer en el Sprint Retrospectiva?

a) ¿Qué salió bien?

b) ¿Qué no salió bien?

c) ¿Qué podemos hacer mejor en el próximo sprint?

d) Todas las opciones anteriores son correctas.

NOTA UAM VIRTUAL: Agregar las anteriores preguntas en formato de opción múltiple con única respuesta.

**TEMA 14: ENTREGA DE PROYECTOS COMO CLIENTE.**

PRIMER PROYECTO VIAJAYA

|  |  |
| --- | --- |
| Descripción | Detalles |
| Funcionalidades básicas de búsqueda y reserva | - Búsqueda de vuelos, hoteles y paquetes turísticos por destino, fechas, precio, etc. - Filtros de búsqueda avanzados por tipo de alojamiento, servicios, características, etc. - Visualización de información detallada sobre vuelos, hoteles y paquetes turísticos. - Posibilidad de comparar diferentes opciones de vuelos, hoteles y paquetes turísticos. - Reserva de vuelos, hoteles y paquetes turísticos en línea. - Pago seguro en línea. - Confirmación de la reserva por correo electrónico y SMS. |
| Gestión de reservas y cuentas | - Consulta y gestión de reservas realizadas. - Modificación y cancelación de reservas. - Creación de una cuenta de usuario para guardar información personal y preferencias de viaje. - Gestión de información personal y preferencias de viaje. - Historial de reservas. |
| Funcionalidades adicionales | - Integración con mapas para visualizar la ubicación de hoteles y puntos de interés. - Recomendaciones personalizadas de vuelos, hoteles y actividades según las preferencias del usuario. - Programa de fidelización para recompensar a los usuarios frecuentes. - Opción de chat en vivo para asistencia al cliente. - Reseñas y valoraciones de usuarios sobre hoteles y actividades. |
|  | - Integración con plataformas de pago como PayPal o Stripe. - Posibilidad de reservar actividades turísticas, como tours, entradas a museos, etc. - Seguro de viaje. - Alquiler de coches. - Transferencias al aeropuerto. |

SEGUNDO PROYECTO FacturaYa

|  |  |
| --- | --- |
| Descripción | Detalles |
| Funcionalidades básicas de facturación | - Generación de facturas electrónicas con los campos requeridos por la DIAN (Dian.gov.co). - Validación de datos fiscales del cliente y emisor. - Cálculo de impuestos (IVA, retefuente, etc.) según la configuración del país. - Posibilidad de adjuntar documentos a las facturas. - Envío de facturas electrónicas a la DIAN y al cliente. - Consulta de facturas electrónicas emitidas y recibidas. |
| Gestión de clientes y productos | - Creación y edición de perfiles de clientes. - Registro de productos con sus características y precios. - Categorización de productos para facilitar la búsqueda. - Búsqueda de productos por nombre, código, categoría, etc. |
| Informes y estadísticas | - Generación de informes de ventas por período, cliente, producto, etc. - Visualización de estadísticas de ventas en gráficos y tablas. - Exportación de informes a Excel o PDF. |
|  | - Integración con pasarelas de pago para pagos en línea. - Gestión de inventario para control de stock. - Envío de recordatorios de pago a los clientes. - Módulo de CRM para gestionar las relaciones con los clientes. |

TERCER PROYECTO DoctorYa

|  |  |
| --- | --- |
| Descripción | Detalles |
| Funcionalidades básicas de gestión de citas | - Agendamiento de citas en línea por parte de pacientes y personal médico. - Visualización del calendario de citas por médico y especialidad. - Confirmación de citas por correo electrónico y SMS. - Recordatorios de citas por correo electrónico y SMS. - Registro de asistencia a citas. - Gestión de listas de espera. |
| Gestión de pacientes y personal médico | - Creación y edición de perfiles de pacientes. - Registro de información médica de pacientes. - Creación y edición de perfiles de personal médico. - Asignación de especialidades y horarios de atención a personal médico. - Gestión de permisos y accesos para personal médico. |
| Funcionalidades adicionales | - Integración con sistemas de historia clínica electrónica (HCE). - Facturación electrónica de citas y servicios médicos. - Gestión de pagos en línea. - Módulo de chat en vivo para asistencia al paciente. - Encuestas de satisfacción para pacientes. - Informes y estadísticas sobre citas, pacientes y personal médico. |
|  | - Integración con calendarios personales de pacientes y personal médico. - Envío de recetas médicas electrónicas. - Gestión de autorizaciones de seguros. - Módulo de telemedicina para citas virtuales. - Sistema de mensajería interna para comunicación entre personal médico. |

