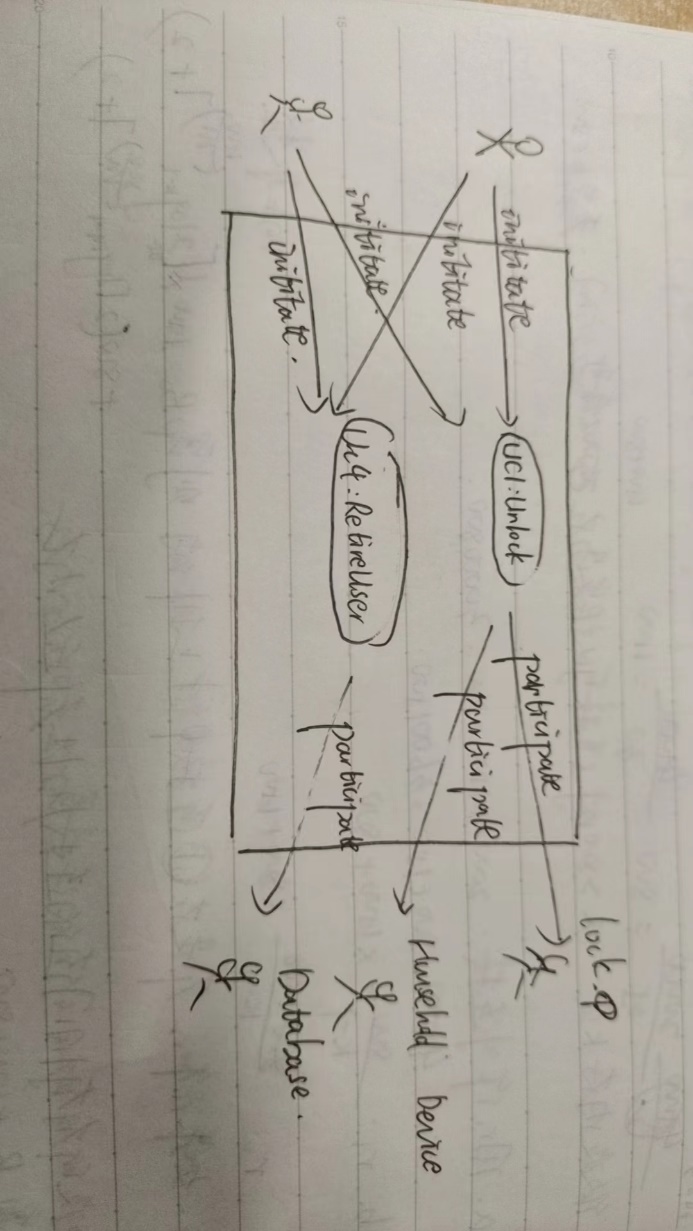
Q1：



Q2：

|  |  |
| --- | --- |
| User Case UC-1: Unlock | |
| **Related Requirem’ts:**  REQ1, REQ3, REQ4, and REQ5 stated in the table of REQs | |
| **Initiating Actor:** | Tenant, Landlord 任意一个 |
| **Actor’s Goal:** | Unlock the door and enter |
| **Participating Actors:** | LockDevice, LightSwitch, Timer |
| **Preconditions:**  (1) Doors are locked.  (2) The system displays the menu of available functions; at the door keypad the menu choices are “Lock” and “Unlock.” | |
| **Postconditions:**  (1)Door is unlocked.  (2)The auto-lock timer has started countdown from autoLockInterval. | |
| **Flow of Events for Main Success Scenario:** | |
| →: 1.**Tenant**/**Landlord** arrives at the door and selects the menu item “Unlock”  ←: 2.**System** (a) signals to the **Tenant**/**Landlord** the lock status, e.g., “disarmed,” (b) signals to **LockDevice** to disarm the lock, and (c) signals to **LightSwitch** to turn the light on  ←: 3.**System** signals to the **Timer** to start the auto-lock timer countdown  →: 4.**Tenant**/**Landlord** opens the door, enters the home [and shuts the door and locks] | |

|  |  |
| --- | --- |
| User Case UC-4: [Retire an Existing User Account](https://github.com/shawn-sheep/SE_course_project_1/blob/main/Homework.md#use-case-uc-4-retire-an-existing-user-account) | |
| **Related Requirem’ts:**  REQ-7: Manage additional user accounts. | |
| **Initiating Actor:** | Landlord |
| **Actor’s Goal:** | [To retire an existing user account and disable its access.](https://github.com/shawn-sheep/SE_course_project_1/blob/main/Homework.md#actors-goal-to-retire-an-existing-user-account-and-disable-its-access) |
| **Participating Actors:** | Database |
| **Preconditions:** | User account exists in the database. |
| **Postconditions:**  (1) User account is retired.  (2)The mobile device cannot access the lock | |
| **Flow of Events for Main Success Scenario:**  →: 1. **Landlord** select the "Manage Users" option.  ←: 2. System validates the request and checks if the user account exists.  →: 3. **Landlord** selects the action “Retire User," and delete the user that he/she want to delete.  ←: 4. System confirms the action to the Landlord. | |

Q3:

Acceptance Test Case for UC-1 **Unlock:**

**Test-case Identifier:** TC-1

**Use Case Tested:** UC-1, main success scenario

**Pass/fail Criteria:** The test passes if the Tenant successfully unlocks the door via the mobile application within the allowable attempts.  
**Input Data:** Mobile device identifier, user credentials

|  |  |
| --- | --- |
| **Test Procedure:** | **Expected Result:** |
| **Step 1. use a wrong phone and a valid door identifier** | **System beeps to indicate failure; records unsuccessful attempt in the database;** |
| **Step 2. Send multiple unsuccessful unlock attempts.** | **After a defined number of attempts, system blocks the mobile device temporarily and alerts the Tenant.** |
| **Step 3. Use an authorized mobile device with proper credentials to send unlock command.** | **System flashes a green light to indicate success; records successful access in the database; disarms the lock device** |

Acceptance Test Case for UC-4 [**Retire an Existing User Account**](https://github.com/shawn-sheep/SE_course_project_1/blob/main/Homework.md#use-case-uc-4-retire-an-existing-user-account)

**Test-case Identifier:** TC-1-2

**Use Case Tested:** UC-4, main success scenario  
**Pass/fail Criteria:** The test passes if the Landlord can retire an existing user account and disable its access, and if the retired user cannot unlock the door afterward.  
**Input Data:** User account details (e.g., username, mobile device identifier)

|  |  |
| --- | --- |
| **Test Procedure:** | **Expected Result:** |
| **Step 1. Select a non-existent user account to retire** | **System returns an error message** |
| **Step 2. Select an existing user account to retire** | **System retires the user account and confirms the action to the Landlord.** |
| **Step 3. Use the mobile device associated with the retired user to send unlock command** | **System denies access; records unauthorized attempt in the database; sends notification to Tenant or Landlord about the denied access.** |