**Juan Pablo Parra**

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

|  |  |
| --- | --- |
| CLIENTE | ReadX |
| USUARIO | ReadX - Usuario |
| REQUERIMIENTOS FUNCIONALES | 1. Registrar Usuarios Regular 2. Registrar Usuarios Premium 3. Registrar Libros 4. Modificar Libros 5. Eliminar Libros o Revistas 6. Registrar Revistas 7. Modificar Revistas 8. Comprar un libro – Suscribirse a una Revista 9. Permitir una sesión de lectura 10. Presentar una biblioteca de Productos 11. Generar informes |
| CONTEXTO DEL PROBLEMA | 1. ReadX nos pide desarrollar un prototipo de software que les permita gestionar su negocio de publicaciones a nivel mundial. Para ellos nos pide gestionar los productos bibliográficos es decir registrar, modificar y eliminar libros o revistas. Los cuales tienen algunos tributos compartidos y otros diferentes. Nos pide registrar usuarios, que pueden ser de dos tipos, regulares o premium. Siendo la diferencia la cantidad de libros que pueden comprar y revistas que se pueden suscribir. Donde el usuario regular tiene permitido comprar 5 libros y suscribirse a 2 revistas y así mismo se le presentaran anuncios. Por el otro lado, el premium puede adquirir tantos productos quieran. Los usuarios pueden comprar o suscribirse a una revista donde se tiene que actualizar la cantidad de libros vendidos o suscripciones activas. Se debe de presentar una biblioteca a los usuarios. Simular una sesión de lectura donde aparezca el título que está leyendo y un menú sencillo. Para finalizar se debe de generar unos reportes. En total son 5: Informar el acumulado total de páginas leídas. Informar el género de libro y categoría de revista más leídas. Informar el Top 5 de libros y el Top 5 de revistas más leídas. De cada género, informar el número de libros vendidos y el valor total de ventas ($). De cada categoría, informar el número de suscripciones activas y el valor total pagado por suscripciones. |
| REQUERIMIENTOS NO FUNCIONALES | 1. Tiene que ser a nivel mundial. 2. Puede ser que a futuro hayan mas tipos de usuarios. |

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

| Nombre o identificador | 1. **Registrar Usuarios Regulares** | | |
| --- | --- | --- | --- |
| Resumen | **El programa debe de permitir registrar usuarios de tipo regular. Pidiéndole al usuario su nombre, el id y automáticamente el programa registrar la fecha de vinculación.** | | |
| Entradas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| **Nombre** | **String** |  |
| **Id** | **String** |  |
| **fechaVincu** | **Calendar** | **(Se debe de pedir 3 ints)** |
| Resultado o postcondición | Se registra el usuario y aparece un mensaje de que se generó el registro o por el contrario un error. | | |
| Salidas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| Mensaje | String |  |

| Nombre o identificador | 1. **Registrar Usuarios Premium** | | |
| --- | --- | --- | --- |
| Resumen | **El programa debe de permitir registrar usuarios de tipo Premium. Pidiéndole al usuario su nombre, el id y automáticamente el programa registrar la fecha de vinculación.** | | |
| Entradas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| **Nombre** | **String** |  |
| **Id** | **String** |  |
| **fechaVincu** | **Calendar** | **(Se debe de pedir 3 ints)** |
| Resultado o postcondición | Se registra el usuario y aparece un mensaje de que se generó el registro o por el contrario un error. | | |
| Salidas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| Mensaje | String |  |

| Nombre o identificador | 1. **Registrar Libros** | | |
| --- | --- | --- | --- |
| Resumen | El programa debe de permitir registrar productos bibliográficos de tipo Libro. Pidiéndole al usuario su nombre, el id, el número de páginas, la fecha de publicación, la URL, el precio, el número de ejemplares vendidos, una corta reseña y su género. | | |
| Entradas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| **Nombre** | **String** |  |
| **Id** | **String** |  |
| **NumPages** | **Int** |  |
| **datePublication** | **Calendar** |  |
| **URL** | **String** |  |
| **Price** | **Double** |  |
| **numSold** | **int** |  |
| **Review** | **String** |  |
| **Genre** | **int** |  |
| Resultado o postcondición | Se registra el libro y aparece un mensaje de que se generó el registro o por el contrario un error. | | |
| Salidas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| Mensaje | String |  |

