Middle East Technical University
Department of Computer Engineering

Computer Engineering Design I
fall 2010

Software Requirements Specification Report

for

bookwiser augmented reality for libraries

The Ballmer Peak

Contents

1 Introduction

1.1 Problem Definition

Libraries are important for all societies since they are banks we store a cultural treasure in. In just one day, hundreds of people visit libraries to read books, make scientific researches or to study. Since there are many choices in a library, it is hard for a user who searches for a specific topic or to have a general idea about organization of books in the library , the relation of a book to the topic or existence of books with the content user may be interested in. A user who wants to read a book but does not have a decision on which book to read has to take each book from the shelf, open each book and try to find some information that may help her/him to decide.

Making the user's travel in the library more efficient by recognizing objects and showing information about the library environment on user's screen using augmented reality concept is a perfect solution to these problems.

1.2 Purpose

This document includes software requirements for **Bookwiser** Software Project.

The project uses augmented reality concept to make user get easily informed about the organization of the library, books, content of the books and also interact with the library. The project is essentially composed of three parts. First part is the object recognition part which includes the shelf recognition, book recognition and librarian recognition. The second part is the interaction with the library database from which all the information about a book is extracted. The last part of the project is the GUI part which is going to show the modified video stream augmented with information related to recognized objects on the camera, provide options for user to choose the flow of the application and make user's experience in the library more effective.

1.3 Project Scope

The project is named **"Bookwiser"**, since the application seems to know everything about every book in the library, and helps user about books.

Our project will be a augmented reality application for a library environment which works on mobile devices. When a user starts the application, **Bookwiser** first enables the video in the library. While the user is moving with the camera in the environment, **Bookwiser** will capture data from environment.

The user may choose to search a specific book or decide just walk and get informed about different kind of books in the library. In the first option the user gives the name of the book that he/she wants to reserve and **Bookwiser** will guide the user in order to find the correct shelf

and the correct book. The other option is that the user has no idea about what he/she exactly wants , therefore **Bookwiser** will guide the user that while user is walking around the library, it will give information about the shelves and books that is in the predefined range of the camera. When **Bookwiser** comes up with a book shelf belonging to a specific kind of books, it will detect the shelf by using object recognition in a predefined range. After the shelf detection & classification, the project will inform the user about the content of the books in the shelf. When **Bookwiser** recognizes a book, it will show the rating of the book to the user using augmented reality and it will give detailed information to the user if the user wants. It will give user choices like rating the book or reserving the book. Moreover, there will be librarians in the library for further questions and they will be detected using the predefined specifications.

We think that people going to the library are mostly wasting their time for searching the correct place of a book or deciding which book to read. So that, **Bookwiser** very helpful to guide people in libraries.

1.4 User and Literature Survey

1.4.1 Literature Survey

Object-recognition and augmented reality become popular since it provides users a better understanding of their environment and it makes it easier to mix computing capabilites of computers with flexible thinking capabilities of human-kind.

On 15 August 2006, Alex Chitu stated on Google Operating System Blog [3] that , search engines like Google are building their future on object-recognition and augmented reality. The experience of searching via a device modified for object-recognition and capable of bringing detailed information about the object is considered to be a bright idea for the future.

A company called GeoVector [4] which is specialized in pointing based search solutions, created a new technology using a GPS sensor. By using this sensor with a mobile device a user can find the location of a specific search interest. But this product doesn't use object-recognition methods to point to the location of an object.

An important project that uses object-recognition and augmented reality is a product called **AugmentedID** [5] developed by a Swedish software and design company   named "The Astonishing Tribe" [6]. This product recognizes faces of users and helps other users contact the user via the recognized person's social network accounts or e-mail.

AugmentedID

Joe Maglitta, on his blog post [7] , states different uses of augmented-reality projects.

According to the text PopSci and partner GE claimed to publish the first interactive 3D cover magazine cover. Three windmills pop off the page, build themselves and start twirling. The project was built using Unifeye Design software, billed by its German creator Metaio as "the first professional augmented reality software, specifically developed for users in information design."

The text also states that there are smart-shopping applications of augmented-reality. It is said that on-line retailer Zugara, whose "Web cam Social Shopper" app lets users select clothes, print a special "marker" app and "hold" up articles of clothing up in front of themselves as it tracks their movements" makes users see how the clothes fit on their bodies.
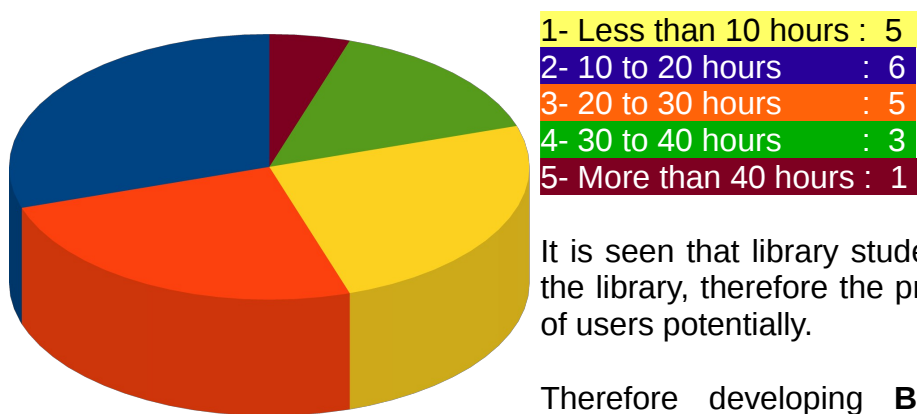
Many software products related to the augmented reality problem offers many kinds of object recognition such as face recognition, vehicle recognition and building recognition. They mostly aim to ease the load of the user or they are just developed for commercially purposes.

The **Bookwiser** Project is an assemble idea to that projects since it is comforting the user, but differs from the current projects with its environment and methods. In other words, there is no current project for the library occasion and the product has a novel approach.

1.4.2 User Survey

The potential users of the **Bookwiser** Project are the library visitors and the librarians. Therefore 20 Middle East Technical University students are asked to answer 5 questions to develop **Bookwiser**'s goals, usability and effectiveness in a more meaningful manner.

1) How many hours a week do you spend in the library?

| 1- Less than 10 hours : 5 |
| 2- 10 to 20 hours   : 6 |
| 3- 20 to 30 hours   : 5 |
| 4- 30 to 40 hours   : 3 |
| 5- More than 40 hours : 1 |

It is seen that library students spend a lot of time in the library, therefore the product has a large number of users potentially.

Therefore developing **Bookwiser** is meaningful because of a great user potential .

2) Assuming that you often use the library and know the places of books with different genres, how many minutes do you spend to decide on a book to read in the library?
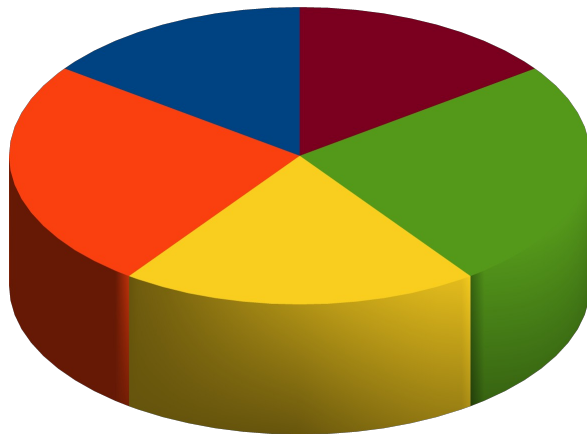
1- About 10 minutes	:	3
2- About 15 minutes	:	5
3- About 20 minutes	:	4
4- About 25 minutes	:	5
5- 30/30+ minutes	:	3

The results of the question showed us that most of the library users spend about 20 minutes to decide which book to read, even if they know the specific locations of books with different genres.

This shows that **Bookwiser** should help the user decide on which book to read to make them consume less time while they are looking for a book they can enjoy.

3) Do you think finding a specific book in a library is difficult? Rate from 1 to 5, 1 means very easy and 5 means very difficult.
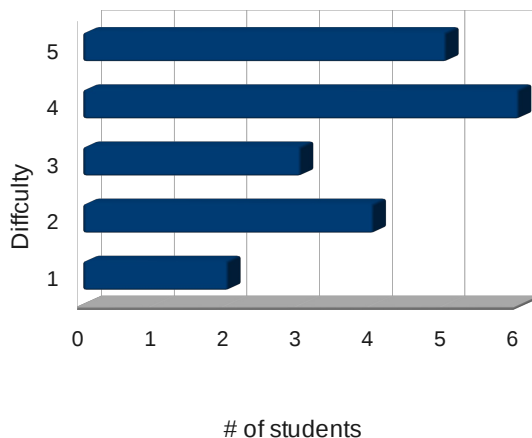
Results of this question showed that most of the students think that finding a book is difficult.

Therefore providing a specific book search for users to make their experience in the library easier is a good decision.

4) Have you ever used a product that uses augmented reality ?
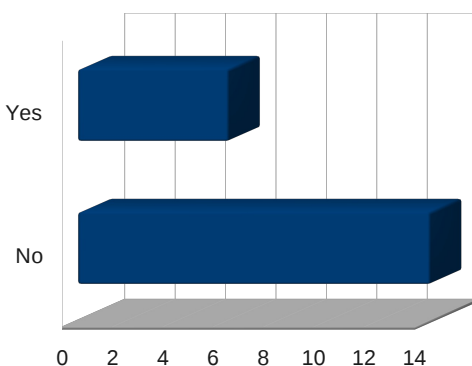
It is seen that 70% of the students answered "no" to this question.

This means that library users are generally not used to the augmented reality concept so **Bookwiser** should really be user friendly and should have a detailed "help" section to make it easy for users to get used to the product.

5) Do you think making libraries more user-friendly would make you more productive ?

Most of the students think that making libraries more user-friendly would increase their productiveness.

Also this means that users would want to use **Bookwiser** to increase their productiveness.

Therefore the project also has a positive effect on the community and will be used for a good purpose.

1.5 Definitions and Abbreviations

Augmented Reality: Term for a live direct or indirect view of a physical real-world environment whose elements are augmented by virtual computer-generated sensory input such as sound or graphics. For **Bookwiser** application, augmented reality represent the user graphical output that shows the recognized object properties.
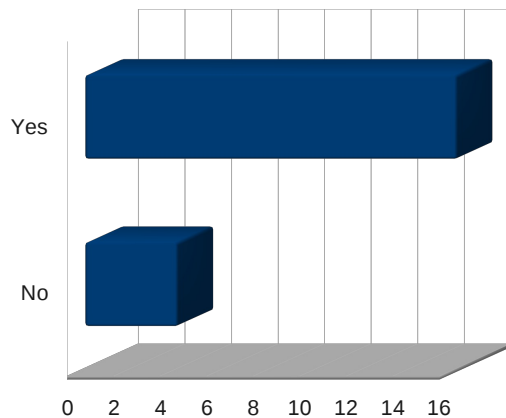
Bookwiser: Name of the project.

Camera: A device that provides a video stream

CPU: Hardware component also named as processor.

GPU: Graphics Processing Unit

DFD : Data flow diagram.

Image Processing: image processing is any form of signal processing for which the input is an image, such as a photograph or video frame. For Bookwiser application, image processing is the process of generating information from video stream captured by user camera.

Library: Real building where books or other kind of documents are stored for public use.

Librarian Recognition: Process of recognition of workers in the library by the Bookwiser system.

Object recognition: Object recognition is the process of retrieving identifying information about objects. Object recognition process is directly connected with image processing.

Object recognition range: The closeness of the camera to the real object which provides enough detail to recognize the object.

SRS: Software requirement specification.

Video Stream: Video stream is the common name of Bookwiser's input and output format. It consists of stream of real time images.

1.6 References

1- IEEE Std 830-1998: IEEE Recommended Practice for Software Requirements Specifications

2-Cocomo Resource Estimation Model, http://www.cs.unc.edu/~stotts/145/cocomo.html

3-"Object Recognition is the future of Google", 15.08.2006, Alex Chitu, http://googlesystem.blogspot.com/2006/08/object-recognition-is-future-of-google.html

4- "3D Search, Real-time Travel Guide" , 17.06.2006, Alex Chitu, http://googlesystem.blogspot.com/2006/07/3d-search-real-time-travel-guide.html

5-"Augmented ID: Augmented Reality Facial Recognition for Mobile" , Sarah Perez, 10.06.2009
http://www.readwriteweb.com/archives/augmented_id_augmented_reality_facial_recognition.php

6-"the Astonishing Tribe", http://www.tat.se/

7-"10 Smart Examples of Augmented Reality" , Joe Maglitta, 08.04.2009 ,
http://www.smartertechnology.com/c/a/Smarter-Strategies/10-Smart-Examples-of-Augmented-Reality/

8- Use Case Diagram, Wikipedia, http://en.wikipedia.org/wiki/Use_case_diagram

9- Data Flow Diagram, Wikipedia, http://en.wikipedia.org/wiki/Data_flow_diagram

10- State Diagram, Wikipedia, http://en.wikipedia.org/wiki/State_diagram

1.7 Overview

This document is organized in an incremental level of specifications, at each level , details of the project is explained in a more extended way.

