

Pivotal CRM 6.0 Advanced Customization (60530)

Audience: Pivotal CRM 6.0 Application customizers.

Duration: 5 days Instructor-led training

Location: CDC Software training locations or on Customer's premises

Price: \$3,000 per person for public classes and \$15,000 for onsite customer classes

Prerequisites:

- 60200-C: Pivotal CRM 6.0 Toolkit Customization / Pivotal CRM 6.0 Foundations and Customization Boot Camp or equivalent working experience with Pivotal Client and Pivotal CRM 6.0 Toolkit Customization
- Working experience with C#

Course Overview

This course is a five-day class designed to instruct C# developers who have Pivotal CRM 6.0 Toolkit Customization experience on modifying the Pivotal CRM application business logic. The course materials include a Participant's Guide that contains textual information, screenshots, illustrative diagrams and exercises structured to enforce the learning of concepts presented by the instructor. The course materials also include a comprehensive project performed by each participant on the last day of the class. The project is a complete Pivotal CRM application business logic customization solution.

You will learn:

- The Pivotal CRM application advanced customization best practices
- About the dataflow between and within the different layers of the Pivotal CRM architecture
- How to configure your customization environment for Pivotal CRM code development
- Setting up a Visual Studio project for developing Client Tasks and Server Tasks
- How to develop and implement Client Tasks and Server Tasks
- About the Pivotal Foundation Library 6.0

Course Benefits

The Pivotal CRM out-of-the-box application business logic meets common business needs, but it's highly customizable. This course will provide you with the skills required to efficiently develop the business logic that fits your company's business processes. The Instructor presentations include real world examples, where the exercises and the final project provide the hands-on learning experience and make up more than 60% of the class time.

Course Benefits

Customizing the Pivotal CRM Application

- Brief overview of the application Toolkit customization and best practices
- Brief overview of the application advanced customization and best practices

Pivotal CRM Architecture and Data Flow

- Pivotal CRM Architecture: Description of the Data, Server and Client tiers
- Analysis of the client-side data flow
 - Client Tasks
 - Client API
 - Foundation Library

- Server Proxy
- Analysis of the server-side data flow
 - Server Tasks
 - Foundation Library
 - Server API
 - COM + Component
- Pivotal CRM Application Programming Interfaces
 - Pivotal Client 6.0 API
 - Pivotal Form Designer 6.0 API
 - Pivotal Foundation Library 6.0 API
 - Pivotal Server 6.0 API
 - Pivotal Integration for Microsoft Outlook 6.0 API
- Pivotal CRM Out-of-the-Box Application Assemblies
 - CMS Application Client Tasks and Server Tasks

Creating a Developer Customization Environment

- Basic Installation Requirements
- Pivotal CRM Software Components Configuration
 - Installing the application Source Coded and Foundation Library
 - Configuring environmental variables
 - Making Pivotal C# project templates accessible form Visual Studio
 - Adjusting the Pivotal Configuration Utility settings

Coding for Pivotal

- CRM Pivotal CRM source code examples
 - Code constructs
 - Access modifiers
 - Classes and methods
 - Inheritance and Polymorphism
 - Additional topics
- Naming Convention
 - Pivotal CRM naming standards: Namespaces, Classes, Methods and Variables

Setting up a Visual Studio Development Project

- Client Task Project
 - Using the Pivotal template to create a Client Task Class Library
 - Creating a Form Client Task Class
 - Creating a Command Handler Class
 - Adding partial classes for the Event Handlers
 - Building and Importing Client Task Code Files
 - Linking a Code file to a Client Form
- Server Task Project
 - Using the Pivotal Templates to create a Server Task Class Library
 - Creating a Form Server Task Class
 - Creating an Application Server Task Class
 - Creating a Service Server Task Class
 - Building and importing Server Task code files
 - Creating an Appserver Rule Reference

- Linking a Server Task to a Client Form
- Debugging
 - Transferring built files to the Server and Client Task folders
 - Attaching Client Task projects to the CDC Smart Client Container process
 - Running Server Task projects in an external program for Pivotal usiness Server
 - Using debugging tools with Client and Server Task projects

