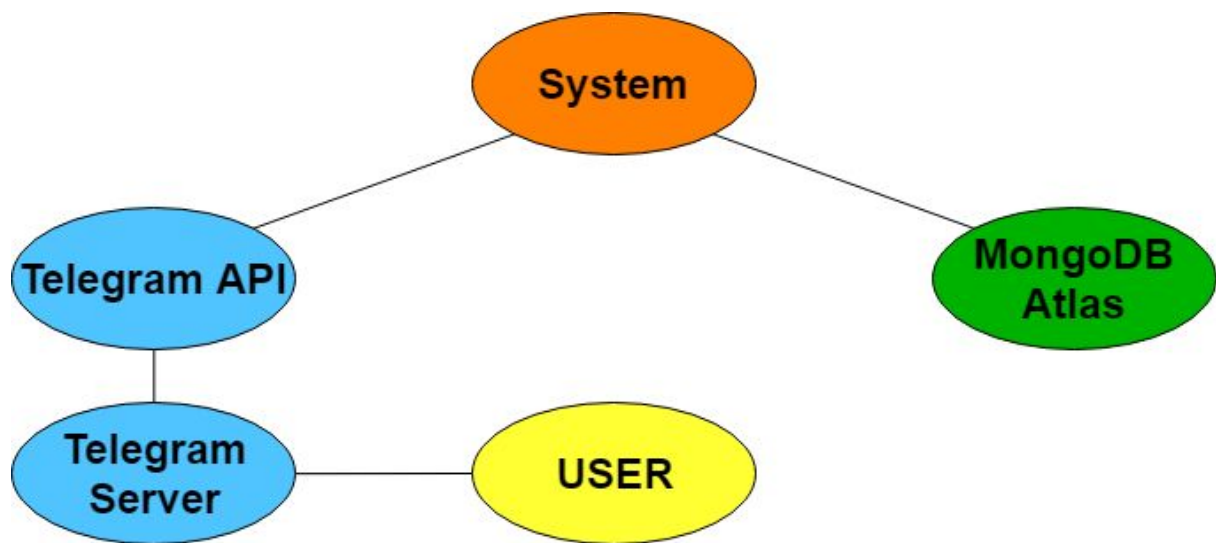


# General information about our project

Our project is telegram bot. This telegram bot is online service, where you can check out document (either Book, Journal, Audio and Video Material). For creating bot we use Telegram API, Telegraph API, GeoDecoder API, and MongoDB as database. As servers we use Telegram servers, and database hosting is MongoDB cloud. GUI is telegram itself. Programming language: Java.



## Github reference:

<https://github.com/temur-kh/InnoLibBot>

## Available Classes

### Classes in folder User

#### *User*

This class represents a User of our Telegram Bot in DataBase.

Classes **Patron** and **Librarian**(see below) are inherited from this class. Class **User** has constructor, getters and setters.

Field	Type	Description
-------	------	-------------

id	long	Identification number in database
name	String	User's first name
surname	String	User's last name
email	String	User's email (..@innopolis.ru)
phoneNumber	String	User's phone number
address	String	User's address
status	Status	To know status of User (Student, Instructor, TA, Professor, VisitingProfessor)

Methods	Access	Return	Description
getInfo ()	public	String	show info about user

### ***Patron***

This class represents a User, who wants checkout document.

This class is inherited from class **User**. Classes **Student** and **Faculty**(see below) are inherited from this class. Class has constructors, getters and setters.

Methods	Access	Return	Description
<u>checkOutDocument</u> (Document)	public	<b>CheckOut</b> (see below)	If document exists in library and it has free copies of it, Patron can check out this book. This method set deadline for returning document back.
getCheckOutTime (Document, String)	public	int	return time, during which you need to return the document
register ()	public	void	register patron

isStudent ()	public	boolean	to know if patron is student (by default is student)
isFaculty ()	public	boolean	to know if patron is faculty
getInfo ()	public	String	show info about patron (Override)

### ***Librarian***

This class represents a User, who has opportunities to add documents to library, see list of user and documents and so on. This class is inherited from class **User**. This class has constructors, getters and setters.

