



BBM487 -Software Engineering Laboratory

Library Book Loan System Architecture Notebook

Group 4:

Ali Burak ÖZAKINCI - 21427162

Sabit ÇATALTAŞ - 21692544

Aishat Temitope OLOMOWEWE – 21603775

| | |
|--------------------------|--------------------|
| Library book loan system | Group 4 |
| Architecture Notebook | Date: <31/03/2017> |

Library Book Loan System Architecture Notebook

1. Purpose

The key purpose is to enable the addition of features without major effort, the ability to sustain changes in general and ensure the project is built on this architecture remains future-proof all along its life cycle.

2. Architectural goals and philosophy

The library book loaning system will be driven by complex deployment concerns so to master the rising complexity, as well as automating development task. It does not need to be robust for long-term maintenance.

The goals that the project is set to achieve is to deliver high quality product, better code reuse hence faster software development and ease of use for the users.

3. Assumptions and dependencies

The assumption plays an important role in developing the risk management plan ,it is made that an interface will be available for the implementation,listed below are a number of assumptions.

Assumptions :

- Users wish to have the books they wanted, in a fast and reliable way.
- Users need a smooth experience,the implementation should be error free.
- The system should be user friendly so that it is easy to use for the users.
- System will serve minimum of 200 people,the storage must be big enough to deal with the throughput.
- The information of all the user, books and libraries must be stored in a database the is easily accesible
- There system should adequate ease of use and leaning,
- The projected maintenance which will be offered once a year would be delivered,
- The software would be able to adapt to different operating systems and environment.

Dependencies :

- There wont be any hardware associated with the system.
- On the basis of listing requirements and specifications ,the database will have its own functional audit trail on its own for backup and other operations.
- Any update regarding books or user registration should be recorded to the database and the data entered should be correct.
- The system should have its general report stored.

| | |
|---------------------------------|--------------------|
| Library book loan system | Group 4 |
| Architecture Notebook | Date: <31/03/2017> |

4. Architecturally significant requirements

- The system must deploy on Microsoft Windows XP and Linux.
- The system must encrypt passwords.
- The system must respond within 5 seconds on average.
- The system must record every modification to customer records and book records for audit purposes.
- Backups must be stored another disk other than current DBMS.
- Easy upgrade and maintenance the system.

5. Decisions, constraints, and justifications

- The system must respond within 3 seconds.
- The systems average startup time must be 1 minute.
- The systems average shutdown time must be 1 minute it can be up to 5 minutes if and only if it includes a backup operation.
- The system software must be up-to-date to technological developments and needs.
- The system must deploy on windows or linux environment.
- The system must always update the user on any late fine check and notification at 5pm.
- The system must be backed up on the second week of every month.
- Software can be run offline or online depending on the customer's order at the start.
- Online one has a cloud database, offline one has a standalone database with the software.

6. Architectural Mechanisms

Architectural Mechanisms are common solutions to common problems that can be used during development to minimize complexity.

Disaster Recovery:

Due to virus or system failure the library book loaning system has the tendency to crash ,therefore it is required that there is a database backup so that the data is not lost

Security and Abstraction:

The library book loaning system will use a secured database, library users can just read and make use of the system but cannot modify anything except from their personal information ,proper user identification would be provided inorder to gain access to the system, there are separate accounts for admin and library users ,librarians cannot modify user's late fine.

Availability:

System will be available all library working days except annual maintenance day.

Auditing:

All transactions and model manipulations are tracked and recorded.

Error Management:

Unexpected errors will be detected, propagated and reported.

Mega-Data:

Cloud database can handle the big data if needed.

Transaction Management:

PyMongo module provides a clean and concise interface for handling ACID transactions with database.

Printing:

Archived transactions can be printed to solve problems occurred or users can take hard copy of

| | |
|---------------------------------|--------------------|
| Library book loan system | Group 4 |
| Architecture Notebook | Date: <31/03/2017> |

their transaction history.

