



Ditt första jobb som programmerare

Steg för steg mot ditt första jobb
inom programmering

Innehåller interaktiva övningar
och inspirerande ljudklipp online

Pierre Gergi

Pierre Gergi

Ditt första jobb som programmerare

Steg för steg mot ditt första jobb inom programmering

För nytexaminerade från universitet, högskola och yrkesutbildning i Sverige

Upphovsrättsligt skyddat material

Alla rättigheter förbehållna. Ingen del av denna bok får mångfaldigas, lagras i ett elektroniskt system eller spridas i någon form eller med några medel – vare sig elektroniska, mekaniska, genom kopiering, inspelning eller på annat sätt – utan skriftligt tillstånd från upphovsrättsinnehavaren.

Automatiserad teknik, vilken används för att analysera text och data i digital form i syfte att generera information, enligt 15a, 15b och 15c §§ upphovsrättslagen (text- och datautvinning), är förbjuden.

© 2026 Pierre Gergi / Lebario IT – www.lebario.se

Språklig granskning: Perla Touma & Giorgio Gergi

Omslagsdesign: Omar Kaddour

Bilder: Vissa bilder i boken är hämtade från Freepik (www.freepik.com) och används i enlighet med deras licensvillkor.

Förlag: BoD · Books on Demand, Östermalmstorg 1, 114 42 Stockholm, Sverige, bod@bod.se
Tryck: Libri Plureos GmbH, Friedensallee 273, 22763 Hamburg, Tyskland.

ISBN: 978-91-8114-433-8

Förord

När jag tog mina första steg mot att få ett jobb som programmerare minns jag tydligt känslan. Allt var nytt, spännande och ibland ganska förvirrande. Det fanns gott om böcker som lärde ut själva programmeringen, men jag upplevde att det nästan helt saknades böcker som verkligen visade vägen från klassrummet till det första riktiga jobbet. Dessutom saknades det böcker som både var anpassade för den svenska arbetsmarknaden och skrivna på en enkel och tydlig svenska. Böcker som kunde guida både studenter och nyutexaminerade längs vägen, på ett sätt som känns nära och begripligt.

Samtidigt har vägen från studier till arbetsliv blivit mer komplex än tidigare. Arbetsmarknaden förändras snabbt, och det som gällde i går är inte alltid självklart i dag. Många undrar därför: **”Kommer de kraftfulla AI-verktygen att ta över jobben, särskilt för oss som är nya i branschen?”**. Den oron är helt förståelig. Tekniken utvecklas snabbt och ibland känns det som att maskiner klarar mer för varje vecka som går.

I praktiken används AI i dag oftast som ett verktyg i vardagen, till exempel för att skriva kod snabbare, hitta buggar eller förstå komplexa problem. Men det är fortfarande du som utvecklare som måste förstå vad som händer, kunna ta ansvar för lösningen och förklara den för teamet.¹

Nyfikenhet, viljan att lära och den mänskliga förmågan att samarbeta och förstå problem är precis det som gör dig viktig. AI kan stötta oss, men den ersätter inte människor som kan tänka självständigt, kommunicera, beskriva och växa tillsammans med andra.²

Att vara junior handlar inte bara om att skriva kod utan också om att utvecklas som person och bidra till teamet på många olika sätt.³ I en tid där AI blir allt vanligare söker många arbetsgivare just juniora utvecklare som kan samarbeta, ta

¹ [Stack Overflow Developer Survey \(2023, 2024\)](#)

² [Arbetsförmedlingen – Yrkesprognoser inom IT-2025](#)

³ [Svenska små och medelstora företags AI-kompetens](#)

feedback och använda nya verktyg på ett klokt sätt. Din roll formas alltså inte bort av tekniken, den formas tillsammans med den.⁴ Det är sant att vissa yrken kommer att förändras, och att en del arbetsuppgifter kan försvinna. Samtidigt kommer nya roller och nya behov att växa fram, ofta snabbare än vi kan föreställa oss.

För att hänga med behöver man därför inte bli expert på AI, utan snarare förstå hur verktygen används i praktiken och fortsätta utveckla sina grundläggande färdigheter som utvecklare.

Det är viktigt att komma ihåg att **AI oftast inte tar ditt jobb, men den förändrar hur jobbet ser ut. Risken att stå utan jobb ökar främst om du inte lär dig använda de verktyg som branschen faktiskt arbetar med.**

Under åren har jag mött många studenter och unga utvecklare som älskar att koda och ändå känner sig osäkra. Hur hittar man rätt väg i en bransch som förändras hela tiden? Det är en fråga som återkommer och som i grunden handlar om mod, riktning och stöd.

