REQUIREMENTS DOCUMENT

PKFIELIBRARY MANAGER

Author Zeufack Vannel

School PKFokam Institute of Excellence

Supervisors Mr. Mekontso Hermann and Mrs. Rochelle De Pacio

Opening Date: 26/11/2016

Library picture

I- General Presentation of the project

PKFLibrary is a modern, spacious and well equipped library. It contains a collection of about 2000 books related to various domains such as computer science, economics, mathematics, physics, literature, national and international magazine and others. The library is opened every day from 8:00 AM to 6:00 PM and offers loan services to students and teachers. But since its creation, the librarian has been using raw tools as Microsoft Word, Microsoft Excel or even raw paper to manage the library. Being in the age of new and fast technologies, the PKFIELibrary MANAGER aims at moving up the library from the old ages to the new ages.

II- Objectives

PKFIELibrary MANAGER aims at making the librarian work easier and more efficient. Indeed, the platform will ease the organization of the library stock, ease the research of documents, and ease the follow up of the state of books and also the follow up of any borrowed book. More, the platform will automatically perform some statistics such as the list of most borrowed books and therefore ease the reports of the librarian.

III- Actors

- Administrator
- Managers
- Students

IV- Expression of needs

1) Functional requirements

a) Login (student)

The login page is the first page of the platform. It permits every user to identify himself either as an administrator, a manager or a student. Each user logs in with an identifier and a password.

b) Dashboard (student)

The dashboard is the welcome page accessible to every user. It presents some important statistics like the percentage of books borrowed and the percentage of people in red. It also shows some recent activities.

c) Book management

- Book list (student): permits to have the list of all the books of the library
- Insert a book (manager): a book has a title, an author, a release date, one or more course in which it can be used, one or more categories in which it is classified, an ISBN and a description. Some documents can be borrowed or not. Those which cannot be borrowed must be consulted at the library. A book can be bought by the school or given to the school. Each book also has a state (good, medium, bad, can be lost). A book also has an editor.
- Modify a book (manager)
- Delete a book (manager)
- Category management (manager): add, modify, delete a category

- Create a course (manager): a course has a title and a description

d) Research (student): research is made by title, author, category
   - Research mode: global, by title, by author, by editor, by date, by category, by user
   - Keep history of researches by mode
   - Research a user

e) Borrow/return (student): a student can borrow at most 3 books once at a time. A reader can reserve at most 3 books. The delay for returning a book varies between 3 and 7 days considering the type of document. If the document has not been reserved, the delay is automatically renewed once at due date. In case of reservation, the document must be returned at the end of the ongoing period.
   At the end of the period, a claiming is sent to the reader. If the document is not returned or if the reader did not requested for a prolongation after 10 days, a penalty is attributed to the borrower. The penalty is multiplied by the number of days the borrower made with the book. After some time, the borrower can be sue and his case will be sent to the academic officer.

