



UNAM



RED·TIC

Red de Responsables TIC

U N A M



Guía para la Evaluación y Selección de Soluciones de Software

Contenido

OBJETIVO	3
ALCANCE	3
INTRODUCCIÓN	4
PROCESO PARA LA IDENTIFICACIÓN, EVALUACIÓN Y SELECCIÓN DE APLICACIONES DE SOFTWARE	4
A) <i>Análisis de las necesidades</i>	4
B) <i>Identificación de alternativas</i>	6
C) <i>Evaluación de alternativas</i>	8
ALTERNATIVAS PARA LA SELECCIÓN DE APLICACIONES DE SOFTWARE	8
<i>Primera alternativa: Adopción de algún software desarrollado a la medida ya existente en otra área universitaria u organización con un proceso similar</i>	8
<i>Segunda alternativa: Software como servicio (Software as a Service SaaS) externo</i>	11
<i>Tercera alternativa: Adquisición de soluciones de pago o de software libre</i>	12
<i>Cuarta alternativa: Fortalecer o migrar de versión o tecnología algún sistema desarrollado a la medida ya existente en el área universitaria</i>	14
<i>Quinta alternativa: Desarrollo de software a la medida</i>	16
GLOSARIO	18
REFERENCIAS	19
CRÉDITOS	20

OBJETIVO

Las recomendaciones contenidas en este documento tienen como propósito brindar una guía de ayuda para los procesos de análisis y la toma de decisiones en relación con la evaluación y selección de software, estableciendo un proceso consistente que permita incorporar aplicaciones de software confiables, seguras y funcionales que apoyen los procesos estratégicos, tácticos y operativos dentro de las entidades y dependencias de la UNAM, eliminando juicios y criterios subjetivos en el proceso de evaluación y selección de alternativas.

La importancia de este documento radica en tomar decisiones informadas, oportunas y justificadas de acuerdo con la naturaleza y escenario de cada proyecto a evaluar.

ALCANCE

Las recomendaciones de esta guía son de uso general para todas las áreas universitarias que emprendan un proyecto para incorporar software en cumplimiento de las funciones y actividades sustantivas y que deseen aplicar prácticas formales al proceso de selección de los productos disponibles como pueden ser los que existen dentro de la Universidad, o suscribirse a uno como servicio (SaaS) o bien aquel software de tipo comercial o libre.

Este documento describe las actividades que los responsables administrativos y técnicos deberán realizar conjuntamente para determinar la opción de mayor valor para el área universitaria en la adquisición de un software, mediante el estudio y evaluación de alternativas, de acuerdo con el establecimiento y valoración de los criterios que se empleen durante este proceso de evaluación y selección.

Introducción

Es relevante mencionar que la decisión de un proceso de incorporación de software, no solamente involucra aspectos técnicos, sino que además depende en gran medida de los procesos internos y la estructura orgánica del área involucrada, de la madurez y competencias en materia de Tecnologías de la Información y Comunicación (TIC) del personal de la propia área, en concordancia con la claridad y alineación del objetivo del proyecto o servicio a resolver con respecto al valor mismo de la solución que se está evaluando.

Para fines de una optimización institucional de recursos y costos, es importante analizar si la incorporación de una aplicación de software dará solución exclusivamente a una problemática detectada en un área universitaria o si ésta se pudiera constituir como una herramienta de utilidad en toda la Universidad, o bien, si la necesidad de incorporar el software están asociadas a oportunidades o mejoras en los procesos internos relativos a: definición de actividades, mejoras académicas y/o administrativas, descripción de roles, asignación de responsabilidades, comunicación organizacional y/o capacitación; de identificarse que alguno de estos últimos es el caso, se recomienda evaluar el proyecto considerando a las opciones tecnológicas como un medio para obtener dichas ventajas y mejoras al área universitaria y no, como un fin por sí mismas.

Proceso para la identificación, evaluación y selección de aplicaciones de software

Es fundamental realizar un análisis acerca de las alternativas viables, determinando cuál será la opción más conveniente para el área universitaria, tomando en cuenta los recursos humanos, financieros, tecnológicos y de tiempo disponibles para realizar con éxito el proyecto.

Por lo anterior, se recomienda llevar a cabo las siguientes actividades para tener la suficiente claridad en la definición de las necesidades y una adecuada selección de alternativas de solución.

A) Análisis de las necesidades

Con el fin de comprender las necesidades que dan origen a la incorporación de una nueva aplicación de software, se sugiere:

1. Elaborar la **matriz de involucrados/interesados** que permita identificarlos y gestionarlos de manera adecuada durante el desarrollo del proyecto o simplemente para mantener una referencia de consulta en caso de que se requiera (**Anexo 1**).

Esto es particularmente útil, como ejemplo no limitativo, para cuando el alcance y la complejidad del proyecto de incorporación del software tiene fronteras más allá de la propia área universitaria, lo que hace que sea un gran número de personas involucradas a las que se deban gestionar en términos de establecer una comunicación efectiva con todos ellos.