Den här boken är mitt försök att fylla det tomrummet, särskilt i en tid där många känner osäkerhet och förvirring efter framväxten av AI-verktyg och diskussionerna om hur de påverkar programmeraryrket och framtidens arbetsmarknad. Jag har samlat de erfarenheter jag fått genom åren, de frågor som återkommer hos utvecklare och kollegor, insikter från ett stort antal vetenskapliga och forskningsbaserade artiklar inom området. Målet är att ge dig konkreta råd, tydliga exempel och små praktiska steg som hjälper dig att hitta din plats i arbetslivet som utvecklare, även när allt känns nytt, osäkert eller snabbt föränderligt.

Jag hoppas att varje kapitel ska kännas som om någon sitter bredvid dig och lugnt säger:

“Du klarar det här, ta ett steg i taget, och glöm inte att branschen verkligen behöver någon som dig.”

⁴ [Harvard Business Review](#)

Introduktion – Din resa börjar här

Du håller just nu i en bok som faktiskt kan bli din biljett till ett nytt liv. Ett jobb som programmerare kan förändra mycket: din **ekonomi**, din **frihet** och din **framtid**.

Vägen dit är inte alltid enkel. Kanske har du precis avslutat en utbildning. Kanske har du lärt dig koda på egen hand hemma på kvällarna, eller så är du ny i Sverige och försöker förstå hur arbetsmarknaden fungerar.

Oavsett vem du är, är du inte ensam. Det här är boken jag själv hade velat läsa när jag tog mina allra första steg in i programmering och IT-branschen.

I den här boken får du bland annat veta:

- Vad en programmerare faktiskt **gör i vardagen**.
- Vad **arbetsgivare** i Sverige letar efter.
- Hur du bygger en **portfölj** som verkligen visar vad du kan.
- Hur du skapar professionella **kontakter** som öppnar dörrar.
- Hur du tolkar **jobbannonser** och förstår vad arbetsgivare menar.
- Hur du skriver ett **CV** och **personligt brev** som får dig att sticka ut.
- Hur du gör en **checklista** för att vara redo.
- Hur du hittar rätt **sajter, verktyg** och **företag** i Sverige.
- Hur du förbereder dig för tekniska och personliga **intervjuer**.
- Hur du förhandlar om din första **lön** på ett smart och tryggt sätt.
- Hur du går från **praktik** till din första **anställning** och tar nästa steg.
- Vad du kan göra efter ett **avslag** för att komma starkare tillbaka.
- Hur du **bygger vidare** på din första anställning.

Allt detta kommer vi att arbeta med tillsammans, steg för steg, kapitel för kapitel, längs din resa.

Så använder du boken

➤ Arbetssätt

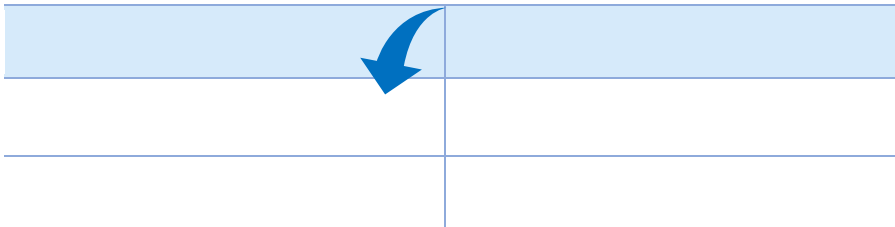
Allt i boken är upplagt för att göra det lättare för dig att lära, minnas och använda kunskapen i praktiken. Kombinationen av text, tabeller, online-övningar och tips gör att du snabbt kan hitta det du behöver och komma vidare. Se boken som ett verktyg, inte bara att läsa, utan också att arbeta med.

▪ Naturligt och personligt språk

Språket i boken är enkelt och lätt att följa. Du ska kunna läsa utan att fastna, förstå direkt och känna att texten talar till dig. Stilen är varm och samtalande, som om du hade en handledare vid din sida som förklarar, uppmuntrar och visar vägen. Vissa engelska termer används medvetet, eftersom de är en naturlig del av vardagsspråket inom utvecklingsområdet, även i Sverige.

▪ Tydligt med tabeller

I boken används tabeller för att göra det lättare att se och minnas det viktiga. De ger en snabb överblick och hjälper dig att repetera utan att läsa långa texter.



▪ QR-koder för länkar

I boken hittar du länkar till externa webbplatser och resurser som kan ge extra stöd och inspiration. Eftersom **innehållet** på dessa sidor kan förändras över tid ansvarar författaren inte för eventuella uppdateringar eller borttagna sidor.

För att göra det enkelt att komma åt några **länkar** som kan ändras över tid finns allt samlat i QR-koder. Dessa leder alltid till den senaste tillgängliga versionen, även om webbadresserna ändras med tiden.



