**Updated Library**

**Management System**

**CSCE 3615 Sect. 003**

**Ryker Valnes, Mark White,**

**Nicholas Wharton, Diante Richardson**

**Problem Overview**

We are tasked with designing a computer system for the small town of Smallville. The library needs a computer system, as they are working with paper at the moment, to track books, fines, mailing, and other services provided to patrons. Their paper system is becoming too difficult to maintain for the library staff, and the head librarian wants to improve the library's capabilities. The computer system must be able to add and remove patrons and books, record borrowing and returning of books, track fines and overdue books, as well as other services detailed by the client. The system should also support business rules such as limiting the number of items patrons can check out and requiring patrons to pay outstanding fines before checking out additional items. Additionally, the system must be designed with future expansion in mind, including inter-library loans with neighboring cities and self-check stations.

**Document Purpose**

The purpose of this document is to clearly define the scope, function, and behavior of the proposed Library management system. The document not only serves as a blueprint for the system, but is also used in outlining the needs and requirements that the client has provided. It will be used as a communication tool between the development team, and the client and business. It is also used as a guide to keep the development team on track as to what is required out of the computer system.

**Client**

The client is the public library of Smallsville, more specifically the person we are in contact with is the head librarian. With limited resources, budget, infrastructure, and knowledge of how computer systems function. Their manual system has become far too outdated, so the client is looking for a computer system that can help to automate alot of the tasks that are currently performed by hand. The head librarian has given us a list of tasks to complete, as well as how certain library rules work, as to implement them within the computer system. Keep in mind as well, that the client would like not only to expand this within the library, but possibly expand it between libraries in other cities as well.

**Constraints**

* Time Constaints
  + This system should be developed within a reasonable time frame, as further delay will only but the library further behind. However, the time of the project could be delayed by the lack of library resources or the complexity of the system itself.
* Solution Constraints
  + The system should be designed to meet the library’s specific requirements outline in the prompt such as adding and removing patros, tracking books, and generating reports
  + The system would be best if it was user friendly as the workers might not understand technology as well.
  + The system must be secure and protect patron information as it can have sensitive information such as address, date of birth, etc.
  + The system needs to be scalable as it will need to accommodate alot of patros and potentially more if that number were to grow.
* Environmental Constraints
  + Needs to be connected to the current technology infrastructure in the library, internet, printers, or other hardware.
  + Should be able to function on one PC at the beginning, as that is all they have, but still have potential to grow as the library will most likely add PC’s in the future.
* Resource Constraints
  + The library most likely has a limited amount of financial resources from the government that they can devote to the development and upkeep of the system.
  + The library has limited technological resources, only one PC, as well as workers who are not very used to using technology and computer based systems.
* Other Constraints
  + The system should keep in mind the route for possible expansion both inside and outside the initial library, as the head librarian says there is a potential between inter-library loan systems or self-checkout stations.
  + It should also comply with any laws that go into regulating personal data and privacy of the patrons and book information.

**Functional Requirements**

4.1 Add a Patron

This is adding a community member to the libraries system, taking down their information such as name, address, phone number and email address. Also generating and providing the community members with their unique identification number. This way they can now start accessing the system and borrowing items. This could be measured by the amount of patrons that are currently in the library system. As it would be successful if multiple accounts exist since they would have needed to be added using this function.

4.2 Remove a Patron

This is removing a community member from the library's system. This would remove the connection between the patron and the ID number making it free to be used for another patron in the future. Additionally all of the accounts information would be removed from the patron database. This could be measured by looking at the amount of patrons in the system. Then by removing a patron it can be determined that it functioned based on the change in the amount of total user accounts in the system.

4.3 Update Patron Information

This would allow for the changing of the information correlated with the patrons account. Since a person's address or contact information could change frequently, it would be annoying to have to delete and recreate the account every time the information changed. So this would just simplify the process. This could be measured by looking at the account information before editing. Then editing the account information. Querying the account information will then provide if the edit function worked correctly.

4.4 Patron ID Generation

An ID number should be provided to each patron that is added to the system. Each of these IDs should be unique for each patron. The ID should start with SV for smalls ville. Then a dash followed by 6 digits with the first ID assigned should start with the number 1 and each ID after should be incremented by 1. This way the ID numbers will never overlap as the library shouldn't be expecting to have over a millions accounts. But this could be easily changed in the future if they pushed towards expanding the system. This could be measured by creating a program that verifies all of the patron IDs are unique.