Además, ayudará a disminuir los riesgos referentes a fallas en esta gestión que pudieran generar una mala ejecución del proyecto o en el peor de los casos hasta en el fracaso del mismo.

2. Definir los aspectos principales de la solución que se requiere, con la finalidad de contar con la visión/misión de la misma, para que sean el referente del alcance y poder cumplir los objetivos del área universitaria. Para recopilar y documentar los puntos principales de la solución se presenta una **propuesta para la definición del proyecto (Anexo 2)**. Para realizar la definición se tienen que analizar los siguientes aspectos que permitan comprender la situación actual del área universitaria:

- a) la problemática actual,
- b) los procesos, áreas y actores involucrados,
- c) los objetivos que se pretenden lograr,
- d) los beneficios e impactos esperados, así como
- e) las restricciones de tiempo y presupuesto.

Este análisis se puede llevar a cabo a través de la aplicación de entrevistas o hacer reuniones de trabajo con todos los involucrados (el área usuaria, directivos, técnicos u otras áreas que se vean impactadas de manera directa o indirecta).

3. Documentar una **matriz de requerimientos (Anexo 3)** que describa a detalle las características requeridas para el nuevo sistema de información o aplicación, indicando si estas características son obligatorias o deseables, incluyendo al menos lo siguiente:

- a) Identificador.
- b) Requerimientos funcionales o no funcionales.

Se han de considerar los criterios de calidad requeridos en el(los) producto(s) de software (también denominados requerimientos no funcionales) tomando en cuenta que la prioridad o enfoque entre estos criterios puede cambiar significativamente de acuerdo con los fines del sistema o aplicación. Para determinar los criterios que ha de tener la solución, se presenta una **guía para la especificación de requerimientos no funcionales (Anexo 4)**:

- | | |
|---------------------|---|
| 1.1. Seguridad | 1.6. Interoperabilidad |
| 1.2. Eficiencia | 1.7. Confiabilidad |
| 1.3. Compatibilidad | 1.8. Portabilidad |
| 1.4. Usabilidad | 1.9. Restricciones de diseño y construcción |
| 1.5. Mantenimiento | 1.10 Documentación y capacitación necesaria, entre otros. |

- c) Propósito.
- d) Clasificación
- e) Área(s) solicitante(s).

4. Validar la matriz de requerimientos mencionada en el punto anterior con los involucrados del proyecto, servicio o proceso que puedan confirmar la necesidad, impacto y valor de cada rubro, las características que son obligatorias, así como aquellas características que pueden

clasificarse como opcionales o deseables. Este es un punto objetivo debido a que en la medida en que se cumpla con los puntos de esta relación, se podrán cubrir los objetivos organizacionales. Esta verificación de requerimientos es una parte relevante del proceso para eliminar la subjetividad al mismo.

5. Realizar la ponderación de la matriz de requerimientos para determinar la prioridad de cada uno de ellos (**Anexo 5**).

B) Identificación de alternativas

Una vez que se realizó el análisis de las características requeridas por la nueva aplicación y se validó que éstas responden a los requerimientos y expectativas de los diferentes involucrados en el proyecto, es recomendable que el área universitaria realice un ejercicio de evaluación completo de las diversas opciones para satisfacer dichas necesidades.

Es conveniente considerar las siguientes opciones:

- a) La existencia de sistemas de información similares desarrollados a la medida en alguna otra área universitaria que puedan ser compartidos o reutilizados. Tomar en cuenta la experiencia que otras áreas universitarias tienen en el tema. Para ello se recomienda revisar las posibles alternativas existentes dentro del inventario de software¹ y desarrollos actuales dentro de la misma Universidad, el cuál proporcionará una primera posibilidad de solución a las necesidades.
- b) La posibilidad de suscribirse por un tiempo definido a un servicio de software para usarlo de manera remota ya sea conectándose a los servidores del proveedor o a la nube de este último, usualmente con un pago mensual o anual.
- c) La existencia de herramientas (comerciales o libres) que cumplan con las características requeridas, que puedan ser fácilmente adaptadas y que requieran una curva de aprendizaje baja.
- d) La conveniencia de fortalecer el sistema desarrollado a la medida que se encuentra actualmente en uso evaluando la alternativa de modificar o crear algunos módulos de un sistema. Privilegiar en este fortalecimiento la visión de uso institucional por otras entidades o áreas universitarias, multientidad.
- e) La posibilidad de realizar una migración por medio de herramientas automáticas que conserven la funcionalidad actual de un software desarrollado a la medida hacia una nueva versión o tecnologías usadas. Esto puede ser necesario cuando las tecnologías que fueron usadas para el desarrollo ya se encuentran obsoletas o sin soporte del proveedor, algunos ejemplos son:

¹ El inventario de software institucional en la Universidad puede consultarse en <https://informaciontic.unam.mx/>

- ◆ Actualizar el código fuente de PHP 5.4 a PHP 8.
 - ◆ Migrar la aplicación desarrollada a la medida de Visual Basic .Net a la tecnología .Net 5.
- f) La opción de realizar un desarrollo completo a la medida, ya sea de manera interna en el área universitaria o subcontratando a un proveedor externo. En este caso debe estimarse el esfuerzo, tiempos, implicaciones y los riesgos del nuevo sistema de información o aplicación, en términos del proceso de desarrollo, de implantación, de mantenimiento, por mencionar algunas dimensiones. Es imprescindible considerar la visión de uso institucional por otras entidades o áreas universitarias, multientidad en el diseño del nuevo sistema.

