**Deployment/Runbook for DTN ADMLOD Export Prototype**

**Purpose**

This runbook provides step-by-step instructions for building, configuring, running, and troubleshooting the DTN ADMLOD Export Prototype.

**Prerequisites**

* .NET 6.0 SDK or later installed (https://dotnet.microsoft.com/download)
* Access to the project source code (local or version control)
* Command-line access (Windows, Linux, or Mac)

**Directory Structure**

/AdmLodPrototype

Program.cs

MockData.cs

AdmLodFileGen.cs

FtpClientMock.cs

ResponseParser.cs

Logger.cs

/MockFiles

adm\_export.txt

adm\_response\_success.txt

adm\_response\_error.txt

**Build Instructions**

1. Open a terminal or command prompt.
2. Navigate to the project root directory.
3. Run:
4. dotnet build

**Run Instructions**

* **Basic Export (Success Scenario):**
* dotnet run export --success
* **Export with Error Scenario:**
* dotnet run export --error
* **Export with Logging to File:**
* dotnet run export --success --logfile export.log

**Expected Behavior**

* The prototype generates an ADMLOD export file in /MockFiles/adm\_export.txt.
* Simulates FTP upload by writing the file locally and logging the action.
* Parses a mock DTN response file (adm\_response\_success.txt or adm\_response\_error.txt).
* Logs all actions and errors to the console and, if specified, to a log file.

**Troubleshooting**

* **Build Errors:**  
  Ensure .NET 6.0+ is installed and your terminal is in the correct directory.
* **File Not Found:**  
  Ensure /MockFiles exists and contains the required mock response files.
* **No Output:**  
  Check for errors in the console or log file. Ensure CLI arguments are correct.
* **Customizing Mock Data:**  
  Edit MockData.cs or the files in /MockFiles to change products, product groups, or response scenarios.

**Maintenance**

* To add new scenarios, update or add new mock response files in /MockFiles.
* To extend functionality, follow the coding guidelines and modular structure.