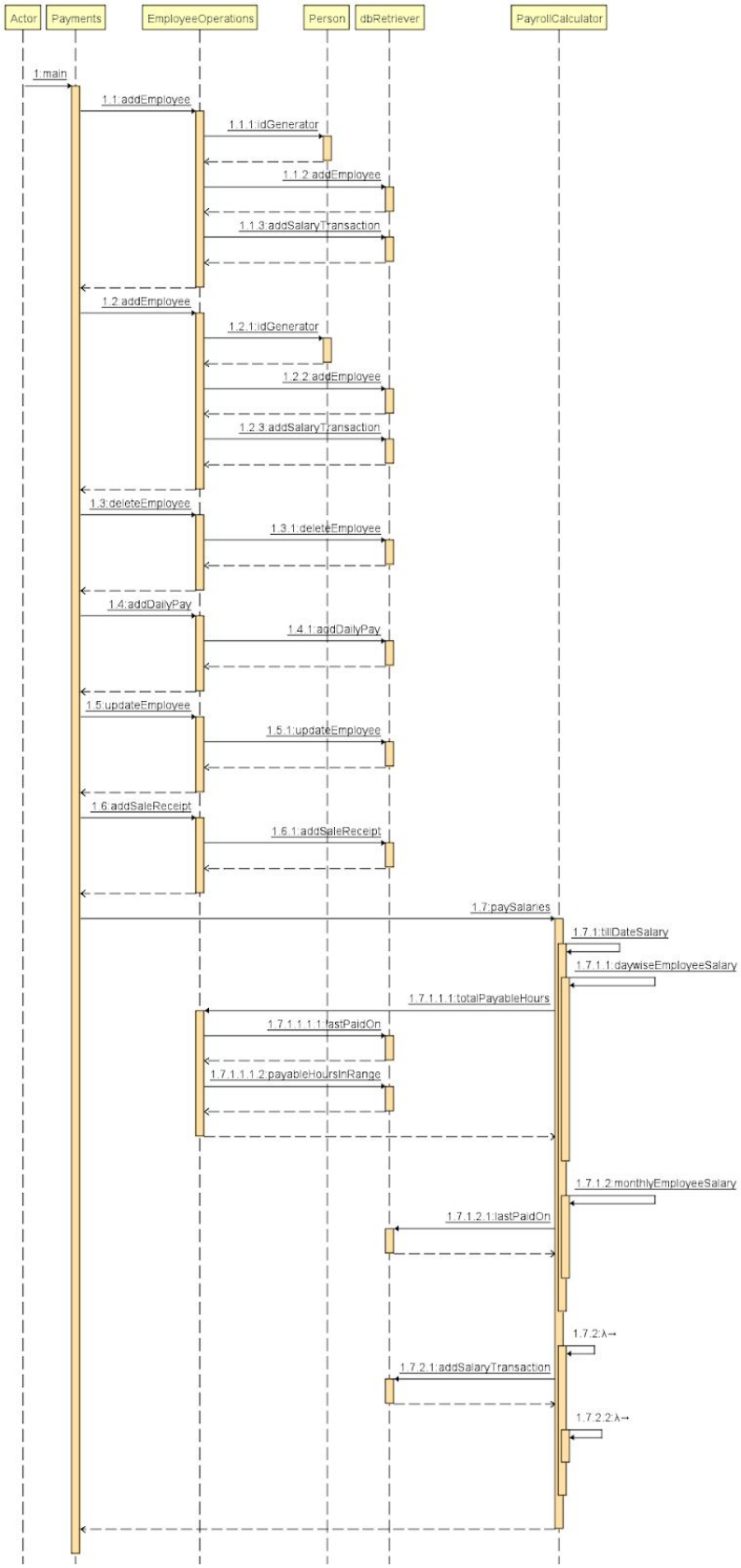