| Nombre o identificador | 1. **Modificar Libros** | | |
| --- | --- | --- | --- |
| Resumen | El programa debe de permitir modificar productos bibliográficos de tipo Libro. Pidiéndole al usuario que atributo quiere modificar. De que libro y su nuevo valor. | | |
| Entradas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| **Nombre** | **String** | **(Solo se puede ingresar el 1)** |
| **Id** | **String** | **(Solo se puede ingresar el 2)** |
| **NumPages** | **Int** | **(Solo se puede ingresar el 3)** |
| **datePublication** | **Calendar** | **((Solo se puede ingresar el 4)** |
| **URL** | **String** | **(Solo se puede ingresar el 5)** |
| **Price** | **Double** | **(Solo se puede ingresar el 6)** |
| **numSold** | **int** | **(Solo se puede ingresar el 7)** |
| **Review** | **String** | **(Solo se puede ingresar el 8)** |
| **Genre** | **int** | **(Solo se puede ingresar el 9)** |
| Resultado o postcondición | Se modificar el libro y aparece un mensaje de que se generó la modificación o por el contrario un error. | | |
| Salidas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| Mensaje | String |  |

| Nombre o identificador | 1. **Eliminar Libros o Revistas** | | |
| --- | --- | --- | --- |
| Resumen | El programa debe de permitir eliminar productos bibliográficos de tipo Libro o Revista. Pidiéndole que libro o revista. | | |
| Entradas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| **ProductPosition** | **Int** |  |
| Resultado o postcondición | Se elimina el libro o revista y aparece un mensaje de que se generó la eliminación o por el contrario un error. | | |
| Salidas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| Mensaje | String |  |

| Nombre o identificador | 1. **Registrar Revistas** | | |
| --- | --- | --- | --- |
| Resumen | El programa debe de permitir registrar productos bibliográficos de tipo Revista. Pidiéndole al usuario su nombre, el id, el número de páginas, la fecha de publicación, la URL, el precio, cada cuanto se publica y la categoría. | | |
| Entradas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| **Nombre** | **String** |  |
| **Id** | **String** |  |
| **NumPages** | **Int** |  |
| **datePublication** | **Calendar** |  |
| **URL** | **String** |  |
| **Price** | **Double** |  |
| **periodicalEmission** | **String** |  |
| **Category** | **Int** |  |
| Resultado o postcondición | Se registra la revista y aparece un mensaje de que se generó el registro o por el contrario un error. | | |
| Salidas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| Mensaje | String |  |

| Nombre o identificador | 1. **Modificar Revistas** | | |
| --- | --- | --- | --- |
| Resumen | El programa debe de permitir modificar productos bibliográficos de tipo Revista. Pidiéndole al usuario que atributo desea modificar. De que revista y su nuevo valor. | | |
| Entradas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| **Nombre** | **String** | **(Solo se puede ingresar el 1)** |
| **Id** | **String** | **(Solo se puede ingresar el 2)** |
| **NumPages** | **Int** | **(Solo se puede ingresar el 3)** |
| **datePublication** | **Calendar** | **((Solo se puede ingresar el 4)** |
| **URL** | **String** | **(Solo se puede ingresar el 5)** |
| **Price** | **Double** | **(Solo se puede ingresar el 6)** |
| **periodicalEmission** | **String** | **(Solo se puede ingresar el 7)** |
| **Category** | **Int** | **(Solo se puede ingresar el 8)** |
| Resultado o postcondición | Se modifica la revista y aparece un mensaje de que se generó la modificación o por el contrario un error. | | |
| Salidas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| Mensaje | String |  |

