Students' Library

RATNODADHI

Nikhil Sharma(TCS15B032)

Mehar Chaturvedi(TCS15B019)

Vanka Sai Sumanth(TCS15B029)

15th November, 2017

Link to Ratnodadhi's Website : http://ratnodadhi.000webhostapp.com/Books/

Introduction- Need for Non-Academic Library

Our non-academic **library** is attached to serve two complementary purposes : to support students' curiosity in other non-academic fields, and to give students a means to escape from the monotonous subjects they study everyday. To facilitate all this, we bring the sea of jewels, ***RATNODADHI,*** in the form of a website open and accessible to all the students of the college. To support this idea and execution , we have developed:

- An application for both the non-academic librarian .
- Website for the students to keep a check on the books available and to have their favourite books come to them by creating a wish list.
- An API to programmatically test my database and the query execution

Requirements:

Academic Affairs Secretary

- Keep a record of all the books and their donors.
- Issue a book to a student.
- Receive a book on return.
- Get information about any book whether it is available or issued by someone.
- Store and update penalty for students who are defaulting and not returning books in time.
- Update the student via an email or some other means about the present status of book being issued, returned or in case of fine being added or reduced in his account.

- The system must be secured with a password so that any unauthorized person cannot issue or return a book.
- Know which books the students want so that they can be bought using the collected fine.

Student User

- Get information about a book and with whom it is presently available
- To be notified on any book issue or penalty that occurs using his roll number or on his name
- Request for the books he wishes to read so that if possible they can be purchased by the secretary.

Relational Schemas :

We are using 4 tables in our schema: issue (or borrower table), inventory, fine, wishlist

- Borrower : it contains the QR code of the book being issued along with the name, roll number of the student, date when the book is being issued and the name of the person issuing the book.

qr	roll	sname: Student Name	issuedate	Iname: Incharge Name

- Books : It contains the information about a book along with whether its available or not, name of the person who donated the book.

qr	roll	available	bname: Book Name	author	donor

- Fine: It contains the name of the student who has not returned the book on time along with date of return , the book name and the penalty he has to pay.

qr	roll	sname: Student Name	fdate: Date of Issue	amount

- wishlist : This contains the name and roll no of the student who wishes to get a new book along with author of the book and its name.

roll	sname: Student Name	bname: Book Name	author

Normal Form- BCNF :

All the relations used in this design exhibit a Boyce-Codd Normal Form. It can be clearly observed by looking at the primary keys of different tables. Only the primary key of each table is used to determine the other attributes of a table.

Table Statements

```
CREATE TABLE  `inventory` (
  `qr` VARCHAR(10) NOT NULL,
  `bname` VARCHAR(30) NOT NULL,
 `author` VARCHAR(30) NOT NULL,
   `available` VARCHAR(10) NOT NULL,
   `donor VARCHAR(30) NOT NULL,
  PRIMARY KEY (`qr`)
) ;
```

