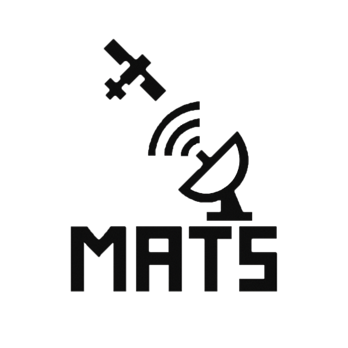
User Interface Subsystem Verification Test Plan

Mobile Antenna Tracking System (MATS)

9/26/2025

Zachary Murray, Dylan Vilcek, Ma Minghao



This document outlines the test plan for the User Interface. The document guides the reader through the system overview, requires, and testing procedures required to validate the User Interface to ensure that it operates well within the MATS system requirements.

Table of Contents

[Subsystem Overview 3](#_Toc209766761)

[Subsystem Requirements and Specifications 3](#_Toc209766762)

[Objectives 3](#_Toc209766763)

[Required Equipment 3](#_Toc209766764)

[Testing Procedure 4](#_Toc209766765)

[Subsystem Test Results 4](#_Toc209766766)

## Subsystem Overview

The User Interface subsystem provides the primary operator touchpoint within the MATS. It delivers feedback through a 7” touchscreen, communicates system status via LED indicators, and provides removable media access through USB and SD ports. The UI runs a lightweight Raspberry Pi OS desktop, auto-launching SatDump at startup. Status LED’s are driven using a Python daemon that subscribes to SatDump telemetry, providing live feedback for RF lock, recording, and power state.

## Subsystem Requirements and Specifications

Table 1: User Interface Subsystem Specifications

|  |  |
| --- | --- |
| Category | Requirement |
| Display | 7 Inch capacitive touchscreen with Linux driver support. |
| Inputs/outputs | USB-A user port, SD/microSD card reader, power switch, RF & Power status LEDs. |
| Environmental | Operating 0ºC-50ºC, front panel IP-54 splash resistance |
| Electrical | Single 5V DC rail, 2A steady-state draw |
| Software | I2C touch controller, GPIO LEDs |
| Mechanical | Front-panel mounting to aluminum chassis, maximum depth behind panel mm |

## Objectives

The testing objectives for the User Inteface are:

* Verify touchscreen responsiveness and accuracy
* Confirm automatic startup and SatDump launch.
* Validate USB and SD card hot-plug enumeration
* Verify LED indicators function under scripted test
* Confirm LEDs respond correctly to SatDump Telemetry.

These objectives ensure the User Interface provides reliable control and feedback to operators.

## Required Equipment

* UI Hardware
* Test Media: 32 GB microSD card and 128 GB USB Flash drive
* Multimeter
* Python I2C test script (led\_test.py

## Testing Procedure

Table 2: User Interface Testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Test** | **Method** | **Expected Result** | **Observed Result** |
| Display Touch Response | Run 10-point accuracy grid or calibration app | Touch deviation < 2 mm |  |
| SatDump Auto-Launch | Boot system and observe startup | Pi logs into desktop and SatDump auto-launches within ~15 s |  |
| USB Enumeration | Hot-plug 128 GB USB drive | Drive mounts automatically in <3 s |  |
| SD Card Enumeration | Hot-plug 32 GB microSD card via panel reader | Card mounts automatically in <3 s |  |
| LED Functionality | Run led\_test.py scripted I²C pattern test | Each LED (Power, RF, Recording) cycles correctly through colors |  |
| LED Telemetry Response | Start/stop SatDump recording session | RF/Recording LEDs toggle in real time with telemetry |  |

## Subsystem Test Results

|  |  |  |
| --- | --- | --- |
| **Test Section** | **Pass/Fail** | **Notes** |
| Touchscreen Accuracy |  |  |
| Auto-Launch of SatDump |  |  |
| USB Enumeration |  |  |
| SD Card Enumeration |  |  |
| LED Functionality |  |  |
| LED Telemetry Response |  |  |