<https://lebario.se/books/dfj/qr>

- **Inspirationsresor och ljudklipp**

I varje kapitel finns en eller flera Inspirationsresor, korta berättelser som speglar verkliga upplevelser eller tänkta situationer, där personernas namn har ändrats av hänsyn till integriteten. Några av dem finns även som ljudklipp som du kan lyssna på genom att skanna QR-koden.

- **Författarens tips**

I slutet av varje kapitel får du även personliga råd från författaren. Det är små insikter, erfarenheter och tips som kan ge dig extra styrka på din resa.

- **Interaktiva övningar.**

Varje kapitel innehåller interaktiva online-övningar som du kan göra direkt. På så sätt får du inte bara läsa, utan också öva och testa dina kunskaper i praktiken.

Om systemet ber dig ange användarnamn eller både användarnamn och lösenord, finns informationen tillgänglig via länken eller genom att skanna QR-koden.

- **Bokens senaste uppdateringar**

I slutet av boken hittar du de senaste uppdateringarna och rättelserna, som du enkelt kan nå via en QR-kod eller en länk.

➤ Struktur

Boken är uppdelad i **13** kapitel, fördelade på **4** delar.

Varje kapitel innehåller:

-  Kapiteltitel
-  Mål och vad du kommer att lära dig
-  Kapitelöversikt
-  Huvudinnehåll
-  Inspirationsresor
-  Viktiga begrepp att förstå
-  Vanliga missförstånd
-  Vanliga misstag att undvika
-  Verktyg och plattformar att utforska
-  Branschtips – i Sverige och globalt
-  Miniuppgift
-  Reflektionsfrågor
-  Pierres tips – personliga råd från författaren
-  Kort och gott
-  Interaktiva övningar

Om författaren

Pierre har en ingenjörsexamen i datateknik med inriktning mot programmering och en masterexamen i IT-projektledning. Han har lång erfarenhet av mjukvaruutveckling och digitala lösningar samt bred teknisk kompetens inom IT, med en stark grund i datorsystem, kommunikation, molnbaserade miljöer, IT-säkerhet och projektledning.



Han arbetar som legitimerad IT-lärare med undervisning i programmering och IT. Hans arbete präglas av ett starkt fokus på pedagogisk kvalitet, didaktisk utveckling och hur tekniska ämnen kan göras begripliga och tillgängliga för fler.

Han är djupt engagerad i att göra tekniskt lärande tillgängligt för alla, särskilt genom utveckling av e-lärande och moderna pedagogiska metoder. Utöver sitt pedagogiska arbete är han även aktiv inom digitalisering av kulturarv, med intresse för hur teknik kan bidra till ett hållbart bevarande och ökad tillgänglighet. Detta samlade engagemang speglas tydligt i det arbete han bedriver genom sitt företag i Sverige, Lebario IT.

Före flytten till Sverige arbetade han i nästan 20 år som universitetslärare i datavetenskap, med fokus på mjukvaruutveckling och IT-drift, och drev samtidigt ett eget IT-företag. Genom att kombinera undervisning och arbetslivserfarenhet utvecklade han en stark och verklighetsnära förståelse för hur IT-system fungerar i praktiken.

Ditt första jobb som programmerare är hans första bok på svenska och riktar sig till dig som vill ta steget från studier till ditt första jobb som utvecklare.

Innehållsförteckning

Del 1: Innan du börjar söka	16
Kapitel 1: Välkommen till kodarvärlden	17
1.1 Utbildningen är klar, vad händer nu?	19
1.2 Vad gör egentligen en programmerare?	19
1.3 Skillnaden mellan att plugga kod och jobba som utvecklare	20
1.4 Vad gör att vissa utvecklare trivs och andra inte?	22
1.5 Olika typer av programmeringsjobb	22
1.6 Hur ser arbetsmarknaden ut för nya utvecklare i Sverige?	24
1.7 Min väg in i kodvärlden (författarens berättelse)	25
Kapitel 2: Vad letar arbetsgivare efter?	31
2.1 Det här tittar arbetsgivare på, mer än bara kod	33
2.2 Tekniska och mjuka färdigheter, vad förväntas?	33
2.3 Vad säger arbetsgivare själva?	35
Kapitel 3: Bygg din portfölj, visa vad du kan	41
3.1 Varför portföljen är viktigare än ditt CV	43
3.2 Vad bör finnas i en juniorportfölj?	43
3.3 Så optimerar du din GitHub	45
3.4 Så presenterar du ett projekt	45
Kapitel 4: Bygg professionella kontakter som utvecklare	51
4.1 Varför kontakter spelar roll	53
4.2 Så här börjar du, steg för steg	53
4.3 Bygg relationer som håller på lång sikt	54
4.4 Kombinera digitala och fysiska möten	54
Del 2: Ansökningsprocessen	60