Derivado de este análisis se resumen las siguientes cinco alternativas:

- I. Adopción de algún software ya existente en una entidad universitaria u organización con un proceso similar.
- II. Software como un servicio (Software as a Service, SaaS).
- III. Adquisición de soluciones comerciales o de software libre.
- IV. Fortalecimiento o migración de versión o tecnología de algún sistema hecho a la medida ya existente en la entidad o dependencia universitaria.
- V. Desarrollo de software a la medida.

Las tres primeras involucran obtener un producto de software existente ya sea comercial, libre o de otro tipo de licencia o bien para suscribirse y hacer uso de un software. Por lo tanto, pueden seguir un procedimiento específico de adquisición de software; las siguientes dos opciones implican realizar un esfuerzo de desarrollo a la medida o una actualización o migración de versión y/o tecnología, por lo que en caso de elegir la alternativa IV o V puede seguirse un procedimiento específico distinto al sugerido en esta guía, aunque pueden apoyarse en el proceso general de adquisición de software cuando sea necesario.

Es importante señalar que ninguna de las opciones es mejor que otra por sí misma, sino que la elección de una alternativa depende en gran medida de las necesidades del área universitaria.

Cada una de las alternativas se explica más ampliamente en la sección **Alternativas para la selección de aplicaciones de software**.

C) Evaluación de alternativas

1. Identificar aquellas alternativas que cumplan con la mayor cantidad de características definidas como indispensables, obligatorias, así como la mayor parte de las deseables indicadas en la **matriz de requerimientos** al realizar un ejercicio de valoración (**Anexo 6**).
2. Realizar un **análisis costo-beneficio (Anexo 7)** de las alternativas, tomando en cuenta los recursos humanos, financieros y de tiempo disponibles para realizar el proyecto.

3. Concentrar la información y presentar un resumen de los resultados obtenidos de la evaluación que apoye la toma de decisiones de los directivos (**Anexo 8**).

Alternativas para la selección de aplicaciones de software

La creencia generalizada dicta que toda organización es única y que es improbable que encuentre una solución existente que se ubique implantada en otra organización que le pudiera resultar de utilidad y cubrir sus necesidades. La experiencia práctica en procesos formales relacionada con procesos de adquisición de software indica que realizar un estudio completo propicia la obtención de resultados favorables; no únicamente en términos de costo y tiempo, sino en diversos criterios de calidad de software, empezando por la propia funcionalidad que ofrece un producto determinado.

Para realizar la evaluación es necesario contar con la **matriz de los requerimientos** indicada en el cuarto y quinto punto del apartado *Proceso para la identificación, evaluación y selección de aplicaciones de software*.

Primera alternativa: Adopción de algún software desarrollado a la medida ya existente en otra área universitaria u organización con un proceso similar

Algunas organizaciones crean una barrera para aceptar esta alternativa dado que tienen la creencia de que son únicas y que es improbable que pueda utilizarse una solución que está en otra organización. También es importante mencionar que dentro de la Universidad existen procedimientos que deben ser llevados de manera estandarizada en diversas dependencias y debido a ello se sugiere tomar en cuenta esta alternativa para poder tomar una correcta decisión, en este sentido se ha de evaluar la posibilidad de adoptar algún software existente en algún área afín en la propia Universidad o en otra organización donde se encuentren similitudes en la funcionalidad del producto. Existe una gran cantidad de procesos que son transversales (de manera común en diversas entidades) o, verticales que se asocian a una función de la organización, donde por lo anterior, algunas veces “tocando las puertas adecuadas” y haciendo las preguntas correctas se pueden encontrar estas importantes similitudes y oportunidades para adquirir la solución esperada y requerida.

La ventaja de adquirir una solución existente usualmente es muy significativa en cuestión de reducir los tiempos de implementación para tener la solución operando, en costos de adquisición (en contraste directo contra el desarrollo a la medida y en comparación con la adquisición de soluciones comerciales, en donde se toma en cuenta los costos de desarrollo, de recursos de hardware y humanos), en el aprovechamiento de los recursos de la Universidad, disminución de riesgos al cubrir la mayor cantidad de requerimientos, rapidez en el tiempo de capacitación y aprendizaje de utilización de los usuarios, así como en la evolución del software.

