

IOWA STATE UNIVERSITY



Teaching Software Architecture Process to Undergraduate Students: A Case Study

Lotfi ben Othmane
Iowa State University

2nd International Workshop on Engineering IoT Systems:
Architectures, Services, Applications, and Platforms
Seattle, April 30, 2018

Challenges of Teaching Software Architecture

Characteristics of software architecture:

- Requires strong experience
- Fuzzy - deals with unclear problems and solutions
- Applies to complex software

Characteristics of the learners (Undergrad)

- Have limited knowledge and experience
- Used to concrete concepts, precise problems and solutions
- Use simple applications

Classic Program - A Set of Topics

1. Overview of software architecture: what? And why?
2. Software architecture structure and styles
3. Software quality attributes
4. Overview of software engineering principles
5. Introduction to design by contract
6. Introduction to UML
7. Software reengineering
8. Software evolution
9. Design patterns
10. Component based software engineering
11. Model-driven languages
12. Architecture description languages

The Need

The students want to learn:

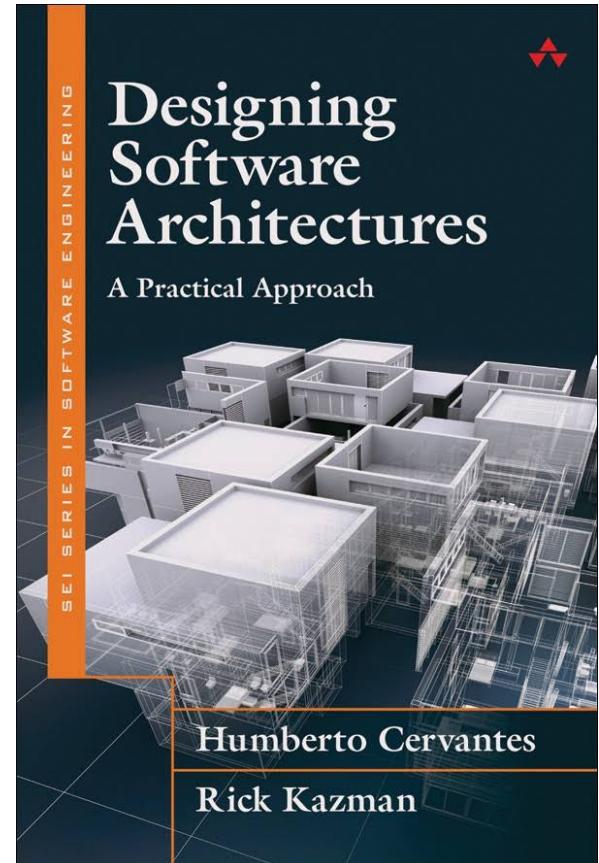
- How to design an architecture?
- How to evaluate an architecture?

Approach

1. Focus on the main topics: requirements elicitation, design, and evaluation
2. Use a software architecture design process
3. Use Smart Home (IoT) project for experimentation

Course Redesign - Structure

- Use the Attribute-Driven Design process
- Use the three case studies as examples
- Complement the book with other resources such as big up front design vs. agile
- The project is to extend a smart home application (developed by students)



Course Redesign - Topics

1. Overview of software architecture
2. Unified Modeling Language (UML)
3. Architecture drivers
4. Architecture styles, patterns, and tactics
5. Architecture design process
6. Documenting a software architecture
7. Architecture evaluation (ATAM)
8. Software security architecture
9. Architecture recovery

Study Design - Goals

1. Assess the capabilities of the students to synthesis acquired knowledge
2. Assess the cognitive levels of the students using Bloom taxonomy
3. Assess the self-confidence of the students in designing software architecture
4. Assess the student perceptions of the effectiveness of the course

Study Design - Preparation

- Designed a questionnaire with open-ended questions
- The students took the questionnaire in Nov. 2018
- 51 students out of 60 participated – submission is anonymous
- Coding – we associated used verbs with Bloom levels of cognitive levels

Results - Student Expectations

<u>Expectation</u>	<u># students</u>
No expectations	18 (35%)
Design of architecture	11 (22%)
Curious about the topic	6 (12%)
Relation to another course	6 (12%)
Heavy coding	6 (12%)
Types of architecture	2 (4%)
Architectures styles	2 (4%)

.....

Results - Learning Efficiency

Question: Assume you are given a project and asked to design an architecture for it. How would you do the design?

<u>Cognitive level</u>	<u>Percentage</u>
Creating	4%
Evaluating	11%
Analyzing	4%
Applying	72%
Understanding	9%
Irrelevant	5 out of 51

E.g: If you are referring to the process taught in class,
I am not going to use it.

Results – Differentiate Web-based vs IoT-based Projects

<u>Aspect</u>	<u># of students</u>
Architecture drivers	30
Patterns related to the projects	23
Architectural knowledge	17
Simplicity	5
Technology stack	3
Configuration management	1

Smart Home Projects



Examples of Architecture Drivers for IoT

Question: Would you use your system at home?

- Accuracy for facial recognition setup
- Performance for using face recognition-based authentication to unlock doors
- Performance for playing music
- Reliability for using temperature to open windows
- Reliability for smart alarm
- Reliability for smart watering
- Accuracy of motion sensors

Results – Preferred Learning Methods

<u>Method</u>	<u>#of students</u>
Group assignments	12
Individual assignments	11
Case studies	11
Reading	8
In-class group activities	7
No definitive answer	5
Quizzes	4
Drawing diagrams	4
None	3
Learn on their own	2

Results - Confidence

<u>Level Codes</u>	<u># of students</u>
High confidence	22%
Moderate confidence	25%
Fair confidence	29%
Not confident	10%
No definite answer	14%

Effectiveness of the Course

<u>Level Codes</u>	<u># of students</u>
Effective and above	10
Moderate	18
Not effective	12
Unknown	11

Conclusion

- Using a ADD process can help the students to reason about software architecture
- IoT-based projects could help the students to understand the value of quality attributes and architecture
- The students prefer learning software architecture using case studies and assignments
- There is a challenge to improve the confidence of the students in designing software architecture

Thank you