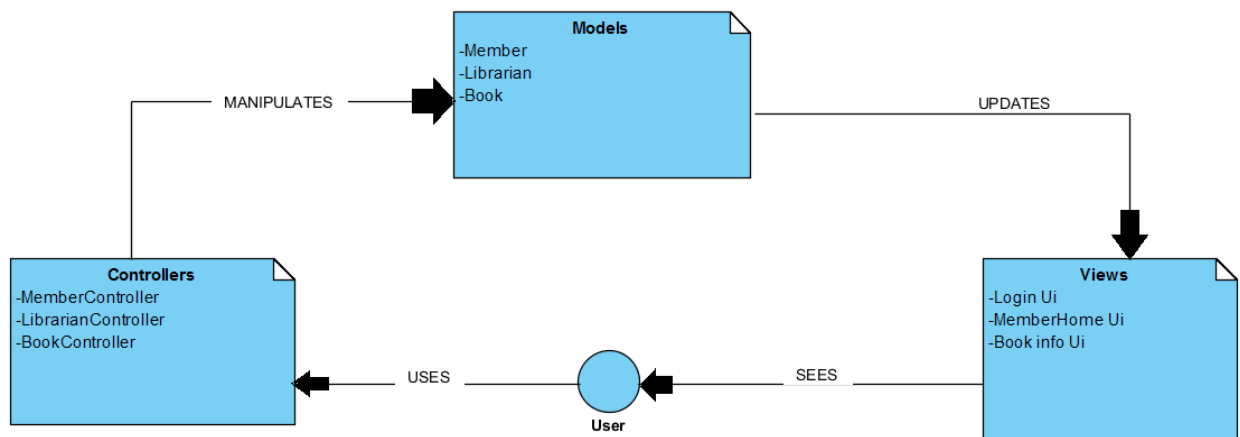
7. Key abstractions

- **Member class:** Members have id,name,surname,username,password,fine amount and email address.
- **Book class :** Books have a waiting list,title,id,barcode id,author info and publishing year.
- **Librarian class :** Has the same attributes as the member but with different privileges.

8. Layers or architectural framework

Library Loan Software will use 3 layered model-view-controller pattern.

- **Model:**Includes the data types that will take place within interactions.Book,User,Librarian data types and their manipulations included in the models layer.
- **View:**Includes interface elements and their instantianate methods.User flows through the different views to achieve the task he\she wants with the software.
- **Controller:**Provides the communication between model and view layers.Ui elements and models have controllers to provide functionality.

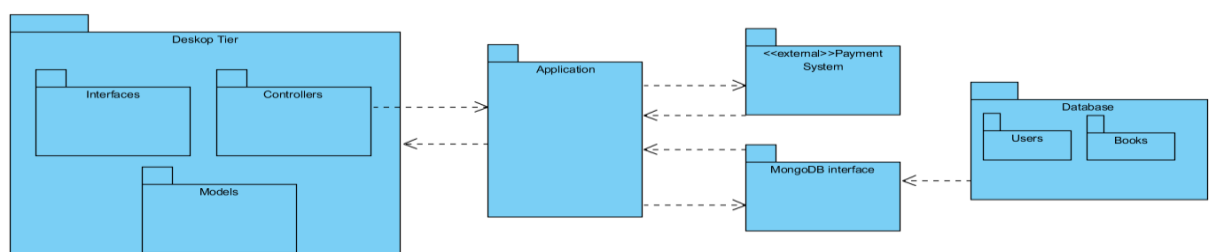


9. Architectural views

We used 4+1 Architectural model.It helps us cover the project by different angles with separate views.

