Use Case Model 1: Profile Setup

<u>Primary Actor:</u> User

<u>Scope:</u> Tandem t:slim X2 Insulin Pump CRUD Personal Profile Interface (CRUD)

<u>Level:</u> User goal

<u>Stakeholders and Interests:</u>

User - wants to set account profile with corresponding information and delivery dosage

<u>Pre-condition:</u>

User was a working Tandem t:slim X2 Insulin Pump

<u>Success Guarantee:</u>

Account profile is set up with proper dosage instructions with no issues in executing dosing.

<u>Trigger:</u> User wants to create a new profile

<u>Main Success Scenario:</u>

1. User selects the Options button in the main hub.
2. CRUD displays Options menu.
3. User selects Personal Profiles in the options menu.
4. CRUD displays the list of profiles.
5. User select the Create New Profile button in the Personal Profiles hub.
6. CRUD displays an input form for profile creation.
7. User enters profile information into a new profile form.
    7.1. User enters profile name.
    7.2. User enters basal rate.
    7.3. User enters carbohydrate ratio.
    7.4. User enters correction factor.
    7.5. User enters target glucose level.
8. CRUD verifies form inputs.
9. User clicks Save button at the bottom.
10. CRUD displays new profile information as a page.

<u>Extensions:</u>

8a. CRUD detects invalid input in one or more inputs: highlights input box in red and returns to step 6.

(1, 3, 5, 7)a. User wants to go back to previous page: User can press the Back button on the top left to return a step.

Use Case Model 2: Modifying and switching personal profiles

<u>Primary Actor:</u> End User

<u>Scope:</u> Tandem t:slim X2 Insulin Pump CRUD Personal Profile Interface (CRUD)

<u>Stakeholders and Interests:</u>

End User

<u>Pre-conditions:</u>

Pump battery is charged and powered on

The user has created at least one personal profile

<u>Success guarantees:</u>

Users modifications and to personal profile is saved

<u>Trigger:</u> User wants to edit a profile

<u>Main success scenario:</u>

1. User clicks on Personal Profiles drop down menu
2. User selects existing profile
3. CRUD displays the profile information.
4. User clicks on the Edit button.
5. CRUD displays an input form containing existing information.
6. User modifies the corresponding information.
7. User clicks the Save button.
8. CRUD verifies inputs.
9. In case of invalid inputs, go back to step 4
10. CRUD displays updated profile information.

<u>Extensions:</u>

2a. User only wants to switch profile

User clicks Save button

4a. Delete a profile

User clicks Delete button

Profile is removed from the profiles

4b. Review profile

User clicks Review button

User can see profiles settings

6a. User decides to not modify profile

User selects Save Button

8a. Information is invalid

CRUD displays error message

Go to step 6

Use Case Model 3: Managing basal insulin delivery

<u>Primary Actor:</u> End User

<u>Stakeholders:</u>

End User

<u>Pre-conditions:</u>

> Pump battery is charged and powered on
>
> Personal profile is set up and selected

<u>Success guarantees:</u>

> Insulin delivery is active and continuous at specified rate

<u>Main success scenario:</u>

1. User selects a personal profile
2. User selects basal rate from active personal profile
3. User confirms selection
4. ControlIQ dynamically calculates insulin level to deliver
5. Pump begins to continuously deliver insulin at specified rate

<u>Extensions:</u>

2a. No basal rate configured or users wants to manually configure through options menu

4a. Manually stopping insulin delivery

1. User presses "stop delivery" option on the pump
2. User confirms action to stop delivery
3. Insulin pump suspends delivery

4b. Automatically stopping insulin delivery

1. Control IQ detects glucose levels below 3.9mmol/L
2. Control IQ sends a signal to the insulin pump to suspend basal insulin delivery
3. Insulin pump suspends delivery
4. The event is logged into the system history

4c. Resuming Insulin delivery

1. Glucose levels stabilize
2. User selects a personal profile
3. Previous basal rate is restored

Use Case Model 4: Access Insulin Pump History

Primary Actor: End User

Stakeholders and Interests: System (t:slim X2 Insulin Pump Simulation)

Preconditions:

The user has an active profile.

