

# بسمه تعالی

درس مهندسی نرم افزار ۱  
نیمسال اول ۹۳-۹۲

مرضیه سادات طباطبایی

# فصل چهارم

اصول راهنما در مهندسی نرم افزار

# نگاهی گذرا

- از یک دیدگاه کلی، کار مهندسی نرم افزار به مجموعه ای از مفاهیم، اصول، روشها و ابزارها گفته می شود که یک مهندس نرم افزار در طول روز با آنها سر و کار دارد.
- نیمه عمر دانش توسعه نرم افزار (دانشهای مربوط به فن آوری)، سه سال است. نیمی از آنچه امروز می دانید سه سال بعد دیگر به کارتان نمی آید.
- اما در مورد نوع دیگری از دانش توسعه نرم افزار که از آن به عنوان «اصول مهندسی نرم افزار» یاد می شود، اینطور نیست و این اصول در سرتاسر زندگی یک مهندس نرم افزار به او کمک خواهد کرد.

# اصول هسته ای

- مجموعه ای از اصول هسته ای وجود دارد که راهنمای مهندسی نرم افزار است و به استفاده از یک فرایند نرم افزار با معنا و اجرای اثربخش روش های مهندسی نرم افزار کمک می کند.
- برخی از این اصول عبارتند از:
  - فراهم ساختن ارزش برای کاربر نهایی
  - حفظ سادگی
  - حفظ چشم انداز محصول و پروژه
  - دانستن این مطلب که دیگران آنچه را که شما ساخته اید مصرف می کنند و باید بتوانند آن را درک کنند
  - نگاه به آینده
  - برنامه ریزی برای استفاده مجدد
  - تفکر

# اصول راهنمای فرآیند مهندسی

- اصل ۱: چابک باشید.
- اصل ۲: در هر مرحله کیفیت را در کانون توجه قرار دهید.
- اصل ۳: آمادگی انطباق را داشته باشید و انعطاف پذیر باشید.
- اصل ۴: تیمی اثربخش و خودسازمانده تشکیل دهید.
- اصل ۵: سازوکارهایی برای برقراری ارتباط و هماهنگی برقرار کنید.
- اصل ۶: مدیریت تغییرات
- اصل ۷: ارزیابی ریسک
- اصل ۸: ایجاد محصولات کاری که برای دیگران ارزش فراهم می کنند.

# اصول راهنمای کار مهندسی نرم افزار

- برای دستیابی به هدف کار مهندسی نرم افزار که تحویل به موقع محصول است اصول فنی زیر را در نظر داشته باشید:
- اصل ۱: تقسیم و حل؛ یک مسئله بزرگ را به مجموعه ای از مسائل کوچک تقسیم کنید تا حل آن راحتتر شود.
- اصل ۲: درک بکارگیری انتزاع ها؛ انتزاع شکل ساده از یک سیستم پیچیده است.
- اصل ۳: تلاش برای سازگاری؛ مثلاً سازگاری رنگ، ظاهر و گزینه های منو در طراحی واسط کاربر
- اصل ۴: توجه ویژه به انتقال اطلاعات؛ کار نرم افزار انتقال دادن اطلاعات است.
- اصل ۵: توسعه نرم افزاری که ساختار پیمانه ای اثربخش داشته باشد؛ هر سیستم پیچیده را به چند پیمانه (مولفه) به طور اثربخش طراحی کنید.

# اصول راهنمای کار مهندسی نرم افزار

- اصل ۶: جستجو به دنبال الگوها؛
- اصل ۷: هرگاه که امکان دارد مسئله و راهکار آن را از چند دیدگاه متفاوت به نمایش بگذارید. مثلاً یک مدل از خواسته ها را می توان با به بکارگیری دیدگاهی داده گرا، عملیات گرا، رفتار گرا، به نمایش گذاشت.
- اصل ۸: به خاطر داشته باشید که نرم افزار را نگهداری خواهید کرد.

