

# DELPHI IN DEPTH

## DATASNAP

Full Stack Development



Embarcadero Delphi MVP authored by You-Bin.

---

# Publication Information

Pre-cataloging data of National Library publications

-----

Book Name

**Delphi in Depth DataSnap Full Stack Development**

Publisher: You-Bin Wu

Publisher: Master of Wind (Self-Published)

Blog: <http://grandruru.blogspot.com/>

Facebook Page: <https://www.facebook.com/DaJiang.fans/>

-----

---

<b>Publication Information</b>	<b>1</b>
<b>Copyright Notice</b>	<b>10</b>
<b>Endorsements by Friends in the Delphi Community</b>	<b>10</b>
<b>Foreword</b>	<b>14</b>
I'm native, I'm proud	14
Delphi faces impact of modern design practices	14
Problem Summary	15
Technical Integration	16
Content Structure	17
Achieving Better Results	17
Full Book Example Link	18
<b>Building Full-Stack Services from 0 to 1</b>	<b>19</b>
Architecture diagram of the web application in this book	19
Getting to know the Delphi development tool	19
Starting your first DataSnap project	22
First execution	31
Code explanation	32
Building jQuery EasyUI Operating Environment	44
Updates are required for projects when using Delphi XE (2011)	48
<b>Introduction to commonly used EasyUI components</b>	<b>51</b>
Parser	51
SearchBox	52
Progressbar	56
Tooltip	58
Panel	61
Tabs	66
Accordion	73
Layout	76
LinkButton	81
Form	84
ValidateBox	87
ComboBox	90
DateBox	95

---

Window	99
Dialog	102
Messenger	106
DataGrid	109
<b>How to interact between a web application and DataSnap using</b>	
<b>AJAX</b>	<b>113</b>
What is AJAX?	113
Using VanillaJS with DataSnap	122
Introduction to DataSnap REST Message Passing Protocol	124
Using jQuery with DataSnap	127
JavaScript API provided by DataSnap	128
<b>Delphi common syntax</b>	<b>131</b>
Naming Convention	131
Variable Declaration	131
Methods (procedure, function)	132
String Processing	133
Numeric Processing	138
Operator Symbols	140
Flow Statements	146
Containers	154
<b>DataSnap Framework</b>	<b>163</b>
Introduction to Delphi Web Architecture	163
Unicode Support	165
Creating a database	166
DataSnap REST Application Architecture	168
DataSnap API and URI Mapping	169
Request Process and Session Management	170
Introduction to DataSnap REST Server authentication flow	171
State Management and Stateless Protocol Design in DataSnap	173
JSON Parsing	185
<b>Delphi Database Connectivity Component - dbExpress</b>	<b>191</b>
What is dbExpress?	191
Component Access Layer - TSQLConnection	191
Driver layer - TDBXConnection	199
<b>Real-world Development with DataSnap</b>	<b>236</b>

---

Login and main page production	236
Employee Management System Development	264
Building Aquarium World System	298
Summary	321
<b>DataSnap Server Performance Tuning</b>	<b>323</b>
Thread Pooling	323
Avoid creating duplicate files repeatedly	324
Deploying Service as ISAPI Mode	325
<b>Appendix</b>	<b>347</b>
Project Output Configuration	347
Cross-Origin Resource Sharing (CORS)	348
Link to source code for all examples in this book	351

## Copyright Notice

- Delphi is a registered trademark of Embarcadero Corporation.
- Devart dbExpress Driver for SQL Server is copyrighted by Devart.
- Microsoft SQL Server is a registered trademark of Microsoft Corporation.
- jQuery is licensed under the MIT license and is copyrighted by the jQuery team at [jquery.com](http://jquery.com).
- EasyUI is copyrighted by the EasyUI team at [www.jeasyui.com](http://www.jeasyui.com).

## Endorsements by Friends in the Delphi Community

*We are honored to have received these endorsements from our friends in the Delphi community. Their support and encouragement have been instrumental in the development of this book. We would like to express our heartfelt gratitude to each of them.*

*I met You-Bin at a Delphi conference and seminar. However, our relationship didn't progress beyond an introduction until we began discussing Delphi third-tier technology online. Third-tier technology involves adding an application layer (AP layer) between the client and*

---

*server in a client-server architecture to enhance security, particularly when crossing regions or countries.*

*You might ask, "Why not just set up a VPN?" Yes, that would work, but you would be ignoring another security threat that VPNs pose: Trojan viruses.*

