

Progress Report

Project Details

Type of Program	HR System
Course	CSC207: Software Design
Group Members	Shillin Zang, Jennifer Tan, Ruixin Liu, Hamza Khan and Somtochukwu Oriaku

Specification summary

Our group has decided to create a HR system to help manage the most valuable part of any business, the people. The software will allow easy management of employee's salary, time sheets along with department heads. With a smaller group, the software will have limited functionality, but we aim to leave room for possible growth in the program if need be.

Running the software allows the user to interact with an empty system. Users should be allowed to create new employees, storing useful information such as name, salary, department and schedule. Each employee will be given an id upon creation.

CRC model summary

As of now, our software intends to follow the SOLID design and clean architecture principles. From the CRC cards, you will find we expect to have four entity classes; Employee class, which will be a parent class to the worker and department head along with a Schedule class that stores employees working times. These classes will be the building blocks of our program.

Our initial functionality can be seen through the 2 use case classes workerManager and departmentheadManager. The responsibilities of the classes can be seen on the CRC cards. However, we intend to possibly develop these classes with additional functionality if time permits.

Our controller/presenter class will (InOutHandler) will interact with the user through the UI and collaborate with the use case classes to produce the outputs we seek for the HR system.

Finally, the frameworks and driver include the HRSystem class. Importantly, this class will store the information created by the user from the moment the program begins till the user exits. Along with displaying welcome messages. This class will handle all the user inputs

Scenario walk-through summary

Our scenario walkthrough attempts to run through the fundamentals of our program to assess the validity of our design. We attempt to create a new worker along with a department head and assign salaries respectively. Next, we set schedules for the workers that should be validated in the schedules class. We also run through trying to change the salary of a given employee. Given our smaller group, the scenario walk-through could not be as thorough as we had hoped. More walk-throughs and testing will be required before we fully commit to the design.

When running the scenario shown in the phase 0 file, follow the specific inputs to get the desired results.

Skeleton program

The skeleton program can be found in our group repository. Detailed comments have been left in order help anyone reading this walk-through and understand the program.

Group Member Responsibilities Breakdown

Shillin Zang – Initial design/specification discussion and creating CRC cards. Working on Skeleton Program and scenario walkthrough provided. Creating Github repository for other group members to clone.

Jennifer Tan – Initial design/specification discussion and creating CRC cards. Working on Skeleton Program and scenario walkthrough provided.

Ruixin Liu – Initial design/specification discussion and creating CRC cards. Working on Skeleton Program and scenario walkthrough provided.

Hamza Khan – Initial design/specification discussion and creating CRC cards. Writing up progress report as tasks were completed.

Somtochukwu Oriaku – Initial design/specification discussion and creating CRC cards. Working on Skeleton Program and scenario walkthrough provided.

What has worked well so far

- We have implemented a structure that seems to be working well so far
- CRC cards are very detailed to allow users to quickly understand the design of our software
- Skeleton Program is completed and works well
- Good use of the SOLID design principles. Allows for easy modification of the code since we applied the SRP to our code.
- A clean architecture design allows for different group members to work on different layers of the code

Struggles

- Hard to arrange meetings with the entire group due to difference in schedules.
 - Less group members so disadvantage in scope of functionality for the software.
 - We were discussing whether the software should connect to a database to store information after program is closed.
-