Material and some slide content from:
- Emerson Murphy-Hill
- Software Architecture: Foundations, Theory, and Practice
- Essential Software Architecture

Basics of SW Architecture

Reid Holmes

Lecture 4 - Thursday, September 22 2011.

Proposal Presentations

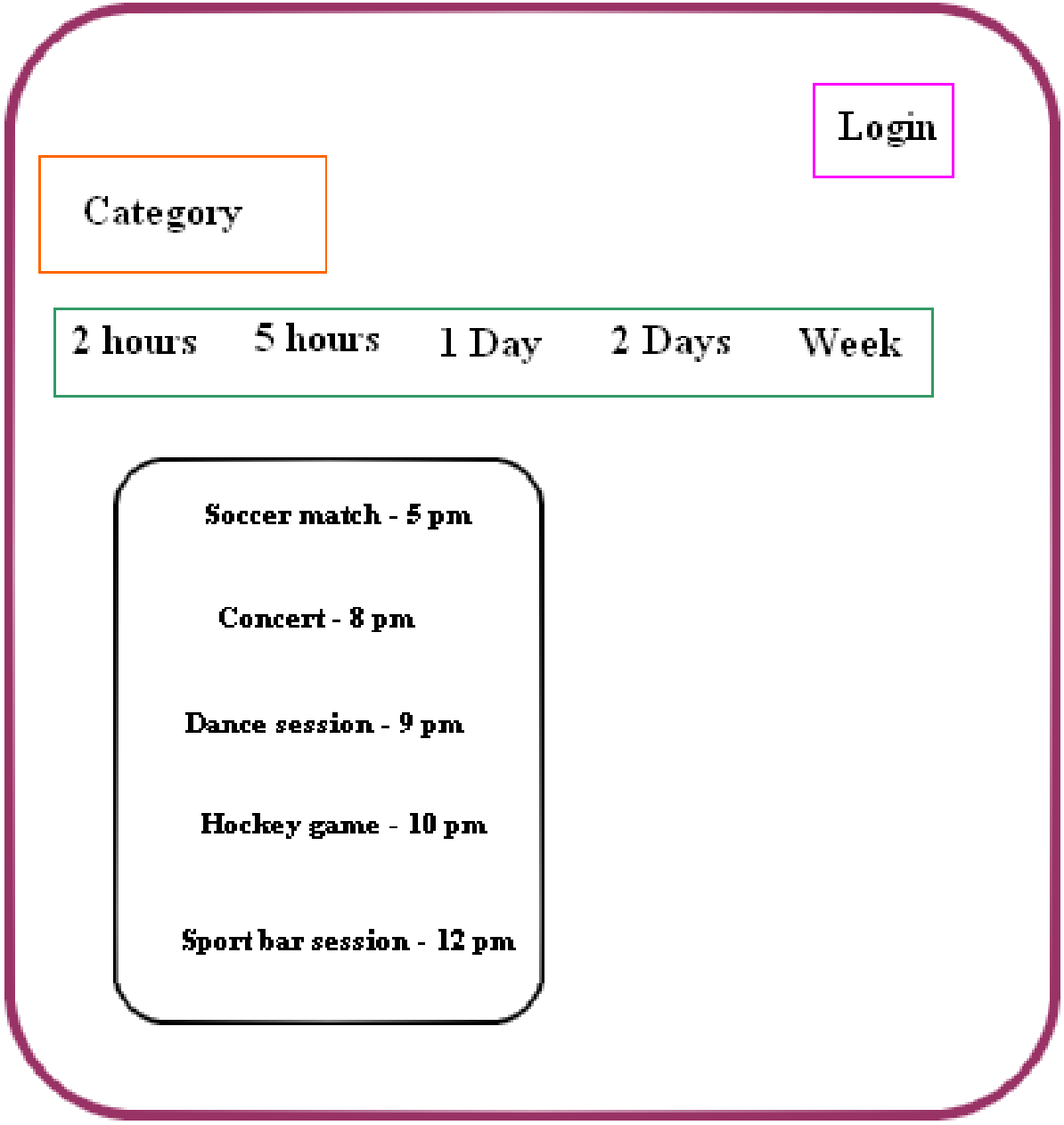
- Finish the last presentation.