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
setAdmin (User)	public	void	set new admin into the system
addUser (User)	public	void	add new user
modifyUser (long, String, Object, String)	public	void	modify information about user
removeUser (long, String)	public	void	remove user from the system
confirmCheckout (Checkout)	public	void	
addDocument (BasicDBObject, String)	public	void	add new document into the system
addDocument (Document, String)	public	void	add new document into the system
removeDocument (ObjectId, String)	public	void	remove document from the system
updateDocument (BasicDBObject, String)	public	void	update document in the system

updateDocument (Document, String)	public	void	update document in the system
modifyDocument (ObjectId, String, Object, String)	public	void	modify information about document
addCopy (ObjectId, DocAddress)	public	void	add new copy of document
addCopy (Copy)	public	void	add new copy of document
removeCopy (ObjectId)	public	void	remove copy from the system
updateCopy (BasicDBObject)	public	void	update copy
updateCopy (Copy)	public	void	update copy
modifyCopy (ObjectId, String, Object)	public	void	modify information about copy
addIssue (BasicDBObject)	public	void	add new issue
addIssue (Issue)	public	void	add new issue
removeIssue (ObjectId)	public	void	remove issue from the system
updateIssue (BasicDBObject)	public	void	update issue
updateIssue (Issue)	public	void	update issue
modifyIssue (ObjectId, String, Object)	public	void	modify information about the issue
addArticle (BasicDBObject)	public	void	add new article
addArticle (JournalArticle)	public	void	add new article
removeArticle (ObjectId)	public	void	remove article from the system
updateArticle BasicDBObject)	public	void	update article

updateArticle (JournalArticle)	public	void	update article
modifyArticle (ObjectId, String, Object)	public	void	modify information about the article
getCheckOutList ()	public	ArrayList <CheckOut>	show list of checkouts by dates
sendOutstandingRe quest (ObjectId)	public	void	send request for outstanding document
getInfo ()	public	String	show info about librarian (Override)

### ***Student***

This class represents a Student(delay before returning document back is less).This class is inherited from class **Patron**. Has constructor.

### ***Faculty***

This class represents a Faculty(delay before returning document back is bigger). This class is inherited from class **Patron**. Also we have different types of faculty (Instructor, Professor, TA). Has constructor.

### ***VisitingProfessor***

This class represents a Visiting Professor. This class is inherited from class **Patron**. Has constructor.

### ***Status***

Simple enum with status: Student, Instructor, TA, VisitingProfessor, Librarian.

## **Classes in folder Document**

### ***Document***

This class represents Documents, that can be checked out in library. Classes **AVMaterials**, **Book**, **Journal**(see below) are inherited from this class. Has constructors, getters and setters.

Field	Type	Description
-------	------	-------------

id	ObjectId	Identification number in database(ObjectID is used in MongoDB)
title	String	Title of document
url	String	Reference to a document
authors	ArrayList<String>	Names of authors in ArrayList
photoid	String	Reference to a photo of document
price	double	price of document
keywords	ArrayList<String>	Keywords, which help to find document
copyids	ArrayList<ObjectId>	Ids of copies of this document

Methods	Access	Return	Description
<u>hasFreeCopies ()</u>	public	boolean	If library has copy of document, which is not checked out yet, method returns true.
<u>getFreeCopy (boolean)</u>	public	<b>Copy</b> (see below)	Set free copy of document as checked out.
<u>getAuthorsLine ()</u>	public	String	returns line with all authors of document.
<u>getKeywordsLine ()</u>	public	String	returns line with all keywords of document.

## ***Book***

This class represents Books, that can be checked out in library. This class is inherited from class **Document**. Has constructors, getters and setters.

Field	Type	Description
-------	------	-------------

The same as in class Document..		
edition	String	Shows edition of book
bestSeller	boolean	To know if the book is best seller or no(need for deadline)
canBeCheckedOut	boolean	To know if book is not a reference book

### ***Journal***

This class represents Journals(magazines), that can be checked out in library. This class is inherited from class **Document**. Has constructors, getters and setters.

