

Library Automation System

Manoj Rapaka 200271025

R. Rajesh 200271026

Emre Geylani 200271007

Msc. Computer Engineering

Çankaya University

Instructor:

Dr. Semih Çetin

19.12.2003

Abstract:

This paper has been written to design a Library Automation System. The related design tools have been listed below:

- CRC Cards
- UML techniques
- Class Diagrams
- Use Cases.

To generate Case Diagrams we have used the software Enterprise Architect ShareWare Version.

CRC CARDS for Library Automation

Class name: Catalog	
Responsibilities:	Collaborations:
<ul style="list-style-type: none">• <u>search()</u>• <u>add()</u>• <u>remove()</u>• <u>catalog()</u>	<ul style="list-style-type: none">• <u>NonText</u> <u>Item</u>• <u>Location</u>• <u>Book</u>• <u>Serial</u>

Class name: NonTextItem	
Responsibilities:	Collaborations:
<ul style="list-style-type: none">• <u>purchase()</u>• <u>shelve()</u>• <u>checkStatus()</u>• <u>repair()</u>• <u>bindSerial()</u>	<ul style="list-style-type: none">• <u>Video</u>• <u>Audio</u>• <u>CD</u>• <u>Tape</u>• <u>Map</u>• <u>Image</u>• <u>Status</u>

Class name: Location	
Responsibilities:	Collaborations:
<ul style="list-style-type: none">• <u>set()</u>• <u>get()</u>• <u>back()</u>	

Class name: Book	
Responsibilities:	Collaborations:
<ul style="list-style-type: none">• <u>purchase()</u>• <u>bindSerial()</u>• <u>shelve()</u>• <u>checkStatus()</u>• <u>repair()</u>	<ul style="list-style-type: none">• <u>Status</u>

Class name: Serial	
Responsibilities:	Collaborations:
<ul style="list-style-type: none">• <u>bind()</u>• <u>remove()</u>	

Class name: Expiration	
Responsibilities:	Collaborations:
<ul style="list-style-type: none">• <u>report()</u>• <u>fine()</u>	<ul style="list-style-type: none">• <u>Date</u>

Class name: Video	
Responsibilities:	Collaborations:

Class name: Audio	
Responsibilities:	Collaborations:

Class name: CD	
Responsibilities:	Collaborations:

Class name: Tape	
Responsibilities:	Collaborations:

Class name: Map	
Responsibilities:	Collaborations:

Class name: Image	
Responsibilities:	Collaborations:

Class name: Customer	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>register()</u> • <u>loanRequest()</u> • <u>unregister()</u> • <u>complaint()</u> 	<ul style="list-style-type: none"> • <u>Person</u>

Class name: Circulation	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>data()</u> 	<ul style="list-style-type: none"> • <u>OverDue Material</u>

Class name: OverDueMaterial	
Responsibilities:	Collaborations:
	<ul style="list-style-type: none"> • <u>Expiration</u>

Class name: ReceiveMaterial	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>record()</u> • <u>shelve()</u> • <u>setDate</u> 	<ul style="list-style-type: none"> • <u>Date</u>

Class name: PurchaseMaterial	
Responsibilities:	Collaborations:
	<ul style="list-style-type: none"> • <u>Receive Material</u> • <u>Account Manager</u>

Class name: ReserveMaterial	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>unShelve()</u> • <u>setReserveExpiration()</u> 	<ul style="list-style-type: none"> • <u>Expiration</u>

Class name: Supplier	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>request()</u> 	<ul style="list-style-type: none"> • Shop • Other Library • Donor

Class name: Shop	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>buy()</u> • <u>search()</u> 	

Class name: OtherLibrary	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>search()</u> 	

Class name: Donor	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>thanks()</u> 	<ul style="list-style-type: none"> • <u>Person</u>

Class name: TelnetSession	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>connect()</u> • <u>disconnect()</u> • <u>gatherInfo()</u> 	<ul style="list-style-type: none"> • Other Library • Customer • Staff

Class name: FtpSession	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>connect()</u> • <u>disconnect()</u> • <u>transfer</u> <u>File()</u> 	<ul style="list-style-type: none"> • Other Library • Customer • Staff

Class name: WebSession	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>signIn()</u> • <u>signOut()</u> • <u>searchInternet()</u> 	<ul style="list-style-type: none"> • Index • Current Content • Abstract

Class name: Index	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>indexSearch()</u> 	

Class name: CurrentContent	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>contentSearch()</u> 	

Class name: Abstract	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>abstractSearch()</u> 	

Class name: BranchLibrary	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>request()</u> 	<ul style="list-style-type: none"> • <u>BranchLibraryCatalog</u> • <u>Courier</u>

Class name: Staff	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>hire()</u> • <u>train()</u> • <u>coordinate()</u> • <u>fire()</u> 	<ul style="list-style-type: none"> • <u>Person</u>

Class name: Transaction	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>checkout()</u> • <u>lossPrevention()</u> • <u>report()</u> 	<ul style="list-style-type: none"> • <u>Catalog</u> • <u>Customer</u> • <u>Date</u>

Class name: OfficeStock	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>repair()</u> • <u>buy()</u> • <u>discard()</u> • <u>clean()</u> 	<ul style="list-style-type: none"> • Furniture • BookShelf • Machinery • Computer

