

# DEPAUL UNIVERSITY



## Course Overview

**Object-oriented Software Development**  
**SE 350– Spring 2021**

Vahid Alizadeh



Week 1.1  
March 30, 2021



# Introduction

## Who am I? Who are you?





COULD WE IMPROVE PROGRAMMERS'  
**PRODUCTIVITY** VIA DESIGNING AND VALIDATING  
USABLE **ARTIFICIAL ASSISTANTS** WHICH CAN  
BEHAVE LIKE A REAL MEMBER OF A DEVELOPMENT  
TEAM?





# The Importance of Software Quality

“Software never was perfect and won’t get perfect. But is that a license to create garbage? The missing ingredient is our reluctance to quantify quality.”  
– Boris Beizer

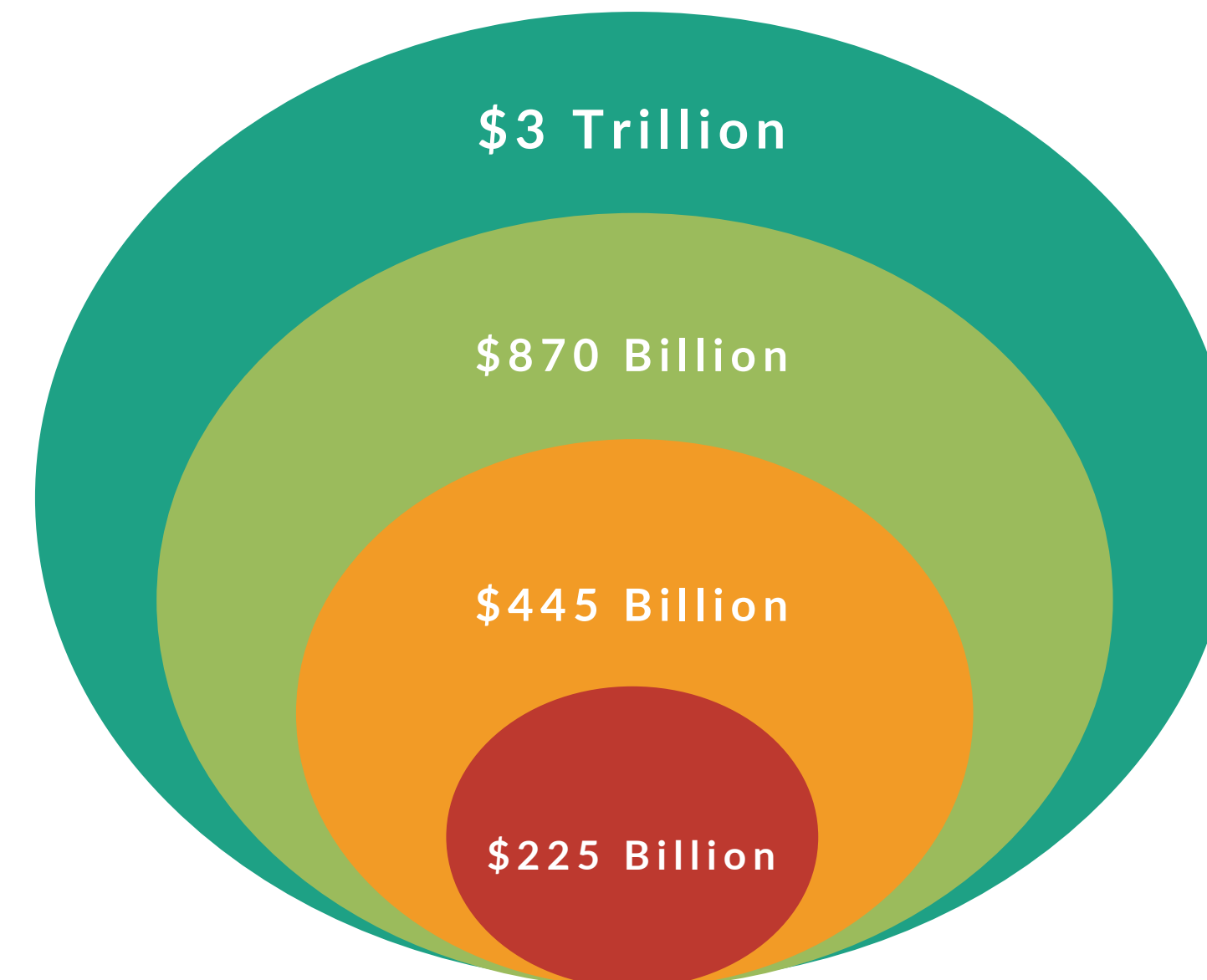
Software quality can have a serious effect on the bottom-line.