*You-Bin is the most passionate Delphi engineer I've ever met, bar none. While other peers have given up on Delphi and switched to other programming languages, You-Bin continues to work tirelessly in the Delphi community, exploring ways to make it successful. By combining frontend JavaScript with backend Delphi, he aims to create a "happily ever after" for Delphi's Snow White and Prince Charming.*

*Why did he choose JavaScript for the frontend instead of using Delphi? It's because of You-Bin's love for Delphi. He firmly believes that Delphi can do anything, so he chose a seemingly unrelated technology like JavaScript to complement Delphi.*

*This book is the culmination of You-Bin's years of hard work and dedication to Delphi. Along with technical specifications, it includes detailed descriptions of commonly used properties and methods of the components, making it easy for readers to use as a reference. So, let us wield the sword of Delphi and move forward bravely.*

*By Jason Wong, Senior Moderator at Delphi.KTOP.*

*Delphi has introduced the three-tier architecture technology of MIDAS since Delphi 5. With the evolution of several versions over time, DataSnap has become more efficient and versatile than the original MIDAS technology. It supports not only desktop applications, but also mobile devices and web applications, and provides transmission methods that are different from binary transmissions, such as transmission via HTTP protocol in JSON format. In today's world*

---

*where mobile devices and web applications are prevalent, Delphi is still an excellent development tool for desktop applications, although its usage rate may not be as high as it was in the 1990s.*

*In the field of information-related sciences, people often talk about Java and repeatedly use Python. Although software practitioners mostly come from universities, private schools, or IT training programs, they have various choices due to different interests and school preferences. However, Delphi is still the development tool for many applications in the industry, ranging from the tax software in Taiwan to the course selection and scheduling systems in schools. It is irreplaceable.*

*You-Bin's insistence is the indispensable spirit of Delphi in Taiwan today. From this book, I see You-Bin's dedication to DataSnap, which not only enables beginners to learn the essential spirit and implementation methods of Delphi but also provides many valuable insights. Delphi can be used to develop applications for both mobile and desktop devices. Although it may not be the fastest, it is the most versatile, convenient, and powerful. It covers everything from user interfaces, databases, network communication protocols to the control of various devices such as RS-232. There is nothing that cannot be achieved, only the time it takes.*

*Thanks to You-Bin, and I hope more readers can begin to learn Delphi through his book and achieve great success in their careers.*

*By author Ziren Zhang of "Indy TIdDNSServer, Delphi/Kylix - Indy  
Network Programming"*

*In recent years, microservice architecture has become popular and there are many frameworks available for the backend, such as Laravel and CodeIgniter for PHP, and .NET MVC for C#. Although Delphi is a language with a long history, it also proposed a corresponding solution*



---

*about a decade ago: DataSnap. This means that Delphi can not only be used to develop general apps, but also can implement a certain level of MVC backend microservices.*

*You-Bin has been continuously focusing on the research and practical development of DataSnap for the past decade, and has written and replied to many related questions on the Delphi KTop forum and personal blog. He is an expert in this field!*

*This book starts with front-end Javascript and gradually dives into the implementation of DataSnap on the back-end, finally connecting the front and back-ends. Such a layout is especially suitable for developers who are not familiar with the full stack, and we believe that they can quickly learn and apply it!*

*By Xiao Chong, Vice President at Delphi KTop.*

# Foreword

## I'm native, I'm proud

Delphi is the strongest Win32 application development tool on earth.

Developers who use Delphi surely agree with this statement. With the release of Delphi XE2 in 2012, Delphi entered the era of cross-platform development, with support for Windows 32/64bit, Macintosh, iPhone, Android, Linux, and more through the new generation FMX framework.

The philosophy of Delphi is to create native code-level applications.

Applications built with native code can ensure their performance and stability, which is the romance of Delphi!

However, using the FMX framework for small, limited cross-platform needs can be too cumbersome. This is where web applications come in, satisfying these requirements.

## Delphi faces impact of modern design practices

Currently, the mainstream language for the frontend of web applications is JavaScript, which is an interpreted language and conflicts with the philosophy of Delphi. Therefore, there is still no suitable frontend framework for the most widely used web applications. On the other hand, the backend of web applications is

---

Delphi's main battlefield. In 2009, Delphi 2009 introduced a new version of DataSnap and began to approach the mainstream of web development with RESTful APIs. With the release of the JavaScript DataSnap framework in Delphi XE, the key to connecting the frontend and backend, AJAX (Asynchronous JavaScript and XML), was established. This architecture has been maintained until the upcoming release of Delphi 10.4 in 2020.

