

Rescantando a MasterDevelop

Renato Josué Flores Pérez, 201709244

2 de enero de 2021

1. ¿Qué opinas sobre la decisión de Armando Guerra de no invertir más recursos humanos y económicos en un producto de software innovador?

- La decisión de Armando de recortar el personal para ahorrar recursos y disminuir el presupuesto es sin duda la solución más sencilla para resolver los déficits terribles en que se encuentra la empresa.
- La solución de Armando es únicamente una solución a corto plazo y de hecho, estoy convencido de que a largo plazo, Softics disminuirá su presencia en el mercado, perderá relaciones importantes con sus clientes e inclusive podría llegar a la quiebra de seguir un rumbo similar.
- Debe encontrarse un equilibrio entre el Ahorro y la Inversión en investigación, desarrollo e innovación. Esto no puede ser pasado por alto, más aún en una empresa como Softics, cuyo lema principal es la innovación y desarrollo de sistemas de información. De lo contrario, Softics se estancará y no será capaz de seguir adelante, perdiendo toda ventaja contra sus competidores y eventualmente cayendo en la obsolescencia.

2. ¿Qué conocimientos, habilidades y actitudes debe tener Luis para sortear con éxito las dificultades que enfrenta al elaborar la propuesta de rescate de MasterDevelop?

2.1. Conocimientos

2.1.1. Definición de valor para el cliente

Es fundamental que Luis posea una clara definición de lo que significa valor para cada uno de sus clientes, tanto actuales como potenciales. Esto es de vital importancia para poder identificar exitosamente que funcionalidades y tareas deben ser priorizadas para su implementación y que funcionalidades pueden ser retrasadas o incluso deprecadas y removidas del proyecto si no aportan valor al cliente.

2.1.2. Noción de calidad

Una vez Luis conozca la definición de valor según el cliente, debe tener una firme noción de calidad y de cómo se espera confirmar y probar dicha calidad, para poder asegurar que las funcionalidades del sistema que espera utilizar el cliente, efectivamente funcionen correctamente y produzcan los resultados esperados.

2.1.3. Entendimiento del organigrama de la empresa

Es fundamental que Luis conozca a fondo el organigrama vigente de la empresa, para de esta forma poder elaborar un plan de comunicación efectivo que facilite el intercambio de información pertinente al proyecto dentro de la empresa.

2.1.4. Conocer los casos de uso prácticos de sus clientes

Es importante que Luis posea un conocimiento claro de los casos de uso prácticos que sus clientes le dan al MasterDevelop, para de esta forma poder estudiar de que forma se pueden mejorar, extender y otorgar valor agregado a sus clientes.

2.1.5. Entender la importancia de pruebas automatizadas

Para poder probar la calidad del Software es necesario que Luis posea conocimiento de la importancia de pruebas automatizadas y de esta forma implementarlas él mismo o contratar a un ingeniero de calidad que las implemente por él. Siempre en base a los criterios de calidad definidos para MasterDevelop que Luis sobre los cuáles Luis debe tener pleno conocimiento.

2.1.6. Una forma de modelado estándar para expresar sus ideas y conceptos a su equipo de desarrollo

Es de vital importancia que Luis sea altamente versado en algún lenguaje de modelado estándar que le permita expresar sus ideas y diseños a su equipo de desarrollo. Se recomienda UML por ser el lenguaje por excelencia de modelado de Sistemas. Por tanto, Luis debe ser altamente versado en UML al igual que su equipo, para poder comunicar ideas y diseños del sistema de manera entendible, rápida y eficaz.

2.1.7. Estadística e Inferencia

Sería muy útil, más no obligatorio, que Luis posea conocimientos intermedio-avanzados sobre temas de Estadística e Inferencia, para de esta forma poder llevar a cabo análisis de tendencias de uso, históricos de gastos, entre otros que le serían útiles para descubrir funcionalidades desperdiciadas por los usuarios o puntos críticos de gran acumulación de costos y tiempo en el proyecto.

2.2. Actitudes

2.2.1. Hacer las cosas, no solo pensarlas

Para rescatar a MasterDevelop, es de vital importancia que Luis posea una actitud proactiva y se encargue de llevar a cabo sus planes, no solamente pensarlos. Debe estar dispuesto a tomar medidas y realizar las tareas que otros no se atreverían o no podrían hacer.

2.2.2. Saber cómo priorizar

Luis debe tener una actitud que sepa respetar prioridades y debe saber como priorizar tareas según su nivel de importancia y urgencia.

2.2.3. Disposición de aprendizaje continuo

Para rescatar a MasterDevelop y continuar innovando, es de vital importancia que Luis adopte una actitud de aprendizaje continuo y nunca piense que lo sabe todo o que no se puede mejorar.

2.2.4. No hacer excusas

Si algo falla durante la ejecución de su plan de rescate, es importante que Luis adopte una actitud que le permita asumir responsablemente las consecuencias de dichos fallos, en vez de culpar a otros o hacer excusas.

