**1. Define the system of interest**

For this research, the SOI consists of three separate subsystems. The first subsystem is the Clue-meister Subsystem, which tracks the group's progress in the room using a built-in flowchart that employees update as the group makes progress. It also incorporates the timing of the room, so the employee does not have to remember it, and it can send clues automatically based on the timing and the group's location within the room. The second subsystem is a database for the entire system. It will be hosted in the cloud and feature a database that stores all the data used by the system. The third subsystem is a mobile application that will be used by employees to remotely send clues, receive clue requests, start and stop their time, draft clues, and perform other tasks. The mobile application may be written as a native app for both the iOS and Android operating systems, which will require the subsystem to be coded twice.

**System of Interest**

A white background with blue arrows

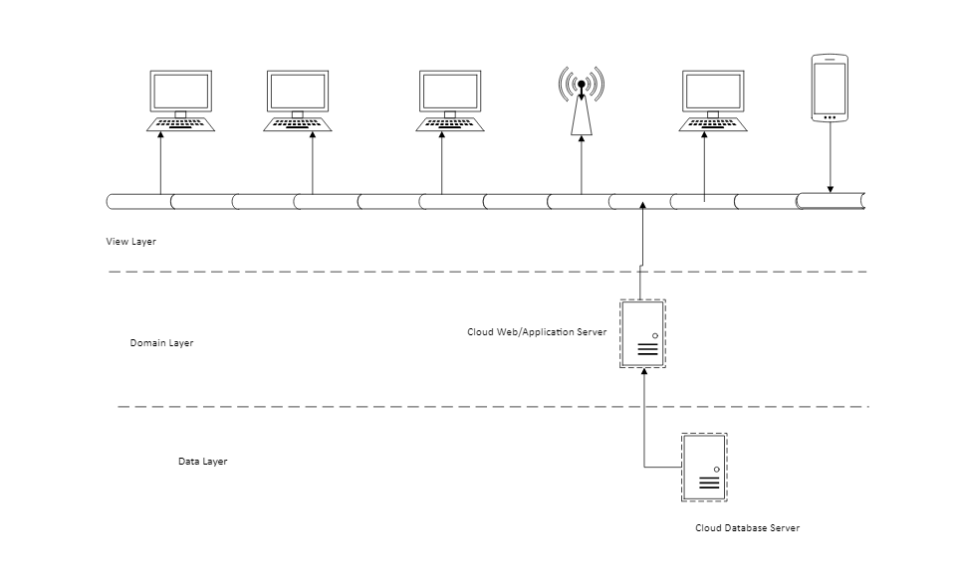
AI-generated content may be incorrect.

