

Requirements Traceability Matrix for Insulin Pump System

ID	Requirement	Related Use Case	Fulfilled By (Design & Implementation)	Test	Description
1	Pump must support initial setup and configuration	UC-1: Initial Pump Setup	MainWindow, MainWindow.ui, WarningChecker	Run application, verify pump powers on and initializes properly.	Ensures the pump initializes correctly with battery and insulin indicators.
2	System must support configurable date/time settings	UC-1: Initial Pump Setup	MainWindow, Qt's QDateTime	Configure date/time and confirm accuracy.	Validates accurate date/time handling after setup.
3	System must support measurement unit configuration (mmol/L)	UC-1: Initial Pump Setup	UserProfile, UserProfileManager	Configure measurement units and verify correctness.	Ensures the consistent use of mmol/L for BG measurement.
4	System must support Control IQ Technology enablement	UC-1: Initial Pump Setup	PumpController, UserProfileManager	Toggle Control IQ feature on/off and verify function.	Confirms Control IQ functionality can be activated or disabled.
5	System must provide battery level indication	UC-1: Initial Pump Setup	WarningChecker, MainWindow	Simulate battery usage, verify accuracy of battery indicator.	Ensures correct real-time display of battery level.
6	System must support CRUD operations for personal profiles	UC-2: Manage Personal Profiles	ProfileDialog, UserProfileManager	Perform CRUD operations on personal profiles, verify data consistency.	Validates profile creation, retrieval, updating, and deletion.
7	Profiles must store basal rates, carb ratios, correction factors, target BG	UC-2: Manage Personal Profiles	UserProfile, UserProfileManager	Enter profile parameters, verify correct storage.	Ensures essential insulin delivery parameters are correctly stored.
8	System must validate parameter ranges	UC-2: Manage Personal Profiles	ProfileDialog, Qt Input Validation (QInputDialog)	Enter invalid parameters, verify rejection.	Ensures only valid numerical inputs are accepted for profiles.

9	System must support manual bolus delivery	UC-3: Deliver Manual Bolus	BolusDeliveryWidget, PumpController	Enter bolus inputs, verify bolus delivered accurately.	Confirms the manual bolus is calculated and delivered correctly.
10	System must support extended bolus functionality	UC-3: Deliver Manual Bolus	BolusDeliveryWidget, PumpController	Enter extended bolus parameters, verify timed delivery.	Validates insulin delivery over an extended specified duration.
11	System must calculate suggested bolus based on carbs and BG	UC-3: Deliver Manual Bolus	BolusDeliveryWidget, UserProfile	Enter BG and carb values, verify suggested bolus accuracy.	Ensures correct insulin dose calculation based on input parameters.
12	System must integrate with CGM for BG auto-population	UC-3, UC-4	CgmSimulator, BolusDeliveryWidget	Simulate CGM data, verify correct import into bolus calculator.	Ensures CGM data auto-populates BG input for bolus calculation.
13	System must warn of potential hypoglycemia risk	UC-3: Deliver Manual Bolus	BolusSafetyManager, WarningChecker	Simulate low BG scenarios, verify warnings appear.	Ensures system warns user if BG inputs suggest hypoglycemia risk.
14	System must support basal insulin start/stop/resumption	UC-4: Start/Stop/Resume Basal	PumpController, HistoryManager	Simulate basal insulin control, verify start/stop/resume functionality.	Ensures manual control of basal insulin delivery functions correctly.
15	System must support Control IQ automatic adjustments	UC-4: Start/Stop/Resume Basal	PumpController, CgmSimulator, HistoryManager	Simulate BG predictions, verify automatic insulin adjustments.	Validates automatic insulin delivery changes by Control IQ logic.
16	System must visually indicate current basal status	UC-4: Start/Stop/Resume Basal	MainWindow, Status Icons	Observe icons during basal insulin status changes.	Ensures visual indication correctly reflects basal insulin status.
17	System must track and display delivery history	UC-5: View Pump History	HistoryManager, HistoryDialog, HistoryRecord	Perform actions, verify accuracy of history records.	Validates comprehensive tracking and logging of insulin and BG events.

18	System must provide data visualization	UC-5: View Pump History	CGMGraphWidget, CgmSimulator	Simulate CGM data, verify graphical accuracy.	Confirms BG levels accurately visualized in real-time graphs.
19	System must detect and alert for malfunctions	UC-6: Handle Pump Malfunctions	WarningChecker, HistoryManager, AlertDialog	Trigger malfunctions (e.g., low insulin), verify alert appearance.	Ensures the system detects errors and promptly notifies users.
20	System must suspend insulin delivery for critical errors	UC-6: Handle Pump Malfunctions	PumpController, WarningChecker, HistoryManager	Simulate critical errors, verify insulin delivery suspension.	Ensures insulin delivery safely stops during critical malfunction conditions.
21	System must log all errors for troubleshooting	UC-6: Handle Pump Malfunctions	HistoryManager, HistoryRecord, WarningChecker	Generate errors, verify complete and accurate logs.	Confirms detailed error logging for effective troubleshooting.