Field	Type	Description
The same as in class Document..		
issuelds	ArrayList<String>	List of issues of the journal
canBeCheckedOut	boolean	To know if book can be checked out

### ***AVMaterials***

This class represents Audio and Video Materials, that can be checked out in library. This class is inherited from class **Document**. Has constructors, getters and setters.

Field	Type	Description
The same as in class Document..		

### ***Copy***

This class represents copies of documents. Has constructors, getters and setters.

Field	Type	Description
id	ObjectId	Identification number in database

docId	ObjectId	to which document this copy belongs to
address	<b>DocAddress</b> (see below)	to know where this copy is
checkedOut	boolean	to know if the copy is free or no.

### ***DocAddress***

This class represents place, where copies are. Has constructors, getters and setters.

Field	Type	Description
room	String	Show room where the copy is
level	String	Show level where the copy is
docCase	String	exact place in room

### ***JournalArticle***

This class represents articles of Journals. They can be collected in **Issue**(see below). Has constructors, getters and setters

Field	Type	Description
id	ObjectId	Identification number in database
title	String	Title of journal article
authors	ArrayList<String>	Names of authors in ArrayList
journalId	ObjectId	Id of journal which this article belongs to.
issueId	ObjectId	Id of issue which this article belongs to.

### ***Issue***

This class represents collections of articles for some publication date. Has constructors, getters and setters.

Field	Type	Description
-------	------	-------------



id	ObjectId	Identification number in database
journalId	ObjectId	Id of journal which this issue belongs to
editors	ArrayList<String>	Names of editors in ArrayList
publicationDate	Calendar	Date of publication
articleIds	ArrayList<String>	Ids of articles which consist in the issue

Methods	Access	Return	Description
addArticleId (String)	public	boolean	add new id for article

## Separated Classes

### *CheckOut*

This class represents objects, which keep all needed information about documents, which are checked out. Has constructors, getters and setters.

Field	Type	Description
id	ObjectId	Identification number in database
fromDate	Calendar	Date of checking out
toDate	Calendar	Deadline
patronId	long	Id of User, who checked out document
docId	ObjectId	Which document has been checked out
collection	String	from which collection
copyId	ObjectId	Which copy of document has been checked out

isRenewed	boolean	to know if checkout is renewed
-----------	---------	--------------------------------

Methods	Access	Return	Description
getInfo ()	public	String	get info about checkout
getInfoForPatron ()	public	String	get info about checkout for patron

### ***Notification***

This class represents object, which keep all needed information in order to send notification to patron. Has constructors, getters and setters.

Field	Type	Description
patronId	long	id of patron
docId	ObjectId	id of document
docType	String	type of document
date	Date	date of notification

Methods	Access	Return	Description
unavailableDocumentText ()	public	String	create text for unavailable document
unavailableDocumentSendMessage ()	public	SendMessage	send message with text to chat with id of patron
expiredNotificationText ()	public	String	create text for expired notification
expiredNotificationSendMessage ()	public	SendMessage	send message with text to chat with id of patron

## **Classes in folder Database**

### ***DatabaseManager***

This class represents managing of database. Has constructors, getters and setters.

Field	Type	Description
instance	DatabaseManager	instance of database manager
mongoClient	MnogoClient	client of MongoDB
LOGTAG	String	used to trace logs

### ***SuperDatabase***

This class represents basic functions of all classes, used in Database.

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
createDBObject (String)	public	ObjectId	Create object in database
insertObject (BasicDBObject, String)	public	void	Insert object into database
getObject (Object, String)	public	Object	get object
getObjectList (String)	public	ArrayList<Object>	Show list of objects
updateObject (BasicDBObject, String)	public	void	Update object
modifyObject (Object, String, Object, String)	public	void	modify object
removeObject (Object, String)	public	void	remove object
toDBObject (Object)	public	BasicDBObject	Transform object into DBObject
toObject (DBObject, String)	public	Object	Transform BasicDBObject into object

### ***PatronDB***

This class represents patron in database type. This class is inherited from class SuperDatabase.

