

Requerimiento Funcional	Nombre de la Clase	Nombre del método
R1: Register users, regular and premium.	Clase Main	registerUser()
	Clase Controller	createUser(int userType, String name, String cc, Calendar date)
	Clase User	User(String name, String cc, Calendar date)
	Clase UserPremium	UserPremium(String name, String cc, Calendar date)
	Clase UserRegular	UserRegular(String name, String cc, Calendar date)
R2. Manage bibliographic products: <ul style="list-style-type: none"> <li>• register books and journals.</li> <li>• modify books and journals.</li> <li>• delete books and journals</li> <li>• .</li> </ul>	Clase Main	registerBook() registerMagazine() modifyBook() modifyMagazine() deleteBook() deleteMagazine()
	Clase Controller	registerBook(String id, String nameBP, int numPages, int dayB, int monthB, int yearB, String URL, int acumReadPages, String reviewShort, int genderIndex, double sellingValue, int numCopies)  registerMagazine(String id, String nameBP, int numPages, int dayM, int monthM, int yearM, String URL, int acumReadPages, int categoryIndex, double valueSubscription, String emissionPeriod, int numSubscription)  modifyBook(int optionModify, String idBModify, String nameBPModify, int numPagesModify, int dayBModify, int monthBModify, int yearBModify, String URLModify, int acumReadPagesModify, String reviewShortModify, int genderIndexModify, double sellingValueModify, int numCopiesModify)  modifyMagazine(int optionMModify, String idMModify, String namePBModify, int numPagesModify, int dayMModify, int monthMModify, int yearMModify, String URLModify, int acumReadPagesModify, int categoryIndexModify, double valueSubscriptionModify, String emissionPeriodModify, int numSubscriptionModify)

		deleteBook(String idB)  deleteMagazine(String idM)
	Clase BibliographicProducts	BibliographicProducts(String id, String nameBP, int numPages, Calendar publicateDate, String URL, int acumReadPages)
	Clase Book	Book(String id, String nameBP, int numPages, Calendar publicateDate, String URL, int acumReadPages, String reviewShort, TypeBook gender, double sellingValue, int numCopies)
	Clase Magazine	Magazine(String id, String nameBP, int numPages, Calendar publicateDate, String URL, int acumReadPages, TypeMagazine category, double valueSubscription, String emissionPeriod, int numSuscription)
R3. Automatically generate objects in the system for each type of user and bibliographic product.	Clase Main	System.out.println(controller.getUsers()) System.out.println(controller.getProducts())
	Clase Controller	testCase()
R4. Allow a user to purchase a book or subscribe to a journal.	Clase Main	buyProduct()
	Clase Controller	buyOrSubscribe(String ccUser, String idBP)
	Clase Buyable	buy()
	Clase UserRegular	buyBook(Buyable productToBuyB) addBill(Bill newBill) subscribeMagazine(Buyable productToSubscribeM)
	Clase UserPremium	buyBook(Buyable productToBuyB) subscribeMagazine(Buyable productToSubscribeM)
R5. Allow the user to unsubscribe from a journal	Clase Main	cancelSubs()
	Clase Controller	cancelSubs(String ccUser, String idBP)
	Clase User	cancelSubscription(String idBP); showMagazine();
	Clase UserPremium	cancelSubscription(String idBP) showMagazine()
	Clase UserRegular	cancelSubscription(String idBP) showMagazine()

R6 Allow the user to view his own bookstore	Clase Main	viewLibrary()
	Clase Controller	viewLibrary(String ccUser)
	Clase User Clase UserRegular Clase UserPremium	getProducts()
R7: Allow a user to simulate a reading session (regular and premium).	Clase Main	simulateReadingSession()
	Clase Controller	identifyProduct(String ccUser, int row, int column) cantPages(String ccUser, int row, int column)
R8.Reports generated	Clase Main	reportAcumReadPages() controller.showMostReadGenresAndCategories() sellingBooks() numSubsActiveValueTotal()
	Clase controller	numPagesReadBook() showMostReadGenresAndCategories() sellingBooks() numSubsActiveValueTotal()