Assignment T1/T2

Team IceBerg
Zeyang Chen (zc2483)
Wenjie Chen (wc2685)
Chong Hu (ch3467)
Weijie Huang (wh2447)

Last meeting date with IA: Nov 3rd
IA's feedbacks are marked with blue; revisions are marked with red.

Part 1:

1. What will your project do?

The application value of the budget reimbursement system is reflected in three functional parts: the budget function; the reimbursement function and the data visualization function.

The administrator(e.g. CEO of the company) can create groups. Groups specify different sectors of a company, such as development group, testing group, hr group. Each group can create an account for each employee. Employees can use this account to apply for financial reimbursement. The group manager can review and approve employee claims. In addition, the employee can view the financial status of the reimbursement data over a period of time. (What is the analysis function?) Through the dashboard, one can see the details of the reimbursements in numbers and different charts in order to insight from historical reimbursements and manage expenses.

2. Who or what will be its users?

The target user can be Normal Employee; Group Manager; Administrator(CEO);

```
                    Administrator
                    /           \
Group 1(Group Manager)     Group 2(Group Manager)
     /      \                    /      \
Employee 1   Employee 2     Employee 3   Employee 4
```

About authentication: we will provide a signup and login button in the home page. (Previous two sign up ways are somehow in conflict. One signup function can keep the process consistent)

1. The administrator can create an account for a new employee with corresponding authorization.

2. A person can just sign up through the default account and password provided by the administrator. During this process, the user needs to provide his personal information, such as email address, bank account.
3. The user can also login into the system after initializing his/her account.

The Group Manager account and the Administrator account can only be created by the previous administrator, and cannot be created by using the signup function.

3. What do you think you'll be able to show in your demo?

We will display a web page, and then show that the administrator creates the managers of each group and how employees login into the system. Before the demo, we will prepare several data in the system to visualize the result better. We will also show the employee submits the reimbursement certificate, the manager reviews the reimbursement certificate, and the administrator checks the company's reimbursement data. (extra external APIs can be applied, such as some OCR API or associating bank account) When employees upload their invoice picture, the system will automatically detect text in the picture and generate a pre-filled form for the employee to check and confirm. After the reimbursement is approved by the manager, the system will send corresponding money to the employee's bank account (we will use some fake accounts in the demo) and send an notification email to the employee. When the manager views statistics through the dashboard, the system will provide data and different charts.

4. About database

We will store every user's information, including name, roles, id, account name and password. Each user account will be associated with corresponding email address and bank account. Also, for every user, we will store their reimbursements, including submitted date, amount of money, description, invoice picture and request status. For every group, we will store group id, group name and members' id.

5. About using external APIs:

- **Receipts/Invoice OCR**
  - https://www.veryfi.com/
  - This is an OCR API, and it will scan the e-bill or invoice of the user and output the result in JSON format. We will use it to scan the invoice and fill the reimbursement application automatically instead of users filling it by themselves. Users can upload their invoice through the front-end and only need to check and revise the form provided by our system.
- **Stripe Connect API**
  - https://stripe.com/docs/api

- We can use the Stripe API to create the visual payment accounts for accepting payment and sending payout and managing their businesses.
- Basically, we want to use this API to help us simulate the whole reimbursement process that is one person makes an application and if it has been approved, then the money will go into his account from the company's official account.

Part 2:

1. Employee

a). As an employee, I want to apply for reimbursement so that I can upload invoices to submit an expense account.
My conditions of satisfaction are:
Users can fill reimbursement forms and upload invoices to the financial department to examine.

b). As an employee, I want to check reimbursement status so that I can keep track of the reimbursement process and manage my expenses.
My conditions of satisfaction are:
i. Users can check the status of each reimbursement.
ii. Users can revise or supplement their reimbursement information.
iii. Users can check how much reimbursement amount they can still use for the current month.

c). As an employee, I want to revoke my request so that I can make some changes if I typed in wrong information.
My conditions of satisfaction are:
i. I can revoke my processing requests, then the status of that request will be changed to "CANCELLED".

d). As an employee, I want to receive emails so that I can get noticed when I successfully post a request or my requests' status changes.
My conditions of satisfaction are:
i. When I finish a request and submit, I should receive an email that indicates whether I made a valid request.
ii. When group managers approve or deny my request. I should receive an email that tells me how my request is going so that I needn't log in to the system to do that.

e). As an employee, I want that it can automatically help me fill the reimbursement application form when I upload the scanning file of my invoice or receipt, so that it will save me some time.
My conditions of satisfaction are:

i. I can upload my scanning file of invoice and receipt.

ii. The system can recognize the character on my file and fill in the form automatically (it is acceptable that there may be some mistakes), and provide me with the form that I can re-edit and revise the information.

f). As an employee, I want to see my statistics of my historical reimbursements so that I can manage my expenses better.
My conditions of satisfaction are:

i. I can view my historical reimbursement data in terms of different charts, such as pie charts (different types of reimbursements), bar charts (expense versus data), etc.

ii. I can check how much budget that I can still use this month and how much reimbursement amount I used this month.