Class name: Machinery	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>maintenance()</u> 	

Class name: BookShelf	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>Paint()</u> 	

Class name: Computer	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>maintenance()</u> • <u>StaffSearch()</u> • <u>CustomerSearch()</u> 	<ul style="list-style-type: none"> • <u>Catalog</u>

Class name: Furniture	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>Paint()</u> 	

Class name: Insurance	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>make()</u> • <u>financial</u> <u>Commitment()</u> 	<ul style="list-style-type: none"> • <u>Account</u> <u>Manager</u>

Class name: Date	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>today()</u> • <u>get()</u> • <u>set()</u> 	

Class name: LoanMaterial	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>setCustomer</u> <u>Info()</u> • <u>setLoan</u> <u>Expiration</u> <u>Date()</u> 	<ul style="list-style-type: none"> • <u>Expiration</u>

Class name: Person	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>remove()</u> • <u>printInfo()</u> 	

Class name: Status	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>setIn()</u> • <u>setOut()</u> • <u>setMiss()</u> • <u>setStolen()</u> 	

Class name: OnlineInformation	
Responsibilities:	Collaborations:
	<ul style="list-style-type: none"> • <u>Telnet</u> <u>Session</u> • <u>FtpSession</u> • <u>WebSession</u> • <u>Catalog</u>

Class name: Chairman	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>inspect()</u> 	<ul style="list-style-type: none"> • <u>Person</u>

Class name: AccountManager	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>payMoney()</u> • <u>receiveMoney()</u> • <u>record()</u> • <u>report()</u> 	<ul style="list-style-type: none"> • <u>Person</u>

Class name: BranchLibraryCatalog	
Responsibilities:	Collaborations:

Class name: Courier	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> • <u>deliver()</u> 	<ul style="list-style-type: none"> • <u>Person</u>

Use Cases for Library Automation

Name: search
Actor: Person
Pre-Conditions: User must be logon or use one of PCs in library
Post-Conditions: Person will be ready to decide which materials he/she need.
Purpose: To give available material information about library stocks
<u>Descripton:</u> <ul style="list-style-type: none">• Actor initiates this use case by selecting “search”• System searches the catalog and displays the related data• Sytem waits another search

Name: PurchaseMaterial
Actor: AccountManager
Pre-Conditions: Staff should be ready to get the material(s)
Post-Conditions: AccountManager pays and records the invoice
Purpose: To develop library stocks
<u>Descripton:</u> <ul style="list-style-type: none">• Actor initiates this use case by selecting “PurchaseMaterial”• The material is ordered and sent by shop couriers• The amount has been paid by Account Manager

Name: checkout
Actor: Customer
Pre-Conditions: Customer should be already a member and material is available
Post-Conditions: Customer loans the material for a period
Purpose: To circulate the material
<u>Descripton:</u> <ul style="list-style-type: none">• Actor initiates this use case by selecting “checkout”• Staff gives the requested material to customer• System changes the status of material

Name: train
Actor: Staff
Pre-Conditions: Person should be hired already
Post-Conditions: Person ready for library services
Purpose: To prepare staff to work efficiently in the library
<u>Descripton:</u> <ul style="list-style-type: none">• Staff initiates this use case by selecting “train”• New staff has been applied to related courses• Staff is ready for library services

Name: printInfo
Actor: Person
Pre-Conditions: Person must have dealings with library
Post-Conditions: All available information about Person has been printed
Purpose: To get detailed information about person
<u>Description:</u> <ul style="list-style-type: none"> • Actor initiates this use case by selecting “printInfo” • System prints out all related information about person

Name: setStolen
Actor: Material
Pre-Conditions: Material is on the shelf
Post-Conditions: Material could not be found
Purpose: To set the materials new status and inform the staff about the loss
<u>Description:</u> <ul style="list-style-type: none"> • Staff initiates this use case by selecting “setStolen” • The status of material has been changed to stolen

Name: register
Actor: Customer
Pre-Conditions: Person is not registered
Post-Conditions: Person is a member of library
Purpose: To register person
<u>Description:</u> <ul style="list-style-type: none"> • Actor initiates this use case by asking staff to “register” • Customer will sign the related papers • Customer is ready to use library services

Name: fine
Actor: Customer
Pre-Conditions: Customer has not returned the material in time
Post-Conditions: Customer has to pay the fine
Purpose: To compensate customer
<u>Description:</u> <ul style="list-style-type: none"> • Staff initiates this use case by selecting “fine” when a compensation occurs • Customer pays the requested money

Name: set
Actor: Date
Pre-Conditions: Wrong date has been used on material
Post-Conditions: Date has been corected for material
Purpose: To set the new date on related material
<u>Descripton:</u> <ul style="list-style-type: none"> • Staff initiates this use case by selecting “set” • New date has been updated

Name: transferFile
Actor: Person
Pre-Conditions: Person should be connected to Ftp Session
Post-Conditions: File transfered to person
Purpose: To transfer requested file
<u>Descripton:</u> <ul style="list-style-type: none"> • Actor initiates this use case by selecting “transferFile” • System checks for requested file • File Transfer completes • Prompts for new command