2.2.5. No darse por vencido

No darse por vencido, Luis ya ha demostrado tener esta actitud y es efectivamente importante para poder llevar a cabo el plan de rescate de MasterDevelop. Siempre y cuando no se haya tocado fondo, siempre hay una salida y debe trabajarse arduamente para llegar a ella. Luis ha demostrado claramente esta actitud.

2.2.6. Atención al detalle

No hay tarea lo suficientemente pequeña como para ser sobreestimada. Si una tarea está planeada para su ejecución debe ser realizada correctamente, prestando especial atención a los detalles. Nunca se debe suponer ni sobreestimar nada.

2.3. Habilidades

2.3.1. Liderazgo

Como es de esperarse, la habilidad principal que Luis debe poseer para llevar a cabo exitosamente el rescate de MasterDevelop es la de liderazgo, Luis será responsable no solamente de planear el rescate y delegar las tareas. Luis debe hacerse cargo por velar por que dichas tareas se cumplan, tomar medidas, incentivar al equipo, definir los roles del proyecto y guiar a su equipo hacia tales goles.

2.3.2. Comunicación

Para todo jefe de proyectos es indispensable su habilidad para la comunicación, Luis debe ser capaz de expresar sus ideas, diseños, preocupaciones y sugerencias efectivamente tanto a sus superiores como a sus delegados.

2.3.3. Planificación

Sin duda es importante que Luis posea una alta habilidad para la planeación de tareas y que tome en cuenta el perfil de los miembros de su equipo, los recursos financieros disponibles, el tiempo estipulado y los recursos digitales a su disposición.

2.3.4. Manejo del riesgo

En todo proyecto siempre hay riesgos, y los hay más aún en un proyecto de rescate. Luis debe ser habilidoso en la forma en que se prepara y reacciona a los posibles riesgos que puedan presentarse durante el rescate de MasterDevelop. Luis siempre debe estar preparado.

2.3.5. Negociación

Finalmente, pero no menos importante, Luis debe ser capaz de llevar a cabo negociaciones importantes con sus superiores para acordar y llegar a un convenio respecto a los recursos necesarios que serán dispuestos para llevar a cabo el rescate de MasterDevelop.

3. ¿Consideras que Luis Aguilar era o no una persona idónea para elaborar con éxito un plan de rescate para MasterDevelop?

Si. Considero que Luis Aguilar es, efectivamente la persona idónea para elaborar y llevar a cabo el plan de rescate de MasterDevelop. Luis posee una alta trayectoria (más de quince años) en el área de innovación, por lo que una de sus principales características es su alta habilidad de adaptación y facilidad de adoptar los cambios. Esto sin duda le permitirá adaptarse al nuevo modelo de negocios de Softics y poder sortear con éxito el drástico recorte de presupuesto que estaba su disposición en el pasado.

Además de su habilidad para adaptarse al cambio, Luis estuvo presente en todo momento durante el desarrollo de MasterDevelop, desde su concepción como una idea, hasta su implementación en su primera versión y posteriores que situaron a MasterDevelop 3.0 como uno de los productos insignia de Softics. Esta experiencia sin duda es indispensable para que Luis pueda evaluar que características del software mejorar, que características remover y que se puede agregar para rescatar la viabilidad de MasterDevelop como un producto competitivo en el mercado.

Además de su experiencia en el desarrollo del sistema y de su experiencia en el área académica, otra habilidad importantísima que indudablemente califica a Luis para tomar la responsabilidad de enunciar y llevar a cabo el plan de rescate de MasterDevelop, es su conocimiento y relaciones actuales con los clientes de Softics que ya han

adoptado a MasterDevelop como parte de sus herramientas. Al estar Luis familiarizado con ellos, este podrá identificar de mejor manera que exactamente esperan los clientes obtener de MasterDevelop y de como poder mantener el producto de manera que sea posible su rentabilidad e innovación.

4. ¿Qué tipo de productos y servicios podrían integrarse a MasterDevelop para ser económicamente auto-sustentable dadas las restricciones que lo limitaban?