Para el análisis de esta alternativa se sugieren los siguientes pasos:

1. Revisar el inventario de software² y desarrollos actuales dentro de la misma Universidad para identificar las posibles alternativas ya existentes que proporcionan una primera posibilidad de solución a las necesidades. Un paso importante que se recomienda realizar, si es el caso, es identificar el área universitaria que de manera lógica o por su naturaleza solicita ciertos requerimientos o procesos que dieron origen a la necesidad a cubrir, para verificar si podría proporcionar una solución institucional (generalizada para adaptarla a cada entidad o dependencia).
2. Obtener o elaborar la relación de la funcionalidad que ofrece la solución que se encuentra implementada en otra área u organización para poder hacer una evaluación ponderada que permita eliminar subjetividades.
3. Es relevante evaluar:
 - a) La flexibilidad de la solución.
 - b) La portabilidad y madurez de la misma en operación.
 - c) Nivel de similitud de los procesos que maneja la solución con los procesos que desea automatizar la dependencia adoptante.
 - d) Las necesidades de infraestructura, entorno y mantenimiento requerido, para determinar si el área universitaria que desea hacer uso de la solución, puede adquirir la plataforma e infraestructura tecnológica en que se recomienda opere, y en su caso, determinar también si cuenta con el personal requerido para la operación y/o mantenimiento de la solución.

Las soluciones de software que se pueden encontrar dentro de la Universidad suelen ser adquiridas a través de alguno de los dos modelos de adopción que a continuación se mencionan:

- ◆ **Adoptar un sistema transferible.** En este modelo, el área universitaria que adopta al software suele recibir una copia del código fuente y de la documentación que complementa a la solución. Esto permite adaptar o modificar el software (respetando los derechos morales) a requerimientos que podrían ser más específicos. Se sugiere analizar qué tanto cubre la solución transferible para tomar la decisión de adoptarla tal cual, o adaptarla en ciertas funcionalidades o, si es viable, complementar con algunos módulos de otra solución o bien modificar el proceso del área universitaria que está evaluando las soluciones para adoptar la solución completa. Sin embargo, al mismo tiempo, el área universitaria adoptante adquiere la responsabilidad de contar con la infraestructura y los recursos necesarios para operar y mantener el software por sí misma.
- ◆ **Usar un sistema operado por un área universitaria.** En este otro modelo, existe un área universitaria que se encarga de operar, actualizar y mantener un desarrollo continuo

² El inventario de software institucional en la Universidad puede consultarse en <https://informaciontic.unam.mx/>

del software, además de ofrecer a las otras áreas universitarias interesadas la posibilidad de usarlo a través de la suscripción de usuarios, teniendo siempre en cuenta la importancia de mantener privados los datos de cada una de las áreas que están suscritas.

En este segundo modelo, el área universitaria adoptante tiene la ventaja de no requerir infraestructura, personal especializado ni invertir tiempo adicional para poner en funcionamiento la solución, gracias a que el uso del software puede llegar a estar disponible en pocas horas y en algunos casos hasta en minutos, una vez que se formaliza la adopción. A pesar de estas ventajas, es importante considerar que la flexibilidad para adaptar y modificar una solución de este tipo a las necesidades de una sola área puede ser un poco más limitada, sin embargo, esto no quiere decir que no sea posible realizar estas adecuaciones por el área encargada de mantener el software.

Es importante no confundir este modelo de adopción con la segunda alternativa de selección de software que se menciona en esta guía, a saber, Software como Servicio (SaaS) externo, en donde básicamente la diferencia es que un tercero ajeno a la Universidad ofrece el servicio de usar el software en la mayoría de los casos por un pago obligatorio.

4. Tomar en cuenta como parte de la valoración, la recopilación de información y entrevistas con la organización que emplea el producto que se desea adquirir, para recopilar testimonios de las áreas usuarias, resultados logrados, documentación existente, así como indicadores y métricas, en caso de existir. Entendiendo las diferencias propias de procesos, culturas y alguna funcionalidad específica, usualmente esta información es muy valiosa para la toma de decisión final conforme a esta alternativa.
5. Acordar un periodo de prueba³ de uso del producto (30, 60 o 90 días comúnmente) en que se pueda hacer uso del producto en un ambiente controlado y hacer una prueba real para determinar en qué medida el producto cumple con la especificación inicial de requerimientos. En caso de ser factible, y más en el caso entre entidades y dependencias universitarias o con instituciones de educación superior.
6. Asimismo, se vuelve muy importante negociar en un primer acercamiento con el área universitaria o la organización que cuente con los derechos patrimoniales e industriales del producto, si es factible realizar la transferencia de uso del producto bajo la modalidad legal correspondiente o de compartir el servicio. El proceso pudiese incluir alguna formalización para el licenciamiento o autorización de uso, cesión de derechos u otra figura, ya sea que se determine o no, algún pago o apoyo de recuperación.

Al realizar los pasos anteriores se obtendrán los siguientes resultados:

³ Este periodo de prueba podría ser aplicable a otras alternativas de adquisición, sin embargo, es recomendable solicitarlo, aunque es menos probable que se acepte.

