PDSA .NET Productivity Framework – User Manual

By: PDSA, Inc.

August, 2016
Published By: PDSA, Inc.

Copyright © 2002-2016, PDSA, Inc.
All Rights Reserved Worldwide

Written By: Paul D. Sheriff
Technical Editors: PDSA, Inc.
Quality Assurance: PDSA, Inc.

Every effort has been made to supply complete and accurate information. However, PDSA, Inc. assumes no responsibility for its use, nor for any infringement of the intellectual property rights of third parties which would result from such use.

Copyright © 2002-2015 PDSA, Inc.

All rights reserved worldwide. No part of this publication may be stored in a retrieval system, transmitted, or reproduced in any way, including but not limited to photocopy, photograph, magnetic or other record, without the prior agreement and written permission of the publisher.

Printed in the United States of America

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

Table of Contents

Table of Contents 3
PDSA .NET Productivity Framework Introduction 11
Goals of the PDSA .NET Productivity Framework 12
Product Highlights 12
Reusable Application Components 12
Reusable Code Library 13
Sample Applications and Templates 13
Developer Utilities 14
Framework Utilities 14
Standards for IT Projects 14
System Requirements 14
Supported ADO.NET Providers 14
Your Knowledge 15
Technical Support 15
Fee based support 16
Support Options 16
Support Policy 16
Ownership of the Software 17
Redistribution Rights 17
Use and Redistribution Rights 17
Publication of Software 18
Restrictions 18
Per-Developer License 18
Framework Source Code Restrictions 18
Warranty 18
Acknowledgements 20
Chapter 1 1-1
Overview of the PDSA .NET Productivity Framework 1-1
Why you need a Framework 1-1
Goals of an Application Framework 1-2
Application Templates 1-2
Guidelines & Standards 1-2
Overview of the PDSA Framework Namespaces 1-3
PDSA Framework Utilities 1-4
PDSA Database Tables 1-4
Summary 1-6

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

Chapter 2 2-1
Installation of the PDSA .NET Productivity Framework 2-1
Installing the PDSA .NET Productivity Framework 2-1
PDSA Product Activation Screen 2-2
Windows Firewall 2-3
Limitations of Trial Version 2-3
Error 1925 or Problems Installing PDSA Framework 2-4
Framework Folders Installed 2-5
PDSA Sample Application Login Info 2-5
Change the Connect String in the Samples 2-6
Uninstalling the Framework 2-6
Summary 2-7
Chapter 3 3-1
Creating a Web Forms Application 3-1
Web Forms Application Generation 3-1
Step 1: Project Type 3-2
Step 2: Project Info 3-3
Step 3: Application Database 3-4
Step 4: PDSA Database 3-4
Step 5: Security 3-5
Step 6: Create the Project 3-6
Your New Project 3-7
Generate Code Using Haystack 3-9
Copy Generated Files into Application 3-12
Add Menu Items 3-13
Summary 3-16
Chapter 4 4-1
Creating an MVC Application 4-1
MVC Application Generation 4-1
Step 1: Project Type 4-2
Step 2: Project Info 4-3
Step 3: Application Database 4-4
Step 4: PDSA Database 4-4
Step 5: Security 4-5
Step 6: Create the Project 4-6
Your New Project 4-7
Generate Code Using Haystack 4-8

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

Copy Generated Files into Application 4-12
Add Menu Items 4-14
Summary 4-16
Chapter 5 5-1
Creating a WPF Application 5-1
Windows Presentation Foundation Application 5-1
Step 1: Project Type 5-2
Step 2: Project Info 5-3
Step 3: Application Database 5-4
Step 4: PDSA Database 5-5
Step 5: Security 5-6
Step 6: Create the Project 5-7
Your New Project 5-8
Generate Code Using Haystack 5-9
Copy Generated Folders/Files into WPF Application 5-12
Call Your Generated Control 5-13
Run the Application 5-14
Summary 5-15
Chapter 6 6-1
Namespaces in the PDSA Framework 6-1
Common Assemblies 6-1
.NET Framework DLLs needed 6-2
PDSA.ActiveDirectory 6-2
Sample Folder 6-2
Assemblies 6-2
PDSA.Cache 6-3
Sample Folder 6-3
Assemblies 6-3
PDSA.Common 6-3
Sample Folder 6-3
Assemblies 6-3
PDSA.Configuration 6-4
Sample Folder 6-4
Assemblies 6-4
PDSA.Cryptography 6-4
Sample Folders 6-4
Assemblies 6-4
PDSA.Cryptography.Hash 6-5

