

# اسطوره های نرم افزاری

بسیاری از دردهای نرم افزاری را می توان در اسطوره هایی جست که طی نخستین روزهای تکوین نرم افزارها پدید آمد. برخلاف اسطوره های باستانی که غالباً درس عبرت به بشر می آموزند، اسطوره های نرم افزاری باعث اطلاع رسانی نادرست و سردرگمی شده اند. اسطوره های نرم افزاری دارای چند ویژگی هستند که آنها را زیانبار ساخته اند؛ برای نمونه، به ظاهر بیانی منطقی از واقعیتها بوده اند (گاه چند عنصر واقعی در آنها وجود دارد) ولی دارای احساسی نبوغ آمیز بوده غالباً توسط برنامه نویسان کارآموده ای که «قدر می دانند» به آگاهی عموم می رسند.

**اسطوره های مدیریتی.** مدیرانی که مسئولیت نرم افزاری دارند، همانند مدیران دیگر، غالباً تحت فشار کاهش هزینه ها، جلوگیری از بی برنامه گی و بهبود بخشیدن به کیفیت هستند. مدیر نرم افزاری همانند غریقی که به هر چیزی دست می اندازد، غالباً به اسطوره های نرم افزاری اعتقاد پیدا می کند، اگر بداند این اعتقاد باعث کاهش فشار می شود (حتی به طور موقت).

**اسطوره.** ما از قبل کتابی داریم که آکنده از استانداردها و روالهای لازم برای ساختن نرم افزارهاست. آیا این، آنچه را که افراد من باید بدانند در اختیارشان قرار نخواهد داد؟  
**واقعیت.** ممکن است کتاب استانداردهای خیلی خوبی وجود داشته باشد، ولی آیا از آن استفاده می شود؟ آیا سازندگان نرم افزار از وجود آن آگاهند؟ آیا مهندسی نرم افزار نوین را ارائه می دهد؟ آیا کامل است؟ آیا آنقدر روان هست که زمان تحویل را بهبود بخشد و در عین حال کیفیت را حفظ کند؟ در بسیاری از موارد، پاسخ اکثر این پرسشها «خیر» است.

**اسطوره.** افراد من ابزارهای نرم افزارسازی حرفه ای را دارند؛ به علاوه جدیدترین کامپیوترها را برای آنها خواهیم خرید.

**واقعیت.** برای ساخت نرم افزارهایی با کیفیت بالا، فقط داشتن آخرین مدل کامپیوتر یا ایستگاه کاری کافی نیست. برای رسیدن به بهره وری و کیفیت خوب، ابزارهای مهندسی نرم افزار به کمک کامپیوتر (CASE) اهمیت بیشتری از سخت افزار دارند. با این وجود، اکثر نرم افزار نویسان باز هم از آنها استفاده مؤثر به عمل نمی آورند.

**اسطوره.** اگر از برنامه عقب بیفتیم، می‌توانیم بر تعداد نویسان بیفزاییم و عقب افتادگی را جبران کنیم (این وضعیت را گاه «horde مغولی» می‌گویند).

واقعیت. ایجاد نرم افزار، یک فرآیند مکانیکی نظیر ساخت تولیدات معمولی نیست. به قول بروکز [BRO75]: «... با افزودن افراد دست اندرکار به نرم افزاری که تأخیر دارد، بر میزان تأخیر آن افزوده خواهد شد». در نگاه نخست ممکن است این گفته خلاف منطق به نظر برسد، ولی با از راه رسیدن افراد جدید، افراد قدیمی باید زمانی را صرف آموزش آنها کنند و در نتیجه زمانی که باید صرف کار روی نرم افزار شود، هدر می‌رود. اضافه کردن افراد، عملی است ولی به شیوه‌ای هماهنگ و با برنامه ریزی منظم.

**اسطوره های مشتریان.** مشتری که درخواست یک نرم افزار کامپیوتری دارد، ممکن است پشت میز کناری باشد، یک گروه تکنیکی در آن سوی سالن باشد، بخش فروش و بازاریابی باشد، یا یک شرکت دیگر باشد که قراردادی برای نرم افزار منعقد نموده است. در بسیاری موارد، مشتری به اسطوره هایی درباره نرم افزارها اعتقاد دارد، زیرا مدیران نرم افزار و سازندگان آن کمتر سعی در برطرف کردن سوء تفاهم‌ها دارند. این اسطوره منجر به انتظارات نادرست (از جانب مشتری) و در نهایت عدم رضایت از سازنده می‌شود.

**اسطوره.** نیازهای پروژه پیوسته در حال تغییر است، ولی این تغییرات را به راحتی می‌توان در نرم افزار جای داد زیرا نرم افزار انعطاف پذیر است.

**واقعیت.** این درست است که نیازمندی‌های نرم افزار تغییر می‌کند ولی تأثیر تغییر به زمان اعمال تغییر بستگی دارد.

اگر به تعریف صریح توجه جدی شود، درخواستهای اولیه برای تغییر را به راحتی می‌توان پاسخ گفت. مشتری می‌تواند نیازمندی‌ها را مرور کند و اصلاحاتی را با تأثیر نسبتاً کم بر هزینه‌ها توصیه کند. هنگامی که تغییرات در اثنای طراحی نرم افزار درخواست می‌شوند، هزینه‌ها به سرعت بالا می‌رود. منابع مصرف شده‌اند و یک چارچوب طراحی مشخص شده است. تغییر می‌تواند باعث تغییرات مشکل آفرینی شود که نیاز به منابع اضافی و اصلاح اساسی طراحی دارد و این یعنی بالا رفتن هزینه‌ها. تغییرات در عملکرد، کارایی، واسطه‌ها، یا

ویژگیهای دیگر در اثنای پیاده سازی (دستورها و آزمایش) اثری شدید بر هزینه دارد. تغییر، در صورتی که نرم افزار به مرحله استفاده رسید، هزینه‌ای به مراتب بالاتر خواهد داشت.

**اسطوره های سازندگان.** اسطوره‌هایی که نرم افزارنویسان باور دارند، نتیجه ۵۰ سال فرهنگ برنامه نویسی است. در نخستین دهه‌های ساخت نرم افزار، برنامه نویسی شکلی از هنر پنداشته می‌شد. سنتهای قدیمی دیر از بین می‌روند.

**اسطوره.** هنگامی که برنامه را نوشتیم و برنامه کار کرد، دیگر کار تمام است.

**واقعیت.** یک بار کسی گفته بود: «هر چه زودتر دست به کار نوشتن دستوره‌های برنامه شوید، زمان بیشتری صرف به

**پایان بردن آن خواهد کرد».**

**اسطوره.** تا هنگامی که برنامه را «اجرا» نکرده‌ام، راهی برای ارزیابی کیفیت آن ندارم.

**واقعیت.** یکی از مؤثرترین راهکارهای تضمین کیفیت نرم افزار از زمان آغاز پروژه قابل اجراست- یعنی **مرور تکنیکی رسمی.** مرور نرم افزار یک فیلتر کیفیتی است که از آزمایش نرم افزار برای یافتن گروههای معینی از معایب نرم افزاری مؤثرتر است.

**اسطوره.** تنها چیز قابل تحویل برای یک پروژه موفق برنامه‌ای است که کار کند.

**واقعیت.** برنامه‌ای که کار می کند فقط بخشی از پیکربندی نرم افزار است که شامل عناصر فراوان می شود. مستندسازی، بنیاد مهندسی موفق بوده و مهمتر از آن راهنمایی برای پشتیبانی نرم افزار فراهم می آورد.