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
InsertPatron (Patron)	public	void	insert patron into database
getPatron (long)	public	Patron	get exact patron with his id
getPatronList ()	public	List<Patron>	show list of patrons in database
updatePatron (Patron)	public	void	update characteristic of patron
remove (long)	public	void	remove patron from database
toObject (DBObject)	public	Patron	transform DBObject to patron

### ***LibrarianDB***

This class represents librarian in database type. This class is inherited from class SuperDatabase.

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
InsertLibrarian (Librarian)	public	void	insert librarian into database
getLibrarian (long)	public	Librarian	get exact librarian with his id

getLibrarianList ()	public	List<Librarian>	show list of librarians in database
updateLibrarian (Librarian)	public	void	update characteristic of librarian
updateLibrarian (BasicDBObject)	public	void	update characteristic of librarian
removeLibrarian (long)	public	void	remove Librarian from database
toObject (DBObject)	public	Librarian	transform DBObject to librarian

### ***JournalDB***

This class represents journal in database type. This class is inherited from class SuperDatabase.

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
createJournal (journal)	public	ObjectId	create journal in database
getJournal (String)	public	Journal	show journal from database
getJournalList ()	public	List<Journal>	show list of journals in database
insertJournal (Journal)	public	void	insert journal into database
updateJournal (Journal)	public	void	update characteristics of journal
removeJournal (ObjectId)	public	void	remove journal from database using id
toObject (DBObject)	public	Journal	transform DBObject to journal

### ***JournalArticleDB***

This class represents journal article in database type. This class is inherited from class SuperDatabase.

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
createArticleI ()	public	ObjectId	create article in database
getArticle (String)	public	JournalArticle	show journal article from database
getArticleList ()	public	List<JournalArticle>	show list of articles in database
insertArticle (JournalArticle)	public	void	insert journal article into database
updateArticle (JournalArticle)	public	void	update characteristics of article
removeArticle (ObjectId)	public	void	remove article from database using id
toObject (DBObject)	public	JournalArticle	transform DBObject to JournalArticle

### ***IssueDB***

This class represents issue in database type. This class is inherited from class SuperDatabase.

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
createIssue ()	public	ObjectId	create issue in database
getIssue (String)	public	Issue	show issue from

			database
getIssueList ()	public	List<Issue>	show list of issues in database
insertIssue (Issue)	public	void	insert Issue into database
updateIssue (Issue)	public	void	update characteristics of issue
removeIssue (ObjectId)	public	void	remove Issue from database using id
toObject (DBObject)	public	Issue	transform DBObject to Issue

### ***CopyDB***

This class represents copy in database type. This class is inherited from class SuperDatabase.

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
createCopy ()	public	ObjectId	create copy in database
getCopy (String)	public	Copy	show copy from database
getCopyList ()	public	List<Copy>	show list of copies in database
insertCopy (Copy)	public	void	insert copy into database
updateCopy (Copy)	public	void	update characteristics of copy
removeCopy (ObjectId)	public	void	remove copy from database using id
toObject (DBObject)	public	Copy	transform DBObject to copy

### ***BookDB***

This class represents book in database type. This class is inherited from class SuperDatabase.

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
createBook ()	public	ObjectId	create book in database
getBook (String)	public	Book	show book from database
getBookList ()	public	ArrayList<Book>	show list of books in database
insertBook (Book)	public	void	insert book into database
updateBook (Book)	public	void	update characteristics of book
modifyBook (ObjectId, String, Object)	public	void	modify book
removeBook (ObjectId)	public	void	remove book from database using id
toObject (DBObject)	public	Book	transform DBObject to book

### ***AVMaterialDB***

This class represents audio and video materials in database type. This class is inherited from class SuperDatabase.

