

# Azure Deployment Guide - Agentic TCO Calculator v5.0

## Zero-Cost Deployment Using Azure Static Web Apps

This guide will help you deploy the calculator to Azure at **zero cost** so your team can test it.

---

### Prerequisites

Before starting, ensure you have:

- ☐ Azure account (free tier works: <https://azure.microsoft.com/free>)
  - ☐ Node.js installed (v16 or higher)
  - ☐ Git installed
  - ☐ Code editor (VS Code recommended)
  - ☐ Azure CLI installed (optional but recommended)
- 

## Part 1: Prepare Your React App

### Step 1: Save the Calculator Code Locally

#### 1. Create project folder:

```
bash
mkdir agentic-tco-calculator
cd agentic-tco-calculator
```

#### 2. Initialize React app:

```
bash
npx create-react-app .
```

#### 3. Replace `src/App.js` with the calculator code from the artifact above

#### 4. Install dependencies:

```
bash
npm install lucide-react
```

## 5. Test locally:

```
bash  
  
npm start
```

Visit <http://localhost:3000> to verify it works

## Step 2: Create Production Build

```
bash  
  
npm run build
```

This creates a `build/` folder with optimized static files.

---

## Part 2: Deploy to Azure (3 Methods)

Choose ONE method below based on your preference.

---

### METHOD 1: Azure Portal (Easiest - NO CODE NEEDED)

#### Step 1: Create Static Web App via Portal

1. Go to <https://portal.azure.com>
2. Click "+ Create a resource"
3. Search for "Static Web Apps"
4. Click "Create"

#### Step 2: Configure Basic Settings

##### Basics Tab:

- Subscription: Select your subscription
- Resource Group: Create new → `rg-agentic-calculator`
- Name: `agentic-tco-calculator`
- Plan type: **Free**
- Region: **East US 2**

- Deployment source: **Other**

Click "**Review + create**" → "**Create**"

### Step 3: Upload Build Files

1. Wait for deployment to complete (~2 minutes)
2. Click "**Go to resource**"
3. In the left menu, click "**Configuration**"
4. Under "**Source**", you'll see you can upload files manually
5. Click "**Upload**" (or use Azure Storage Explorer)

#### Upload method using Azure Storage Explorer:

1. Download Azure Storage Explorer: <https://azure.microsoft.com/features/storage-explorer/>
2. Connect to your Azure account
3. Navigate to Static Web Apps → your app → `$web`
4. Upload all files from your `build/` folder

### Step 4: Get Your URL

1. In Azure Portal, go to your Static Web App resource
2. Copy the URL (looks like: `https://agentic-tco-calculator-xyz123.azurestaticapps.net`)
3. Share with your team!

**Cost: \$0/month** 

---

## METHOD 2: Azure CLI (Faster for Developers)

### Step 1: Install Azure CLI

**Mac:**

```
bash  
  
brew install azure-cli
```

**Windows:** Download from: <https://learn.microsoft.com/cli/azure/install-azure-cli-windows>

## Linux:

```
bash
```

```
curl -sL https://aka.ms/InstallAzureCLIDeb | sudo bash
```

## Step 2: Login to Azure

```
bash
```

```
az login
```

This opens a browser window. Sign in with your Azure credentials.

## Step 3: Create Resource Group

```
bash
```

```
az group create \  
  --name rg-agentic-calculator \  
  --location eastus
```

## Step 4: Create Static Web App

```
bash
```

```
az staticwebapp create \  
  --name agentic-tco-calculator \  
  --resource-group rg-agentic-calculator \  
  --location eastus2 \  
  --sku Free \  
  --source Other
```

## Step 5: Deploy Build Files

### Option A: Using Azure Static Web Apps CLI

```
bash
```

```
# Install SWA CLI
npm install -g @azure/static-web-apps-cli

# Deploy
swa deploy ./build \
  --app-name agentic-tco-calculator \
  --resource-group rg-agentic-calculator
```

## Option B: Using Azure CLI (upload to storage)

```
bash


# Get the default hostname
az staticwebapp show \
  --name agentic-tco-calculator \
  --resource-group rg-agentic-calculator \
  --query "defaultHostname" -o tsv
```

Then upload via Azure Storage Explorer as described in Method 1.

## Step 6: Verify Deployment

```
bash

az staticwebapp show \
  --name agentic-tco-calculator \
  --resource-group rg-agentic-calculator \
  --query "{URL:defaultHostname, Status:provisioningState}"
```

**Cost: \$0/month** 

---

## METHOD 3: GitHub + Azure (Best for CI/CD)

This method automatically redeploys whenever you push code to GitHub.

### Step 1: Create GitHub Repository

1. Go to <https://github.com/new>
2. Name:
3. Visibility: Private (or Public if you want)

#### 4. Create repository

### Step 2: Push Your Code to GitHub

```
bash

# In your local project folder
git init
git add .
git commit -m "Initial commit - Agentic TCO Calculator v5.0"
git branch -M main
git remote add origin https://github.com/YOUR-USERNAME/agentic-tco-calculator.git
git push -u origin main
```

### Step 3: Create Static Web App with GitHub Integration


1. Go to <https://portal.azure.com>
2. Create Static Web App (as in Method 1)
3. BUT for **Deployment source**, select **GitHub**
4. Sign in to GitHub when prompted
5. Select:
  - Organization: Your GitHub username
  - Repository:
  - Branch:
  - Build presets: **React**
  - App location:
  - API location: *(leave blank)*
  - Output location:
6. Click "**Review + create**" → "**Create**"

Azure will automatically:

- Create a GitHub Actions workflow
- Build and deploy your app
- Redeploy on every push to main

## Step 4: Monitor Deployment

1. Go to your GitHub repo
2. Click "**Actions**" tab
3. Watch the build/deploy workflow run
4. Once complete, visit your Azure Static Web App URL

**Cost:** \$0/month 

---

## Part 3: Post-Deployment Configuration

### Enable Custom Domain (Optional)

If you want a custom URL like `calculator.yourcompany.com`:

1. Go to Azure Portal → Your Static Web App
2. Click "**Custom domains**" in left menu
3. Click "**+ Add**"
4. Enter your domain
5. Follow DNS configuration instructions
6. Validate and approve

**Cost:** Custom domain is **FREE** on Azure Static Web Apps

### Configure HTTPS (Automatic)

Azure Static Web Apps automatically provisions SSL certificates. No action needed!

### Set Up Environment Variables

If you need to store API keys or config:

1. Go to Azure Portal → Your Static Web App
2. Click "**Configuration**" in left menu
3. Click "**Application settings**"
4. Add key-value pairs
5. Save

**Note:** For this calculator, no env variables are needed yet.

---

## Part 4: Share With Your Team

### Get the URL

#### Option 1: Azure Portal

1. Go to your Static Web App resource
2. Copy the URL at the top of the Overview page

#### Option 2: Azure CLI

```
bash

az staticwebapp show \
  --name agentic-teo-calculator \
  --resource-group rg-agentic-calculator \
  --query "defaultHostname" -o tsv
```

### Share Access

**For Azure Portal Access** (if team needs to update):

1. Go to Azure Portal → Your Static Web App
2. Click "**Access control (IAM)**"
3. Click "**+ Add**" → "**Add role assignment**"
4. Role: **Contributor** or **Reader**
5. Add team members' Azure accounts

**For Calculator Use** (no Azure account needed):

- Just share the URL!
- Calculator is public, no authentication required

### Create QR Code for Easy Access

Use a free QR code generator:

- <https://www.qr-code-generator.com/>
- Paste your calculator URL



- Download QR code
  - Add to presentations/docs
- 

## Part 5: Monitoring & Maintenance

### View Usage Analytics

1. Azure Portal → Your Static Web App
2. Click "**Metrics**" in left menu
3. View:
  - Total requests
  - Data transfer
  - Response times

### Check Logs

```
bash

az staticwebapp show-logs \
  --name agentic-tco-calculator \
  --resource-group rg-agentic-calculator
```

### Update the Calculator

#### If using GitHub method:

1. Make changes locally
2. `git add .`
3. `git commit -m "Update calculator"`
4. `git push`
5. Azure automatically rebuilds and deploys

#### If using manual upload:

1. Make changes locally
2. `npm run build`






3. Re-upload `build/` folder to Azure Storage

---

## Part 6: Cost Management

### Current Costs (Free Tier Limits)

Azure Static Web Apps Free tier includes:

-  100 GB bandwidth/month
-  0.5 GB storage
-  Custom domains
-  Automatic SSL
-  No limit on requests

**Expected usage for your team:** <1% of limits

### If You Exceed Free Tier

Upgrade to **Standard** tier:

- Cost: ~\$9/month
- Includes: 100 GB bandwidth + additional

But for internal team testing, you'll never exceed free tier.

### Monitor Costs

1. Azure Portal → **Cost Management + Billing**
2. View spending by resource group
3. Set up budget alerts

```
bash
```

```
# Set up $5 budget alert
```

```
az consumption budget create \  
  --budget-name agentic-calculator-alert \  
  --amount 5 \  
  --time-grain Monthly \  
  --start-date 2025-10-01 \  
  --end-date 2026-10-01
```

---

## Part 7: Troubleshooting

### Calculator Not Loading

#### Check 1: Build succeeded?

```
bash
npm run build
# Look for errors
```

#### Check 2: Files uploaded correctly?

- Azure Storage Explorer → `$web` folder should contain:
  - `index.html`
  - `static/` folder with JS/CSS

#### Check 3: URL correct?

- Should be: `https://your-app-name.azurestaticapps.net`
- NOT: `https://your-app-name.azurestaticapps.net/build`

### 404 Error on Refresh

Add `staticwebapp.config.json` to your `public/` folder:

```
json
{
  "navigationFallback": {
    "rewrite": "/index.html"
  }
}
```

Then rebuild and redeploy.

### CORS Issues

Static Web Apps don't have CORS issues (everything is client-side).

If you add an API later:

1. Create `api/` folder in project
  2. Azure will auto-deploy it
  3. CORS is handled automatically
- 

## Part 8: Backup & Version Control

### Save v5.0 Snapshot

#### Create Git Tag:

```
bash

git tag -a v5.0 -m "Agentic TCO Calculator v5.0 - Platform TCO"
git push origin v5.0
```

#### Export Azure Resource:

```
bash

az staticwebapp show \
  --name agentic-tco-calculator \
  --resource-group rg-agentic-calculator \
  > azure-config-v5.0.json
```

### Backup Build Files

```
bash

# Create backup
mkdir -p backups
cp -r build/ backups/build-v5.0-$(date +%Y%m%d)
```

---

## Quick Reference Commands

```
bash
```

*# Login to Azure*

```
az login
```

*# Check deployment status*

```
az staticwebapp show \  
  --name agentic-tco-calculator \  
  --resource-group rg-agentic-calculator
```

*# Get URL*

```
az staticwebapp show \  
  --name agentic-tco-calculator \  
  --resource-group rg-agentic-calculator \  
  --query "defaultHostname" -o tsv
```

*# Delete (if needed)*

```
az staticwebapp delete \  
  --name agentic-tco-calculator \  
  --resource-group rg-agentic-calculator
```

*# Delete resource group (removes everything)*

```
az group delete --name rg-agentic-calculator
```

---

## Success Checklist

- ☐ Calculator loads at Azure URL
- ☐ All 4 platforms selectable
- ☐ T-shirt sizing works
- ☐ TCO calculations accurate
- ☐ Mobile responsive
- ☐ Team can access without Azure login
- ☐ QR code created and shared
- ☐ Documentation distributed
- ☐ Backup created

---

## Next Steps After Deployment

1. **Test with team** - Gather feedback
2. **Track actual vs. estimated** - Start calibration data

3. **Plan v5.1 enhancements** - Based on usage patterns
  4. **Present to leadership** - Using Google Slides content
  5. **Integrate with workflow** - Add to RFP process
- 

## Support

### Issues?

- Azure docs: <https://learn.microsoft.com/azure/static-web-apps/>
- Open issue on GitHub repo
- Contact: [Your email/Slack]

### Questions?

- Calculator usage: See documentation artifact
  - Methodology: See full documentation
  - Deployment: This guide
- 

**Document Version:** 1.0

**Last Updated:** October 19, 2025

**Deployment Cost:** \$0/month (Free tier)

**Deployment Time:** 15-30 minutes