## Problem Summary

The author has encountered a common situation where skilled and experienced Delphi professionals suddenly need to develop web applications, requiring them to delve into the three major areas of frontend development: HTML5, CSS, and JavaScript. Additionally, they must also consider the changing demands of the backend. Faced with a sudden increase in workload, many people give up on the idea of web development or seek solutions using third-party frameworks such as IntraWeb or UniGUI. However, they still cannot avoid the harsh reality of the three major frontend areas.

Adaptive Delphi professionals may leave Delphi and switch to ASP.NET, PHP, or NodeJS for web application development. Learning more is always a good thing, and it's important to avoid the misconception of "if all you have is a hammer, everything looks like a nail."

In the author's opinion, when faced with nails, what is needed is a hammer! Once again, no matter what backend architecture is used, the reality of the frontend must be faced. The frontend framework for Delphi is lacking, and there are so many frontend frameworks to choose from, including VanillaJS, jQuery, VueJS, ReactJS, and many more. How deep should one go to master frontend technology for

---

web application development? Some people even ask if their applications can still be compatible with Windows XP, which is still used in some businesses.

These are just some of the many questions that prompted the author to write this book.

## Technical Integration

This book adopts a gradual approach to develop and manage a complete full-stack web application, suitable for readers with a foundation in Delphi and in need of advanced integrated learning materials. The frontend technologies used include HTML, CSS, JavaScript, jQuery, and EasyUI, with efforts made to ensure compatibility down to IE7. The backend technologies used include Delphi, DataSnap, and Microsoft SQL Server, integrated to design a RESTful API application.

The first part provides an overview of DataSnap Server in combination with the EasyUI environment, allowing readers to experience the fun of building websites at high speed.

The second part introduces commonly used EasyUI components, providing readers with an understanding of them.

The third part explains the JavaScript DataSnap framework, which seamlessly integrates AJAX for smooth front-end and back-end connections.

The fourth part introduces database connection components, discussing the application of EMBT and DEVART dbExpress drivers and components.

---

The fifth part covers practical case development, including common methods, components, databases, JSON parsing, and performance tuning.

## Content Structure

The following software and minimum versions are used in this book:

- Backend and IDE: Delphi XE Enterprise or Architect
- Database: Microsoft SQL Server 2008 Express
- Frontend framework: jQuery EasyUI 1.3.2

If you have a need to be compatible with Internet Explorer 6, you can use EasyUI version 1.3.2 to achieve compatibility, while projects interested in following HTML5 standards can use the latest version of EasyUI directly. Although the content of this book is compatible with Delphi XE, for optimal learning results in using EasyUI components, it is recommended to use Delphi XE7 or higher versions.

## Achieving Better Results

Through the process of learning C++ Builder and transitioning to Delphi over the past decade, I have come to realize that Delphi emphasizes the use of a large number of control components to quickly complete development tasks. From the perspective of "completing projects," it can take as little as six months for a newcomer to become proficient enough to generate profits for a business. However, to gain a deeper understanding of Delphi's various extension frameworks, it may take one to several years. As a

---

result, a technical toolbox that can quickly navigate essential development knowledge is extremely important.

I hope that this book will serve as the first stepping stone for readers to step out of their comfort zones. If it can help you achieve greater success, I would be honored.