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

PDSA.Cryptography.KeyManagement 6-5
PDSA.Cryptography.Symmetric 6-5
PDSA.DataAccess 6-5
Sample Folders 6-5
Assemblies 6-5
PDSA.DataLayer 6-6
Sample Folders 6-6
Assemblies 6-6
PDSA.DataLayer.DataClasses 6-6
PDSA.DataLayer.Schema 6-6
Sample Folders 6-7
Assemblies 6-7
PDSA.FileIO 6-7
Sample Folders 6-7
Assemblies 6-7
PDSA.Framework 6-7
Sample Folders 6-7
Assemblies 6-8
PDSA.Framework.BusinessLayer 6-8
Assemblies 6-8
PDSA.Framework.DataLayer 6-8
Assemblies 6-8
PDSA.Framework.EntityLayer 6-8
Assemblies 6-8
PDSA.Framework.ViewModelLayer 6-9
Assemblies 6-9
PDSA.Logging 6-9
Sample Folder 6-9
Exception Logging: General Usage 6-9
Configuration Settings 6-10
Assemblies 6-10
PDSA.Menus 6-10
Assemblies 6-10
PDSA.MessageBroker 6-11
Sample Folders 6-11
Assemblies 6-11
PDSA.Provider 6-11
Assemblies 6-11
PDSA.ReferenceTable 6-11
Assemblies 6-12

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

PDSA.ReferenceTable.Web 6-12
Assemblies 6-12
PDSA.Resource 6-12
Sample Folder 6-12
General Usage 6-12
Assemblies 6-13
PDSA.Security 6-13
PDSA.ServiceModel 6-14
Assemblies 6-14
PDSA.SqlServer 6-14
Sample Folder 6-14
Assemblies 6-14
PDSA.SqlServer.Picker 6-15
Sample Folder 6-15
Assemblies 6-15
PDSA.Tips 6-15
Sample Folder 6-15
Assemblies 6-15
PDSA.UI 6-15
Assemblies 6-16
PDSA.Validation 6-16
Sample Folder 6-16
Assemblies 6-16
PDSA.Web 6-16
Sample Folder 6-16
Assemblies 6-16
PDSA.WindowsService 6-17
Assemblies 6-17
PDSA.WPF 6-17
Sample Folder 6-17
Assemblies 6-17
Summary 6-18
Chapter 7 7-1
Using the PDSA .NET Providers 7-1
Overview of PDSA Providers 7-1
Caching Provider 7-2
Sample Folder 7-2
Configuration 7-3
Caching Data in Desktop Application 7-3
User Specific Cache Data 7-4

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

Absolute Expiration on Cache Data 7-4
Sliding Expiration on Cache Data 7-5
Removing Data from Cache 7-5
Caching Data in a Web Application 7-5
Assemblies 7-5
Configuration Provider 7-6
Sample Folder 7-6
Configuration 7-6
Read Configuration Settings 7-7
Save Configuration Settings 7-8
Assemblies 7-8
Cryptography Provider 7-8
Sample Folders 7-8
Configuration 7-9
Encrypting and Decrypting 7-10
Hashing 7-11
Assemblies 7-12
Data Layer Provider 7-12
Sample Folder 7-12
Configuration 7-12
Attributes in Data Provider 7-13
Assemblies 7-14
Database Schema Provider 7-14
Sample Folder 7-14
Configuration 7-14
Attributes in DBSchema Provider 7-15
Assemblies 7-16
Logging Provider 7-16
Sample Project 7-16
Things you can Log 7-16
Configuration 7-17
Using the Logging Provider 7-17
Logging Exceptions 7-18
Use the AppLog Classes 7-19
Resource / Localization Provider 7-19
Sample Folder 7-20
General Usage 7-20
Storing Resources in SQL Server 7-20
Assemblies 7-22
Summary 7-23
Chapter 8 8-1