**2. Describe the environment**

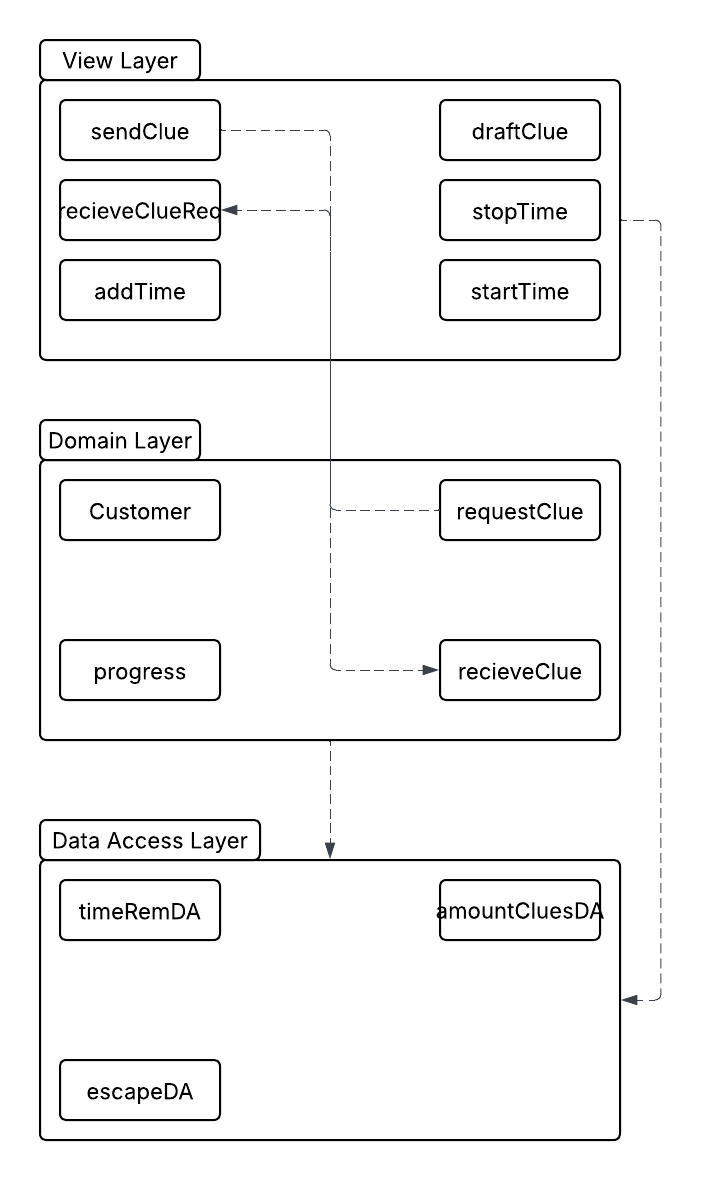
For the Clue-meister subsystem, the environment it will be used in is mainly by employees. This system is built to make running a room and clueing groups easier for the room host and management. To do that, the software will be downloaded on computers that the company already owns, with mid-level processors. Clue-meister will also project an image with a running clock and a place to display clues in the room. The monitors this will be displayed on were also previously owned by the company. The Clue-meister subsystem will interact with the main system by providing information to the mobile app through the cloud. It will also log room data (such as time remaining and number of clues used) to a database stored on the cloud.

