

**developer.\*****The Independent Magazine for Software Professionals**

## The Art of the Developer Resume

By Daniel Read

Resumes may seem like something of a mundane topic, but after spending the last few weeks wading through resumes from software developers, it is clear to me that most developers need help with their resumes. This impression is backed up by many past resume reading experiences. While I have come across very few truly awful resumes, the majority of the resumes I have read in the last week have been substandard. Only a few have been what I would call really well done.

Does that mean that the majority of the developers represented by these resumes are themselves substandard? Maybe so, but I would like to think not. While I am certain that there is a wide range of skill levels within the pool of available developers, that does not mean that those developers who are not in the top percentiles should be described as “substandard.” Not at all. Maybe not all of them are as sophisticated as I might be looking for, and maybe not all of them write the best looking code, but they will undoubtedly make some employer happy.

My theory, instead, is that, because the majority of these developers do not understand the art of the resume (and more specifically, the art of the developer resume), their resumes may not do them justice. In the paragraphs that follow, my aim is to reveal that art to you, and to give you my perspective as someone who has often been in the position of sifting through resumes and hiring developers.

Before getting into specifics, I would like to emphasize one thing: the goal of your resume is to land you an interview. Period. If your resume does not get you an interview for a job, then it has failed. Keep this thought in the front of your mind as you think about and work on your resume.

(Getting the interview is the first step in a four step process: getting the interview, excelling in the interview, deciding whether you want the job, and closing the deal—but I’ll save those last three steps for future essays.)

Put yourself in the shoes of the hiring manager, who is likely a senior developer (or formerly was one): she is sitting there at her desk, with a fairly good idea of what sort of developer she needs to hire (or maybe not), and in front of her is a thick stack of resumes. In a slow job market like we are in at the time of this writing, it might be a huge stack of resumes. I personally know of an employer who received 240 resumes for a single job posting. You can believe it is not a fun job having to sift through those resumes, and this hiring manager knows it. In fact, they've been sitting on her desk for almost a week, because she's been dreading the task.

This hiring manager is very busy, and only has time to perform a limited number of phone screenings and interviews, so it's a given that only a small percentage of these resumes is going to make it into the "keepers" pile. The ones that make the keepers pile get a phone call and an interview. The ones that don't go in the trash. Given this scenario, it is important to realize three things:

- First, this hiring manager desperately wants to find a sufficient number of stand-out resumes as soon as possible.
- Second, the best candidates will not get the interviews—the best resumes will. The ideal candidate's resume may very well end up in the trash.
- Third, since the majority of the developers out there are not good at writing resumes, if you can make yours shine, it will almost always make it into the keepers pile (assuming you are a fit for the position).

So how do you make sure your resume stands out? What follows are my guidelines for winning developer resumes. Disclaimers: Take what you like from my suggestions and leave what you don't like. Many people have very specific ideas about how a resume should look and what it should contain, and I certainly do not mean to suggest that a resume that does not follow my "formula" is automatically a bad resume. There is no one-size-fits-all formula. Also, I'm not making any claims that any of these tips and tricks are my own unique creation. My "formula" is culled from reading several books on job hunting and resume writing, reading hundreds of resumes, frequent tweaking and updating of my own resume, and helping friends with their resumes over the years.

## Length

In my opinion, the ideal resume length is two to three pages. If you have several years of experience, three pages is ideal. If you have less experience, then two pages is ideal. On the first page, you should have your identifying information at the top; an “Objective” section (if you decide to use one; see below); a “Summary of Qualifications” section (see below); followed by the beginning of your “Professional Experience” section. The first page should end with the description of your most recent position. Then the next most recent position would appear at the top of the second page, and so on.

If you’re having trouble filling out two pages, chances are you are not digging deep enough or going into enough detail in your “Professional Experience” section (see below). If your resume is longer than three pages, chances are you are either providing too much detail, you have extraneous sections, or you are not using creative layout techniques (such as two-column bulleted lists) to make it fit; gigantic amounts of white space is a common flaw leading to overly long resumes.

Beware of making your resume go beyond three pages. Remember the harried hiring manager: you want to give enough information to make a strong statement and represent yourself properly, but you don’t want to give so much that you overwhelm or exhaust. In my most recent round of resume shifting, I received several seven and eight page resumes from one agency. Seven and eight pages! I didn’t read past page three on a single one of them.