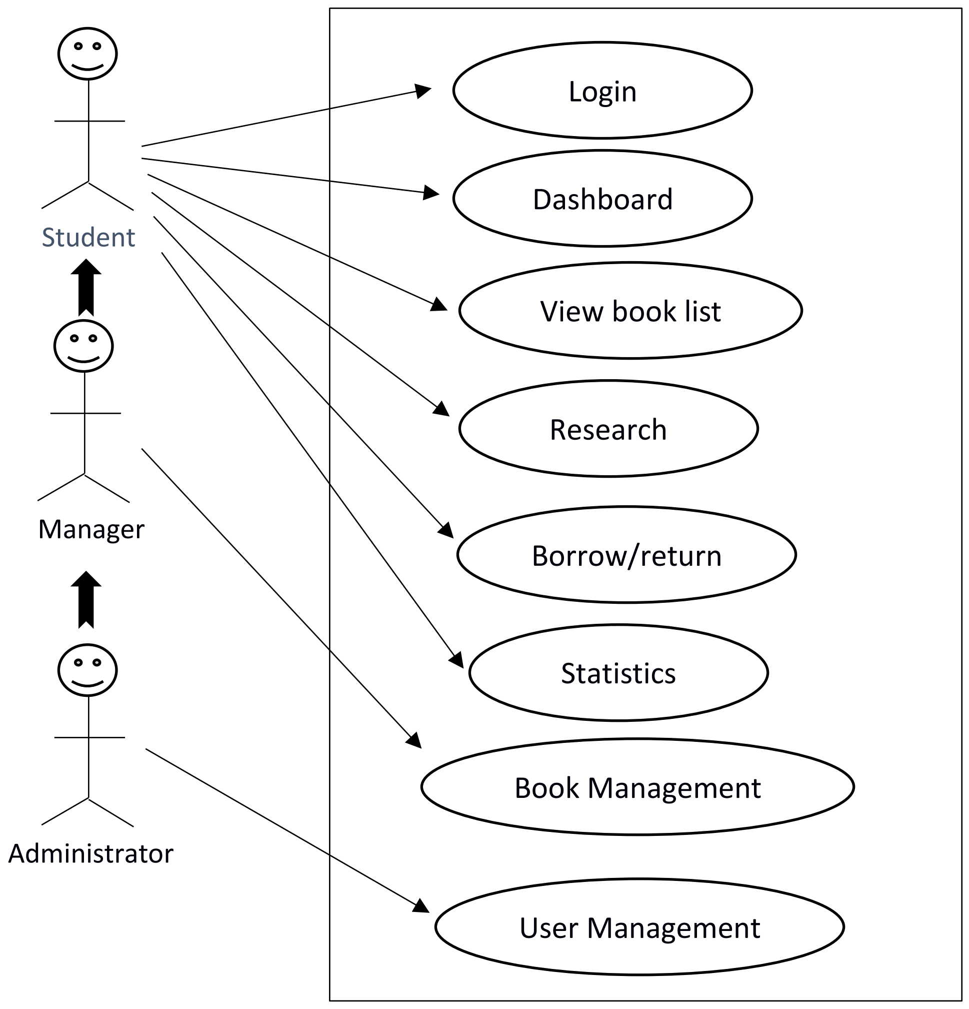
f) Statistics (manager)
   - Number of references (global and by document type)
   - Most borrowed documents
   - Most entered keywords on the research tab
   - Number of books lost
   - Number of loans by group and by year
   - Debt list

g) User Management (administrator)
   - Show the user list: permits to see user list either in full or filtered by some criteria like the role or the name.
   - Add a user: permits to add a user
   - Modify a user: permits to modify a user
   - Delete a user: permits to remove a user
   - Forced disconnection of a user: permits the administrator to disconnect a user
   - Automatic disconnection of a user: permits the system to disconnect a user after some period of inactivity

V) Constraints

a) Organization: documents must be classified in 10 subgroups each containing 10 subgroups (See DEWER Classification). Subgroups can be modified or created.
b) Interface:
   - It must be simple
   - Each screen must show in which mode the actual user is: dashboard, user management, borrow/return etc.
   - Windows' superposition must be avoided except for administrators and managers.
c) About one hundred users must be able to connect to the library platform
d) The platform will be hosted by the PKF server.

VI) Use case Diagram



VI) Detailed description of use cases

1) Login

Login	**Summary** : Permits to authenticate a user and give him access to the application	
Main actors : any user in the system	**Secondary actors :**	
Pre-conditions : • User exists and his status is authorized		
Trigger: User request for a connection		
Data entered :		

- User ID
- User password

Invariants : User ID

Main success scenario

1. User starts the application
2. The system shows the login window
3. User enters his user id and password in the fields provided
4. User validates to create the connection
5. The system checks the validity of the combination user ID/password (E1) and the status of the user (E2) and if the user is not yet connected (E3)
6. The system opens the application and notifies the user of success of the connection by providing the date of the last connection

Alternatives :

Exceptions :

<u>E1 : the combination user ID/password is not valid</u>

Occurs at step 5 of the main success scenario. The system notifies the user and the system goes back to step 2 of the main success scenario.