4.1. Diagramas de casos de uso

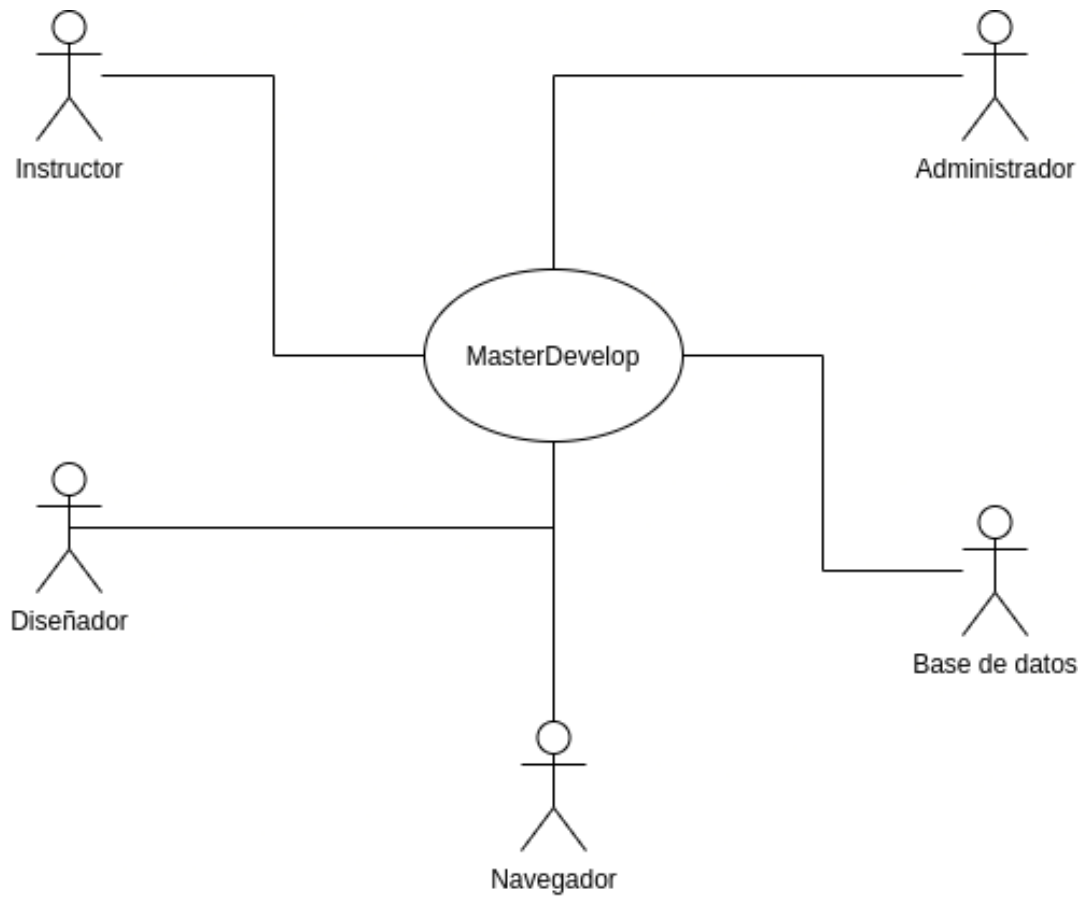


Figura 1: Diagrama de casos de uso de más alto nivel

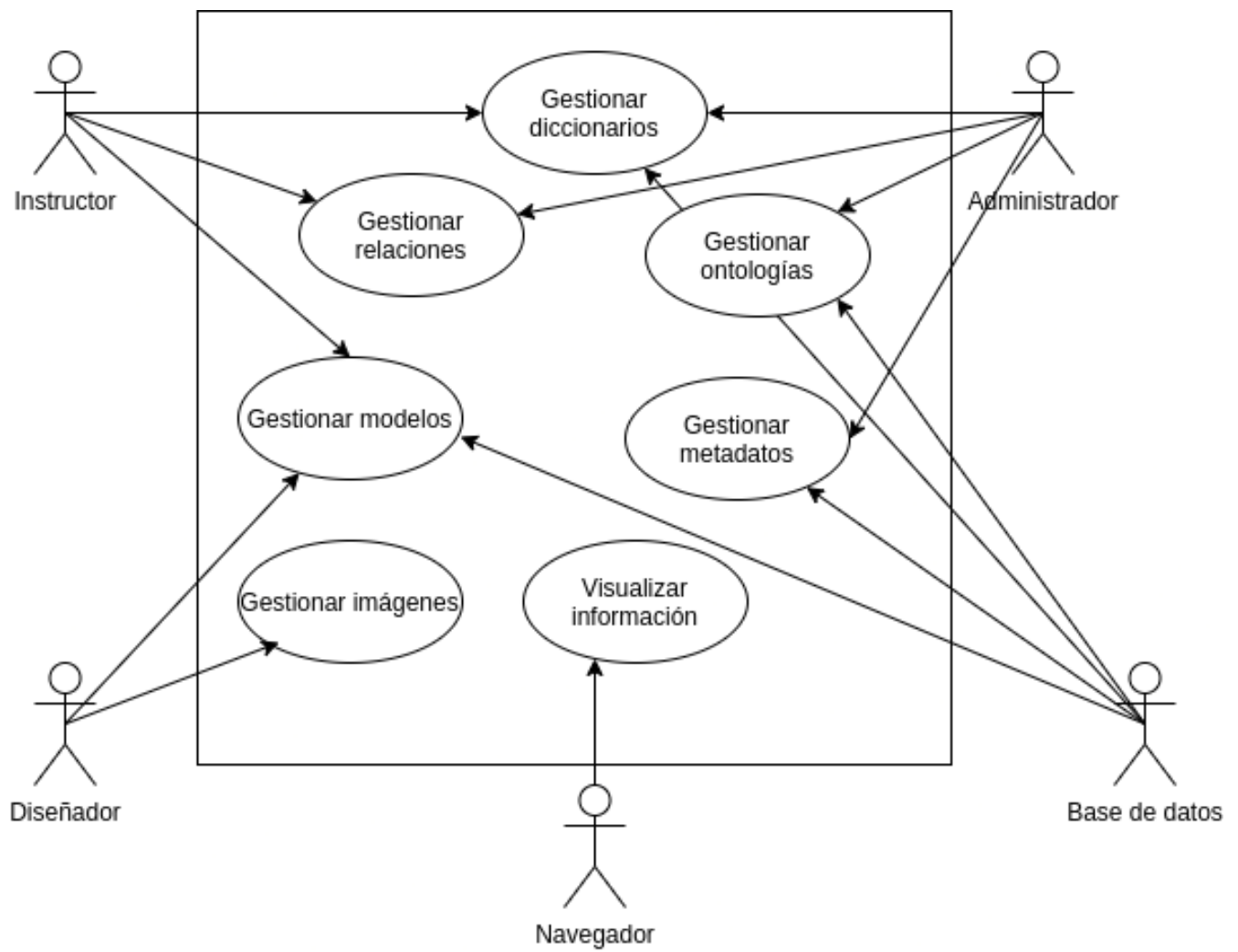


Figura 2: Diagrama de casos de uso de alto nivel

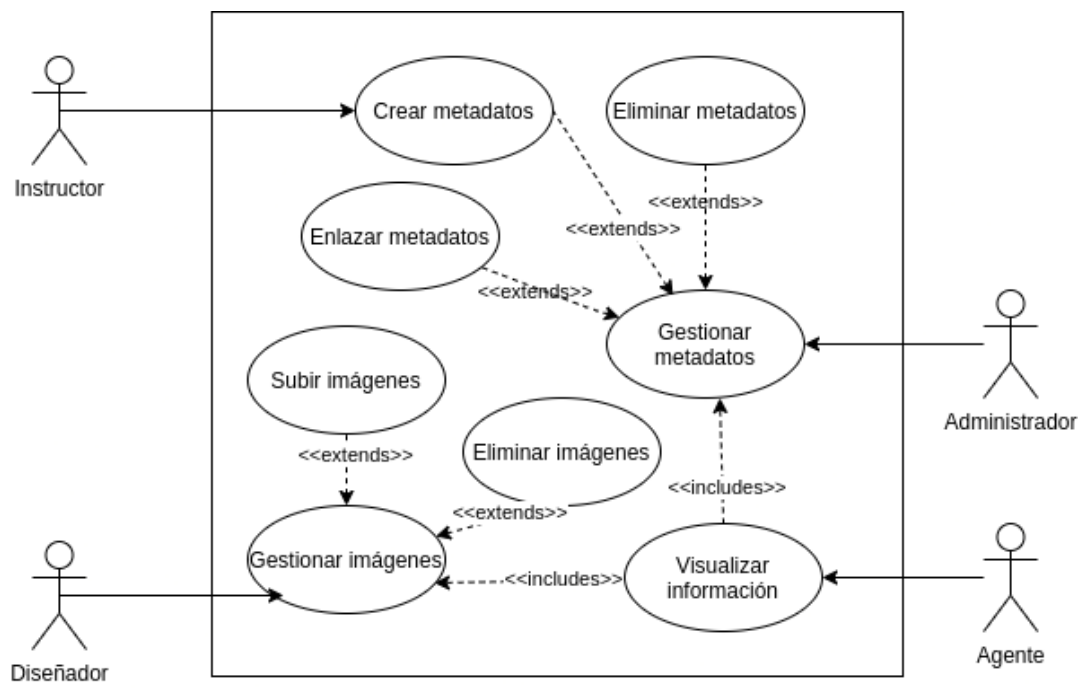


Figura 3: Diagrama de casos de uso expandido

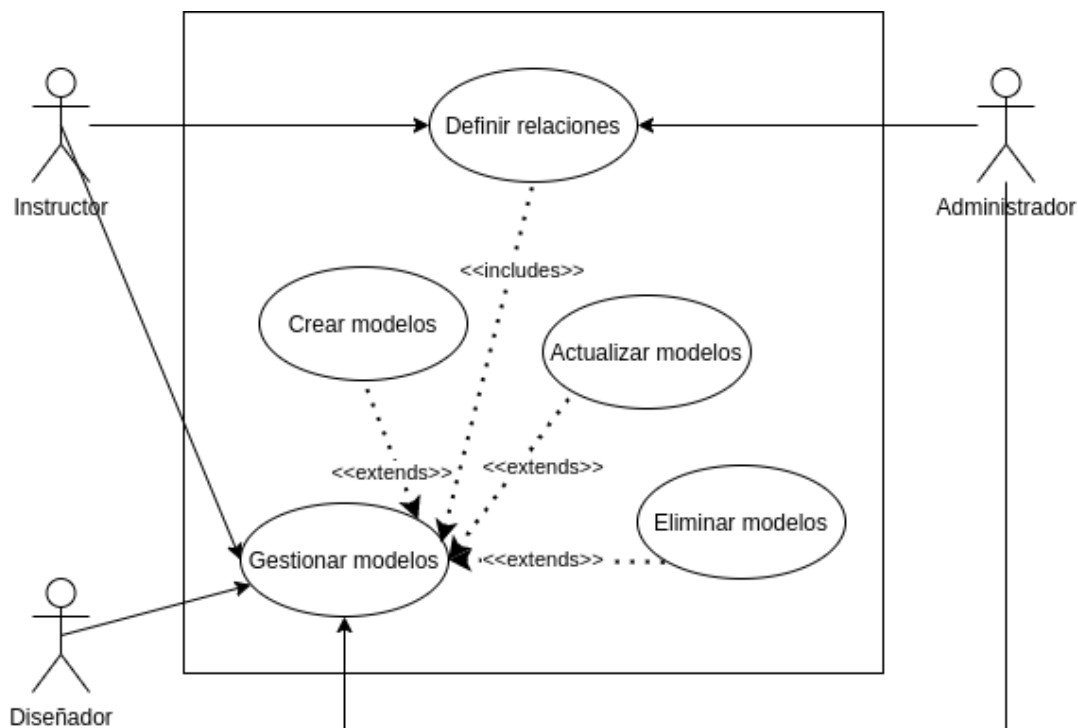


Figura 4: Diagrama de casos de uso expandido

4.2. Casos de uso

4.2.1. Crear diccionarios

| | |
|-----------------------|--|
| Nombre | Crear diccionarios |
| Autor | Renato Josué Flores Pérez |
| Fecha | 2 de enero de 2021 |
| Descripción | Los diccionarios ayudan a la web semántica a dar significado a las palabras encontradas en los textos de las vistas que desean crear los usuarios. Ayudan a identificar sustantivos, verbos, artículos, adverbios y demás en base al contexto, la semántica y los metadatos del documento. Crear y definir diccionarios es de vital importancia para la web semántica por esta razón. |
| Actores | Administrador, Instructor |
