1.0 Introduction

To demonstrate how our system works, we need to specify customers' requirements, and then analysis it. In this part, we will make clear what our requirements are and analysis those requirement in different scenarios (in 2.0), and develop class diagram(in 4.0), E-R diagram(in 3.0), state diagram(in 5.0). In the end, we do a brief analysis about system's constrains, and decided necessary quality control measures.

Goals and objectives

Our system's goal is providing a platform to track books. The book has 2 statues: checked out or in stock. If a book is in "checked out" statue, the book shall have been checked out by a teacher or staff to a student. If a book is in "in stock" statue, the book can be checked out by a teacher or staff to a student who can borrow books. (A student can't borrow books if he/she has already checked out book in numbers of classes he/she is registered for). Our system will store data for where the book is (in student or in library).

Aside of tracking books, our system provide security to access the system. All user need to log in by using their ID and userId before they use services of our system . There are 3 types of user in our system: student, staff and teachers. And for different user, they have different access level and according to user's access level, they enjoy different service.

Statement of scope

Here is a detailed description for different user's service.

For student, their access level is 1 and they can use 2 types services:create/update/view parents' information for his/her account and view other account profile: wether checked out books, balance due and etc. Because parents and their students share same account in our system, parents info is required to be created by student who first access to our system, updated by student if there are any changes about their parents' personal info and viewed if a student log in their account and want to see those info. Parents' info should include parents' names and phone number. For a student account, there are 2 types of statues: finish and checking_out. If a student account is in "finish" statues, he/she doesn't borrow books. If a student account is in " checking_out" statue, he/she has been borrowed at least 1 books. In student account, they can view their balance due which is pending by staff if they keep a book after the due date of that book.

For teacher, their access level is 2 and they can use 2 types of services: check out books for his/ her class and create/update/view his/her account information. A teacher can only checked out books for student in his classes. To check out book, they will change a checked out book's statues to "in stock" in our system and may change the statue in student account to "finish". Teacher shall maintain their personal info: name, phone number and can create, update and view those info.

For staff, their access level is 3 and they can use 13 types of services:check out any books,refill any books if needed, update any user/books/classes information, delete any user/ books/classes information, add any user/books/classes information,list all students with books checked out, list all authorized users and their access level, list all books in the system with search capabilities, list of all books checked out by class or book name/number, list of students and parents info with amount due, create/update/view his/her account information, Post bill to a student account, maintain our system in error case(book not enough, provide new userId for student, etc).

The interface is provided to users is more desirable in web based form. For arrange data, rational database is more desirable. The system must respond to all requests within 5 seconds.

Type of requirements are non-functional requirement(NFR), functional requirement(FR) and domain requirement(DR). Functional requirements are services provided by system, domain requirements are required attributes and non-functional requirements are techs to realize our system. Requirement 67,68 are assumptions made by our group.

REQ. NO	DESCRIPTION	Type of requirement	RANK
1	Data Entry : HTML Form	NFR	Desirable
2	Database: SQL server	NFR	Desirable
3	Navigation: HTML Links, control	NFR	Desirable
4	JavaScript	NFR	Desirable
5	Flat file	NFR	Optional
6	Respond within 5 seconds	NFR	Desirable
7	student info:Student ID	DR	Essential
8	teacher info:Teacher ID	DR	Essential
9	staff info:Staff ID	DR	Essential
10	student/staff/teacher info:User ID	DR	Essential
11	Book_Info: name	DR	Essential
12	Book_Info: due date	DR	Essential
13	Book_Info: borrow date	DR	Essential
14	Book_info: ISBN number	DR	Essential
15	student info:type	DR	Essential
16	staff info:type	DR	Essential
17	teacher info:type	DR	Essential
18	student info:access level	DR	Essential
19	staff info:access level	DR	Essential
20	teacher info:access level	DR	Essential
21	staff info: name	DR	Essential
22	teacher info: name	DR	Essential
23	staff info: phone number	DR	Essential
24	teacher info: phone number	DR	Essential
25	class info: name	DR	Essential

26	class info: ID	DR	Essential
27	book info: ISBN number	DR	Essential
28	book info: school generated number	DR	Essential
29	book info: name	DR	Essential
30	student info: parents name	DR	Essential
31	student info: parents number	DR	Essential
32	date info: year	DR	Essential
33	date info: month	DR	Essential
34	data info: day	DR	Essential
35	Teacher: create account information	FR	Essential
36	Teacher: update account information	FR	Essential
37	Teacher: view account information	FR	Essential
38	Teacher: check out books for his/ her class	FR	Essential
39	Student: create parent information	FR	Essential
40	Student: update parent information	FR	Essential
41	Student: view parent information	FR	Essential
42	Student: view balance due	FR	Essential
43	Student: view account state	FR	Essential
44	Staff: update user information	FR	Essential
45	Staff: update book information	FR	Essential
46	Staff: update class information	FR	Essential
47	Staff: delete user information	FR	Essential
48	Staff: delete book information	FR	Essential
49	Staff: delete class information	FR	Essential
50	Staff: add user information	FR	Essential
51	Staff: add book information	FR	Essential
52	Staff: add class information	FR	Essential
53	Staff: list all students with the books checked out	FR	Essential
54	Staff: list all book checked out by class	FR	Essential
55	Staff: list all book checked out by book name	FR	Essential
56	Staff: list all book checked out by book number	FR	Essential

57	Staff:Maintain out system in error case:book not enough	FR	Desirable
58	Staff:Maintain out system in error case:provide new userId for student	FR	Desirable
58	Staff: list books with search capabilities	FR	Essential
59	Staff: list students and parents info with amount due	FR	Essential
60	Staff: list authorized user and their access level	FR	Essential
61	Staff: list all book checked out by class or name/No	FR	Essential
62	Staff: refill any books if needed	FR	Essential
63	Staff: create account information	FR	Essential
64	Staff: update account information	FR	Essential
65	Staff: view account information	FR	Essential
66	Staff:Post bill to a student account	FR	Essential
67	For each check out operation, user shall check out 1 book each time.	FR	Essential
68	All students are registered.	FR	Essential

Software context

The software is consist by 2 part: web-based interface, and java application. By using Tomcat(apache tomcat 8.0 and higher version), we can use the web site to provide service to users and java application will handle retrieve information in flat file and forward those data to website.

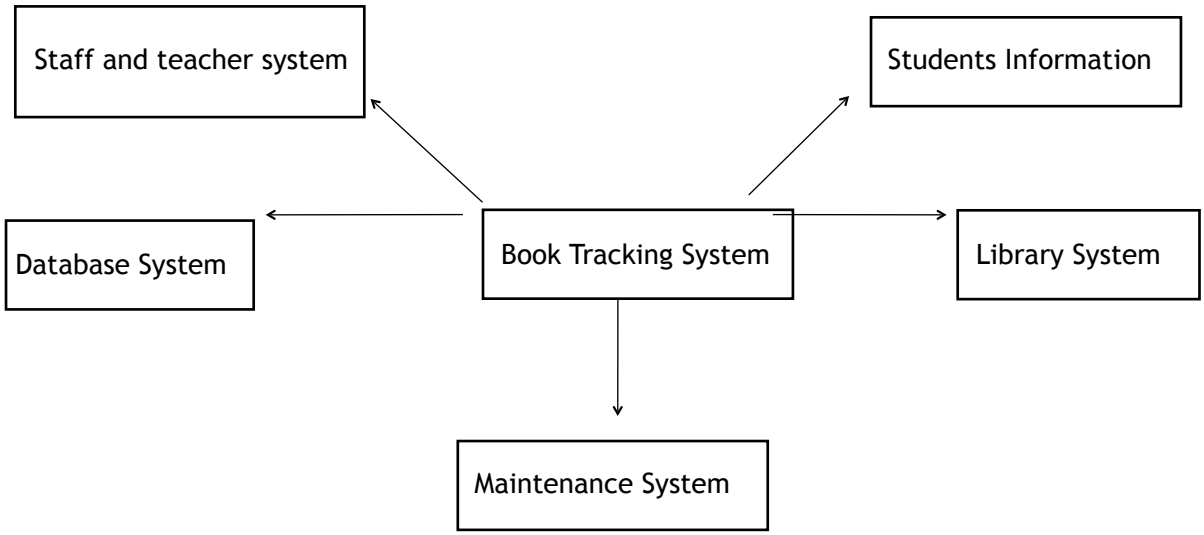
Because we use flat file to manage our data, it will be more tricks to organize our flat file. To increase the performance of our application: responding time, we may choose a better way to manage our flat file: try to retrieve as less file as we can, because we known that with more pressure in local IO, it will takes more time to get a response under the same hardware condition. In future test, we may change our organization of flat files to achieve better performance.