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

Using the PDSA Security System 8-1
Overview of the PDSA Security System 8-1
Multi-Tenant Architecture 8-2
Application (Forms-Based) Security 8-3
Active Directory Security 8-3
Application Single Sign-on Security 8-4
Active Directory with PDSA Roles 8-5
User Stories when using PDSA Security 8-6
Application Security Entities 8-7
Application 8-7
Entity 8-7
Users 8-8
Roles 8-8
Permissions 8-8
Things you can do with the PDSA Security System 8-8
MVC: Security through Code 8-11
Web Forms: Security through Code 8-12
Access Settings from Config File 8-12
Menu Item Settings for Active Directory with PDSA Roles Security 8-13
Summary 8-14
Chapter 9 9-1
Localizing PDSA Framework Applications 9-1
Localization 9-1
Advantages of using XML or SQL Server 9-3
Sample 9-4
MVC Sample 9-4
DLL's Needed 9-5
Summary 9-6
Chapter 10 10-1
Reference Table System 10-1
Standard Schema 10-1
Using Reference Tables Step-By-Step 10-2
Using the PDSA Reference Tables 10-6
Samples 10-6
Summary 10-7
Chapter 11 11-1
User Filter State 11-1

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

Overview of User Filter State 11-1
The PDSAUserFilterState Class 11-1
Public Methods 11-2
Overridable Methods 11-3
Events 11-4
Filter State Pre-Requisites 11-5
Adding Filter State 11-5
Filter State Sample 11-7
How-To's 11-7
Create a Default State 11-7
Customize State Handling 11-9
Handle an Unknown Control 11-10
Summary 11-12
Chapter 12 12-1
Multi-Tenant Framework Applications 12-1
PDSAApplication Configuration Section 12-2
PDSASecurityConfiguration Section 12-3
Entities 12-3
Create an Entity 12-3
Associate Admin Account with Entities 12-4
Roles 12-5
User Roles 12-7
Users 12-9
Web.config Settings 12-11
Summary 12-13

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

PDSA .NET Productivity Framework Introduction

The PDSA .NET Productivity Framework is a set of productivity tools, classes, providers, database tables and complete template applications that will help you develop .NET Service-Oriented Architecture (SOA) applications in a fraction of the time it would take if you started from scratch! Never start from File | New Project again! We give you a complete set of templates to start your typical business applications. These templates already have a working set of SOA data classes, business classes, and validation classes. They contain complete working maintenance forms, a data-driven menu system, data-driven security system, localization, caching, logging, exception handling and hundreds of other classes and tools that you can use to build your business applications in a fraction of the time than if you started from scratch.

Included in the PDSA .NET Productivity Framework is the Haystack Code Generator for .NET. The Haystack Code Generator for .NET is a utility that will help you build SOA-ready data classes for add, edit, delete and searching. MVC, Web Form pages, and WPF CRUD user controls are generated from Haystack. A Data Access Class (DAC) is a wrapper around a single table, view, or stored procedure in a database. This wrapper class contains all of the CRUD (Create/Read/Update/Delete) logic for a table. Or this wrapper class will be a read-only class that can call a view. Or it can be a wrapper class that allows you to fill in parameters and then call a stored procedure and retrieve the output parameters from that stored procedure.