| Precondiciones | Haber creado uno o más modelos. |
| Flujo Normal | <ol style="list-style-type: none"> 1. Ingresar al panel de administración de MasterDevelop. 2. En la barra de navegación, seleccionar Diccionarios. 3. En el menú que se presenta, seleccionar crear Diccionario. 4. Llenar el diccionario a mano (no recomendable) o seleccionar importar. MasterDevelop soporta diccionarios simples en formato XML, RDF, TOML, JSON y YAML. |
| Poscondiciones | El diccionario será registrado y estará disponible para su posterior uso por parte del mecanismo de inferencias y la asignación automática de metadatos. |

4.2.2. Editar diccionarios

| | |
|-----------------------|---|
| Nombre | Editar diccionarios |
| Autor | Renato Josué Flores Pérez |
| Fecha | 2 de enero de 2021 |
| Descripción | <p>Si bien es sencillo importar diccionarios, estos comúnmente son conjuntos simples de clave valor. Sin embargo, un diccionario mucho más útil es aquel que aporta anotaciones, tipos, temas, sinónimos y relaciones entre las palabras. MasterDevelop soporta la asignación y aplicación de estas propiedades. Este caso de uso demuestra como pueden agregarse estas propiedades para enriquecer los diccionarios creados previamente de Master Develop</p> |
| Actores | Administrador, Instructor |
| Precondiciones | Haber creado uno o más diccionarios. |
| Flujo Normal | <ol style="list-style-type: none">1. Ingresar al panel de administración de MasterDevelop.2. En la barra de navegación, seleccionar Diccionarios.3. En el menú que se presenta, seleccionar editar Diccionario.4. En el drop down de diccionarios registrados, seleccionar el diccionario de interés.5. Utilizar la barra de búsqueda y los filtros para navegar con facilidad el diccionario. Por cada término de interés, seleccionar editar.6. Editar los datos del término seleccionado o agregar la información pertinente. |
| Poscondiciones | <p>El diccionario será enriquecido con la información extra disponible a cada término y como consecuencia, las inferencias y la asignación de metadatos será más acertada y rápida.</p> |