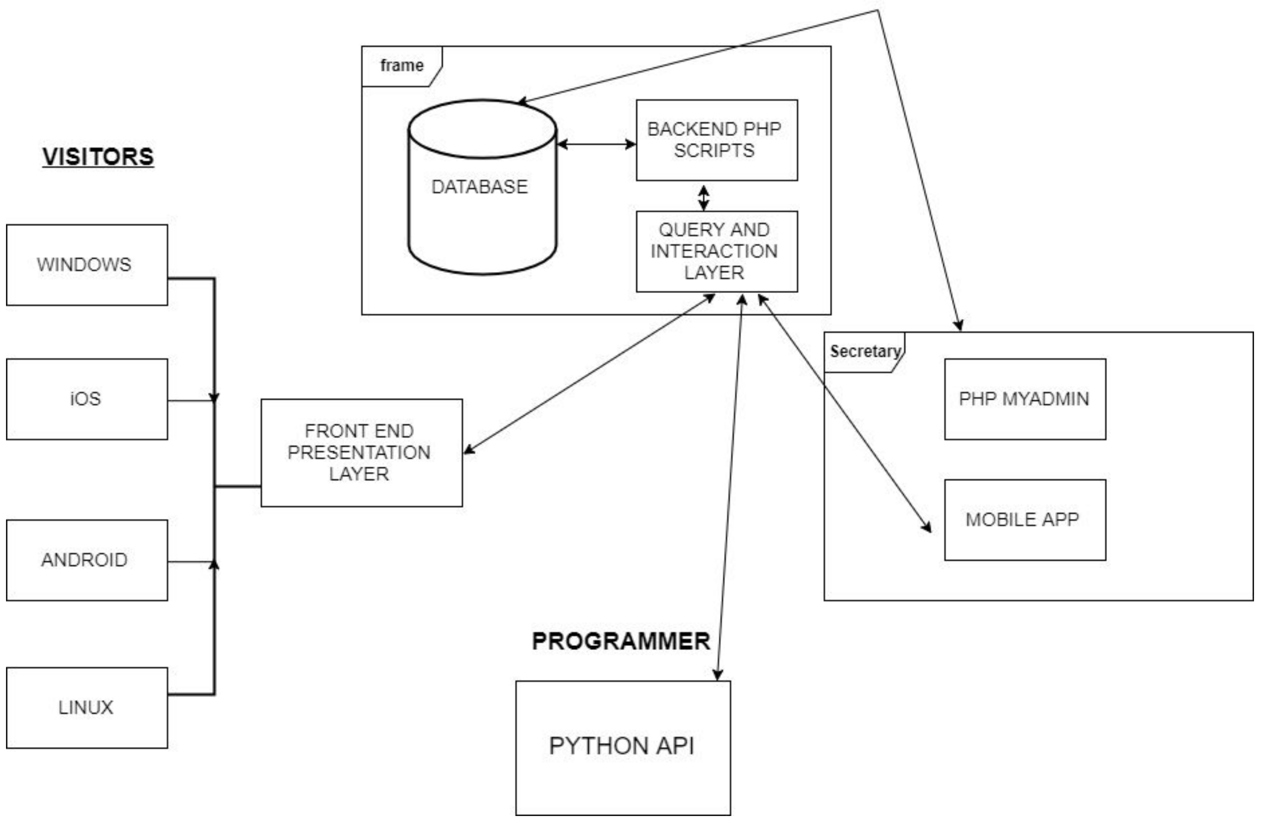
```
CREATE TABLE  `issue` (
  `qr` VARCHAR(14) NOT NULL,
  `rollno` VARCHAR(30) NOT NULL,
 `sname` VARCHAR(30) NOT NULL,
   `issuedate` DATE NOT NULL,
`iname`  VARCHAR(30) NOT NULL,
 PRIMARY KEY (`rollno`),
FOREIGN KEY   (`qr`), REFERENCES  inventory (`qr`)
            ON DELETE  SET NULL  ON UPDATE CASCADE) ;


CREATE TABLE  `fine` (
  `qr` VARCHAR(14) NOT NULL,
  `roll` VARCHAR(30) NOT NULL,
 `sname` VARCHAR(30) NOT NULL,
   `fdate` DATE NOT NULL,
`amount`  VARCHAR(30) NOT NULL,
FOREIGN KEY   (`rollno`), REFERENCES  student (`roll)
            ON DELETE  SET NULL  ON UPDATE CASCADE
FOREIGN KEY   (`qr``), REFERENCES  inventory (`qr`)
            ON DELETE  SET NULL  ON UPDATE CASCADE);



CREATE TABLE  `wishlist` (
  `roll` VARCHAR(30) NOT NULL,
 `sname` VARCHAR(30) NOT NULL,
   `bname` VARCHAR NOT NULL,
`author` VARCHAR(30) NOT NULL,
FOREIGN KEY   (`roll`), REFERENCES  student (`roll`)
          ON DELETE  SET NULL  ON UPDATE CASCADE
);
```

