Charter School Testing Support Playbook

# 📘 Purpose

To ensure seamless technology support during school-wide assessments, reduce testing disruptions, and improve test data validity through structured processes and communication.

# 🔧 1. Device Readiness Checks (Pre-Testing)

Timeline: Begin 2–4 weeks before testing
Goal: Ensure all student and proctor devices meet testing platform requirements

## ✅ Daily Checklist (Start 1 week prior to test)

• Devices are fully charged or charging stations are available
• Operating systems are updated (Chromebooks, Windows, macOS, iPads)
• Approved testing apps or secure browsers installed (e.g., LockDown Browser, TestNav, AIRSecureTest)
• Disable automatic updates during testing windows
• Caching servers tested (if applicable)
• Keyboard/mouse functionality verified
• Network connectivity test per room/device
• Disable pop-ups, alerts, and screen savers
• Check firewall/content filter doesn’t block testing URLs

Documentation:
Maintain a device audit log and testing tech checklist (Google Sheet or ticketing system)

# 📣 2. Escalation Protocols During Testing

Timeline: Real-time during testing days
Goal: Minimize downtime and respond quickly to disruptions

## 🚨 Tiered Support Workflow

|  |  |  |
| --- | --- | --- |
| Issue Type | Who to Notify First | Action Taken |
| Device won’t boot | Classroom Proctor | Swap device, notify onsite tech support |
| Secure browser crash | Classroom Proctor → Tech Lead | Restart app, verify internet, clear cache |
| Network outage | Tech Lead → MSP/IT Manager | Check local access points, escalate to ISP |
| Login/auth errors | Testing Coordinator → MSP | Reset credentials, check SSO server |
| Repeated disconnections | MSP/Network Engineer | Review firewall logs, student activity logs |

## 🧽 Escalation Flow

1. Proctor logs the issue via form or hotline
2. Site IT Lead triages within 5 min
3. MSP Tier 2/3 engaged for deeper tech problems
4. Testing Coordinator notifies principal/school leader of critical incidents

Communication Channels:
• Shared Slack/MS Teams war-room channel
• Emergency SMS/phone tree
• Ticketing platform (Zendesk, Freshservice, etc.)

# 📋 3. Post-Test Review & Incident Debrief

Timeline: Within 3–5 days after testing ends
Goal: Identify root causes, improve preparedness, and document lessons learned

## 🔍 Post-Test Technical Review

• Review logs from testing platform (e.g., test disruptions, restarts)
• Network uptime and device usage analysis
• Root cause analysis of escalated tickets
• Identify chronic device or connectivity issues

## 🧑‍🏫 Proctor & Staff Feedback

Distribute a brief survey (Google Form) to:
• Capture ease of login
• Report recurring issues
• Suggest improvements

## 🛠️ Recommendations Report

• Create a report that includes:

 - Number and type of incidents
 - Downtime minutes per testing session
 - Recommendations for hardware refresh, training, or Wi-Fi expansion
• Schedule a post-mortem meeting with IT team, MSP, and school leadership

# 🛡️ Supporting Tools & Templates

|  |  |
| --- | --- |
| Resource | Purpose |
| Device Readiness Checklist | Pre-test tech prep per room |
| Incident Log Template | Capture all issues with timestamp/resolution |
| Testing Day Tech Hotline Sheet | Contact list with escalation order |
| Post-Test Debrief Form | Collect insights from staff and tech teams |
| Testing War Room Communication Plan | Centralize real-time response |

# 🧐 Best Practices

• Dry runs: Conduct a mock testing session a week before.
• Test device rotation: Ensure spares are available per class.
• Tag & monitor devices used during testing for audit trail.
• Cloud dashboard monitoring: Use tools like GoGuardian, Meraki, or Aruba Central to monitor usage in real time.