Section 2 gives an overall description of the project , specifying general functionality of the application and mentioning some foreseen limitations. Section 3 gives a detailed explanation of major functions of the project, predefined requirements and specifications. Section 4 includes the data structures to be used in the project and defines relationships between them. Section 5 explains the major flow of the application in details. Section 6 gives an estimation of the basic schedule and development methods to be followed by the developers and Section 7 concludes all of the sections mentioned above.

2 Overall Description

In this section an overall description of **Bookwiser**, its functional specifications and limitations are provided in detail.

2.1 Product Perspective

Bookwiser is a semi-independent system which is designed for assisting library users in active tour in the library. It collaborates with the library database. Since the main function of the product is object recognizing and also most of the libraries have their own databases, the library database design is out of the scope.

To use **Bookwiser**, user should login to the system via authentication system. Authentication system checks the validity of login information from library database. After login, user can interact with the system via graphical user interface. Graphical user interface displays modified video stream sent from video stream modifier along with the user options.

The job of the video stream modifier is to integrate captured video stream,which is sent from the camera, with the object properties. In other words, it creates the augmented reality which is shown to the user.

Bookwiser uses the video stream captured by the user camera as basic input data. This data is processed by the object recognizer. The object recognizer part of **Bookwiser** processes the video stream using information retrieved from object recognition database that is internal database of **Bookwiser**. This database holds the information for describing objects. The object recognizer generate an object identifier data for each object which uses for querying object properties from the library database.

All these system components and external elements with which the system interacts are shown in the figure below. Detailed explanations for these elements are included in next chapters of the document.
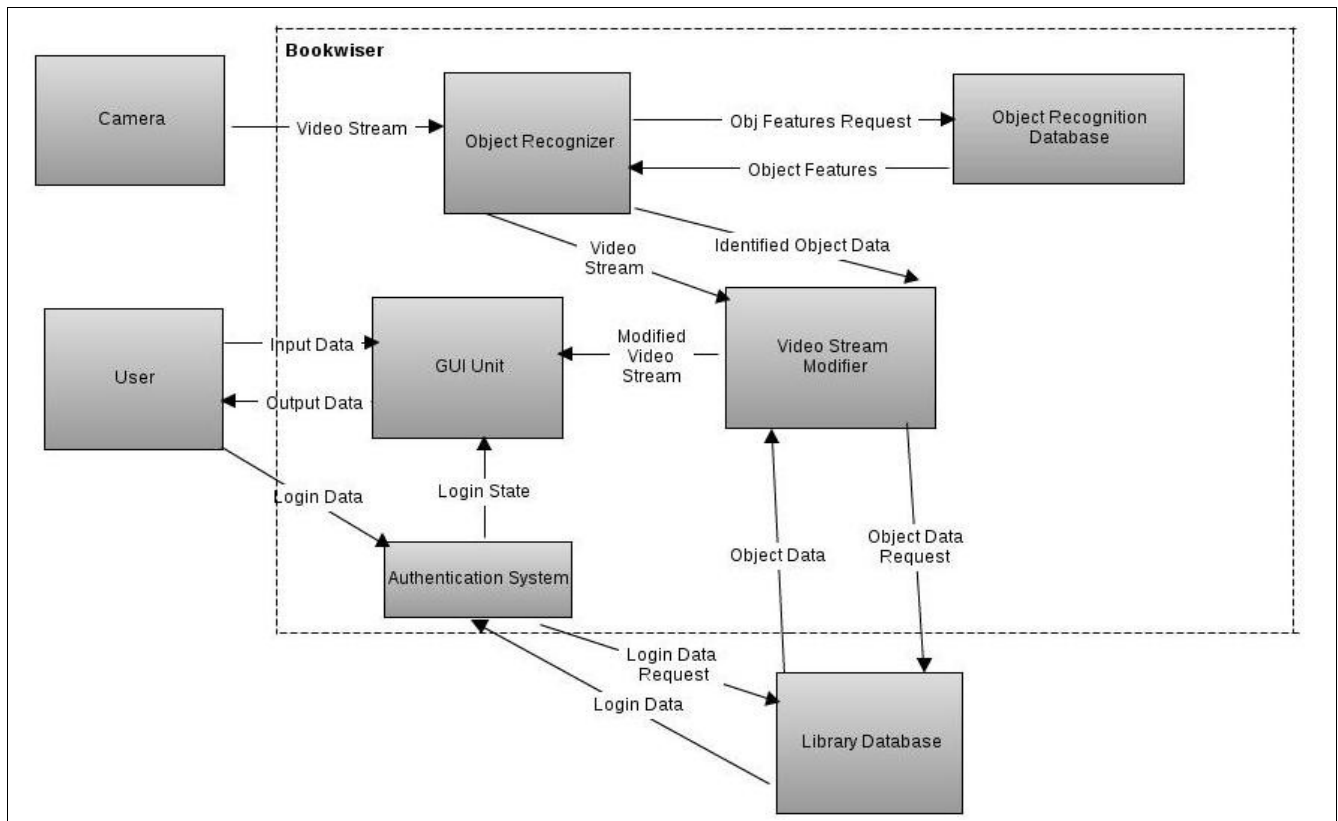
Diagram 2.1 - Block Diagram

2.2 Product Functions

In this section use-case diagrams are provided and explanations of the major functions of **Bookwiser** are explained in detail. In the diagram above you can see the use-cases of Bookwiser to understand the functionalities of Bookwiser better.
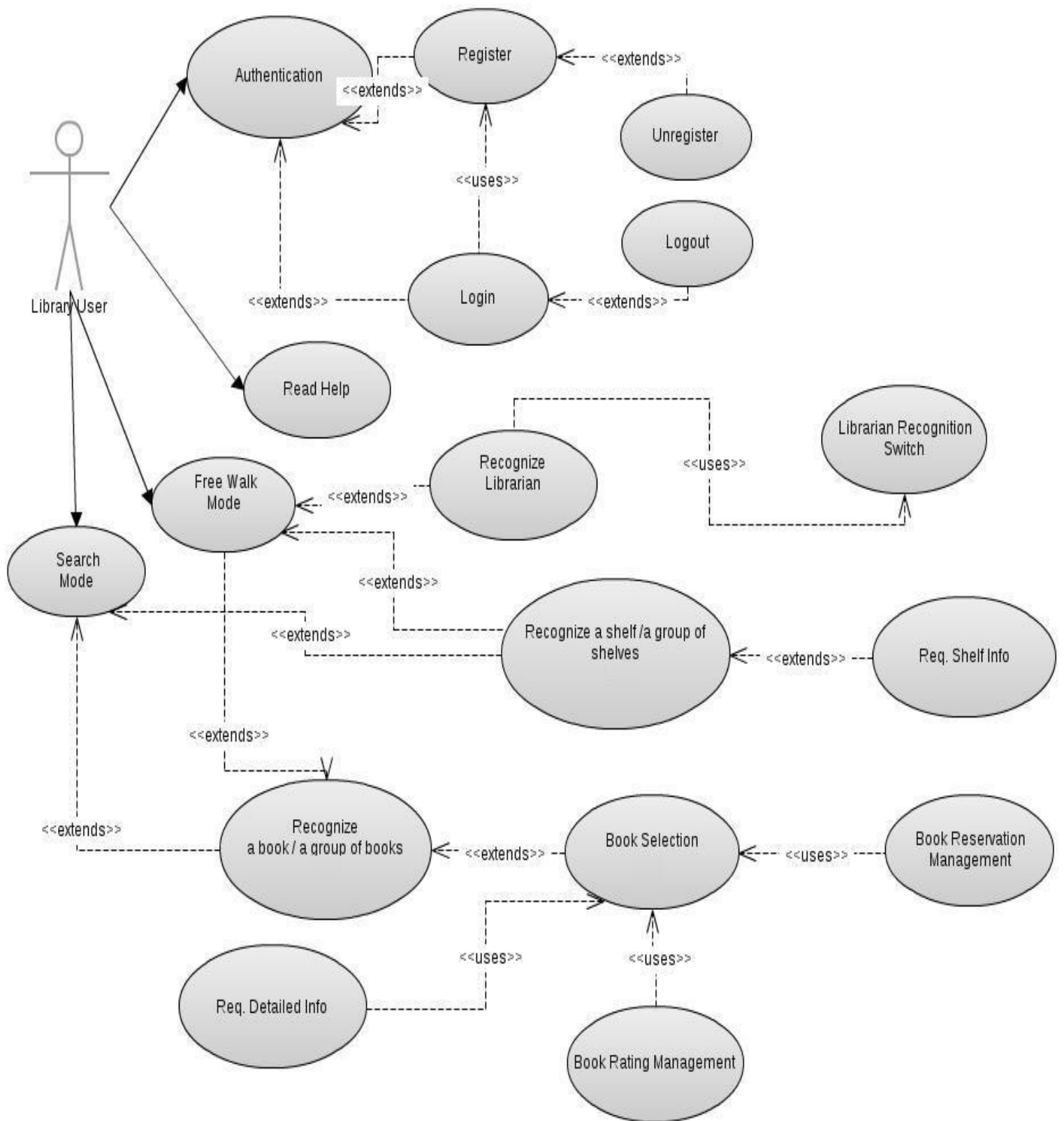
Diagram 2.2 – Use Case Diagram

(01) Register:

One-time registration is a must for a **Bookwiser** user. User should give these information at this step:

- **Name**: Real name of the user.
- **Surname**: Surname of the user.
- **Date of Birth**: Date of birth of the user should be given.
- **E-mail**: E-mail address of the user.
- **Nationality**: Nationality of the user.
- **Address**: Contact address of the user.
- **Phone number:** Users phone number.
- **Gender**: User's gender.
- **Profession**: Profession of the user.
- **Preferred Language:** User's preferred language.
- **Password**: User's password for authentication. The information will be encrypted with '*' marks on the display. An encrypted version of the password will be sent to the library database. This information is required to continue registration.

(02) Login

A **Bookwiser** user has to be registered before using the application therefore the login phase is a must for the user to start using the application. The login functionality will require the user to enter some specific information , and user will try to succeed in authentication process by entering correct information. The required fields are:

- **E-mail:** User enters the e-mail address he/she provided at the one-time registration phase.

- **Password:** User enters the password he/she provided at the one-time registration phase.

If the login process is successful , the application automatically goes to free-walk mode.

(03) Logout

After log-in is done and the session for a user starts, A **Bookwiser** user will use this functionality to end his/her session when his/her use of the system is over. The logout function will clear all variables related to that session and will put the application back on the login state for another user to log in and use the system. Before proceeding the logout function user will be prompted if he/she is sure about the operation.

(04) Unregister

If a **Bookwiser** user decides to remove his/her account in the library system for some reason, by selecting this function , his/her request will be sent to the library database and the account deletion process will be done at the library database level. Before the information is sent to

the library database, user will be prompted if he/she is sure about the operation which can't be undone and will be asked for reasons of unregistration. Answering to the question for reasons of unregistration is not a mandatory step. The user will be able to continue the process without giving an answer to this question.

(05) Recognize Librarian

After a successful login, if a **Bookwiser** user choose the "librarian recognition option", whenever a librarian is in the librarian recognition range of the application, the user is informed about existence of a librarian in the librarian recognition range of the camera, who will be recognized by specific clothing , color and special marks. The recognized librarian will be highlighted on user's display and the user will be told that he/she can get any help from the librarian. The librarian recognition mode wont be active if the "librarian recognition option" is deselected.

(06) Select Librarian Recognition Option

If a **Bookwiser** user selects "librarian recognition option" during the flow of the application, Librarian Recognition Mode will be activated. Whenever a librarian is in the librarian recognition range of the application, the user will be informed about existence of a librarian in the librarian recognition range of the camera, who will be recognized by specific clothing , color and special marks. This option's state for current user will be saved in the library database for future uses.