The Following is the architectural diagram of the Project:-

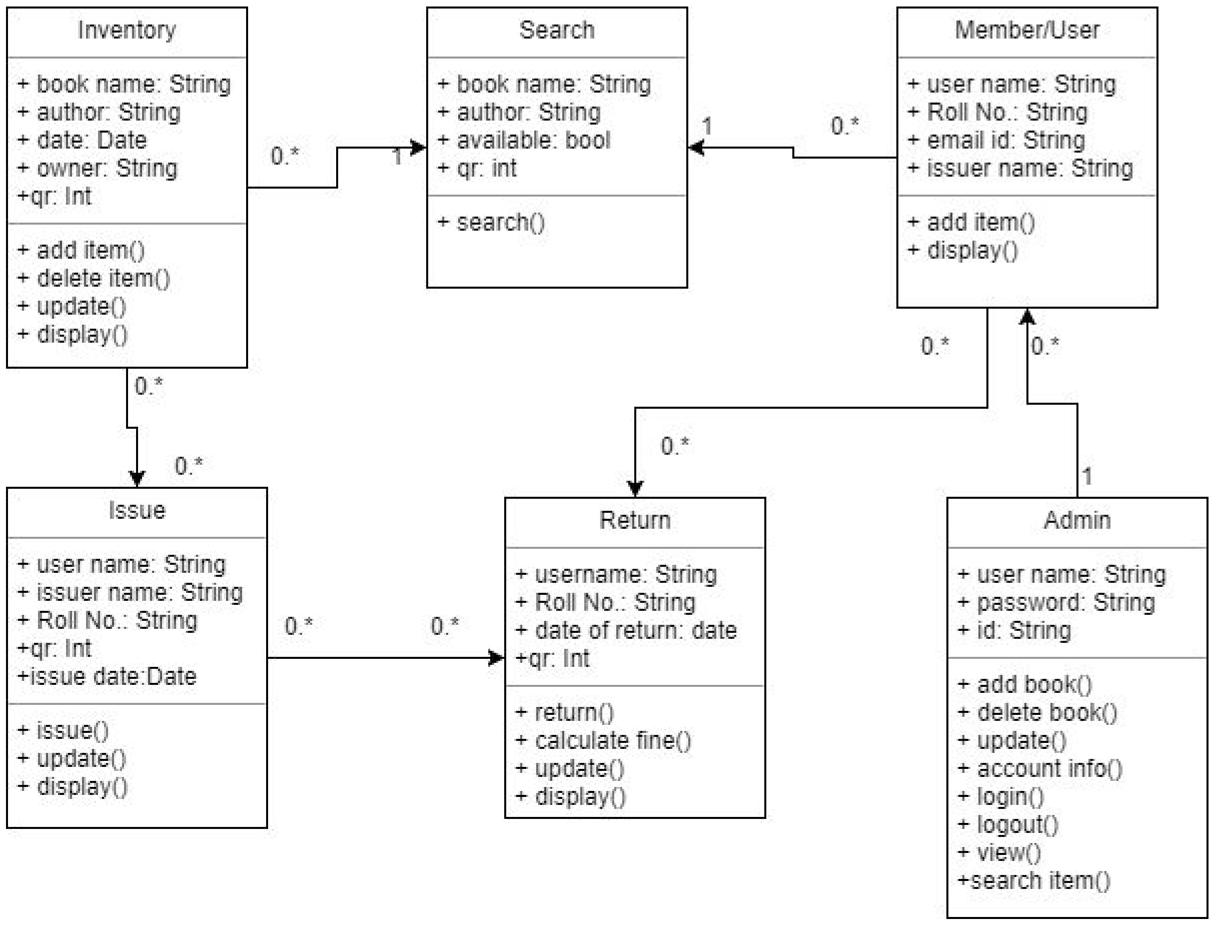
Client-server architecture (client/server) is a network architecture in which each computer or process on the network is either a *client* or a *server.* It allows multi-user updating through a GUI front end to a shared database. Remote Procedure Calls (RPCs) or standard query language (SQL) statements are typically used to communicate between the client and server.

1) **Database** module is connected to Backend php scripts which in turn can be accessed through interaction layer.
2) **Clients/Users** access the website front end from various different platforms like Windows, Linux, Mac etc.
3) The **Admin** that is the secretary has the access to both the database (through phpMyAdmin) and the Android App. And he manages the user interface of the app.
4) **Programmer/ Developer** will be interacting and testing by the python API.
5) All the users, Admin and developers are internally connected to the **interaction layer** where any change in the whole system will get updated.
6) The environment is typically **heterogeneous** and multi-vendor. The hardware platform and operating systems of client and server are not the same.

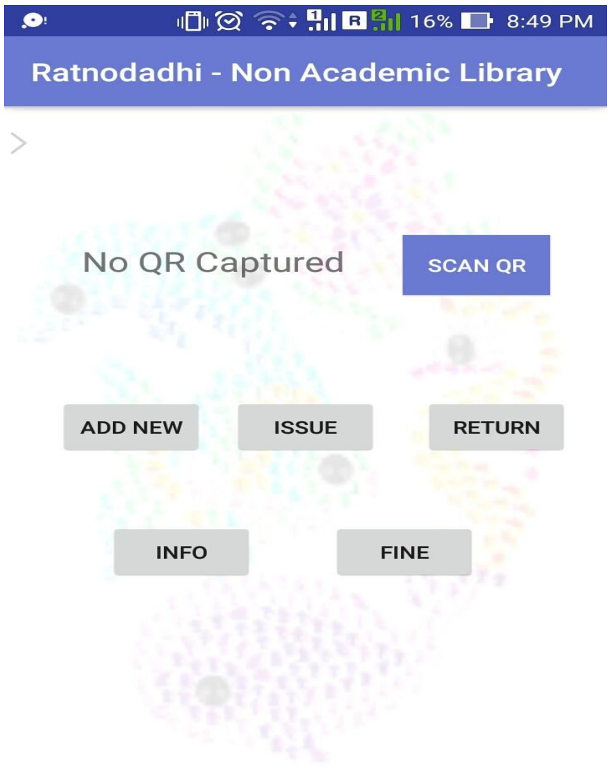
ARCHITECTURE OVERVIEW (Client-Server)



