

CSC207 Phase 0 Report

Tian Shu, Canyang, Shihan,
Yuelin, Yiteng, Mihir

Specification:

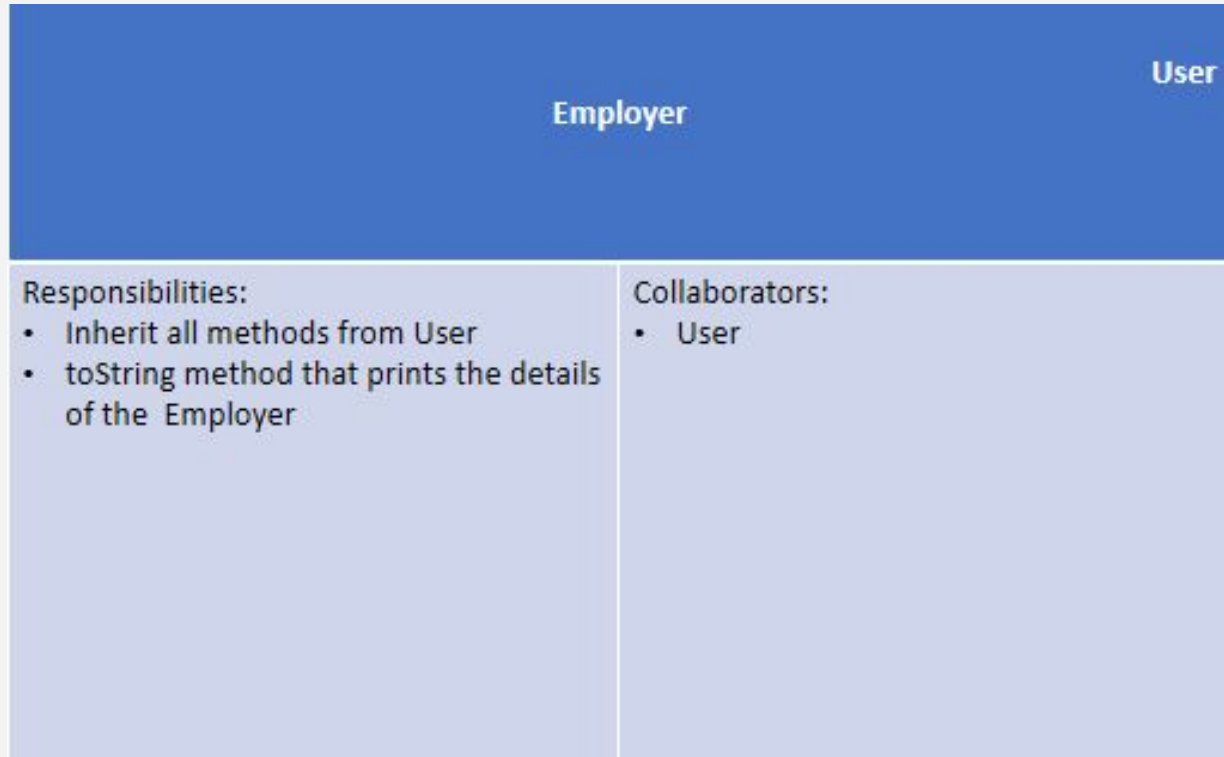
- HR System
- User → Employer or Employee
 - a username, a password, salary, attendance, id
 - Employee: access information
 - Employer: access information of all users, has editing capability
- UI Commands
 - create the personal profile for new employees
 - allow the employer to change information of employees
 - allow employees to check their information. e.g. salary and id
 - exit the system