Major constraints

For this system, the major constrains comes form non-functional requirements. Because it is hiding behind the service provide by the system and we can't say wether we fulfill those until we test those service. For example, the response time.

Another problem is the specification of requirements, although we have clearly known the major services provided by our system, we still need to negotiation with our customer to prove our assumption is reasonable for customers.

The context of an book tracking system



2.0 Usage scenario

From requirement document, our system can let user query and update book information , update balance due, and change or query account information.

2.1 User profiles

Staff

A staff has the highest access level in our system (level 3), which means a staff can have whole access to any data in our system's database and we assumed that a staff means an administrator. And he/she does following things:

Number	Activity
1	check out any books
2	refill any books if needed
3	update any user/books/classes information
4	delete any user/books/classes information
5	add any user/books/classes information
6	list all students with books checked out
7	list all authorized users and their access level
8	list all books in the system with search capabilities.
9	list of all books checked out by class or book name/number.
10	list of students and parents info with amount due
11	create/update/view his/her account information

12	Post bill to a student account
13	maintain our system in error case(book not enough, provide new userId for student, etc)

Teacher

Number	Activity
1	check out books for his/ her class
2	create/update/view his/her account information

A teacher has level 2 access level in our system, which means a teacher can access part of data in our system's database. And he/she does following things:

Student

A student has level 1 access level in our system, which means a student can only access to his/her account. And he/she does following things:

Number	Activity
1	create/update/view parents' information for his/her account
2	view other account profile: wether checked out books, balance due

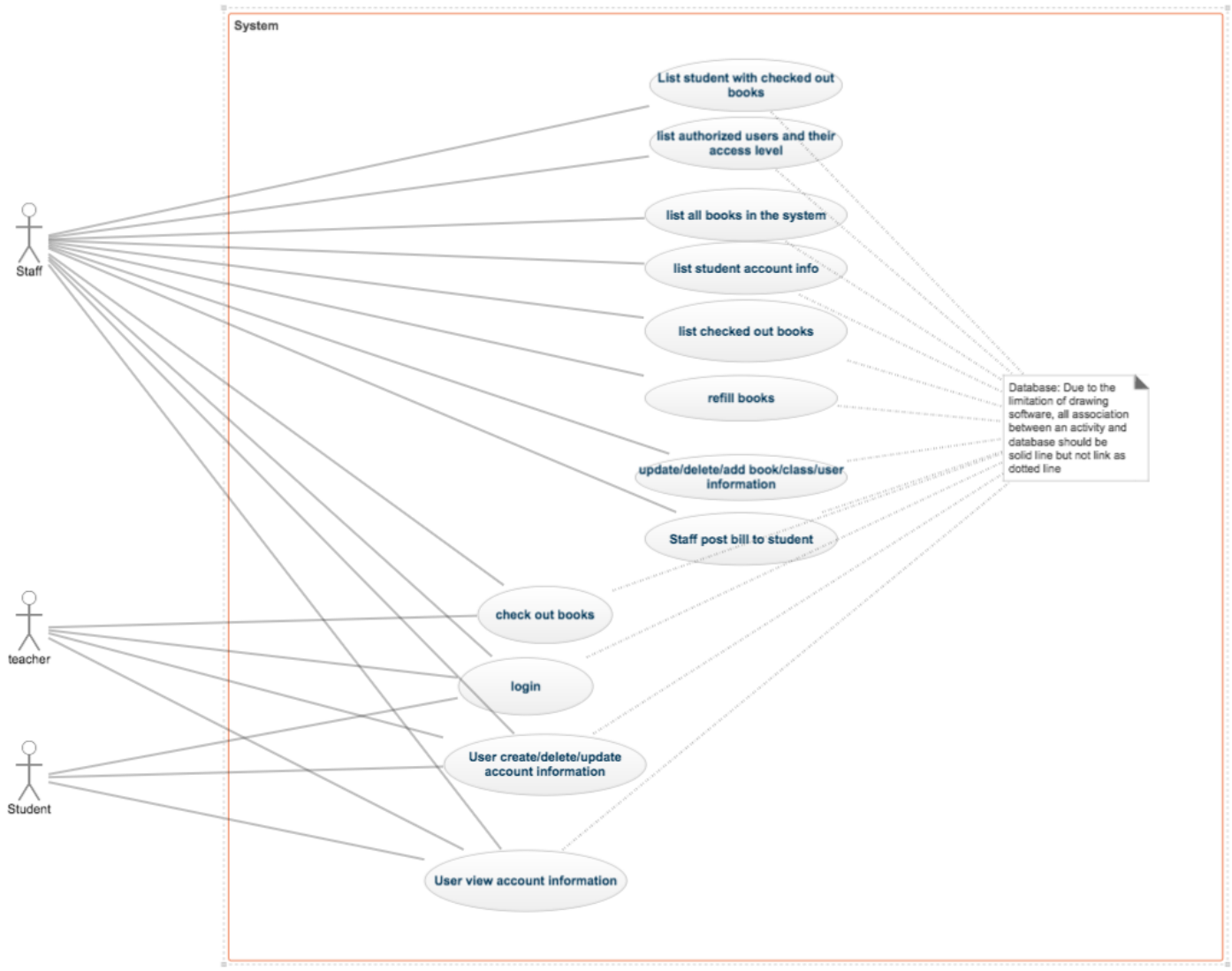
2.2 Use-case

2.2.1 Description

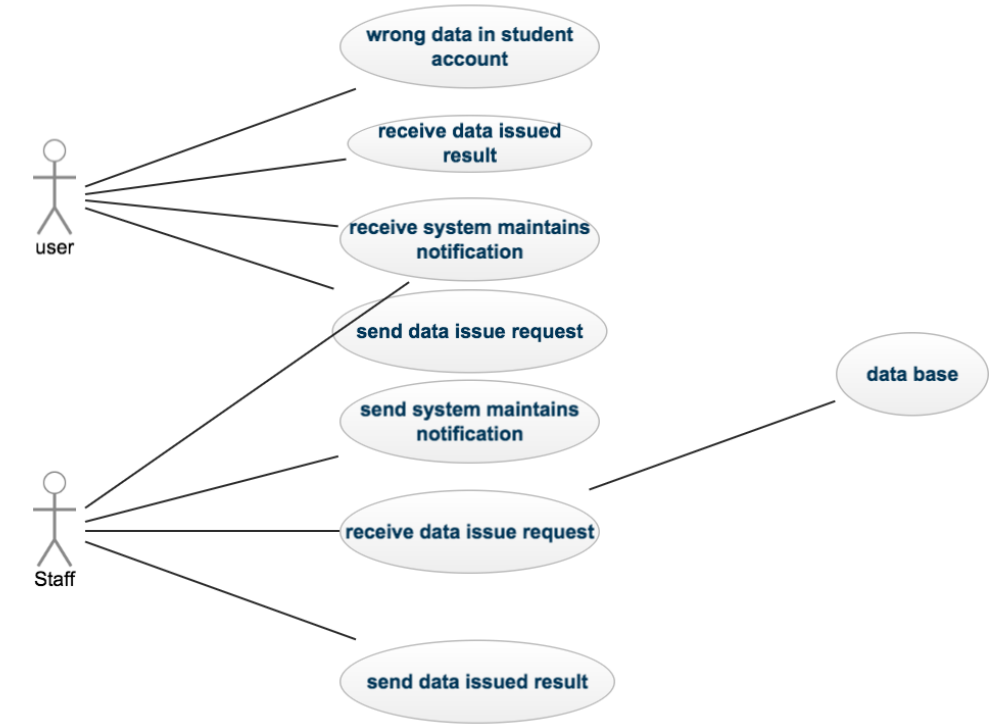
Here are different Use case for our system.

Name	Description
Overall Use case	Include all activities listed in 2.1.
Use case diagram for staff maintain system	When error occur(student find wrong data in their account:staff/teacher checked out wrong books for student, system stop service for maintain), staff shall correct data or send notice to all user.
Use case diagram for post bill to student account	If a student can't keep the book anymore(exceed due date), staff shall post a bill buy using due date, number of books.