| Nombre o identificador | 1. **Comprar Libros- Suscriberse Revistas** | | |
| --- | --- | --- | --- |
| Resumen | El programa debe de permitir comprar productos bibliográficos de tipo Libro. Pidiéndole al usuario que libro desea comprar. Esto se almacena en el arrayList de transacciones | | |
| Entradas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| **bookSell-magazineSell** | **int** |  |
| Resultado o postcondición | Se compra el libro y aparece un mensaje de que se generó la compra o por el contrario un error. | | |
| Salidas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| Mensaje | String |  |

| Nombre o identificador | 1. **Permitir una sesión de lectura** | | |
| --- | --- | --- | --- |
| Resumen | **El programa debe de permitir una sesión de lectura a usuarios de tipo regular y premium. Para cualquier tipo de producto bibliográfico. Pidiéndole que usuario es y que producto quiere simular leer. Así mismo, permite ir una página adelante o atrás como salir de la bibloteca.** | | |
| Entradas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| **Nombre** | **Int** |  |
| **NombreProducto** | **Int** |  |
| **Siguiente** | **String** | **Solo si escribe S** |
| **Atrás** | **String** | **Solo si escribe A** |
| **Salir** | **String** | **Solo si escribeB** |
| Resultado o postcondición | Aparece la simulación o aparece un menaje de error. | | |
| Salidas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| Todo el mensaje que hace referencia a la sesión de lectura | String |  |