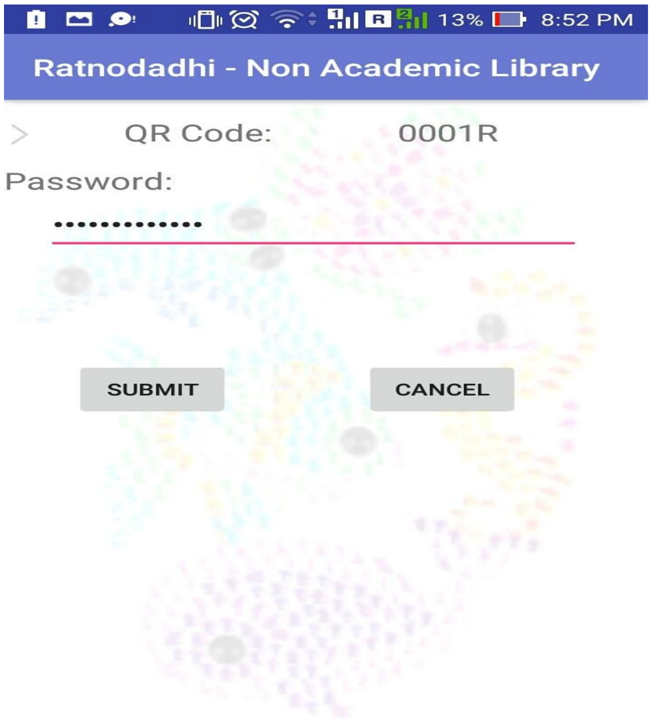
CLASS DIAGRAM



APPLICATION STRUCTURE



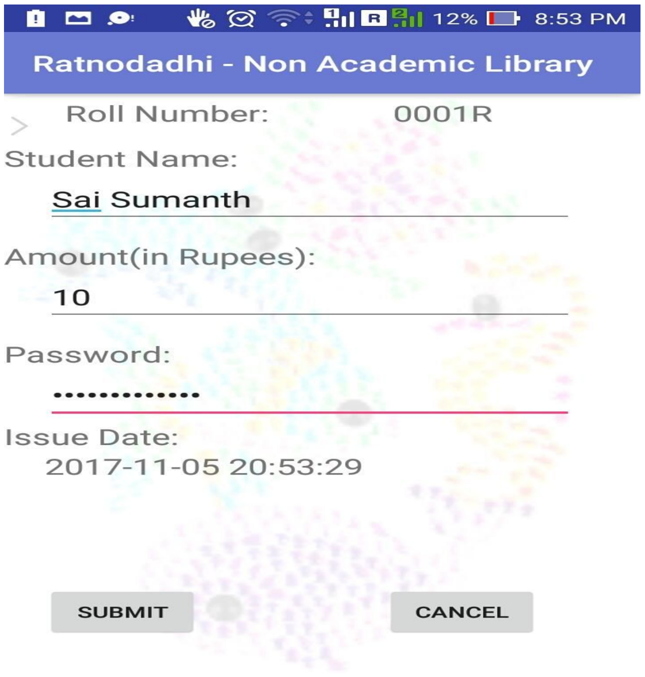
Main application screen


Ratnodadhi - Non Academic Library

QR Code: 0001R

Password:

<u>This screen occurs for verification of the authorised person on return.</u>



Ratnodadhi - Non Academic Library

Roll Number: 0001R

Student Name:

Sai Sumanth

Amount(in Rupees):

10

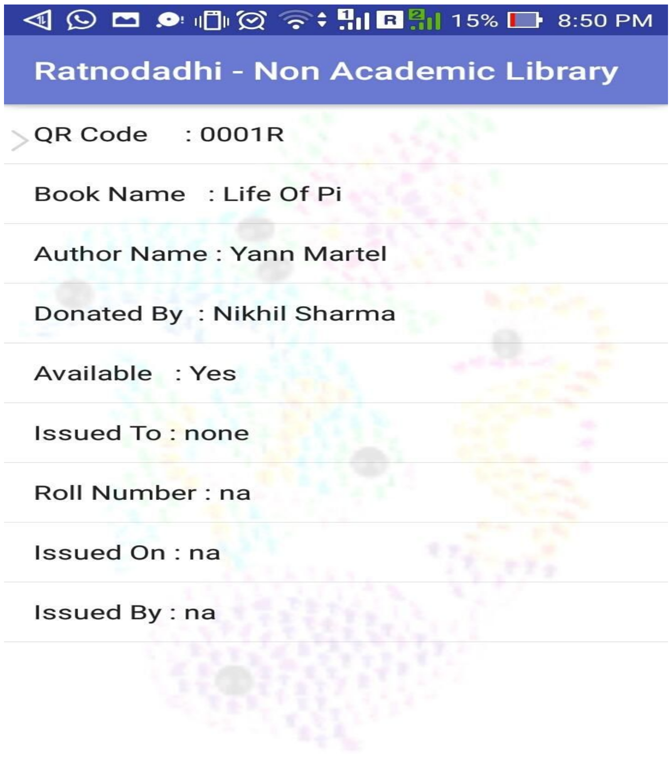
Password:

••••••••••••••

Issue Date:

2017-11-05 20:53:29

This screen occurs when we need to add fine corresponding to a particular student.