- El grado de compatibilidad del proceso a atender con el de la solución existente en función de los resultados de cumplimiento de la matriz de requerimientos.
- Los recursos necesarios para materializar la adopción (tecnológicos, infraestructura, humanos, capacitación de usuarios, etc.).
- Las acciones y tareas adicionales una vez que la adopción del software sea concretada como pueden ser: la reingeniería de procesos, adaptación de módulos o incluso la creación de nuevos módulos completos.
- Estimación del tiempo para poner en funcionamiento el software adoptado.

El análisis de los resultados determinará de una manera más precisa la decisión final de esta alternativa considerando el cruce con las necesidades de la dependencia que desea realizar la adopción.

Finalmente, es importante remarcar la relevancia de que la primera alternativa de adquisición que se presenta en esta guía sea la adopción de un software interno de la UNAM; esto es porque son muy altas las probabilidades de encontrar soluciones de software ya existentes en toda la extensión de la Universidad que apoyan en las actividades diarias y que resuelven necesidades que son encontradas en otras áreas universitarias que son análogas a aquellas que dieron origen a la solución de software. Para conocer y analizar las diferentes opciones que ofrece la Universidad es recomendable que visite el inventario de software de la UNAM en la dirección [<https://informaciontic.unam.mx>].

Segunda alternativa: Software como servicio (Software as a Service SaaS) externo

El proveedor proporciona el uso de un Software por medio del pago de una suscripción durante un periodo de tiempo, por ejemplo: por medio de un pago (mensual o anual) o sin pago, mientras que sea necesario el servicio. El software es usado a través de una conexión en internet a los servidores o nube del proveedor.

El suscriptor no gestiona ni controla la infraestructura fundamental del software tales como la red, servidores, sistemas operativos, almacenamiento o incluso capacidades de aplicaciones individuales, podría existir la excepción de aplicar ajustes específicos de configuración del usuario, pero limitados en las aplicaciones.

El proceso de adquisición de esta alternativa es similar al utilizado para el software comercial (tercera alternativa de la presente sección donde se detalla el proceso a seguir):

1. Definir por cuánto tiempo se estima que se requiere la solución. Considerar las normatividades que apliquen, por ejemplo: las restricciones de tiempo para poder hacer el contrato del servicio.

2. Identificar y comparar las alternativas de soluciones de software.
3. Evaluar las ofertas y los proveedores, considerando que dichas propuestas estén alineadas con la normatividad UNAM y marcos jurídicos aplicables, como son la *Normatividad en Materia de Adquisiciones, Arrendamientos y Servicios de la UNAM*, *Ley General de Protección de Datos Personales*, *Normas Complementarias sobre Medidas de Seguridad Técnicas, Administrativas y Físicas para la Protección de Datos Personales en Posesión de la Universidad*, entre otros (**Anexo 9**).
4. Validar para comprobar que la alternativa cumpla con las necesidades del área universitaria con base en la matriz de requerimientos. Elaborar la relación de la funcionalidad que ofrecen las opciones de solución para poder hacer una evaluación ponderada y eliminando subjetividades.
5. Comprobar si existe documentación de la solución, por ejemplo manuales de usuario, en este caso no habrá guías de instalación o similares debido a que no se instalará en máquinas del comprador, sino que se usará el servicio disponible en la nube o en los servidores del proveedor.
6. Revisar el contrato, y en caso de que el contenido sea desfavorable, se sugiere tratar de negociar el contenido del mismo, aunque en el caso del Software como servicio, generalmente estas licencias de uso no son modificables, en este caso se necesitaría considerar si se acepta el contrato tal y como se encuentra o se rechaza y se busca otra alternativa de software. Debe acompañarse de las áreas jurídicas de la entidad o dependencia.
7. Para mayor referencia de este punto, se sugiere apoyarse en el estándar *IEEE 41062:2019 Recommended Practice for Software Acquisition*, como guía para realizar este proceso, el cual incluye formatos que apoyan la definición y seguimiento del proceso de adquisición (**Anexo 10**).

Tercera alternativa: Adquisición de soluciones de pago o de software libre

El proceso de adquisición de software provee una estructura de los pasos más importantes aplicados en la adquisición de cualquier elemento de software y en el uso de software libre. Se espera que las actividades utilizadas en este proceso den como resultado la entrega de productos de alta calidad y bien documentados.

Para iniciar un proceso de adquisición de soluciones de pago o de software libre, se recomienda:

1. Identificar publicaciones de terceros independientes calificados (como casas consultoras, estadistas, revistas especializadas, por mencionar algunas), que hagan referencias de las características positivas y negativas del producto, el servicio, el respaldo del proveedor, el costo/beneficio de valor y propiedad del producto, como algunos puntos a considerar.