Like any application written in .NET the PDSA Framework is never done. We are continually enhancing this product. You can be assured that you will receive all of the upgrades free of charge for your first year. After that there is a nominal fee that is explained under Support Options in this document. In addition, we would like to ask that if you make any additions, that you send those along to us so we can evaluate them for possible inclusion in the next upgrade. We hope you find this tool as useful as we do, and look forward to hearing from you.

Sincerely,

Paul D. Sheriff

President, PDSA, Inc.

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

Goals of the PDSA .NET Productivity Framework

By using this product you will learn several very useful techniques that will allow you to create MVC, Web Forms, and WPF applications in a fraction of the time than if you had no framework at all.

Product Highlights

There are so many features of the PDSA .NET Productivity Framework that it is hard to cover everything in just one place, however, below is an overview of some of the major features of this comprehensive framework.

Reusable Application Components

- Complete security system for MVC, Web Forms and WPF applications
  - Users, Roles, and Permissions tables
  - Web-based maintenance screens for administering these tables
  - Forms based authentication support
  - Windows authentication support
  - Active Directory support
  - Turn on/off controls on pages with no coding
  - Page-level Security
- Data Driven Menu System for MVC and Web Forms
  - Fully customizable
  - Different menu system layouts can be modified within a matter of minutes
- Configuration Management for storing application settings in Config files, the registry, XML file, or a database table
- Data Access Layer for Service Oriented Architecture development
- Logging System
- Exception Management System
  - Get more exception information than normally available
  - Write exceptions to Event Log, Table, File and Email

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

- User Tracking System
- Audit Trail System
- Reference Table System allows you an easy way to work with lookup tables/validation tables
- Localization System to allow you to display messages, labels and data in your tables in any language.

Reusable Code Library

There are a whole collection of classes with literally hundreds of methods that help make using .NET easier.

- Configuration management (XML, Config, Registry and Database Storage)
- ADO.NET Wrapper Classes
- Transaction Class
- DB Schema Classes
- WPF Classes
- Validation System to work with Entity Classes
- Date Handling
- File Handling
- Math classes
- Measurement classes
- Registry classes
- String Handling
- Soundex Class
- String Class
- System Information Class
- XML Handling Classes
- Cryptography Wrapper Class

Sample Applications and Templates

- Samples that show how to use various PDSA Framework Classes
- Template Application for MVC
- Template Application for Web Forms
- Template Application for WPF

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

Developer Utilities

- Computer Cleaner
- Property Builder
- Project Cleaner
- Convert Color to RGB
- XML Builder

Framework Utilities

- Haystack Code Generator for .NET
- Application Builder to build new applications from a template
- Password Reset

Standards for IT Projects

- Programming Standards for C#
- Database Standards
- Best practices for Database Development

System Requirements

Below are the assumptions about your computing environment.

- You are running Visual Studio.NET 2015 (.NET Framework 4.52) or later
- You are using SQL Server 2008 or later

Supported ADO.NET Providers

The PDSA .NET Productivity Framework works with SQL Server 2008 or higher only.

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

Your Knowledge

The PDSA .NET Productivity Framework is a comprehensive set of tools, providers and classes covering intermediate to advanced topics. It is designed to help someone develop SOA applications using C#. The following are some of the assumptions we will make about your knowledge:

- You have a good understanding of IIS
- You know how to create Web Sites
- You have a good understanding of ASP.NET (if applicable)
- You have a good understanding of C#
- You have a good understanding of Windows Presentation Foundation (if applicable).
- You understand Object Oriented Programming principles
- You understand how to use .NET Providers
- You understand .NET Configuration Files
- You know how to use SQL Server

If you unfamiliar with the above concepts, you can purchase mentoring services from PDSA, Inc.

Technical Support

Because of the complexity of software development, PDSA, Inc. does not provide free technical support related to the use of the C#, ADO.NET, the .NET Framework, MVC, Web Forms, or WPF. We will support you in the installation and understanding of what is in the PDSA .NET Productivity Framework. Technical support is also available for any confirmed defects in the Haystack Code Generator, the DLLs that are a part of the PDSA .NET Productivity Framework software, and the accompanying documentation. We will, of course, provide you with answers to questions related to the use of any of our products. Simply email us at Support@pdsa.com.

