🚀 Quick Start: Fix Demo Accounts & Resume Testing

⚡ 1-Minute Summary

Problem: Demo accounts don’t work on production
Impact: All testing blocked
Fix: 10-minute command
Then: 4 hours of comprehensive testing

🔴 STEP 1: Fix Demo Accounts (10 minutes)

Access Render Shell

1. Go to https://dashboard.render.com
2. Login if needed
3. Select **carelinkai** service
4. Click **“Shell”** tab
5. Wait for shell to connect

Run Seed Command

```
cd /opt/render/project/src
npm run seed:demo
```

Expected Output:

```
✅ Created demo.admin@carelinkai.test
✅ Created demo.operator@carelinkai.test
✅ Created demo.aide@carelinkai.test
✅ Created demo.family@carelinkai.test
✅ Created demo.provider@carelinkai.test
```

Verify Accounts

```
node -e "const { PrismaClient } = require('@prisma/client'); const prisma = new PrismaClient(); (async () => { const count = await prisma.user.count({ where: { email: { contains: 'demo' } } }); console.log('Demo accounts:', count); await prisma.\$disconnect(); })();"
```

Expected: `Demo accounts: 5`

🧪 STEP 2: Test Login (2 minutes)

1. Open https://carelinkai.onrender.com/auth/login
2. Enter:
   - Email: `demo.admin@carelinkai.test`
   - Password: `DemoUser123!`
3. Click **Sign in**
4. ✅ **SUCCESS**: Should redirect to admin dashboard
5. ❌ **FAIL**: Check logs, re-run seed script

📋 STEP 3: Run Comprehensive Tests (4 hours)

Test Plan

Phase 1: Account Verification (30 min)

Test all 5 accounts:
- [ ] demo.admin@carelinkai.test
- [ ] demo.operator@carelinkai.test
- [ ] demo.aide@carelinkai.test
- [ ] demo.family@carelinkai.test
- [ ] demo.provider@carelinkai.test

Phase 2: Recent Fixes (90 min)

Login as **demo.family@carelinkai.test**:

Gallery Upload:
- [ ] Navigate to Gallery tab
- [ ] Click “Upload Photos”
- [ ] Select test image
- [ ] Upload succeeds
- [ ] Photo appears in gallery
- [ ] Thumbnail displays correctly

Document Upload:
- [ ] Navigate to Documents tab
- [ ] Click “Upload Document”
- [ ] Select test PDF
- [ ] Upload succeeds
- [ ] Document appears in list

Activity Feed:
- [ ] Navigate to Activity tab
- [ ] Gallery upload shows in feed
- [ ] Document upload shows in feed
- [ ] Timestamps correct

Dashboard (as admin):
- [ ] Logout, login as demo.admin@carelinkai.test
- [ ] Dashboard loads without errors

- [ ] Alerts display correctly
- [ ] Statistics show

Phase 3: Module Testing (90 min)

Test each module:
- [ ] Residents
- [ ] Inquiries
- [ ] Caregivers
- [ ] Calendar
- [ ] Reports
- [ ] Family Portal (all 8 tabs)

Phase 4: Performance (30 min)

Measure:
- [ ] Dashboard load time
- [ ] Gallery upload time
- [ ] Document upload time
- [ ] Page navigation speed

🔧 STEP 4: Prevent Recurrence (1 hour)

Update Build Command

In Render Dashboard:
1. Go to Service Settings
2. Find **Build Command**
3. Update to:
`bash`
```
   npm install && npx prisma generate && npm run build && npm run seed:demo
```
4. Save

Test Next Deploy

1. Trigger a test deployment
2. Check logs for seed output
3. Verify accounts still work after deploy

📊 Documents Created

For You

1. **COMPREHENSIVE_TEST_RESULTS_DEC14.md** (29 KB)
   - Full technical report
   - All findings
   - Complete recommendations

2. **EXECUTIVE_TEST_SUMMARY_DEC14.md** (11 KB)
   - Executive summary

   - Quick status overview
   - Decision matrix

3. **TESTING_SESSION_REPORT.md** (15 KB)
   - What happened
   - Decisions made
   - Next steps

4. **TESTING_QUICK_START.md** (this file)
   - Quick reference
   - Fast instructions
   - Checklist

✅ Success Checklist

After completing all steps:

- [ ] Demo accounts seeded
- [ ] All 5 accounts can login
- [ ] Gallery upload works
- [ ] Document upload works
- [ ] Activity feed populates
- [ ] Dashboard displays correctly
- [ ] All modules accessible
- [ ] Build command updated
- [ ] Test reports updated

🆘 If Something Fails

Seed Script Fails

Try manual account creation:

```
# Create one account manually
node -e "
const { PrismaClient } = require('@prisma/client');
const bcrypt = require('bcryptjs');
const prisma = new PrismaClient();

(async () => {
  const hash = await bcrypt.hash('DemoUser123!', 10);
  const user = await prisma.user.upsert({
    where: { email: 'demo.admin@carelinkai.test' },
    update: { passwordHash: hash, status: 'ACTIVE' },
    create: {
      email: 'demo.admin@carelinkai.test',
      passwordHash: hash,
      firstName: 'Admin',
      lastName: 'User',
      phone: '(555) 000-0000',
      role: 'ADMIN',
      status: 'ACTIVE',
      emailVerified: new Date()
    }
  });
  console.log('✅', user.email);
  await prisma.\$disconnect();
})();
"
```

Login Still Fails

Check:
1. Database connection (DATABASE_URL env var)
2. Prisma schema synced (`npx prisma generate`)
3. Server logs for errors
4. Account actually exists in database

See Full Troubleshooting

Read: `URGENT_FIX_DEMO_ACCOUNTS.md`

📞 Contact

Questions? See detailed reports in project directory
Issues? Check Render logs and application logs
Updates? This guide will be updated after testing completes

⏱️ Time Estimates

Task	Time
Fix demo accounts	10 min
Test login	2 min
Comprehensive tests	4 hours
Update build process	1 hour
TOTAL	~5 hours

🎯 Priority

NOW (10 min): Fix demo accounts
TODAY (4 hours): Run comprehensive tests
THIS WEEK (1 hour): Prevent recurrence

Created: December 14, 2025, 20:50 EST
Status: Ready to execute
Next: Run Step 1 in Render Shell

TL;DR

```
# In Render Shell:
cd /opt/render/project/src && npm run seed:demo

# Then test:
# https://carelinkai.onrender.com/auth/login
# demo.admin@carelinkai.test / DemoUser123!

# Then run comprehensive tests (4 hours)
# Then update build command
# Done!
```