2. Si se decide adquirir una solución deben evaluarse diversas ofertas de proveedores y seleccionar al más adecuado, considerando la normatividad aplicable (**Anexo 9**) y al menos los siguientes criterios:
 - a) Cumplimiento de los requerimientos solicitados (obligatorios y deseables)
 - b) Experiencia del proveedor en el mercado
 - c) Tipo de licenciamiento (por procesador, por usuarios designados, perpetua, temporal, creative commons, entre otros)
 - d) Soporte (tipo, vigencia, tiempos de respuesta y actualizaciones al software) y garantía (duración y aplicación)
 - e) Capacitación
 - f) Vigencia tecnológica de la solución
 - g) Uso de estándares abiertos
 - h) Plataforma necesaria para su funcionamiento
 - i) Forma de entrega del software (medios ópticos o descarga de la red)
 - j) Documentación de la solución. Solicitar u obtener (en el caso de software libre) la documentación necesaria para operar y mantener el sistema, incluyendo como mínimo el Manual de Usuario y el Manual Técnico para la operación.
 - k) Precio
3. Asegurar que el contrato propuesto o licencia de uso, precise el alcance, tiempo, entregables, criterios de aceptación, tipo de licenciamiento, tipo, forma y cobertura del soporte, duración y procedimiento en que aplica la garantía y compromisos de ambas partes, así como dar seguimiento puntual a éste.
4. Realizar pruebas de validación para asegurar que la solución entregada cumpla con todos los requerimientos acordados en el contrato y que dicha validación quede especificada dentro de los criterios de aceptación.
5. Para mayor referencia de este punto, se sugiere apoyarse en el estándar *IEEE 41062:2019 Recommended Practice for Software Acquisition* (**Anexo 10**), como guía para realizar este proceso, el cual cuenta con formatos que apoyan la definición y seguimiento del proceso de adquisición. El estándar recomienda realizar las siguientes fases:



El proceso de adquisición de software debe adaptarse o simplificarse en la mayor medida posible tomando el riesgo en consideración. El proceso debe ser coherente con la madurez técnica, los requisitos validados, y el nivel esperado de esfuerzo y financiación.

Cuarta alternativa: Fortalecer o migrar de versión o tecnología algún sistema desarrollado a la medida ya existente en el área universitaria

El proceso dinámico inherente a las Tecnologías de Información y a las plataformas tecnológicas que soportan a los sistemas, en conjunto con el ciclo de vida propio de un aplicativo de software, crean la necesidad con el tiempo, de abordar un proceso de adquisición de un producto de software. En el caso del software comercial, esta exigencia podría solucionarse de manera más inmediata mediante la adquisición de una nueva versión del producto, o bien la actualización de algún componente de software o de hardware.

En este caso, la evaluación se realiza para conocer si algún sistema existente en el área universitaria, el cual fue realizado a la medida y que actualmente se encuentra en operación (usualmente con varios años en operación), es factible fortalecerlo o migrarlo a una nueva versión o tecnología. En este caso es importante partir de si el cambio debe realizarse debido a:

a) *Un cambio en las reglas de los procesos de operación, negocio o normatividad, para ello se sugiere tomar en cuenta las siguientes recomendaciones:*

1. Determinar el impacto de los cambios para conocer si son ajustes o modificaciones que pueden resolverse mediante un proyecto de actualización a nivel de programación de

ciertos componentes o módulos del sistema, que pudieran dar origen a una nueva versión del sistema a ser probada y liberada (por ejemplo liberar la versión 2.0 de un sistema); o si se trata de un cambio radical o de muy alto impacto en el proceso o metodología de operación que orienten la solución a alguna otra alternativa para sustituir el sistema actual que se venía operando.

En el caso de un sistema operado por un área universitaria, al que tienen acceso otras áreas de la Universidad, los cambios podrían ser por:

- a. Nuevos requerimientos solicitados por los usuarios
 - b. Cambios de versiones de la plataforma tecnológica derivado de la alta demanda de recursos por la cantidad de usuarios suscritos.
2. En cualquier escenario, deben contemplarse los criterios de calidad acordados en el análisis para determinar por ejemplo la funcionalidad, flexibilidad y facilidad de mantenimiento de tomar esta opción. La dimensión de costo asociada al desarrollo e impacto de esta nueva versión debe ser un elemento de peso para la decisión, la cual debe fundamentarse también en si se determina desarrollarlo de manera interna (contando con los perfiles y competencias para ello) o si se decide gestionar que el desarrollo lo realice una organización externa de esta nueva versión.