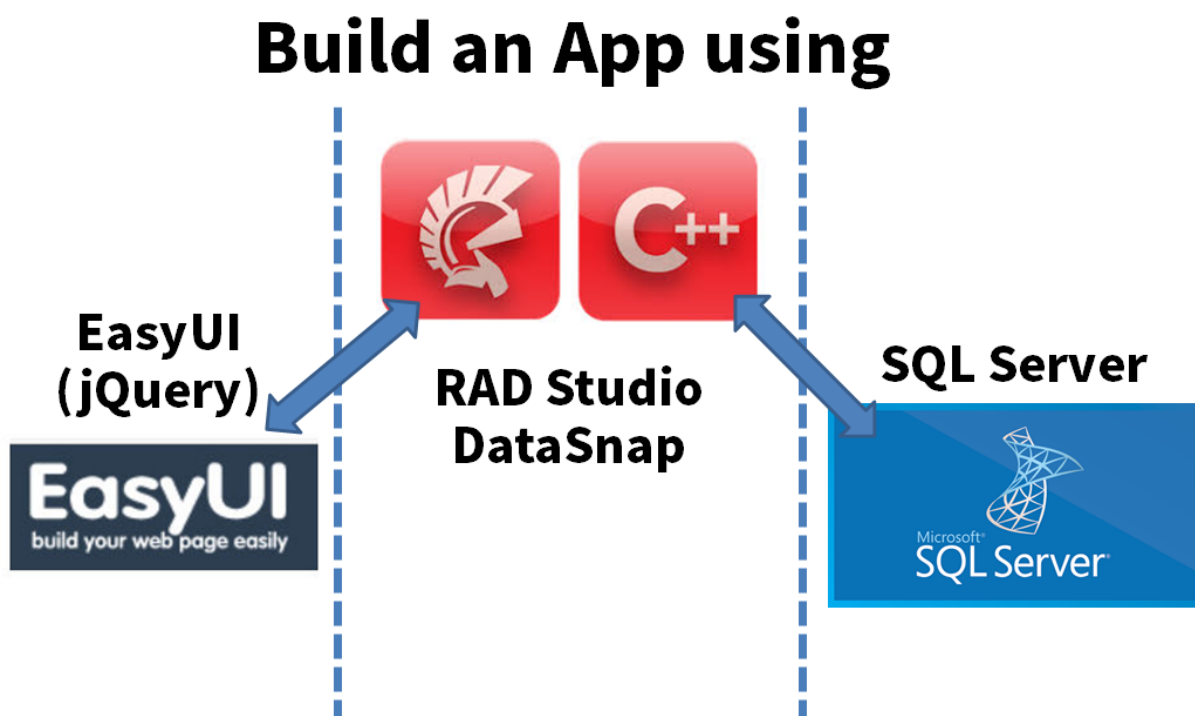
## Full Book Example Link

<https://github.com/Eden5Wu/DELPHI-IN-DEPTH-DATASNAP-EXAMPLE>

---

# Building Full-Stack Services from 0 to 1

Architecture diagram of the web  
application in this book



Getting to know the Delphi development  
tool

Since entering the Delphi world, I have been developing on the Win32 platform. In 2006, I began to encounter the term "Web". A client in the Kingdom of Swaziland had already requested Web support, and at the time, I began to learn about database connectivity technologies

---

such as BDE+Paradox, transitioning to dbExpress, and adding new knowledge about MIDAS. At this time, my understanding of the Web was that "cross-Internet" equals "cross-platform," which means "from computer A to computer B via the Internet" is considered cross-platform—Win32 to Win32. Just by copying the EXE file to a USB drive, it could be used anywhere in the world. This world was perfect!

Then the USB flash drive virus outbreak occurred.

The author, after being impacted by the widespread virus epidemic of USB drives, found that clients considered USB drives and EXE files to be bad things, and began to request that EXEs not be used. So the second form of the web, "web pages," left an impression on the author's knowledge circle.

Subsequently, the author began to study what HTML, CSS, and JavaScript were. Having been in the Delphi world for a long time, the author felt that the aforementioned three languages were like alien languages. The author then started looking for more convenient solutions, such as Classic ASP, PHP, ASP.NET, Flash, and SilverLight. After trying out all of these solutions, the author realized that none of them were as good as Delphi. Then the author turned to look for solutions combining Delphi and the web, such as InterWeb, UniGUI, and Raudus. These frameworks had the common advantage of "maintaining Delphi development habits" in order to create website applications. Upon delving deeper, the author discovered that there were more varied requirements, more in-depth implementations, and a need for unique features. In the end, the author still had to face

