Notes:

1. You can compile and submit your report in either Chinese or English. You are not necessarily 100% correct in language when using English. However, you should make sure you convey everything in a clear and understandable manner.

2. Document any reasonable assumptions you make when analyzing the framework, such as peak load conditions or network latency.

3. There is no minimum length requirement for this assignment, but it should not exceed TWENTYpages. Be concise and to the point.

4. You can use UML, or other appropriate notations for diagrams. Clearly explain any non – standard symbols used.

[1] Brandolini A. Introducing event storming[J]. blog, Ziobrando's Lair, 2013, 18.

[2] Study Materials: https://github.com/mariuszgil/awesome-eventstorming

Appendix: **Case Description**

The Library Management System comprises eight business services, namely: Borrow Book, Return Book, Calculate Overdue Fines, Renew Book, Reserve Book, Search Book, Pay Fine, and Manage Reader Account. The development details of each service are described as follows:

1) Service Name: Borrow Book
Service Description: As a reader, I want to borrow books so that I can read them.
Triggering Event: The reader clicks the "Borrow" button in the system interface.
Basic Flow:

1. Validate the reader's account (i.e., ensure it is registered, not frozen, and has no outstanding fines);
2. Check if the book is available for borrowing (i.e., status is "available" and not reserved);
3. Record borrowing information (reader ID, book ID, borrowing date, due date);
4. Update the book status to "borrowed";
5. Notify the reader of successful borrowing.

Alternative Flows:

1a. If the account is invalid or frozen, prompt: "Account unavailable, please contact the administrator";

2a. If the book is unavailable, prompt: "Book not available due to being borrowed or reserved";

3a. If the borrowing record fails to save, prompt: "System error, please try again."

Acceptance Criteria:

- Reader account is valid and has no unpaid fines;
- Book status must be "available" and not reserved;
- Borrowing record includes both the borrow date and due date (default loan period: 30 days);
- After status update, other users cannot borrow or reserve the book;
- Reader receives a confirmation notification via SMS or email.

2) Service Name: Return Book

Service Description: As a reader, I want to return books so that they become available to others.

Triggering Event: The reader scans the book's barcode at a self-service return station or clicks the "Return" button in the system.

Basic Flow:

1. Verify that the book belongs to the current reader;
2. Check for overdue status—if overdue, trigger the "Calculate Overdue Fines" service;
3. Update the book status to "available";
4. Update the borrowing record to "returned";
5. Create a return record;
6. Notify the reader of successful return. If the book is reserved, notify the highest-priority reserving reader.

Alternative Flows:

1a. If the book does not belong to the reader, prompt: "Return failed. This book is not associated with your account";

2a. If overdue, prompt: "Overdue fine required: XX RMB";

3a. If status update fails, prompt: "System error. Please contact the administrator."

Acceptance Criteria:

- Book must be correctly associated with the reader account;
- Overdue fines are automatically calculated and added to the reader's account;
- Once returned, the book becomes immediately borrowable or reservable;
- Return records include return date and operator (e.g., self-service machine ID).

3) Service Name: Calculate Overdue Fines

Service Description: As the system, I want to automatically calculate overdue fines to ensure timely book returns.

Triggering Events:

- Book is returned past the due date;
- Daily scheduled scan of overdue records at 1:00 AM.

Basic Flow:

1. Calculate the number of overdue days (current date – due date);
2. Compute fine based on policy (e.g., 1 RMB /day, maximum 50 RMB);
3. Generate fine record (reader ID, book ID, amount, overdue days);
4. Update the reader's "outstanding fine" amount;
5. Notify the reader with fine details.

Alternative Flows:

1a. If due date is missing, log an error and skip processing;
2a. If fine rules are not configured, apply default rule (1 RMB /day).

Acceptance Criteria:

- Overdue days must be a positive integer;
- Fine amount must comply with predefined rules;
- Fine records must be persistently stored;
- Reader's outstanding fine field is updated in real time.