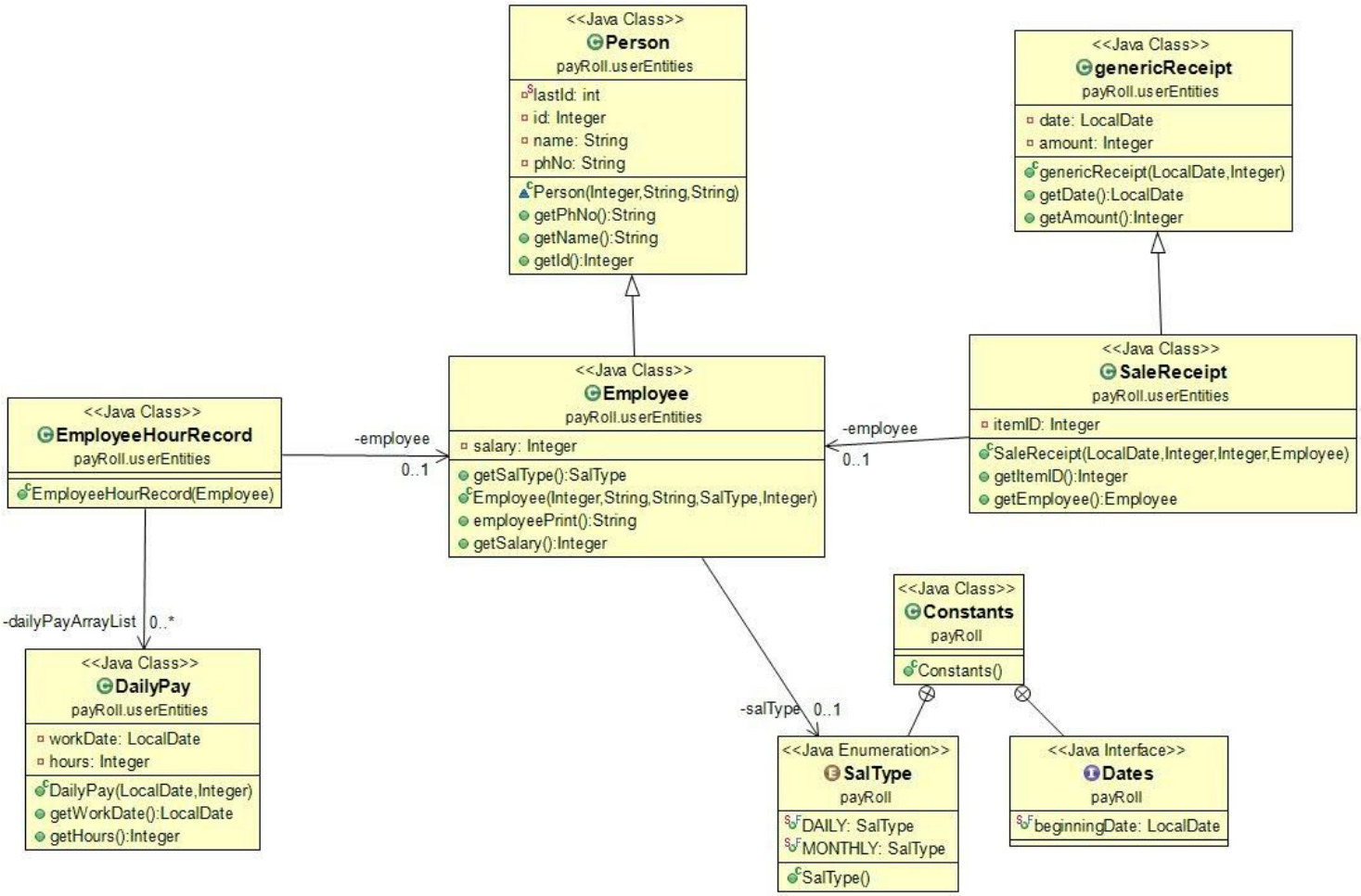
SEQUENCE DIAGRAM

