IT Help Desk and Technical Support

Troubleshooting Methodology for Beginners

WITH REAL EXAMPLES!

© Dan Mill Training

Table of Contents

Introduction to Troubleshooting

Welcome to the world of IT troubleshooting! As a help desk technician, you'll encounter problems ranging from simple password resets to complex network issues. The key to success isn't knowing every solution immediately - it's following a **systematic approach** that will guide you to the answer.

Why Follow a Methodology?

- **Consistency** - Same approach every time
- **Efficiency** - No wasted time or effort
- **Documentation** - Track what you've tried
- **Professional Growth** - Build expertise systematically
- **Customer Confidence** - Appear organized and competent

The 6-Step Troubleshooting Process

Every IT professional should master this fundamental process:

1. **IDENTIFY** the Problem

- Gather information from the user
- Define the scope and impact
- Determine urgency and priority

2. **RESEARCH** Potential Causes

- Check knowledge base and documentation
- Consider recent changes
- Identify possible root causes

3. **DEVELOP** Action Plan

- List potential solutions
- Prioritize from most likely to least likely
- Plan the order of testing

4. **IMPLEMENT** the Solution

- Try solutions one at a time
- Document each attempt
- Test thoroughly after each change

5. **VERIFY** the Fix

- Confirm the problem is resolved
- Test related functionality
- Get user confirmation

6. **DOCUMENT** Everything

- Record the problem and solution
- Update knowledge base
- Note lessons learned

Step-by-Step Breakdown

Step 1: IDENTIFY the Problem

What to Ask the User:

- **"What exactly is happening?"**
- **"When did this start?"**
- **"What were you trying to do when it happened?"**
- **"Has this happened before?"**
- **"What error messages do you see?"**

Information to Gather:

- User details (name, department, contact)
- Computer/device information
- Software being used
- Exact error messages (screenshots help!)
- Impact on work/business

Example Questions in Action:

> User: "My computer is broken!"
> You: "I understand that's frustrating. Can you tell me what specifically is happening when you try to use your computer?"
> User: "It won't turn on."
> You: "When you press the power button, what happens? Do you see any lights, hear any sounds, or see anything on the screen?"

Step 2: RESEARCH Potential Causes

Where to Look:

- **Company knowledge base**
- **Manufacturer documentation**
- **Recent change logs**
- **Known issues database**
- **Team chat/email for similar recent issues**

Questions to Consider:

- Has this happened to other users recently?
- Were there any recent updates or changes?
- Is this affecting multiple systems?
- What are the most common causes of this symptom?

Step 3: DEVELOP Action Plan

Create Your Solution List:

1. **Quick wins** (easy, low-risk solutions)
2. **Most likely causes** (based on research)
3. **Progressive complexity** (simple to complex)
4. **Last resorts** (reinstalls, replacements)

Example Action Plan:

```
Problem: Computer won't turn on
Action Plan:
1. Check power cable connection
2. Try different power outlet
3. Check power button functionality
4. Test with different power cable
5. Check internal power connections
6. Test power supply unit
7. Replace power supply if needed
```

Step 4: IMPLEMENT the Solution

Best Practices:

- **Try ONE solution at a time**
- **Document what you do**
- **Take notes on results**
- **Don't skip steps**
- **Ask for help if stuck**

Implementation Example:

```
Action: Check power cable connection
Result: Cable appears secure, no change
Documentation: "Verified power cable firmly connected to both
                computer and wall outlet - issue persists"

Action: Try different power outlet
Result: Still no power
Documentation: "Tested known-good outlet, no change in behavior"
```

Step 5: VERIFY the Fix

Verification Checklist:

- [ ] Original problem is completely resolved
- [ ] No new problems were created
- [ ] All related functions work normally
- [ ] User can perform their normal tasks
- [ ] Solution is stable (not just temporary)

User Involvement:

You: "I've resolved the power issue with your computer. Can you please log in and test opening your email and the document you were working on yesterday?"
User: "Yes, everything seems to be working normally now."
You: "Perfect! I'll check back with you in 30 minutes to make sure everything is still working well."