TIP:	It is strongly recommended that you read this documentation thoroughly before calling or emailing tech support.

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

Fee based support

Fee-based technical support for C#, MVC, Web Forms, WPF, and SQL Server questions is available via telephone or email. PDSA, Inc. charges per hour billed in 15 minute increments for this type of support (Call us for our latest rates). If we don't answer your question, then you don't pay. You can pre-pay by check or money order, and we accept Visa, MasterCard and American Express.

Please have the following ready when you call with a credit card:

- Credit Card Number
- Name on Card
- Expiration Date
- Address of where you receive the bill for this credit card

Support Options

The following support options are available to registered users of the PDSA .NET Productivity Framework.

- Send us an email with your support question to Support@pdsa.com.
- Fax us your question at (714) 734-9793
- You can mail us a letter at 17852 East 17th Street, Suite 205, Tustin CA, 92780
- You may telephone us at (714) 734-9792. Please ask for Product Support. You will receive 5 free support calls. After that you must pay our normal technical support fees.

Support Policy

The following options are available to you for support on the PDSA .NET Productivity Framework for the first year.

- Automatic notification of upgrades and bug fixes
- 5 free phone calls
- Unlimited email support on PDSA .NET Productivity Framework issues
- Your first year's support on the PDSA .NET Productivity Framework is included in the purchase price. For each year thereafter we charge a 20% yearly maintenance fee

NOTE: Major releases are subject to an additional upgrade charge.

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

Ownership of the Software

The enclosed software program (“Software”) and the accompanying written materials are owned by PDSA, Inc. or its suppliers and are protected by United States copyright laws, by laws of other nations and by international treaties. You must treat the Software like any other copyrighted material except that you may make one copy of the Software solely for backup or archival purpose, and you may transfer the software to a permanent storage device.

Redistribution Rights

You have the right to redistribute any code you generate from our products, the accompanying “Runtime” DLLs from the PDSA .NET Productivity Framework, the CSHTML, ASPX, and ASCX files to the users of your system and any .XAML files contained in our template applications. You may not redistribute the Haystack Code Generator or any DLL that is used exclusively by the code generator. You may not redistribute the source code to any of the classes in the PDSA .NET Productivity Framework DLLs either. Source code to the PDSA .NET Productivity Framework is available for an additional charge.

Use and Redistribution Rights

PDSA, Inc. grants you the right to distribute the source code generated from the Haystack Code Generator and the DLLs from the PDSA .NET Productivity Framework on a non-exclusive, royalty free basis, provided that:

- Each developer using the program owns a license of the software
- You do not distribute the Haystack executable or any DLL that it uses.
- You agree to indemnify, hold harmless, and defend PDSA, Inc. and its suppliers or contractors from and against any claims or lawsuits, including attorneys’ fees that arise or result from the use or distribution of or other activities relating to your software.
- If you claim a copyright, you must add a clause stating “Portions of this program are "Copyright PDSA .NET Productivity Framework" and that you do not claim ownership of these pieces of our software.

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

Publication of Software

You are prohibited from publishing the Haystack Code Generator and PDSA .NET Productivity Framework source code on any medium, including but not limited to, electronic mail, online services, internet newsgroups, and printed matter. The Haystack Code Generator and Productivity Framework source code may not be included in freeware or shareware applications in a non-compiled state.

Restrictions

Under no circumstances may you attempt to reverse engineer this product. You may not rent or lease the Software, but you may transfer the Software and the accompanying written materials on a permanent basis provided you retain no copies, the recipient agrees to the terms in this Software License, and you have not used any of the Haystack Code Generator or PDSA .NET Productivity Framework source code in an application. Ownership transfers must be reported to PDSA, Inc. in writing.