(07) Deselect Librarian Recognition Option

If a **Bookwiser** user deselects "librarian recognition option" during the flow of the application, Librarian Recognition Mode will be deactivated as an inverse effect of the case (06). This option's state for current user will be saved in the library database for future uses.

(08) Search Mode

After login is completed successfully , if a **Bookwiser** user selects the search mode, during the travel of the user in the library no other book than the selected book will be highlighted on the display screen. During the travel of the user in the library, if the shelf containing the wanted book is in the shelf recognition range of the camera, the shelf is highlighted with a special indicator of geometric place of the shelf, otherwise no other shelves are highlighted. If the book the user searches is in the book recognition range of the camera, the recognized book is highlighted with a special indicator of geometric place of the book, otherwise no other books are highlighted. To activate search mode a user should specify a book name as explained in (09). Librarian recognition mode can be active during search mode.

(09) Enter Book Name

If a **Bookwiser** user selects search mode after a successful login, the user should enter the name of the book he/she searches into afield on the display in this case. After the user enters the name of the book, he/she clicks "continue" button and a search on the database is made.

If the book doesn't exist in the database, user is warned and the application automatically prompts if the user wants to search for another book, if user's answer is "no", the application goes back to free-walk mode which is explained in case (11). If the book exists in the database, search mode explained in (08) is activated.

(10) Cancel Search Mode

Although a book ,which the user specified as the target of the search mode, exists on the system, the user can choose to cancel the search operation and get out of search mode to the free-walk mode using this functionality. Free-walk mode is explained in case (11).

(11) Free-Walk Mode

After login is completed successfully , if a **Bookwiser** doesn't choose the search mode, automatically it is accepted that the system is in free-walk mode and during the travel of the user in the library , user is informed about the objects in the recognition range of the camera.

Free-walk mode helps the **Bookwiser** user get information of the content of the library easily, using augmented reality. If a single shelf or a group of shelves is in the shelf recognition range of the camera, the name of each shelf in the shelf recognition range of the camera is displayed on the display screen, over the locations of the shelves on the display. If a single book or a group of books are in the book recognition range of the camera, the rate level of each book (expressed over a 5 stars rating system) is displayed over the locations of books on the display. Librarian recognition mode can be active during Free-Walk mode.

(12) Recognize a shelf / group of shelves

If a single shelf or a group of shelves is in the shelf recognition range of the camera, the name of each shelf in the shelf recognition range of the camera is displayed on the display screen, over the locations of the shelves on the display.

When a single shelf or a group of shelves are recognized, an option called "request detailed info" will appear on the screen as a separate selection item for each shelf recognized in the shelf recognition range of the camera.

(13) Request Detailed Shelf Info

After recognition of a shelf or a group of the shelves in the shelf recognition range of the camera, if a **Bookwiser** user chooses to request detailed shelf information for a specific shelf, a general description of the genre of the books in the chosen shelf, a list of most read books on that genre and number of total books in that shelf is displayed to the user on the screen. User is able to close information box appearing near the shelf by clicking hide button.

(14) Recognize a book/a group of books

The user of **Bookwiser** is informed when encountered with a book/a group of books. This informing task is performed through the graphical user interface. To recognize a book/books, the camera should be directed to the book/books. The information given depends on the

selected mode. User can be either in search mode or free walk mode for recognition of a book/books:

- In free walk mode;
If a book/books are in recognition range of the camera, an indicator of geometric place/places of the book/books is shown in display along with the some extra information. The exact measurement of the recognition range of the camera for book recognition is described in non-functional requirements. Books are showed with their borders and user ratings on top of them as a number of stars. The number of stars is calculated from average ratings given to the book by users. Rating of the book/books and can have a value from zero to five. In both one book case and multiple book cases, user should select a book for detailed information. This situation is described in Book Selection.

- In search mode;
If the recognized book/one of the recognized books matches with the searched book, user is informed by only recognition of this book. This information same as the one book situation in free walk mode. If there are no matches found in the book recognition range of the camera, No information is displayed on the graphical user interface of **Bookwiser**. User should select the book for detailed information. This situation is described in Book Selection.

(15) Book Selection

The user of **Bookwiser** shall be able to select a particular book after recognition of a book/books. This task is performed through graphical user interface and selection buttons. When only one book is recognized, user can select the book by simply clicking the select button. When multiple books are displayed in the screen, user can preselect one of them using direction buttons. When the user click a direction button preselected book is changed to another one according to the button clicked. The indicator of the geometric place of the preselected book is displayed on the screen. The preselected button can be selected by clicking the select button.

After the selection of a book, specific information about the book is given on the screen along with an indicator of the geometric place of the book. Specific information given about the book includes
Rating: Rating is displayed as stars on top of the book. Number of stars which
                 can range from zero to five is direct measurement of the book rating.
Name: Complete name of the book.
Author: Name of the author.
Publisher: Complete name of the publisher of the book with its logo.
Publication Year: Publication year of the book.
Edition: Edition number of the book.
Abstract: Abstract of the book.

User is also given the instructions to do particular actions about the book. These actions are requesting detailed information, requesting the lists of the author of the book, rating the book and reserving the book.

(16) Rate

The user of **Bookwiser** should be able to rate the selected book according to his/her own judgment. To rating the book, user should click the rate button on the selected book screen. After clicking the rate button, a menu that shows five unselected stars will open. User can select desired number of stars as the scale of rating. One stars means that the rating is one and so on. After rating is done, user will automatically displayed the selected book screen.

(17) Request Detailed Book Information

The user of **Bookwiser** should be able to request detailed information about the selected book. To request detailed information, user should click the detailed information button on the selected book screen. After the request, detailed information about the book is displayed on the screen. Detailed information consists of a set of the summary of the book, content of the book, abstract of the book and any other related information about the book in the library. User can quit from detailed information screen with the unselect button.

(18) Request List of Other Books of the Author

The user of **Bookwiser** should be able to request list of other books of the author. To request the list, user should click the books of the author button. After the request, list of the books of the author is displayed on the screen. User can quit from books of the author screen with unselect button.

(19) Book Reservation

The user of **Bookwiser** shall be able to reserve the selected book and doing actions on the his/her reservations. To do this, user should click the reserve button. After clicking the reserve button, a library specific set of reservation rules is displayed on the screen. User will be asked if she/he accept the rules with an approval button on the screen. If user click the button, the book will be reserved to him/her.

(20) Extend Reservation

If a book reservation action is done, user is allowed to extend the time period of the reservation.

(21) Cancel Reservation

User can be able to cancel his/her reservation using the display. Canceling a reservation requires to reserve a book.

(22) Read Help

The user of **Bookwiser** is supplied with help document. He/she can be able to read the help

document of **Bookwiser**. In order to read the help, user should press an assigned button for help in any screen. The document should include the following information:

General information
Modes
    Search mode
    Free walk mode
Display Features
    Shelves
        Shelf recognition
        Shelf genres
        Selecting a shelf
    Books
        Book recognition
        Book ratings
            How to rate?
        Selecting a book
        Request more info
            Detailed info
            Other books of the author
        Reservation
            Rules
            Options
            Canceling a reservation
            Extending a reservation
    Librarians

2.3 User Characteristics

It is assumed that a **Bookwiser** user has basic level knowledge on using computers. User is assumed to have basic experience on interacting with a computer application via an user interface system. Also user should physically be able see the information shown on the display and should physically be able to travel in the library.

2.4 Constraints , Assumptions and Dependencies

2.4.1 Regulatory Policies

- **Bookwiser** uses the library's own database to show data about recognized books. The application itself is not responsible of the correctness of the shown data.

- The library, according to it's own rules, may block a user's access to the system or just book reservation functionality due to misuse of the system. The decision is library's and only responsibility of **Bookwiser** is to inform the user via graphical interface.

- **Bookwiser** should be automatically authorized by the library servers to reach to the database.

2.4.2 Hardware Limitations

- It is assumed that the system needs a minimum of 5.0 mega-pixels camera to feed the system with images with necessary quality and detail to do object recognition.

- The system is assumed to a work on a device with a minimum of 2.0 GHz Intel processor and a minimum of 500 core CPU.

- It is assumed that the device **Bookwiser** works on has a wireless communication device and the library's database is also reachable via wireless Internet connection.

- **Bookwiser** will not use external GPS or any other satellite/GSM based navigation systems or devices to locate or recognize books / shelves.

2.4.3 Interface Dependencies

- **Bookwiser** uses a library's own database to gather user , book and shelf data. The interface to communicate with the library database is dependent on library database's architecture.

2.4.4 Standards

- It is assumed that the library have adequate light sources around the objects

- It is assumed that shelves in the library are equally spaced with a minimum of 1 meters and they are placed parallel to each other.

- It is assumed that shelves in the library are standard and they have the same height/width and shape. Shelves should have a color that can be differentiated from the

library environment and special marks on the shelves should be apparent and proper enough to classify them.

- It is assumed that books are placed on the shelves with their front side facing the camera and the user. The books are assumed to be placed with equal spaces between them, a minimum of 10 centimeters. Books should have a color that can be differentiated from the library environment and special marks on the books should be apparent and proper enough to classify them.

- Librarians should wear a standard t-shirt with the same color and a specific apparent shape on top-left of their t-shirts` front and back sides, making them recognizable.

- The marks, values or features to recognize and classify books, shelves and librarians should be predefined at the object recognition database. **Bookwiser** will only recognize and classify objects with predefined marks , shapes and colors.

2.4.5 Safety & Security Considerations

- The devices which **Bookwiser** is run on should be tested for health issues, the screen and the device components should obey the standards.

- The library database manipulations done by **Bookwiser** are limited by the library constraints, book information addition/deletion/manipulation or access to other users' information is not made by **Bookwiser** to protect library database from possible misuse.

- Users' passwords and detailed information will be transferred with encryption to protect the system from illegal access.

- Before unrecoverable operations users should be asked if they are sure about the operations.

3 Specific Requirements

In this section we will explain the specific requirements in levels with detailed information using data flow diagrams. There are 3 levels in **Bookwiser,** each one of them explains the data flow of the project. How **Bookwiser** interacts with the external systems is generally explained in Level0. In Level1, the library application splits into 4 units in order to distinguish which part of the system interacts with which external system. Finally, in the Level2, these 4 units clearly explained by using data flow diagrams for each one.

3.1 Interface Requirements

In Level0 data flow diagram, it can be seen that the library application interacts with 3 external systems and a user. The user is using the application by clicking a button mainly and the application displays the result to the user. The application interacts with the main library database when the user uses the application at registration, or to get information of a recognized object. The main part of the application is the interaction with camera and internal library database. When the user starts to use application in the building, the application captures data from camera, after that, object recognizing process is done via the communication between the application and the internal object database that we will dynamically produce.
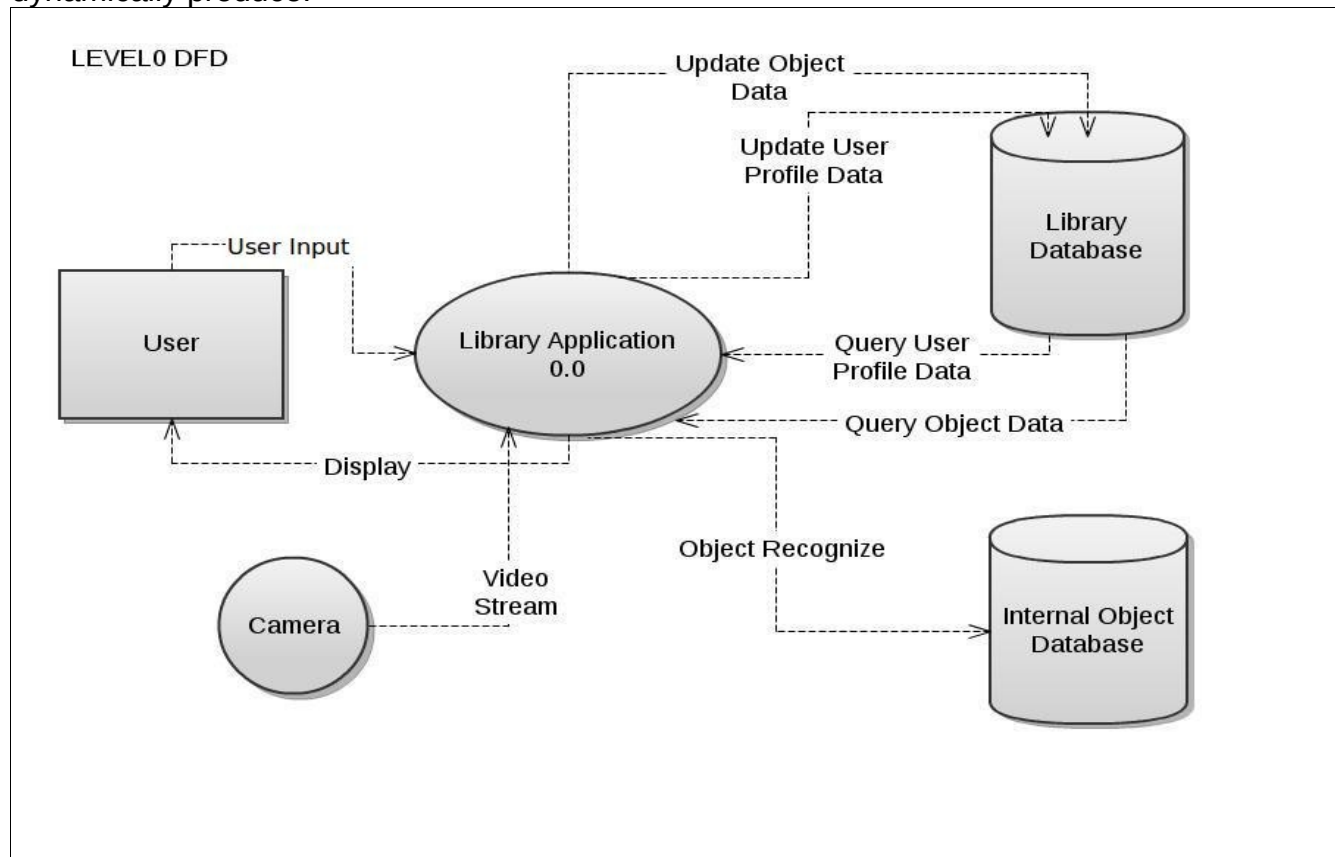