Ratnodadhi - Non Academic Library

QR Code : 0001R

Book Name : Life Of Pi

Author Name : Yann Martel

Donated By : Nikhil Sharma

Available : Yes

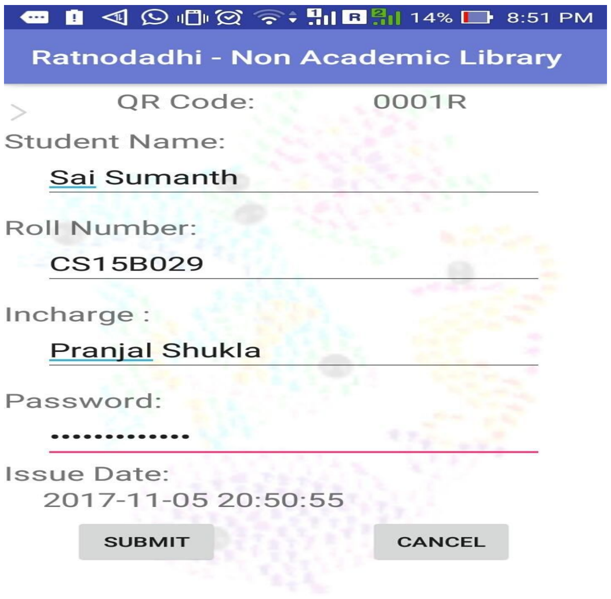
Issued To : none

Roll Number : na

Issued On : na

Issued By : na

<u>The screen occurs when we want to see the information of a book.</u>



Ratnodadhi - Non Academic Library

QR Code: 0001R

Student Name:
Sai Sumanth

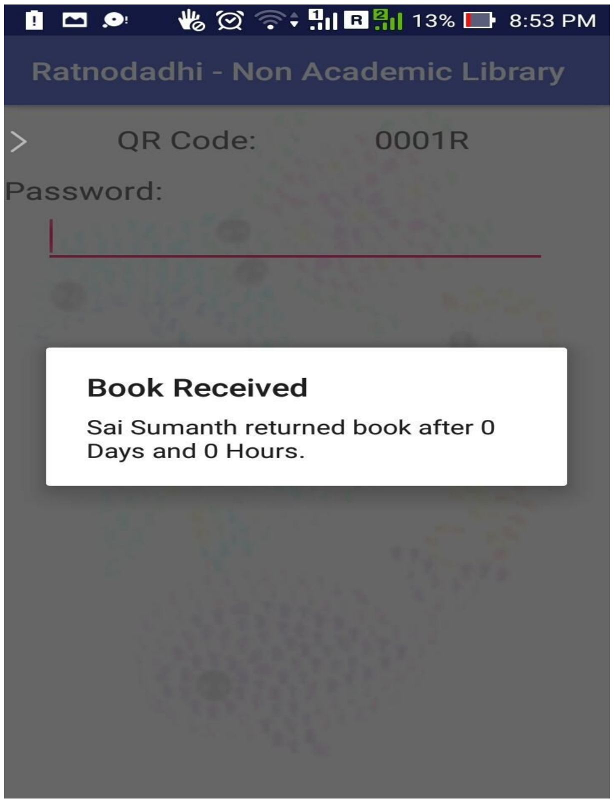
Roll Number:
CS15B029

Incharge :
Pranjal Shukla

Password:
•••••••••••••

Issue Date:
2017-11-05 20:50:55

<u>The above screen occurs when a book is being issued.</u>



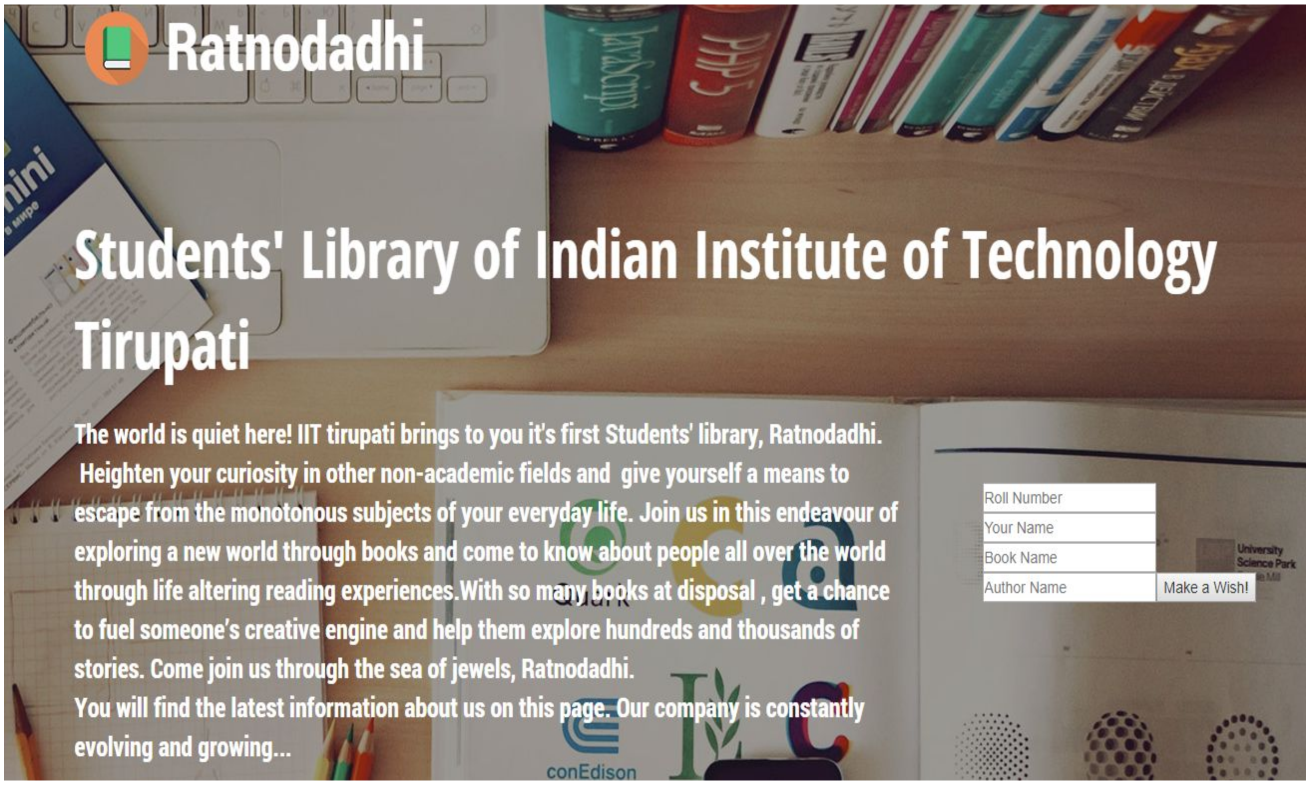
Book Received

Sai Sumanth returned book after 0 Days and 0 Hours.

The screen occurs when a book is returned by a student along with the time he delayed in returning the book.

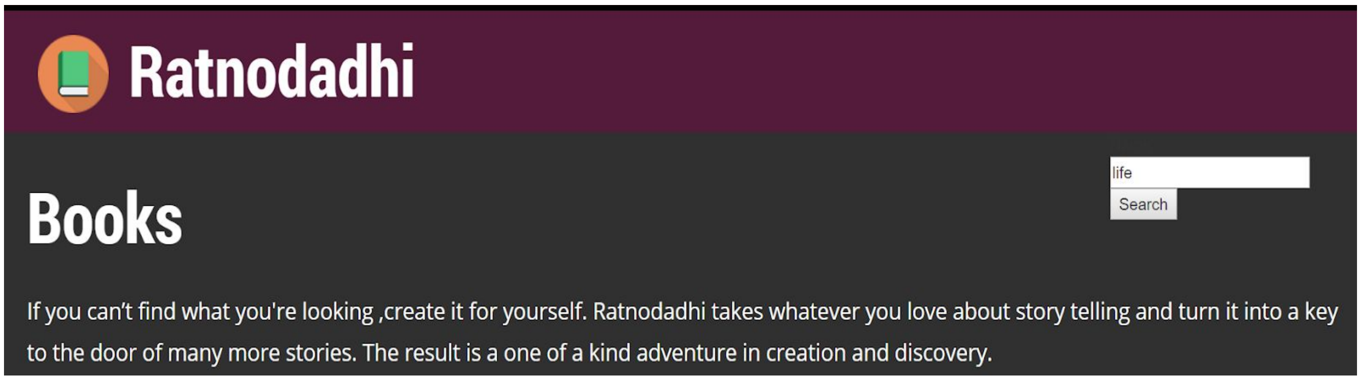
WEBSITE STRUCTURE

Given below is the basic display outline of the website with the basic utility function shown .This includes searching for a book .