Kapitel 5: Jobbannonser, så läser du mellan raderna	61
5.1 Vad menas med "krav" och "meriterande"?	63
5.2 Krav i annonser, vad betyder de egentligen?	63
5.3 Vi söker dig som, vad betyder det egentligen?	64
5.4 Så tolkar du jobbannonsen mellan raderna	65
5.5 Vanliga uttryck och dolda budskap i annonser	65
Kapitel 6: Skriv din ansökan, CV och personligt brev	73
6.1 Vad vill arbetsgivaren veta?	75
6.2 Strukturen: tre enkla delar.....	75
6.3 Vad skiljer ett tekniskt CV från andra?	76
6.4 Struktur och innehåll i ett starkt CV	76
6.5 Hur lyfter du dina projekt och kunskaper?	77
6.6 Personligt brev, visa vem du är bakom koden	78
Kapitel 7: Är du redo? Din checklista före ansökan	89
7.1 Har du byggt något som visar vad du kan?	91
7.2 Kan någon hitta dig online?	91
7.3 Är din ansökan redo att skickas?.....	92
7.4 Klarar du en intervju, eller behöver du öva mer?	92
7.5 Vad driver dig egentligen?	93
7.6 Vad är ditt nästa steg just i dag?	93
7.7 Har du koll på dina styrkor och svagheter?	94
7.8 Har du byggt kontakter ens lite?	94
7.9 Har du byggt något som gör nytta?	95
7.10 Har du en plan eller chansar du bara?	95
Kapitel 8: Här söker du jobb: sajter, verktyg och bolag att känna till.....	101
8.1 Webbplatser för att söka jobb	103
8.2 Techbolag i Sverige att hålla koll på	104

8.3 Digitala verktyg som hjälper dig i jobbsökandet	105
Del 3: Intervjufasen	110
Kapitel 9: Förbered dig för tekniska intervjuer	111
9.1 Klassiska tekniska frågor, vad de egentligen testar	113
9.2 Beteendefrågor som ofta dyker upp i tekniska intervjuer	116
9.3 Hur du kan prata om motivation och kultur, inte bara kod	118
9.4 Svåra och oväntade frågor, och hur du kan hantera dem lugnt	120
9.5 Bra frågor du själv kan ställa till arbetsgivaren.....	121
Kapitel 10: Så förhandlar du om din lön, smart och tryggt.....	131
10.1 Förbered dig rätt, innan du säger ett ord	133
10.2 Steg för steg: Så lyckas du med din löneförhandling	133
10.3 Tänk på helheten, inte bara siffran	134
10.4 När och hur du tar upp lön	134
10.5 Själsäkerhet, tystnad och rätt tidpunkt	135
10.6 Om svaret blir nej	135
Del 4: Efter intervjun och framåt	142
Kapitel 11: Från praktik till jobb, vägen till din första anställning	143
11.1 Praktiken: din chans att prova och glänsa.....	145
11.2 Gör mer än förväntat	146
11.3 Ställ frågor som visar att du bryr dig på riktigt.....	147
11.4 Skriv ner det du gör, det blir guld senare	147
11.5 Fråga om nästa steg	148
11.6 Det blev inget jobb, men du går inte tomhänt	149
Kapitel 12: Att hantera avslag och komma tillbaka starkare	157
12.1 Det är okej att må dåligt, men inte för länge	159
12.2 Analysera och tolka feedback	159
12.3 Justera din strategi, inte dina mål	160

12.4 Bygg upp din motivation igen	160
12.5 Fokusera på nästa steg och håll momentum uppe	160
12.6 Frilans: ibland nästa steg parallellt med jobbsökandet	161
Kapitel 13: Din väg framåt: programmering som karriär och livsresa	167
13.1 Du har fått jobbet, vad händer nu?	169
13.2 De första 90 dagarna, så hittar du fotfästet	169
13.3 Lärande som vana, ditt ansvar för att växa	170
13.4 Be om feedback	171
13.5 Bygg ditt personliga varumärke	171
13.6 Från junior till nästa nivå	172
13.7 Några slutord	173
Efterord.....	180

Del 1: Innan du börjar söka



Kapitel 1: Välkommen till kodarvärlden

Kod är inte bara teknik. Det är ditt verktyg för att förändra världen.

Mål och vad du kommer att lära dig

Det här kapitlet är ditt första steg från studiernas trygga värld till arbetslivet som utvecklare. Du får inte bara en tydligare bild av vad programmeraryrket innebär i praktiken, utan också hjälp att förstå hur du kan navigera i början av din resa.

Du kommer att:

- Ha en klarare bild av vad en programmerare faktiskt gör i vardagen.
- Känna till olika vägar in i branschen.
- Förstå vilka egenskaper som gör att vissa trivs i yrket och andra inte.
- Reflektera över din egen situation och börja forma en plan för dina första steg i karriären.

Kapitelöversikt

- 1.1 Utbildningen är klar, vad händer nu?
- 1.2 Vad gör egentligen en programmerare?
- 1.3 Skillnaden mellan att plugga kod och jobba som utvecklare.
- 1.4 Vad gör att vissa utvecklare trivs och andra inte?
- 1.5 Olika typer av programmeringsjobb
- 1.6 Hur ser arbetsmarknaden ut för nya utvecklare i Sverige?
- 1.7 Min väg in i kodvärlden (författarens berättelse)

1.1 Utbildningen är klar, vad händer nu?

Att bli klar med en programmeringsutbildning är ingen liten bedrift. Du har kämpat med kod, projekt, buggar och sena kvällar. Nu står du här, redo att ta steget in i den riktiga kodarvärlden. Det är en värld som rör sig snabbt, där verktyg och språk förändras hela tiden, men där grunden du byggt är din största tillgång.

Det här kapitlet är ditt startskott. Vi ska inte prata om for-loopar eller syntax. I stället handlar det om att förstå den nya verkligheten du möter som färdigutbildad utvecklare.

Välkommen, inte bara till ett jobb, utan till ett yrkesliv där du varje dag får chansen att skapa, lösa och växa.

1.2 Vad gör egentligen en programmerare?

Att jobba som programmerare är inte bara att skriva kod. I grunden handlar jobbet om att lösa problem och skapa lösningar som fungerar i vardagen. Det kan vara en webbplats, en app eller ett system som gör något enklare, snabbare eller säkrare.

Arbetet börjar ofta med att förstå behov och krav. Därefter skriver programmeraren kod som talar om för datorn vad den ska göra. En stor del av tiden går också åt till att testa, hitta fel och förbättra lösningen.

Programmerare jobbar sällan ensamma. De flesta arbetar i team där samarbete, kommunikation och nyfikenhet är lika viktiga som tekniska kunskaper. I Sverige är titeln ofta bred, och du kan till exempel kallas **utvecklare**, **systemutvecklare**, **webbutvecklare**, **frontendare** eller **backendare**.

Yrket förändras ständigt, vilket gör att man fortsätter lära sig nya saker genom hela karriären.

Kort sagt är en programmerare någon som använder logik, kreativitet och teknik för att bygga lösningar som fungerar i verkligheten.

1.3 Skillnaden mellan att plugga kod och jobba som utvecklare

Att lära sig programmering i skolan och att faktiskt jobba som utvecklare är lite som två olika världar. Det låter kanske överdrivet, men många nyexaminerade märker snabbt hur annorlunda det känns när de går från klassrum till arbetsplats.

I skolan:

I skolan är det mesta ganska strukturerat. Du får uppgifter med ett tydligt rätt svar, instruktioner att följa och en lärare som inte bara hjälper dig om du fastnar utan också förklarar övningarna steg för steg och visar hur du kan skriva dem på ett logiskt sätt. Du får tid att experimentera, göra misstag och försöka igen. Det är ungefär som att lära sig cykla med stödhjul.

På jobbet:

När du kliver in i arbetslivet känns det plötsligt annorlunda. Du får inga färdiga instruktioner längre, ingen som säger exakt hur du ska göra. I stället kan uppgiften låta ungefär så här: *"Kan du bygga en funktion som gör att användare kan spara sina favoritartiklar?"*

Hur löser du det? Det bestämmer du. Du kanske behöver plöja igenom dokumentation, googla runt, fråga en kollega och samtidigt se till att koden blir tydlig, testbar och hållbar.

Och kanske det viktigaste: du gör det inte ensam längre. Nästan allt sker i team. Det betyder att du måste kunna samarbeta, prata med andra och ibland ta egna beslut, även när det känns lite osäkert.

Därför spelar det roll:

Kanske tänker du själv: *"Jag var rätt bra på programmering i skolan"*, men när du började jobba kändes det ändå som att du visste väldigt lite. Det är faktiskt vanligt. Många som får sitt första jobb blir lite chockade. Plötsligt känns allt långsammare, mer osäkert, ibland till och med som om man inte räcker till, trots att man var stark i skolan. Och vet du vad? Det är helt normalt.

Att gå från skolmiljö till verkligheten är som att gå från att cykla med stödhjul till att cykla i trafiken.

Nu ska vi sammanfatta de två tidigare styckena i **en tabell** som är lätt att läsa och komma ihåg.

I skolan	På jobbet
Det mesta är strukturerat, du får uppgifter med ett rätt svar.	Det finns inga färdiga instruktioner, du måste själv hitta lösningen.
Läraren förklarar övningarna steg för steg och visar hur du kan skriva dem logiskt.	Du behöver själv läsa dokumentation, googla och fråga kollegor.
Du får tid att testa, göra misstag och försöka igen.	Du förväntas leverera kod som är tydlig, testbar och hållbar.
Du jobbar oftast ensam med dina uppgifter.	Nästan allt sker i team, du måste samarbeta, kommunicera och ibland ta egna beslut.
Känslan är som att cykla med stödhjul.	Känslan är mer som att cykla fritt, du styr själv men måste hålla balansen.

 **OBS!** I den här boken får du ofta se tabeller. De gör det enklare för dig att snabbt få överblick och minnas det viktiga, mycket lättare än med långa texter.

1.4 Vad gör att vissa utvecklare trivs och andra inte?

Du har pluggat klart, kämpat med projekt och svurit över buggar, men också känt glädjen när allt till slut fungerar. Nu väntar nästa steg: att faktiskt jobba som utvecklare.

Här dyker en viktig fråga upp: varför trivs vissa i rollen, medan andra känner sig fel redan efter några månader?

Det handlar sällan om att vara smartast i rummet. Oftare är det hur man tänker, arbetar och hanterar vardagen som utvecklare.

Några egenskaper som gör stor skillnad:

- **Nyfikenhet:** De som trivs bäst vill förstå hur saker hänger ihop och gillar att gräva när något inte funkar.
- **Tålamod:** Att koda är att misslyckas, många gånger. Den som orkar försöka igen och inte ger upp växer snabbt.
- **Logiskt tänkande:** Du behöver inte vara ett geni, men det hjälper att gilla mönster och struktur.
- **Samarbete och kommunikation:** Även om du ofta sitter själv vid datorn är programmering ett lagarbete. Att prata och lyssna är minst lika viktigt som att koda.
- **Initiativförmåga:** I många grupper är det ingen som säger exakt vad du ska göra. Den som tar egna steg utvecklas mest.

1.5 Olika typer av programmeringsjobb

När man hör ordet *programmerare* tänker många på någon som bara sitter och skriver kod. Men sanningen är att yrket kan se väldigt olika ut beroende på vad du jobbar med. Vissa bygger det användaren ser, andra sköter logiken bakom kulisserna, och några gör både och. Här får du en överblick som kan hjälpa dig att hitta din plats.

Frontend, det som syns

Som frontendutvecklare jobbar du med det visuella: knappar, färger, menyer och designen på hemsidor eller appar. Här används ofta HTML, CSS och JavaScript. Frontend passar dig som gillar form, kreativitet och att snabbt se resultat av ditt arbete.

Backend, det som driver allt

Backend är som motorn bakom skärmen. Här bygger du logiken, databaserna och API:erna som gör att allt fungerar. Mycket av jobbet sker på serverna, den del som kör koden och sparar information. Vanliga språk är Python, Java, C# och PHP. Backend passar dig som gillar struktur, logik och problemlösning.

Fullstack, lite av båda

En fullstackutvecklare kan både **frontend** och **backend**. Du får en helhetsbild och kan bygga kompletta lösningar. Det är bra för dig som gillar variation och vill förstå hur allt hänger ihop.

AI och maskininlärning, när koden blir smart

Inom AI (*Artificial Intelligence*) och *machine learning* handlar det om att få datorer att lära sig från data. Det kan vara att rekommendera filmer, tolka språk eller känna igen bilder. Här krävs både programmering och intresse för matematik och analys. Perfekt för dig som gillar siffror och ny teknik.

Andra viktiga roller på arbetsmarknaden

Förutom dessa finns fler spännande inriktningar:

- **Testutvecklare:** ser till att koden fungerar genom automatiska tester.
- **Apputvecklare:** arbetar med att utveckla appar för mobiltelefoner och surfplattor.
- **Spelutvecklare:** skapar spel och interaktiva upplevelser.
- **DevOps:** bygger broar mellan utveckling och drift.
- **IT-säkerhet:** skyddar system från intrång och hot.

- **Dataanalytiker:** analyserar data för att stödja beslut och förbättra system.

Hur väljer man?

Det fina är att du inte måste välja rätt direkt. Många börjar i en roll och byter senare när de upptäcker vad som passar dem bäst. Praktik, egna projekt och intresse brukar visa vägen.

"Jag började med frontend, men insåg att jag älskar logiken i backend. I dag jobbar jag som fullstackutvecklare och trivs som fisken i vattnet", säger en av mina vänner.

1.6 Hur ser arbetsmarknaden ut för nya utvecklare i Sverige?

Arbetsmarknadens behov

Efterfrågan på programmerare är fortfarande hög i stora delar av landet. Enligt Arbetsförmedlingens rapport *"Regionala utsikter våren 2025"* råder det fortsatt brist på system och mjukvaruutvecklare, särskilt i storstäder som Stockholm, Göteborg, Malmö och Umeå. Prognosen gäller för hela perioden 2025–2026, och behovet väntas till och med öka något under 2026, trots att återhämtningen på arbetsmarknaden går långsamt.⁵

Var finns jobben?

