ID	Requirements	Related Use Case	Fulfilled By	Tested By	Implemented By
1	Ensure pump is powered on when battery level is either not charging nor minimum (low battery)	Use Case 1: t:slim X2 setup	PowerStateMachine::powerOn(), StatusModel::getBatteryLevel()	Change battery to be 2% instead of default 100% and check if it works.	powerstatemachine, statusmodel, Battery
2	Display lock screen and require correct 4-digit PIN to unlock insulin pump	Use Case 1: t:slim X2 setup	LockScreen::on_confirmButton_clicked(), AuthManager::validatePin()	Enter valid and invalid PIN to make sure it only unlocks for correct pin otherwise get error message for wrong PIN	Lockscreen2, authmanager
3	Allow users to update the stored PIN from the settings tab from contentwidget	Use Case 1: t:slim X2 setup	ContentWidget::on_setting_pin_update_button_clicked(), AuthManager::setPinCode()	Change PIN and press update, restart application, test if it works	contentwidget, authmanager
4	Create new insulin profile with name, basal rate, carb ratio, correction factor, and glucose targets	Use Case 2: Profile Creation and management	ContentWidget::createProfile(), Pump::createProfile()	Create profile and check if it exists in settings tab in contentwidget and check if values given are stored are correct	contentwidget, Pump, Profile
5	Delete selected insulin profile and remove from it	Use Case 2: Profile Creation and management	ContentWidget::deleteProfile(), Pump::removeProfile(), Profile::deleteProfile()	Delete and ensure it doesn't exist from profile list in contentwidget	contentwidget, Pump
6	Edit insulin profile fields	Use Case 2: Profile	ContentWidget::editProfile(), Pump::updateProfile()	Update settings from contentwidget	contentwidget, Pump, Profile

			using "edit profile" then check for changes when selecting the profile	
7	Select an existing profile from ones created to use it for CGM and ControlIQ	Use Case 2: Profile Creation and management	ContentWidget::selectProfile(), Pump::selectActiveProfile()	select profile and verify values update and CGM is updated accordingly	contentwidget, Pump
8	Visualize the most recent glucose readings using the graph	Use Case 5: Reviewing Insulin Delivery History Primary	ContentWidget::loadGraphData(), setupBloodSugarGraph()	Simulate readings then check if the values match glucose reading using y axis	contentwidget, QCustomPlot, Profile
9	Display insulin and glucose event history	Use Case 5: Reviewing Insulin Delivery History Primary	ContentWidget::updateHistoryTab(), PumpHistory::get*()	View history tab and validate event sorting and formatting	contentwidget, PumpHistory, Profile
10	Adjust basal rate through the Options window and apply to current profile selected	Use Case 4: Starting, Stopping, or Resuming Insulin	OptionsWindow::on_acceptButton_clicked(), Pump::updateProfile()	Change basal rate press accept update in settings and event log	optionswindow, Pump, Profile
11	Trigger an alert notifying the user when the basal rate is changed	Use Case 6: Insulin pump malfunction	Pump::recordAlert("Basal rate successfully updated to", <rate>), ContentWidget::displayAlert()	Check for alert in contentwidget and look if an alert popped up for change in basal rate	optionswindow, Pump, PumpHistory, contentwidget
12	Automatically deliver basal insulin from profile and	Use Case 4: Starting, Stopping, or	CGM::update(), Pump::recordBasalRateChange(),	Let simulation run for few seconds and see if CGM	CGM, Pump, InsulinCartridge

#					
	decrease glucose accordingly	Resuming Insulin	InsulinCartridge::setRemainingInsulin()	drops along with remaining insulin cartridge is showing lower remaining units	
13	Auto correcting bolus if glucose exceeds safe range, using ControlIQ	Use Case 4: Starting, Stopping, or Resuming Insulin	ControlIQ::adjustInsulinDelivery(), Pump::recordBolus()	Set high glucose and verify correction bolus is recorded and delivered	ControlIQ, Pump, CGM
14	Suspend basal insulin if CGM reads glucose below 3.9 mmol/L	Use Case 4: Starting, Stopping, or Resuming Insulin	ControlIQ::adjustInsulinDelivery(), Pump::basalDeliveryStopped()	Simulate low glucose and confirm suspension event in the log	ControlIQ, Pump, CGM
15	Alert user if insulin cartridge is empty and suspend delivery	Use Case 6: Insulin pump malfunction	Pump::recordAlert(), ControlIQ::adjustInsulinDelivery(), InsulinCartridge	Drain insulin to 0 units and check if alert is given out if its empty and delivery is suspended	ControlIQ, Pump, InsulinCartridge, contentwidget
16	Allow user to receive a bolus manually using entered glucose and carb data	Use Case 3: Manual Bolus	BolusCalculator::total_bolus(), Pump::recordBolus()	Enter carb/glucose values and confirm calculated dose is correct	contentwidget, BolusCalculator, Pump
17	Estimate carbohydrates based on CGM data and glucose rise	Use Case 3: Manual Bolus	CGM::estimateCarbs(), Profile::getGlucoseReadings()	Run simulation and observe the automatic estimation	CGM, Profile
18	Simulate battery draining and	Use Case 1: t:slim X2 setup	Battery::drainBattery(), Battery::setCharging(),	Let battery drain to 0 without	Battery, statusmodel, powerstatemachine

			PowerStateMachine::handleBatteryChange()	charging and verify it powers off and charging behavior by pressing the button to charge make sure its increasing until it reaches 100.	
	charging behavior				
19	Trigger an alert when battery level drops below 20% for battery and also for 10% that battery is critically low.	Use Case 6: Insulin pump malfunction	PowerStateMachine::handleBatteryChange(), Pump::recordAlert()	Simulate battery drop below 20% and see if alert pops up in alert section in contentwidget and do same thing with below 10%	powerstatemachine, Pump, Battery, contentwidget