After 3 unsuccessful trials, the system locks the user account if the account exists and notify the user. User must then contact the administrator.

<u>E2 : the status of the user account does not permit connection</u>

Occurs at step 5 of the main success scenario. The system notifies the user and the system goes back to step 2 of the main success scenario.

User must contact the administrator.

<u>E3 : User is already connected</u>

Occurs at step 5 of the main success scenario. The system notifies the user and the system goes back to step 2 of the main success scenario.

User must first disconnect himself from the already connected session or request a forced disconnection from the administrator.

Output Data

- Identification of the user
- Connected terminal
- Connection time

End : User connected

Post conditions :

• Registration added to the table of users connected.

2) Dashboard

Dashboard	**Summary** : Welcome page
Main actors : any user in the system	**Secondary actors :**
Pre-conditions : • User is authenticated	
Trigger : User authentication	
Input Data:	
Invariants :	
Main success scenario : 1. The user successfully logs in 2. The system shows the dashboard	
Alternatives : **1.** Being already connected, the user clicks on the dashboard menu button	
Exceptions :	
Output data : • User name • User role • User profile • Some statistics • Some recent activities	
End : Dashboard is shown	
Post conditions : • Dashboard is shown	

3) User management

a) Create user

Create user	**Summary** : creates a user in the application
Main actors : Administrators	**Secondary actors :**
Pre-conditions : • User profile must be defined and validated by the qualified authorities	
Trigger : Request for user creation	
Input Data • User ID • User password • User profile (name, phone number, address, email) • User role (reader, manager or administrator) • User status (hierarchical position in the school)	
Invariants : User ID	
Main success scenario : 1. Administrator clicks on the user creation menu ; 2. The system empties the form for creating a user ; 3. The admin enters information about the user and requests registration from the system 4. The system checks if the user ID entered already exist (E1) or if the passwords are conform (E2) 5. The system registers the user and notifies the administrator	
Alternatives : **1.** The administrator researches a user by entering some research criteria. He then selects a user in the resultant list. He clicks on the duplication button. The user creation window opens with the information of the user without the ID. The administrator can then modify to create a new user. The system restarts at step 4 of the main success scenario	
Exceptions : <u>E1 : The User ID already exists</u> Occurs at step 4 of the main success scenario. The system notifies the user and the system goes back to step 3 of the main success scenario. <u>E2 : The password does not respect established password constraints</u> Occurs at step 4 of the main success scenario. The system notifies the user and the system goes back to step 3 of the main success scenario.	

Output data : • User ID • User password • User profile • User role • User status
End : User created and authorized
Post conditions : • User registered with the date of registration • User authorized • User status initialized • Indicator for changing password activated

b) <u>Modify a user</u>

Modify a user	**Summary** : modify a user’s information in the application
Main actors : Administrators	**Secondary actors :**
Pre-conditions : • Request for a user modification from a user or an authority	
Trigger : Request for user modification	
Input Data • User ID • User password • User profile • User role • User status	
Invariants : User ID	
Main success scenario : 1. The Administrator shows off the user list 2. The Administrator selects the user to modify 3. The Administrator modifies the information and requests the system for registration 4. The system checks if the user ID entered already exist (E1) or if the passwords are	

conform (E2)
Alternatives : 1. The Admin researches a user by entering some research criteria. He then selects a user in the resultant list. He continues from step 3 of the main success scenario
Exceptions : E1 : The User ID already exists Occurs at step 4 of the main success scenario. The system notifies the user and the system goes back to step 3 of the main success scenario. E2 : The password does not respect established password constraints Occurs at step 4 of the main success scenario. The system notifies the user and the system goes back to step 3 of the main success scenario.
Output Data : • User ID • User password • User profile • User role • User status
End : User modified and authorized
Post conditions : • User modified • User role initialized if the role has been changed • Change password indicator initialized if the password has been changed

c) Delete user

Delete a user	**Summary** : remove a user in the application
Main actors : Administrators	**Secondary actors :**
Pre-conditions : • Request for a user removal from a user or an authority • The user must exists in the database	
Trigger : Request for user removal	
Input Data	
Invariants :	
Main success scenario :	