CRC Model

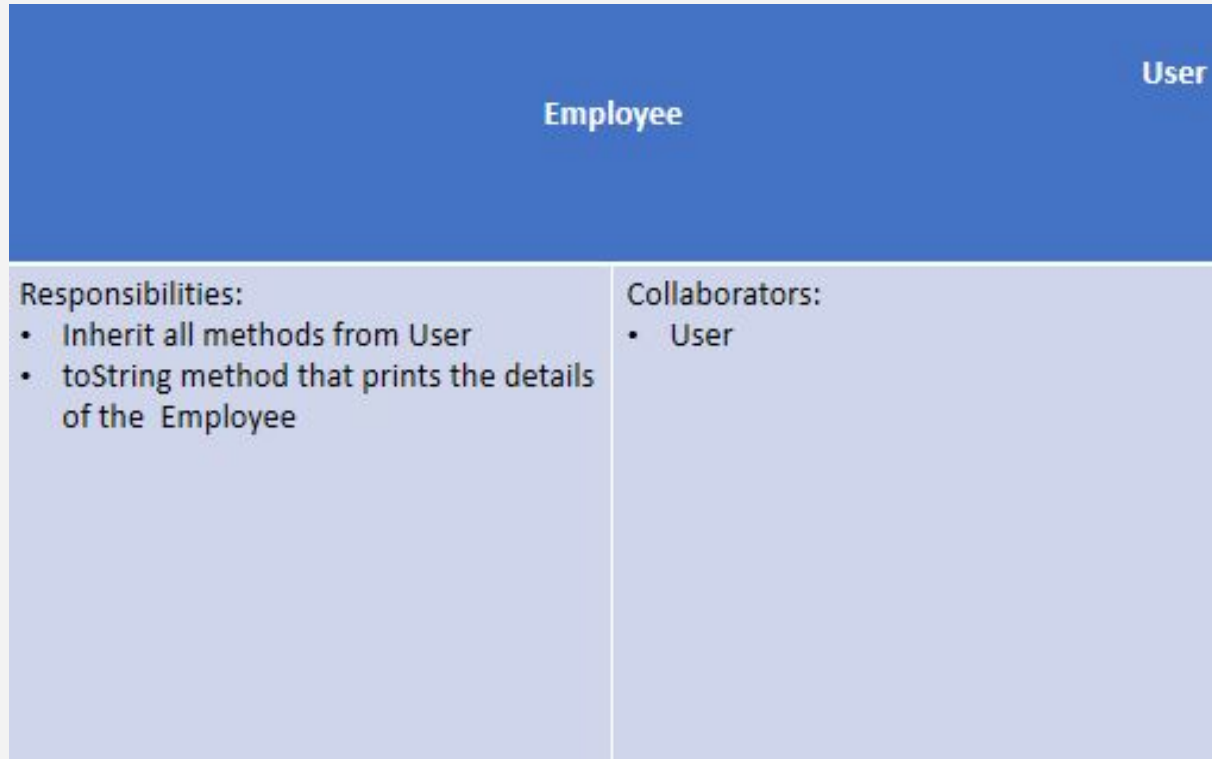
CRC Model

Abstract	
User	
Employee, Employer	
<p>Responsibilities:</p> <ul style="list-style-type: none">• Stores ID , username, password, salary, and attendance.• Getters for everything above• Setters for everything above	<p>Collaborators:</p>

