**Vision Document for “eLibrary“**

**Team members**

|  |  |
| --- | --- |
| **Name** | **ID** |
| **Arsene Dumont Nguezang Nsaha** | 987030 |
| J**orge Pinto** | 987123 |
| **Medor Rodrigue** | 986732 |

Link to GitHub account: <https://github.com/japinto930614/SWELibrary>

**1. Introduction**

Assuming we have been hired by Maharishi University of Management (MUM) to update their library record keeping. All the library member information and book check-in and checkout information, however, is kept on paper. Information about the books was all contained in ledgers. This system was previously workable, because MUM had only a few hundred students enrolled. Due to the increasing enrollment, the library now needs to automate the check-in/checkout system. The new system will have a graphic user interface to allow librarians to check-in and checkout books. All books in the library have will have a unique *bookid*. The books in the library are ordered on the shelves by their *bookid*. The new system should allow library members to search through the library database to find the *bookid* of the desired book. The system will run on several individual desktops throughout the library. Librarians will have their own desktop computers that are not accessible by library members. Only librarians can check-in and checkout books.

The system will retain information on all library members. Only university students, faculty and staff can become library members. Students can check-out books for a maximum of 21 days. If a student returns a book later than 21 days, then he/she must pay an overdue fee of 25 cents per day. University staff can also check out books for a maximum of 21 days but pay an overdue fee of 10 cents per day. Faculty can check out books for a maximum of 100 days and pay only 5 cents per day for every book returned late. The system will keep track of the amount of money that library members owe the library.

**2. Positioning**

**2.1 Problem Statement**

|  |  |
| --- | --- |
| The problem of | Managing the checking-in and checking-out of the increasing number of books in the library. |
| Affects | Librarians, faculty, staff and students |
| the impact of which is | The whole process of book warehousing, checking-in and checking out will be automated and easier to manage |
| a successful solution would be | Building a web-based system that reflects the business model of the library while improving on the management of books and members. |

**2.2 Product Position Statement**

|  |  |
| --- | --- |
| For | The University (MUM) Library |
| Who | Student, staff, Faculty |
| The (product name) | **eLibrary** |
| That | can easily manage the library members and book stored within the library using a web-based information system |
| Unlike | The manual paper-based system |
| Our product | Is fully customized, easy to use, scalable and can evolve with changing business needs, |

**3. Stakeholder Descriptions**

**3.1 Stakeholder Summary**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Responsibilities** |
| **Faculties** | A member of the university who teaches a course or more | Responsible for bringing back the book after a defined period |
| **Staff** | A member of the university who has an administrative role | Responsible for bringing back a book after a defined period |
| **Student** | A member of the university who studies different courses under a program for a defined period | Responsible for bringing back a book after a defined period |
| **Librarians** | The main person Interacting with the eLibrary system | Can manage books and members |
| **Admin** | Oversees the whole system | Can add and delete librarians |
| **Developers and Testers** | Develops the system based on the SRS | Are responsible for developing system features, fixing bugs and maintaining the systems availability |

**3.2 User Environment**

Number of people involved in completing the task.

* *3 members*

Is this changing?

* *No*

How long is a task cycle?

* *2 weeks*

Amount of time spent in each activity.

* *7 days*

Is this changing?

* *No*

Any unique environmental constraints: mobile, outdoors, in-flight, and so on?

* *No*

Which system platforms are in use today?

* *Java web,*

Future platforms?

* *Mobile APP (Android))*

What other applications are in use?

* *None*

Does your application need to integrate with them?

* *No*

**4. Product Overview**

**4.1 Product Perspective**

System is independent and totally self-contained (independent server). The product is not a component of a larger system. The system is made up of a database, enterprise service and web front.

