

در شرکتتان برنامه‌ساز نیاز دارید، نه برنامه‌نویس

www.opatan.ir

info@opatan.ir

You need Developers, not Programmers

ایپاتان
رایانشی



موقعیت بسیار مهم است. پند یک مدیر ممکن است بی‌ارزش یا بدتر باشند اگر مناسب وضعیت شما نباشد. تصمیمی درست برای شرکتی بزرگ ممکن است برای کوچک‌تر کشنده باشد. پیش از آنکه پیشنهادهای مدیریتی هرکسی از جمله من را بشنوید 😊، مطمئن شوید که موقعیت خودتان را مدنظر دارید.

من یک شرکت نرم‌افزاری کوچک را می‌گردانم بدون هیچ کمک مالی بیرونی. شرکت ما (SourceGear) حدود ۶ سال سن و ۲۵ کارمند (در زمان نوشته شدن این مقاله: سال ۲۰۰۳) دارد.

در این مدت درس‌های بسیار جالبی فرا گرفته‌ام. یکی از چیزهایی که یادگرفته‌ام این است که شرکت‌های نرم‌افزاری کوچک نباید هیچ برنامه‌نویسی داشته باشند.

در این مقاله منظور از **برنامه‌نویس (Programmer)** فردی است که کاری جز نوشتن کدهای نو و (اگر خوش‌شانس باشید) اشکال‌زدایی، نمی‌کند. او مشخصه‌ها (**Specifications**) را نمی‌نویسد. او تست‌کیس‌های خودکار را نمی‌نویسد. کمکی برای اینکه سیستم ساخت خودکار بروز باشد، نمی‌کند. به مشتریان کمک نمی‌کند تا مشکلات جدی‌شان را حل کنند. در نوشتن مستندات کمک نمی‌کند. او حتا کُد را هم نمی‌خواند. تنها کاری که او می‌کند نوشتن کد جدید است.

در شرکت‌های کوچک نرم‌افزاری شما یک چنین آدم‌هایی را نمی‌خواهید.

به‌جای برنامه‌نویسان (کسانی متخصص نوشتن کد هستند) آنچه شما نیاز دارید **برنامه‌سازان (Developer)** هستند (کسانی که به روش‌های مختلف تلاش می‌کنند تا محصولی موفق ساخته شود).

اگر در تعریفم از واژه‌ها کمی گستاخ هستم، پوزش مرا بپذیرید، ولی اینکه من چه واژگانی را بکار می‌برم زیاد مهم نیست. در شرکت نرم‌افزاری کوچک شما توانایی مالی داشتن افرادی را که فکرمی‌کنند تنها کارشان نوشتن کُد است را ندارید. کارهای بسیار دیگری برای انجام‌دادن هستند و انجام همه آنها برای محصولی موفق داشتن، اهمیت دارد. اگر شرکتی بزرگ بودید، می‌توانستید برای هرکاری متخصصان آن کار را بگیرید. اما چون کوچک هستید، افراد کمتری نیاز دارید اما کسانی که همه‌فن‌حریف‌تر باشند.

~ ~ ~

حدود (وظایف و اختیارات) در برابر انعطاف‌پذیری

این تفاوت به‌راستی مهم بین شرکت‌های کوچک و بزرگ است:

- در شرکت کوچک بیشتر افراد نقش‌های مختلف دارند. امکان‌پذیر نیست که فردی داشته‌باشید که تنها روی یک زمینه تمرکز می‌کند. شرکت‌های کوچک افراد

همه فن حریفی را نیاز دارند که به شرکت اهمیت می دهند و برای هرکاری که به شرکت کمک کند، پیش قدم می شوند. حسابدار یا کتابدار هرکاری را سر-و-سامان می دهند. معمولاً کسانی وجود دارند که همه کارها را انجام می دهند و سرشان همیشه شلوغ است و هیچ کس هم نمی داند که آنها چه می کنند. در اینجا مفهوم کلیدی انعطاف پذیری است.

• شرکت های بزرگ متخصصان بیشتری دارند. دستمزدها با «حساب های دریافتی» فرق می کند که آن هم از «حساب های پرداختی» جداست. معماران نرم افزار، طراحی می کنند. برنامه نویسان کد می نویسند. مدیران فنی برنامه نویسان را مدیریت می کنند. مدیران پروژه (برنامه) ها مشخصه و زمانبندی را پیگیری می کنند. مدیران محصول جایگاه سازی و پیام رسانی می کنند. مبلغان، خوب، کسی به راستی نمی داند که آنها چه می کنند. به هر حال هرکسی کاری مشخص و تعریف شده دارد. مفهوم کلیدی اینجا احترام به حدود (وظایف و اختیارات) است.