4.5 Item ID Generation

An ID number should be given to each book and AV item that is added to the system. But each copy of a book or AV item should have their own unique ID. This ID number is separate from ISBN or the dewey decimal system. The ID should start with SV for smallsville. Then a dash and 9 digits. The first ID assigned should start with the number one and increment each after by 1 to limit overlapping numbers. The ID of each item should be labeled on the physical book or AV item. This could be measured by creating a small program to verify that each item in the item catalog has a unique ID.

4.6 Patron Data Storage

The Patron information needs a data store that would hold all of the given user information. The storage should be easily accessible to all computers on the network running the application. The database would need to be stored either on the local network or online so when adjustments are made the entire system will receive the updated information. This could be measured by making updates on one computer in the system and seeing if the information gets received by the others.

4.7 Item Data Storage

Each book and AV item copy owned by the library and provided an ID will be stored in the item database. This database should be easily accessible for all of the computers on the network running the application. This is stored online or on the local network. This way the information is always updated on each system. This could be measured by adding a book using one computer and verifying that it has been added to the others on the network.

4.8 Item Catalog

The item catalog would display the information of the Item database with a friendly user experience. This would be a large piece of the application allowing patrons to search for books in the system based on keywords. The search feature should provide filters to make searching for items an easier experience. There should also be different sections to search for books or AV items. This could be measured by having a couple people outside of the development team pull a book off the shelf, and seeing if they can locate it using the catalog. If the book can be found it would give a good indication that the application is somewhat user friendly.

4.9 Add Item to the Item Catalog

The library staff should be able to add new copies of books and AV items to the database and catalog. This feature should be a part of the item catalog application but only seen by those using the library staffs version of the application. When adding the item there should be fields such as category, genre, author, date published, etc, to add information about the book to the online portal. This could be measured by adding an item using the librarian interface of the application. Then seeing if the added copy gets updated to the other computers on the network.

4.10 Remove Item From the Item Catalog

The library staff should be able to remove a copy of a book or AV item from the database and catalog. In case a copy of a book goes missing, or it gets damaged and needs to be removed from shelves. The ability to remove items would only be accessible by the library staff's version of the application. This could be measured by removing a copy of a book using the interface of the application. Then verifying that it was removed from the other computers on the network.

4.11 Edit Item on the Item Catalog

The library staff should be able to edit the information linked with each item. This would give the staff the ability to update information if needed and fix mistakes without having to delete and re-add the complete item listing. This should only be accessible from the library staff's version of the application. This could be measured by editing the information from one of the staff computers. Then seeing if the edit appears on the other computers on the network.

4.12 Borrowing Items

On the application at the top there would be a button to open a menu for borrowing items. You can type in the item's ID number that can be found on the label. You can choose whether to add another item to the order. Once all of the items have been input, you can provide the patron ID number and click submit. Then the application will decide if the order can occur, based on already checked out books or unpaid fees. But if nothing goes wrong it will display the checkout time and the amount of renewals left for each item. It will update the status of the book in the catalog. This can only be done on the library staff's version of the application. You could test this by checking out a book and checking if it's displayed on the catalog as checked out on all systems connected to the network.

4.13 Returning Items

On the application at the top there would be a button to open a menu for returning items. You can type in the ID of the item, then select if you want to add another. Then you can input the patrons ID number and click submit. Once it submits, the application will display whether the books are overdue and calculates the late fee. If the books were returned in time then it would give the option to renew the books if available. If so it will keep the item set as checked out and update the renewal value for the copy. Otherwise it will update the book to be available and it will be put back on the shelves. This can only be accessed on the library staff's version of the application. You could test this by returning an item and seeing if the book appears as available on all computers on the network.

4.14 Tracking Fees

The current owed fees should be attached to the patron account and stored in the database along with it. As the fees are calculated using the information from the item database, they should be added onto the current owed fees for the patron. That way when they need to be paid for the number is already compiled, instead of the staff having to find the occurrences and calculate it. This could be measured by making a dummy account with multiple owed fees. Then computing the fees owed and verifying that it matches up with the one automatically calculated by the system.

4.15 Fee Payment in-person

A separate menu on the application which could only be accessed on the library staff's version of the software should allow for a cash transaction to occur. The employee would be able to enter the patron ID of the person paying to find the amount of money they owe. After the person pays and gets their change returned to them, the employee can acknowledge that the payment has occurred by clicking a button on the menu. Then the fees linked to the patron would be set to zero. This can be measured by taking a dummy account with fees and testing to see if the fees are reset when the transaction is acknowledged.

