cis20.2
design and implementation of software applications 2
spring 2010
lecture # I.1
introduction

today's topics:

- course information
- course overview

instructor:

- Prof Elizabeth Sklar, *sklar@sci.brooklyn.cuny.edu*

course web page:

- `http://www.sci.brooklyn.cuny.edu/~sklar/cis20.2`

course information

- about this course
  - intended to give you hands-on experience designing and building a database-backed web application
- the topics are organized into 4 units:

(I) **Software Development**
foundations of software engineering, design patterns, process models, software testing

(II) **Database Systems**
information models, mysql, php

(III) **Tools and Technologies**
interoperability, xml/json, uml, configuration management, content management

(IV) **Intelligent Systems**
intelligent interface agents, decision making, knowledge representation

course structure

- each of the four units has:
  - **lecture(s)**
  - **lab(s)**
  - **assignment(s)** or **project**
- the labs will be hands-on sessions using the internet in a multimedia classroom (room 5301 N)
- your grade is comprised of:
  - assignments and projects (60% total)
  - midterm (10%) on THU MARCH 18
  - final (30%) (date TBA)

rules and regulations

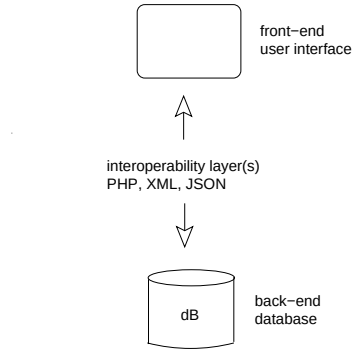
- have a look at the course web page:
  `http://www.sci.brooklyn.cuny.edu/~sklar/cis20.2`
- there you will find links to pages that describe my policies
- in particular look at:
  - **late assignments** policy
  - college policy on **academic integrity**

online resources

- there is no textbook for this class
- lecture notes will be posted on the course web page after there is a lecture
- labs will distributed in class and be posted after the class
- links to online resources will also be posted on the class web page
- for people who want books, much of what we discuss can be found in books that are published by O'Reilly Publishing (`http://www.oreilly.com`), and appropriate publications will be linked on the class web page

course overview

- throughout the course, we will be discussing the concepts, technologies and design principles that underly web-based systems with the following type of structure:



- everything we do will be based on software technologies that are freely available
- we will work in a Linux environment, using "LAMP" technologies
  LAMP = Linux, Apache, MySQL, PHP
  where,
  - Linux is a unix-based operating system
  - Apache is a web (HTTP) server
  - MySQL is a database engine
  - PHP is a scripting language that supports (among other things) interoperability between a database engine (like mySQL) and any web browser

case studies

- with each unit, we will look at some case studies
- web sites that focus on delivery of goods or products (bringing stuff to people)
  - `amazon.com` — a "click and mortar" company that didn't exist before the internet. It delivers products and also provides a means for re-sellers (other companies) to sell products through amazon's gateway.
  - `macys.com` — a traditional "bricks and mortar" company that existed long before there was internet; but now they have a web site and support online purchases in addition to physical stores.
  - `llbean.com` — a catalogue-based company that sells clothing and accessories. Now they have a web site and also sell online, in addition to sending out physical catalogues and supporting sales over the phone. Originally, they did not have any physical stores (except a factory outlet in Maine, where the company is located), but now they have stores in shopping malls.
- web sites that focus on delivery of services (bringing information to people)
  - `yahoo.com` — delivers a wide range of services, from email to games to a search engine to news feeds

- `google.com` — primarily a search engine, but also now supports email, hosting and a very popular mapping application
- `jstor.org` — a not-for-profit services that archives and provides access to a broad range of academic journals

• web sites that support social networking (bringing people together)

- `facebook.com` — hosts profiles, live space for chat, space for asynchronous collaborative blogging
- `linkedin.com` — professional matching
- `match.com` — dating site

• web sites that contain intelligent components

- `hopstop.com` — provides personalized directions to places in NYC via public transportation
- `netflix.com` — provides preference-based matching of customers to potential goods and services
- *multiplayer games* — provide intelligent "agent" players that can be opponents or team mates. you will help decide which online game to use as the example in class.

our running example

• bigNightOut.com

Throughout the course, we will be introducing a number of tools and technologies. It will be helpful to have a running example, as kind of glue that holds everything together. The example we will use is this:
Suppose we want to develop a web site called **bigNightOut.com**.
The idea is to have a place where users can go and plan an evening out on the town. There will be information about where to go for dinner and a movie.

Our site will have components like:

- restaurant
  we might want to know:
  * restaurant review
  * location and directions
  * category of food
  * real-time information like how busy is it right now
- movie
  we might want to know:

  * what is playing where and when
  * the price of a ticket
  * the length of the movie
  * access to view a trailer
  * reviews
  * real-time information like are there any tickets left for the next showing
- companions
  we might want to find someone to go out with:
  * do any friends want to come along?
  * is there anyone we do not want to invite?
  * maybe you want a blind date
  * real-time information like where are your friends right now

We will discuss the technologies that could be used to create such a site. We will design the site and review each other's designs. We will build the site and test it!

to do

• complete the pre-semester survey and return it to me

• check out the class web page:
`http://www.sci.brooklyn.cuny.edu/~sklar/cis20.2`