Diagram 3.1 – Level 0 Data flow diagram

In Level1, the library application splits into 4 units; namely Input Controller, Authentication, Graphical User Interface Unit and Object Recognizer Unit. User interacts with the Input controller Unit by providing user input. If the user start to use **Bookwiser** for the first time, then it will go through the Authentication Unit. The authentication unit wants some personal information from the user and the user determines a user-name and a password. Moreover the authentication unit interacts with the library database in order to update personal information.

After these registration processes, the user can start to use the object detection part for the purpose of finding book s/he wants or s/he likes. Firstly, video stream information coming from the external system, namely camera, the Object Recognizer Unit starts to capture data from this video stream. This process just begins with the user's button click in order to use the camera.
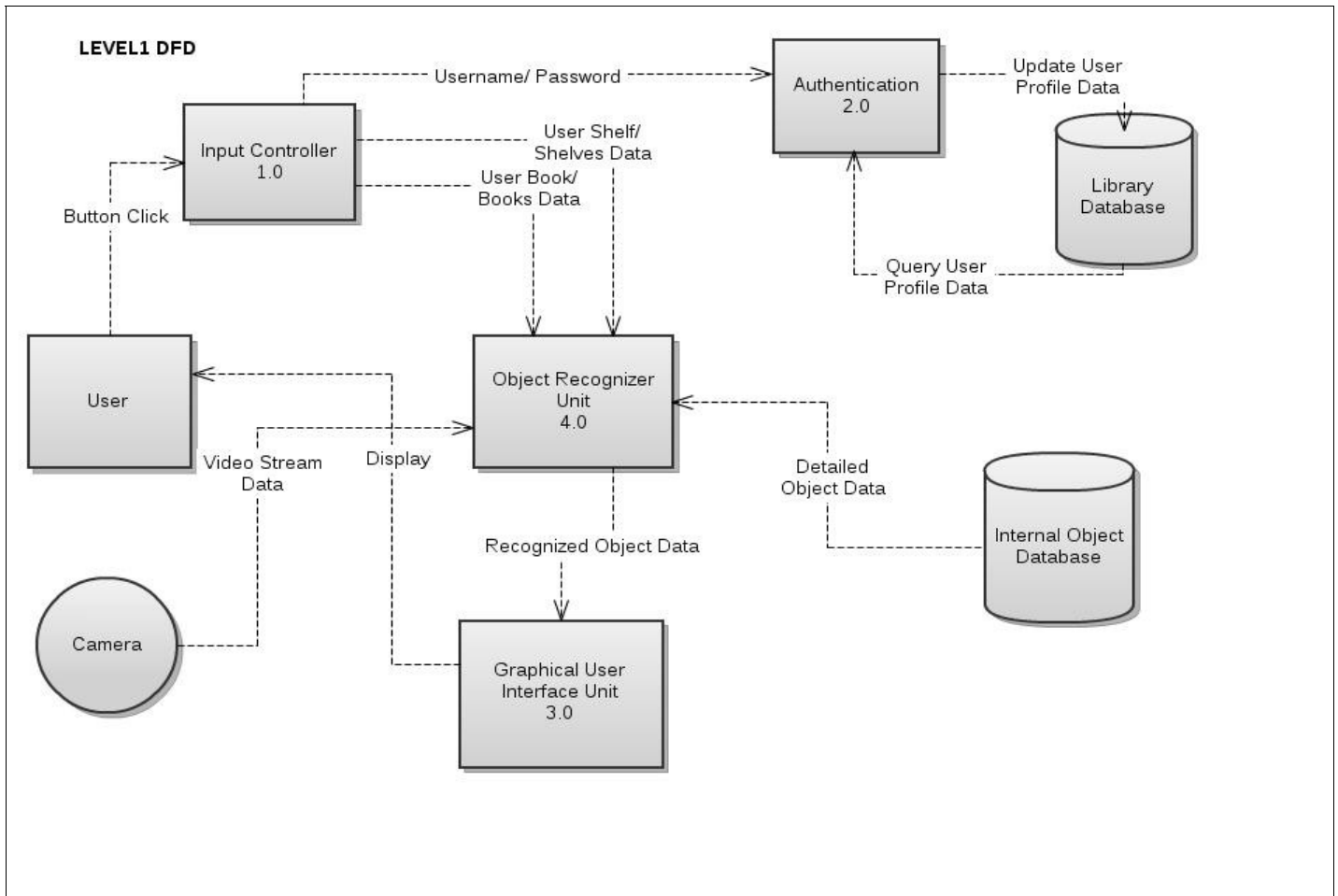
Diagram 3.2 – Level 1 Data flow diagram

In the objects recognizer unit, whenever the user wants to see detailed information about a shelf/ a group of shelves, or a book/ group of books, or other information about book unit and shelf unit, the object recognizer unit gets the detailed object information from the internal object database. After the the object detection, it will be displayed in the Graphical User Interface Unit to the user. The user generally interacts with the Graphical User Interface Unit about the flow of **Bookwiser**.

3.2 Functional Requirements

In this section, we will explain the major functions of **Bookwiser** along with the data flow. So that, Input Controller Unit, Authentication Unit, Graphical User Interface Unit, and Object Recognizer Unit are explained by their major functions with data flow.

Requirement 1:
In free walk mode , Bookwiser detects if theres a shelve / group of shelves in the current frame from a distance of 3-4 meters, and shows user the information below:

-Genre of books in the shelve
-List of highest rated books

Requirement 2:
In book search mode , Bookwiser detects if theres a shelve / group of shelves in the current frame from a distance of 3-4 meters and shows user if the searched book is in one of the detected shelves.

Requirement 3:
In free walk mode , Bookwiser detects if theres a book / group of books in the current frame from a distance of 2-2,5 meters and shows user the information below:

-Name of the book
-Rate of the book
-Author information of the book
-Reservation options
-Rating options

Requirement 4:
In search mode , Bookwiser detects if theres a book / group of books in the current frame from a distance of 2-2.5 meters and shows user if one of the books is the book the user is looking for.

Requirement 5:
In free walk mode , Bookwiser detects if theres a librarian in the current frame from a distance of 3,5-4 meters and shows user the information below:

-Information about the librarian

3.2.1 Input Controller Unit

Input controller unit controls all the input coming from the user and decides the flow of the application as explained in the Level 2 data flow diagram.
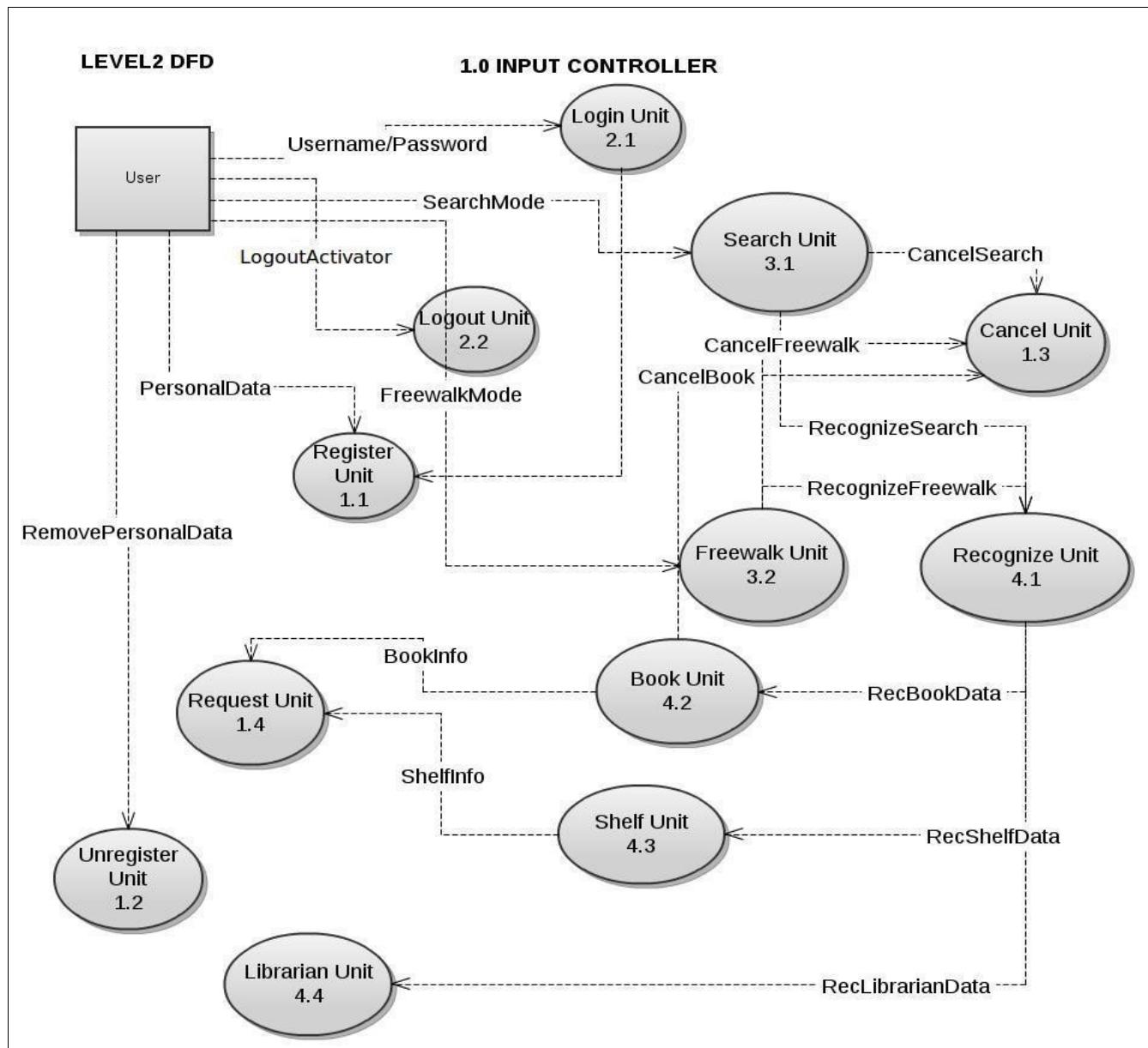
Diagram 3.3 – Level 2 Data flow diagram

Name	Username/Password
From	User
To	Login Unit 2.1
Description	The login input from the user.

Name	LogoutActivator
From	User
To	Logout Unit 2.2
Description	The logout input from the user.

Name	PersonalData
From	User
To	Register Unit 1.1
Description	The registration input from the user.

Name	RemovePersonalData
From	User
To	Unregister Unit 1.2
Description	The unregisration input from user..

Name	SearchMode
From	User
To	Search Unit 3.1
Description	The user selection input of search mode.

Name	FreewalkMode
From	User
To	Freewalk Unit 3.2
Description	The user selection input of freewalk mode.

Name	CancelSearch
From	Search Unit 3.1
To	Cancel Unit 1.3
Description	The user input to cancel the search mode.