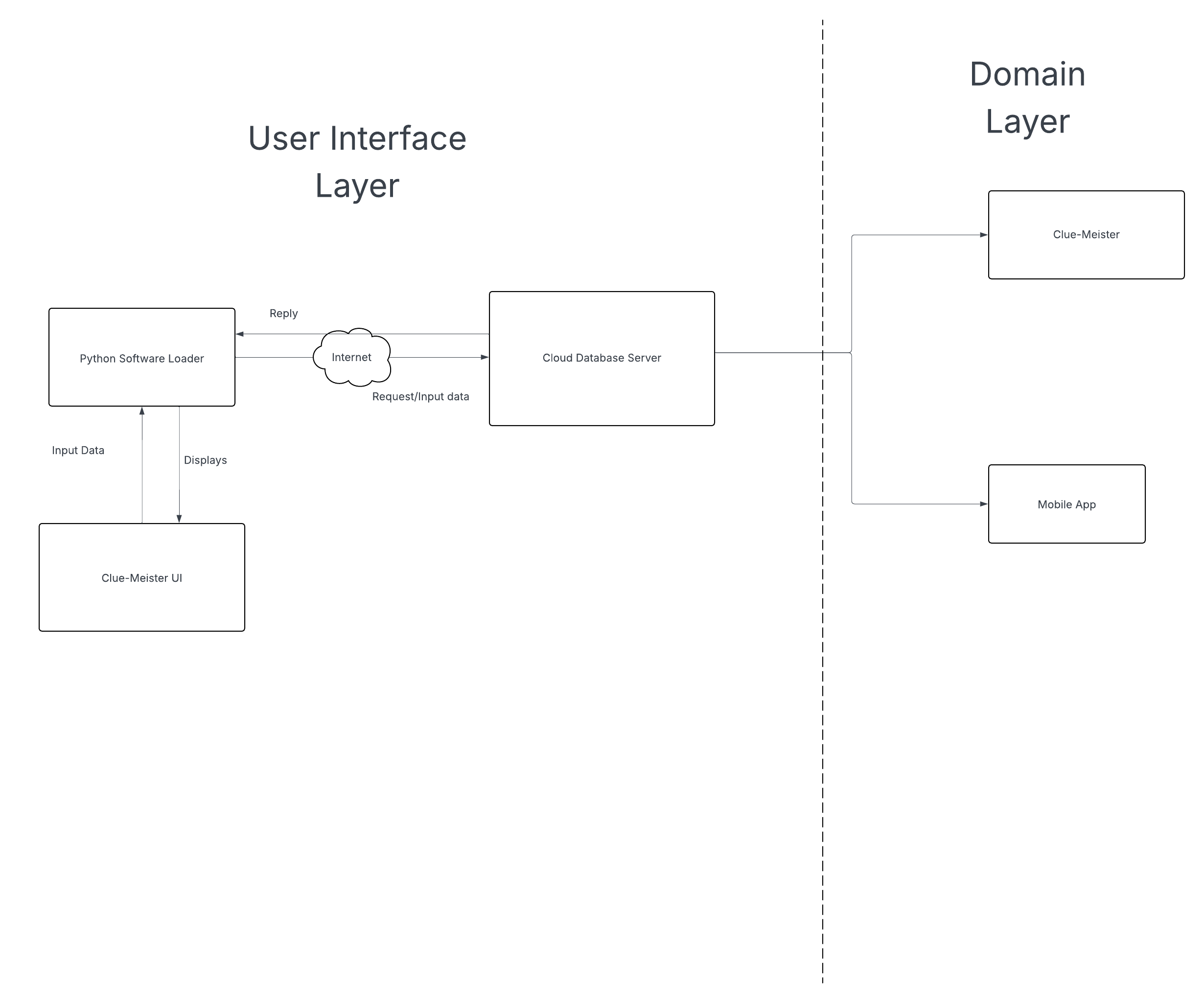
**3. Design Application Components**

The key parts of the proposed system include the clue-meister subsystem, which manages clues. The mobile application provides remote access. Finally, the database, which stores information. The system needs to be divided into 3 subsystems, as listed above.