| Nombre o identificador | 1. **Permitir biblioteca al usuario** | | |
| --- | --- | --- | --- |
| Resumen | **El programa debe de permitir una biblioteca a usuarios de tipo regular y premium. Para regular solo puede ingresar 5 libros y 2 revistas. Para cualquier tipo de producto bibliográfico. Pidiéndole que usuario es. Así mismo, permite ir a la siguiente página de la biblioteca o atrás como salir de la bibloteca.** | | |
| Entradas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| **Nombre** | **Int** |  |
| **CodigoProducto** | **Int** |  |
| **Siguiente** | **String** | **Solo si escribe S** |
| **Atrás** | **String** | **Solo si escribe A** |
| **Salir** | **String** | **Solo si escribeB** |
| Resultado o postcondición | Aparece la biblioteca o aparece un menaje de error. | | |
| Salidas | Nombre entrada | Tipo de dato | Condición de selección o repetición |
| Todo el mensaje que hace referencia a la biblioteca | String |  |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| Registrar Usuarios Regular | Clase User-Regular | User(String nameUser, String iD, Calendar dateVincu) |
| Clase Controller | registerRegular(String nameUser, String iD, Calendar dateVincu) : boolean |
| Clase Executable | registerUsers() : void |
| Registrar Usuarios Premium | Clase User-Premium | User(String nameUser, String iD, Calendar dateVincu) |
| Clase Controller | registerPremium(String nameUser, String iD, Calendar dateVincu) : boolean |
| Clase Executable | registerUsers() : void |
| Registrar Libros | Clase ProductoBiblio | ProductoBiblio(String name, String id, int numPages, Calendar datePublication, String uRL, double price, int numPagesRead) |
| Clase Book | Book(String name, String id, int numPages, Calendar datePublication, String uRL, double price, int numPagesRead, int numSold, String review, Genre genre) |
| Clase Controller | registerBook(String name, String id, int numPages, Calendar datePublication, String uRL, double price, int numPagesRead, int numSold, String review, int genre) : boolean |
| Clase Executable | registerBooks() : void |
| Modificar Libros |  |  |
| Clase ProductoBiblio | ProductoBiblio(String name, String id, int numPages, Calendar datePublication, String uRL, double price, int numPagesRead) |
| Clase Book | Book(String name, String id, int numPages, Calendar datePublication, String uRL, double price, int numPagesRead, int numSold, String review, Genre genre)  setName(String name) : void  setId(String id) : void  setNumPages(int numPages) : void  setDatePublication(Calendar datePublication) : void  setuRL(String uRL) : void  setPrice(double price) : void  setNumPagesRead(int numPagesRead) : void  setReview(String review) : void  setNumSold(int numSold) : void  setGenre(Genre genre) : void |
| Clase Controller | modifyBook(int posicionBook, int atributo, String modificacion) : boolean  consultRegisterBooks() : String |
| Clase Executable | modifyBooks() : void |
| Eliminar Libros o Revistas | Clase Controller | deleteBookAndMagazine(int bookPosition) : String |
|  | Clase Executable | deleteMagazine() : void  deleteBook() : void |
| Registrar Revistas | Clase ProductoBiblio | ProductoBiblio(String name, String id, int numPages, Calendar datePublication, String uRL, double price, int numPagesRead) |
| Clase Magazine | Magazine(String name, String id, int numPages, Calendar datePublication, String uRL, double price,int numPagesRead, String periodicalEmission, int numSuscripActives, Category category) |
| Clase Controller | registerMagazine(String name, String id, int numPages, Calendar datePublication, String uRL, double price,int numPagesRead, String periodicalEmission, int numSuscripActives, int category) : boolean |
| Clase Executable | registerMagazine() : void |
| Modificar Revistas | Clase ProductoBiblio | ProductoBiblio(String name, String id, int numPages, Calendar datePublication, String uRL, double price, int numPagesRead) |
| Clase Magazine | Magazine(String name, String id, int numPages, Calendar datePublication, String uRL, double price,int numPagesRead, String periodicalEmission, int numSuscripActives, Category category)  setName(String name) : void  setId(String id) : void  setNumPages(int numPages) : void  setDatePublication(Calendar datePublication) : void  setuRL(String uRL) : void  setPrice(double price) : void  setNumPagesRead(int numPagesRead) : void  setCategory(Category category) : void  setPeriodicalEmission(String periodicalEmission) : void  setNumSuscripActives(int numSuscripActives) : void |
| Clase Controller | modifyMagazine(int posicionMagazine, int atributo, String modificacion) : boolean  consultRegisterMagazines() : String |
| Clase Executable | modifyMagazines() : void |
| Comprar un libro | Clase Transactions | Transaction(double purchaseValue, String nameBooks) |
| Clase Controller | SellBook() : boolean |
| Clase Executable | SellBook() : void |
| Suscribirse a una revista | Clase Transactions | Transaction(double purchaseValue, String nameBooks) |
| Clase Controller | SellMagazine() : boolean |
| Clase Executable | SellMagazine() : void |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| Permitir una sesión de lectura | Clase ProductoBiblio | ProductoBiblio(String name, String id, int numPages, Calendar datePublication, String uRL, double price, int numPagesRead)  addPagesRead() : void  addPagesReadTemp() : void  removePagesReadTemp() : void  setPagesReadTempFinal() : vid |
| Clase Book | showAdvertisemente () : Boolean  addPagesAd () : void |
| Clase Controller | simulacionLectura1 (int userPosition, String codigoProducto) : String  adReadingSesion(int userPosition, String productoCodigo, String optionMenuSimu) : void  verificarId(String codigoProducto) : boolean |
| Clase Executable | sesionDeLectura () : void |
| Presentar una biblioteca de Productos | Clase Controller | printStringMatrix (int userPositon): String  pasardeArryListAMatrix(int opcionUser) : String |
| Clase Executable | mostrarBiblotecaUser () : void |
| Generar informes | Clase Controller | totalNumberOfPagesRead(): String  genreMostReadBook() : String  categryMostReadMagazine() : String  mostSoldBookMagazines() : String  fillBookList() : void  fillMagazineList() : void  top5Book() : String  top5Magazine() : String |
| Clase Executable | req8() : void |