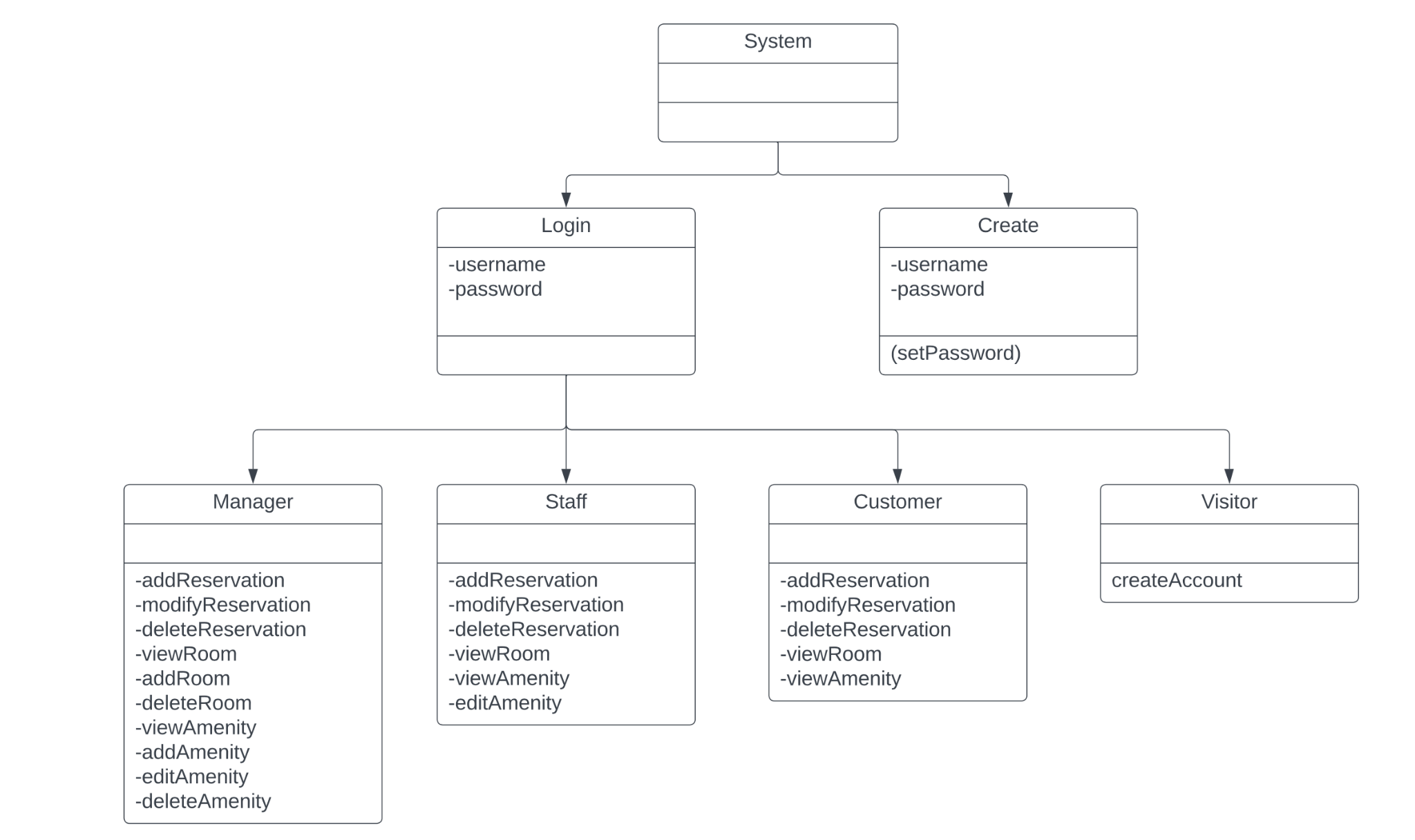
#### **System Architecture and System Design**

*Architectural Styles*

My design features client-server and service-oriented architectural styles. The design provides a service to the customers that use it. The program is also client-server because it would utilize the internet as the client.

*Identifying Subsystems*

**

*Persistent Data Storage*

The system will require data outliving a single execution of the system in order for customers to make/modify/delete their reservations. These persistent objects (customer information) will stay within a relational database (SQL).

#### **Global Control Flow**

*Execution orders:*

My system is both procedure-driven and event-driven, it just depends at what point the customer is at within the system. For example, a user will have to login every single time in order to use the system. Once logged in, it becomes event-driven, meaning that the system waits in a loop for events and every user can generate actions in different orders.

*Time dependency:*

The only time there is a time dependency in the system is when someone is making, modifying, or deleting their reservation. There would be a time dependency at this stage in order to eliminate customers from holding a room “hostage”, and so mistakes in the modify or delete stage do not occur.

*Hardware Requirements*

While the system is just a hotel reservation system, it does not need a minimum screen display. It is recommended that the resolution for the system would be 1920 x 1080. This is recommended because of the photos of the rooms and the amenities. In order to get a good look of everything, it is advised that the user would have a full HD view. A network bandwidth of 56 Kbps is recommended.