1. **Deployment Diagram**

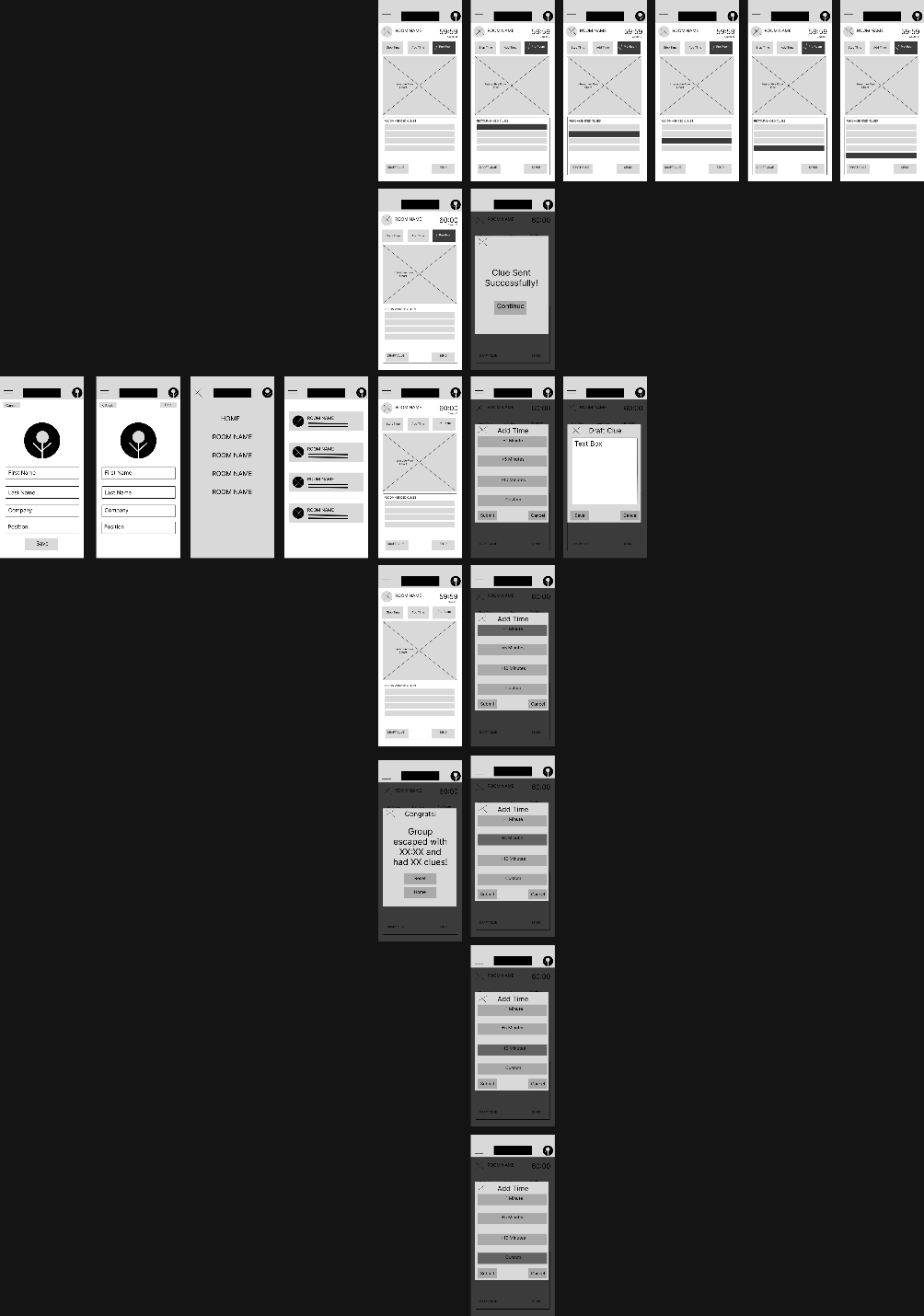
**b.) Package Diagram**



**c.) Component diagram**

**4. Low Fidelity UI Prototype**

**a.)**



**b.)**



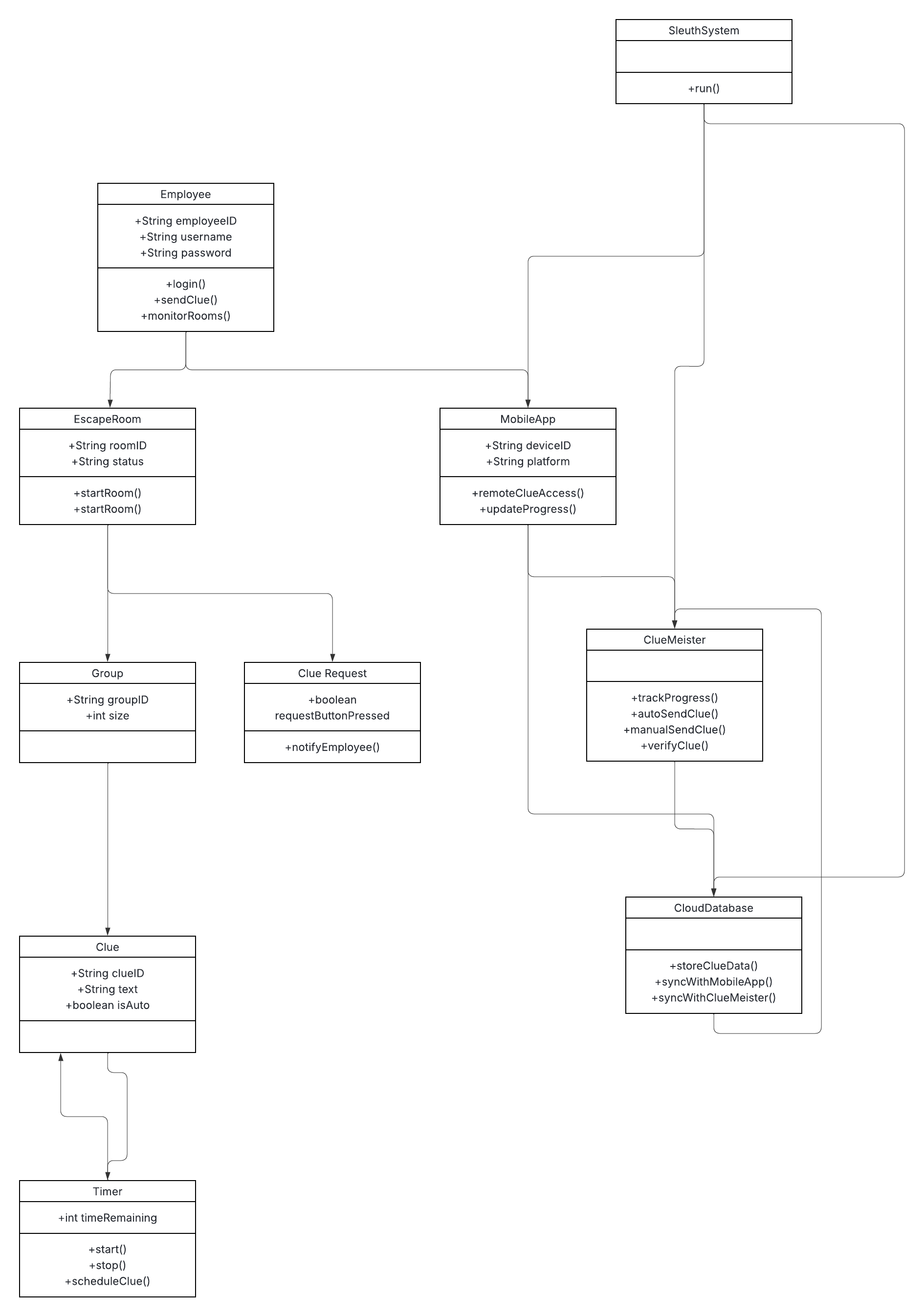
**5. Design the database**

A screenshot of a computer