The insulin pump has logged history data (such as past bolus deliveries, basal rates, alerts, CGM readings).

Success Guarantees:

The user successfully views the history

No modifications to past records are allowed

Main Success Scenario:

1. User navigates to the History section from the GUI.
2. System retrieves the stored history data from the database.
3. System displays:
   - Bolus injections (time, dosage, type)
   - Basal insulin changes
   - Alerts and warnings (low insulin, occlusion, low battery)
   - CGM readings over time
   - User profile changes (if logged)
4. User can filter data based on date/time or type (e.g., only bolus history).
5. User exits the history screen.

Extensions:

1a. No History Data Available

The system displays a message: "No historical data is available."

User can return to the main menu.

Use Case Model 5: Managing t:slim X2 Insulin Pump

<u>Primary Actor:</u> End User

<u>Stakeholders:</u> End User

<u>Pre-conditions:</u>

> Pump battery is charged and powered on
>
> The application must be in the home menu

<u>Success guarantees:</u>

> Battery life is sufficient or charging

<u>Main success scenario:</u>

1. The user holds the power button until the startup sequence completes
2. The user enters the PIN
3. Battery level is displayed showing capacity in top left corner
4. Insulin left in cartridge is displayed in top right corner
5. The pump brings the user to the home menu
6. Displays options and bolus buttons
7. Displays CGM data and insulin on board from bolus injections

<u>Extensions:</u>

1a. Pump may have charger plugged in

1b. User turns off pump

> User navigates to options screen
>
> User clicks "Turn off pump" button
>
> The t:slim X2 pump locks

1c. User powers off pump

> User holds power button until the pump turns off

2a. User incorrectly enters the PIN

> User is locked out of the insulin pump

3a. Battery level is low

> Warning prompt displays low battery
>
> User understands that the battery is lower.

3b. Battery dies

> The system shuts off with an empty battery sign.
>
> User plugs in pump with USB cable to charge battery

5a. User leaves home screen

> Tandem logo shows on every page in the t:slim pump
>
> User clicks the Tandem logo
>
> User returns to home screen

Use Case Model 6: Delivering a Manual Bolus

<u>Primary Actor:</u> End User

<u>Stakeholders:</u>

> End User

<u>Pre-conditions:</u>

> Pump is powered on with sufficient battery and unlocked
>
> Personal profile with bolus settings is configured
>
> Insulin cartridge is loaded

<u>Success Guarantees:</u>

> Correct bolus dose is calculated and delivered.
>
> Delivery can be adjusted, paused, or canceled as needed.

<u>Main Success Scenario:</u>

1. User initiates bolus

   > User presses the bolus button from the home screen

2. User inputs data

   > Enters glucose in grams
   >
   > Enters carbohydrate intake

3. Bolus calculator suggests dose

   > ControlIQ calculates recommended insulin based on active profile and CGM data

4. User confirms dose

5. Pump delivers insulin via standard bolus

6. Completion

   > Pump logs delivery in history.

<u>Extensions:</u>

3a. No CGM data available:

> User manually enters blood glucose value.

3b. User manually overrides recommendation

> User manually adjusts the dose of insulin

4a. User cancels bolus mid-delivery:

> Press the Stop button.
>
> Confirms cancellation.
>
> Pump stops delivery and logs partial dose.

4b. Control-IQ intervenes:

> Detects predicted hypoglycemia during delivery.
>
> Automatically suspends remaining bolus.
>
> Logs event in history.

5a. Extended bolus selected:

> User sets the duration and % of insulin to deliver upfront.
>
> Pump delivers remainder incrementally over time.

5b. Quick bolus for high glucose correction:

> User bypasses the calculator and manually inputs the correction dose