Step 6: DOCUMENT Everything

What to Document:

- **Problem description**
- **Steps taken**
- **Solution that worked**
- **Time spent**
- **Follow-up required**

Documentation Template:

```
Ticket #: 12345
User: John Smith (Accounting)
Issue: Computer won't power on
Symptoms: No lights, no fan noise, no display when power button pressed
Actions Taken:
1. Checked power connections - OK
2. Tested different outlet - No change
3. Replaced power cable - No change
4. Opened case, reseated power connections - RESOLVED
Solution: Loose internal power connector to motherboard
Time: 45 minutes
Follow-up: None required
```

Essential Checklists

Initial Problem Assessment Checklist

- [ ] Obtained user's contact information
- [ ] Documented exact problem description
- [ ] Identified when problem started
- [ ] Determined what user was doing when it occurred
- [ ] Captured any error messages (screenshots if possible)
- [ ] Assessed impact on user's work

- [ ] Determined if other users are affected
- [ ] Checked for recent changes or updates
- [ ] Assigned appropriate priority level
- [ ] Set user expectations for resolution time

Before Implementing Solutions Checklist

- [ ] Created action plan with prioritized solutions
- [ ] Researched potential causes
- [ ] Checked knowledge base for similar issues
- [ ] Confirmed user has saved all important work
- [ ] Noted current system state
- [ ] Prepared to document all actions
- [ ] Identified escalation path if needed
- [ ] Estimated time required for resolution

Solution Implementation Checklist

- [ ] Trying only one solution at a time
- [ ] Documenting each action taken
- [ ] Recording results of each attempt
- [ ] Testing after each change
- [ ] Keeping user informed of progress
- [ ] Not skipping obvious solutions
- [ ] Taking screenshots of errors when helpful
- [ ] Stopping if making problem worse

Problem Resolution Verification Checklist

- [ ] Original problem completely resolved
- [ ] No new problems created
- [ ] All related functions tested
- [ ] User confirmed everything works normally
- [ ] Stable solution (not just temporary fix)
- [ ] User trained on prevention if applicable
- [ ] Follow-up scheduled if necessary
- [ ] User satisfied with resolution

Documentation Completion Checklist

- [ ] Problem clearly described
- [ ] All steps taken documented
- [ ] Working solution identified
- [ ] Root cause explained when known

- [ ] Time spent recorded
- [ ] Knowledge base updated
- [ ] User feedback captured
- [ ] Ticket properly closed

Common Beginner Mistakes

Mistake #1: Jumping to Solutions Too Quickly

Wrong: "Computer slow? Let me reinstall Windows." **Right:** "Let me check what's running and how much memory is being used first."

Mistake #2: Not Asking Enough Questions

Wrong: "Computer won't start" → immediately checking hardware **Right:** Ask "What exactly happens when you press the power button?"

Mistake #3: Trying Multiple Solutions at Once

Wrong: Updating drivers, clearing cache, and restarting simultaneously **Right:** Update drivers, test, then try next solution if needed

Mistake #4: Poor Documentation

Wrong: "Fixed computer problem" **Right:** "Resolved slow performance by removing startup program consuming 80% CPU"

Mistake #5: Not Verifying the Fix

Wrong: Problem appears fixed, closing ticket immediately **Right:** Test thoroughly and confirm with user before closing

Mistake #6: Skipping Research

Wrong: Guessing at solutions without checking knowledge base **Right:** Always check for known issues and documented solutions first

---

Quick Reference Guide

The 6 Steps - Memory Aid

Identify the problem **R**esearch potential causes
Develop action plan **I**mplement solutions **V**erify the fix **D**ocument everything

Remember: "I Really Do Incredible Very Detailed" work!

Essential Questions for Every Call

1. "What exactly is happening?"
2. "When did this start?"
3. "What were you trying to do?"
4. "What error messages do you see?"
5. "Has anything changed recently?"

Red Flags - When to Escalate

- User mentions critical business impact
- Security-related issues
- Network-wide problems
- Hardware failures requiring replacement
- Issues you've spent 2+ hours on without progress
- User requests supervisor involvement

Emergency Response Priority

Priority	Level	Description	Examples
P1	Critical	System down, security breach, multiple users affected	Server crash, malware infection, network outage
P2	High	Single user can't work, important deadline impact	Computer won't boot, software crash before presentation
P3	Medium	Reduced functionality, workarounds available	Printer issues, slow performance, minor software bugs
P4	Low	Minor issues, enhancement requests	Cosmetic issues, feature requests, training questions

---

Remember: Great troubleshooters aren't born knowing every solution - they're made by following consistent methodology and learning from every problem they solve!
