

# What You Need to Use This Book

The following is the recommended system requirements for compiling and running SharpDevelop:

- ❑ Windows 2000 Professional or later
- ❑ The .NET Framework SDK (Freely available from Microsoft)

In addition, this book assumes the following knowledge:

- ❑ Sound knowledge of .NET Framework fundamentals
- ❑ A good understanding of the C# Language

## Summary of Contents

<b>Introduction</b>	<b>1</b>
<b>Chapter 1:</b> Features at a Glance	<b>7</b>
<b>Chapter 2:</b> Designing the Architecture	<b>23</b>
<b>Chapter 3:</b> Implementing the Core	<b>51</b>
<b>Chapter 4:</b> Building the Application with Add-ins	<b>81</b>
<b>Chapter 5:</b> Providing Functionality with Workspace Services	<b>107</b>
<b>Chapter 6:</b> The User Interface	<b>135</b>
<b>Chapter 7:</b> Internationalization	<b>169</b>
<b>Chapter 8:</b> Document Management	<b>189</b>
<b>Chapter 9:</b> Syntax Highlighting	<b>219</b>
<b>Chapter 10:</b> Search and Replace	<b>235</b>
<b>Chapter 11:</b> Writing the Editor Control	<b>263</b>
<b>Chapter 12:</b> Writing the Parser	<b>291</b>
<b>Chapter 13:</b> Code Completion and Method Insight	<b>329</b>
<b>Chapter 14:</b> Navigating Code with the Class Scout and the Assembly Scout	<b>369</b>
<b>Chapter 15:</b> The Designer Infrastructure	<b>413</b>
<b>Chapter 16:</b> Implementing a Windows Forms Designer	<b>437</b>
<b>Chapter 17:</b> Code Generation	<b>465</b>
<b>Index</b>	<b>499</b>

# Table of Contents

<b>Introduction</b>	<b>1</b>
<hr/>	
<b>The Code in This Book</b>	<b>2</b>
<b>Conventions</b>	<b>2</b>
<b>Customer Support</b>	<b>3</b>
How to Download the Sample Code for the Book	3
<b>Chapter 1: Features at a Glance</b>	<b>7</b>
<hr/>	
<b>The idea behind SharpDevelop</b>	<b>8</b>
<b>SharpDevelop Software Requirements</b>	<b>8</b>
<b>SharpDevelop's Panels and Views</b>	<b>9</b>
<b>Limitations</b>	<b>12</b>
<b>Customizing SharpDevelop</b>	<b>12</b>
Internationalization	12
Defining the Appearance	13
<b>Customizations for Coding</b>	<b>14</b>
Getting Started with Templates	14
Changing Syntax Highlighting	14
Code Completion and Method Insight	15
Bookmarks	16
Search and Replace Functionality	16
Compiling the Code	17
<b>Managing Projects</b>	<b>17</b>
Combines and Projects – What's in a Name?	17
Navigating the Project	17
The Project and Class Views of the World	18
<b>Creating Windows Forms</b>	<b>18</b>
Adding Controls to Windows Forms	19
Properties and Formatting of Controls	20
<b>Summary</b>	<b>20</b>

<b>Chapter 2: Designing the Architecture</b>	<b>23</b>
<b>History of Architectural Design Decisions</b>	<b>23</b>
The Early Stages	23
Building SharpDevelop with SharpDevelop	24
Correcting Bad Design Decisions	26
<b>The Design Decisions</b>	<b>27</b>
Designing Component-Exchangeability	29
<b>Best Practices</b>	<b>30</b>
Design Patterns	30
Singleton	31
Factory	31
Decorator	33
Strategy	35
Memento	36
Proxy	37
Coding Style Guideline	39
Defect Tracking and Testing	39
Bug Tracker	39
Unit Tests	40
Refactor Frequently	41
Design and Refactoring	42
<b>Summary</b>	<b>48</b>
<b>Chapter 3: Implementing the Core</b>	<b>51</b>
<b>The AddIn Tree</b>	<b>52</b>
Advantages of Using the AddIn Tree	53
The AddIn Tree Superstructure	56
Add-in Definition	57
From Tree Node to Running Object	60
Codon Creation	62
Conditions	64
<b>AddIn Management</b>	<b>66</b>
<b>Property Management in SharpDevelop</b>	<b>70</b>
The Idea behind the IXmlConvertible Interface	71
Overview of the IProperties Interface	71
The Default Implementation	72
Properties at Work	75
Property Persistence	76
<b>Summary</b>	<b>79</b>

<b>Chapter 4: Building the Application with Add-ins</b>	<b>81</b>
<b>Working with Codons</b>	<b>82</b>
The ICommand Interface	82
Making Menus Work	85
Codon Overview	93
<b>Wiring up Add-ins with Conditions</b>	<b>96</b>
Condition Structure	97
Defining Conditions	98
Overview of Available Conditions	100
<b>Summary</b>	<b>104</b>
<b>Chapter 5: Providing Functionality with Workspace Services</b>	<b>107</b>
<b>Implementation Considerations</b>	<b>107</b>
Requirements for Services	108
The ServiceManager	109
Defining Services	111
<b>Common Services at your Service</b>	<b>112</b>
File Utility Service	113
Property Service	119
Resource Service	119
String Parser Service	119
Ambience Service	121
Class Browser Icons Service	123
File Service	124
Project Service	126
Parser Service	131
Other Services	133
<b>Summary</b>	<b>133</b>
<b>Chapter 6: The User Interface</b>	<b>135</b>
<b>Display Management</b>	<b>136</b>
The Workbench Window	136
Views	137
Pads	139
Views and Pads Applied – An Integrated HTML Help Viewer	141
The HTML View	142
Navigating the Help File	148
Layout Managers	153
<b>The Current and Future Implementation</b>	<b>166</b>
<b>Summary</b>	<b>167</b>