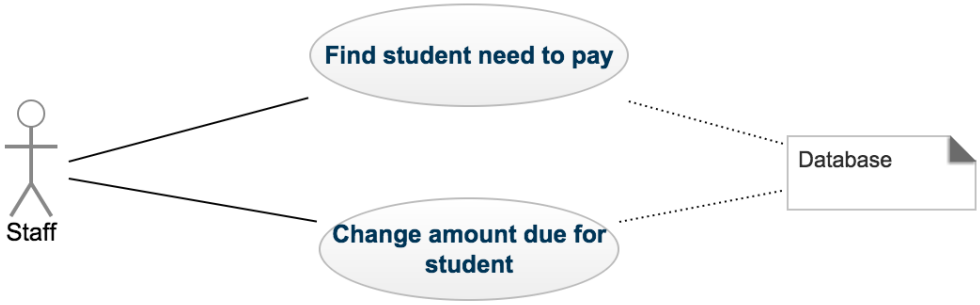
2.2.2 Use case diagram



Overall Use case



Use case diagram for staff maintain system



Use case diagram for post bill to student account

2.3 Special usage considerations

This is a web-based application, a standards compliant browser is essential. For example, Google chrome. To arrange and store data, we can choose suitable approach like flat file, which is cheaper but may not convenient to arrange large amount of data.

2.4 Activity Diagrams: Swim lane Diagrams

2.4.1 Description

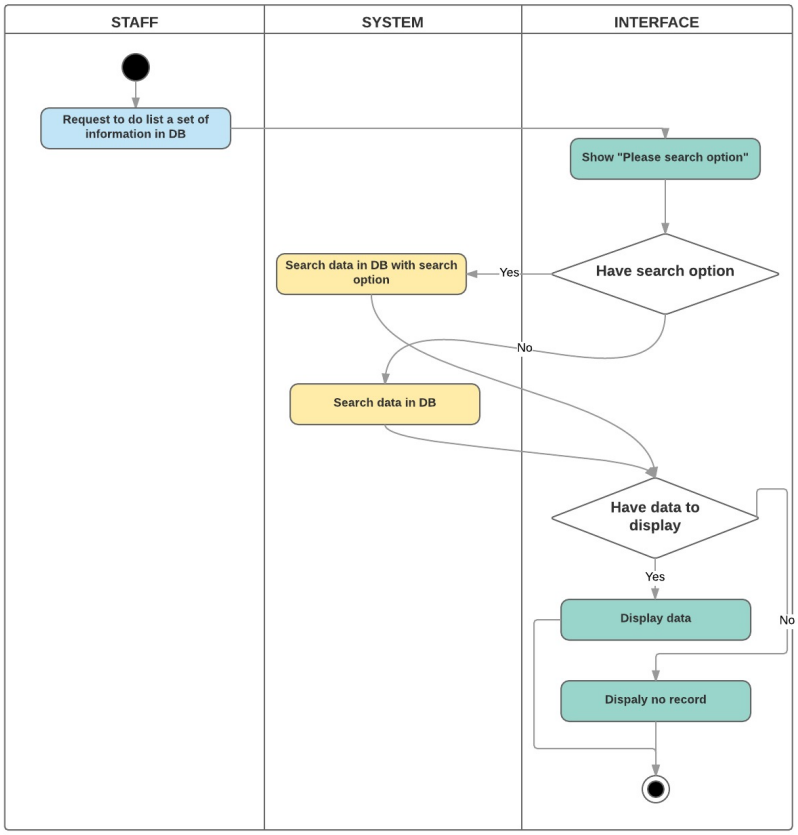
As we known, swim lane diagrams are used to explain activities in use case diagram. So we only list activities which need to tell more details. Following are the list of activity diagrams based on the Swim lane model.

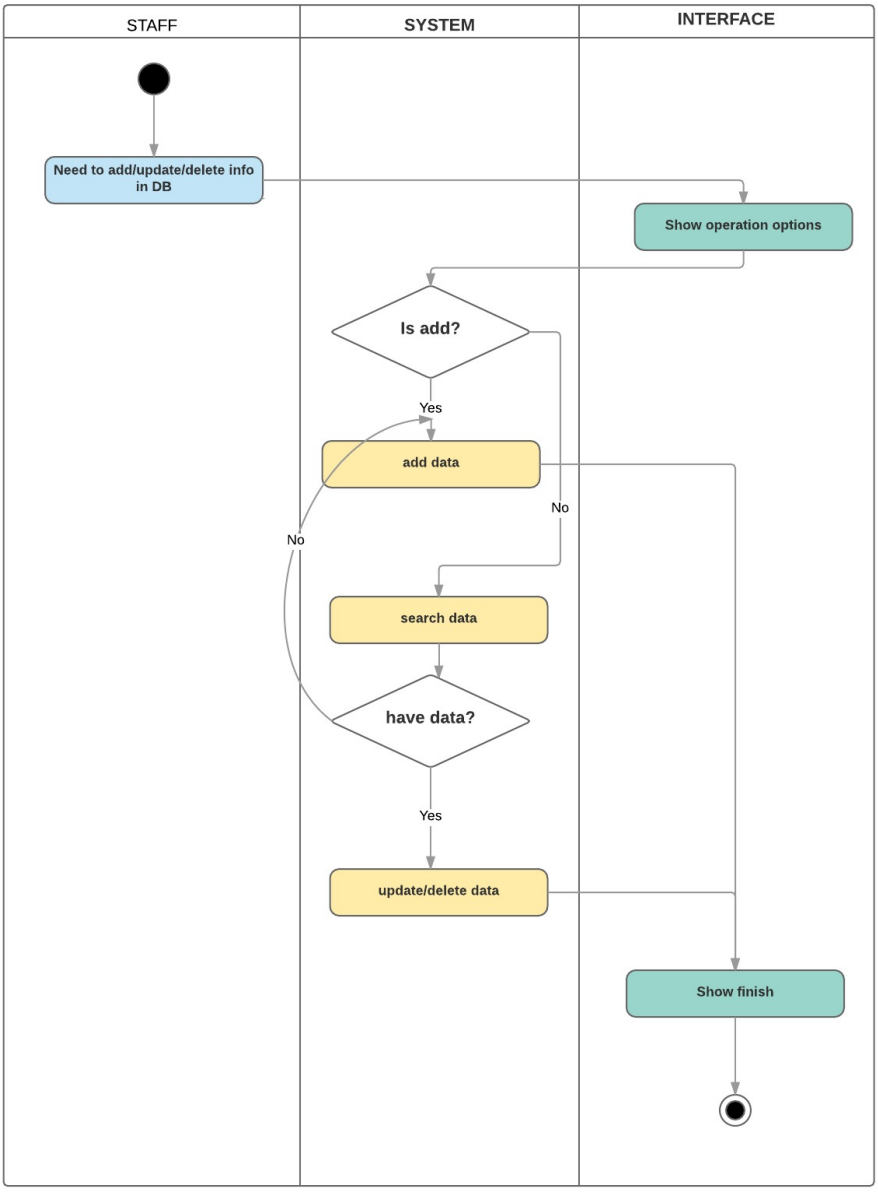
name	description
List information in DB	After staff do query in DB, we can list all required information
Refill book	If a student want to borrow a book with no stock, he need to send a request to refill this book. Staff will take this request and refill this book and send the result of students' request to student.
Add/delete/update information in DB by staff	Staff can add/delete/update a specific information in data base.
Add/delete/update personal information in DB by user	User can add/delete/update his/her personal info in DB.
User login	First of all, user input user Id and ID in our system. Then if those info is correct, the user can login. If those info is not correct, the system will send a warn. If user want to query their user Id, they will send a request to staff, after staff query that user id in DB, they can login.
teacher/staff checked out books	Use book info,date and student info staff/user can check out books for student. If student can't check out book, the request will be decline, if student can check out books, the request will process.
Staff post bill	If a checked out book exceed due, staff shall change amount due in student account based on number of exceed due date books and due date.
Maintain system	For error case system maintains, staff should send maintains notification to all users. For error case wrong data, student thinks his/her account shows wrong data in balance due or check out state, he/she will send a request to staff. Once staff receive that request, he/she will query that information in DB and record in library, if it is correct, send refuse to student to show his/her request has been refuse. If it is incorrect, send result with approval to student.