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
createAVMaterial ()	public	ObjectId	create AVMaterial in database



getAVMaterial (String)	public	AVMaterial	show AVMaterial from database
getAVMaterialList ()	public	List <AVMaterial>	show list of AVMaterials in database
insertAVMaterial (AVMaterial)	public	void	insert AVMaterial into database
updateAVMaterial (AVMaterial)	public	void	update characteristics of AVMaterial
modifyAVMaterial (ObjectId, String, Object)	public	void	modify AVMaterial
removeAVMaterial (ObjectId)	public	void	remove AVMaterial from database using id
toObject (DBObject)	public	AVMaterial	transform DBObject to AVMaterial

### ***CheckoutDB***

This class represents checkout in database type. This class is inherited from class SuperDatabase.

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
createCheckout ()	public	ObjectId	create Checkout database
getCheckout (ObjectId)	public	Checkout	show Checkout from database
getCheckOut (long, ObjectId)	public	CheckOut	show Checkout from database
getCheckOut (BasicDBObject)	public	CheckOut	show Checkout from database
getAllCheckoutList ()	public	ArrayList <Checkout >	show list of all Checkouts in database

getCheckoutList ()	public	ArrayList <Checkout >	show list of Checkouts in database
getOverdueCheckoutList ()	public	ArrayList <Checkout >	show list of overdue Checkouts in database
getCheckOutListByUserId (long)	public	ArrayList <CheckOut>	show list of Checkouts in database by id of patron
getOverdueCheckOutListByUserId (long)	public	ArrayList <CheckOut>	show list of overdue Checkouts in database by id of patron
getCheckOutListByCursor (DBCursor)	public	ArrayList <CheckOut>	show list of checkouts in database by cursor
insertCheckout (Checkout )	public	void	insert Checkout into database
updateCheckout (Checkout )	public	void	update characteristics of Checkout
removeCheckout (ObjectId)	public	void	remove Checkout from database using id
toObject (DBObject)	public	Checkout	transform DBObject to Checkout

### ***RegistrationStateDB***

This class represents state which User stopped on (while registration).

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
insertNewState (long)	public	void	insert new state of registration
getState (long)	public	BasicDBObject	show state using id
getValue (long,	public	Object	get value from state

String)			(where stopped)
getStateList ()	public	ArrayList <BasicDBObject>	show list of states
dropStateCollection (User)	public	void	delete state of exact user
modifyState (BasicDBObject, Integer)	public	void	modify (continue) state
removeState (long)	public	void	delete state using id

### ***PriorityQueueDB***

This class represents queue with priority of patrons.

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
insert (Patron, ObjectId, String)	public	void	insert patron into queue
contains (Long, ObjectId)	public	boolean	to know if queue contains exact patron
getNextPatron (ObjectId, boolean)	public	Patron	get next patron
getQueue (ObjectId)	public	ArrayList <Patron>	show queue in ArrayList
getNotifications (ObjectId)	public	ArrayList <Notification>	show Notifications in ArrayList
removeQueue (ObjectId)	public	void	remove queue
removePatron (Long, ObjectId)	public	void	remove patron from queue

### ***NotificationDB***

This class represents queue with priority of patrons.

Field	Type	Description
-------	------	-------------

LOGTAG	String	used to trace logs
--------	--------	--------------------

Methods	Access	Return	Description
insert (long, Objectld, String)	public	void	insert new notification
insert (Notification)	public	void	insert new notification
getExpiredNotification ()	public	ArrayList <Notification>	show list of expired notification
toObject (DBObject)	public	Notification	transform DBObject to notification

### ***DBObjectCreator***

This class transform different types to BasicDBObject

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
toDocumentDBObject (Document)	public	BasicDBObject	transform Document to BasicDBObject
toUserDBObject (User)	public	BasicDBObject	transform User to BasicDBObject
toCopyDBObject (Copy)	public	BasicDBObject	transform Copy to BasicDBObject
toIssueDBObject (Issue)	public	BasicDBObject	transform Issue to BasicDBObject
toArticleDBObject (JournalArticle)	public	BasicDBObject	transform JournalArticle to BasicDBObject
toCheckOutDBObject (CheckOut)	public	BasicDBObject	transform CheckOut to BasicDBObject
toPriorityQueueDBObject (Patron, Objectld, String)	public	BasicDBObject	transform PriorityQueue to DBObject
toNotificationDBObject	public	BasicDBObject	transform

ect (Notification)			Notification to DBObject
--------------------	--	--	--------------------------

## Classes in folder Services

### ***DateTime***

Transform String line (date) into calendar type, and vice versa.