# اصول راهنمای فعالیتهای چارچوبی

- اصول ارتباطی
- اصول برنامه ریزی
- اصول مدلسازی
  - اصول مدلسازی خواسته ها
  - اصول مدلسازی طراحی
- اصول ساخت
  - اصول آماده سازی
  - اصول برنامه نویسی
  - اصول آزمون
- اصول استقرار



# اصول ارتباطی

- اصل ۱: گوش سپردن
- اصل ۲: خود را قبل از برقراری ارتباط آماده کنید.
- اصل ۳: یک نفر باید در جلسه ارتباطی فرایند ارتباط را مدیریت کند.
- اصل ۴: بهترین راه ارتباط رودررو است.
- اصل ۵: یادداشت بردارید و تصمیم گیری را مستند کنید.
- اصل ۶: تلاش برای همکاری
- اصل ۷: توجه خود را معطوف کنید. بحث خود را پیمان‌ه ای کنید.
- اصل ۸: اگر چیزی واضح نبود یک تصویر را بکشید.

# اصول ارتباطی

## • اصل ۹

– هنگامی که بر سر مبحثی به توافق رسیدید به مبحث دیگر پردازید.

– اگر به توافق نرسیدید به مبحث دیگر پردازید.

– اگر ویژگی یا قابلیت واضح نیست و نمی توان در حال حاضر آن را واضح کرد باز هم به مبحث دیگر پردازید.

• اصل ۱۰ : مذاکره یک مسابقه یا بازی نیست، وقتی بهترین نتیجه را می دهد که هر دو طرف برنده باشند.

# مشتری و کاربر نهایی

- مشتری
  - درخواست می کند نرم افزاری ساخته شود.
  - اهداف تجاری کلی برای نرم افزار را تعیین می کند.
  - خواسته های پایه را فراهم می سازد.
  - بودجه پروژه را تامین می کند.
- کاربر نهایی
  - نرم افزار ساخته شده را واقعا مورد استفاده قرار می دهد.
  - جزئیات عملیاتی نرم افزار را تعیین می کند.
- در بسیاری از پروژه ها مشتری و کاربر نهایی افرادی متفاوت اند که برای مدیران متفاوت و در سازمانهای متفاوت کار می کنند.

# اصول برنامه ریزی

- اصل ۱: شناخت حوزه پروژه؛ اگر ندانید مقصد کجاست استفاده از نقشه راه غیرممکن است.
- اصل ۲: طرف های ذینفع را در فعالیت برنامه ریزی دخالت دهید.
- اصل ۳: این را بدانید که برنامه ریزی ماهیتی متغیر و منعطف دارد.
- اصل ۴: برآوردهای خود را بر اساس آنچه که میدانید و به آن اطمینان دارید انجام دهید.
- اصل ۵: همزمان با برنامه ریزی، ریسک را هم در نظر بگیرید.
- اصل ۶: واقع بین باشید. مردم هر روز صددرصد کار نمی کنند.

# اصول برنامه ریزی

- اصل ۷: هنگام تعریف برنامه ریزی گرانولیتة را تعیین کنید. منظور از گرانولیتة سطحی از جزئیات است که در برنامه ریزی پروژه به آن پرداخته می شود.
- اصل ۸: تعیین کنید که چگونه می خواهید از کیفیت محصول اطمینان یابید. (بازبینیهای کیفی یا برنامه نویسی جفتی)
- اصل ۹: چگونگی انجام دادن تغییرات را شرح دهید.
- اصل ۱۰: برنامه ریزی را به وفور پیگیری کنید و در صورت نیاز تنظیماتی به عمل آورید.

# اصول مدلسازی

- مدلها برای درک بهتر یک موجودیت واقعی ساخته می شوند.
- در مهندسی نرم افزار دو نوع مدل ساخته می شود:
  - مدل‌های خواسته‌ها (مدل تحلیل): بیان خواسته‌های مشتری در ۳ قالب
    - دامنه اطلاعاتی
    - دامنه عملیاتی
    - دامنه رفتاری
  - مدل‌های طراحی
    - معماری
    - واسط کاربر
    - جزئیات مولفه‌ها
- مدل خواسته‌ها، خواسته‌های مشتری به نمایش می‌گذارد. مدل طراحی یکسری مشخصات معین برای ساخت نرم افزار ارائه می‌دهد.
- هدف و مقصود هر مدل، انتقال اطلاعات است. برای رسیدن به این هدف، از قالبی سازگار استفاده کنید. فرض کنید شما برای توضیح دادن مدل حضور ندارید. پس آن را طوری بسازید که به حضور شما نیاز نداشته باشد.

# اصول مدلسازی

- اصل ۱: هدف اصلی تیم نرم افزاری ساخت نرم افزار است نه ایجاد مدل. مدل هایی را انتخاب کنید که به رساندن نرم افزار به مشتری در سریعترین زمان ممکن کمک کند.
- اصل ۲: سبک بار سفر کنید. یعنی مدل هایی بیش از نیاز خود ایجاد نکنید.
- اصل ۳: بکوشید ساده ترین مدلی را بسازید که مسئله یا نرم افزار را توصیف کند.
- اصل ۴: مدل ها را طوری بسازید که قابل تغییر باشند.
- اصل ۵: توانایی بیان صریح هدف هر مدل ایجاد شده را داشته باشید.

# اصول مدلسازی

- اصل ۶: مدل هایی را توسعه دهید که با سیستم مورد نظر مطابقت دارد.
- اصل ۷: سعی کنید مدل های مفید بسازید ولی ساخت مدل های کامل را فراموش کنید.
- اصل ۸: در مورد قالب و نحو مدل، تعصب به خرج ندهید. اگر در انتقال مفاهیم موفق است نمایش در مرحله دوم اهمیت قرار دارد.
- اصل ۹: اگر گزینه شما می گوید مدلی درست نیست، هر چند که روی کار درست به نظر می رسد احتمالاً دلیلی برای این نگرانی دارید، پس به گزینه خود اطمینان کنید.
- اصل ۱۰: به محض اینکه توانستید بازخورد بگیرید.



## اصول مدلسازی خواسته ها

- اصل ۱: دامنه اطلاعاتی یک مسئله، باید نمایش داده و درک شود.
- اصل ۲: عملکردهای نرم افزار باید تعریف شوند.
- اصل ۳: رفتار نرم افزار باید نمایش داده شود.
- اصل ۴: مدل هایی که اطلاعات، قابلیت ها و رفتارها را تصویر می کنند باید به شیوه ای تقسیم بندی شوند که جزئیات را به گونه ای لایه ای یا سلسله مراتبی نمایش دهند.
- اصل ۵: وظیفه تحلیل باید از اطلاعات ضروری به سمت جزئیات پیاده سازی حرکت کند.

# اصول مدلسازی طراحی

- اصل ۱: طراحی باید تا مدل خواسته قابل ردگیری باشد.
- اصل ۲: همواره معماری سیستمی را که قرار است ساخته شود در نظر داشته باشید. معماری اسکلت سیستم است.
- اصل ۳: طراحی داده‌ها به اندازه طراحی عملکردها اهمیت دارد.
- اصل ۴: واسط‌ها (چه درونی و چه بیرونی) باید با احتیاط طراحی شوند.
- اصل ۵: طراحی واسط کاربر باید مطابق با نیازهای کاربر نهایی تنظیم گردد. واسط کاربر نمود آشکار نرم افزار است.
- اصل ۶: طراحی در سطح مؤلفه‌ها باید مستقل از عملکرد باشد. (استقلال عملیاتی)

# اصول مدلسازی طراحی

- اصل ۷: ارتباط مؤلفه ها با یکدیگر و با محیط خارجی باید در سطحی منطقی حفظ گردد.
- اصل ۸: نمایش های (مدل های) طراحی باید به آسانی قابل درک باشند.
- اصل ۹: طراحی باید به صورت تکراری توسعه یابد، در هر دوره از تکرار طراحی باید بکوشد تا سادگی بیشتر شود.