1. The Administrator shows off the user list 2. The Administrator selects the user to remove 3. The Administrator clicks on the button to remove a user and the system removes the user 4. The system registers the deletion and notifies the administrator
Alternatives : 1. The Admin researches a user by entering some research criteria. He then selects a user in the resultant list. He continues from step 3 of the main success scenario
Exceptions :
Output Data :
End : User deleted
Post conditions : • Deletion registered

4) Book management

a) Insert a book

Insert a book	**Summary** : Insert a book in the application
Main actors : Manager	**Secondary actors :** Administrator
Pre-conditions : • The book must be given to the school library as a gift or bought	
Trigger : Request for book insertion	
Input Data • Book title • Book author • Date • Book editor • Course list in which the book can be used • Book category • Book ISBN	

- Book description

Invariants : Book ISBN

Main success scenario :

1. Manager clicks on the book insertion menu ;
2. The system empties the form for inserting a book ;
3. The manager enters information about the book and requests registration from the system
4. The system checks if the book ISBN entered already exists (E1)
5. The system registers the book and notifies the manager

Alternatives :

1. The manager researches a book by entering some research criteria. He then selects a book in the resultant list. He clicks on the duplication button. The book insertion window opens with the information of the book without the book's ISBN. The manager can then modify to insert a new book. The system restarts at step 4 of the main success scenario

Exceptions :

E1 : The Book ISBN already exists

Occurs at step 4 of the main success scenario. The system notifies the user and the system goes back to step 3 of the main success scenario

Output data :

- Book title
- Book author
- Date
- Book editor
- Course list in which the book can be used
- Book category
- Book ISBN
- Book description

End : The book is inserted

Post conditions :

- The book is registered

b) Modify a book

Modify a book	**Summary** : Modify a book in the application
Main actors : Managers	**Secondary actors :** Administrators
Pre-conditions : • The book must exist the book database	
Trigger : Request for book modification	
Input Data • Book title • Book author • Date • Book editor • Course list in which the book can be used • Book category • Book ISBN • Book description	
Invariants : Book ISBN	
Main success scenario : 1. Manager clicks on the button to show off the book list ; 2. He then select the book to modify 3. The manager modifies information about the book and requests registration from the system 4. The system checks if the book ISBN entered already exists (E1) 5. The system registers the modification and notifies the manager	
Alternatives : **1.** The manager researches a book by entering some research criteria. He then selects a book in the resultant list. He continues from step 3 of the main success scenario	
Exceptions : E1 : The Book ISBN already exists Occurs at step 4 of the main success scenario. The system notifies the user and the system goes back to step 3 of the main success scenario	
Output data :	

• Book title • Book author • Date • Book editor • Course list in which the book can be used • Book category • Book ISBN • Book description
End : The book is modified
Post conditions : • Modification is registered

c) Delete a book

Delete a book	**Summary** : Delete a book from the application
Main actors : Manager	**Secondary actors :** Administrator
Pre-conditions : • The book must exist the book database	
Trigger : Request for book removal	
Input Data	
Invariants :	
Main success scenario : 1. Manager clicks on the button to show off the book list ; 2. He then selects the book to remove 3. The manager clicks on the button to remove a book and the system removes the book 4. The system registers the deletion and notifies the manager	
Alternatives : **1.** The manager researches a book by entering some research criteria. He then selects a book in the resultant list. He continues from step 3 of the main success scenario	
Exceptions :	
Output data :	