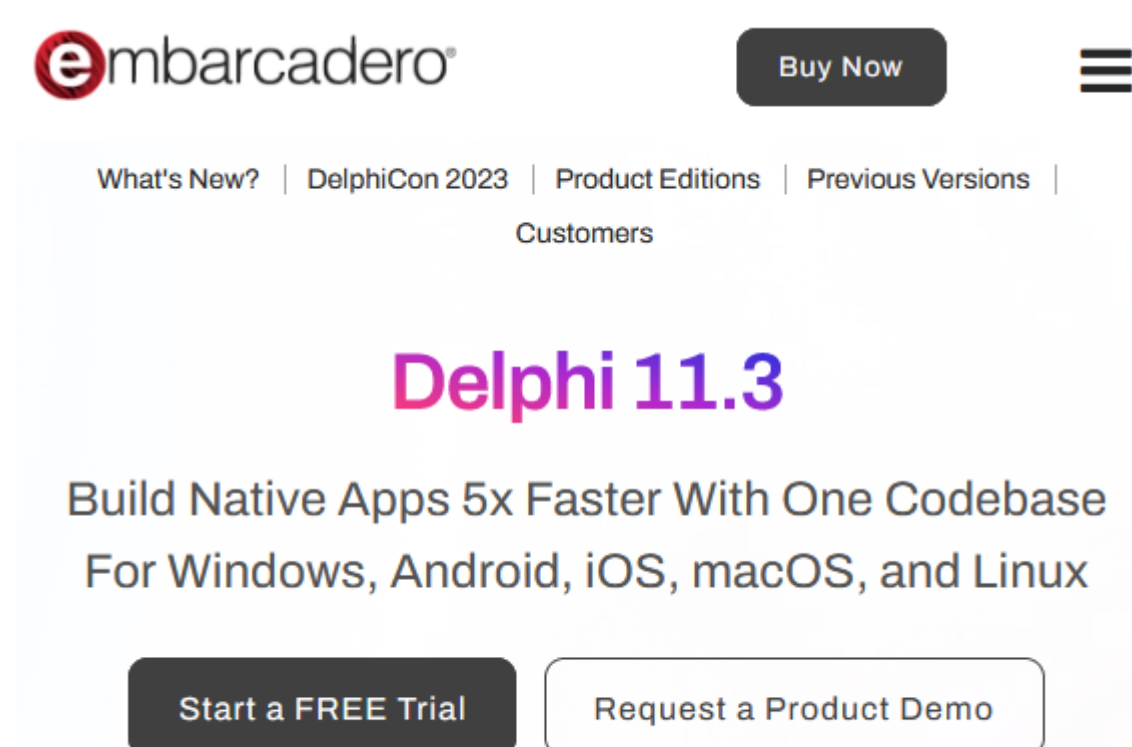


HTML standards (HTML, CSS, JavaScript), since these third-party frameworks were all based on HTML standards.

DataSnap framework released a new version in Delphi 2009 that embraced the JSON data interchange standard. JSON is also the JavaScript object notation, making this a new milestone for DataSnap. In 2011, Delphi XE introduced the JavaScript DataSnap framework, which allowed Delphi developers who wanted to achieve cross-platform goals on the web to gain access to the platform.

Delphi, as a language and development tool, is not only a cross-platform development tool, but also includes front-end web editing capabilities, which few people know about. We can download the latest version of Delphi through the official website:

<https://www.embarcadero.com/products/delphi>



The screenshot shows the top section of the Embarcadero website. At the top left is the Embarcadero logo. To its right is a dark button labeled "Buy Now". Further right is a hamburger menu icon. Below these is a navigation bar with links: "What's New?", "DelphiCon 2023", "Product Editions", "Previous Versions", and "Customers". The main heading is "Delphi 11.3" in a large, bold, purple font. Below this is a subheading: "Build Native Apps 5x Faster With One Codebase For Windows, Android, iOS, macOS, and Linux". At the bottom are two buttons: "Start a FREE Trial" (dark) and "Request a Product Demo" (light).

embarcadero® Buy Now

What's New? | DelphiCon 2023 | Product Editions | Previous Versions | Customers

## Delphi 11.3

Build Native Apps 5x Faster With One Codebase  
For Windows, Android, iOS, macOS, and Linux