*CUARTO PROYECTO: COMPRAYA*

|  |  |
| --- | --- |
| Descripción | Detalles |
| Funcionalidades básicas productos | Catálogo de productos completo con información detallada (características, precios, imágenes, disponibilidad, etc.).  Posibilidad de crear, editar y eliminar productos.  Categorización y filtrado de productos para una fácil navegación.  Búsqueda avanzada por palabras clave, características y atributos.  Gestión de inventario en tiempo real con alertas de stock bajo.  Integración con proveedores para actualizaciones de stock y precios automáticas. |
| Funcionalidades básicas de ventas | Punto de venta (TPV) con interfaz intuitiva para realizar ventas rápidamente.  Procesamiento de pagos electrónicos y tradicionales.  Gestión de descuentos y promociones.  Emisión de facturas y tickets.  Registro de historial de ventas para análisis y seguimiento.  Integración con sistemas de contabilidad para una gestión financiera completa. |

*QUINTO PROYECTO: CONCIERTOSYA*

|  |  |
| --- | --- |
| Descripción | Detalles |
| Funcionalidades básicas de eventos | Creación de eventos: El sistema debe permitir la creación de nuevos eventos con información detallada como nombre, fecha, hora, lugar, artistas, precio de las entradas, etc.  Modificación de eventos: El sistema debe permitir la modificación de la información de los eventos creados.  Eliminación de eventos: El sistema debe permitir la eliminación de eventos que ya no estén disponibles para la venta.  Publicación de eventos: El sistema debe permitir la publicación de eventos en la plataforma para que sean visibles para los usuarios.  Consulta de eventos: El sistema debe permitir a los usuarios consultar la lista de eventos disponibles, filtrando por fecha, lugar, artista, género, etc.  Visualización de detalles de eventos: El sistema debe permitir a los usuarios visualizar la información detallada de cada evento, como nombre, fecha, hora, lugar, artistas, precio de las entradas, descripción, etc. |
| Funcionalidades básicas de entradas | Definición de tipos de entradas: El sistema debe permitir definir diferentes tipos de entradas con diferentes precios y características (VIP, general, etc.).  Creación de inventarios de entradas: El sistema debe permitir crear inventarios de entradas para cada evento, especificando la cantidad de entradas disponibles para cada tipo de entrada.  Venta de entradas: El sistema debe permitir la venta de entradas a los usuarios, registrando la información de la compra (cliente, evento, tipo de entrada, cantidad, precio total, etc.).  Selección de asientos: El sistema debe permitir a los usuarios seleccionar asientos en el lugar del evento (en caso de que aplique).  Reservación de entradas: El sistema debe permitir a los usuarios reservar entradas sin necesidad de realizar el pago inmediato.  Pago de entradas: El sistema debe permitir a los usuarios realizar el pago de las entradas reservadas a través de diferentes métodos de pago (tarjeta de crédito, débito, PayPal, etc.). |

*(Fuente: Propia, Jorge Aguirre. 2024)*

**TEMA 15: CONFORMACION DE EQUIPOS Y ROLES**

Según la cantidad de estudiantes se hará la actividad de conformación de un equipo de desarrollo bajo la metodología ágil del SCRUM, donde cada estudiante asumirá un rol de la siguiente forma:

* Rol de Scrum Master.
* Rol de Product Owner
* Rol de Líder Técnico.
* Rol de desarrollador.
* Rol de desarrollador de pruebas.

Cada estudiante deberá apropiarse de las funciones de cada rol según la temática vista en la anterior unidad.

**TEMA 16: CREACION DE ORGANIZACIÓN Y PROYECTO CON HERRAMIENTA DEVOPS.**

Cada equipo de trabajo podrá acceder a la plataforma Azure DevOps para hacer la creación de la organización y del proyecto.

**TEMA 17: CREACION DE EPICAS E HISTORIAS DE USUARIO CON PLATAFORMA DEVOPS.**

Posterior a la realización del BrainStorming el estudiante deberá definir las épicas del proyecto según los objetivos de desarrollo que se quieran construir, además hacer la creación del sprint en Azure DevOps.