Es importante destacar que un sistema puede evolucionar a una nueva versión del mismo para responder a las necesidades de la universidad, sin implicar que se vuelva costoso o que su mantenimiento se vuelva complicado. Esto se debe resolver dentro de las consideraciones de diseño, si se decidiera actualizar un producto existente.

b) Por obsolescencia o problemas de mantenimiento de la plataforma tecnológica o de seguridad, para lo cual es importante:

3. Asesorarse y contar con un panorama general de las plataformas tecnológicas disponibles, para poder determinar si es factible realizar la migración del producto a una versión más actualizada y vigente o es necesario contemplar alguna de las otras alternativas planteadas en el documento.
4. Si la plataforma tecnológica que se está empleando está reconocida por la industria como obsoleta o no vigente, se debe evaluar si se realiza por medio de un desarrollo (ajustes al código) o validar si existe algún producto libre o comercial que permita realizar la migración del producto hacia una nueva plataforma por medio de herramientas automáticas que conserven la misma funcionalidad del sistema actual. Por ejemplo: de Visual Basic .Net a la tecnología .Net 5 (evaluando las implicaciones y costos del proceso) o si es necesario seleccionar alguna otra alternativa de adquisición y si es así realizar un proceso de documentación y cierre de la solución actual en cuanto se implante exitosamente la nueva solución. Dado el último escenario, pudiera ser necesario considerar una migración de datos de un modelo de datos a otro.

Las herramientas “automatizadas” de migración de plataformas pueden incluir cambio de aplicaciones de usuarios únicos o multi-usuarios, de cliente servidor a un modelo vista controlador, de un lenguaje estructurado a un lenguaje orientado a objetos, por mencionar algunos ejemplos. Algo similar puede encontrarse en relación con ciertas bases de datos en el mercado. Cabe mencionar que hay herramientas maduras y probadas que pueden entregar resultados bastante favorables y que requieren ajustes posteriores mínimos.

5. Es importante tratar de contemplar dentro de los planes de trabajo anuales de ser posible, la actividad de realizar actualizaciones de productos de software a nuevas versiones estables (liberadas y probadas) para evitar tener aplicaciones en versiones muy antiguas que sean muy difíciles de mantener, debido a la dinámica cambiante que se ha comentado en las plataformas tecnológicas.
6. Por seguridad contemplar además cambios o ajustes necesarios con la finalidad de disminuir riesgos como: robo de información, pérdida de privacidad, perjuicio económico, suplantación de identidad.

Quinta alternativa: Desarrollo de software a la medida

Partiendo de que se ha realizado un estudio formal de todas las opciones para la incorporación de una solución de software, el desarrollo a la medida es una alternativa cuya factibilidad y ejecución se debe determinar de acuerdo con los recursos, restricciones y expectativas del área universitaria. En comparación con las alternativas anteriores, usualmente realizar un desarrollo a la medida es el que toma una dimensión en tiempo mayor.

1. Considerar las restricciones de tiempo, costo, calidad, alcance y satisfacción de las necesidades de los involucrados en el proyecto. Para este último punto, es fundamental realizar un análisis de requerimientos detallado con dichos involucrados que incluya puntos de validación considerando desglosar a un nivel mayor de detalle, la matriz de necesidades identificadas como obligatorias y/o deseables dentro del proceso de evaluación de alternativas.
2. Para realizar un proyecto de esta naturaleza, es necesario prever la conformación de un equipo de trabajo multidisciplinario con representantes tanto de las áreas de negocio y usuarias como del área de cómputo, liderados por un responsable del proyecto con la competencia y autoridad necesaria para coordinar la realización del mismo. Si bien los proyectos para incorporar un nuevo sistema de información o aplicación requieren en gran medida de aspectos tecnológicos, no deben ser considerados como proyectos de carácter meramente técnico en donde se transfiera toda la responsabilidad al área de informática o de tecnología, es muy importante que se involucre el personal de negocio para lograr los objetivos del software.
3. Considerar en el diseño del sistema la perspectiva de que el software pueda ser compartido (transferible o como servicio interno de la Universidad) con otras áreas universitarias.

4. Se recomienda, y es deseable, que el equipo de trabajo técnico cuente con conocimientos y tenga experiencia tanto en las tecnologías de desarrollo como en la aplicación de metodologías formales para su administración y desarrollo, especialmente en sistemas de alta complejidad y en consideración de las restricciones mencionadas anteriormente.

En el caso de la **subcontratación** para el desarrollo del aplicativo de software, se sugiere contemplar los siguientes puntos:

1. Se puede buscar la subcontratación con un tercero para el desarrollo de software cuando se determine que el costo es menor o si el área universitaria no cuenta con los recursos, competencias y/o experiencia necesarios para desarrollar y/o actualizar un nuevo sistema. Para ello es indispensable considerar prácticas para la administración de subcontratos que permitan asegurar el cumplimiento de los compromisos acordados.
2. Para la gestión de un servicio de subcontratación con un tercero o de colaboración entre entidades para el desarrollo del proyecto de software, es importante establecer los requerimientos funcionales y no funcionales, el alcance, criterios de aceptación, restricciones, factores a considerar y responsabilidades que adquiere la propia área universitaria en la participación, colaboración y gestión del proyecto para la generación del producto.
3. Para la selección del proveedor en su caso, es importante considerar:
 - ◆ Su experiencia en proyectos anteriores o similares.
 - ◆ Las competencias y experiencia de su personal.
 - ◆ El reconocimiento o certificaciones deseables.
 - ◆ Los derechos patrimoniales del producto.
 - ◆ La garantía.
 - ◆ La documentación a entregar.
 - ◆ Los estándares de control de calidad que se utilizan.
 - ◆ La capacitación y transferencia del conocimiento.
4. Como práctica probada, la plataforma tecnológica a seleccionar, debe considerar la compatibilidad con las plataformas actuales que tiene la propia área universitaria, así como las competencias y experiencia del personal con diferentes ambientes, sistemas operativos, lenguajes de programación, marcos de trabajo y bases de datos, por mencionar algunos. En este tema podría solicitarse orientación de expertos en el área para partir de una opinión objetiva de acuerdo con criterios base.