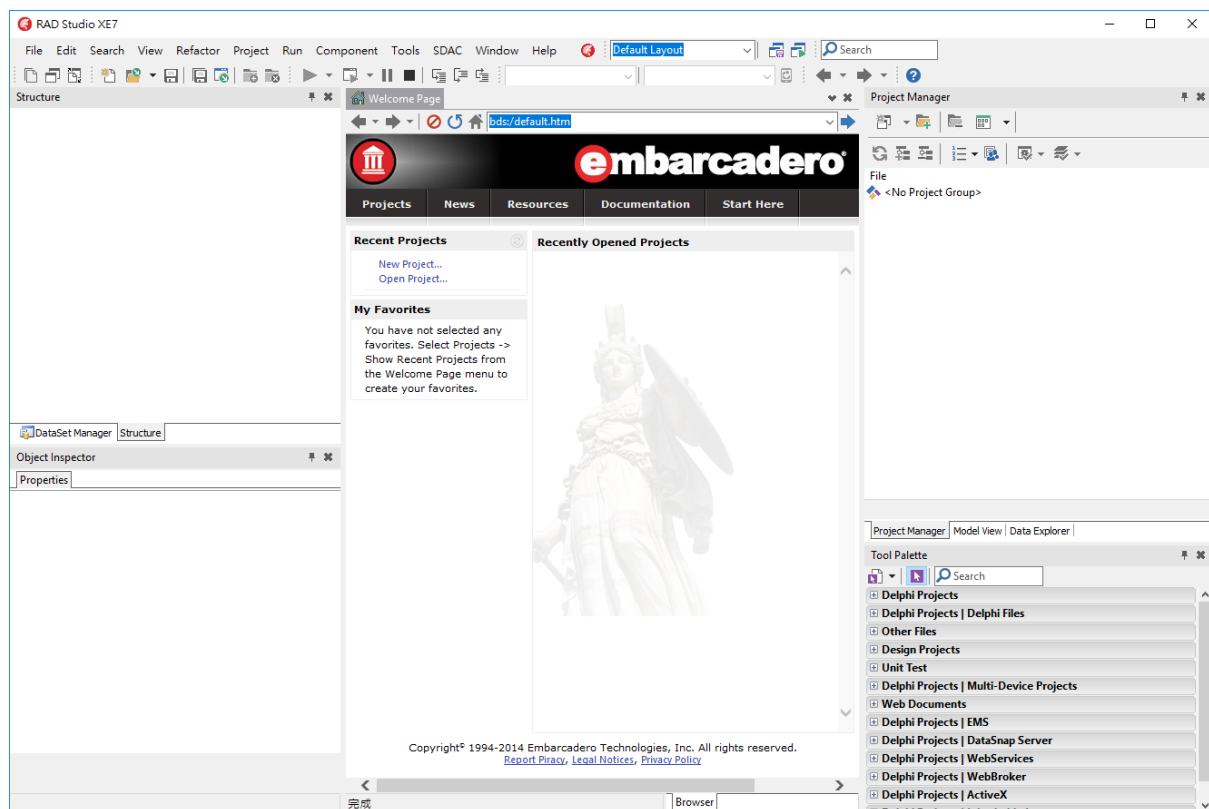
Start a FREE Trial Request a Product Demo

The trial version is valid for 30 days and allows you to experience the smoothness of Delphi's rapid development.

The programs in this book mainly use Delphi XE, which was released in 2011. The DataSnap framework was mature in this version, so the content of this book is compatible with the latest version (as of 2020, version 10.4). Please use it with confidence.

In the future, if "new version" is mentioned, it refers to versions after Delphi 10.2.

Below is the screen that appears when Delphi is opened:



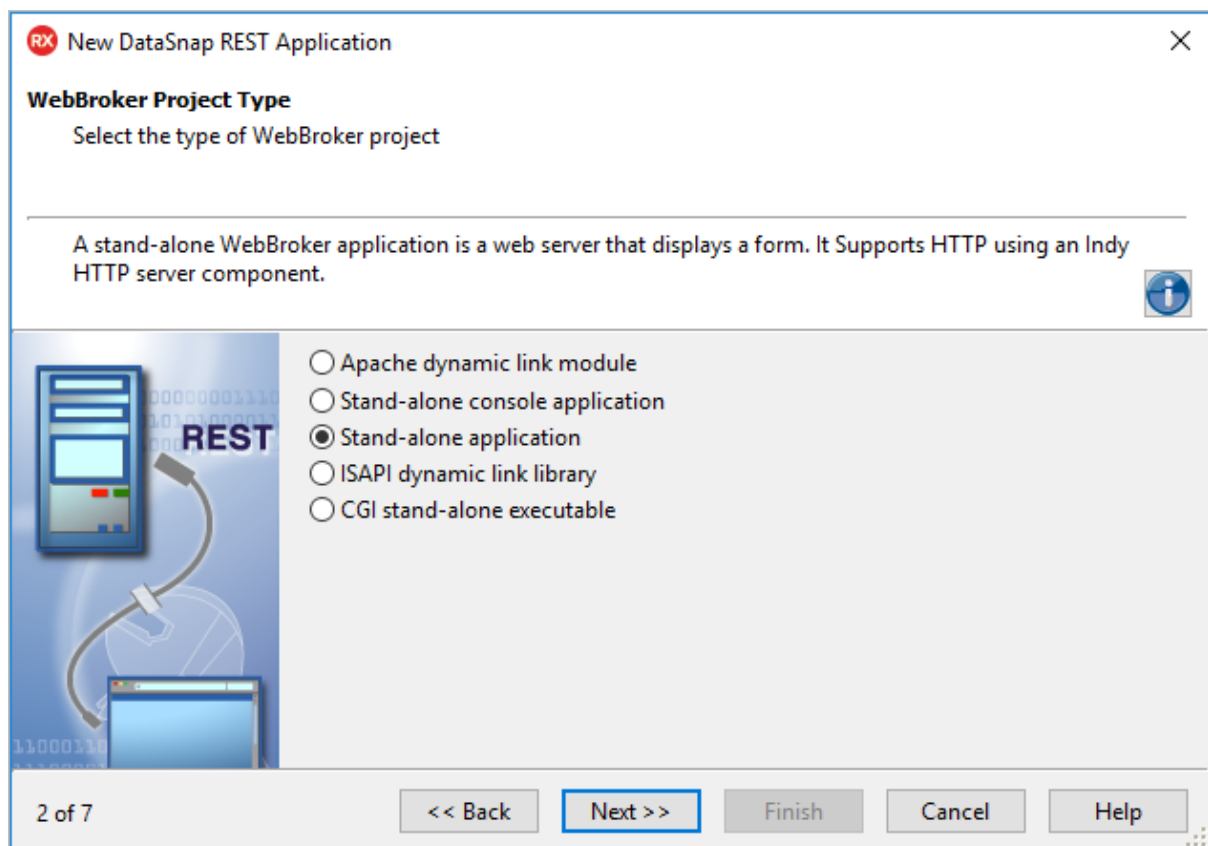
## Starting your first DataSnap project

After launching Delphi and accessing the main interface, select the "File" menu, then "New" and "Other..." to open the "New Items"

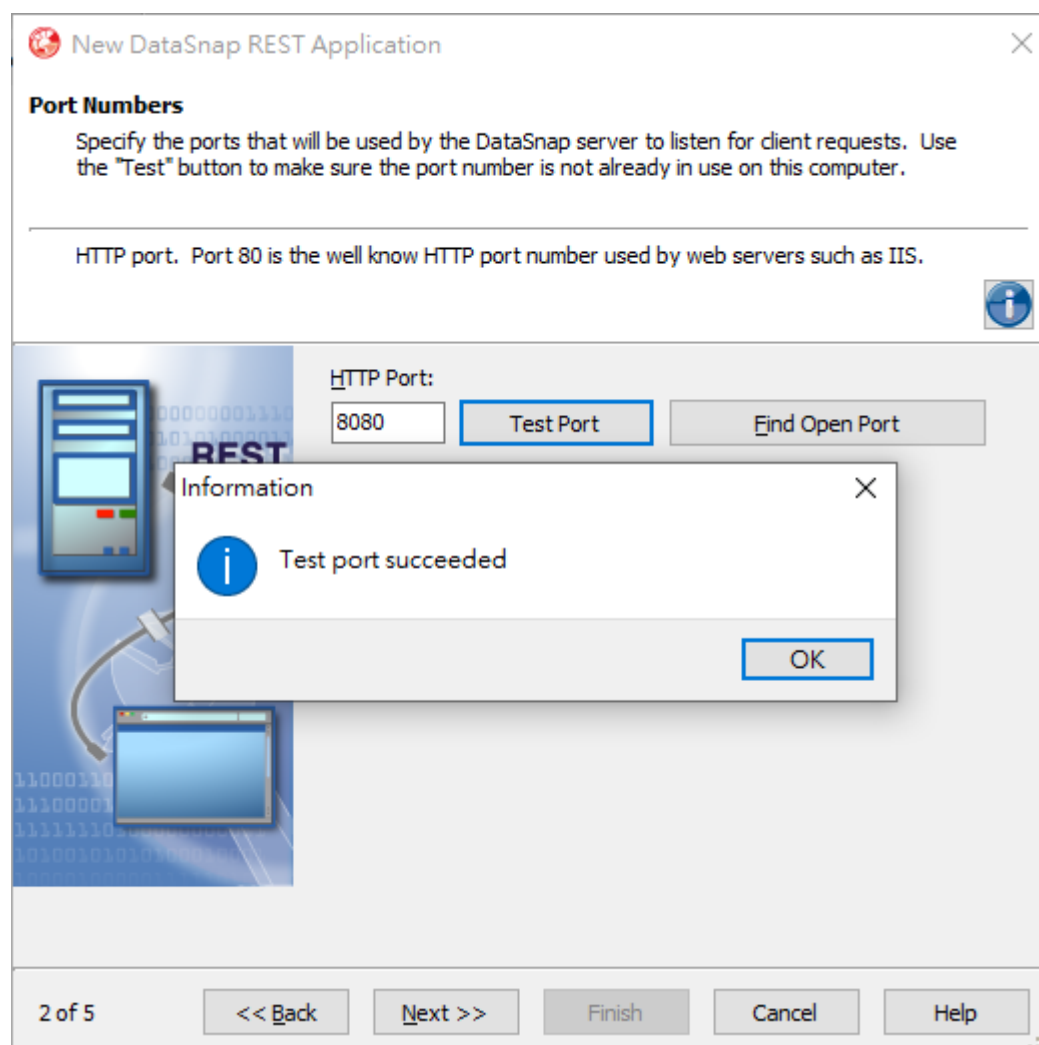
window. On the left-hand side of the window, select "Delphi Project" and then choose "DataSnap Server" and "DataSnap REST Application". After selecting, click "OK" to bring up the "New DataSnap REST Application" project wizard.

