



Programuotojo karjera: kaip uždirbti daugiau pinigų

Turinys

Įvadas - ši knyga ne apie IT	4
Skyrius 1. Karjeros startas: ko ir kaip mokintis?	5
Kas sakė, kad programuoti lengva?	5
Kokią kalbą geriausia mokintis? Neteisingas klausimas.	6
Kaip pradėti savo kelią nuo nulio?	7
Kur gauti pirmąsias pamokas	8
Kur gauti pirmuosius darbus	9
Kodėl teorijos kalimas yra blogai	10
Kas sakė, kad amžius yra svarbus?	11
Kaip teisingai googlinti	12
Smalsumas ir tingumas - geriausios savybės	13
Kaip elgtis per darbo pokalbius	14
Kas geriau - dirbti nuolatinį darbą ar freelancinti?	15
Jums nepasiseks. Daugybę kartų.	16
Skyrius 2. Darbas ofise: kaip greičiau tobulėti?	17
Komandinis darbas vyksta komandoje	17
Kaip geriau išnaudoti laiką	18
Kodėl darbo reikia ieškoti, net jei jį turite	19
Kodėl verta turėti papildomų projektų	20
Kaip prašyti didesnės algos	21
Kaip išvengti konfliktų su kolegomis	22
Kaip išvengti konfliktų su vadovais	23
Kada laikas išeiti į kitą darbą?	24
Skyrius 3. Darbas sau: kaip gauti daugiau klientų?	25
Kodėl laisvai samdomas darbas yra ne visiems	25

Kaip ieškoti klientų	26
Rašykite blogą	27
Kaip rašyti pasiūlymus, kurių klientai negalės atsisakyti	28
Kaip nustatyti darbų kainą	29
Kaip žinoti, ar su klientu verta dirbti	30
Kodėl reikia galvoti ir apie šiandieną, ir apie vakar, ir apie rytojų	32
Kiek projektų vienu metu apsiimti?	33
Kaip optimizuoti savo darbą	34
Skyrius 4. Ateitis: į vadovus ar į savo verslą?	35
Kokia jūsų asmenybė - ar sugebate vadovauti?	35
Ką turi išmanyti CTO?	36
Kiek kainuoja sukurti savo startupą?	37
Kodėl geriau pradėti verslą iki 30ties	39
Kaip samdyti kitus programuotojus	40
Kaip (su)valdyti programuotojus	41
Kaip atleisti programuotojus	42
Freelancerių komanda	43
Skyrius 5. Interviu su kolegomis programuotojais	45
Skyrius 6. Pabaigai: 10 trumpų patarimų, kaip “save parduoti”	49
Trumpas apibendrinimas	55

Įvadas - ši knyga ne apie IT

Sveiki, gerbiamieji programuotojai ir tie, kas nori jais tapti. Ši knyga yra apie jūsų šviesią ateitį. Bet ne apie tai, kaip rašyti kodą, o apie tai, kaip išnaudoti savo gebėjimus ir uždirbti daugiau pinigų.

Bendraudamas su programuotojais Lietuvoje ir užsienyje, pastebiu ryškų skirtumą mūsų nenaudai - deja, mums trūksta gebėjimų. Bet ne techninių - IT specialistai mes labai geri. Spragos yra **verslo pusėje** - neturime pakankamai verslumo ir pardavimų gebėjimų, o to rezultatas - mažesni atlyginimai, užsakovai iš serijos "Clients from hell", projektai kurie užsitęsia dvigubai ilgiau. Tai bandysime pasiaiškinti, kas kaltas ir ką daryti.

Turbūt, reikia paaiškinti, kas aš toks kad galėčiau to mokyti? Tai trumpai - turiu 13 metų patirtį web-projektų kūrime įvairiais lygiais: pagrindinės pajamos ateina iš programavimo, bet buvau 4 metus projekto vadovu, 4 metus freelanceriu, dirbau dviejose Londono kompanijose ir pats esu sukūręs eilę mini-projektų bei verslų. Taip pat pastaruoju metu "užauginau" ne vieną programuotoją - tiek per savo projektą phpamokos.lt, tiek ir šiaip patarimais, jei kas kreipėsi pagalbos. Kitaip tariant, esu matęs visą procesą nuo-iki tuzinus kartų, ir galiu pasidalinti patirtimi.

Knyga padalinta į 6 skyrius: pirmi keturi skirti programuotojo karjeros etapams nuo pradinio iki viršūnių, tada penktame skyriuje patalpinti interviu su kolegomis programuotojais, kurie irgi yra šį tą pasiekę karjeroje, ir užbaigsime "tips'n'tricks" skyreliu su trumpais patarimais, kaip save geriau parduoti.

Skyrius 1. Karjeros startas: ko ir kaip mokintis?

Pradėkime nuo pradžių. Pirmas skyrelis skirtas tiems, kas tik pradeda programuoti ar dar tik nori pradėti. Šioje studijoje daugiau klausimų nei atsakymų. Ir labai didelė tikimybė pasukti ne tuo keliu, pradėti mokytis ne to ko reikia ar ne taip kaip yra efektyvu. Pabandyti padėti patarimais.

Kas sakė, kad programuoti lengva

Programuotojai ir IT specialistai vadinami "svajonių jaunikiais", jų algos dvigubai didesnės už šalies vidurkį, ir dar paklausa yra didžiulė, nes visiems verslams reikia IT sprendimų. Tai turėtų skatinti mokintis programuoti visus jaunuolius, ar ne? Bet aš matau kiek kitokią realybę.

Taip, jaunuoliai susižavi aukso kalnais ir bando. Mokinasi, skaito, rašo kodą. Bet iki realaus programuotojo, sugebančio pagaminti pilnavertį projektą - išauga labai nedaugelis. Pats gaunu nemažai laiškų su skundais - "per sudėtinga", "o gal patartumėte kaip greičiau/paprasčiau", "ar tikrai reikia mokėti tiek kalbų, galvojau užteks tik PHP" ir pan. Kursus ar pamokas pradeda daug, bet užbaigia mažiau nei dešimtadalis. Ir ne be priežasties.

Programavimas nėra tiesiog kodo rašymas, tai nėra gramatikos diktantai. Tai loginio mąstymo ir analizės reikalaujantis mokslas - prieš rašant kodą, reikia sugalvoti kokį ir kur, ir dar taip, kad veiktų ne tik teisingai, bet ir greitai bei saugiai. Pasak tyrimų, profesionalai prie kodo rašymo praleidžia vos 10-20% laiko, visa kita yra galvojimas, informacijos paieška, užduoties analizė.

Taigi jei karjeros pradžioje galvojate, kad užteks perskaityti kokią knygą, įdiegti WordPress ir jau galėsite kalti pinigus, teks nusileisti ant žemės. Programuotojams mokami dideli pinigai ne šiaip sau - o už tai, kad jie žino daug ir moka skirtingas kalbas ir technologijas pritaikyti pagal situaciją. O tokia patirtis **ateina tik su realių projektų kūrimu, kas reiškia ne mėnesius, o metus.**

Prigašdinau? Labai gerai. Toks ir buvo tikslas. Nes jei esate kelio pradžioje ir sprendžiate apie būsimą profesiją, tai reikia daryti nusiėmus rožinius akinius.

Kokią kalbą geriausia mokintis? Neteisingas klausimas.

Vienas dažniausių klausimų iš pradedančiųjų yra "**kokią kalbą geriausia mokintis**". Ir tai yra fundamentaliai neteisingas požiūris - klausama apie įrankį, nežinant tikslo. Čia kaip klausti "Su koku autobusu man geriausia važiuoti", nepatikslinant nei kur važiuojate, nei koks jūsų biudžetas.

Programavimas turi nemažai atšakų, pagal tai kam jis pritaikomas: galima kurti tinklalapius, žaidimus, mobilias programėles, sistemas Windows ar kitoms operacinėms sistemoms, kasos aparatų įrangą ir dar šimtus kitokių programinės įrangos tipų. Ir daugumoje atveju bus taikomos skirtingos programavimo kalbos ir technologijos.

Tad reikia apversti klausimą ir nustatyti tikslą - **ką būtent jūs norite kurti?** Ar pagal tai kas įdomiau, ar kas perspektyviau ateičiai, ar kas atneš daugiau pinigų (bet kaip taisyklė tada ir mokytis ilgiau/sunkiau). Ir tada einate klausinėti būtent tos srities specialistų ir draugų, kokių kalbų (taip, dažniausiai vienos neužtenka) reikės būtent ten. Tada ieškote, kur tų kalbų galima išmokti - ar interneto kursuose, ar universitete, ar pasamdžius mokytoją, ar yra super-naudingų knygų. Ir tada pirmyn prie tikslo!

Mano asmeninė patirtis - 11 klasėje pamačiau internetą. Tada informatikos mokytojas papasakojo trumpai apie HTML. Supratau, kad galiu KURTI internetą ir, kas svarbiausia, iš to uždirbti. Toks ir tapo tikslas - greitai kurti svetaines, kad galėčiau prisidurti kapeiką. Taip ir prasidėjo - HTML, po to CSS, po to lengviausia rezultata buvo pasiekti su PHP, ir tada pirmasis uždarbis iš užsakymų, o vėliau jau pirmasis oficialus darbas pirmame kurse. Labiau galvojau ne "kuri kalba geriausia", o "kaip greičiausiai galiu pasiekti tikslą - įsidarbinti ir uždirbti".

Kaip pradėti savo kelią nuo nulio?

Prisimenu kaip prieš 13 metų žengiau savo pirmuosius žingsnius programavime. Tai buvo mokyklinis Pascal, po to HTML. Buvo labai smalsu kaip viskas veikia, bet tuo pačiu buvo stiprus pasimetimo jausmas. Kur ieškoti informacijos, ko klausinėti, kuo tikėti. Juk nesinori švaistyti laiko kažkam, kas "ne į temą". Gerai kad turėjau tris informatikos mokytojus, kurie nukreipė teisinga kryptimi.

Šiaip aš pavydžiu dabartinei kartai - jūs turite internetą! Tiksliau, pats internetas buvo ir tada, bet informacijos kiekis buvo nepalyginamai mažesnis. Dabar gi, iš esmės, nebegalima "atsimazinti", kad neturiu pinigų knygai ar kad nemoku skaityti angliškai ar kad, khm, biblioteka sekmdienį nedirba (true story). **Visa informacija prieinama, tik reikia mokėti ją surasti ir atfiltruoti tinkamą.** Kas irgi nėra lengva, sutinku.

Taigi, pirmiausia, ko reikia, tai rasti tinkamos medžiagos ir tinkamiausią mokymosi būdą. Bet, iš principo, mokyti teks pačiam. Kažkada manęs paklausė - ar tiesa, kad dauguma programuotojų yra savamoksliai. Tikslių skaičių neturiu, bet esu įsitikinęs kad taip ir yra. Nes čia, kaip jokioje turbūt kitoje disciplinoje, reikia pačiam ne tik parašyti kodą, bet ir jį suprasti, kad parašytum dar kažką toliau. Tai mokytojas gali tik papasakoti pagrindus, kiek turi laiko, bet visą praktiką ir ugnį bei vandenį teks praeiti pačiam.

Tai pradėti reikia nuo nusiteikimo ilgam keliui. Kažką greitai išmokti ir minimalias programas sukurti įmanoma, bet kažkokiam realiam lygiui ir uždarbiui pasiekti reikės ilgų mėnesių ar metų. Ir pakeliui bus nesėkmių - kažkas nesigaus, kažkur strigsite, kažkur jums nesumokės ir t.t. Tai pirma pradžia turi būti jūsų ryžtas ir apsišarvojimas kantrybe.