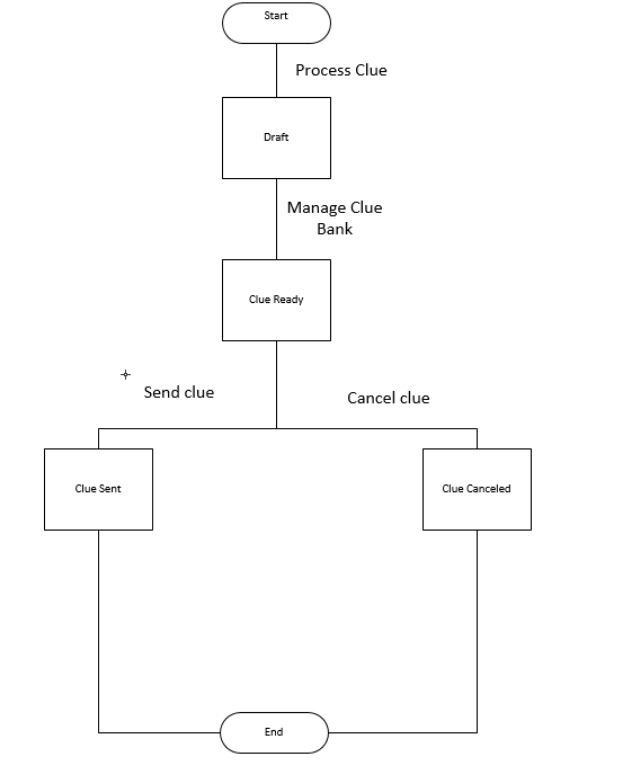
AI-generated content may be incorrect.

**6. Design the software classes and methods**

**a.) Class Model Diagram**



**b.) State Machine Diagram**



A screen shot of a login form

AI-generated content may be incorrect.**c.) System Sequence Diagram**