4.16 Fee Payment online

Fees could be payed online using the online portal. Any person could open up the application and log in using their patron ID. A button will appear that will open a menu for paying fees. The page will show the amount owed and have fields to type in card information. The card information would not be saved in any way. The application will process the transaction and verify to the user that it went through. The application will then update the fees owed for the user to zero. This could be measured by setting up a dummy account with some debt and putting in card information. One faulty to make sure to doesnt let it go through. Then once with real card information to make sure it works properly.

4.17 Library Loaning

The Library will be able to loan books to other libraries. A field for each item should be in the item database showing the status of the item. Showing if it is available in the current library, or other libraries using the software. On the library staffs version of the application they can click the loan item button which opens a menu. In the menu the librarian can type the items id and what school it is being loaned to. Then it will update the location of the book on the item catalog. But not changing the id number so its known the book came from smallsville being denoted with the SV before the numbers. This could be measured by loaning a book to a dummy library and seeing if it changes the status of the book. Seeing that the book location is listed as the dummy library set would mean the requirement was achieved.

4.18 Print Lists

This will take the information from either the items or patron database. The application will generate the information in a easily readable pdf format that can be printed out for the library staff if needed. This could be measured by generating the item pdf file, and checking if all items in the catalog appear in the file. If so the list is being printed properly.

4.19 Notify Patron of Overdue Books

This will be an automated process that notifies patrons just before a book becomes overdue, and reminds them of the fees that accumulate as it remains borrowed. These messages would be sent as emails. This would relieve stress from the library staff from having to track down and notify people of this constantly. This could be checked by creating a dummy account with a email set, and setting their books as overdue. If an email is received with the appropriate information then the program functioned properly.

4.20 Reference Books

Reference books such as encyclopedias should be shown as items in the catalogue. But the status of the book should be set as in library only. This would give people the ability to still find these reference books in the catelog, and it would show them that they would have to be in the library to access the materials. This could be checked by adding an encyclopedia section to the catelog. Then checking on the other devices on the network to see if the listing appears as in library only.

4.21 Print Overdue Books List

This would print a list of all the over due books still being borrowed from the library. The list would include the patron borrowing the books and their contact information. This could be helpful to the library staff so they wont have to find the information individually themselves. The list should be in a easily readable pdf format that could be printed. This could be checked by printing the list, and manually making sure that all overdue books in the system along with the correct information is displayed.

**Non-Functional Requirements**

Look and Feel

* The system will have a user-friendly interface that is easy to navigate. Additionally, it will be visually appealing and allow users to easily manage patron information, item catalog, and fee information. There will be separate sections on the website to allow easy access to the individual information that is needed.
* The patron section will allow library staff to add, remove, and update patron data, along with generating the patron ID. A separate tab will include the item section where you will be able to see item data, catalog, add data, remove information, edit data, and borrow and return items.
* The last section will include information about fees, allowing you to easily track information about a patron’s fees.

Usability

* The system will be easy to use and require minimal training for library staff, those young and old. Organized sections will allow easy access to data and less room for user error. The application will be used mostly via computer and will require a library staff username and password in order to gain access.
* If the user has any questions there will be a list of commonly asked questions and solutions to each, along with a number to call if further help is needed.

Performance

* The system will have fast and responsive performance, along with the ability to handle a large number of transactions. Interactions between the user and the system should have a maximum response time of two seconds.
* The system will be able to cater to hundreds of users, especially during busy library periods.

Operational

* The system should be available 24/7, as long as the library is open with minimal downtime for maintenance. It will be created for the library staff usage.

Maintainability and Portability

* The system will be easy to maintain and update, also allowing individuals to operate with the app on different software platforms. In order to keep the system up to date there will be weekly maintenance reports.
* The system will be expected to run on windows, apple and linux.

Security

* The system will have strong security measures to protect the library and patient information, this includes backup and recovery procedures along with user authentication to keep data private.
* When using the library computer to access the system, specific sites will be blocked to lower the risk of unwanted installation of any other application

Cultural and Political

* The system should respect cultural and political sensitivities, including privacy concerns and data protection laws. There will be no icons or information that could be considered offensive or culturally insensitive.

Legal

* The system will be up to date and comply with all laws and regulations, including copyright and intellectual property laws as well as data protection and privacy laws.