راستی که این دو، فرهنگ هایی به کلی متفاوت هستند و اگر این دو با هم تداخل داشته باشند ممکن است چیزهای زشتی رخ دهد. انعطاف پذیری و حدود با هم جور در نمی آیند. کسی که در یکی از این فرهنگ ها موفق بوده، اغلب هنگامی که به دیگری گذار می کند، کارش بیخ پیدا می کند.

~ ~ ~

برنامه سازان

در شرکت کوچک نرم افزاری هر برنامه سازی پیش و بیش از هر چیز برنامه نویس است. بخش اصلی وقتش باید صرف نوشتن کد و اشکال زدایی گردد. اما نیاز است که برنامه سازان در دیگر زمینه ها هم کمک کنند، زمینه هایی چون:

- مستندات مشخصه ها
- مدیریت پیکربندی
- بازیابی کد
- تست کردن
- تست های خودکار
- مستندسازی
- حل مشکلات جدی مشتریان

در واژگان من، این کارها برای برنامه ساز و برنامه نویس فرق می کنند. برنامه ساز بینش و چشم اندازی بزرگتر دارد و توانایی آنرا دارد که دیدی بزرگتر (از موضوع) داشته باشد. برنامه نویس کد می نویسد، آنرا به تست کننده ها می سپارد و منتظر می ماند تا آنها اشکالها را بیابند. برنامه ساز می داند که بهتر است هم اکنون اشکال زدایی کند، زیرا ممکن است او در آینده همان کسی باشد که با مشتری درباره آن محصول صحبت می کند.

زمانیکه تیم برنامه‌نویسان شما به مرور برنامه‌ساز می‌شوند، شاید سلسله مراتب کاری شما تغییر کند. ممکن است که بهترین برنامه‌ساز شما همان کسی نباشد که مشکلات به‌راستی سخت را برطرف می‌سازد. یک برنامه‌نویس با استعداد فراوان می‌تواند برنامه‌سازی بسیار بد باشد. گدنویسی بخشی لازم اما ناکافی برای برنامه‌ساز بودن است. همین‌گونه، افراد با استعداد کمتر در تیم شما هنوز می‌توانند خودشان را به عنوان برنامه‌سازانی عالی نشان دهند با مشارکتی که در بخش‌های غیر گدنویسی محصول می‌کنند.

~ ~ ~

پرسشهای پُرسامد (متداول)

می‌توانیم برنامه‌نویسان خود را فقط به گدنویسی بگماریم و برای همه دیگر کارها گروهی دیگر را بکار بگیریم؟

شاید، شرکت ما پیش‌تر این کار را کرده‌است و گاهی خوب جواب داده است. اما در درازمدت شما، از این تصمیم که برنامه‌نویسانتان را از همه‌چیز جز برنامه‌نویسی دور نگاه داشته‌اید، پشیمان می‌شوید. برنامه‌نویسان در یک شرکت نرم‌افزاری کوچک بیشتر از آن موثر هستند که به آنها چشم‌اندازی باریک بدهیم. به آنها اجازه بدهید که چشم‌انداز کاربر را ببینند. برنامه‌نویسانتان را پشت تلفن بگذارید تا به مشتریان در رفع مشکلی جدی کمک کنند. همین‌طور که برنامه‌نویسان را مجبور می‌سازید اشکال‌هایی را که خود ایجاد کرده‌اند را رفع کنند، کیفیت محصول شما بهبود پیدا خواهد کرد.

برنامه‌نویسان ما نمی‌دانند که چگونه همه این کارهای دیگر را انجام دهند؟

راستی؟ مطمئن هستم که دانشکده‌های کامپیوتر ابزارهای مدیریت نسخه نرم‌افزار (SCM) و مهارت‌های پشتیبانی فنی را درس می‌دهند، درست؟ 😊

گرچه دانشکده من این چیزها را یاد نمی‌داد. روشن است که برنامه‌نویسان شما نمی‌دانند که چگونه جنبه‌های جز-گدنویسی ساخت محصول را انجام دهند. به آنها آموزش دهید.

آموزش رسمی ممکن است نقشی ایفا کند، اما بهترین یادگیری از تجربه می‌آید. برنامه‌سازان شما باید پند شرکت نایکی (Nike) «انجامش بده» را بکار بندید. مطمئن شوید که آنها فرصت اشتباه داده‌اید. به آنها اجازه دهید از اشتباهاتشان یادگیرند.

برنامه‌نویسان ما نمی‌خواهند همه این کارها را انجام دهند

بعضی آدمها نمی‌خواهند برنامه‌ساز باشند، آنها می‌خواهند برنامه‌نویس باشند. اشکالی ندارد. برنامه‌نویس می‌تواند کار خوبی پیدا کند. اما شما و آن برنامه‌نویس می‌توانید بنشینید و درباره اینکه آیا شرکت شما محیط مناسبی برای کار او هست یا نه، بی‌پرده سخن بگویید.

