



طراحی شیء‌گرای سیستم‌ها (۴۰۴۸۴) Object-Oriented Systems Design

طراح درس: رامان رامسین

مقطع: کارشناسی	گروه: نرم‌افزار
نوع درس: تخصصی	واحد: ۳
هم‌نیاز: -	پیش‌نیاز: تحلیل و طراحی سیستم‌ها

اهداف درس

هدف این درس، آشنا کردن دانشجویان کارشناسی نرم‌افزار با مفاهیم، اصول و روش‌های تحلیل و طراحی شیء‌گرای سیستم‌های نرم‌افزاری است. دانشجویان ضمن آشنایی کامل با یک متدولوژی مطرح تحلیل و طراحی شیء‌گرا (نسل سوم)، با الگوهای طراحی GoF و چگونگی بکارگیری آنها نیز آشنا خواهند شد.

ریز مواد

(۱) مقدمه - مروری بر شیء‌گرایی و معرفی تاریخچه تکاملی تحلیل و طراحی شیء‌گرا (۱ جلسه - مدت هر جلسه، نود دقیقه است)

(۲) مروری بر زبان مدل‌سازی یکپارچه (UML) (۴ جلسه)

(۳) مراحل و جریان‌های کاری در USDP

(a) مراحل چهارگانه (۳ جلسه)

(b) جریان کاری خواسته‌ها - شناسایی و مدل‌سازی موارد کاربرد (۳ جلسه)

(c) جریان کاری تحلیل

(i) شناسایی و مدل‌سازی اشیاء و کلاس‌های تحلیل (۲ جلسه)

(ii) شناسایی و مدل‌سازی روابط بین اشیاء و کلاس‌های تحلیل (۲ جلسه)

(iii) بسته‌های تحلیل (۱ جلسه)

(iv) محقق‌سازی موارد کاربرد در تحلیل (۲ جلسه)

(v) مدل‌سازی فعالیت‌ها (۲ جلسه)

(d) جریان کاری طراحی

(i) شناسایی و مدل‌سازی اشیاء و کلاس‌های طراحی (۱ جلسه)

(ii) پالایش روابط (۱ جلسه)

(iii) واسط‌ها و مؤلفه‌ها (۱ جلسه)

(iv) محقق‌سازی موارد کاربرد در طراحی (۱ جلسه)

(e) جریان کاری پیاده‌سازی (۱ جلسه)

(۴) الگوهای طراحی

(a) اصول و قواعد طراحی: اصول ششگانه پایه، الگوهای GRASP، طراحی بر اساس قرارداد (۱ جلسه)

(b) الگوهای طراحی GoF

(i) الگوهای آفرینشی: Singleton, Prototype, Builder, Abstract Factory, Factory Method (۱ جلسه)

(ii) الگوهای ساختاری: Proxy, Facade, Decorator, Composite, Bridge, Adapter (۱ جلسه)

iii) الگوهای رفتاری: Visitor, Strategy, State, Observer, Memento, Mediator, Iterator, Chain of Responsibility (۲ جلسه)

ارزیابی

- آزمون: آزمون‌های میان‌ترم و پایان‌ترم (۶۰٪ کل نمره)
- تمرین و پروژه: تمرینات در قالب یک پروژه درسی تحلیل و طراحی، تعریف شده و بتدریج در طول نیمسال انجام و تحویل داده می‌شوند (۴۰٪ کل نمره).

مراجع اصلی

- J. Arlow and I. Neustadt, *UML 2 and the Unified Process*, 2nd ed. Addison-Wesley, 2005.
- H. Goma, *Software Modeling and Design: UML, Use Cases, Patterns, and Software Architectures*. Cambridge University Press, 2011.
- G. Booch, R. A. Maksimchuk, M. W. Engel, B. J. Young, J. Conallen, and K. A. Houston, *Object-Oriented Analysis and Design with Applications*, 3rd ed. Addison-Wesley, 2007.
- E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, 1995.
- C. Larman, *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*, 3rd ed. Prentice-Hall, 2004.