4.2.3. Definir relaciones

| | |
|--|---------------------------|
| Nombre | Definir relaciones |
| Autor | Renato Josué Flores Pérez |
| Fecha | 2 de enero de 2021 |
| Descripción Una de las funciones más útiles de MasterDevelop y que sin duda es un factor decisivo al momento de posicionarlo como un producto innovador y útil, es la habilidad de poder definir relaciones entre vistas, modelos y ontologías. De esta forma, la generación de inferencias y sugerencias útiles para el usuario se vuelve una realidad. Ya que de lo contrario, el tiempo de búsqueda que le llevaría al sistema para analizar todos los diccionarios, ontologías y modelos para calcular sus inferencias sería demasiado prologado debido a su gran volumen y en consecuencia el usuario marcaría la prolongada espera como tediosa y MasterDevelop como poco útil y práctico. | |
| Actores Administrador | |
| Precondiciones <ol style="list-style-type: none">1. Estar registrado en el sistema como usuario administrador2. Haber definido una o más ontologías3. Haber definido uno o más diccionarios4. Haber definido uno o más modelos5. Haber definido una o más vistas | |
| Flujo Normal <ol style="list-style-type: none">1. Ingresar a la página de administración de MasterDevelop.2. Seleccionar la opción de relaciones.3. El sistema mostrará un mapa conceptual gráfico que le ayudará al administrador a comprender las relaciones actualmente definidas en el proyecto.4. El usuario será capaz de crear, eliminar o modificar relaciones según le convenga con ayuda de una interfaz gráfica. | |
| Poscondiciones La red de relaciones del sistema que puede crear el usuario a partir de esta herramienta le será de vital utilidad al sistema mientras este crece en robustez y dominio de datos. La red de relaciones le permitirá al sistema de inferencias saber donde priorizar sus búsquedas y le permitirá obtener resultados válidos con una gran velocidad. | |

4.2.4. Crear ontologías

| | |
|--|---------------------------|
| Nombre | Crear Ontologías |
| Autor | Renato Josué Flores Pérez |
| Fecha | 2 de enero de 2021 |
| Descripción Si bien el Diccionario representa un conjunto de datos que poseen nombre, significado y tipo. Este es de poca utilidad si no está bien definido qué es cada tipo y que tipos se encuentran disponibles. Los tipos por defecto son aquellos encontrados en el lenguaje natural que los desarrolladores de MasterDevelop consideraron oportunos para su implementación. Algunos ejemplos son “Adverbios”, “Artículos”, “Pronombres”, “Sustantivos”, “Verbos”. Sin embargo, estos tipos pueden ser extendidos o removidos por el usuario. Nuevos tipos pueden ser definidos y cargados al sistema. Al conjunto de tipos junto a su definición y propiedades MasterDevelop le llama Ontología. | |
| Actores Administrador, Instructor. | |
| Precondiciones 1. Estar registrado en el sistema como usuario administrador. | |
| Flujo Normal 1. Ingresar a la página de administración 2. Seleccionar la opción Ontologías. 3. Seleccionar la opción Crear. 4. Seleccionar la opción Agregar Tipo. 5. Llenar la información requerida por el Sistema para la definición de un Tipo. | |
| Flujo Alternativo El sistema también soporta la carga de nuevos Tipos de manera masiva. Estos deben ser cargados en un documento con formato XML que defina de manera concisa e inequívoca a cada tipo que será parte de la Ontología. Si alguno de los tipos a cargar viola las reglas de integridad de Tipos definidas por el sistema se alertará al usuario y se rechazará el archivo entero. | |
| Poscondiciones El usuario será capaz de consultar, manipular y eliminar la Ontología creada. La ontología quedará registrada en el sistema para su posterior uso al momento de relacionarla con el resto de elementos del sistema. | |

4.2.5. Asignar Metadatos

| | |
|--------------------------|---|
| Nombre | Asignar Metadatos |
| Autor | Renato Josué Flores Pérez |
| Fecha | 2 de enero de 2021 |
| Descripción | <p>Los metadatos en el contexto de MasterDevelop son pequeñas piezas de información que se le añaden a un documento o vista que le permite a la red semántica identificar su nombre, tamaño, categoría, tema y demás información relevante para que el sistema de inferencias sea capaz de relacionar el documento con los modelos, diccionarios u ontologías definidas en el sistema. Los metadatos pueden ser añadidos a un documento o vista de forma manual o automática. Esta última es una de las funcionalidades que hacen a MasterDevelop tan útil y reconocido.</p> |
| Actores | Diseñador, administrador, Instructor |
| Precondiciones | <ol style="list-style-type: none">1. Estar registrado en el sistema.2. Haber creado una o más vistas. |
| Flujo Normal | <ol style="list-style-type: none">1. Ingresar a la página de inicio del sistema.2. Seleccionar la opción Gestionar Vistas.3. Escoger la vista a la cuál se le desea asignar metadatos.4. Seleccionar la opción Asignar Metadato.5. Llenar el formulario que se le presentará con la información relevante para definir el metadato.6. Seleccionar guardar. |
| Flujo Alternativo | <p>La forma en que se asignan metadatos de manera automática es bastante simple. El usuario simplemente debe enlazar la vista objetivo con un modelo en concreto. El sistema posteriormente se encargará de realizar un análisis de inferencia recursivo en busca de documentos similares relacionados con dicho modelo y creará metadatos relevantes para el nuevo documento. La certeza de estos metadatos dependerá de la cantidad de documentos cargados al sistema y la robustez de las relaciones dentro de la red semántica definidas por el usuario. Por esta razón esta opción no es recomendada cuando aún se están creando los primeros documentos de MasterDevelop. Sin embargo, es extremadamente útil cuando el sistema ha crecido y en este momento facilitará al usuario la creación de nuevas vistas, volviendolo en cierto modo, auto-sustentable a partir de este punto.</p> |
| Poscondiciones | <p>El usuario será capaz de recibir inferencias y sugerencias cuando visualice sus documentos por medio del navegador en base a los metadatos asociados a la vista en cuestión.</p> |