<b>Chapter 7: Internationalization</b>	<b>169</b>
<b>Handling Internationalization in SharpDevelop</b>	<b>171</b>
Redrawing with Events	173
Accessing Resources	176
<b>Managing Translations</b>	<b>180</b>
The Translation Web Application	181
Localization Implications	182
Compiling to Resource Files	182
Generating Resource Files from XML	182
Generating Resource Files from the Database	184
<b>Summary</b>	<b>186</b>
<b>Chapter 8: Document Management</b>	<b>189</b>
<b>Text Representation</b>	<b>190</b>
Basic Sequence Data Structures	190
Arrays	190
Linked Lists	192
The Gap Buffer Approach	192
Theory of the Gap Buffer	193
The Gap Buffer in Practice	194
The Future – The Piece Table	197
Composite Sequence Data Structures	198
<b>Representing Lines</b>	<b>199</b>
<b>Caret and Selection Management</b>	<b>206</b>
<b>The Text Model</b>	<b>210</b>
<b>Putting It All Together</b>	<b>212</b>
<b>Summary</b>	<b>216</b>
<b>Chapter 9: Syntax Highlighting</b>	<b>219</b>
<b>Syntax Highlighting Definitions</b>	<b>220</b>
Increasing Ease of Use with XML	220
Implementing Syntax Highlighting	226
Validation of the Syntax Highlighting Definition File	226
The Sequence of Events from the Opening of a File to the Highlighting of the Text Editor Window	231
<b>Summary</b>	<b>233</b>

<b>Chapter 10: Search and Replace</b>	<b>235</b>
<b>The Search Strategy</b>	<b>235</b>
<b>Inside Search and Replace</b>	<b>249</b>
Basic Find and Replace Implementation	249
Using Algorithms	252
The Brute Force Algorithm	254
The Knuth-Morris-Pratt (KMP) Algorithm	256
Wildcard Search Strategy	257
<b>Summary</b>	<b>260</b>
<b>Chapter 11: Writing the Editor Control</b>	<b>263</b>
<b>Introduction to the Editor</b>	<b>263</b>
<b>The TextArea Control</b>	<b>265</b>
Theory	265
Getting Started	266
Event Handling	272
Updating	277
<b>The TextAreaPainter</b>	<b>279</b>
<b>Mouse Management</b>	<b>285</b>
<b>Folding</b>	<b>286</b>
<b>SharpPad</b>	<b>287</b>
<b>Summary</b>	<b>289</b>
<b>Chapter 12: Writing the Parser</b>	<b>291</b>
<b>Need for a Parser</b>	<b>291</b>
<b>Parser and Language Theory</b>	<b>292</b>
Basic Definitions and Theory	292
Grammar	292
Parsing	295
Technical Requirements	296
<b>The SharpDevelop Parser</b>	<b>297</b>
Design Decisions	297
Implementation	298
The Big Picture	298
Reflection Parsing for Assemblies	299
Persistent Parsing for the Framework Class Library	304
The C# Parser for the Editor	309
The Abstract Parser	316
Putting It All Together in the Parser Service	318
<b>Summary</b>	<b>326</b>

<b>Chapter 13: Code Completion and Method Insight</b>	<b>329</b>
<b>Resolving Matters</b>	<b>329</b>
<b>Code Completion</b>	<b>340</b>
<b>Method Insight</b>	<b>357</b>
<b>Summary</b>	<b>366</b>
<b>Chapter 14: Navigating Code with the Class Scout and the Assembly Scout</b>	<b>369</b>
<b>The Class Scout</b>	<b>369</b>
Implementing the Class Scout	370
<b>The Assembly Scout</b>	<b>383</b>
Browsing References with an Assembly Scout	383
Writing the Assembly Scout	385
<b>Summary</b>	<b>411</b>
<b>Chapter 15: The Designer Infrastructure</b>	<b>413</b>
<b>Forms Designer Overview</b>	<b>413</b>
Implementation Considerations	414
Design Overview	415
<b>The .NET Component Model</b>	<b>416</b>
The Service Interfaces	418
.NET Designers	419
The Root Designer	420
<b>Designer Host Implementation</b>	<b>421</b>
Designer Host Services	431
The ComponentChangeService	431
The Name Creation Service	432
The Design Panel	433
<b>Summary</b>	<b>434</b>

---

<b>Chapter 16: Implementing a Windows Forms Designer</b>	<b>437</b>
<b>Designer Services</b>	<b>437</b>
Toolbox Service	438
Menu Command Service	442
Executing the Standard Commands	446
Implementing a Key Event Handler	447
Selection Service	450
Additional Important Services	454
Designer Option Service	455
Dictionary Service	457
UI Service	457
Type Descriptor Filter Service	460
Unimplemented Services	462
<b>Summary</b>	<b>462</b>
<b>Chapter 17: Code Generation</b>	<b>465</b>
<b>Making Components Persistent</b>	<b>465</b>
The XML Forms Persistence Format	466
Generating the XML	467
Loading the XML	470
The Designer Serialization Service	474
<b>Generating C#/VB.NET Code</b>	<b>475</b>
<b>Round-tripping</b>	<b>484</b>
<b>Summary</b>	<b>496</b>
<b>Index</b>	<b>499</b>

---