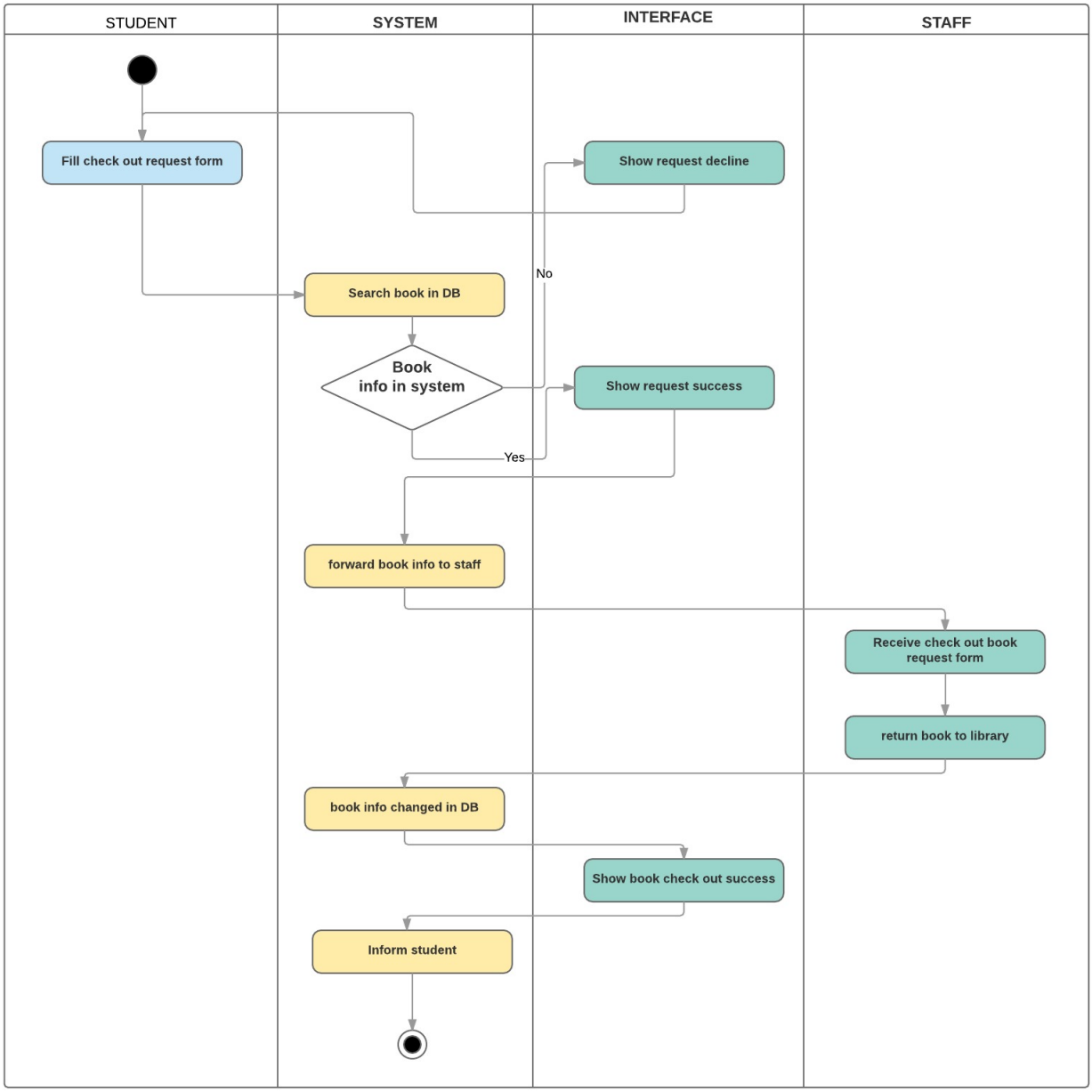
2.4.2 Swim lane diagram

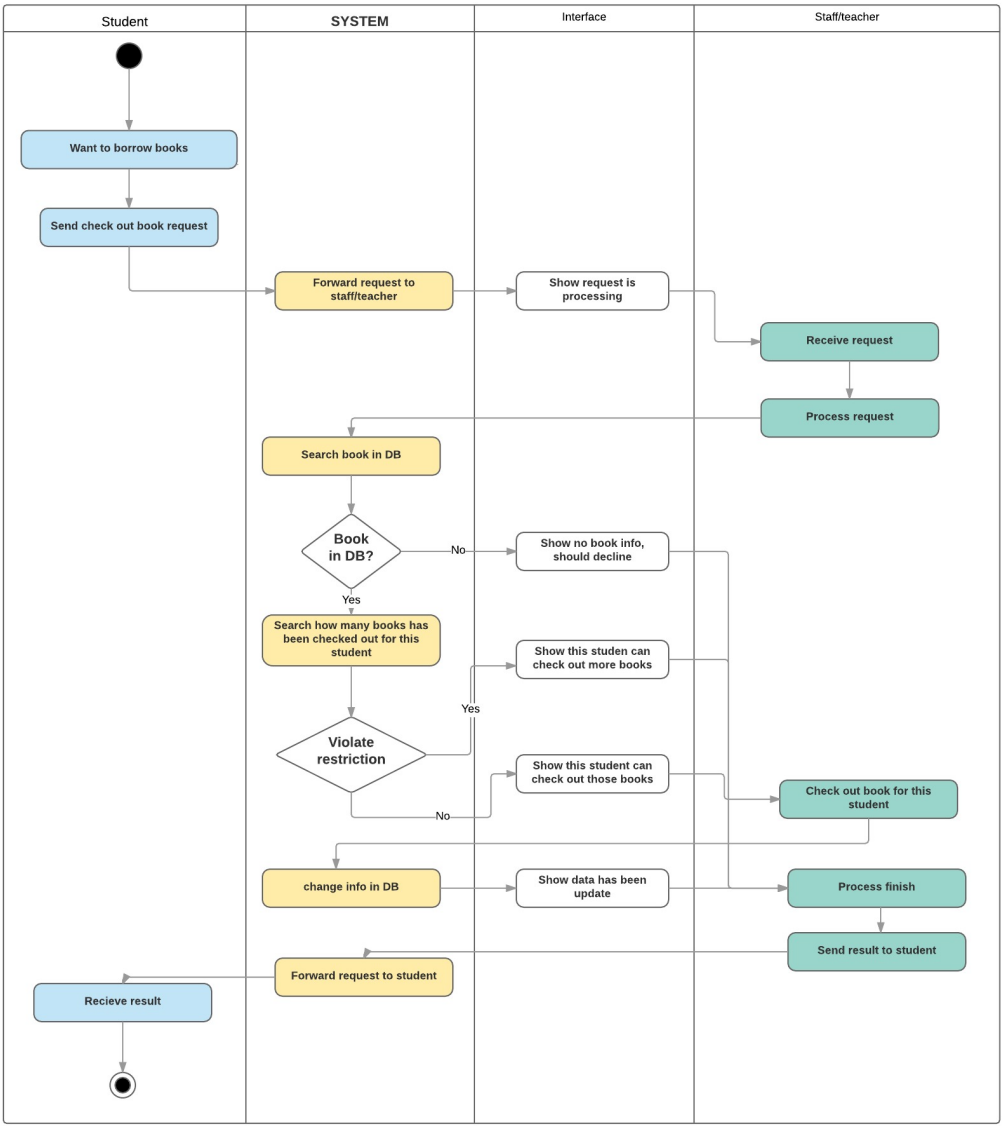
List information in DB

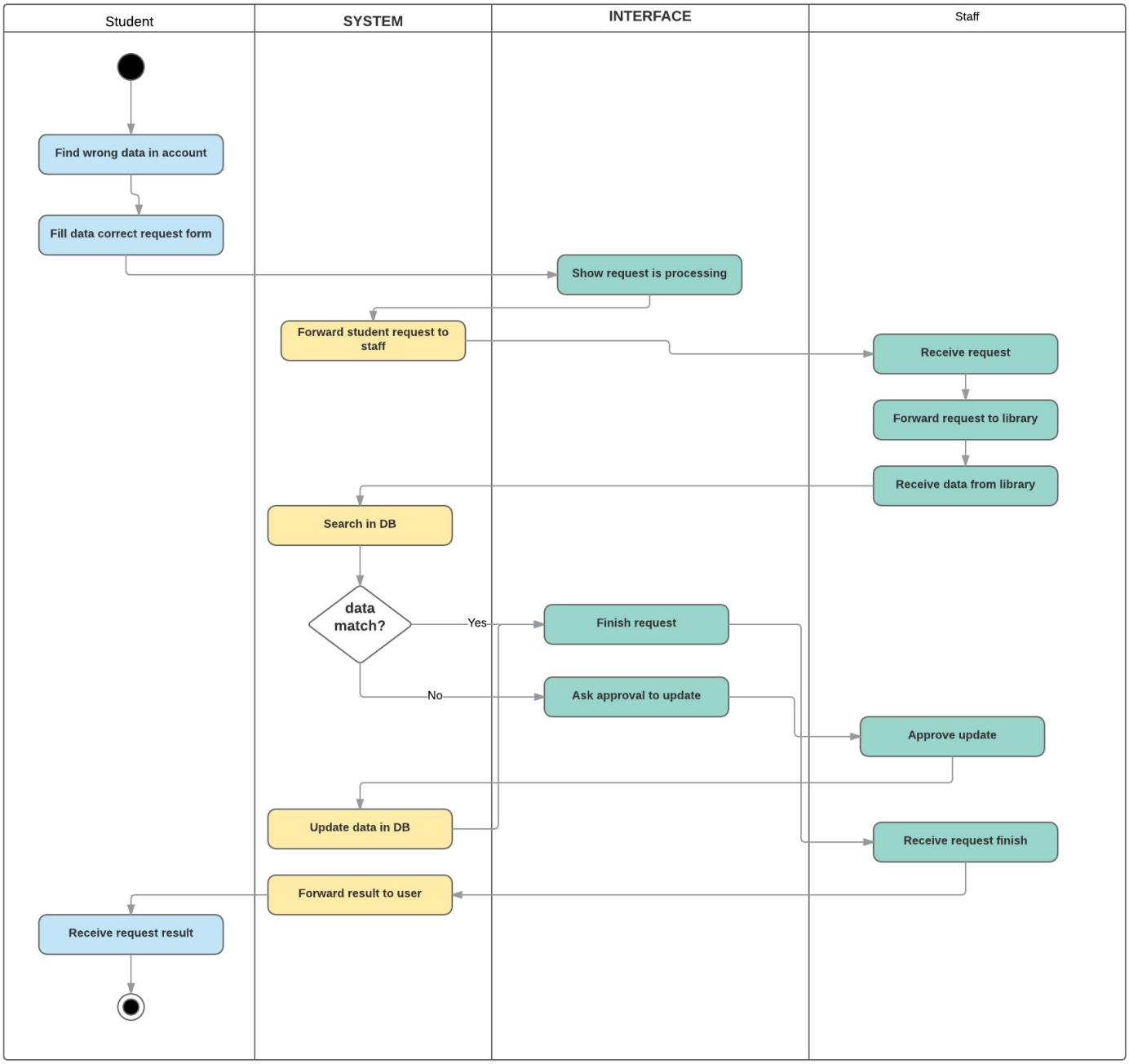
WENTING TANG | February 18, 2017

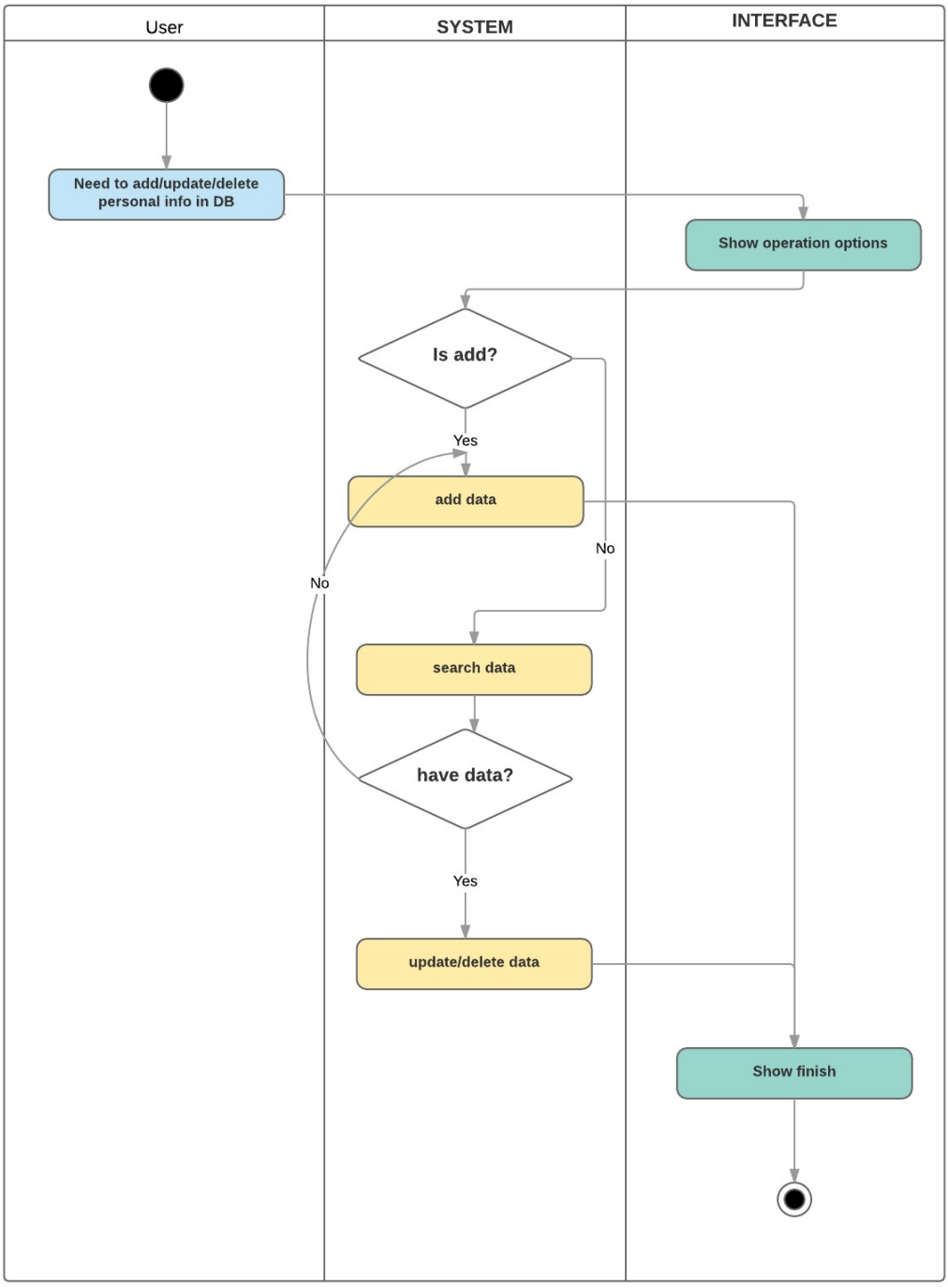