4.2.6. Gestión de imágenes

| | |
|-----------------------|---|
| Nombre | Gestión de imágenes |
| Autor | Renato Josué Flores Pérez |
| Fecha | 2 de enero de 2021 |
| Descripción | <p>El manejo de recursos multimedia es sin duda clave para cualquier red web. MasterDevelop por el momento soporta únicamente el manejo de imágenes. Sin embargo, si el proyecto es rescatado exitosamente, se planea introducir soporte para otros formatos multimedia en la versión 4.0</p> |
| Actores | Diseñador |
| Precondiciones | <ol style="list-style-type: none">1. Haber creado una o más vistas |
| Flujo Normal | <ol style="list-style-type: none">1. Ingresar a la página de inicio del sistema.2. Seleccionar Gestionar Vistas.3. Seleccionar la Vista a la cuál se le desea adjuntar imágenes.4. Seleccionar Editar.5. Seleccionar la opción Agregar Imágenes6. La interfaz de usuario cambiará a un formato que no le permitirá editar texto, sin embargo creará un formato Drag and Drop sobre el cuál el usuario será capaz de arrastrar y posicionar imágenes donde considere pertinente.7. Seleccionar Guardar8. La vista será actualizada e incorporará las imágenes que ha decidido el usuario. |
| Poscondiciones | <p>El usuario podrá navegar y visualizar las imágenes en el navegador y asimismo será capaz de editarlas o removerlas en caso de ser necesario.</p> |