Name	CancelFreewalk
From	Freewalk Unit 3.2
To	Cancel Unit 1.3
Description	The user input to cancel the freewalk mode.

Name	RecognizeSearch
From	Search Unit 3.1
To	Recognize Unit 4.1
Description	The data to object recognizer after the search mode selection.

Name	RecognizeFreewalk
From	Freewalk Unit 3.2
To	Recognize Unit 4.1
Description	The data to object recognizer after the freewalk mode selection.

Name	RecBookData
From	Recognize Unit 4.1
To	Book Unit 4.2
Description	The data after the recognition of the book.

Name	RecShelfData
From	Recognize Unit 4.1
To	Shelf Unit 4.3
Description	The data after the recognition of the shelf.

Name	ReclibrarianData
From	Recognize Unit 4.1
To	Librarian Unit 4.4
Description	The data after the recognition of the librarian.

Name	BookInfo
From	Book Unit 4.2
To	Request Unit 1.4
Description	After the detection of book unit (consisting of a book / a group of books) , the user requests a number of actions .

Name	ShelfInfo
From	Shelf Unit 4.3
To	Request Unit 1.4
Description	After the detection of shelf unit (consisting of a shelf/ a group of shelves) , the user requests a number of actions .

3.2.2 Authentication Unit

Authentication unit handles the login, register operations interacting with the user and the library database as shown in the Level 2 data flow diagram.
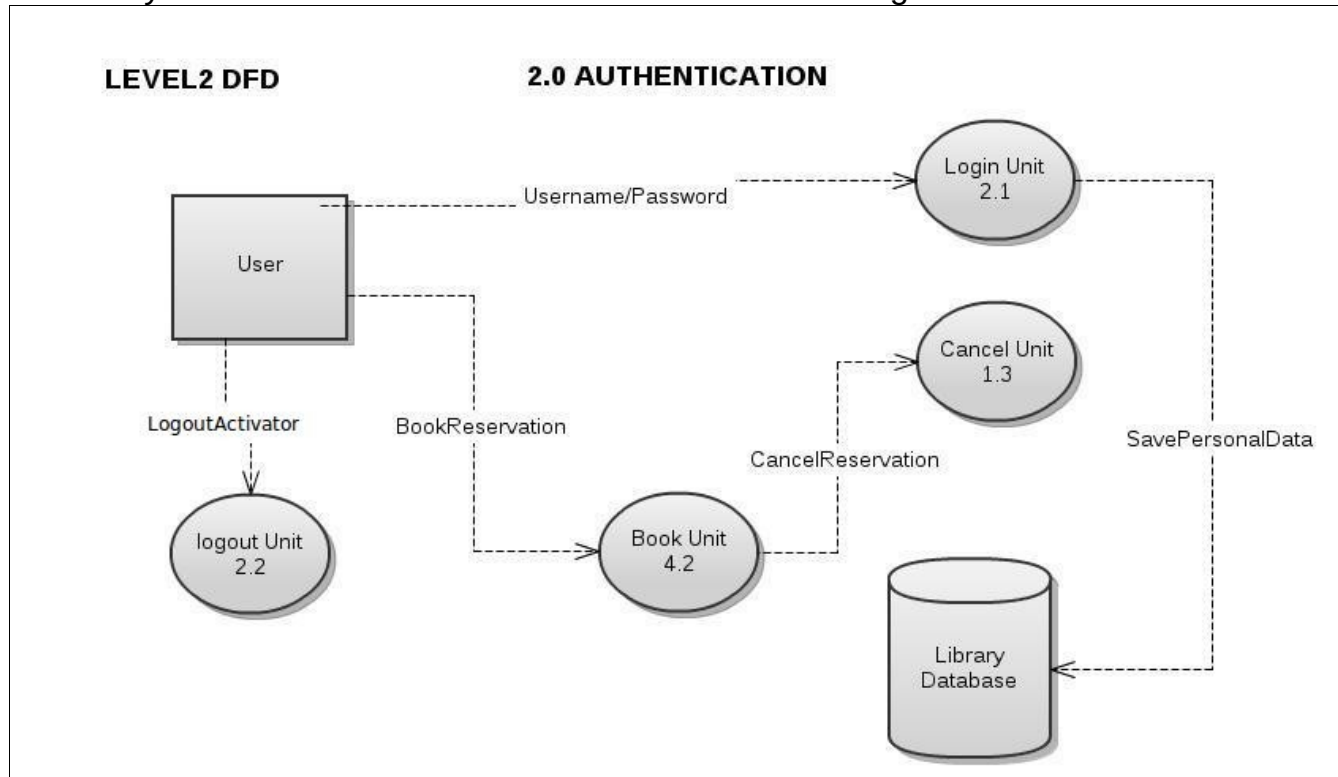
Diagram 3.4 – Level 2 Dataflow diagram.

Name	Username/Password
From	User
To	Login Unit 2.1
Description	User data in order to authenticate to the system.

Name	LogoutActivator
From	User
To	Logout Unit 2.2
Description	User data in order to logout from the system.

Name	BookReservation
From	User
To	Book Unit 4.2
Description	User data for the reserve a book for a time period.

Name	CancelReservation
From	Book Unit 4.2
To	Cancel Unit 1.3
Description	User data for unreserve the book when the user's work is finished with that book.

Name	SavePersonaData
From	Login Unit 2.1
To	Library Database
Description	Data loaded in library database when a user logs in to the system

3.2.3 Graphical User Interface Unit

Graphical User Interface Unit handles the information shown to the user according to the input coming from the user as explained in the diagram below.
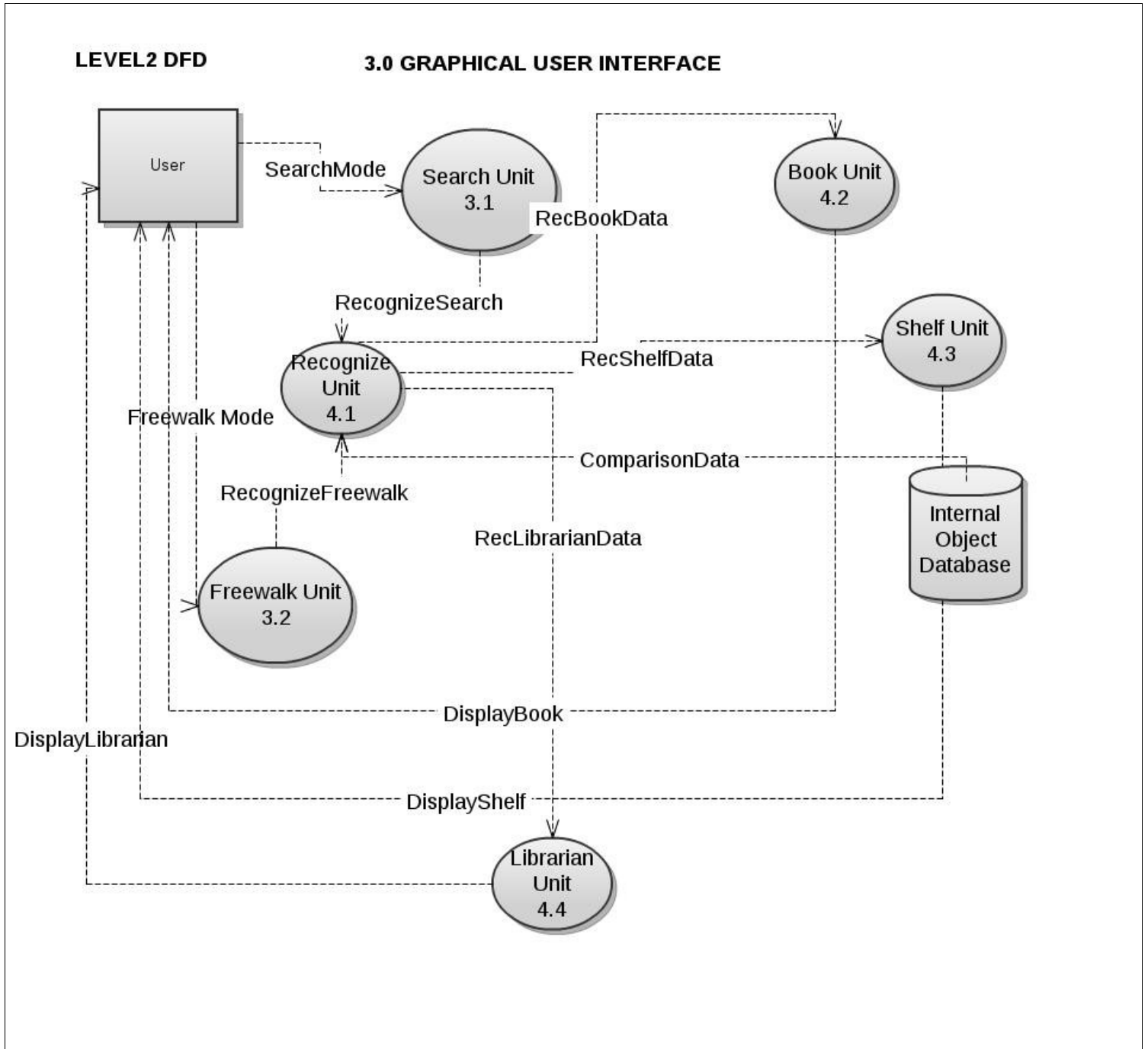

Diagram 3.5 – Level 2 Data flow diagram

Name	SearchMode
From	User
To	Search Unit 3.1
Description	The data when the user selects the search mode in order to find a specific book s/he wants.

Name	FreewalkMode
From	User
To	Freewalk Unit 3.2
Description	The data when the user selects the freewalk mode in order to walk around and be advised by the application that s/he can like.

Name	RecognizeSearch
From	Search Unit 3.1
To	Recognize Unit 4.1
Description	Data after the search mode option is chosen by the user.

Name	RecognizeFreewalk
From	Freewalk Unit 3.2
To	Recognize Unit 4.1
Description	Data after the freewalk mode option is chosen by the user.

Name	RecBookData
From	Recognize Unit 4.1
To	Book Unit 4.2
Description	Data after the book/a group of books detected by the user.

Name	RecShelfData
From	Recognize Unit 4.1
To	Shelf Unit 4.3
Description	Data after the shelf/a group of shelves detected by the user.

Name	RecLibrarianData
From	Recognize Unit 4.1
To	Librarian Unit 4.3
Description	Data after the librarian detected by the user.

Name	ComparisonData
From	Recognize Unit 4.1
To	Internal Object Database
Description	Mapping objects with the equivalents in the library is processed for definition of detected objects.

Name	DisplayBook
From	Book Unit 4.2
To	User
Description	After the detection and the processes about book unit is done, data is displayed to the user.

Name	DisplayShelf
From	Shelf Unit 4.3
To	User
Description	After the detection and the processes about shelf unit is done, data is displayed to the user.

Name	DisplayLibrarian
From	Librarian Unit 4.4
To	User
Description	After the detection of unit is done, data is displayed to the user.

3.2.4 Object Recognizer Unit

Object recognizer unit handles the most crucial part of the application including all the object recognition algorithms and methods, data flow of the unit is explained in the below data flow diagram.

