Sample Office Packages

office-packages/abc-company-20250111/

Structure:

```
abc-company-20250111/
├── windows-package/
│   ├── PrintListener.exe
│   ├── appsettings.json
│   ├── install-print-monitor.bat
│   └── README-WINDOWS.txt
├── ubuntu-package/
│   ├── ubuntu_print_monitor.py
│   ├── install-ubuntu-print-monitor.sh
│   └── README-UBUNTU.txt
├── quick-test.sh
├── validate-installation.sh
└── README.txt
```

Configuration (appsettings.json):

json

```json
{
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft": "Warning",
      "Microsoft.Hosting.Lifetime": "Information"
    },
    "EventLog": {
      "LogLevel": {
        "Default": "Information"
      }
    }
  },
  "ApiSettings": {
    "BaseUrl": "http://192.168.1.100:3000/api",
    "Timeout": 30000
  },
  "OfficeConfig": {
    "OfficeName": "ABC Company",
    "Location": "Main Office"
  }
}
```

Ubuntu Configuration (config.ini template):

ini

```ini
[server]
api_url = http://192.168.1.100:3000/api/print-jobs
office_name = ABC Company
timeout = 30

[cups]
monitor_interval = 5
log_level = INFO

[logging]
log_file = /var/log/printmonitor/printmonitor.log
log_level = INFO
max_size_mb = 10
backup_count = 5
```

Client Information:

- **Company**: ABC Company
- **API Server**: 192.168.1.100
- **Dashboard**: http://192.168.1.100
- **Computers**: 10 (8 Windows, 2 Ubuntu)
- **Printers**: 2 (HP LaserJet Pro, Canon ImageRunner)

---

office-packages/tech-solutions-20250111/

Structure:

```
tech-solutions-20250111/
├── windows-package/
│   ├── PrintListener.exe
│   ├── appsettings.json
│   ├── install-print-monitor.bat
│   └── README-WINDOWS.txt
├── ubuntu-package/
│   ├── ubuntu_print_monitor.py
│   ├── install-ubuntu-print-monitor.sh
│   └── README-UBUNTU.txt
├── quick-test.sh
├── validate-installation.sh
└── README.txt
```

Configuration (appsettings.json):

json

```
{
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft": "Warning",
      "Microsoft.Hosting.Lifetime": "Information"
    },
    "EventLog": {
      "LogLevel": {
        "Default": "Information"
      }
    }
  },
  "ApiSettings": {
    "BaseUrl": "http://192.168.2.50:3000/api",
    "Timeout": 30000
  },
  "OfficeConfig": {
    "OfficeName": "Tech Solutions Inc",
    "Location": "Regional Office"
  }
}
```

Ubuntu Configuration (config.ini template):

ini

```
[server]
api_url = http://192.168.2.50:3000/api/print-jobs
office_name = Tech Solutions Inc
timeout = 30

[cups]
monitor_interval = 5
log_level = INFO

[logging]
log_file = /var/log/printmonitor/printmonitor.log
log_level = INFO
max_size_mb = 10
backup_count = 5
```

Client Information:

- **Company**: Tech Solutions Inc
- **API Server**: 192.168.2.50
- **Dashboard**: http://192.168.2.50
- **Computers**: 15 (12 Windows, 3 Ubuntu)
- **Printers**: 3 (HP LaserJet Pro, Xerox WorkCentre, Brother MFC)

Custom README.txt for Tech Solutions:

Print Monitor Deployment Package for Tech Solutions Inc
==========

Created: January 11, 2025
Office: Tech Solutions Inc
API Server: 192.168.2.50
Dashboard: http://192.168.2.50

Contents:
========
1. windows-package/   - Windows Print Listener installation
2. ubuntu-package/   - Ubuntu Print Monitor installation
3. quick-test.sh   - Test script for all computers
4. validate-installation.sh - Installation validation
5. README.txt   - This file

Quick Start:
===========

Windows Computers (12 machines):
1. Copy windows-package/ to each Windows computer
2. Right-click install-print-monitor.bat → "Run as administrator"
3. Follow installation prompts

Ubuntu Computers (3 machines):
1. Copy ubuntu-package/ to each Ubuntu computer
2. Run: sudo ./install-ubuntu-print-monitor.sh
3. Verify with: sudo systemctl status printmonitor

