

Use case: Validate client registration form input

Primary actor: Arm in Arm client

Goal in context: To ensure accurate and secure data entry into the Arm in Arm client registration database by validating user inputs against expected data types and formats

Preconditions: The user is entering data into the client registration form.

Trigger: The user attempts to submit the registration form

Scenario:

1. The user fills in the registration form and attempts to submit it
2. The system validates each input field against its expected data type and format
3. If an input does not match the expected format, the system prompts the user to recheck and correct their entry
4. The user corrects any flagged entries and resubmits the form
5. The system successfully validates all inputs and adds the data to the database

Exceptions:

1. User attempts to submit the form with invalid or malicious data that could lead to SQL injection
2. User provides data in the correct format but outside acceptable ranges or conditions

Priority: Essential

When available: Always

Frequency of use: Many times per day

3. **Channel to the main actor:** Web interface of the client registration form

Secondary actor(s): Arm in Arm staff

Channels to secondary actors:

1. Direct feedback and requirements meeting discussions via email

Open issues:

1. What strategies can be employed to balance strict validation requirements with user-friendliness, ensuring that the validation process does not discourage clients from completing the registration form?