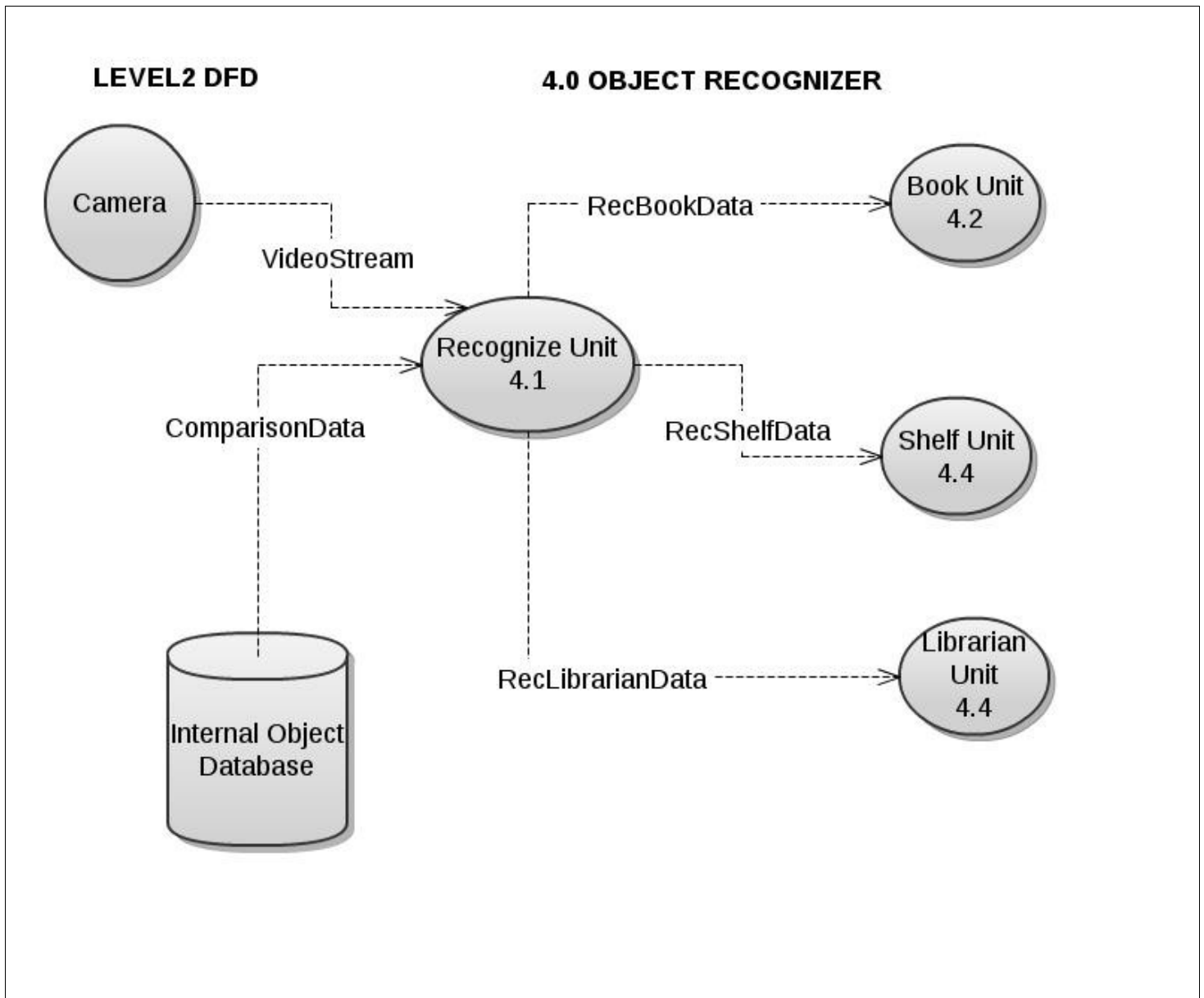**LEVEL2 DFD**   **4.0 OBJECT RECOGNIZER**

Diagram 3.6 – Level 2 Data flow diagram

Name	VideoStream
From	Camera
To	Recognize Unit 4.1
Description	Data coming from camera to be able to start detection.

33

Name	ComparisonData
From	Internal Object Database
To	Recognize Unit 4.1
Description	Mapping objects with the equivalents in the library is processed for definition of detected objects.

Name	RecBookData
From	Recognize Unit 4.1
To	Book Unit 4.2
Description	After detection of a book/a group of books, data is sent to book unit to satisfy user's desires.

Name	RecShelfData
From	Recognize Unit 4.1
To	Shelf Unit 4.3
Description	After detection of a shelf/a group of shelves, data is sent to shelf unit to satisfy user's desires.

Name	RecLibrarian
From	Recognize Unit 4.1
To	Librarian Unit 4.4
Description	After detection of the librarian, data is sent to librarian unit (Then it is displayed on the screen).

3.3 NON-FUNCTIONAL REQUIREMENTS

3.3.1 PERFORMANCE REQUIREMENTS

In the **Bookwiser**, application it is assumed that recognition process is done in near-rela time. So that, the efficiency of the application is quite high.

- **Number of users supported:**
  -There is no restriction about the number of users to be supported in the application. Since the number of users is not relevant with how the **Bookwiser** works, the number of users that supported can be variable according to the library administration.

- **Recognition time:**
  - Since this is an augmented reality project, all processes are handled in near-real-time. The use of library database is provided only when the user registers to the system and wants to reserve a book or the system requests detailed information about a recognized object .

  - The object recognition mechanism of the application is independent from library database part. There is an internal object database in **Bookwiser**. However, it is <u>not</u> in an actual database form. Internal object database is a dynamic database that during the process of object detection, the objects shown on camera is dynamically compared with the equivalent objects in database. A number of specifications determined for objects are in the internal object database.

  - Since external database access is only handled whenever the object is detected, **Bookwiser's** object recognition should work efficiently.

  - The information processed in the **Bookwiser** will be a video stream. In a 1 second of a video stream, there are 24 frames (images). To achieve near-real-time goal , **Bookwiser**'s aim is to respond to the user in near 1 second. Amount of the information to be processed is limited with frames of the video stream and amount of pre-defined information in the object recognition database.

3.3.2 DESIGN CONSTRAINTS

- **Language:**
  - The system is an application designed to work on portable devices.

  - For this purpose, a programming language that works on a portable platform is appropriate, C++ and/or Java will be used.

  - For object recognition , a computer vision library that supports the chosen language is

required , OpenCV is a suitable library .

- **Hardware Constraints:**
  - The application is assumed to require 5.0 mega-pixel camera in order to be able to detect the objects in the library environment.

  - A minimum of 2.0 GHz Intel processor and a 500 core GPU is required for the system.

  - The system should be able to communicate with the library database via a wireless device.

- **Software System Attributes:** There are 4 attributes that **Bookwiser** holds.

1. **Usability:**

   - The user should be able to see information about books in a fast and without touching the book itself and also to find what they want   easily with the help of **Bookwiser**.

   - Moreover, the program should be user-friendly ; after the registration, user interacts with only a button and a screen. No more extra effort is needed in order to interact with **Bookwiser**.

2. **Reliability:**

   - Object recognition process should be in real-time and reliable.

   - **Bookwiser** should display the consequences of detections correctly.

   - The object detection algorithms should be tested to give correct results to not to misguide the users.

   - Apart from that **Bookwiser** <u>is not</u> responsible of the correctness of the information about books in the library database.

3. **Security:**

   - Library database keeps profiles of registered users and they should be kept in the library database which is assumed to be secure , safe and apart from the application and all the information transfer between **Bookwiser** and the library database is done via an encryption system which protects the system from illegal access.

   - **Bookwiser** uses an authentication system for security. Since people need to authenticate to the system by giving correct information about themselves, only known users will use **Bookwiser** and performed actions of users can be monitored.

- Except user information and book reservation data, the application does not change the information kept in library database so no action in **Bookwiser** may harm the library database.

4. **Portability:**

- Bookwiser is designed to work on portable devices , also the system is independent of the type of the database the library uses therefore the system **Bookwiser** uses can be used in any library without changing the library's own systems completely.

- Because object recognition is done via an internal object recognition database, also objects defined in the database should be modifiable with respect to the object-type used in different library buildings.

4 Data Model and Description

4.1 Data Objects

This section describes the tables and their properties which will be used in **Bookwiser** system. The reason behind the fact that the data should be categorized into specific classes is to improve the understandability of the project by notifying the library environment objects.

Enrollees

This table is needed in order to keep some basic information for library enrollees registered to the **Bookwiser** system. There is only very necessary information about a library enrollee.

The UserId is the primary key of Enrollees table. There is Name, SurName attributes to hold the information about the name and surname of the enrollee . The Telephone, Address and MailAddress attributes determine the connection information for the enrollee. There is also Job and Sex attributes to provide information for some surveys in case of a need. For example: What genre of books does the doctors prefer mostly?

- UserId
- Name
- SurName
- Job
- Gender
- Telephone
- Address
- MailAdress

Users

This table is definitely needed in order to keep the information of the **Bookwiser** Software System users in a safe manner. The primary key of User table is ID, it corresponds the UserId of Enrollees table. The PrefferedLanguage field stores the desired language of the user to use the Bookwiser System.

- ID
- Password
- PrefferedLanguage

Books

This table corresponds to the books in the library, it holds the information about the books in the library. The BookID attribute is the primary key of the Books table, it is unique to each book. BookName defines the name of the book, where Author represents the author of the book. Rating stores the rating of a book. The BriefText field of the Book stores little information about the book.

- BookID
- BookName
- Author
- Rating
- BriefText

Shelfs

This table is needed to represent the shelfs in the library, it holds the information about the shelfs in the library. The ShelfID is the primary key of Shelfs table, and Genre is the genre of books that shell contains. Note that, one shell can contain only one genre of books.

- ShelfID
- Genre

Librarians

This table is needed to represent the librarians working in the library. The LibrarianID is the primary key of the Librarians table. The Name and SurName correspond to the name and surname of the librarian respectively.

- LibrarianID
- Name
- SurName

4.2 Relationships

The relationships between the tables in the database are explained in this section.

FavoriteBooksOfUser

We have Books and Users. When a user rates a book highly it is added to the favorite book of that user. It is critically needed to determine the favorite books of the user. The related two attributes are namely;

- BookID from Books

- ID from Users

RecommendedBooksforUser

We have Books and Users. The books that a user whether like or not can be guessed by some search on his favorite books, etc. Then, the respective books and the user can be put in a relationship. These two attributes to constrain the relationship are relatively ;

- BookID from Books

- ID from Users

BookInShelf

We have Books and Shelfs. The fact that a book stands on a specific shell is a relationship,and with the information of the shelf that the book lays on, he genre of book also can be determined. So the attributes that belong to that relationship are namely;

- BookID from Books
- ShelfID from Shelfs

4.3 E/R Diagrams

This section includes the E/R diagrams of the tables that the Bookwiser Software System is going to use from the database table.
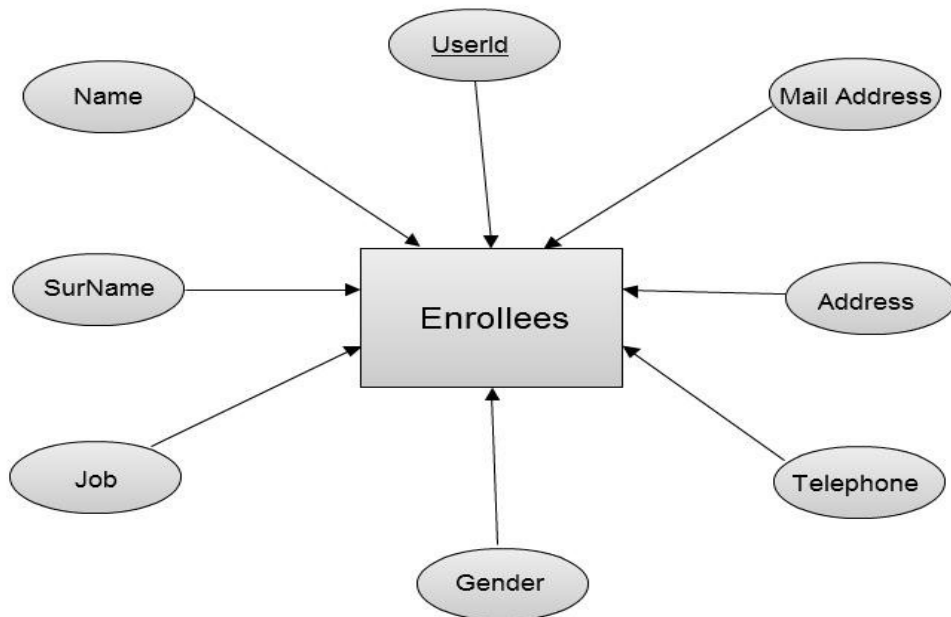
Enrollees

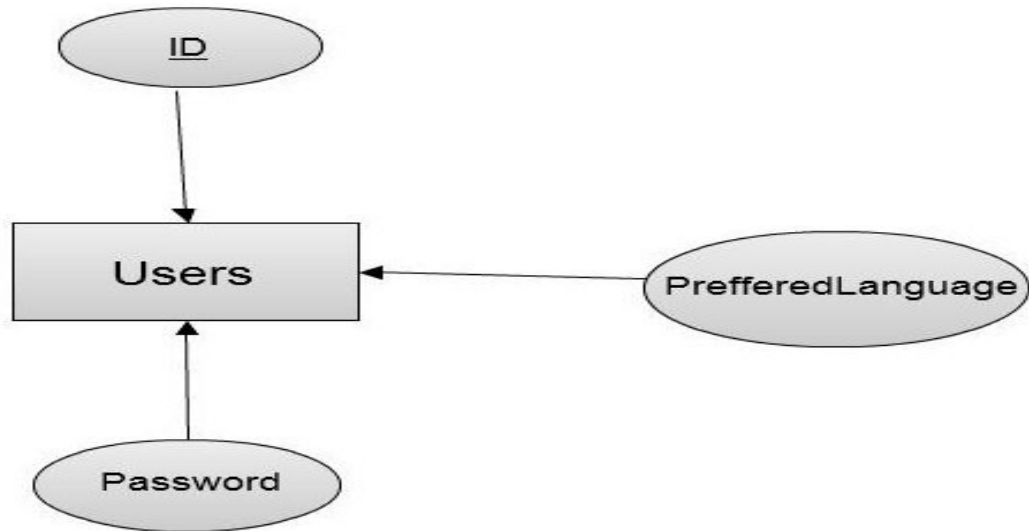**Diagram 4.1 The Enrollees E/R Diagram**