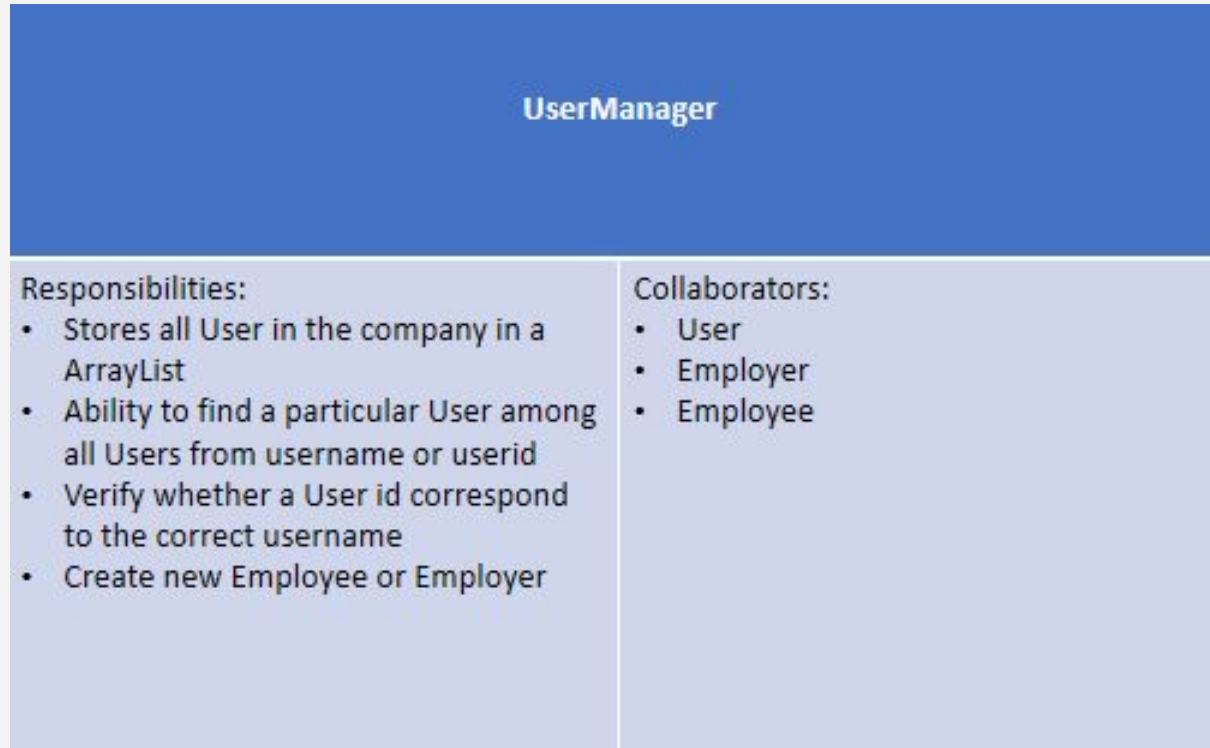
CRC Model



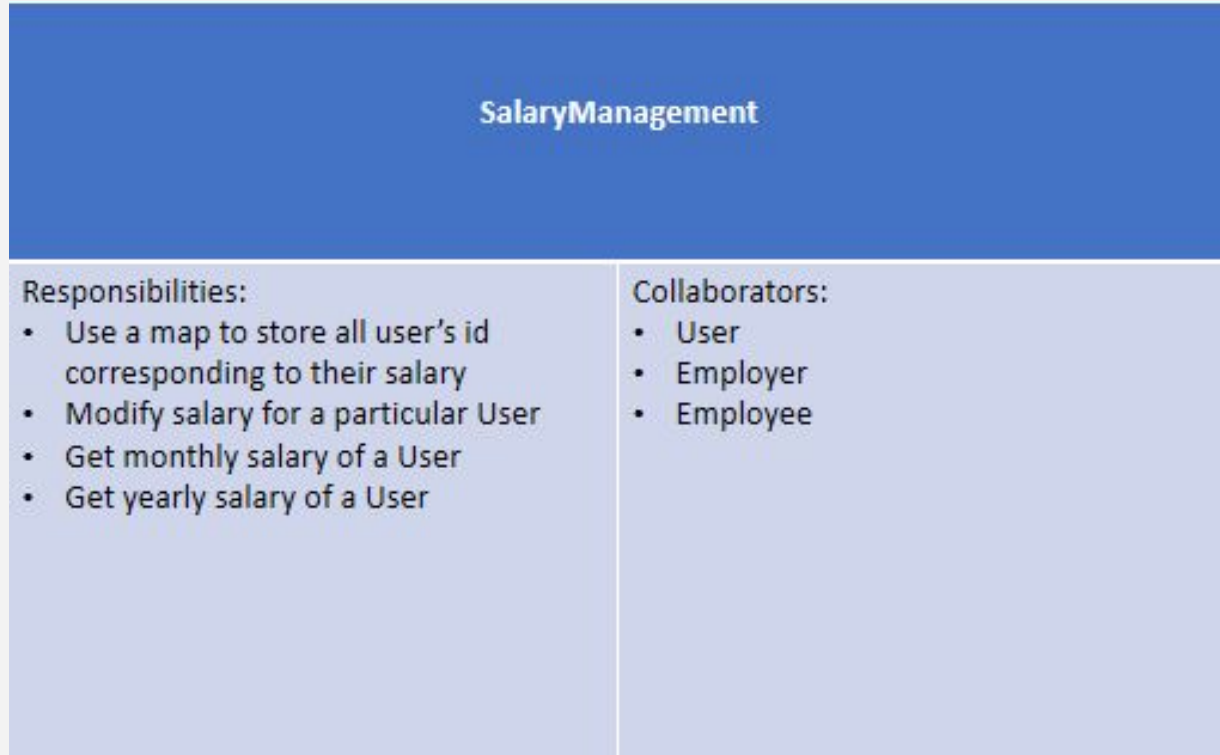
CRC Model



