



The Newbies Guide To Making Software

Resell Rights License Registration

WARNING: This is NOT a free book and you cannot sell or give it away to others unless you are an **AUTHORIZED DEALER!**

To get **resell rights** and become an **authorized** dealer, please click on the link below:

 [Click Here To Become An Authorized Reseller](#) 

You'll receive the **newest version** of this ebook and be notified of future products with resell rights.

Michael Rasmussen and Jason Tarasi

www.ResellRightsBlowout.com

Making Money with Custom Software

Making Money with Custom Software	2
You Can Make Money in the Software Business	4
Step 1: How to Get Started: The Idea.....	5
An Unlimited Marketplace – It’s Not Just Microsoft	5
Filling a Need – Or Defining One	5
Major Software Categories.....	6
Software for Individuals.....	6
Software for Businesses	7
Before You Start Developing.....	7
Step 2: How to Make Your Program – Or Get It Made For You	9
Major Development Languages and Tools.....	9
Java.....	10
C.....	10
C++.....	11
PHP.....	11
Visual Basic.....	11
Perl.....	12
C#.....	12
Python	12
JavaScript.....	13
Delphi	13
SAS	13
PL/SQL.....	13
VB.net.....	14
Lisp.....	14
COBOL.....	14
Ada.....	14
Pascal.....	14
ABAP.....	15
Cold Fusion	15
Fortran.....	15
Finding Programming Resources.....	15
Elance	16
Guru.com.....	16
Rentacoder	17

Scriptlance.....	17
Custom Software Development Companies.....	17
Step 3: Coding and Testing	19
Working with Programmers.....	19
The Importance of Testing	20
Step 4: Finding Your Target Markets Online	21
Upload.com/Download.com	21
Selling Your Software.....	22
Supporting Your Software	24
Creating Your Software “Brand”	26
Software Development FAQ	27
In simple English what are the basic steps for developing software?.....	27
Explain what Open Source means?	27
Do I need to pay some kind of ongoing license fees for commercial development products I use to make my software?	28
What do I need to worry about in terms of liability with my software?	28

You Can Make Money in the Software Business

We all know about Bill Gates – the richest man in the world. He made almost all of his money with software.

As you look at all the fortunes made in software in the last 20 years, you may have wondered how *you* could take advantage of that opportunity.

You can.

Maybe you hesitated because you don't know much about programming.

Or perhaps you haven't gone into the software business because you didn't know where to start, how to market your product, how to sell it.

This is all much easier today than ever before, and all you need is a good software concept, and to know how to get your program built and marketed. We can help.

The broadband revolution of the last few years has made designing, developing, delivering and supporting software easier than ever before – and cheaper, too, since you can skip a whole bunch of steps that used to be expensive (such as packaging, printing manuals, traditional advertising, etc).

We'll walk you through the basics, from concept to completion, helping you develop your product idea, have it made and tested, how and where to sell it, and how to work with the people you'll probably be working with, like programmers.

By the end of this report, you'll be ready to go start making money with software right away!

To start making money with software you need to do four things: Develop requirements (what you want to make and what it will do), code the requirements or have them coded for you, test and improve the code, and sell the software!

Let's think about each step now.

Step 1: How to Get Started: The Idea

The most important thing you need to have is a goal – in this case, what kind of software you want to develop.

An Unlimited Marketplace – It's Not Just Microsoft

There are tens of thousands of software products introduced each year by developers large and small. These run the gamut from every kind of personal entertainment product to utilities, to specialized business software of almost any kind, to applications designed to super-charge or unlock special features of other software or make them work better.

Actually not only is the majority of new software not developed by big companies like Microsoft, almost all real software innovation is from smaller firms, since they have the impetus and flexibility to be adventurous in their thinking and to move quickly without layers of management, funding, and outside stakeholders causing delays and offering opinions.

You may well not make billions with software, but software can be a fine supplement to many kinds of online businesses.

Filling a Need – Or Defining One

Chances are if you are thinking about developing some software products, you already have some idea of who you want to sell to.

While the possibilities are really only limited by your imagination, let's take a moment and think about *major* software buying audiences and *major* categories of software by type that it can make the most sense for a small independent developer to work on.

Major Software Categories

There are a lot of ways to categorize all the kinds of software out there but here are a few convenient, generally agreed-upon MAJOR categories. There are of course dozens, if not hundreds, of specialized categories within them:

- Audio-Visual Software – such as music and video management tools
- Business Software – such as CRM and ERP tools
- Design and Photo Software
- Desktop Enhancements
- Developer Tools – software for making software
- Drivers – software to make software or hardware work
- Home and Education Software
- Internet Software – for getting more out of the Web
- IS/IT Software – for managing enterprise systems
- Utilities – specialized software that typically does one thing such as spyware removal or registry management

Software for Individuals

Individuals buy a lot of software for fun, to keep their computers working well, to organize information, for fun and entertainment and for other reasons. The popular categories include:

- A/V Software – the popularity of digital music and video has created a revolution in this area in the last couple of years
- Design and Photo Software – while businesses use many of these products on the higher end, individuals buy the majority of simpler tools to manage graphics
- Desktop Enhancements – mostly geared towards individuals, they add fun and function to the computer experience
- Drivers – system-resident programs that make other programs and hardware work together, though probably too technical an area to experiment with for a novice development company
- Home and Education Software – by definition for the personal market
- Internet software
- Utilities – specialized small programs that perform a custom function. Among the more popular in recent years would be programs that deal with eliminating spyware and viruses; firewalls and other Internet security functions; organization of large groups of files like photos, music, video

clips; file conversion programs to allow users to use different file types with different programs; and much more.