Developing Client Tasks

- Types of Client Tasks: Classifying Client Tasks by role and scope
 - Form Client Tasks
 - Command Handlers
 - Notification Handler
 - Global (static) Client Tasks
- Form Client Tasks – Base Methods
 - Breaking execution flow inside Transition Point methods
 - Inserting pre-processing and post-processing code
 - Skipping the default functionality via Transition Point
 - Override OnForminitialized and OnFormReloaded events
- Form Control Events
 - Creating an Event Handler method
 - Linking Event Handler to various interface components
 - Reviewing methods available for specific Pivotal Controls on a Client Form
 - Programming Secondary Grid Events
 - Canceling Events
- Messages and Dialogs
- Working with DataSets
- Form Control Properties
- Displaying Forms, Reports, and Searches
 - Invoking Foundation Libraries ShowForm methods
 - Using ShowReport method
 - Running Reports against a Search Result
 - Search show Methods
 - Displaying objects in Current, Modeless and Modal windows
 - Passing parameters to Forms, Reports, Searches
- Notification Handlers
 - Marking methods with NotificationEvent attribute
 - Writing Notification Handler Signatures with various parameters
 - Using Windows, Form, Search, Report, Portal and other element-specific Notification Events
- Command Handlers
 - Marking methods with CommandClientTask attribute
 - Communication between the CommandHandler and current Client Form
 - Linking Command Handlers to the Client Task Commands
- Task Pads
 - Task Pad class hierarchy
 - Disabling and Hiding the Tasks
- Transition Point Parameters
 - Transition Point Parameter Array structure

- Constructing the Transition Point Parameters
- Passing Parameters between objects
- Extracting Parameters
- Transition Point Parameters
 - Transition Point Parameter Array structure
 - Constructing the Transition Point Parameters
 - Passing Parameters between objects
 - Extracting Parameters
- Global Methods
 - Creating static classes, methods and properties
 - Calling static methods from Client Tasks
 - Inheriting Foundation Library Global class
 - Utilizing Global methods in a subclass

Developing Server Tasks

- Passing Data between tiers with Transition Point Parameter Array
 - Communication between Client and Server Tasks via Transition Points
 - Passing Default Columns array from a Client Task to a Server Task of Another form
 - Set values of Default Columns
 - Exploring more patterns of Transition Point programming
- Communicating between Server Tasks
 - Referring Out-of-the-Box Server Tasks
 - Instantiating Out-of-the-Box Server Tasks
 - Calling the Out-of-the-Box Server Task methods
- Cascade Delete Logic
- Execute Server Task Methods from a Client Task
 - Call Server Tasks Directly with ExecuteServeTask method
 - Create a Server Proxy Class
 - Use Server Proxy methods to call Server Task methods from Client Tasks
- Inheritance
 - Inherit Out-of-the-Box Server Task class
 - Override Out-of-the-Box methods in a subclass to change the superclass behavior
 - Re-reference the AppServer Rules for the OOB class and a custom subclass
- Scheduled Service
 - Configuring a Scheduled Task Service via Pivotal Configuration Utility
 - Setting up a Scheduled Task in a Customization System
 - Calling the Server Task method according to the Schedule
 - Additional Server Task techniques
 - Determining Users Security group
 - Querying a Database from a Server Task
 - Using a Language Dictionary
 - Exception Handling based on the Foundation Library Classes

Interactions

- Concepts and Terminology
- Extension forms
 - Add Fields to an existing Extension form
 - Create a new extension form (& Table)

- Discussing Interaction class hierarchy in Foundation Library
 - Instantiating interaction objects
 - Adding Players to Player Collection
 - Sending Email programmatically from server
 - Creating Todos, Meetings and other interactions from the Server Task
 - Unlinking Interactions

- Project: Create and Implement the business logic for custom business processes according to Pivotal CRM Advanced Customization best practices
 - Create a Client Task
 - Create a Server Task
 - Set Communication between Client and Server Task