**TEMA 18: PRACTICA DE REALIZACION DE PLANNING.**

Con base al pre proyecto y a la realización de un BrainStorming con los StakeHolders, Los estudiantes al tener los roles definidos dentro del equipo según el esquema de SCRUM realizarán el Planning en donde harán la creación de cada historia de usuario y con la respectiva asignación a un desarrollador, como también se validará una descripción adecuada, los criterios de aceptación, y la estimación de esfuerzo; finalmente la persona asignada creara las respectivas tareas en la historia de usuario para cumplir con la misma.

**TEMA 19: PRACTICA DE REALIZACION DE DAILY.**

De acuerdo con el trabajo que vayan realizando los estudiantes con las historias de usuario asignadas, harán una reunión de Daily cada día para ir evaluando los avances además de que actualizaran progresivamente los criterios de aceptación y las tareas de las historias de usuario.

**TEMA 20: PRACTICA DE REALIZACION DE REVIEW.**

Al finalizar los días de sprint, se realizará la Review para validar el progreso de las historias de usuario y la demostración del trabajo realizado en el proyecto, además del cumplimiento de cada criterio de aceptación de las diferentes historias de usuario.

**TEMA 21: PRACTICA DE REALIZACION DE RETROSPECTIVE.**

En última instancia se realizará la Retrospective donde básicamente se harán las reflexiones sobre el trabajo hecho durante el sprint y se establecerá las ideas para la mejora continua del proyecto.

*(Fuente: Propia, Jorge Aguirre. 2024)*

**CIERRE DE LA UNIDAD**

***INDICACIONES:***

*Una vez terminado el desarrollo del contenido, procure formar una visión sintética, integradora e incluso crítica de la unidad, a través de estrategias como: resumen, mapa conceptual, preguntas o conclusiones generales de los temas presentados en el escrito.*

*Proponga material o lecturas complementarias si lo considera necesario.*

Concluyendo las metodologías ágiles son un conjunto de enfoques para el desarrollo de software que se basan en la colaboración, la adaptación y la entrega incremental de valor. En contraste con los enfoques tradicionales de desarrollo de software, que suelen ser más estructurados y lineales, las metodologías ágiles valoran la flexibilidad y la capacidad de responder a los cambios.

Lo que aportan las metodologías agiles a la industria del software:

* Enfoque en el cliente: Las metodologías ágiles se centran en satisfacer las necesidades del cliente y brindarle valor lo antes posible. Esto se logra a través de una comunicación constante con el cliente y la entrega frecuente de software funcional.
* Ciclos de desarrollo cortos: Las metodologías ágiles dividen el desarrollo de software en ciclos cortos, que suelen durar de unas pocas semanas a un par de meses. Esto permite que el equipo reciba comentarios del cliente con mayor frecuencia y realice ajustes en el producto según sea necesario.
* Trabajo en equipo colaborativo: Las metodologías ágiles enfatizan la importancia del trabajo en equipo y la colaboración entre todos los miembros del equipo, incluidos los desarrolladores, los probadores, los clientes y las partes interesadas.
* Entrega incremental de valor: Las metodologías ágiles se enfocan en entregar software funcional al cliente de manera incremental, en lugar de esperar hasta que se complete todo el proyecto. Esto permite que el cliente comience a usar el software y proporcione comentarios lo antes posible.
* Adaptación al cambio: Las metodologías ágiles reconocen que los requisitos y las necesidades del cliente pueden cambiar a lo largo del proyecto. Por lo tanto, están diseñadas para ser flexibles y adaptarse a estos cambios.

*(Fuente: Atlassian. 2024. Disponible en:* [*https://www.atlassian.com/es/agile/scrum#:~:text=Scrum%20es%20un%20marco%20de,de%20valores%2C%20principios%20y%20pr%C3%A1cticas*](https://www.atlassian.com/es/agile/scrum#:~:text=Scrum%20es%20un%20marco%20de,de%20valores%2C%20principios%20y%20pr%C3%A1cticas)*.)*

**REFERENCIAS BIBLIOGRÁFICAS**

***INDICACIONES:***

*Registre la documentación (NORMAS APA O VANCOUVER) consultada para desarrollar el contenido.*

*Utilice bibliografía con una antigüedad no superior a los cinco años.*