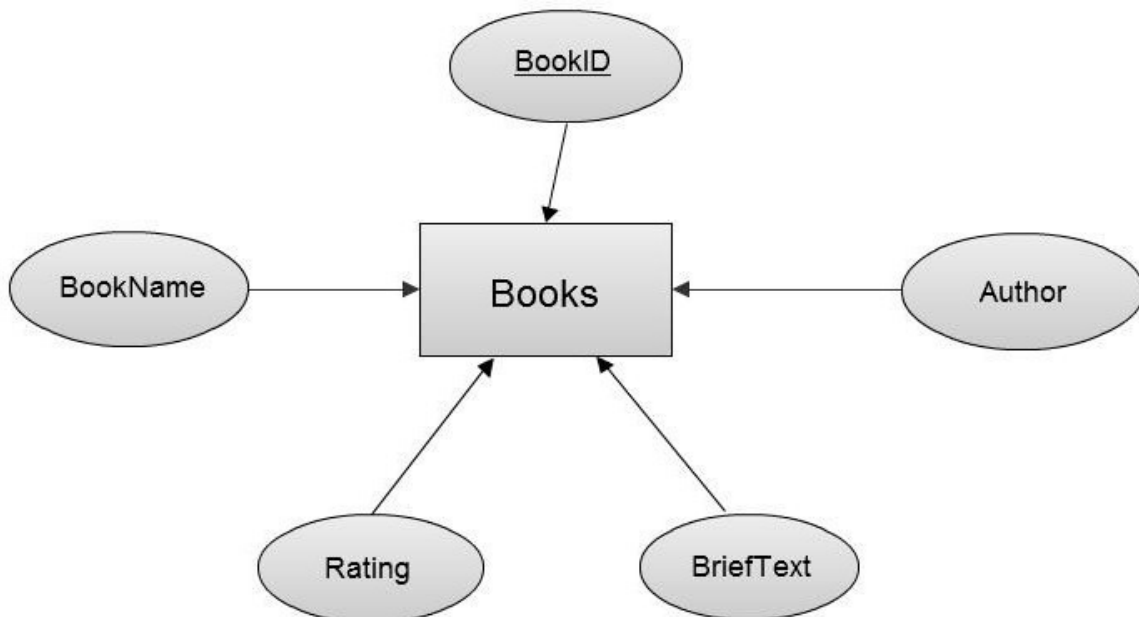**Users**

Diagram 4.2 - The Users E/R Diagram

Books

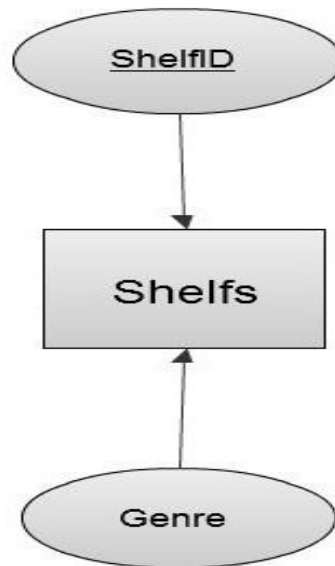**Diagram 4.3 The Books E/R Diagram**

Shelfs

Diagram 4.4 The Shelfs E/R Diagram
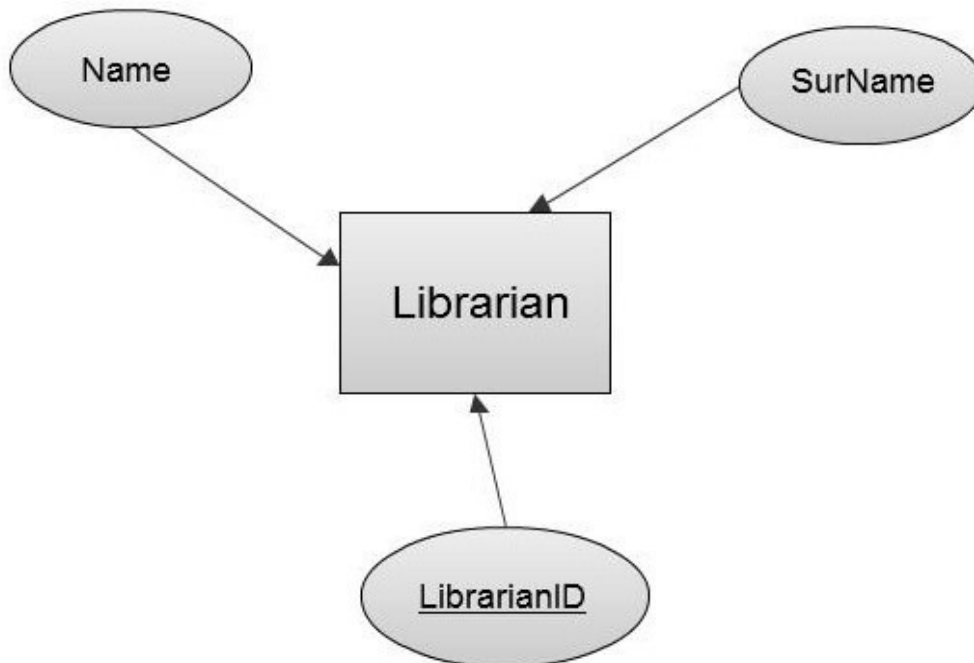
Librarians

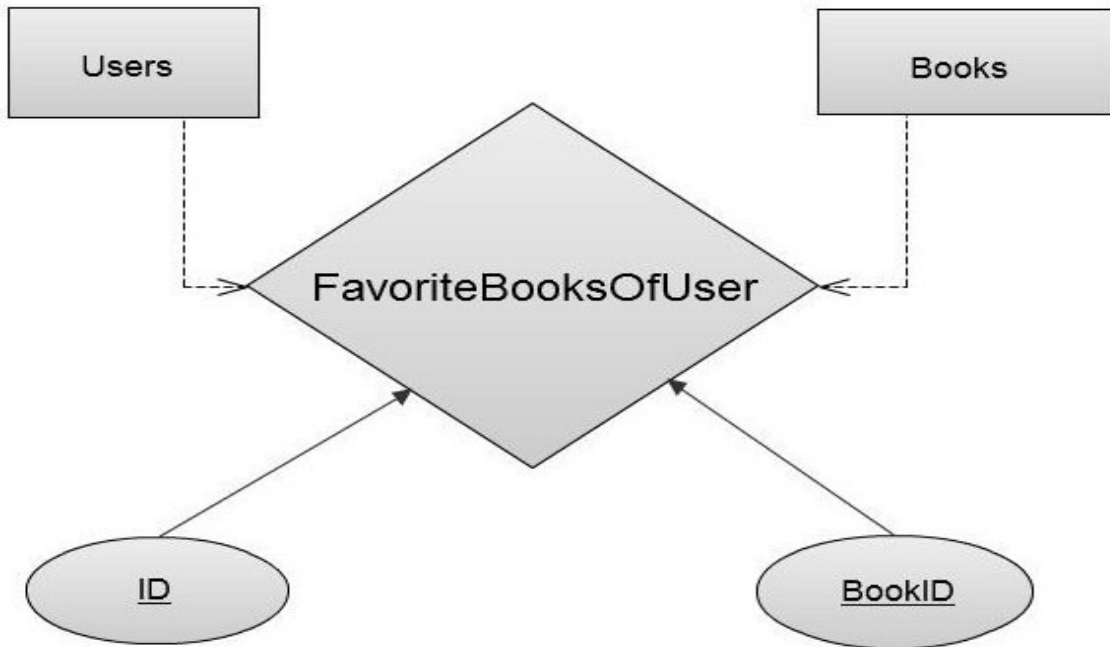**Diagram 4.5 The Librarians E/R Diagram**

FavoriteBooksOfUser

Diagram 4.6 The FavoriteBooksOfUser E/R Diagram
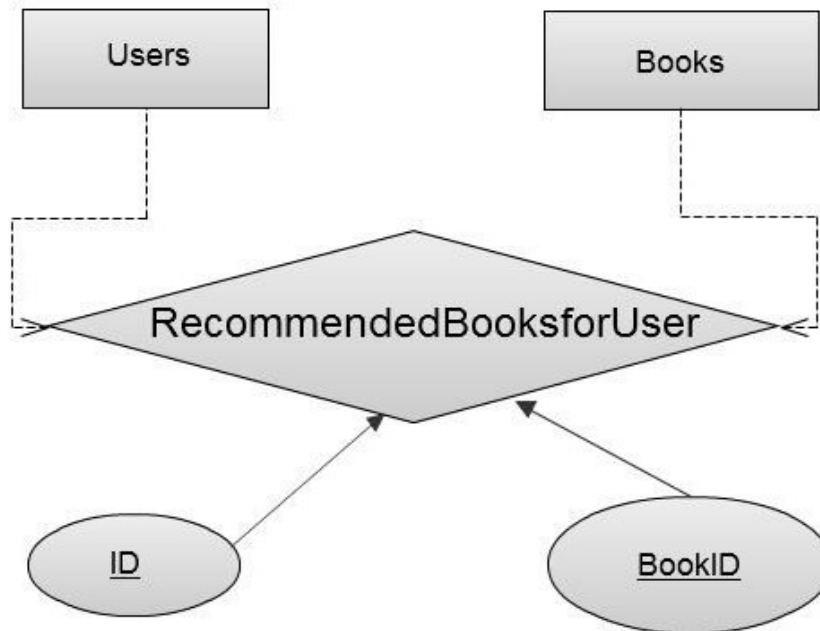
RecommendedBooksForUser

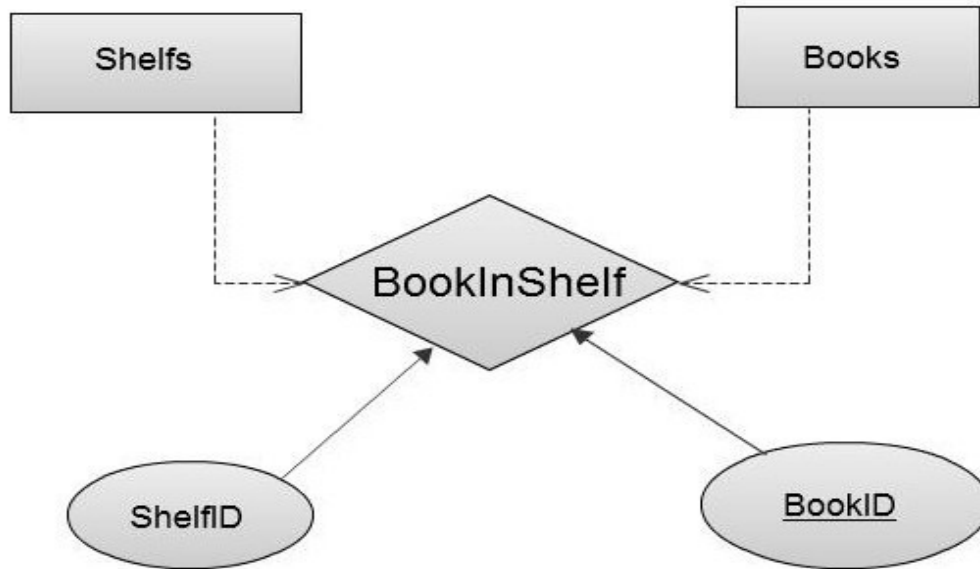**Diagram 4.6 The RecommendedBooksOfUser E/R Diagram**

BookInShelf

Diagram 4.7 The BookInShelf E/R Diagram

5 Behavioral Model and Description

Behavior of **Bookwiser** is based on user actions. For any event which user triggers, there is a state transition. Each state which **Bookwiser** operates ,is determined by the user's selections and movement. User actions triggering a transition can be either an input or physical movement of camera which causes a change in camera view.

All the states along with their particular transitions are given in 5.1 and overall state diagram is given in 5.2.