4) Service Name: Renew Book

Service Description: As a reader, I want to renew borrowed books to extend the loan period.

Triggering Event: The reader clicks the "Renew" button in the system interface.

Basic Flow:

1. Validate whether the book is eligible for renewal (i.e., not reserved, not overdue, and remaining renewals > 0);
2. Extend the due date (original due date + renewal period, e.g., 15 days);
3. Decrease the remaining renewal quota (e.g., max one renewal per book);
4. Notify the reader of successful renewal.

Alternative Flows:

1a. If the book is reserved, prompt: "Renewal denied due to an active reservation";
2a. If renewal quota is exhausted, prompt: "Renewal limit reached."

Acceptance Criteria:

- Due date must be extended after renewal;
- Renewal record must include original borrow ID and new due date;
- Reader receives a renewal confirmation notification.

5) Service Name：Reserve Book

Service Description： As a reader, I want to reserve borrowed books so that I can borrow them as soon as they are returned.

Triggering Event： The reader clicks the "Reserve" button on the book details page.

Basic Flow：

1. Confirm that the book status is "borrowed";
2. Check whether the reader has exceeded the reservation limit (e.g., max 3 books);
3. Record reservation details (reader ID, book ID, reservation timestamp, priority);
4. Set the book's reservation status to "true";
5. Notify the reader of successful reservation.

   Alternative Flows：

   1a. If the book is available, prompt："Reservation unnecessary. You may borrow it now";

   2a. If reservation limit is reached, prompt："Maximum of 3 active reservations allowed."

   Acceptance Criteria：

- Only "borrowed" books may be reserved;
- Reservation queue must be sorted by timestamp (priority);
- Upon return, the system automatically notifies the top-priority reserver.

6) Service Name：Search Book

Service Description： As a reader or administrator, I want to search for books to locate their availability and position.

Triggering Event： User enters a keyword in the search bar and clicks "Search."

Basic Flow：

1. Retrieve books using keywords (e.g., title, author, ISBN);
2. Return a result list (including title, author, status, shelf location);
3. Allow access to detailed view (borrowing history, reservation queue, fine history).

   Alternative Flows：

   1a. If no results are found, show："No matching books found";

   2a. If query times out, show："Network busy, please try again later."

   Acceptance Criteria：

- Support fuzzy search (e.g., "Harry" matches "Harry Potter");
- Results must be paginated (20 items per page);
- Details page must display book cover and summary (if available).

7) Service Name：Pay Fine

Service Description：As a reader, I want to pay overdue fines to remove borrowing restrictions.

Triggering Event：Reader clicks "Pay Fine" on the "My Account" page.

Basic Flow：

1. Retrieve the amount of outstanding fines;
2. Integrate with payment gateway (e.g., Alipay, WeChat Pay, UnionPay);
3. Upon successful payment, update fine status to "paid";
4. Lift any borrowing restrictions on the reader.

   Alternative Flows：

   1a. If payment fails, prompt："Payment failed. Please check your account balance";
   2a. If payment gateway is unavailable, log the error and prompt："System busy."

   Acceptance Criteria：

- Payment amount must exactly match the outstanding amount;
- Generate an electronic receipt upon successful payment;
- Reader's outstanding fines field is cleared.

8) Service Name：Manage Reader Account

Service Description：As an administrator, I want to manage reader accounts to ensure system security and data accuracy.

Triggering Event：Administrator clicks "Add/Edit/Freeze Reader Account" in the backend system.

Basic Flow：

1. Add reader：input name, ID number, and contact details;
2. Edit reader：update contact details or borrowing privileges;
3. Freeze account：disable borrowing/renewal for rule-violating users;
4. Log all operations (operator, timestamp, action type).

   Alternative Flows：

   1a. If ID number already exists, prompt："Reader already registered";
   2a. If required fields are missing, prompt："Please complete all required fields."

   Acceptance Criteria：

- Reader ID number must be unique;
- Frozen accounts cannot initiate borrowing or renewal operations;
- All actions must be recorded in an audit log.