|  |  |  |
| --- | --- | --- |
| **Functional requirement** | **Class name** | **Method name** |
| RF0: Register users | Main  Controller  UserFactory  Standar  Premium | main()  executeOption(int option)  validateIntegerInput()  registerUser()  createUser()  addUser()  showUserInfo() |
| RF1: Register Bibliographic Product | Main  Controller  BibliographicProduct  BookFactory  MagazineFactory  Book  Magazine  Utils | main()  executeOption(int option)  validateIntegerInput()  registerBibliographicProduct()  createBook()  createMagazine()  showProductInfoByName()  crateBibliographicProduct()  addProduct()  idGenerator()  reCreateID()  stringToCalendar() |
| RF2: Delete bibliographic product | Main  Controller  BibliographicProduct  AbstractUser  Book  Magazine | main()  executeOption(int option)  validateIntegerInput()  deleteProduct()  showProductWithSimilarName()  findProductByID()  removeProduct() |
| RF3: Modify bibliographic product | Main  Controller  BibliographicProduct  changeProduct  Utils  Book  Magazine | main()  executeOption(int option)  validateIntegerInput()  showProductWithSimilarName()  showProductLabeledAttributes()  displayInformationByProduct()  changeProduct()  getProductByID()  stringToCalendar() |
| RF4: The software must allow the user to acquire a bibliographic product | Main  Controller  BibliographicProduct  AbstractUser  Receipt  Standar  Premium  Magazine  Book | main()  executeOption(int option)  validateIntegerInput()  obtainProduct()  showProductWithSimilarName()  addProductToUser()  checkoutShoppingCart()  getProductByID()  amountOfBooks()  amountOfMagazines  addProductToCar()  isProductInCar() |
| RF5: The software must allow the user to unsubscribe from a magazine at any time | Main  Controller  BibliographicProduct  AbstractUser  Standar  Premium  BibliographicProduct | main()  executeOption(int option)  validateIntegerInput()  unsubscribeMagazine()  showProductWithSimilarName()  getProductByID()  removeProduct() |
| RF6: The software must allow the user to simulate a reading session | Main  Controller  BibliographicProduct  AbstractUser  Magazine  Book  Premium  Standar | main()  executeOption(int option)  validateIntegerInput()  startAReadingSesion()  productsOfAUser()  findProductByID()  getProductByID()  hasProduct()  previousPage()  nextPage()  showPublicity() |
| RF7: The software must allow the user to present the products of a user | Main  Controller  BibliographicProduct  AbstractUser  Magazine  Book  Premium  Standar  Library | main()  executeOption(int option)  validateIntegerInput()  startAReadingSesion()  startAReadingMatrix()  isTheUserTryingToStartAreadingSession()  startMatrixReadingSession()  searchProduct() |
| RF8: Create Reports | Main  Controller  BibliographicProduct  AbstractUser  Magazine  Book  Premium  Standar | main()  executeOption(int option)  validateIntegerInput()  reportMenu()  executeReport()  amountOfPagesByProduct()  genreAndCategoryWithTheMostAmountOfPagesRead()  top5BooksAndTop5MagazinesRead()  salesByGenre()  salesByCategory()  top5books() |