

Building End-to-End Multi-Client Service Oriented Applications

Module 01

A Car Rental System

Demo of a service-oriented application



The Car Rental System

- **Why this type of app?**
 - Business requirement lead to multiple types of clients (UIs)
- **Pieces of the Car Rental system**
 - WCF-based internal services
 - Web API-based public API
 - ASP.NET MVC/Knockout public-facing web site
 - Customer account creation & car reservations
 - WPF desktop client
 - Used in home office to execute a rental and handle a return

Goals of the Case Study Application

- **Implementing connectivity among layers in SOA**
- **Patterns of abstraction, reusability, and testability among all layers of application**
- **Code elegance, cleanliness, and manageability**
- **Give you a great kick-off point for any future apps**
 - Continuously works for me !

Software Configuration Management

- **Flat folder structure for simplicity**
- **Project/folder naming implies relationships**
 - Dot-notation for project and folder naming
- **Abstractions held in own project**
 - Facilitates test dependencies