Methods	Access	Return	Description
createCalendarObject (String)	public	Calendar	transform String line (date) into calendar type
todayCalendar ()	public	Calendar	today date in Calendar type
todayDate ()	public	Date	today date in Date type
now ()	public	Date	
tomorrowDate ()	public	Date	tomorrows date in Date type
createCalendarLine (Calendar)	public	String	transform calendar (date) into String type
convertToCalendar (Date)	public	Calendar	transform Date into Calendar
convertToDate (Calendar)	public	Date	transform Calendar into Date
daysAddedDate (Date, int)	public	Date	Date plus exact number of days
daysUntilToday (Calendar)	public	int	to know days until exact day

## ***LocationDecoder***

This class used to help with address of users (opportunity to use geolocation while registration). Has constructor.

Field	Type	Description
instance	LocationDecoder	instance of location decoder
api	GeoApiContext	api of geocontext
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
getInstance ()	public	LocationDecoder	instance of <i>LOcation Decoder</i>
getAddress (Float, Float)	public	String	Find address by longitude and latitude
convertToDate (Calendar)	public	Date	transform calendar type to date

## ***PageCreator***

This class is used to create pages.

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
createDocumentContent (Document)	public	ArrayList<Node>	create page of document
createBookPage (Book)	public	String	create page of book
createAVMaterialPage (AVMaterial)	public	String	create page for audio and video materials
createJournalPage (Journal)	public	String	create page for journal

## ***SendMail***

This class is used to send emails.

Methods	Access	Return	Description
sendMail(String, String, String)	public	void	send TEXT, to exact ADDRESS

## ***TelegraphAccount***

This class represents usage of telegraph (while looking through book)

Field	Type	Description
instance	TelegraphAccount	instance of <i>Telegraph Account</i>
api	GeoApiContext	api of <i>Telegraph Account</i>
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
createAccount ()	public	void	create account
getInstance ()	public	TelegraphAccount	create instance of <i>Telegraph Account</i>

**BotConfig, Commands, Constants, Texts are classes with constant variables.**

## **Classes in folder UpdateHandler**

### ***BookingSystem***

This class represents booking system. Using it we can check out documents and receive message with information about deadline and this document.

Methods	Access	Return	Description
checkOut (Update, String, String)	public	SendMessage	user checks out document, and he should receive message (deadline, name of document)

### ***DocumentViewSystem***

This class represents system, where we can see list of accessed documents.

Field	Type	Description
LOGTAG	String	used to trace logs
commandList	ArrayList <String>	list of commands

Methods	Access	Return	Description
belongTo (String)	public	boolean	to know if command belong to this class
convertToDocument (ArrayList<Object>)	public	ArrayList <Document>	transform list of objects to list of documents
handle (Update)	public	SendMessage	execute a page-message to go through list of documents
goToPage (Update, int, String)	public	EditMessageText	go to the next document
getInlineKeyboard (Document, ind, int, String)	public	InlineKeyboardMark up	return inline keyboard markup for page-message

### ***GUISystem***

This class represents graphical user interface.

Methods	Access	Return	Description
initialGreetingView (Update)	public	ArrayList<Object>	return initial greeting for user
getInitialMenu (long)	private	ReplyKeyboardMark up	return main menu keyboard markup
documentsView (Update)	public	SendMessage	return document view and then switch to DocumentViewSystem
getDocumentViewMenu ()	private	ReplyKeyboardMark up	return keyboard markup for document view
profileView (Update)	public	SendMessage	profile view



getProfileViewMenu ()	private	ReplyKeyboardMark up	show menu of profile view
simpleMenu ()	public	ReplyKeyboardMark up	return simple menu keyboard markup containing "Back to menu" button
simpleInlineMenu ()	public	InlineKeyboardMark up	return simple menu inline keyboard markup
backToInitialMenu (Update)	public	SendMessage	go to main menu

### ***LibrarianSystem***

This class represents system, which use librarian.

