

Azure Deployment Fixes Applied

Overview

Applied comprehensive fixes to `.github/workflows/azure-deploy.yml` to resolve deployment timeouts and startup failures based on the provided recommendations.

Fixes Applied

✅ 1. AZURE_LOCATION Debug Check (Lines 84-85)

Issue: Need to verify AZURE_LOCATION secret is being read correctly

Fix Applied: Added debug logging to confirm secret presence

```
# Debug: print DESIRED_LOCATION to confirm secret presence (mask not shown in logs)
echo "AZURE_LOCATION secret (DESIRED_LOCATION) length:
${#DESIRED_LOCATION}"
```

Benefit: Quickly identify if secret is missing or empty

✅ 2. Requirements.txt Validation (Lines 160-165)

Issue: Oryx needs requirements.txt to install dependencies

Fix Applied: Added explicit validation after poetry export

```
# Verify requirements.txt exists and print it
if [ ! -f requirements.txt ]; then
    echo "ERROR: requirements.txt not found. poetry export failed."
    ls -la
    exit 1
fi
echo "requirements.txt (first 50 lines):"
head -n 50 requirements.txt
```

Benefit: Fail fast if requirements.txt generation fails

✅ 3. Gunicorn + Uvicorn Startup Command (Line 121)

Issue: Using `python -m uvicorn` relies on system pip/uvicorn which may be missing

Fix Applied: Changed to use gunicorn with uvicorn worker

```
# Before:
az webapp config set --resource-group "$RG" --name "$APP_NAME" --
```

```
startup-file "python -m uvicorn app.main_saas:app --host 0.0.0.0 --port 8000"
```

After:

```
az webapp config set --resource-group "$RG" --name "$APP_NAME" --  
startup-file "gunicorn -w 4 -k uvicorn.workers.UvicornWorker  
app.main_saas:app -b 0.0.0.0:8000"
```

Benefit: Uses production-ready server from app environment, more reliable than system python

✓ 4. Increased Health-Check Timeout (Lines 201-235)

Issue: 10-minute timeout too short for Oryx build + dependency install + app startup

Fix Applied: Extended timeout from 10 to 15 minutes

```
# Before: 30 attempts × 20s = 10 minutes
```

```
for i in {1..30}; do  
  # ... health checks ...  
  sleep 20  
done
```

```
# After: 60 attempts × 15s = 15 minutes
```

```
for i in {1..60}; do  
  # ... health checks ...  
  sleep 15  
done
```

Benefit: Allows sufficient time for Azure platform build and app startup

✓ 5. Fixed Diagnostic JSON Quoting (Line 315)

Issue: Complex nested quotes in JSON payloads cause YAML parsing errors

Fix Applied: Simplified JSON file creation to avoid YAML conflicts

```
# Before: Complex here-doc syntax causing YAML errors
```

```
cat > /tmp/ls_root.json <<'JSON'  
{"command":"..."}  
JSON
```

```
# After: Simple echo with proper quoting
```