Computer List:
=============
TECH-PC-01 (Windows) - sarah.johnson - Development
TECH-PC-02 (Windows) - mike.davis - Development
TECH-PC-03 (Ubuntu) - alex.wilson - DevOps
TECH-PC-04 (Windows) - lisa.brown - QA
TECH-PC-05 (Windows) - john.miller - Sales
TECH-PC-06 (Windows) - emma.taylor - Marketing
TECH-PC-07 (Ubuntu) - david.lee - DevOps
TECH-PC-08 (Windows) - anna.clark - Support
TECH-PC-09 (Windows) - james.white - Development
TECH-PC-10 (Windows) - maria.garcia - Finance
TECH-PC-11 (Ubuntu) - chris.anderson - System Admin
TECH-PC-12 (Windows) - rachel.martin - HR

```
TECH-PC-13 (Windows) - robert.jones - Sales
TECH-PC-14 (Windows) - jennifer.hall - Marketing
TECH-PC-15 (Windows) - kevin.thomas - Management

Printer Information:
==================
1. HP LaserJet Pro M404dn - Development Floor (192.168.2.101)
2. Xerox WorkCentre 6515 - Main Office (192.168.2.102)
3. Brother MFC-L8900CDW - Executive Floor (192.168.2.103)

Testing:
=======
1. Run quick-test.sh to send test jobs from all computers
2. Check dashboard at http://192.168.2.50
3. Verify all 15 computers are reporting

Validation:
==========
Run validate-installation.sh to check system health

Support:
=======
- Dashboard: http://192.168.2.50
- API Health: http://192.168.2.50:3000/api/health
- IT Contact: Chris Anderson (chris.anderson@techsolutions.com)
- Installation Support: System Admin Team

Installation completed successfully for Tech Solutions Inc!
```

Custom quick-test.sh for Tech Solutions:

bash

```bash
#!/bin/bash
# Quick test script for Tech Solutions Inc

API_SERVER="192.168.2.50"

echo "Testing Print Monitor System for Tech Solutions Inc"
echo "================================================="
echo ""

echo "1. Testing API connection..."
if curl -s "http://$API_SERVER:3000/api/health" > /dev/null; then
    echo "✅ API is responding"
else
    echo "❌ API is not responding"
    exit 1
fi

echo ""
echo "2. Sending test print jobs for all computers..."

# Test jobs for all 15 computers
computers=(
    "TECH-PC-01:sarah.johnson:Windows"
    "TECH-PC-02:mike.davis:Windows"
    "TECH-PC-03:alex.wilson:Ubuntu"
    "TECH-PC-04:lisa.brown:Windows"
    "TECH-PC-05:john.miller:Windows"
    "TECH-PC-06:emma.taylor:Windows"
    "TECH-PC-07:david.lee:Ubuntu"
    "TECH-PC-08:anna.clark:Windows"
    "TECH-PC-09:james.white:Windows"
    "TECH-PC-10:maria.garcia:Windows"
    "TECH-PC-11:chris.anderson:Ubuntu"
    "TECH-PC-12:rachel.martin:Windows"
    "TECH-PC-13:robert.jones:Windows"
    "TECH-PC-14:jennifer.hall:Windows"
    "TECH-PC-15:kevin.thomas:Windows"
)

for computer in "${computers[@]}"; do
    IFS=':' read -r pc_name username os <<< "$computer"

    echo "Testing $pc_name ($username)..."
```

```
    curl -s -X POST "http://$API_SERVER:3000/api/print-jobs" \
        -H "Content-Type: application/json" \
        -d "{
            \"jobId\": \"test-${pc_name,,}-001\",
            \"userName\": \"$username\",
            \"machineName\": \"$pc_name\",
            \"printerName\": \"HP LaserJet Pro M404dn\",
            \"documentName\": \"Test from $pc_name.pdf\",
            \"pageCount\": 1,
            \"status\": \"completed\",
            \"fileSize\": 50000
        }" > /dev/null

    if [ $? -eq 0 ]; then
        echo "✅ $pc_name test job sent"
    else
        echo "❌ $pc_name test job failed"
    fi
done

echo ""
echo "3. Check dashboard at: http://$API_SERVER"
echo "   You should see 15 test print jobs"
echo ""
echo "Test completed for Tech Solutions Inc!"
```

Package Creation Commands

To create ABC Company package:

bash

```
./create-deployment-packages.sh 192.168.1.100 "ABC Company"
```

To create Tech Solutions package:

bash

```
./create-deployment-packages.sh 192.168.2.50 "Tech Solutions Inc"
```

Both packages will be created in:

```
office-packages/
├── abc-company-20250111/
│   ├── windows-package/
│   ├── ubuntu-package/
│   ├── quick-test.sh
│   ├── validate-installation.sh
│   └── README.txt
├── tech-solutions-20250111/
│   ├── windows-package/
│   ├── ubuntu-package/
│   ├── quick-test.sh
│   ├── validate-installation.sh
│   └── README.txt
├── abc-company-20250111.tar.gz
└── tech-solutions-20250111.tar.gz
```

Distribution Methods

USB Drive Distribution:

1. Copy entire office package folder to USB
2. Include instructions for IT staff
3. Test on one computer before mass deployment

Email Distribution:

1. Send compressed .tar.gz file
2. Include extraction instructions
3. Provide support contact information

Network Share Distribution:

1. Upload to company file server
2. Provide download links
3. Include checksums for verification

Customization Notes

Each package is automatically customized with:

- ✅ **Correct API server IP address**

- ✅ **Company-specific office name**
- ✅ **Pre-configured settings files**
- ✅ **Custom computer lists**
- ✅ **Tailored test scripts**
- ✅ **Company-specific documentation**

This ensures each client gets a ready-to-deploy package that requires minimal configuration during installation.