2. Group Manager

a). As a group manager, I want to examine and approve all reimbursement applications submitted by employees so that I can manage all reimbursements through the system.
My conditions of satisfaction are:

i. Group managers can examine and approve reimbursement applications submitted by employees.

ii. Group manager can change the status reimbursement applications to PROCESSING, MISSING INFO, DENIED, APPROVED.

iii. Group manager can revise the reimbursement limits for next month.

b). As a group manager, I want to check my group members' reimbursement history so that I am able to know my group's budget.
My conditions of satisfaction are:

i. On the group page, I can click a member's name and see all his history reimbursement information.

ii.On the group page, I can see all visualized statics charts of the group's budget. I can view my group historical reimbursement data in terms of different charts, such as pie charts (different types of reimbursements), bar charts (expense versus data), etc.

3. Administrator

a. As an administrator, I want to manage the reimbursement groups accounts so that I can give them the right to the group manager to manage their group member's reimbursement.
   My satisfaction conditions are:
   I. Administrator can create, edit or delete the accounts
   II. Administrators can create many groups depending on setting different group numbers.
b. As an administrator, I want to have the report of the reimbursement statistics of each group so that I can understand the allocation and use of the company's funds.
   My satisfaction conditions are:
   I. Administrators can see the reimbursement statistics of each group.
   II. The application displays various statistical visualization images for administrators. I can view my group historical reimbursement data in terms of different charts, such as pie charts (different types of reimbursements), bar charts (expense versus data), etc.

Part 3:

Normal User/Employee:

Acceptance test:

1. Given an employee, he is able to post reimbursement requests. When he clicks "Request for reimbursement", then fills in the forms of usage, amount, date and upload the invoice. Then he can post a request.
2. Given an employee, he is able to check the status of his past requests through his own page. The status may include "PROCESSING", "CANCELLED", "DENIED", "MISSING INFO" and "APPROVED".
3. As an employee, he is able to cancel his past requests by himself. But the request won't disappear in the system. Its status will be changed to "Cancelled".
4. **(Special case)** Given an employee, when he wants to create or edit some accounts, the system should fail his action.
5. Given an employee, when he uploads his scanning file of invoice or receipt, the system will extract the information of the file and automatically fill in the form.
6. Given an employee, when he submits the application, he will receive an email from the system notifying that his application has been successfully submitted and need to wait for the response.
7. Given an employee, when his application has been approved or denied, he will receive an email telling him the review status of the application.

8. Given an employee, when he want to view his reimbursement, he is able to view analysis including different types of reimbursements (pie charts), expense versus data (bar charts)

Group manager:

Acceptance test:

1. Given a group manager, he or she is able to check the reimbursement requests of the whole team. The requests include text submission and image of invoice evidence. Then he or she is able to directly reject, approve, or request more necessary information.
2. Given a group manager, when a normal user opens a request case, he or she is able to change the state of the case including processing the request, case resolved.
3. Given a group manager, when he or she wants to check the team budget or reimbursement summary, the system allows him or her to check the status and analysis of the summary.
4. Given a group manager, when he or she wants to know the finances of the team, he or she is able to view the teams' different types of reimbursement, expense versus data.
5. **(Special case)** Given a group manager, when he wants to create an administrator account, the system will send an error and fail his action.

Administrator:

Acceptance test:

1. Given an administrator of the application, when he wants to create the reimbursement groups for each company, then the application will provide him with a form to fill in some information about the groups and accounts and creates the groups for the administrator.
2. Given an administrator of the application, when he wants to see the statistical charts of his company in the system, then it will show him the reimbursement statistics of each group in his company.
3. **(Robustness)** Given an administrator, when he or she is trying to create groups/accounts and the website/network fails, the system is able to resume to the latest state.
4. Given an administrator, when he or she wants to know the finances of the company, he or she is able to view the companies' different types of reimbursement, expense versus data.

5. **(Special case)** Given an administrator, when he wants to edit the existing group's ID number, then the system will send an error and fail his option because the group's ID number cannot be changed once it has been created.

Part 4:

JDK: Jdk1.8
IDE: IntelliJ IDEA
Package manager: Maven 3.6.1
Code style: IDEA Google CheckStyle
JUnit: JUnit 5
Code coverage: EMMA for IntelliJ
Bug finder: SpotBugs for IntelliJ
Database: SQLite / mysql 8.0.17
(Front-end framework?)
Front-end: HTML+CSS+JS
Back-end: Spring Boot + Mybatis