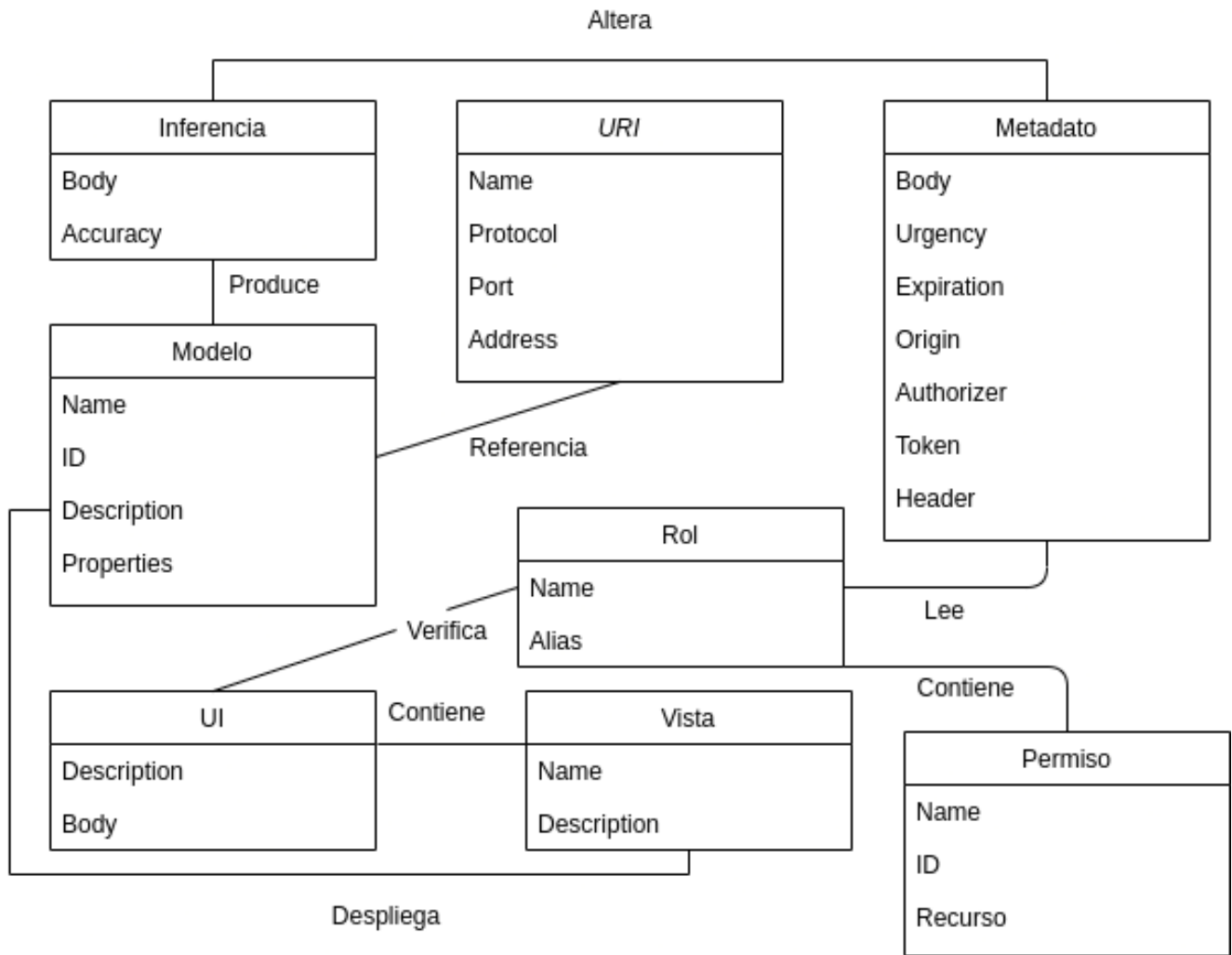


Figura 5: Modelo conceptual, identificando los principales objetos de negocio

5. ¿Cómo resolverías el caso de mantener vivo a Master Develop?

5.1. Aplicar la filosofía Lean

Debido a la naturaleza de la problemática que enfrenta Softics, es preciso adoptar la filosofía Lean para el desarrollo y mantenimiento del software. Esto implica que el primer paso debe ser entender y tener bien definido el significado de valor para cada uno de los clientes. En base a esta información se debe crear un mapa de procesos que identifiquen todos los procesos que están involucrados con el sistema. En base a este mapa y la definición de valor del cliente, se debe proceder a seleccionar cuidadosamente los procesos más críticos e importantes, así como los procesos que no aportan valor al cliente en lo absoluto y deben ser eliminados.

Una vez identificados y eliminados los desperdicios presentes en el proyecto se obtendrá un recorte de gastos y ahorro considerable. Lo cual se acopla apropiadamente a las nuevas políticas de negocio de Softics.

5.2. Cambiar el modelo de negocio

Actualmente, Softics utiliza un modelo de licencia de código propietario para la venta de MasterDevelop así como un plan de soporte técnico y consultoría. Si bien el plan de soporte técnico representa una gran parte de las ganancias de Softics en cuanto a MasterDevelop, esta estrategia dejará de ser viable en el futuro cercano debido al recorte de personal y en consecuencia dificultad de proveer soporte técnico para nuevos clientes. Por esta razón, se sugiere

cambiar el modelo de negocio sobre el cuál se basa la venta y publicidad de MasterDevelop por un enfoque más académico. Es decir, dada la innovación inherente del producto y el alta estima hacia Softics por parte del estado como empresa tecnológica nacional, deben aprovecharse estas propiedades y vender una versión del software con un enfoque didáctico y académico. Los posibles clientes deben ser universidades nacionales e institutos de enseñanza técnica. De esta forma, la necesidad de soporte técnico dedicado podrá ser reducido en gran medida y reemplazado por modelos de suscripción por parte de entidades educativas que pagarán por el software y clases de capacitación de personal inicial. Una vez esté capacitado el personal educativo que elija nuestro producto ellos se encargarán de enseñar su uso a sus propios alumnos y de proveer el soporte técnico durante los laboratorios o actividades que se lleven a cabo.

Nuevamente, cabe recalcar la ventaja que posee MasterDevelop para ser distribuido como material académico, puesto que será una gran oportunidad para distintas organizaciones educativas poder utilizar material nacional de alta calidad que les permita enseñar conceptos clave sobre redes, hipertexto, redes semánticas, ontologías digitales, compiladores, interpretes, entre otros conceptos clave que se encargan del correcto funcionamiento de MasterDevelop.

5.3. Autosostenibilidad

Una vez adoptado el nuevo modelo de negocio y luego de negociar exitosamente algunos acuerdos con diferentes instituciones educativas entonces deberá prestarse especial atención a las capacitaciones del personal que impartirá los cursos de interés. Este proceso podrá requerir de una inversión de recursos considerable, sin embargo estos recursos financieros serán cobrados a las instituciones como parte de los costos de distribución e instalación.

Una vez las capacitaciones hayan finalizado, el proyecto será autosustentable, ya que el personal interno de las instituciones estará calificado y será responsable de dar el soporte técnico necesario a sus alumnos, provocando que Softics se preocupe solamente por cobrar las cuotas de uso mensuales o anuales.