Para mayor detalle en este tema, se sugiere consultar marcos de referencia y metodologías de desarrollo de software internacionales, así como otras recomendaciones de prácticas probadas de desarrollo de software aplicadas en la industria o emitidas por la Universidad.



Glosario

Para consultar los términos empleados en el presente documento diríjase al documento de **Glosario de términos TIC**: <https://www.red-tic.unam.mx/glosario>

Referencias

Bayse, Gail. (2004). ***A Security Checklist for Web Application Design***. Recuperado el 26 de mayo de 2021 de <https://www.sans.org/reading-room/whitepapers/securecode/security-checklist-web-application-design-1389>.

Estándares

Institute of Electrical and Electronics Engineers. (1998). ***Recommended Practice for Software Requirements Specifications*** (IEEE 830). Reaffirmed 9 December 2009. <https://ieeexplore-ieee-org.pbidi.unam.mx:2443/document/720574>

Institute of Electrical and Electronics Engineers. (2016). ***Recommended Practice for Software Requirements Specifications in IEEE Std 1062-2015*** (Revision of IEEE Std 1062-1993). <https://ieeexplore-ieee-org.pbidi.unam.mx:2443/document/7419835>

Institute of Electrical and Electronics Engineers. (2019). ***Software engineering — Recommended practice for software acquisition (ISO/IEC/IEEE 41062)***. <https://ieeexplore-ieee-org.pbidi.unam.mx:2443/document/8645777>

Institute of Electrical and Electronics Engineers. (2013). ***Software and systems engineering --Software testing --Part 1: Concepts and definitions*** (ISO/IEC/IEEE 29119-1). <https://ieeexplore-ieee-org.pbidi.unam.mx:2443/document/6588537>

Institute of Electrical and Electronics Engineers. (2013). ***Software and systems engineering --Software testing --Part 2: Test processes*** (ISO/IEC/IEEE 29119-2). <https://ieeexplore-ieee-org.pbidi.unam.mx:2443/document/6588543>

Institute of Electrical and Electronics Engineers. (2012). ***Standard for Configuration Management in Systems and Software Engineering***. (IEEE 828). <https://ieeexplore-ieee-org.pbidi.unam.mx:2443/document/6170935>

Institute of Electrical and Electronics Engineers. (2017). ***Systems and software engineering - Requirements for acquirers and suppliers of information for users*** (ISO/IEC/IEEE 26512). <https://ieeexplore-ieee-org.pbidi.unam.mx:2443/document/8288807>

Institute of Electrical and Electronics Engineers. (2017). ***Systems and software engineering - Vocabulary*** (ISO/IEC/IEEE 24765). <https://ieeexplore-ieee-org.pbidi.unam.mx:2443/document/8288807>

Créditos

Rector

Dr. Leonardo Lomelí Vanegas

Secretaria de Desarrollo Institucional

Dra. Diana Tamara Martínez Ruíz

Director General de Cómputo y de Tecnologías de Información y Comunicación

Dr. Héctor Benítez Pérez

Coordinación

Dra. Marcela Peñaloza Báez

Elaboración

Primera versión basada en el estándar IEEE 1062:1998

Susana Laura Corona Correa

Alberto González Guízar

Hugo Alonso Reyes Herrera

María Teresa Ventura Miranda

Segunda versión basada en el estándar IEEE 1062:2015

Susana Laura Corona Correa

Alberto González Guízar

Heidi Alejandra Pérez Vera

Hugo Alonso Reyes Herrera

María Teresa Ventura Miranda

Tercera versión basada en el estándar IEEE 41062:2019

Ricardo Arroyo Mendoza - CUAIEED

Susana Laura Corona Correa - DGTIC

Francisco Dante Benítez Victoria – FES Aragón

Jesús Esquivel Martínez – Facultad de Psicología

Margarita González Trejo - DGTIC

Alberto González Guízar - DGTIC

Gerardo León Lastra – Filmoteca UNAM

Leticia Martínez Calixto - DGTIC

Esmeralda Martínez Montes - CISAN

Hugo Alonso Reyes Herrera - DGTIC

Irene Gabriela Sánchez García - DGTIC

María de los Angeles I. Sánchez Zarazúa - DGTIC



Revisión

José Luis Chávez Sánchez - DGTIC

Revisión a la tercera versión basada en el estándar IEEE 41062:2019 (2024)

Leticia Martínez Calixto - DGTIC

María de los Angeles I. Sánchez Zarazúa - DGTIC