Ratnodadhi

Students' Library of Indian Institute of Technology Tirupati

The world is quiet here! IIT tirupati brings to you it's first Students' library, Ratnodadhi. Heighten your curiosity in other non-academic fields and give yourself a means to escape from the monotonous subjects of your everyday life. Join us in this endeavour of exploring a new world through books and come to know about people all over the world through life altering reading experiences.With so many books at disposal , get a chance to fuel someone's creative engine and help them explore hundreds and thousands of stories. Come join us through the sea of jewels, Ratnodadhi.
You will find the latest information about us on this page. Our company is constantly evolving and growing...



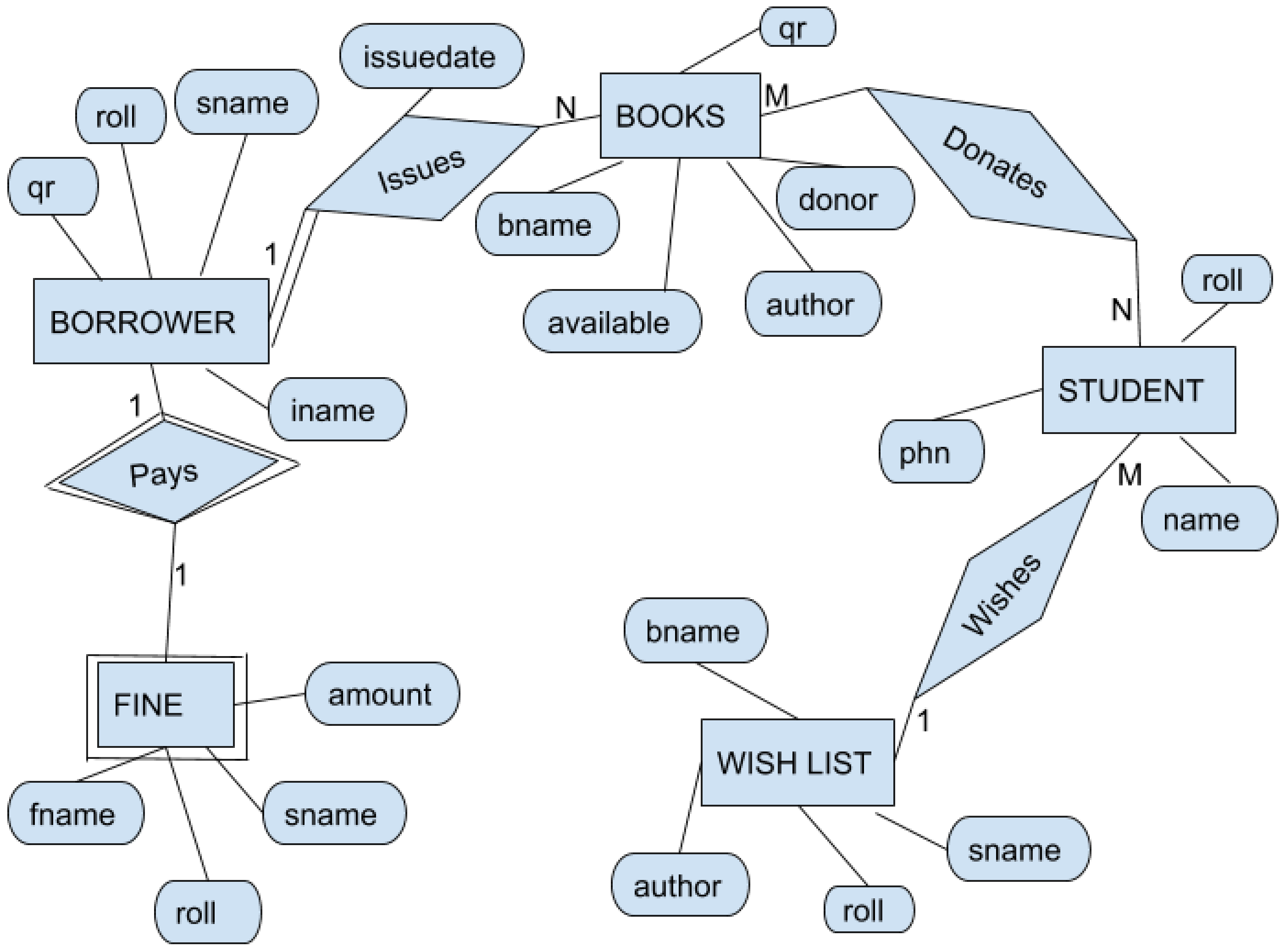
Books

If you can't find what you're looking ,create it for yourself. Ratnodadhi takes whatever you love about story telling and turn it into a key to the door of many more stories. The result is a one of a kind adventure in creation and discovery.

