**TABLA DE ESPECIFICACIÓN DEL PROBLEMA DE INGENIERÍA DE SOFTWARE, identificando los siguientes elementos**

|  |  |
| --- | --- |
| CLIENTE | ReadX |
| USUARIO | Personas interesadas en usar la aplicación |
| REQUERIMIENTOS FUNCIONALES | 1. Registrar usuarios, regulares y premium. 2. Gestionar productos bibliográficos: registrar, modificar y borrar libros y revistas. 3. Generar automáticamente objetos en el sistema para cada tipo de usuario y producto bibliográfico. 4. Permitir a un usuario comprar un libro. 5. Permitir a un usuario suscribirse a una revista. 6. Permitir a un usuario simular una sesión de lectura (regular y premium). 7. Presentar al usuario su Biblioteca de Productos Bibliográficos. 8. Generar informes con los datos registrados: |
| CONTEXTO DEL PROBLEMA | Para desarrollar un prototipo de software que les permita gestionar su negocio de publicaciones a nivel mundial. A continuación se presentan las funcionalidades a desarrollar |
| REQUERIMIENTOS NO FUNCIONALES | Rendimiento:  El sistema debe ser capaz de manejar de manera eficiente un gran número de usuarios, productos y transacciones.  Los tiempos de respuesta para las interacciones de los usuarios, como cargar la biblioteca o iniciar una sesión de lectura, deben ser rápidos y dentro de límites aceptables.  Escalabilidad:  El sistema debe estar diseñado para adaptarse al crecimiento futuro y manejar un número creciente de usuarios, productos y transacciones.  Debe ser capaz de admitir la adición de nuevos tipos de productos y roles de usuario sin modificaciones significativas.  Seguridad:  Los datos de los usuarios, incluyendo información personal y detalles de pago, deben almacenarse y transmitirse de manera segura.  El acceso a operaciones sensibles, como modificar o eliminar productos, debe estar restringido a personal autorizado.  Confiabilidad:  El sistema debe ser robusto y confiable, minimizando la posibilidad de fallos, pérdida de datos o tiempo de inactividad.  Debe manejar los errores de manera adecuada, proporcionando mensajes de error apropiados y recuperándose de fallas sin comprometer la experiencia del usuario.  Usabilidad:  La interfaz de usuario debe ser intuitiva, fácil de usar y visualmente atractiva.  El sistema debe proporcionar instrucciones claras y orientación a los usuarios, ayudándolos a navegar por diferentes funciones y características.  El sistema debe ser accesible para usuarios con discapacidades, cumpliendo con estándares y pautas de accesibilidad.  Mantenibilidad:  El código fuente debe ser modular, bien estructurado y fácil de entender, fomentando la reutilización de código y facilitando futuras mejoras o correcciones de errores.  Debe proporcionarse documentación adecuada, incluyendo comentarios de código, documentación de API y manuales de usuario, para ayudar en el mantenimiento del sistema y su desarrollo futuro.  Extensibilidad:  El sistema debe diseñarse con flexibilidad, permitiendo la adición de nuevos tipos de productos, roles de usuario y funcionalidades con un esfuerzo mínimo.  La arquitectura debe admitir la integración con sistemas o servicios externos, como pasarelas de pago o herramientas de análisis.  Informes y análisis:  La función de generación de informes debe generar informes en tiempo real con datos precisos y actualizados.  Los informes generados deben presentarse en un formato claro y comprensible, proporcionando información valiosa sobre el comportamiento de los usuarios, la popularidad de los productos y el rendimiento de las ventas. |

**Tabla de análisis de requerimientos funcionales (Nota: Una tabla por cada requerimiento funcional)**

| Nombre o identificador | Registrar usuarios, regulares y premium. | | |
| --- | --- | --- | --- |
| Resumen | El Sistema debe permitir registrar usuarios Premium y Standard correctamente | | |
| Entradas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| user\_name | String | No pasar el numero de caracteres permitido. |
| Id | String | No pasar el numero de caracteres permitido y ser hexadecimal |
| vinculartionDate | Calendar | A la hora de ingresar los valores deben ser numéricos. |
| User\_type | String | El tipo de usuario no puede ser diferente a Standard o Premium |
|  |  |  |
| Resultado o postcondición | Registro exitoso o erróneo de Usuarios | | |
| Salidas | Nombre Salida | Tipo de dato | Resultado |
| Usuario | Objeto | String que compruebe que se ha creado exitosamente |
| msg | String | String que niegue el correcto registro |
|  |  |  |
|  |  |  |
|  |  |  |