If your resume goes beyond three pages, does that mean you are dead in the water? Not necessarily. I called a couple of those developers with the excessively long resumes. I am talking here about the “ideal resume length.” Maybe you have a couple decades of experience, multiple certifications, a couple degrees, some honors and awards, and magazine and book publication to your credit. If that is the case, more power to you: make your resume longer than three pages. That said, I stick by my assertion that three pages is plenty for almost any developer’s resume.

## The “Objective” Section

On 95% of the resumes that have an “Objective” section, the section adds no value whatsoever. I know that all the resume books, and your high school guidance counselor, told you to put this section at the top of your resume, but I personally do not use an “Objective” section on my resume. Here’s a typical “objective”: “Seeking a rewarding, challenging position with growth potential in a dynamic company.” Well of course you are! Who isn’t?

Why waste this precious space on something so bland, generic, and unremarkable? Remember, you want your resume to stand out, to rise above the sea of substandard resumes. On my resume, I want the first thing that hiring manager reads to make a strong statement about me—to “hook” the reader, so to speak. The goal of my resume is to sell me, to land that interview. I only have a few moments of the reader’s attention.

I start my resume off with the heading, “Summary of Qualifications.” (I’ll go into more detail on this section below.) Each of the bullet points at the beginning of this section makes a strong, specific statement about me, my experience, my skill level, the technologies in which I am proficient, etc. In this way, not only am I making a statement about what kind of candidate I am, my objective is implied: I’m looking for a position that is going to match my skill level and experience, using the technologies in which I am an expert, and utilizing my leadership skills and experience. If you’ve got something within those parameters, let’s talk.

Does that mean I would never use an “Objective” section on my resume? Does that mean you should not have an “Objective” section on your resume? Not necessarily. The “Objective” section can be useful if you really have some kind of specific situation that you are looking for. Maybe you only want to work in a specific industry, or on a particular kind of project, or only using bleeding edge technology.

However, for my own resume, unless I really had a strong feeling about one of these criteria, I’d probably leave the “Objective” section off—especially in a slower job market, because I’d rather have the opportunity to turn a position down than never know it might have been offered to me because of my narrowly stated objective.

## **Executive Summary**

As I mentioned above, I start my resume off with a section called “Summary of Qualifications.” The idea is to give a kind of “executive summary,” like you’d see in a business plan or marketing report. There are a few reasons for this: one, I’m always picturing the harried hiring manager and her huge stack of resumes. I want to grab her attention as quickly as possible. Two, I know from my studies of marketing and journalism that many (if not most) people will never read past the first few sentences or paragraphs of anything if they don’t have to.

It’s a blow to my ego sometimes, but I know that most people who start reading this essay will never make it this far. The same goes for my resume—maybe even doubly so. For this large percentage of people, I need to get my point across quickly. So there’s reason number three: even for the people who do read the rest of the resume thoroughly, I want to

sell them on the idea that I might be their ideal candidate right away, so that they spend the rest of their time with my resume seeking support and justification for that conclusion. In other words, I make bold claims in my “Summary of Qualifications” section, and then I make sure the information in my “Professional Experience” section (see below) supports those claims.

Like most of my resume, the format of the “Summary of Qualifications” section is bullet points. I like bullet points because they are easy to read, they don’t look daunting on the page like a thick paragraph does, and they appeal to people’s short attention spans. I start the section off with more general points, and then get more specific. As I mentioned, I like to make bold statements that “sell” me to the reader. Here are a few examples from my resume (in no particular order):

Daniel Read is a well-rounded professional software developer with over eleven years experience developing and maintaining a wide variety of systems for large international corporations and small businesses. Daniel’s specialties are working with business experts, designing technical and non-technical solutions to meet the needs of the business, crafting and implementing development processes, and leading teams to deliver software. He is also a skilled programmer.

From there I start getting more specific. You should, of course, tailor the “boldness” of these kinds of statements to match your experience and qualifications. Don’t make claims you can’t back up. Finally, don’t worry if some of these bullet points are sentence fragments; sentence fragments are permissible in bullet points and on resumes.

I end the “Summary of Qualifications” section with a two-column bulleted list of specific technologies, languages, and tools with which I am proficient. By this point, I have used up at least the first half of the first page.

## **Professional Experience**

This is where most developers need the most help on their resumes. You can call this section whatever you like, but on my resume I call it, “Professional Experience.” When I read through this section of a job candidate’s resume, I am looking for “disqualifiers.” In other words, I am looking for reasons not to put this resume in the keepers pile. Remember that as the hiring manager I have a huge stack of resumes, and if I don’t use some kind of criteria to trim them down, I’ll never get to go home.

What has this developer really done? Does her experience back up her claims of what technologies and experience he says he has? Has he worked on similar projects to what I am looking for? Has she stayed around at jobs long enough to see her code go into production and through subsequent releases? Does he jump around jobs too frequently? Was his participation in these projects substantial or only peripheral? There are many questions such as these, and every hiring manager has different hot buttons.

Knowing this, and knowing I cannot hope to please everyone or be the ideal candidate for every job, my goal on my own resume is to simply be as specific as possible (within reason, of course) and back up the claims I made in my “Summary of Qualifications” section. My goal is also to continue to “sell.” What I am selling are my abilities (which are proven by the technologies I have worked on, the types of software I have built) and the positive effect I have had wherever I have been. This latter part is what too many developers miss.

Putting it in marketing terms, it’s about selling the benefits. When advertisers promote a car or box of laundry detergent, they don’t just list the facts about the product—they present the product so that the consumer imagines the benefits of owning the product. On your resume you don’t want to just say, “I wrote code.” You want to say, “I wrote code, which had the lowest defect rate on the team, resulting in a shortened testing cycle.” Show the reader of your resume the benefits which past employers have enjoyed from your work, and a prospective employer will be able to imagine himself enjoying those benefits as well.

Going back to formatting, what I do is make sure that the second half of the first page of my resume contains the entirety of my most recent position. Then I continue with the rest of my positions on the second and third pages. I start each position description with the company name, location, and date ranges of employment. Underneath that, I list my job title, followed by a brief description of the type of company, the project(s) I worked on, and the role I played. This description should not exceed two sentences.

This is a big mistake that I see on many resumes. Too many people will spend a paragraph or two describing the project and the company, and then barely spend any time at all getting specific about what they did. These developers will go into detail about what type of company it was, what division they worked in, what the purpose of the software was, what technologies were used (whether or not they personally had anything to do with those technologies or not), how many team members there were, etc., etc. Is some of this information relevant? Probably—but you have to keep this sort of thing to a bare minimum. Your resume’s job is to sell you, to land you an interview.

When I encounter this technique on a resume, I am left to conclude that this developer didn't really do all that much on this project, and is therefore trying to make it look like he worked on a lot of stuff he really didn't by describing a lot of stuff he was simply associated with. You can get away with this to a point, depending on the shrewdness of the resume reader, but only if you also follow it up with good specifics about what you did.

Another thought regarding this problem: I see this technique often on the resumes of people from other countries besides the United States. I don't have any facts to prove this, but my sense is that perhaps in some cultures it is considered unseemly to draw too much attention to oneself; in other words, you worked as part of a team, and it would be disrespectful to the team to shine a spotlight on yourself apart from the team.

If this is indeed a cultural difference between the US and other countries, then I absolutely respect that. If you live and work in a country besides the US, then maybe none of this resume advice of mine applies. I can't say, because my view is totally provincial to the US job market. That said, if you are from another country or culture, and are seeking a job in the US, then you cannot afford on your resume to be shy. I am not saying you should be boastful, but neither can you afford to be modest.

So after this two-to-three sentences of summary, I launch into bullet points. These bullet points are intended to describe what specific things I did for the company and for the project, what technologies I worked with, and whenever possible, the benefits that my efforts had for the team and for the company. Here are a few random examples from my resume:

- “Designed a large web-based e-commerce application from scratch, including requirements, database design, object/component design, security design, design of logical and physical tiers, and user interface design. Wrote a 100+ page design document for the entire system.”
- “Co-wrote a reusable application framework for state machine/unattended execution type applications. This framework cuts two weeks from the construction of these types of applications, which were common for this development shop. Also wrote additional applications using the framework.”
- “Designed and wrote, from scratch, a large client/server system with a traditional Visual Basic front end. This system was very successful for the company's largest client, and consists of thousands of lines of code in over 50 files, plus dozens of stored procedures and triggers. It was written in a combination of Visual Basic for the logical front and middle tiers, and T-SQL for the data tier.”

- “Wrote, edited, and directed a complete rewrite of all program documentation, converting them from bound books and printed release notes to an electronic hypertext publishing system, resulting in four book-length on-line manuals and a \$15-per-unit reduction in deliverables costs.”

Here are some tips for writing bullet points for your “Professional Experience” section:

- Start each bullet with a strong action verb: wrote, designed, researched, compiled, supported, built, led, coordinated, initiated, conceived, etc. Try to avoid using the same verbs over and over. Imagine the word “I” in front of each bullet: I conceived, I wrote, I built, etc.—only leave out the actual word “I” when you write it.
- Use the active voice. You do not want to sound wishy-washy or passive. Here is passive voice: The system was written in C. And here is active voice: I wrote the system in C. Your resume is selling you, and the passive voice takes you out of the picture.
- Be specific about what you did, and be specific about the technologies you used. Don’t try to be clever by mentioning a lot of technologies that other people worked on, because you will get busted in the interview.
- Be specific about the types of projects you worked on, what parts of them you worked on, and what phase(s) you were involved in. Did you work on GUI elements, the middle tier(s), or the database? Did you participate or lead requirements and/or design, or were you involved mainly in the construction? Did you spec out the hardware? Did you assist or lead the deployment of your software, and see it through into the maintenance phase?
- Sell success. Don’t be shy about pointing out the benefits that the team and company accrued from your efforts. Did your project have a successful rollout with a very low bug count? Did the software you worked on make a lot of money for the company? Did the improvements you made to the software result in reductions to the help desk staff? Sell that stuff.
- When selling the benefits of your work, use numbers where possible. Managers, especially, love numbers. Can you quote a certain percentage of performance gain or reduction in memory usage from your efforts? Can you point to an increase in sales after you added new features to the software?
- Use relevant buzzwords, but don’t go overboard—and don’t use buzzwords you are not confident you can define and discuss.

- Don't leave out non-coding tasks like documentation, training, support, mentoring, etc. These things demonstrate that you are a well rounded individual. When reading developer resumes, I am always looking for this.

As you go from position to position on your resume, if you are having trouble keeping within the two or three page limit, you may need to start progressively shortening the section for each position, even getting really old positions down to one bullet point. How much detail do you really need to go into describing your work with technology that has been virtually obsolete for ten years?

Finally, if you have part of your work experience in non-computer-related fields, and do not feel that it is relevant, then just list that period of time as "Non IT-Related Employment" and put the date range. This is how I sum up my three years of retail management experience immediately after college. The harried hiring manager does not care that right out of college I spent a year and a half at a department store building shelf displays in the middle of the night.

## Other Sections

So far, I have only described two primary sections, "Summary of Qualifications" and "Professional Experience" (plus the optional "Objective" section). In fact, these two primary sections are the only sections on my resume. What about some other traditional resume sections? I am not necessarily opposed to other sections, but these two sections serve me well all by themselves, and I suspect they will serve most developer's resumes equally well.

Some other sections you might be wondering about:

**Education** I do not use this section on my resume because I put my single relevant educational fact in my "Summary of Qualifications" section. I have a Bachelor of Arts degree in Creative Writing, and have never felt that this one modest fact is enough to warrant a whole section on my resume. Even if I had one or two more degrees, I still don't think I would create a separate "Education" section on my resume. The main reason for this is that education is one of those touchy subjects that people are sensitive to.

Many people are insecure about their education level, and this seems to be equally true of people who never went to college as it is of people who have a two Bachelors degrees, a Ph.D., and an MBA. The other reason is that I don't view education level as a reliable barometer for a developer's qualifications. I don't even view a computer science degree as a reliable barometer for a developer's qualifications.

Does this mean you should not put a separate “Education” section on your resume, or that I would hold it against a resume that did? Absolutely not. I just don’t like to waste the space on my own resume.

**Training and Certification** This is another gray area. Some people might lump training and certification into the “Education” section, and others like to break it out into its own section. I don’t do either. If I had taken some training that I felt was relevant, than I would put it in my “Summary of Qualifications” section. I do the same with certifications. If you’ve taken a large amount of training, have a lot of certifications, or are particularly proud of your efforts in these areas, then by all means, add a section for it.

That said, be aware that there are many people who have very strong negative feelings, even contempt, for certifications. These people are usually fairly irrational about it, in my experience, and I would not advocate hiding your hard-earned certifications from these narrow minded people. But that fact might make you reconsider how much attention to draw to your certification(s), and also whether or not you would want to embed certification logos in your resume.

**Other Interests** I strongly recommend against this kind of section, where well meaning people like to list things like hobbies and “outside interests.” Some people like to also create “Professional Affiliation” sections in which to list memberships in clubs and associations. The reason I recommend against these kind of sections is that I believe in keeping anything potentially controversial in the background, or preferably off the resume altogether. Remember that harried hiring managers are looking for disqualifiers on your resume—don’t hand them over on a silver platter.

Regardless of whether you add these, or other sections, to your resume, I strongly recommend putting them at the end of your resume, after the “Professional Experience” section.

## Layout

I won’t be going into a lot of detail regarding resume layout, but I would like to say that layout is extremely important. Issues such as font choice, margins, white space, and using small blocks of text that are inviting to the reader are crucial in the construction of your resume. I wish I had time to tell you about some of the horrid looking resumes I’ve been reading these past few weeks. If desktop publishing, layout, and word processor magic are not your strengths, then by all means get the help of a friend who is good in these areas. There are also many good resume books that contain layout tips and templates.

The bottom line is this: your resume represents you. When I see a developer's resume that is sloppy and unbalanced, then I have to assume that his code will look that way too.

## Agency Mutilation

This is a huge pet peeve of mine. Contracting and placement agencies have this nice habit of taking the perfect resume, the one you spent hours working on, and then making a real mess of it. It's shameful. I understand that agencies want to cover up the personal contact info at the top of your resume and that they want to put their logo at the top. I have no problem with that, which is why I leave room at the top of my resume for the agency info.

But too often (most of the time, really) the agency people who are changing these resumes a) don't have a clue what they are doing, and b) don't give a damn about how good the layout on your resume looks. Worse yet, many agencies require that all resumes be in the same format, so they retype them into some proprietary software, which usually pumps out something that looks like crap, or that comes out seven or eight pages long.

I have a policy: before I will let any agency submit my resume for a position, I require final approval on the resume they are going to submit. This is important enough to me to insist on it if a polite request is denied. It sounds harsh, but if an agency won't work with me on this, then they don't get to represent me. If they need a copy of my resume without my personal info and with their logo added, then they can send me the logo graphic, and I'll be happy to make them one. I have been burned too many times on this: I show up for the interview, get a look at the mess of a resume the agency sent over, and I end up apologizing for the way it looks. Maybe I'm being silly, but the way I look at it is that until a potential employer meets me and gets to know me and my work, the resume is me and my work.

## Footers

An often overlooked tidbit: when you have a multi-page resume, make sure you put a footer on all of the pages after the first one. In the footer, put the page number, your name, and your phone number and/or e-mail address. The reason for this is that pages get lost and separated and mixed up. Beware of agency mutilation of your footers (see above).

## What To Avoid

Quickly, here are some things to avoid on your resume:

- Don't stretch or otherwise misrepresent the truth.
- Don't oversell.
- Don't be trivial. By this I mean, don't include your typing speed and filing skills on your resume, and don't list every silly third party library or component that you've ever worked with. Trivialities expose or suggest lack of experience.
- Don't be negative. This tip applies to the whole job seeking process. Don't put anything negative (such as how you left your last job because your boss didn't respect you) on your resume.
- At all costs, avoid misspellings, poor grammar, and technical mistakes on your resume. Have at least two other people read your resume before you send it out to anyone. No disrespect intended to my beloved international readers, but if English is not your first language and you are applying for a job in the US, I strongly recommend getting a native English speaker with good writing skills to proofread your resume.

## Final Notes

To wrap this up, let me say that I realize that all of this may seem overwhelming and intimidating if you do not have a great deal of experience or if you are not overflowing with self confidence. If you are lacking experience, do not let that discourage you. Everyone starts out with a lack of experience. Put extra effort into your resume and demonstrate that you are a person who cares about quality, is willing to put in extra effort, has a good attitude, and has a strong interest in software development as a craft and discipline.

If you lack experience and are not doing a lot of reading and a lot of self practice, then you are not helping yourself. You of course need to be reading technical materials about your chosen tools, but you also need to be doing a lot of non-technology-specific general software development reading. There are plenty of companies and teams looking to hire less experienced people who have the right attitude, who show a lot of initiative, and who demonstrate dedication to quality. I myself have hired less experienced developers many times. Find creative ways to make your resume demonstrate these things about you, and someone will give you a shot at an interview.

As for lack of self confidence, the best method I know for overcoming that is to fake it. The first place to start faking it is in your resume. Your resume cannot reveal a lack of confidence. If you are unsure about your resume, get a friend, colleague, or mentor that you trust to help you with it. A strong resume that you are proud of is a great way to make sure that you can walk into an interview feeling confident. Remember where this essay started: most of the resumes you are competing with are crap, and now you know some techniques for making sure yours will stand out. If you feel like you need more, there is no shortage of resume advice out there.

###

Daniel Read is editor and publisher of **developer.\***. He lives in Atlanta, GA, where he works as a software architect and programmer. He is currently at work on about a million different things.