Waterloo Event Finder

Mike Ye

Martin Lacombe

Main functionalities - Student



Main functionalities - Organizers

Name :

Description:

Date :

Time :

Location:

Aditional information :

So what?

- What makes building systems so hard?
  - Young field.
  - High user expectations.
  - Software cannot execute independently.
- Incidental difficulties [Brooks MMM].
  - Problems that can be overcome. (e.g., ...)
- Essential difficulties [Brooks MMM].
  - Those problems that cannot be easily overcome.

[TAILOR ET AL.]

Essential Difficulties

- Abstraction alone cannot help.
  - Complexity
    - Grows non-linearly with program size.
  - Conformity
    - System is dependent on its environment.
  - Changeability
    - Perception that software is easily modified.
  - Intangibility
    - Not constrained by physical laws.

[TAILOR ET AL.]

Attacks on Complexity

- High-level languages.
- Development tools & environments.
- Component-based reuse.
- Development strategies.
  - Incremental, evolutionary, spiral models.
- Emphasis on design.
  - Design-centric approach taken from outset.

[TAILOR ET AL.]

Architecture Analogies

- We live in them.
- We know (approximately) how they are built.
  - Requirements.
  - Blueprints (design).
  - Construction (implementation).
  - Use in practice.

[TAILOR ET AL.]

The architect

- Distinctive role.
- Broadly trained.
  - Requirements, design, implementation, & use.
- Has a keen sense of aesthetics.
- Strong understanding of the domain.
  - What are these for buildings?
  - What are these for software?

[TAILOR ET AL.]

The architect

How is building architecture different from software architecture?

What common benefits can software gain from an architect that a building gets from its architect?

What is SW architecture?

- Definition:

> "The set of principal design decisions about the system"

- Blueprint for construction and evolution.
- Encompasses:
  - Structure
  - Behaviour
  - Interaction
  - Non-functional properties

Prescriptive vs descriptive

- Prescriptive architecture dictates how the system will be built *a priori*.
  - (as-conceived)
- Descriptive architecture captures how the system was actually built after the fact.
  - (as-implemented)

Architectural degradation

- Drift
  - Introduction of changes that are not captured in the current architecture but do not violate it.
- Erosion
  - Introduction of changes that violate the current architecture.
- How can this happen?

Architectural recovery

- [ICSE 1999: Bowman, Holt, and Brewster]
- Conceptual architecture
  - How developers think about the system.
  - Focuses on meaningful relationships.
- Concrete architecture
  - How the system was actually built.
  - Necessary: the devil is in the details.

Components

- Elements that encapsulate processing and data at an architectural level.
- Definition:
  - Architectural entity that:
    - encapsulates a subset of functionality.
    - restricts access via explicit interface.
    - has explicit environmental dependencies.

Connectors

- Definition:
  - An architectural entity tasked with effecting and regulating interactions between components.
- Connectors are often more challenging than components in large heterogenous systems.
- Often consists of method calls, but be much more.
  - Examples?
- Often provide application-independent interaction mechanisms.

Configurations

- Bind components and connectors together in a specific way.
- Definition:
  - An architectural configuration, or topology, is a set of specific associations between the components and the connectors of the system's architecture.
- Differentiates a bag of components and connectors from an implementable system.

Architectural styles

- Some design choices are better than others.
  - Experience can guide us towards beneficial sets of choices (patterns) that have positive properties.
    - Such as?
- An architectural style is a named collection of architectural design decisions that:
  - Are applicable to a given context.
  - Constrain design decisions.
  - Elicit beneficial qualities in resulting systems.