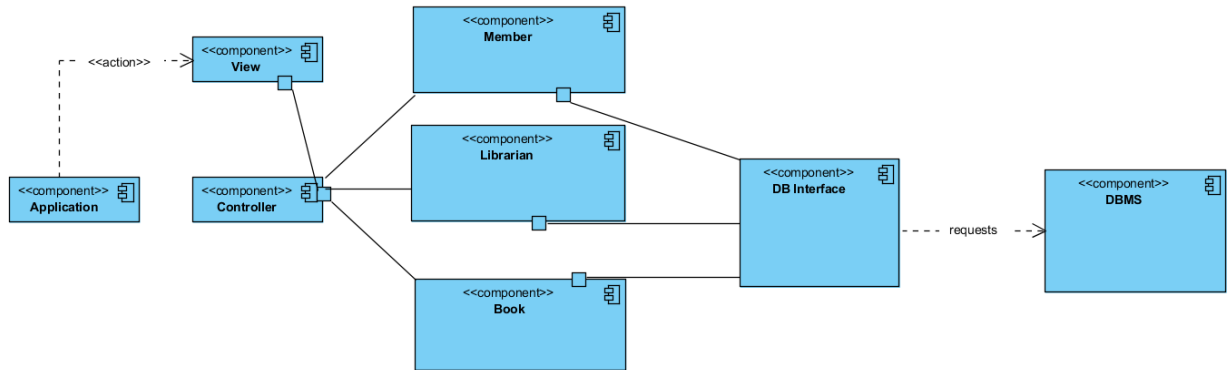
- **Deployment View:**
It has component and package diagrams.
Package diagram depicts the dependencies between the packages that make up a model.

Package Diagram

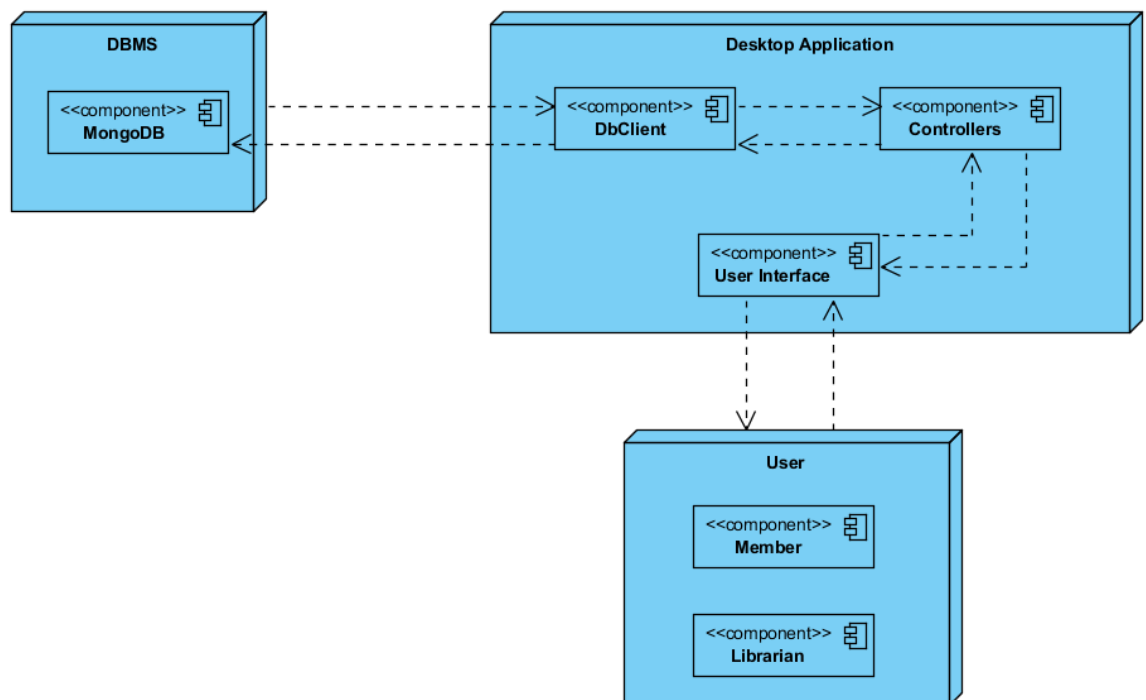


| | |
|---------------------------------|--------------------|
| Library book loan system | Group 4 |
| Architecture Notebook | Date: <31/03/2017> |

Component diagram shows how components are wired together to form larger components or software systems.