Starting from Delphi 10.2, there is a choice of Windows/Linux platforms, but for this case, only Windows needs to be selected.

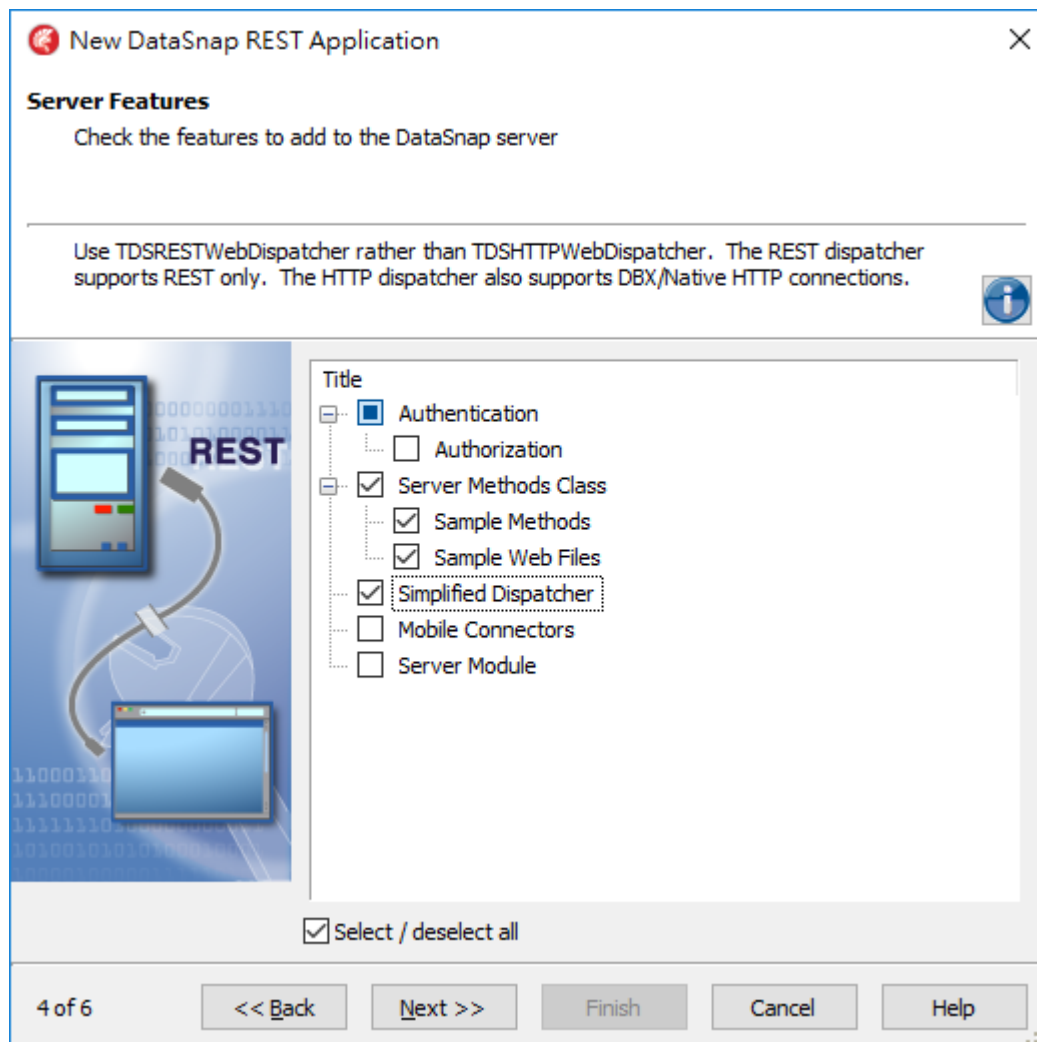
Select "Stand-alone application" on the WebBroker Project Type page.



In the Port Numbers page, the default HTTP Server Port is 8080. You can first click the "Test Port" button to test it. If it is not occupied by other services (as shown in the figure below), you can set it to the default value.



## Server Features Page



The optional features for DataSnap REST Application are selected based on the above image, and their descriptions are as follows: