



Software engineering careers

Steve Chenoweth
Computer Science & Software Engineering,
RHIT

This file is out on your Moodle web page

The idea of a career goes beyond just “starting salary.” Instead, we ask, what’s the whole picture?

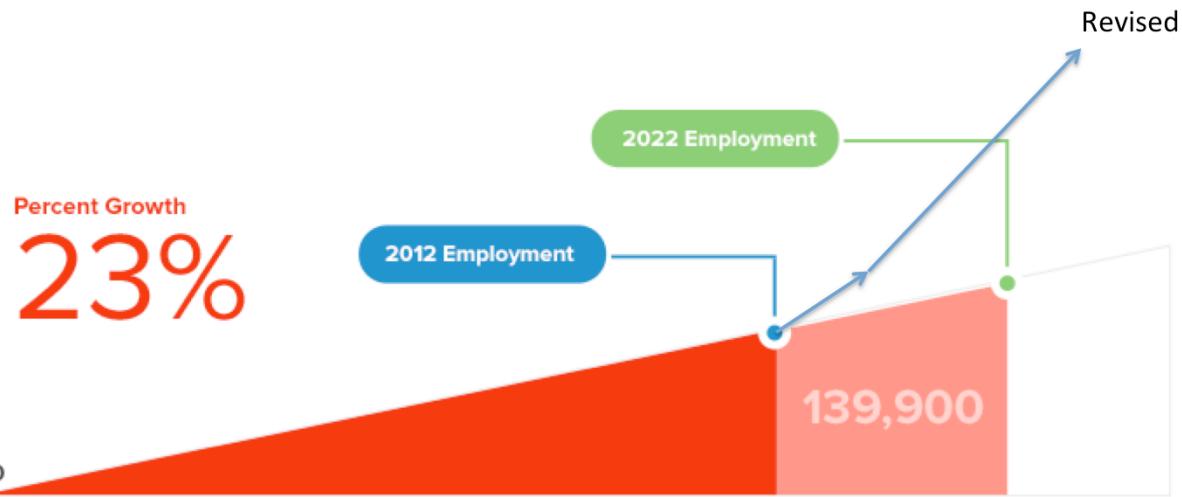
Will the people shown in the picture have a rewarding life, with rich, challenging work?

You also can ask, “How much will they make?” And, in fact, in software they do make good money. We show that on some of these slides. However, it doesn’t really take a lot to make people “happy.” Economists Daniel Kahneman and Angus Deaton showed that in a 2015 study. If you make what many engineers in the US make – all kinds of engineers – you are over the threshold where this makes a big difference. (See <http://time.com/money/4070041/angus-deaton-nobel-winner-money-happiness/>).

So, we’ll also talk about other values that are important, which might make software engineering an interesting and enriching career.

In this discussion, majoring in both Computer Science, and also in Software Engineering itself, can be seen as paths toward this career. In a few spots we clarify some of the differences in these majors.

Will there be jobs?



The US Bureau of Labor Statistics now has the job growth for 2016-2026 as 30.7% (255,000 jobs)

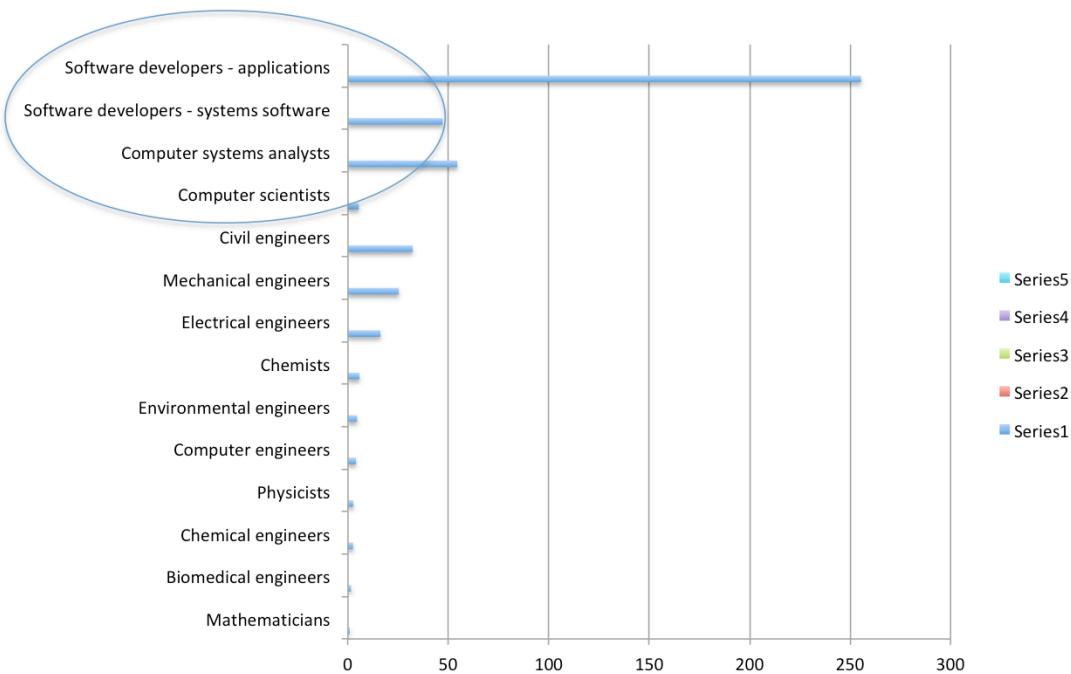
It is important to know if you are likely to have job opportunities to consider, over your career.

This is hard to predict, for any kind of work. However, much of the general threat comes from automation itself. And software people are the ones who automate many things we do – we sell software systems which simplify and speed-up tasks for humans. So, in a way, perhaps we should feel guilty for doing this, though it certainly falls within the bounds of how an efficient economy is supposed to operate. And, at the same time, we may be the least likely group to lose our own jobs, because businesses understand the advantages of the efficiencies we provide.

There really is a serious long-term shortage of software people in the US and elsewhere. The supply of qualified people generated by universities, or homegrown through experience, or created by moving people from other areas into software, simply isn't enough to meet the demand for people building software in the US. This supply problem surely is exacerbated by new government policies intended to cut all kinds of immigration. The result is that the growth is slowed, of profitable software that could be developed here.

The problem also is made worse by our historic inability to attract women and minorities into this field, in the US. That inequity also maintains the concern that software-based products and services mismatch the needs of users, because they are developed by a different group of people. This is true for example in social media, where over half the audience is women, but only a small percentage of the developers are women.

STEM career comparisons



Total projected 2016-2026 growth in the 4 software-based jobs in the US, shown at top: 362.3 K.

Total projected 2016-2026 growth in all the rest of the STEM-based jobs in the US: 95.4 K.

Source: <https://data.bls.gov/projections/occupationProj>

What will you do?

Google software engineering career choices

All Images Videos News Shopping More Settings Tools

About 2,840,000 results (0.52 seconds)

Salaries in Software Engineering

- Systems programmer. \$100,063.
- **Software engineer.** \$97,035.
- **Software developer.** \$88,335.
- Mobile specialist/ technologist. \$87,087.
- Application **developer.** \$86,162.
- Programmer/analyst. \$73,463.
- Web **developer.** \$68,087.

Percent Growth: 23%
2020 Employment: 139,900
2021 Employment: 163,000

[Software Engineering Careers | ComputerScienceOnline.org](https://www.computerscienceonline.org/software-engineering/)
<https://www.computerscienceonline.org/software-engineering/>

Consider the top job areas, within software engineering, that Google lists here.

The variety of software work is the most important thing to see here, though Google lists the hiring salaries, as well.

Systems programmers, for example, write software that is used by other programmers. While most of these other groups write software for people who are not programmers. See Slide 7 for more details about the kinds of work.

Typical activities

- Talk to customers about what they want
- Design systems
- Write and test code – on teams
- Test the whole system
- Deliver it to customers
- Lead other developers
- Improve how software is developed

"This is exactly what I wanted..."



Contrary to popular opinion, software engineers do not just write code. They do all these other activities, as equally valuable parts of the work.

Talking to customers, like you see pictured here, is crucial. If you get it wrong, as to what a major customer wants, you can imagine it doesn't work out as well as the caption I gave to the picture. Gathering and managing customer requirements is an expert part of software work.

The last bullet is key – software engineers help set their own destiny, by deciding how they will go about it. This is part science and part engineering – things like developing best practices. The fact we play a role in deciding the processes we use is a part of being Agile – something discussed in the next slide.

Image from <https://dissolve.com/video/Friendly-bank-workers-talking-customers-offering-financial-royalty-free-stock-video-footage/001-D84-120-050>

Growing trend -- The teams are “Agile”

What's that mean?

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

➤ In other words –
“empowered” teams vs
“managed” teams

Right – An Agile team decides how they are doing, using PostIt notes.



The fact that software engineers are in high demand has led to our having a lot of leverage in the workplace. In many or even most US software development organizations, the “team” of developers has a lot of power to decide the right way to do things, and whether or not they are doing a good job, and the specific processes they need to be using to be successful.

This same “empowerment” of teams does not exist in all engineering organizations, though it seems to be shifting in this direction, based on the successes of “Agile” in software development. In some areas of engineering, Agile is similar to the idea of Lean product development, which also is growing in popularity.

You can see that a goal of Agile, in the meaning described above, is to cut out unnecessary steps in getting the job done. Agile also tries to avoid problems which have plagued software development in the past, like having a fixed contract with customers, every bit of which must be satisfied in a predetermined timeframe.

All these changes help make the life of the software engineer much better!

Is it rewarding?

In 2014, [U.S. News & World Report](#) listed software developer as the best job in the country and [Career Cast](#) ranked it as the 7th best career field.

From <https://www.computer scienceonline.org/software-engineering/>

The fast pace and competitiveness of software engineering don't exactly make it a "dream job" in the sense of being easy. But, for a hard job, it's very nice.

Using Agile development processes, just mentioned, plays a big role in making software engineering a satisfying thing to do. We interact directly with the parties who make a difference, like our customers, and we control how we go about doing our work.

Maybe other types of engineering will move equally far in this direction? Right now, the fact we largely set our own destinies, versus mostly waiting to hear what to do from the people one works for, is an advantage for picking a software engineering career. You feel especially valued as a technical contributor, as your company rolls out software-based products and services.

Typical work

APPLICATIONS DEVELOPMENT

- Problem solving-based, non-Web-based software development that includes programming languages such as Java and C#.

SYSTEMS DEVELOPMENT

- Designing and coding background software created to support application development; includes program languages like C and C++.

WEB DEVELOPMENT

- Designing software or applications to run in a Web browser and used programming languages such as HTML, JavaScript and PHP.

EMBEDDED SYSTEMS DEVELOPMENT

- Designing computing systems and software to work on non-computing devices, such as automobiles. Uses programming languages such as C and assembly languages. (But see note, below here.)

Here's where we sketch some of the different styles of work that a software engineer may do. All of these involve the activities described on Slide 4.

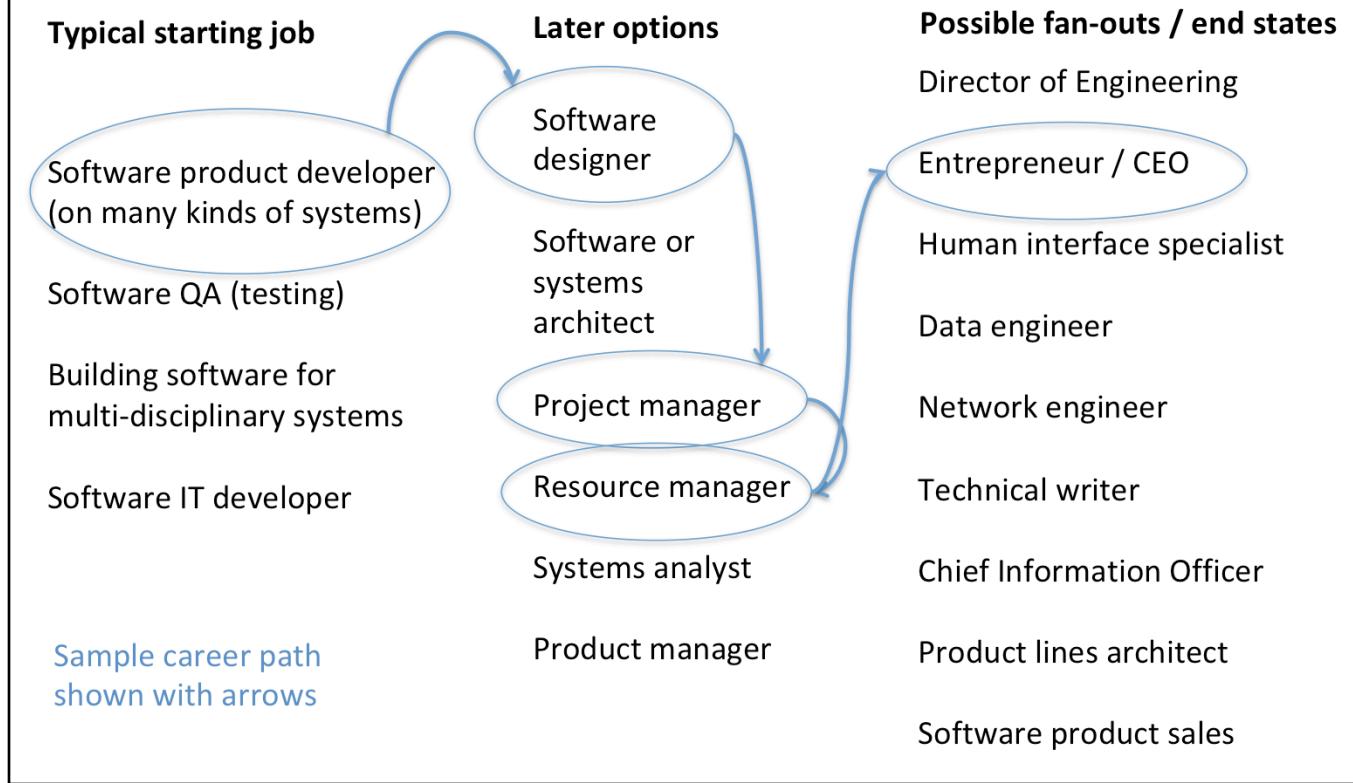
Applications Development also includes a strong aspect of Interaction Design – creating User Experiences. Skills at doing this are increasingly valued in this field.

Systems Development is a place where a degree in Computer Science can be a good choice because the application of underlying scientific principles can make a big difference in how fast or how reliably this software works, and the software created impacts a lot of software built on top of it.

In Web Development, there often is much more to it – not emphasized here is developing the server side of the web software, something that is a big part of the job and very interesting to do. Consider, for example, developing web services in the Cloud to meet the needs of large numbers of users.

For Embedded Systems Development, an understanding of Computer Engineering can be very helpful, because a lot of the software closely interacts with hardware. The comment at the end of this description refers to the changing nature of such software. Much of it now is done in higher-level languages and more resembles Applications Development. This is because of the growing amount of software that looks like applications and serves users in new ways, that we see on automobiles and appliances.

Typical career paths



"Later" in this picture could be any time – about a third of Rose alumni progress to some position where they have significant leadership, within 5 years. That includes jobs like shown under "Later".

IT is "Information Technology". Typically this refers to systems built for internal use by an organization. Like the payroll system for a company. In technically-focused companies, IT can include complex systems – like the production system for a pharmaceutical company.

A Systems Analyst or Business Analyst talks to customers about their requirements.

A Resource Manager is in charge of the people working on projects. They often have multiple projects, if these are small. They do hiring, manage budgets, etc.

Trends

- Mobile has become a \$3.3 trillion dollar industry
- Security is projected to be a \$60 billion dollar industry in 2015
- Big Data is just starting to scratch the surface as a \$125 billion dollar industry
- Internet of Things is projected to become a \$7.1 trillion dollar industry by 2020
- Robotics is a \$12.3 billion dollar industry
- Cloud computing is a \$100 billion dollar industry
- Health informatics is a \$6 – \$7 billion dollar industry

This is another way of saying that there will be high demand for software engineers for a long period of time.

Note that some of these numbers are Trillions, not Billions.

You can imagine how competitive most of these areas are – Like, anyone in the world can write an app for an iPhone and sell it from Apple's online store for 99 cents. So, the fortunes of companies large and small tend to vary a lot over time. One year Apple or Amazon is in trouble, the next it is among the most profitable in the world.

Just picking a particular area from this list could be a career – like digesting all the data produced by some category in the Internet of Things.

Cool places to work

TWITTER Social Media \$121,642	NETAPP Enterprise Computer-Storage Products \$93,726
LINKEDIN Social Networking \$127,557	RACKSPACE Cloud Computing and Web-Hosting \$82,000
FACEBOOK Social Networking \$118,445	INTEL Semiconductor Manufacturing \$97,403
GUIDEWIRE Life and Property Insurance Software \$108,918	APPLE Consumer Technology Devices \$110,867
GOOGLE Search Engine \$118,968	CAREERBUILDER.COM Job Recruitment Website \$73,850
QUALCOM Semiconductor Manufacturing \$88,312	CITRIX Enterprise Software \$88,728
RIVERBED Enterprise Network Hardware/Software \$109,464	TEXAS INSTRUMENTS Semiconductor Manufacturing \$91,633
INTUIT Financial and Tax Preparation Software \$137,424	EBAY E-Commerce \$120,424
MATHWORKS Computational Software for Engineering \$81,060	SALESFORCE Cloud Computing \$112,942
RED HAT Open Source Software \$79,725	NATIONAL INSTRUMENTS Test Equipment for Building Technology \$64,129



New SalesForce "Obelisk"
In San Francisco

These are some leading employers of software people. There are many more, including branches of the US government, for instance.

The salaries noted are average hiring salaries for people who do software work there.

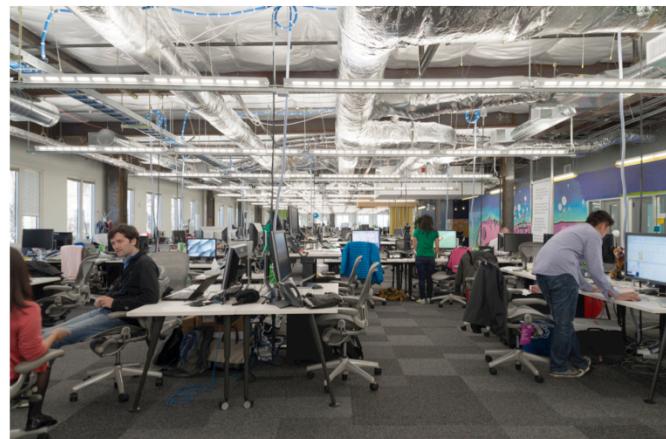
The fact these are so high is largely influenced by the shortage – the demand for qualified people significantly exceeds the supply. This also affects working conditions and the ability of software engineers to help control their own destinies on the job – see Slide 5 on Agile.

Image from <https://www.archdaily.com/889519/salesforce-tower-pelli-clarke-pelli-architects>

What's it like?

- Social, not solo all the time
- Intense and competitive
- More turnover than some “stable” jobs
- Creative – you get to see your stuff work
- Good vibes – you benefit users
- Cross-disciplinary – a part of everything
- Long hours? It depends...

Right – software engineers working at Google in India, and at Facebook’s headquarters.



The list of “what it’s like” qualities are intended to frame your expectations of this kind of work. While the Facebook people shown all seem to be off programming on their own, this is just a part of what they do. One reason they are in “open” offices is to promote interaction. That is key to their success on a software project.