Både privata företag och offentliga verksamheter söker aktivt efter rätt IT-kompetens. En stor del av platsannonserna gäller roller som frontend, backend och fullstackutvecklare.

Enligt Arbetsförmedlingens sommaruppdatering 2025 hör systemutvecklare till de yrken med bäst jobbchanser under 2025–2026. Den ökade digitaliseringen inom både näringslivet och den offentliga sektorn driver på behovet av kunniga utvecklare.⁶

Statistik och insikter

Arbetsförmedlingen bedömer att efterfrågan på IT-kompetens kommer att vara

⁵ Arbetsförmedlingen (2025). Regionala utsikter våren 2025

⁶ Arbetsförmedlingen (2025). Här är yrken med bäst jobbchanser.

stabil under hela perioden 2025–2026, även om konjunkturen är svag. I rapporten *”Var finns jobben och var finns arbetskraften? (våren 2025)”* framgår att data och IT hör till de områden som har störst rekryteringsproblem i Sverige. Nästan hälften av arbetsgivarna uppger att de har svårt att hitta rätt kompetens.⁷

Du är inte ensam, konkurrensen finns där

Det räcker inte längre att bara kunna koda lite grann. Du behöver visa upp dina kunskaper på ett tydligt sätt så att arbetsgivare förstår och värdesätter dem. Det handlar alltså både om att kunna utveckla och om att kunna kommunicera ditt värde som utvecklare.

1.7 Min väg in i kodvärlden (författarens berättelse)

När jag själv började studera programmering vid Aleppos universitet i Syrien såg vardagen helt annorlunda ut än i dag. I labbet fanns det så få datorer att vi ofta stod fyra eller fem personer runt samma maskin. Att ha en egen dator var för många helt omöjligt, eftersom priserna var alldeles för höga. Internet fanns ännu inte på den tiden, så det gick varken att söka hjälp eller hitta felmeddelanden online. Om något inte fungerade fick man klura själv, fråga en kompis eller bläddra i en tjock bok tills man hittade något som liknade en lösning.

Jag minns tydligt när jag försökte skriva mitt allra första lilla program. Det fungerade inte alls i början. Jag testade om och om igen, och varje gång dök ett nytt fel upp. Men till slut, efter flera försök, fungerade det. Programmet var enkelt, men känslan när det äntligen fungerade var enorm. Den där blandningen av lättnad och glädje gjorde att jag fastnade ännu mer för programmering.

De första stegen jag tog in i kodens värld var långt ifrån så bekväma som dagens verktyg, men just enkelheten gjorde att man verkligen lärde sig grunderna. Med varje rad kod blev det tydligare att programmering inte handlar om att kunna allt från början. Det handlar mer om tålamod, nyfikenhet och modet att fortsätta

⁷ Arbetsförmedlingen (2025). *Var finns jobben – och var finns arbetskraften?*

även när man känner sig helt fast. Programmering är en resa där man testar, misslyckas, tänker om och så småningom hittar rätt väg igen.

I dag, många år senare, när jag undervisar mina egna studenter, ser jag samma resa hos dem: förvirringen, envisheten och sedan den där lilla gnistan av glädje när deras program äntligen fungerar. Varje sådan stund påminner mig om mina egna första steg.

Min väg in i kodvärlden var långt ifrån rak, men varje problem, varje felmeddelande och varje liten framgång formade mig med tiden till den utvecklare och lärare jag är i dag.

Inspirationsresa: (Sabina, 28 år)

Sabina kom till Sverige som vuxen och började läsa webbutveckling på en tvåårig YH-utbildning (Yrkeshögskola). Under studierna byggde hon flera egna projekt, bland annat en liten receptapp som hon var extra stolt över.

När det var dags att söka jobb visade hon upp sin kod och förklarade hur hon hade löst olika problem och det gav resultat. I dag arbetar hon som frontendutvecklare i Stockholm.

"Jag var rädd att min bakgrund skulle stå i vägen. Men det jag hade byggt betydde mer än allt annat", säger hon.

Viktiga begrepp att förstå

Begrepp	Enkel förklaring
Systemutvecklare	En bred roll som ofta innebär att bygga och underhålla hela systemet, både frontend och backend.
Juniorroll	Din första tjänst i branschen, där arbetsgivaren räknar med att du fortfarande lär dig.

Kodkultur	Osynliga regler och arbetssätt i ett utvecklingsteam, t.ex. hur man skriver kod, testar och samarbetar.
DevOps	Ett arbetssätt som kombinerar utveckling (<i>Development</i>) och drift (<i>Operations</i>) för att snabba upp leveransen av mjukvara. En viktig del av DevOps är CI/CD, som automatiserar bygg, test och driftsättning.
CI/CD (Continuous Integration och Continuous Delivery)	Ett arbetssätt där kod automatiskt byggs, testas och integreras, och sedan levereras eller driftsätts snabbt och stabilt.