# اصول ساخت

- فعالیت ساخت شامل مجموعه ای از وظایف کدنویسی و آزمون می شود که نتیجه آن نرم افزاری عملیاتی و آماده تحویل به مشتری است.

- به همین دلیل اصول ساخت را در سه سطح زیر بررسی می کنیم:

– اصول آماده سازی

– اصول برنامه نویسی

– اصول اعتبارسنجی

# اصول آماده سازی

● پیش از آنکه حتی یک خط برنامه بنویسید اطمینان حاصل کنید که:

- می دانید که چه مسئله ای را قرار است حل کنید.
- اصول و مفاهیم طراحی پایه را می دانید.
- زبانی برای برنامه نویسی انتخاب کنید که نیازهای نرم افزار و محیطی را که قرار است در آن کار کند برآورده سازد.
- محیطی برای برنامه نویسی انتخاب کنید که ابزارهای لازم برای آسان تر کردن کار را در اختیارتان قرار دهد.

# اصول برنامه نویسی

- با شروع به کد نویسی، اطمینان حاصل کنید که:
  - استفاده از برنامه نویسی جفتی را در نظر داشته باشید.
  - ساختمان داده هایی را انتخاب کنید که نیازهای طراحی را برآورده کند.
  - معماری نرم افزار را بشناسید و واسطه هایی سازگار با آن بسازید.
  - منطق شرطی را تا حد امکان ساده نگه دارید.
  - حلقه های تودرتو را به شیوه ای بنویسید که به آسانی قابل آزمون باشد.
  - نام های با معنا برای متغیرها انتخاب کنید.
  - کدهای خود را مستند سازی کنید.
  - یک چیدمان بصری ایجاد کنید. مثلاً با تورفتگی و خطوط خالی که به فهم کدهای شما کمک کند.

## قواعد آزمون

- قاعده ۱: آزمون فرایند اجرای برنامه به قصد یافتن خطاهاست.
- قاعده ۲: یک مورد آزمون خوب باید خطاهای کشف نشده را با احتمال زیادی کشف کند.
- قاعده ۳: آزمون موفق، آزمونی است که خطای کشف نشده تا کنون را کشف کند.

# اصول آزمون

- اصل ۱: همه آزمون ها تا خواسته های مشتری قابل رهگیری باشند زیرا هدف از آزمون نرم افزار، کشف خطاهاست.
- اصل ۲: آزمون ها را باید مدتها قبل از شروع آزمون، برنامه ریزی کرد.
- اصل ۳: اصل پارتو در آزمون نرم افزار کاربرد دارد، یعنی اثر ۸۰ درصد از همه خطاهای کشف شده طی آزمون را احتمالاً در ۲۰ درصد از کل مؤلفه های نرم افزار می توان پیدا کرد.
- اصل ۴: آزمون باید در مقیاس کوچک آغاز شود و به سمت مقیاس بزرگ پیش رود.
- اصل ۵: آزمون کامل امکان پذیر نیست.



# اصول استقرار

- اصل ۱: انتظارات مشتری برای نرم افزار باید مدیریت شود.
- اصل ۲: پکیج تحویل کامل باید مونتاژ و آزمایش شود.
- اصل ۳: پیش از تحویل نرم افزار یک روال پشتیبانی باید مشخص کرد.
- اصل ۴: مواد آموزشی مناسب باید برای کاربران نهایی تهیه شود.
- اصل ۵: نرم افزار مشکل دار، ابتدا باید اصلاح و بعداً تحویل داده شود.