Field	Type	Description
LOGTAG	String	used to trace logs
commandList	ArrayList <String>	list of commands

Methods	Access	Return	Description
belongTo (String)	public	boolean	to know if command belong to this class
librarianMenu ()	public	ReplyKeyboardMark up	go to librarian menu
handle (Update)	public	SendMessage	execute a page-message
goToPage (Update, int, String)	public	Object	update page-message (go to next checkout)
getCheckoutInlineKeyboard (Checkout, int, int)	public	InlineKeyboardMark up	get inline keyboard for checkout

### ***MainBot***

This class represents part of bot, where we control everything

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
onUpdateReceived (Update)	public	void	receives an object of class Update
sensNotification ()	public	void	
handleUpdate (Update)	public	void	handles received update to execute proper message
getCallbackQueryKey (String)	public	String	return a key part of callback query data
executeMessage (Object)	public	void	execute message
executeMessages (ArrayList <SendMessage>)	public	void	execute messages in list
executeAll (ArrayList <Object>)	private	void	execute all objects in list
getCallbackQueryCollection (String)	public	String	return a collection part of callback query data
getCallbackQueryValue (String)	public	String	return a value part of callback query data
getBotUsername ()	public	String	show username of bot from final variables
getBotToken ()	public	String	show token of bot from final variables

### ***UserProfileSystem***

This class represents information about users, who are already registered.

Field	Type	Description
commandList	ArrayList <String>	list of commands

Methods	Access	Return	Description
belongTo (String)	public	boolean	to know if command belong to this class

handle (Update)	public	SendMessage	execute registration
isEmailCorrect (String)	public	boolean	to know if email is correct (@innopolis.ru)
getSecurePassword ()	public	String	generate random password
personalDataView (Update)	public	SendMessage	return personal data and then switch to PersonalDataSystem

### ***ReturnSystem***

This class represents system, where user can return document, and receive message (Success)

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
handle (Update, String, String)	public	ArrayList<SendMessage>	execute returning of document
returnDocument (Update, String)	public	ArrayList<SendMessage>	return document
receiveReturnRequest (Long, Long, CheckOut)	private	SendMessage	receive message with return request
receiveReturnRequestKeyboard (CheckOut)	private	InlineKeyboardMarkup	receive keyboard for returning request
sendPatronReturnRequest (long, CheckOut)	private	SendMessage	send message with return request

### ***NotificationSystem***

This class is created in order to notify patrons.

Methods	Access	Return	Description
notifyPatron (Long,	public	ArrayList	notify patron

ObjectId, String)		<SendMessage>	
-------------------	--	---------------	--

### ***FineSystem***

This class is created in order to notify patrons if they did not return document on time.

Methods	Access	Return	Description
payForDocument (Update, String)	public	SendMessage	if patron did not return document on time, he should pay for it

### ***PatronSystem***

Field	Type	Description
LOGTAG	String	used to trace logs

Methods	Access	Return	Description
belongTo (String)	public	boolean	if command belong to this class
handle (Update)	public	SendMessage	execute patron system
goToPage (Update, int, String)	public	Object	go to page
getMyCheckOutInlineKeyboard (CheckOut, int, int, String)	private	InlineKeyboardMarkup	get inline keyboard of checkout

### ***RenewSystem***

This class represent system where you can renew checkouts.

Methods	Access	Return	Description
renewDocument (Update, String)	public	ArrayList <SendMessage>	renew document
handle (CheckOut)	public	boolean	execute renew system
requestReturnKeyboard	private	InlineKeyboardMarkup	show keyboard of

ard (CheckOut)		up	requesting return
----------------	--	----	-------------------

## Borrowed classes:

### ***CustomTimerTask***

Author Ruben Bermudez

Version 2.0

Data 28/01/15

Abstract class create tasks. (need for notifications)

### ***TimeExecutor***

Author Ruben Bermudez

Version 2.0

Data 28/01/15

This class is used in order to create thread and execute tasks there in exact time. (need for notification)