| Nombre o identificador | Registro, Modificación y Eliminación de libros y revistas. | | |
| --- | --- | --- | --- |
| Resumen | El sistema debe permitir registrar, modificar por atributo escogido, y eliminar productos bibliográficos. | | |
| Entradas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| product\_id | String | No pasar el número de caracteres permitido y ser alfanumerico. |
| Product\_name | String | No pasar el número de caracteres permitido |
| publicationDate | Calendar | A la hora de ingresar los valores deben ser numéricos. |
| productCost | String | No debe sobrepasar el número de dígitos permitido |
| producto\_url | String | La cadena de texto debe estar dentro del rango de caracteres permitidos |
| Resultado o postcondición | Registro, Modificación y Eliminación exitoso o erróneo de productos bibliograficos | | |
| Salidas | Nombre Salida | Tipo de dato | Resultado |
| Producto | Objeto | String que compruebe que se ha creado exitosamente |
| msg | String | String que niegue el correcto registro |
|  |  |  |
|  |  |  |
|  |  |  |

| Nombre o identificador | Auto generación de objetos | | |
| --- | --- | --- | --- |
| Resumen | El Sistema debe permitir generar automáticamente una vez ejecutado el código, tanto productos bibliográficos como usuarios | | |
| Entradas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| user\_Atributes | String, int, Calendar | No pasar el número de caracteres permitido y llenar acordemente su tipo de dato. |
| Bibliographic\_atributs | String, int, Calendar | No pasar el número de caracteres permitido y llenar acordemente su tipo de dato. |
|  |  |  |
|  |  |  |
|  |  |  |
| Resultado o postcondición | Creación del método de casos de testeo (creación de objetos Usuario y Biblio Products )correcto. | | |
| Salidas | Nombre Salida | Tipo de dato | Resultado |
| Usuario /Producto | Objeto | Casos de testeo creados y generados automáticamente una vez ejecutado el código |
| msg | String | String que niegue el correcto registro |
|  |  |  |
|  |  |  |
|  |  |  |

| Nombre o identificador | Compra de Productos por parte de los usuarios | | |
| --- | --- | --- | --- |
| Resumen | El Sistema debe permitir registrar usuarios Premium y Standard correctamente | | |
| Entradas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| User | Objeto | Al ingresar el número de referencia del Usuario, asegurase que sea un numero valido o que el Usuario exista |
| Product (Book/Magaizine) | Objeto | Al ingresar el número de referencia del Producto, asegurase que sea un numero valido o que el producto exista |
|  |  |  |
|  |  |  |
|  |  |  |
| Resultado o postcondición | Compra exitosa o errónea | | |
| Salidas | Nombre Salida | Tipo de dato | Resultado |
| msg | String | String que compruebe que se ha hecho exitosamente |
| msg | String | String que niegue el correcto registro, ya sea por restricciones de usuario como la capacidad de compra u otro motivo. |
|  |  |  |

| Nombre o identificador | Sesión de lectura | | |
| --- | --- | --- | --- |
| Resumen | El Sistema debe permitir realizar una sesión de lectura, escogiendo el libro del que desea leer, si el usuario es Standard le saldrá un anuncio al principio de la sesión y cada 20 paginas. | | |
| Entradas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| User | Objeto | Al ingresar el número de referencia del Usuario, asegurase que sea un numero valido o que el Usuario exista |
| Product | String | El producto debe pertenecer al Usuario escogido |
|  |  |  |
|  |  |  |
|  |  |  |
| Resultado o postcondición | Registro exitoso o erróneo de Usuarios | | |
| Salidas | Nombre Salida | Tipo de dato | Resultado |
| product\_name | String | String que compruebe que se ha escogido el libro deseado |
| read\_pages | int | Contador que vaya acumulando las paginas leídas |
|  |  |  |
|  |  |  |
|  |  |  |

| Nombre o identificador | Biblioteca de usuaro | | |
| --- | --- | --- | --- |
| Resumen | El Sistema debe permitir presentarle al usuario la biblioteca con cada uno de sus productos bibliográficos comprados o adquiridos | | |
| Entradas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| Usuario | Objeto | Al ingresar el número de referencia del Usuario, asegurase que sea un numero valido o que el Usuario exista |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| Resultado o postcondición | Se muestra la biblioteca del usuario ingresado | | |
| Salidas | Nombre Salida | Tipo de dato | Resultado |
|  |  |  |
| [][]Matrix | String | Se muestra en pantalla la matrix de los productos adquiridos por ese usuario en específico |
|  |  |  |
|  |  |  |
|  |  |  |