5.1 Description of the Software Behavior

1.name	Login
2.entry actions	Opening the Bookwiser
3.exit actions	Entering login information
4.internal transitions	None
5.explanations	For activating Bookwiser, user should enter login information in this state.

1.name	Logout
2.entry actions	User decides to exit the system
3.exit actions	None
4.internal transitions	None
5.explanations	Logout is in fact the final state of BookWiser. When user wants to finish to use BookWiser he/she is directed to logout.

1.name	FreeWalk
2.entry actions	Entering login Info, unselecting search mode, moving camera out of recognition range
3.exit actions	Entering logout state , recognizing shelf or book, search mode
4.internal transitions	Librarian detection option is provided in this state
5.explanations	After login, user is directed to FreeWalk automatically. In this mode recognition is possible. User can also to get in SearchMode.

1.name	SearchBook
2.entry actions	Selecting search mode, moving camera out of recognition range provided that book search is active
3.exit actions	Logout state, recognizing shelf or book, exiting search mode
4.internal transitions	Library detection option is provided in this state
5.explanations	Bookwiser in this state informs user when it detects the selected book

1.name	RecognizeBook
2.entry actions	Entering to the book recognition range, unselecting book
3.exit actions	Selecting book, moving camera out of recognition range
4.internal transitions	None
5.explanations	In this state, user is given some information about recognized book/books.

1.name	RecognizeShelf
2.entry actions	Entering to the shelf recognition range, unselecting shelf
3.exit actions	Shelf selection, moving camera out of recognition range
4.internal transitions	None
5.explanations	In this state, user is given some information about recognized shelf/shelves.

1.name	BookSelected
2.entry actions	Selecting book, exiting information gathering state
3.exit actions	Unselecting book, requesting particular book information
4.internal transitions	None
5.explanations	User can be able to see various information about the object. He/she also can request specific information and reserve the book.

1.name	ShelfSelected
2.entry actions	Shelf selection
3.exit actions	Shelf deselection
4.internal transitions	None
5.explanations	User can be able to see various information about selected shelf in this state.

1.name	RateBook
2.entry actions	Rating state
3.exit actions	Exiting rating state
4.internal transitions	None
5.explanations	This state is where rating is done.

1.name	GetDetailedInfo
2.entry actions	Requesting detailed information about the book
3.exit actions	Exiting information gathering state
4.internal transitions	None
5.explanations	Detailed information described in sec 2.2 is displayed

1.name	ReserveBook
2.entry actions	Reservation state
3.exit actions	Exiting reservation state
4.internal transitions	User can invoke extend reservation state and can invoke cancel reservation state.
5.explanations	In reservation state, user can be able to modify his/her reservations.

1.name	ListAuthorBooks
2.entry actions	Requesting the list of books of the author
3.exit actions	Exiting information gathering state
4.internal transitions	None
5.explanations	A list of books of the author is given to the user.

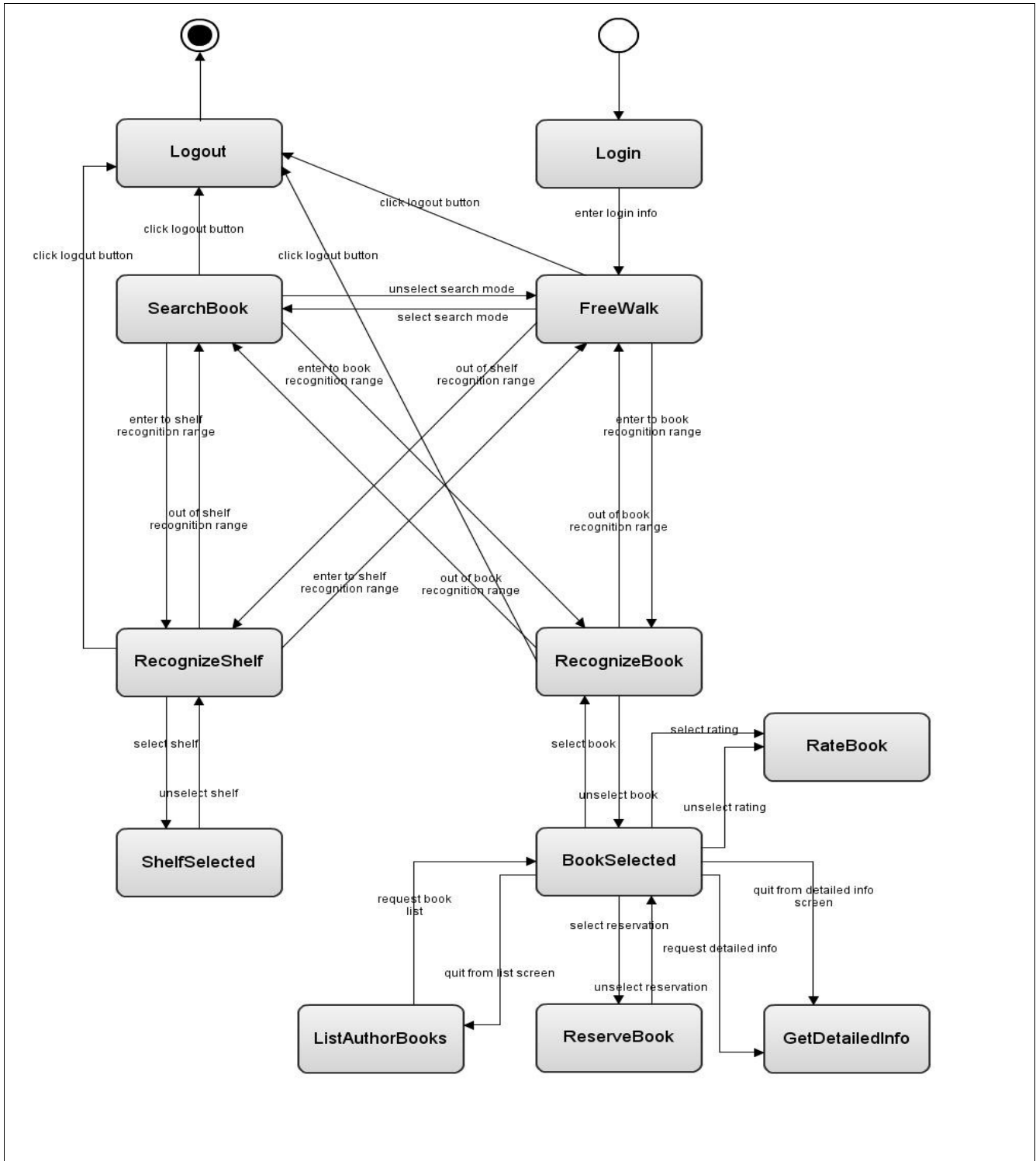## 5.2 State Transition Diagram

Diagram 5.1 − State transition diagram

6 Planning

6.1 Team Structure

The Bookwiser project mainly consists of three parts; object recognition, GUI part and maintenance of database. We mainly share the project according to the first two parts, respectively GUI and object recognition. Since the object recognition is the most difficult and challenging component of our project , the three of the team is dealing with it, namely Batuhan Karagöz, Ozan Tabak & Betül Dolay. The GUI part is assigned to Elif Alev Ahmet. The database maintenance part of the BookWiser System is not committed to any specific person, it is handled by division of work.

As a team, we are in contact all the time, we meet continuously in regular periods since the communication and coming together to discuss what is best for the project is really critical for the project. There is no team leader in project, we emphasize on each idea of each team member and progress according to the best of them. Each member is assigned with a duty and is responsible for that, and the other three members can check on him/her.

6.2 Estimation

FP Based Estimation

	Simple	Count	Average	Count	Complex	Count	Total
Inputs	3	4	4	1	6	1	22
Outputs	4	3	5	2	7	4	50
Inquiries	3	1	4	3	6	2	27
Internal Files	7	0	10	0	15	2	30
External Interfaces	5	0	7	3	10	1	31

Count TOTAL = 160

Inputs

- Simple Inputs : Login, BookName for Searched Book, Reservation, Rating
- Average Inputs : Register
- Complex Inputs : VideoStream

Outputs

- Simple Outputs: Geometric Indicator, Librarian Info, Screen Menu Entries
- Average Outputs: Undetailed Book Information, Undetailed Shelf Information
- Complex Outputs: Detailed Book Book Information, Detailed Shell Information, Screen Image, Help Document

Inquiries

- Simple Inquiries: Requesting reservation
- Average Inquiries : Requesting detailed book info, Requesting detailed shelf info, Requesting the other books of the author.
- Complex Inquiries: Searching for a book mode, Free walk mode

Internal Files

- Complex Internal Files: The Information-File for Book Recognition, The Information-File for Shelf Recognition

External Interfaces

- Average External Interfaces: Register, Login, Help
- Complex External Interfaces: Library-System Interface

Questions	Scale
Does the system require reliable backup and recovery?	3
Are data communications required?	5
Are there distributed processing functions?	2
Is performance critical?	5
Will the system run in an existing, heavily utilized operational environment?	5
Does the system require on-line data entry?	5
Does the on-line data entry require the input transaction to be built over multiple screens or operations?	3
Are the master files updated on-line?	1
Are the inputs, outputs, files, or inquires complex?	5
Is the internal processing complex?	5
Is the code designed to be reusable?	4
Are conversion and installation included in the design?	5
Is the system designed for multiple installations in different organizations?	5
Is the application designed to facilitate change and ease of use by the user?	5
TOTAL	58

FP=160*(0.65+0.01+58)=196.8

Since we are using object-oriented programming we are going to hold the LOC 30 per FP.

COCOMO

Effort = 3.0*5.907^1.12=21.9 month*person
Time = 2.5*(21.9)^0.35=7.37 month
Personnel = 21.9/7.37=2.97=3

Each person works for 3000 TL per month, so the project is 21.9* 3000 = 65700 TL

❶		Task Name	Duration	Start	Finish
1	▦	Literature Survey	8 days?	Mon 11.10.10	Wed 20.10.10
2	▦	General Design	11 days?	Fri 15.10.10	Fri 29.10.10
3	▦	Database Design	5 days?	Mon 01.11.10	Fri 05.11.10
4	▦	Library Database Entegration	4 days?	Mon 08.11.10	Thu 11.11.10
5	▦	Basic User Interface Impleme	4 days?	Fri 05.11.10	Wed 10.11.10
6	▦	Authentication	3 days?	Thu 11.11.10	Mon 15.11.10
7	▦	Shelf Recognition	5 days?	Tue 16.11.10	Mon 22.11.10
8	▦	Book Recognition	10 days?	Tue 16.11.10	Mon 29.11.10
9	▦	Librarian Recognition	6 days?	Mon 22.11.10	Mon 29.11.10
10	▦	Library Network Interface De	5 days?	Tue 30.11.10	Mon 06.12.10
11	▦	User Interface Development	8 days?	Wed 01.12.10	Fri 10.12.10
12		Team Presentation	1 day	Mon 13.12.10	Mon 13.12.10
13	▦	Demo Release	3 days?	Tue 14.12.10	Thu 16.12.10
14	▦	Demo Test	4 days?	Fri 17.12.10	Wed 22.12.10
15	▦	Project Improvement	7 days	Thu 23.12.10	Fri 31.12.10
16	▦	Product Release	5 days?	Mon 03.01.11	Fri 07.01.11
17	▦	Product Test	4 days?	Mon 10.01.11	Thu 13.01.11

6.3 Process Model

We are going to progress the project according to an incremental process model. After then specifying the requirements and design of system, we are going to implement the prototype of the project. Thereafter, we will increasingly develop the Bookwiser Software System. We decided that Spiral Model best fits to Bookwiser Project.

The project is going to be progressed according to an incremental process model. After then specifying the requirements and design of system, the prototype of the project is going to be implemented. Thereafter, the **Bookwiser** Software System will increasingly be developed. It is decided that Spiral Model best fits to **Bookwiser** Project.

7 Conclusion

Bookwiser is a valuable project that will both serve the users and the owners of the system in a positive way, being a innovative idea and aiming to create a more comfortable environment for its users.

The project should be developed in a complete understanding of the facts behind the need for the project, the importance of the results of the project and should be considered as a part of the instinct that creates the need of progress.

Requirement analysis is very critical for the project to guide the developers in a definitive way. This document gives a determined idea and perspective for the **Bookwiser** project defining all the methods, structures, limitations , requirements and planning to guide the developer through all the development process.

All the specifications stated in this document is the result of a mixture containing innovative thinking, analytic approach and a proper analysis of users' needs; therefore , developers should not forget that these specifications have a critical importance for all elements of all levels in the project as a whole.

All the requirements stated in this document should give the developers a mainframe.