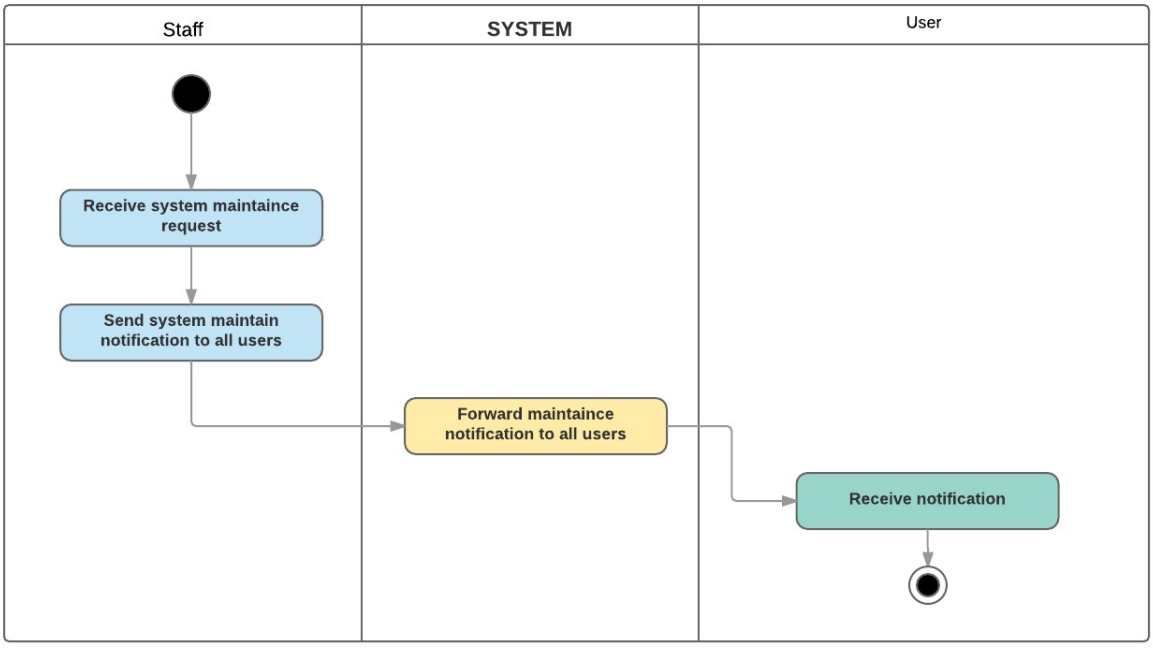

STAFF
SYSTEM
INTERFACE
Need to add/update/delete info in DB
Show operation options
Is add?
Yes
No
add data
No
search data
have data?
Yes
update/delete data
Show finish