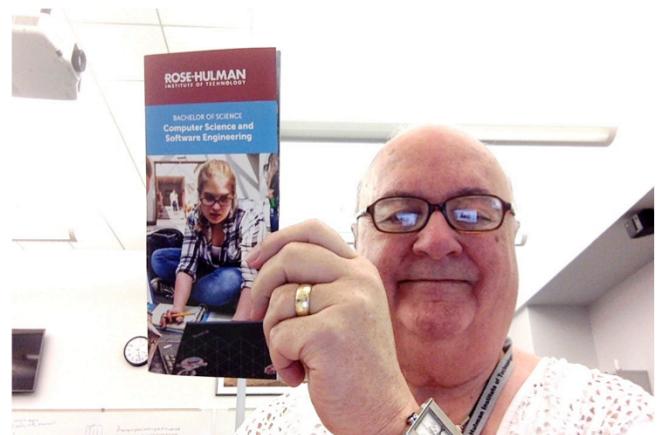
The “Long hours” point at the end has to do with the variability of this aspect. In adopting Agile practices, some teams are able to bargain for a very reasonable work week, arguing that this makes them more focused while they are on the job. At some large companies, software workers are encouraged to just work a normal schedule. However, in other businesses, overtime is in fact a way of life. Especially notorious for this is the software game development industry, who have a reputation for squeezing as much work out of each developer as they can. So, this aspect of working conditions varies a lot – you need to find a way to discover what it’s like at a particular place of work, before signing up there.

Facebook image from <https://newsroom.fb.com/media-gallery/menlo-park-headquarters/inside-facebook-headquarters/>

Google India image from <https://www.livemint.com/Industry/DAMO9oEbyMfTqgOhsz9ofJ/Putting->

Rose-Hulman's CS and SE programs

- See the brochure we handed out!
- Experiential learning emulates industry.
- Faculty support based on our experience.
- Students support each other's studies.
- Project and team-based courses.
- Choice of CS, SE, or both majors.
- International components, including getting an international CS degree.
- Data science second major and Robotics minor options.
- Opportunities for great internships as you study here.



Above – I “uphold” our programs!

The opportunity to try doing this work is the most important background you can get.

Just hearing theory doesn't do much to improve your skills – Doing it does.

Doing it in a supportive environment is golden.

Rose's department is “CSSE”

- We offer both CS and SE majors:
 - CS has extra courses in theory of computing.
 - SE has extra courses in building industry-quality systems.
- Both majors have theory as well as practice.
 - E.g., both include a Senior Project.
 - CS majors can opt to do a Thesis instead.
- Rose is only 1 of 29 schools in the US to have an ABET-accredited SE program.

What you'll do in your **career** as a software engineer:

Mostly the “practice”

CS = Computer Science
SE = Software Engineering

ABET: See <http://main.abet.org/aps/Accreditedprogramsearch.aspx>.

Why are there so few programs in Software Engineering, when this is the name of the vocation that CS majors also will work in?

We think it's the theoretical bias of most colleges in the US. They typically started out as part of a Math department. Some began as a part of Electrical and Computer Engineering.

The PhD's who populate CS departments are trained more in theory, in graduate school. So, why wouldn't they think this is more important for their undergraduate students?

A modest proposal

- If you are sure you want to do something else,
- Go get a degree in that AND CS or SE. Or,
- Get a minor in CS or SE.
- A huge percentage of people with other majors end up working with software because
 - Software is a big part of their field, and
 - Creating software is where the jobs are.



Say, you love working for non-profits. They need software, badly. How's that going to happen? How about you?

Above picture – Mixing software development with political science and working for non-profit organizations. From <https://andela.com/insights/doing-work-that-matters-rehema-wachiras-developer-journey/>.

Or, Maybe you want to be an ME?

- VP from Ford who is on one of our department advisory committees, in 2014:
- “By 2016, every Ford will need to have 200 million lines of code running in it. I don’t know how we are going to get there.”

The 2016 Ford Fusion Hybrid – a pretty complex machine. Or, is it a computer with a machine?



The person writing all this software for a mechanical device like a car has to know about both the car and the software. You can't have teams of people who just know one or the other, and expect to be successful.

Ford used software to gain advantages over competitors – for example, having a V6 engine in their trucks, which was lighter and more powerful and got better mileage than other companies' V8 engines.

Image from https://www.cars.com/research/ford-fusion_hybrid-2016/.