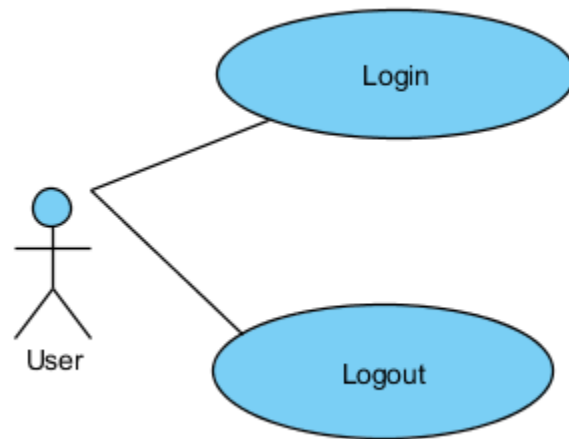
- **Physical View:**
It includes deployment diagram.
Deployment diagrams are used to describe the static deployment view of a system.



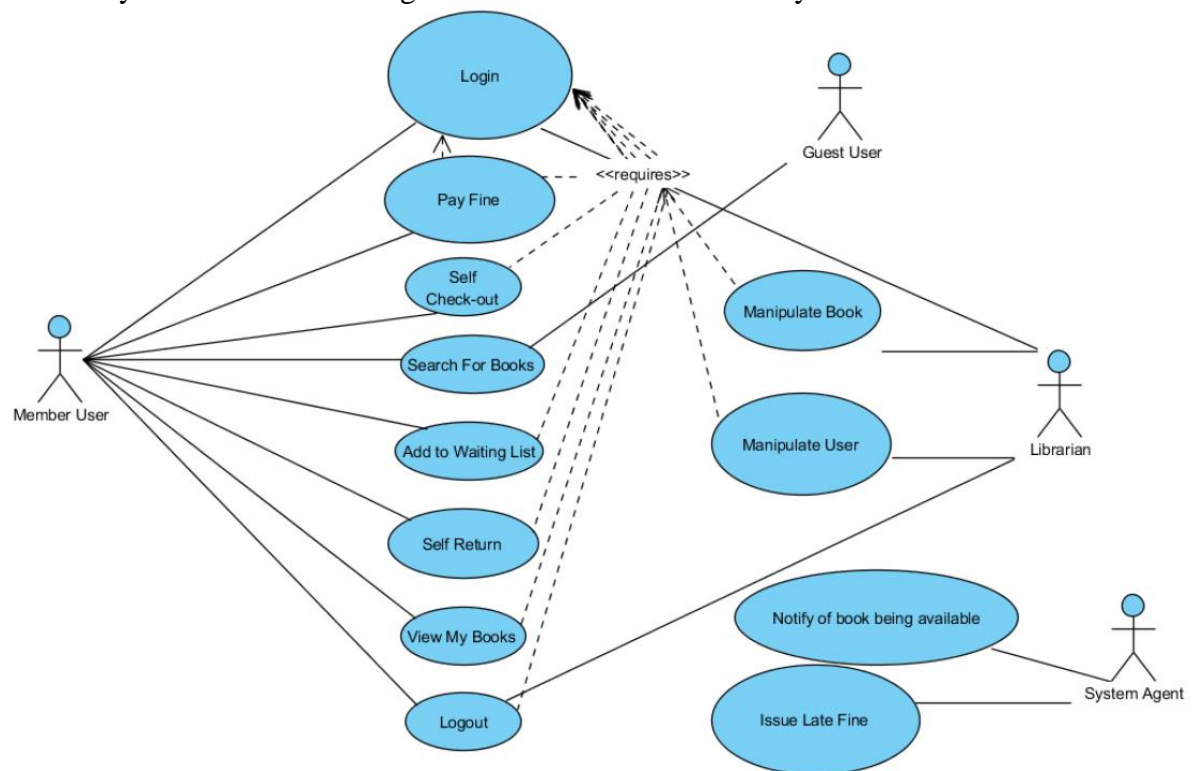
Deployment Diagram

| | |
|--------------------------|--------------------|
| Library book loan system | Group 4 |
| Architecture Notebook | Date: <31/03/2017> |

- **Logical View:** Library Book Loan System's Use Case Diagram For Demo



Library Book Loan System's Use Case Diagram With Whole Functionality:



- **Process View:**

This view contains activity diagrams and tabular information. You can access them System-Wide Requirements Documentation.