End : The book is deleted
Post conditions : • Deletion is registered

d) <u>Category management</u>

Add a category	**Summary** : Permits to add a book category
Main actors : Manager	**Secondary actors :** Administrator
Pre-conditions :	
Trigger : Request for category creation	
Input Data • Category name • Category description	
Invariants : category name	
Main success scenario : 1. User clicks on the category add menu ; 2. The system empties to form for creating a category 3. The user enters the category name and description 4. The system checks if the category name already exists (E1) 5. The system registers the category and notifies the user	
Alternatives :	
Exceptions : <u>E1: The category name already exits</u> Occurs at step 4 of the main success scenario. The system notifies the user and goes back to step 3 of the main success scenario	
Output data : • Category name • Category description	
End : The category is added	
Post conditions : • Category is registered	

Modify a category	**Summary** : Permits to modify a book category
Main actors : Manager	**Secondary actors :** Administrator
Pre-conditions :	
Trigger : Request for category modification	
Input Data • Category name • Category description	
Invariants : category name	
Main success scenario : 1. User clicks on the category management menu; 2. The system shows the list of categories 3. The user selects the category to modify 4. The user modifies the category information 5. The system checks if the category name already exists (E1) 6. The system registers the modification and notifies the user	
Alternatives : 1. The user goes to research tab and research the category to modify. He then selects the category and the system continues from step 4 of the main success scenario	
Exceptions : <u>E1: The category name already exits</u> Occurs at step 5 of the main success scenario. The system notifies the user and goes back to step 4 of the main success scenario	
Output data : • Category name • Category description	
End : The category is modified	
Post conditions : • Modification is registered	

Delete a category	**Summary** : Permits to delete a book category
Main actors : Manager	**Secondary actors :** Administrator

Pre-conditions :
Trigger : Request for category removal
Input Data
Invariants :
Main success scenario : 1. User clicks on the category menu ; 2. The system shows off the list of categories 3. The user selects the category to delete and request the system to delete 4. The system shows a confirmation message 5. The system deletes the category 6. The system registers the deletion and notifies the user
Alternatives :
Exceptions :
Output data :
End : The category is deleted
Post conditions : • Deletion is registered • Every book in the category will be classified as non-categorized

5) <u>Research</u>

a) <u>Research a book</u>

Research a book	**Summary** : Permits to find a book or any book information in the application
Main actors : Any user in the platform	**Secondary actors :**
Pre-conditions : • The book is registered	
Trigger : Request for a book research	
Input Data • Book title	

- Book author
- Date
- Book editor
- Course list in which the book can be used
- Book category
- Book ISBN
- Book description

Invariants :

Main success scenario :

7. User clicks on the book research menu ;
8. The system shows the full list of books classified by category by default, the user can sort the list by title, by ISBN, by course, by editor, by author ;
9. The user selects some research criteria, enter his keyword in the space provided and validate the research
10. The system shows the result of the research

Alternatives :

Exceptions :

Output data :

- Book title
- Book author
- Date
- Book editor
- Course list in which the book can be used
- Book category
- Book ISBN
- Book description

End : The research result is shown

Post conditions :

- Search parameters are registered

b) Research a user

Research a user	**Summary** : Permits to find a user or any user information in the application	
Main actors : Administrator		**Secondary actors :** manager
Pre-conditions : • The user is registered and authenticated		
Trigger : Request for a user research		
Input Data • User ID • User password • User profile • User role • User status		
Invariants :		
Main success scenario : 1. User clicks on the user research menu ; 2. The system shows the full list of users classified by role by default, the user can sort the list by id, by status or by name ; 3. The user selects some research criteria, enter his keyword in the space provided and validate the research 4. The system shows the result of the research		
Alternatives :		
Exceptions :		
Output data : • User ID • User password • User profile • User role • User status		
End : The research result is shown		

Post conditions :

- Search parameters are registered