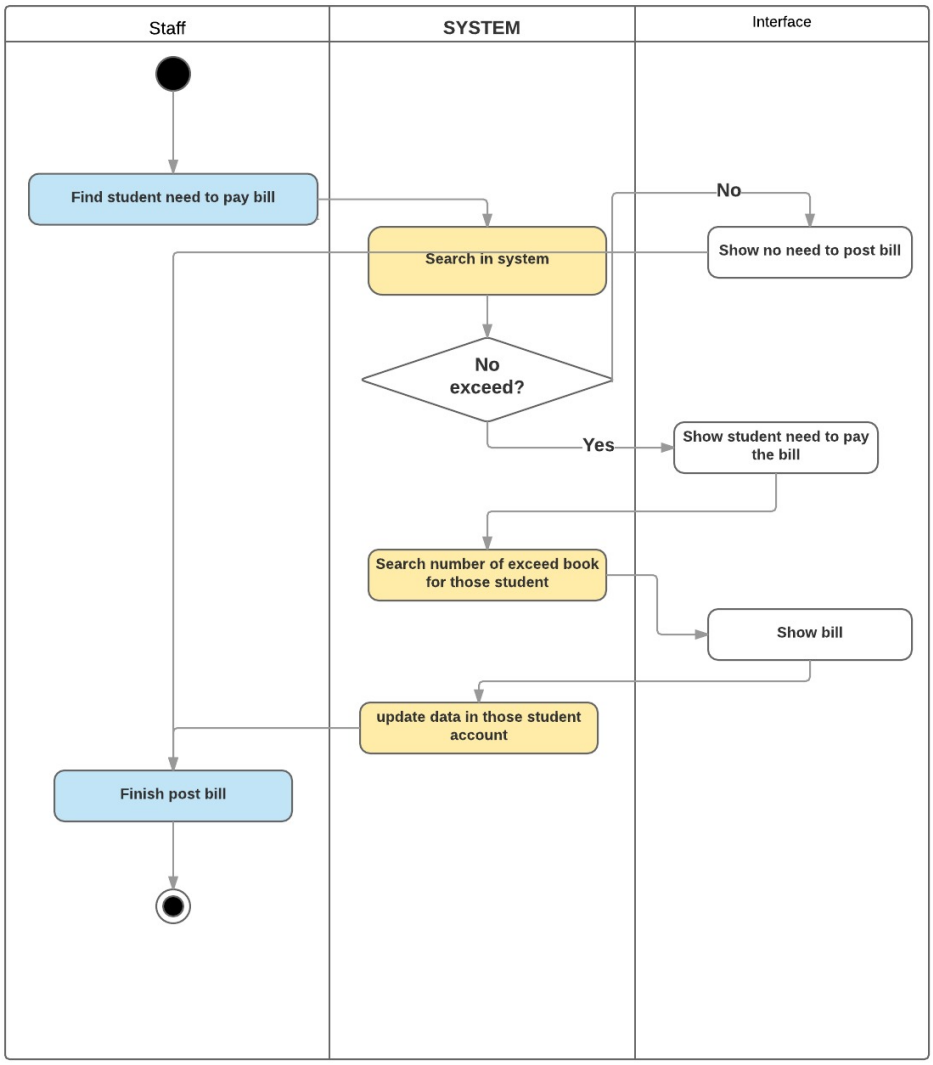

STUDENT
SYSTEM
INTERFACE
STAFF
Fill check out request form
Show request decline
No
Search book in DB
Book
info in system
Show request success
Yes
forward book info to staff
Receive check out book
request form
return book to library
book info changed in DB
Show book check out success
Inform student


Student
SYSTEM
Interface
Staff/teacher
Want to borrow books
Send check out book request
Forward request to staff/teacher
Show request is processing
Receive request
Process request
Search book in DB
Book in DB?
No
Show no book info, should decline
Yes
Search how many books has been checked out for this student
Show this studen can check out more books
Yes
Violate restriction
Show this student can check out those books
Check out book for this student
No
change info in DB
Show data has been update
Process finish
Send result to student
Forward request to student
Recieve result


Student
SYSTEM
INTERFACE
Staff
Find wrong data in account
Fill data correct request form
Show request is processing
Forward student request to staff
Receive request
Forward request to library
Receive data from library
Search in DB
data match?
Yes
No
Finish request
Ask approval to update
Approve update
Update data in DB
Receive request finish
Forward result to user
Receive request result


User
SYSTEM
INTERFACE
Need to add/update/delete personal info in DB
Show operation options
Is add?
Yes
add data
No
No
search data
have data?
Yes
update/delete data
Show finish


Staff
SYSTEM
User
Receive system maintaince request
Send system maintain notification to all users
Forward maintaince notification to all users
Receive notification


Staff
SYSTEM
Interface
Find student need to pay bill
Search in system
No
Show no need to post bill
No exceed?
Yes
Show student need to pay the bill
Search number of exceed book for those student
Show bill
update data in those student account
Finish post bill

3.0 Data Model and Description

3.1 data objects

Staff : This table stores the basic information of the staff Information, and manage the information for all users

```
create table staff(
  ID              char(256) not null,
  user ID          char(256),
  type            varchar(256),
  access level      varchar(256),
  name           varchar(256),
  phone number    char(256),
  primary key (ID)
  check  (type =‘student’ or type = ‘staff’ or  type = ‘teacher’ )
  check  (access level =‘1’ or access level = ‘2’ or  access level = ‘3’)
);
```

Teacher: This table stores the basic information of the teacher, and the the teacher shall check the books for his/her class.

```
create table teacher(
  ID              char(256) not null,
  user ID          char(256),
  type                      varchar(256),
  name                       varchar(256),
  access level        varchar(256),
  phone number             char(256),
  primary key (ID);
  check  (type =‘student’ or type = ‘staff’ or  type = ‘teacher’ )
  check  (access level =‘1’ or access level = ‘2’ or  access level = ‘3’)
 );
```

Class：This table stores the basic information of the class

```
create table class(
  name           varchar(256),
  ID                      char(256) not null,
  primary key (ID)
);
```

Book：This table stores the basic information of the book

```
create table book(
  ISBN number                        char(256) not null,
  school generated number          char(256) not null,
  name                                      varchar(256),
  primary key (ISBN number, School generated number)
);
```

Student :This table stores the basic information of the student and check out the state of student that is finish or checking_out

```
create table student(
  ID                                char(256) not null,
  User ID                           char(256),
  name                              varchar(256),
  parents name                      varchar(256),
  state                varchar(256)
  parents phone number              char(256),
  access level                      varchar(256),
  amount due                      varchar(256),
  type                              char(256),
  primary key (ID),
  check (state ='finish' or state = 'checking_out')
  );
```