سرانجام با توجه به موقعیت خود می‌توانید تصمیم بگیرید که در این باره چه کنید. پیشنهاد من این است که هر برنامه‌نویسی در شرکت نرم‌افزاری کوچک نیاز دارد که در چیزی فراتر از کدنویسی دخالت داشته باشد، البته تمام قاعده‌ها استثنا هم دارند.

آیا یافتن برنامه‌سازی که مهارت‌هایی اینچنین گوناگون دارد، سخت نیست؟

بله هست. برنامه‌سازان واقعی و راستین ارزشمند هستند. احتمالاً شما نمی‌توانید آنها را از بیرون بکارگیرید. مجبور هستید که به برنامه‌نویسان کمک کنید که برنامه‌ساز شوند.

زمانیکه شدند، باید بسیار تلاش کنید که آنها را نگه‌دارید. هنگامیکه برنامه‌نویسی از شرکت جدا می‌شود یا در شرکتی بزرگ شغل مدیریتی می‌گیرد یا شرکت خود را برپا می‌کند.

چگونه برنامه‌سازان گد بنویسند در حالیکه پیوسته با کارهای دیگر در کارشان وقفه می‌اندازند؟

بله، این مساله‌ای است. از این فرصت بهره می‌جویم تا از اصل اریک (خودم) در مدیریت نرم‌افزار یاد کنم:

نمی‌توانید مسایل را حذف کنید،
اما می‌توانید بده-بستانی انجام دهید
مسایلی که هم‌اکنون دارید را بدهید
در برابر آنها مسایلی را که ترجیح می‌دهید، بستانید.

برنامه‌سازان وقت آزاد بیشتری نیاز دارند وگرنه هیچ‌کدی نخواهند نوشت. خوب این مساله‌ای است که ترجیح دارد بر مسایلی که برایتان پیش خواهد آمد هنگامی که با افرادی با چشم‌انداز باریک «تنها گد» سروکار پیدا می‌کنید. تلاش کنید زمان خود را ساماندهی کنید. برای نمونه در یک روز هم‌زمان کدنویسی و پشتیبانی فنی را انجام ندهید.

پس شما می‌گویید که برنامه‌سازان ما مجبورند همه دیگر کارها را انجام دهند؟

خیر. بهتر این است که برنامه‌سازان هنوز بیشتر (اما نه همه) وقتشان را صرف نوشتن گد کنند. اگر بودجه شما اجازه می‌دهد دسته‌های گوناگونی از متخصصان هستند که می‌توانید بکار بگیرید:

پشتیبانی فنی «سطح یک»

بیشتر کار پشتیبانی فنی شامل پاسخگویی پیاپی به پرسشهای یکسان است. احتمالاً اینکار بهترین کاربری از وقت یک برنامه‌ساز نیست. برنامه‌ساز را در پشتیبانی فنی «سطح دو» درگیر کنید؛ تشخیص و حل مشکلات سخت و پیچیده مشتریان. اما اینکه یک یا دو متخصص برای رسیدگی به کارهای ساده‌تر داشته باشید، ایده خوبی است. برای اینکار دنبال فردی بگردید که

مهارت‌های ارتباطی خوبی دارد و پیش‌زمینه‌ای هم در مسایل فنی.
تستِ دستی، بازیابی اشکال‌زدایی و ...

همینطور که یک تیم پشتیبانی فنی سطح یک می‌سازید، برای این تیم کمی بیش از نیازتان کارمند بگیرید. به آنها کمی هم وقت آزاد بدهید تا بتوانند با مقداری مشخص از کارهای پرسش/پاسخ درگیر شوند. بگذارید آنها مقداری از کار تستِ دستی و بازیابی اشکال‌زدایی را انجام دهند. این کار از بارکاری برنامه‌سازان شما خواهد کاست و نرخ ازپای درآمدن افرادی که در جایگاه پشتیبانی فنی محض هستند را کاهش می‌دهد.

مستندسازی

برنامه‌سازانتان را مجبور کنید که مشخصه‌ها یا پیش‌نویس نخست محتوا را بنویسند، اما آماده‌سازی و ویرایش نهایی و ویرایش مستندات شما کاری است که برای کسیکه می‌تواند بنویسد مناسب است.

مدیر سیستم (System Administration)

احتمالا برنامه‌سازان شما می‌توانند کار یک مدیر سیستم را انجام دهند، اما هیچ دلیل خاصی مرتبط با محصول برای اینکه آنها این کار را نکنند وجود ندارد. بکارگیری یک مدیر سیستم روشی است برای کاستن از دقت‌های برنامه‌سازان.

این نوشته برگردانی است از:

Sink, Eric. "Small ISVs: You need Developers, not Programmers"