Software for Businesses

Businesses buy a lot of software, to keep their computers working well, to organize information, to manage aspects of their businesses, for security, and many other reasons. With businesses, bear in mind that the transaction methods may be different and that you may need to consider more complex licensing rules, which we'll consider in the sections on selling and supporting your programs.

The popular categories include:

- Business Software – by definition, for businesses of course! We'll discuss some of the major sub-categories below.
- Design and Photo Software – higher-end tools
- Internet software
- IS/IT Software
- Utilities – specialized small programs that perform a custom function. For businesses, the most important utilities would have to do with keeping computers safe (such as firewall, encryption, anti-virus, anti-spyware kinds of programs) and specialized business functions, which will vary by the industry.

While all businesses are different, most businesses have many common business functions they need to perform, and today, there are software programs – and software opportunities – to address these areas, especially for the small and medium business marketplace.

Major business functions that benefit from software include applications, auction tools, finance (including spreadsheets, tax programs, and more), presentations, sales management, program and project management, legal and accounting, add-ins and custom extras for major business programs like MS Office, communication tools and much more.

Before You Start Developing

Even with the hints we'll share, making software can be fairly expensive. Before you get started, have in mind the following:

- What you want the software to do
- What you want the interface to look like
- What kinds of other products are on the market that are similar
- What you plan to spend to have the software developed or “made”
- When you plan to make the software available

Great...now that you know what kind of software you want to make and sell, let's start thinking about how to do that!

Step 2: How to Make Your Program – Or Get It Made For You

The most “vexing” thing for many would-be software sellers is: comprehending how to actually “make” the software. The good news is that in today’s world there are a number of good choices for development, and even better, there are thousands (maybe hundreds of thousands) of talented programmers who you can hire for very little money.

Quality programming is now within reach of everyone because of these “developments,” and you don’t even need to know how to program even slightly. Did you know that Leo Fender, who developed the first practical electric guitar, didn’t know how to play a note?

Major Development Languages and Tools

You don’t need to know what language is the most appropriate for your particular software and you certainly don’t need to know how to do any programming yourself, but you should be aware of the basic popular languages and software tools – if only so you can make intelligent choices when working with programmers and can “talk the talk” of programming even if you can’t, and don’t need to, walk the walk so to speak.

There are basically 20 major active programming languages in the world today. Not all of them are equally practical for every application but if you’re familiar with this list you’ll be familiar with almost every general concept of language selection.

While about a third of the following are *very unlikely* to be used in your development, we thought taking a couple of pages to touch on them would be time well spent. The “top 20” list below comes from a November 2005 market study of demand for programmers – meaning all of the following languages have active projects, whether new developments or maintenance.

In order of approximate popularity *by hiring of programmers*, and with a quick comment on typical uses for each, and the relative expense of development in each based on what a typical contract programmer might cost:

Java

Java is an object-oriented language developed by Sun Microsystems in the early 90s and intended as a replacement to C++.

Java is currently more popular than C/C++ mainly because of the percentage of new application development that is Web oriented, Java having been developed especially for online use. Anytime you see a page with “JSP” in its name that is a page being served via Java.

Java is one of four major contenders if you are planning on building online applications as opposed to desktop software or system utilities – the others being PHP and Cold Fusion, as well as for some things, Perl.

(Note that JavaScript has nothing to do with Java, just to confuse everyone a little more!)

Java programmers are readily available and there are more of them every day.

C

Developed in the early 70s by Bell Labs and consistently the most popular microcomputer language for the last 25 years or more, C is powerful, simple, and works well for both system software (for which it was originally designed) and custom application development.

There are some technical limits to C – such as lacking object support – that make other languages including C++ more flexible for certain kinds of applications.

C also “allows” programming that other more modern programming languages will stop when compiling, which means C will let bad routines get through to “final” versions of software if the programmer is not careful.

C programmers are extremely easy to find and accordingly not that expensive.

Thank You for previewing this eBook

You can read the full version of this eBook in different formats:

- HTML (Free /Available to everyone)
- PDF / TXT (Available to V.I.P. members. Free Standard members can access up to 5 PDF/TXT eBooks per month each month)
- Epub & Mobipocket (Exclusive to V.I.P. members)

To download this full book, simply select the format you desire below

