# Introduction

In the eShopOnContainers scenario, the Catalog.WebFormas application enables internal users to update the catalog. It’s a forms-over-data CRUD app built on WebForms. Over time, it’s been refactored to use the Catalog API microservice. However, there isn’t a business imperative to rewrite this application. It’s user base is small and internal, and there aren’t many new features that are needed.

Scenarios like these benefit from a “lift and shift” operation to enable running these applications in a Docker container. The phrase “lift and shift” describes the scope of the task: You *lift* the entire application from a physical or virtual machine, and *shift* it into a container. In ideal situations, you don’t need to make any changes to the application code to run it in a container.

You get many benefits from this small amount of work. Your deployments will all be running using the same image. That means all instances are running in the same environment. The entire development team can run and test the application in a Docker container. This workflow ensures everyone is running the application in the same environment, and that environment is the same as (or very similar to) the production environment.

This “lift and shift” scenario may be either the goal for your application, or a first step toward more of a microservices architecture. Either is valid, and which you choose depends on the returns from further investment for each application.

# Tour of the existing application

## Read Catalog

## Add item

## Edit

## Delete

# Adding Docker Support

## Building

## Running

## Debugging

# Notes on the Catalog service in the development environment

# Notes on the guidance

Just add docker (Add -> Docker)

The first build takes quite a while because it is pulling down the images.