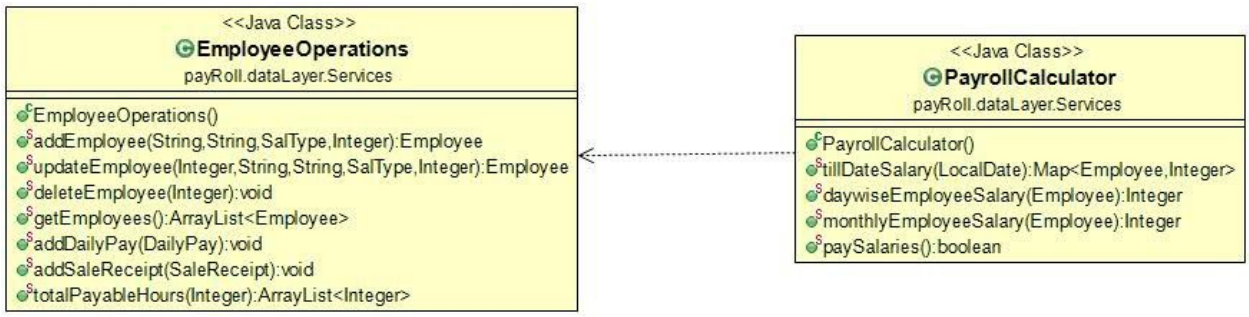


CLASS DIAGRAMS

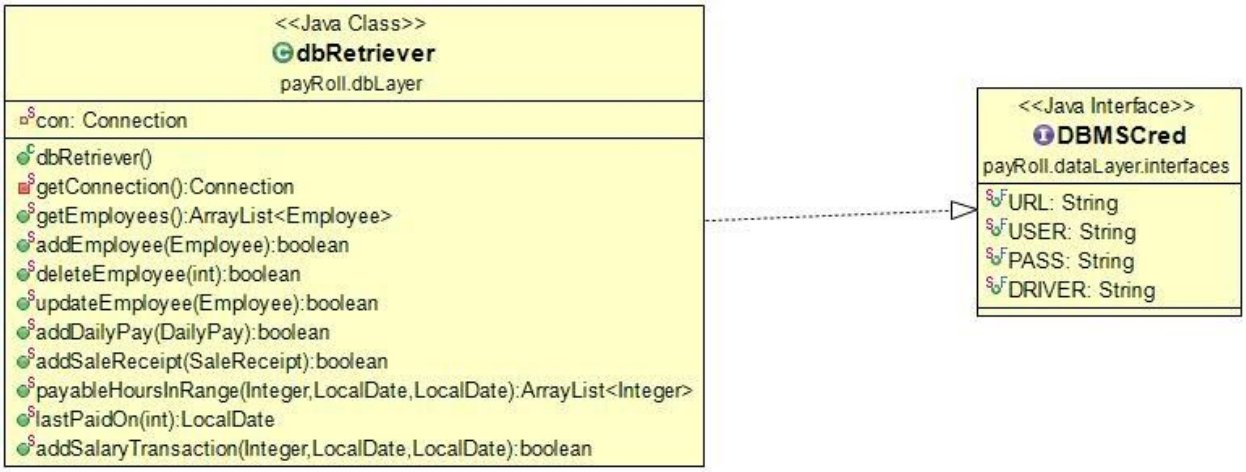
1.) Top Layer



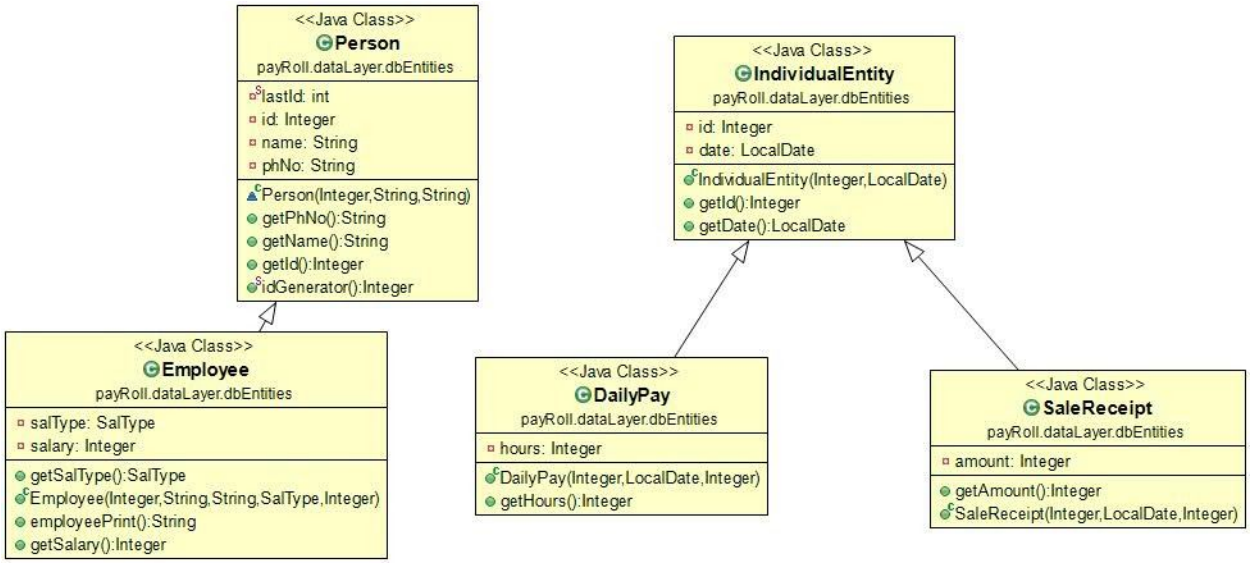
2.) Service Layer(Controllers)



3.) Database layer(Interacts with DB directly)



4.) Entity replicating database schema



Documentation

1.) Design Objectives and Approaches:

Code is designed in a layered architecture. Trying to keep all layers independent to improve extensibility and adoption.

2.) Design Choices and Your Preferences with Reasoning

Having experience with Service Oriented Architecture. I modeled my solution accordingly.

3.) Future Design Improvements

Code is written keeping extensive extensibility for future. Whether a rich UI is expected or Robust performance code can be extended and deployed.

4.) Design Challenges, Experiments and Alternative Design

The major issue I faced was that without the use of ORM(Object Relational Mapper) maintaining entities near to Database schema is a challenge. I ended up keeping two different types of data entities. One which is coherent with User perspective and other replication Database schema.