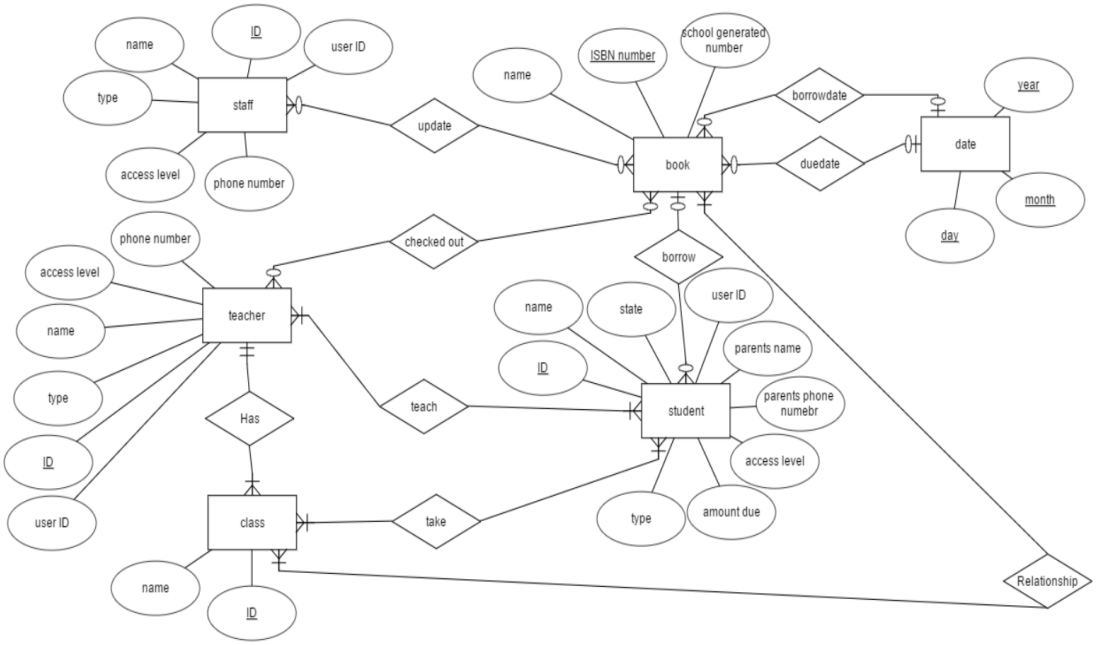
Date: This table stores the basic information of the date that are due date and borrow date.

```
create table date(
  year              numeric(4.0) not null,
  month             numeric(2.0.) not null,
  day               numeric(2.0.) not null,
  primary key (year, month, day)
);
```

3.2relationships

The entity book and entity date should have 2 relationships that are borrow date and due date.

3.3 Complete data model



4.0 Functional Model and Description

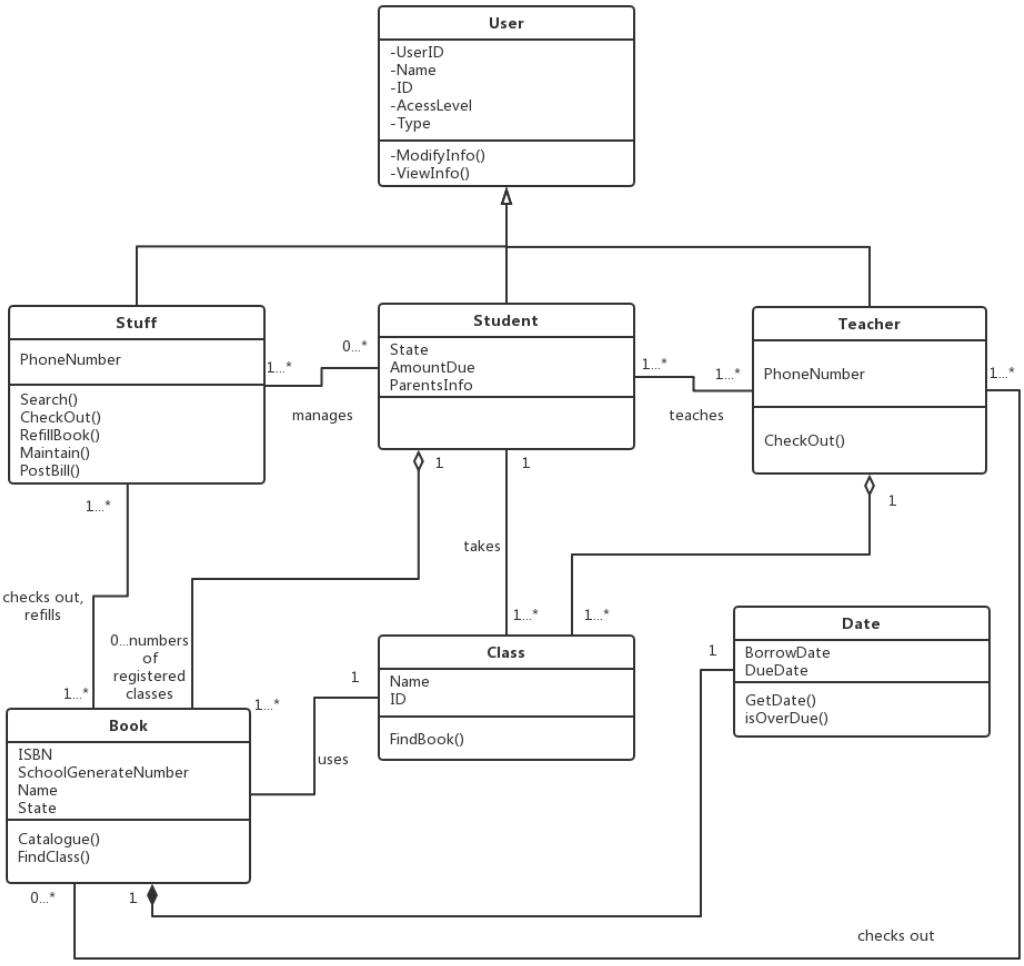
This is a high school book tracking application. Based on three different access levels, customers and administrator (staff) can interact with the system and achieve different operations.

4.1 Class diagrams

The class diagram shows some main classes in our system.

Organize the domain object classes into a hierarchy. User class is a super-class which reflects the common features of all users in this system. Sub-classes (staff, teacher, and student) inherit their attributes and services from super-classes, with their own features and operations.

Book class describes the features of the books provided to the users. Date class is a part of Book class. It includes two kinds of important information: check out date of a book and due date of a book. Class class includes the information of the class. Other detailed information is shown in the diagrams.



User Class (with subclasses: Staff, Teacher, and Student), Book Class, Date class,and Class Class

4.2 Software Interface Description

The software which is based on web will provide an interface to the

administrators and customers (different access levels) online.

4.2.1 External machine interfaces

There must be one server application which is necessary for the software to work,

- A web server.
- Need a web server to send and receive data over the Internet through simple clients and servers. All users (customers and administrators) can access the system both within the local network and from outside.
- There is a GUI for users to access the application.

4.2.2 External system interfaces

Since this is a web-based application, the external system interfaces will be the user's web browser which will be used to render the GUI for the user to access.

4.2.3Human interface

Users have appropriate access levels and privileges. Based on the access levels, operations can be performed by the customers and admin. Main human interfaces are described as follow:

1. Login

Main page

Name	Required Attributes	Description
Login	ID, userID	A user needs to enter his/her ID, userID, and password to transfer to 3 ports: staff, teacher, and student, or error case: invalid information or forget password.(If the times of entering incorrect password > max tries, the account will be locked.)
Forget Password	ID	If a user forgets his/her password or userID, he/she can transfer to the page of finding password.

Finding Password page

Name	Required Attributes	Description
Security Questions	ID	Need to answer some security questions about the user. Or wrong answers will lead to return to login page.
Get password and userID	ID	Send user's password and userID to his/her Email after a few seconds.

2. Student page (Level 1 access level)

Name	Required Attributes	Description

View	Student ID, userID	Student can view parents' information. If there is any book has been checked out by him/her, he/she can view the balance due.
Modify Parents Info	Student ID, userID	Allow student to create/delete/update parents' information.
Logout		Student can log out of this system.

3. Teacher page (Level 2 access level)

Name	Required Attributes	Description
View	Teacher ID, userID	Teacher can view personal information.
Modify Personal Info	Teacher ID, userID	Allow teacher to create/delete/update personal information.
Check Out	Teacher ID, userID, class	Teacher can check out books for students in his/her class.
Logout		Teacher can log out of this system.

4. Staff pages (Level 3 access level)
Main page

Name	Required Attributes	Description
View Info	Staff ID, userID	Staff can list all students with books checked out, all authorized users and their access level, students and their parents' info with amount due.
Modify Info	Staff ID, userID, userID or ID of modified user	Allow staff to create /delete/update all users/books/ classes information.
Search	Book's ISBN, or school generated number, or class name/ID.	Staff can search and list all books in the system, all books checked out by class or book name/number.
Check Out	Staff ID, userID	Staff can check out books for student.
Post Bill	Staff ID, userID, Student ID	Staff can post a bill to the student who holds a book beyond the due date.
Refill	Staff ID, userID, Book's name or ISBN	Staff can refill any book if needed.