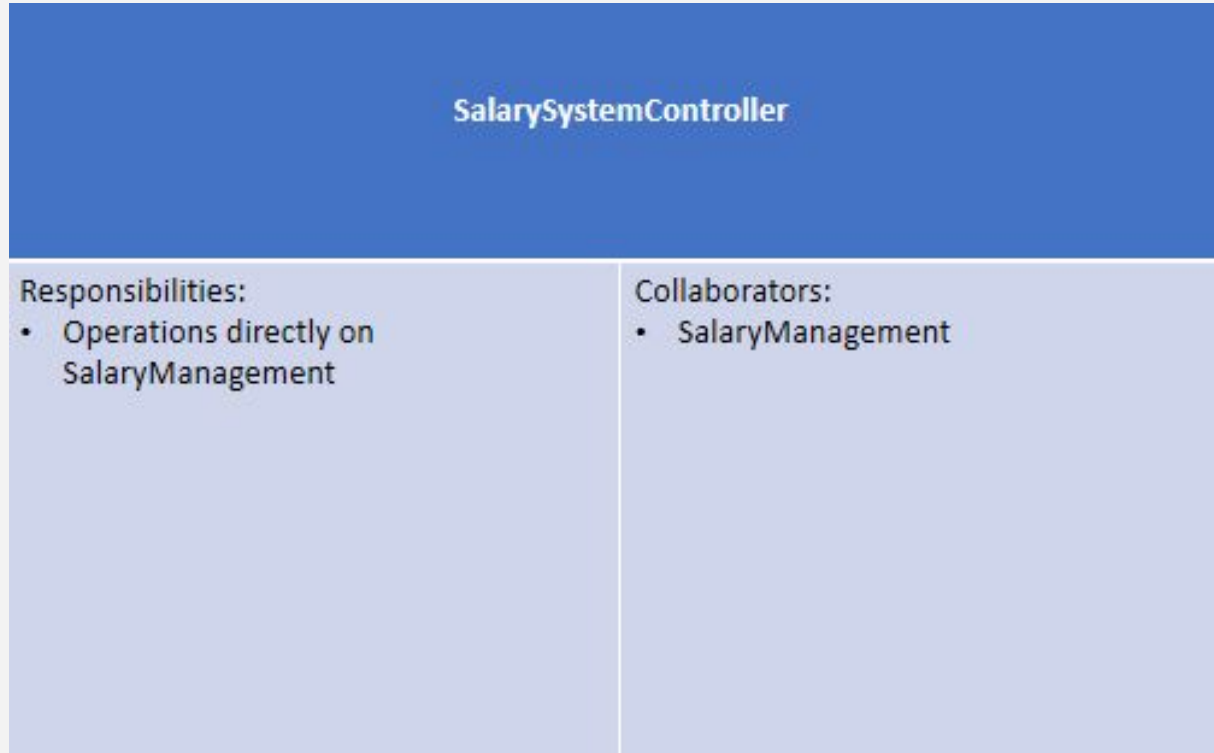
CRC Model



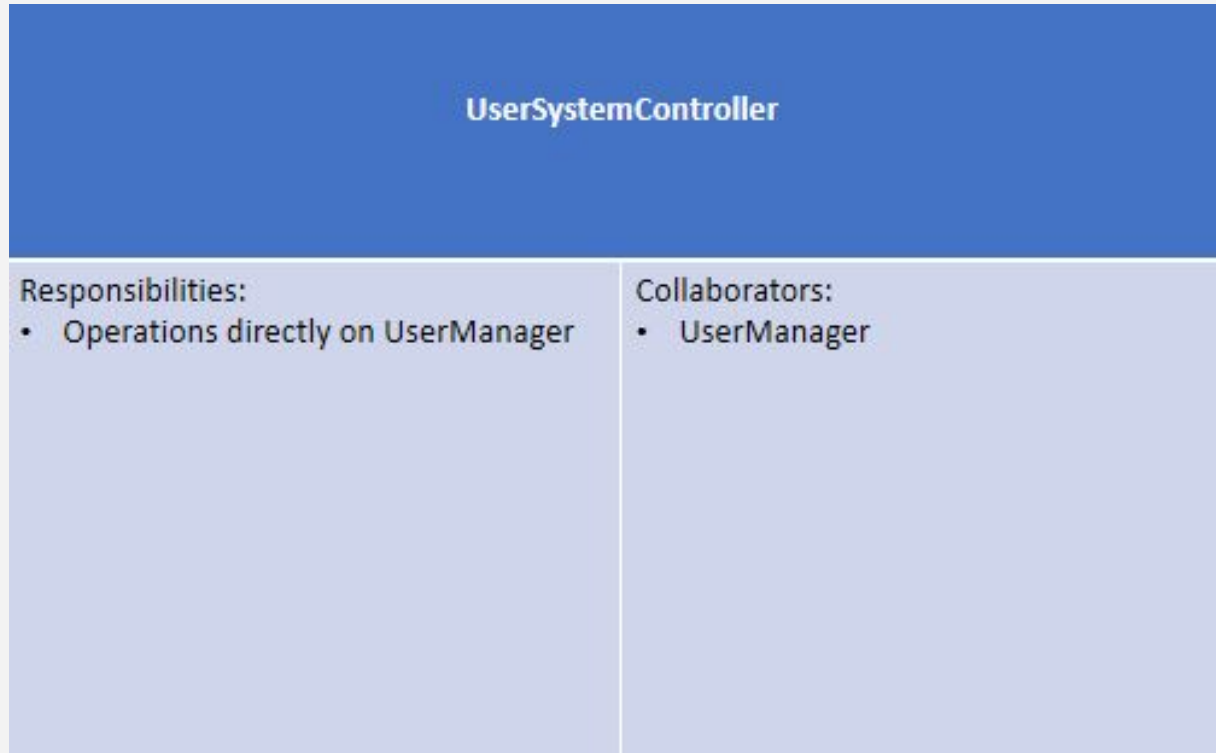
CRC Model