Vanliga missförstånd

Missförstånd	Vad som stämmer
Man måste vara ett geni för att bli programmerare.	De flesta utvecklare är vanliga människor som lär sig steg för steg.
Programmering är tråkigt och ensamt.	Det kan vara roligt just tack vare utmaningarna. I verkligheten jobbar man i team, diskuterar lösningar och bygger saker tillsammans på ett kreativt sätt.
Man måste kunna allt om teknik innan man får jobb.	Ingen kan allt. Du lär dig det du behöver längs vägen, med stöd av kollegor. Att googla och använda AI-verktyg är en naturlig del av jobbet.
Kodare jobbar dygnet runt.	De flesta har vanliga arbetstider. Balansen mellan jobb och fritid är viktig.
Jobb finns bara i storstäder.	Störst behov finns i Stockholm, Göteborg och Malmö, men även mindre orter söker utvecklare.

🚫 Vanliga misstag att undvika

Vanliga misstag	Bättre sätt att göra i stället
Att jämföra sig för mycket med erfarna kollegor.	Acceptera att du är ny. Fokusera på lärande och utveckling i din egen takt.
Att undvika att ställa frågor.	Våga fråga. Det hjälper både dig och teamet att komma vidare snabbare.
Att tro att skolan förberett dig fullt ut.	Förstå att arbetslivet är annorlunda. Skolan ger dig grunden och hjälper dig ta första steget in i programmeringens värld.

🔧 Verktyg och plattformar att utforska

Verktyg / Plattform	Länk	Vad det hjälper dig med
Arbetsförmedlingen IT-yrken		Ger en bild av efterfrågan och aktuella roller inom programmering i Sverige.
Stack Overflow	stackoverflow.com	Fråga och hitta lösningar på vanliga kodproblem, lär av andra utvecklare.
LinkedIn	linkedin.com	Bygg ditt professionella nätverk och följ företag som söker utvecklare.

🌐 Branschtips

I Sverige: Efterfrågan på utvecklare är fortfarande hög, särskilt i Stockholm, Göteborg och Malmö. Många företag söker juniora utvecklare, men samtidigt har AI-verktyg som *Copilot* och *ChatGPT* förändrat

marknaden. En del uppgifter kan lösas med AI, vilket gör att arbetsgivare i dag uppskattar kreativa juniora utvecklare som är duktiga på att arbeta med moderna AI-verktyg och använda dem på ett smart sätt⁸.

Globalt: Fjärrjobb blir allt vanligare. Länder som Nederländerna och Estland erbjuder möjligheter där engelska räcker långt. Här letar arbetsgivare ofta efter motivation, problemlösningssförmåga och vilja att lära sig, inte bara efter ett perfekt CV⁹.



Miniuppgift

Skriv ner tre saker som du tycker verkar mest spännande med att jobba som programmerare och två saker som känns mest utmanande. Fundera sedan på vilka av dina styrkor som kan hjälpa dig att klara de utmaningarna.



Reflektionsfrågor

1. Vilken del av programmeraryrket lockar dig mest just nu: frontend, backend eller något helt annat?
2. Hur tror du att du själv kommer att reagera på skillnaden mellan skola och arbetsliv?
3. Vad skulle du vilja utveckla hos dig själv för att trivas bättre i rollen som programmerare?

⁸ Arbetsförmedlingen – Yrkesprognoser inom IT-2025

⁹ Hackerrank-Developer Skills Report-2025

Pierres tips – personliga råd från författaren

- ✓ Behåll glädjen i problemlösning och våga ställa frågor.
- ✓ Små egna projekt ger erfarenhet och självförtroende.
- ✓ Gör **AI** till din **hjälp**are, inte din **ersätt**are.
- ✓ Stressa inte över att kunna allt direkt, lärandet tar tid, även efter att du har avslutat dina studier eller tagit examen.

Kort och gott

Lärdom	Tillämpning i verkligheten
Arbetslivet skiljer sig mycket från skolan.	Var beredd på att hitta egna lösningar och samarbeta mer.
Ingen kan allt från början.	Våga växa steg för steg, använd resurser som dokumentation och kollegor.
Det finns många vägar i yrket.	Testa olika inriktningar tills du hittar det som passar dig bäst.
Arbetsmarknaden är stark för utvecklare.	Satsa på att visa dina kunskaper i projekt och kommunikation.

Interaktiva övningar



<https://lebario.se/library/dfj/fk>

Detta var bara början. I nästa kapitel och resten av boken går vi vidare med konkreta strategier, tydliga verktyg, praktiska råd, fler tips, interaktiva övningar och ljudinspelningar som hjälper dig hela vägen fram till ditt första jobb.