**4.2 Assumptions and Dependencies**

Assume we use a LINUX server and a free database (community version of MySQL database, community version), if they prefer using a windows server or any other commercial server they will have to buy it and we can offer technical support

**4.3 Needs and Features**

This tables lists the problems which we map to needs and resulting features to be implemented. It also delimits the scope of the whole project.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Problem | Need | Priority | Features | Planned Release |
| Admin | | | | | |
| 1 | The university may have one or more librarians working at different moments | The system should be able to authenticate the different librarians created by the admin | moderate | Admin must be able to add and delete librarians (He /she is the superuser if the system) | 05-10-2019 |
| Faculty | | | | | |
| 1 | The faculty needs to borrow a book and use it for a specified period | We need to keep track of the checkout date and check-in dates of the book | high | The system should be able to register new faculty and keep track of his/her records with regards to checking in, checking out and payment of overdue fees |  |
| 2 | The faculty should be able to search for a book from a list of books | A database that stores the list of books from which a Faculty can make a search | moderate | The Faculty should be able to login to the system and has an interface dedicated to book search |  |
| Student | | | | | |
| 1 | The faculty needs to borrow a book and use it for a specified period | We need to keep track of the checkout date and check-in dates of the book | high | The system should be able to register new faculty and keep track of his/her records with regards to checking in, checking out and payment of overdue fees |  |
| 2 | The Student should be able to search for a book from a list of books | A database that stores the list of books from which a Faculty can make a search | moderate | The Student should be able to login to the system and has an interface dedicated to book search |  |
| Librarian | | | | | |
| 1 | The librarian should be able to charge Overdue fines for books that are checked in late | The system needs a either a sub system that will automatically calculate the charging and is connected to an external system that handles electronic payments or we can calculate the overdue fees directly based on the business logic | high | The librarian should be able to determine the overdue fines associated to a member easily |  |
| 2 | We need Manage members that belong to the library | The librarian should be able to add, delete and modify members of the library | moderate | We need add automated registration system that allow for the librarian to add new members who are eligible to checkout and check in books from the library |  |
| 3 | Check overdue book list | At any time, we should be able to view the list of all books that have exceeded the check-in periods and calculate the overdue fee associated with all or each of them | moderate | The system will have an automated functionality to handle this with ease |  |
| 4 | Check out books | The librarian is the only authorized personnel to check out a book to a library member | high | Each time a member requests for a book, depending on its availability, the librarian should be able to give to the member and keep track of the check-out date |  |
| 5 | Check in books | The librarian is the only authorized personnel to check in a book from a library member | high | Once a member comes to return(check-in) a book, the librarian should validate the check-in based on the overdue associated with that book. This can be done by the librarian or an associated external sub-system (Bar code scanner) |  |

**4.4 Alternatives and Competition**

MUM library has the choice to maintain the existing system, buy a competitor’s product or use our system

1. Existing Manual system

Advantages

1. Little or no computer literacy needed to use such a system

* Disadvantages

1. In the event of an accident or a disaster, all data are lost
2. A lot of mistakes associated with processing information manually
3. Unreliable and time consuming
4. Competitor’s product

* Advantages

1. easy to maintain

* Disadvantages

1. May not depict the current business logic

2. Costly and irrelevant for the current business goals of the library

3. Complex and poor user experience

4. The system may have too many shutdowns or bugs

**5. Other Product Requirements**

Applicable standards

* ISO/IEC/IEEE 12207

Hardware

* Computers (Desktops)
* Routers
* A Server

Performance requirements

* The system needs to be up and running 99% (Optimal performance) of the time. All the features in the system should work correctly all the time, the system should be robust enough to support 5 library members using the system at a time and 3 librarians logged in. The User experience should be good as the librarians and members may not be good computer literates.
* We define an error margin of 1% of when the system must be down
* Period updates on both the hardware and software may affect the whole performance
* The system will be robust and secure

Documentation

* A user manual for deploying and using the system will be made available for all stakeholders to ease system maintenance
* A telephone number will be made available for any request with regards to the system
* Periodic inquiry mails will be sent for survey purposes
* Online resources will be made available for quick help
* Videos on how to use the system will be provided

Design Constraints

* The main constraint is time given to implement all the features associated with the solution
* The design reflects the current business needs and logic of the library and may need to evolve over time
* The system depends on the availability of electricity and on the optimal performance of all hardware associated with it

Environmental Requirements

* The whole system should work on a hardware that works on solar energy