Given below is the search result of "life"

[{"qr":"0001R","bname":"Life Of Pi","author":"Yann Martel","donor":"Nikhil Sharma","available":"Yes","roll":null,"issuedate":null,"iname":null,"sname":null}]

ER DIAGRAM


qr
issuedate
roll
sname
qr
N
BOOKS
M
Issues
Donates
bname
donor
1
available
author
roll
BORROWER
N
STUDENT
1
iname
phn
Pays
M
name
1
Wishes
bname
FINE
amount
1
WISH LIST
fname
sname
sname
roll
author
roll

OVERVIEW OF THE FLOW OF PROJECT

1. The database was designed by brainstorming different possible alternatives for ER diagram to help understand the flow of different schemas used.
2. This was followed by generating SQL queries by creating tables.
3. In parallel, a mobile application was developed using Android Studio and the website was designed.
4. We used 000Webhost as a server for integrating all the components of the project such as the mobile app and the website.
5. The communication between all these components was done using php script.
6. The first design that was tested contained only the Book Issue and Book Info activity. Later on, it was extended to Book return and Fine activity.
7. To notify the students of the updated status, php mailer was used. Student's id-card could be scanned using the same QR scanner.
8. To maintain the consistency of the database where more than one queries are being executed, we make use of transactions.

FUTURE DIRECTIONS

1. Present the details of the searched book in a format that is more easily readable and capable of displaying more number of books.
2. Make the database presentable for the secretary rather than making him visit the phpMyAdmin of the website each time he wishes to see the details.
3. Store the fine along with the borrower information in the issue table.
4. With this we present the suggested architecture for the next version of the system with a slight change in current architecture.

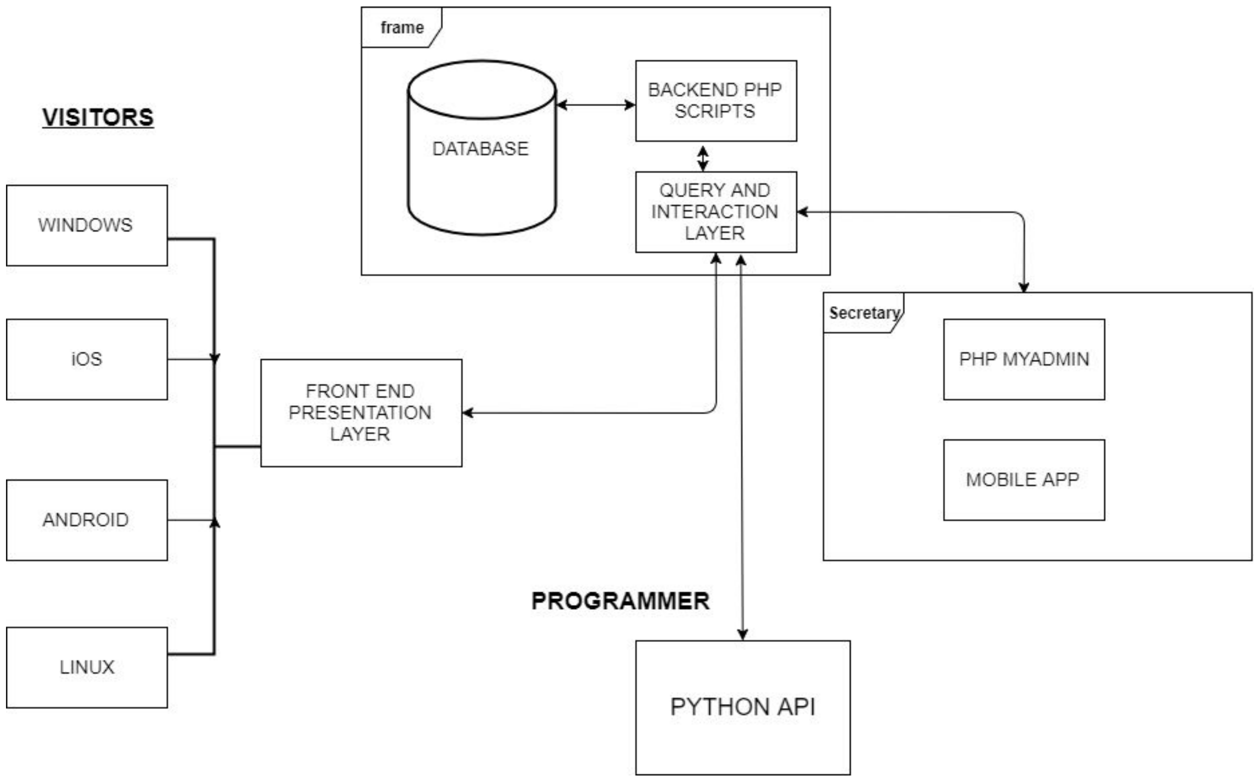


Fig. Proposed Future Architecture