Maintain	Staff ID, userID	If our system is in error cases, staff can click the maintain button to transfer to the maintain page and do some operations (books are not enough, account lock and recovery, provide new userID for student etc.)
Logout		Staff can log out of this system.

Maintain page (Only staff can access this page!)

Name	Required Attributes	Description
Account Lock	Staff ID, userID and userID of an illegal account	If there are some errors or exceptional cases (illegal login) in an account, staff has right to lock it.
Account recovery	Staff ID, userID and userID of an illegal account	Staff can recover an locked account.
New userID	Staff ID, userID, student ID	Staff can provide a new userID for student.
Label	Staff ID, userID, Book's school generated number	Staff can mark some status for books (not enough, damaged etc.).

5.0 Behavioral Model and Description

5.1 Description for software behavior

The following consists of the descriptions about the various states that form a part of the system.

5.1.1 Events

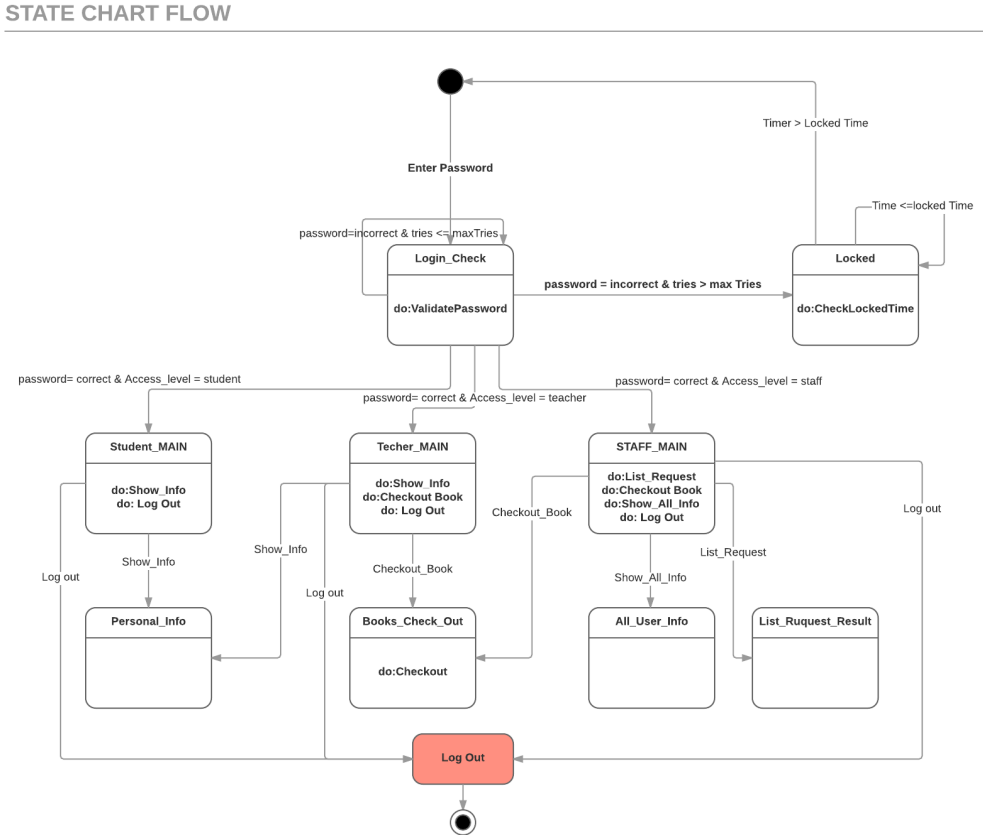
Sr No.	Event	Actor
0	Open home page	All
1	Create Login	All
2	Change Password	All
3	User log in/log off	All
4	Account Recovery and Lock	Staff
5	View, add, update, delete Student's info	Staff, Student
6	View, add, update, delete Teacher's info	Staff, Teacher
7	View, add, update, delete Books' info	Staff
8	Refill Books	Staff
9	Check out Books	Teacher, Staff
10	List all students with books checked out	Staff
11	List all books in the system with search capabilities	Staff
12	List all authorized users and access level	Staff
13	List all books checked out by class or book name	Staff
14	List students and parents info with amount due	Staff

5.1.2 States

S.no	State Name	State Description	Events handled
0	START	The initial state of the system. When the system starts, we initialize and move on to the LOGIN state	Initialize
1	LOGIN-CHECK	If login is validate, check the access level of users.	Display_User_main
2	LOCKED	If login is invalidate two many times, account locked.	Login_incorrect
3	STUDENT_MAIN	This is the main page of students. Students can view update personal information	update information, log out
4	TEACHER_MAIN	This is the main page of Techer. Teacher can check out books for students, view and update personal information.	update information, check out books, log out
5	STAFF_MAIN	This is the main page of Staff. Staff can do everything with highest level. Finish the List_request	Checkout books, account lock and recovery, log out, etc.
6	Personal_Info	Show students and Teacher's Personal Information	
7	All_User_Info	Staff views and updates all users' info	

8	Books_Check_Out	Staff and Teacher checkouts books here.	
9	STOP		

5.2 State Chart Diagram



6.0 Restrictions, Limitations, and Constraints

Next, we need to focus on the Restrictions, Limitations, and Constraints. In the below, it explains the cases that have impact on the specification, design and implementation of the whole software system. Also, they contribute to restrict the scalability and performance of the whole system.

Constraints

1. The information of all users and books must be stored in a database that is accessible by the Book Tracking Application
2. The users must have their correct usernames and passwords to enter into the application.
3. Users with different access-level only have their independent usernames and passwords.

Assumptions and dependencies

- The users have sufficient knowledge of computers.
- The users know the English language, as the user interface will be provided in English
- Application manager has created Usernames and passwords for users in advance.

7.0 Validation Criteria

The test plan describes the overall methods to be used to verify that the software meets the product specification and the customer's needs. It includes the quality objectives, resource needs, assignments, methods, and so forth.

7.1 Classes of tests

Types of tests to be concluded:

- Unit Testing
- Integration Testing
- System Testing
- Black box Testing

Starting from unit testing, then we will do the integration testing and system testing. In the testing process, we will authenticate the integrity, security and quality of the software to access if it can meet the requirements.

7.2 Expected software response

Test cases will be generated for each of the modules and analysis will be done to check if we get the expected results or not. Some of the examples and main expected software responses should be:

No.	Name	Expected Response for Success Cases	Expected Response for Failed Cases
1	Login	Show success and jump to the corresponding port: staff, teacher, and student.	Show error message: invalid account, try again. If times of trying beyond max tries, the account should be locked.
2	Create Info	Additional Information should be shown correctly on the page, and write it in the corresponding file.	Show failed message: Creating failed, try again. Or you are not authorized to do this operation.
3	Delete Info	Deleted information should be removed from the page, and delete it in the corresponding file.	Show failed message: Deleting failed, try again. Or you are not authorized to do this operation.
4	Update Info	Change information and update to the latest information, refresh the page.	Show failed message: Updating failed, try again.
5	View Info	List the information correctly for the user as needed.	Show failed message: Errors! Please check your network connection.
6	Check Out	Change a book's status, and change it in the corresponding file.	Show failed message: Operation failed. Or you are not authorized to do this operation.

7	Refill	Change the numbers of the book, and change it in the corresponding file.	Show failed message: Refilling failed. You are not authorized to do this operation.
8	Post Bill	Add some bill's information for a student, and write it in the corresponding file.	Show failed message: Operation failed. Or you are not authorized to do this operation.
9	Search	Retrieve books by given valid information, show book's information if there is any, show none if there is no one.	Show failed message: Search failed, try later. Or you are not authorized to do this operation.
10	Maintain	Implement the operations correctly. And show success message.	Show failed message.

7.3 Performance bounds

1. Information effectiveness/completeness

Information or operation should not be processed if required fields are invalid or incomplete.

2. Privilege

Every operation based on this system should follow the authority of the corresponding access level. These tests all include an access level checking to guarantee the security and privileges. If user try to do some operations beyond his/her access level, the system will show " You are not authorized to do this operation.".

3. Response time

The time taken to operate books (check out and refill), add information, delete information, update information, search and view information should not take more than 5 seconds time to retrieve the information and display to the user (students, teachers, stuffs). It will give correct information immediately when error occurs without any interrupt. The system shall be allowed to take more time when doing large processing jobs.

4. Capacity

For each check out operation, user can check out 1 book each time.