Per-Developer License

The PDSA .NET Productivity Framework and the Haystack Code Generator is sold on a per-developer basis. This means any programmer using any of the software must have a license for working on each computer.

Framework Source Code Restrictions

The Source Code to the PDSA .NET Productivity Framework (purchased separately) may only be distributed to people working at your physical location and only to employees of your company. Any contractors or outside entities not a part of your actual organization may NOT be given the source code in any form.

Warranty

Except as specifically stated in this agreement, the Software and Software Documentation is provided and licenses “As Is” without warranty of any kind, either express or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose.

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

PDSA, Inc. warrants for a period of 60 days from the date of delivery to you that (1) the Software will substantially conform to the Software documentation and specifications published by PDSA, Inc., and (2) the media on which the Software is furnished and the documentation will be free from defects in materials and workmanship under normal use. PDSA, Inc. does not warrant that the functions contained in the Software will meet your requirements or that the operation of the Software will be either error free or appear precisely as described in the Software documentation. You are responsible for selecting the Software to achieve your intended results and for the installation, use and results obtained from the Software. The Software must be used with the computer hardware and operating system for which it is designed, and in accordance with the Software Documentation and specification requirements.

PDSA, Inc. will replace defective media and documentation and correct substantial Software non-conformities at no charge, provided you return the item with dated proof of payment (in case of defect) or otherwise notify PDSA, Inc. in writing (in case of substantial Software non-conformity) within 60 days from the date of delivery to you. If PDSA, Inc. is unable to replace defective media or documentation or correct substantial Software non-conformities, PDSA, Inc. will refund the license fee. These are your sole remedies for any breach of warranty. You or your dealer may return the enclosed registration card to PDSA, Inc. in order for these remedies to be available to you and for you to be eligible for customer support and service during the warranty period.

In no event will PDSA, Inc. be liable to you for any damages, including without limitation loss profits or revenues, loss of data, business interruption loss, recovery or substitution costs, or claims by third parties, or other indirect incidental or consequential damages, arising out of the use of or inability to use the Software, even if PDSA, Inc. or a PDSA, Inc. dealer has been advised of the possibility of such damages. In no case shall PDSA, Inc.'s liability exceed the amount of the license fee paid by you for the Software.

This limited warranty gives you specific legal rights. Some jurisdictions provide other rights, and some jurisdictions do not allow the exclusion or limitation of implied warranties or limitation of liability for incidental or consequential damages. As a result, the above limitations and/or exclusions may not apply to you.

U.S. Government Restricted Rights:

The Software and Software Documentation are provided with Restricted and Limited Rights. Use, duplication or disclosure by the Government is subject to restrictions as set forth in DFAR 252.277-7013(c)(1)(ii) or FAR 52.227.19 as applicable, and additional restrictions set forth in this agreement. Contractor/Manufacturer is PDSA, Inc. located at 17852 E. 17th Street, Suite 205, Tustin, CA 92780.

General:

This agreement is governed by the laws of the state of California. The prevailing party in any action or proceeding brought in connection with a breach of this agreement will be entitled to reimbursement by the other party for costs and reasonable attorneys' fees.

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.

Any questions concerning this Agreement should be referred in writing to PDSA, Inc. located at 17852 17th Street, Suite 205, Tustin, CA 92780.

You acknowledge that you have read this agreement, and by opening this Software package indicate your acceptance of its terms and conditions. You also agree that it is the complete agreement between us and that is supersedes any prior information or representations you received relating to the subject matter of this agreement. No PDSA, Inc. dealer, distributor representative or employee is authorized to make any modifications or additions to this agreement.

THE WARRANTY AND REMEDIES SET FORTH IN THIS LIMITED WARRANTY ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESSED OR IMPLIED.

Acknowledgements

Thanks to all the employees of PDSA, Inc for their help in developing this product.

Copyright © 2002-2016 by PDSA, Inc.
All rights reserved. Reproduction is strictly prohibited.