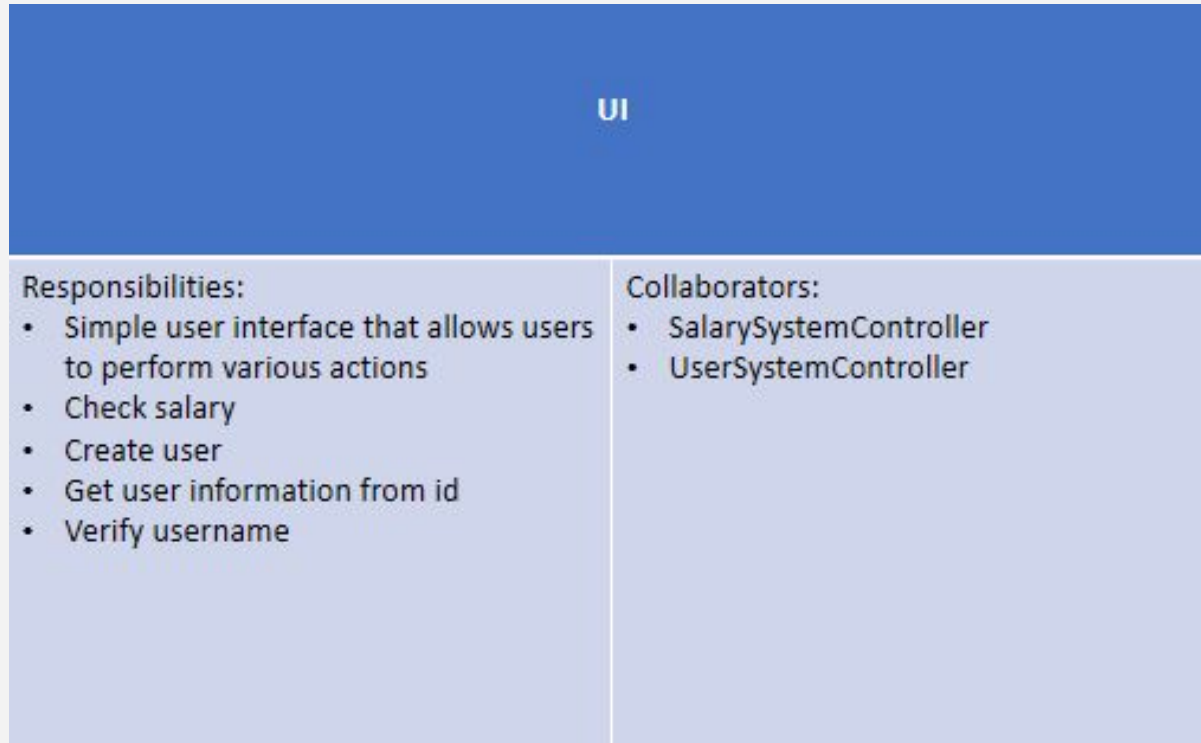
CRC Model



CRC Model



CRC Model



Scenario Walk Through

❖ Employer Scenario:

- Firstly, **employer** uses this HR system and the system shows him the **log-in page** using the UserSystemController.
- Register himself through the **registration button** UserManager use case, we will need to enter his **ID, username, password, salary, and attendance**.
- After he **successfully registered** himself, the system will **send him back to the login page**.
- The employer logs in, then we can see **a list of all the employees with their detailed information**.
- The employer **has access to all the files and can edit** if the employer wants to raise or lower an employee's salary using the SalaryManagement use case of the SalarySystemController.
- Noticed that **only the employer has the authorization to create an employee account**. *(So far, everyone can create a User since our code did not add the restriction on different users' rights.)*

❖ Employee Scenario:

- An **employee** who is recruited by an employer should **get his username and temporary password from the employer**. The system will show the employee the **log-in page**, he will use his username and password to **login in**.
- We believe the **first time the employee logs in**, the system will **ask him to reset the password** (since normally it is a temporary password).
- The employee logs in to the system and he can see his profile page, **he now can see all his personal information** including the attendance and the detailed components of his salary.

Skeleton Program

- Designing a program that keeps track of three different types of users in the HR system: the employer, the employee.
- Each user is given an ID, username, password, salary and attendance value that is associated with them.
- The employer should be able to add new users or update user information in the system.
- The SalaryManagement case class : Track the monthly and yearly salaries of every user in the system using their IDs.
- The UserManager case class: Access all the users in the system using their usernames and IDs.
- The SalarySystemController uses the SalaryManagement case class
- The UserSystemController uses the UserManager case class.

Teammate Contribution and Plan

Tian Shu

What I've done so far:

- CRC model and some Unit Tests

Plan for Phase 1:

- Expand CRC model
 - more necessary controllers and use cases (e.g. Fireable Interface, Intern class)
- Write Unit Test before the implementation of new classes and interfaces
 - **test-driven design**

Canyang

What I've done so far:

- I wrote some use cases, controllers and UI

Plan for Phase 1:

- Expand use cases, controllers and UI(GUI)

Shihan

What I've done so far:

- Comments
- Unit tests

Plan for Phase 1:

- Checking mistakes/Debug through writing comments and Unit tests.

Yuelin

What I've done so far:

- Scenario Walk Through, Specification
- some code, annotation

Plan for Phase 1:

- Implement more controller class to enhance user experience.
- Connect use case and controller better

Yiteng

What I've done so far:

- I wrote some use cases, controllers and UI

Plan for Phase 1:

- Expand use cases, controllers and UI(GUI)

Mihir

What I've done so far:

- Progress Report

Plan for Phase 1:

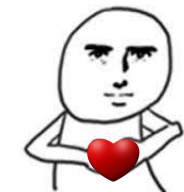
- Expand Use classes and interfaces - leave days for attendance for users

What Worked Well

- multiple entities with inheritance (User, Employee and Employer).
- Use cases that interact with the entities
- Methods in two controllers to interact with use cases.
- implemented a user-friendly interface in our UI part
 - registration
 - login
 - salary check
- Thought through new classes: Schedule Class and fireable Interface.
- possible GUI system
- unittests for methods in Entity, Use Case, and Controller

Open Questions

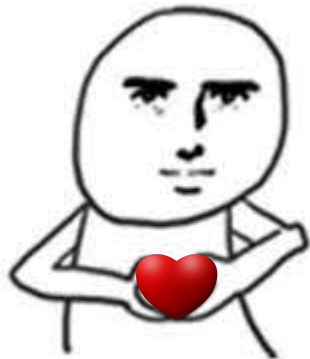
1. If UI system require user input to be type String, but an integer is entered, the program will not proceed. Then, the user need to restart the program. We need ways to make the UI more user friendly.
2. Now, every User data we entered is temporary, so we want to find a way to store the data permanently, for instance in a database. This means if we re-run the program, we can access and edit the stored old User data.



感恩的心 感谢有你

Can we get Full

Mark?



感恩的心 感谢有你





Thank You