malfunctions caused by defective software

Transnational organized crime

Cybercrime

Air Pollution Deaths



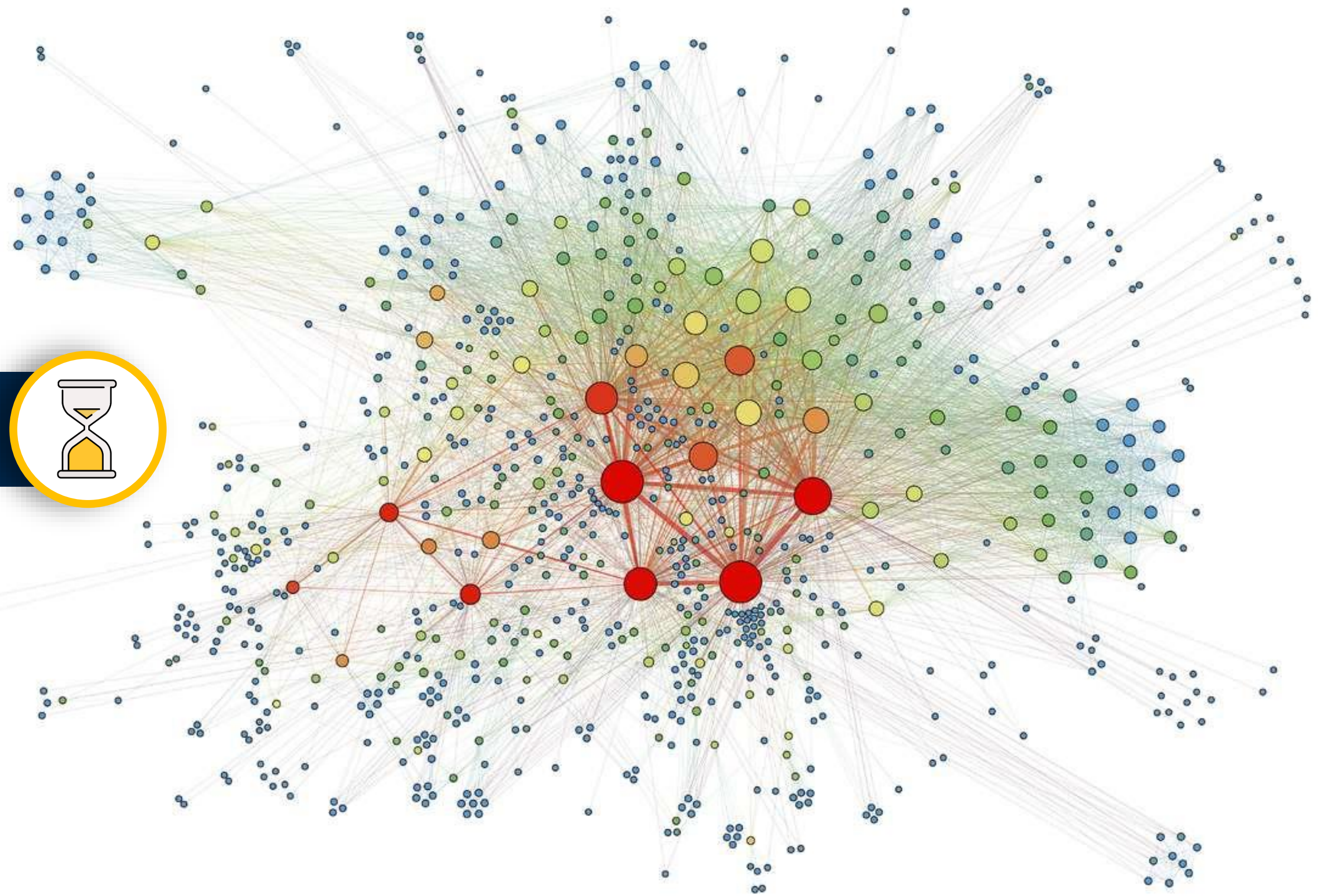
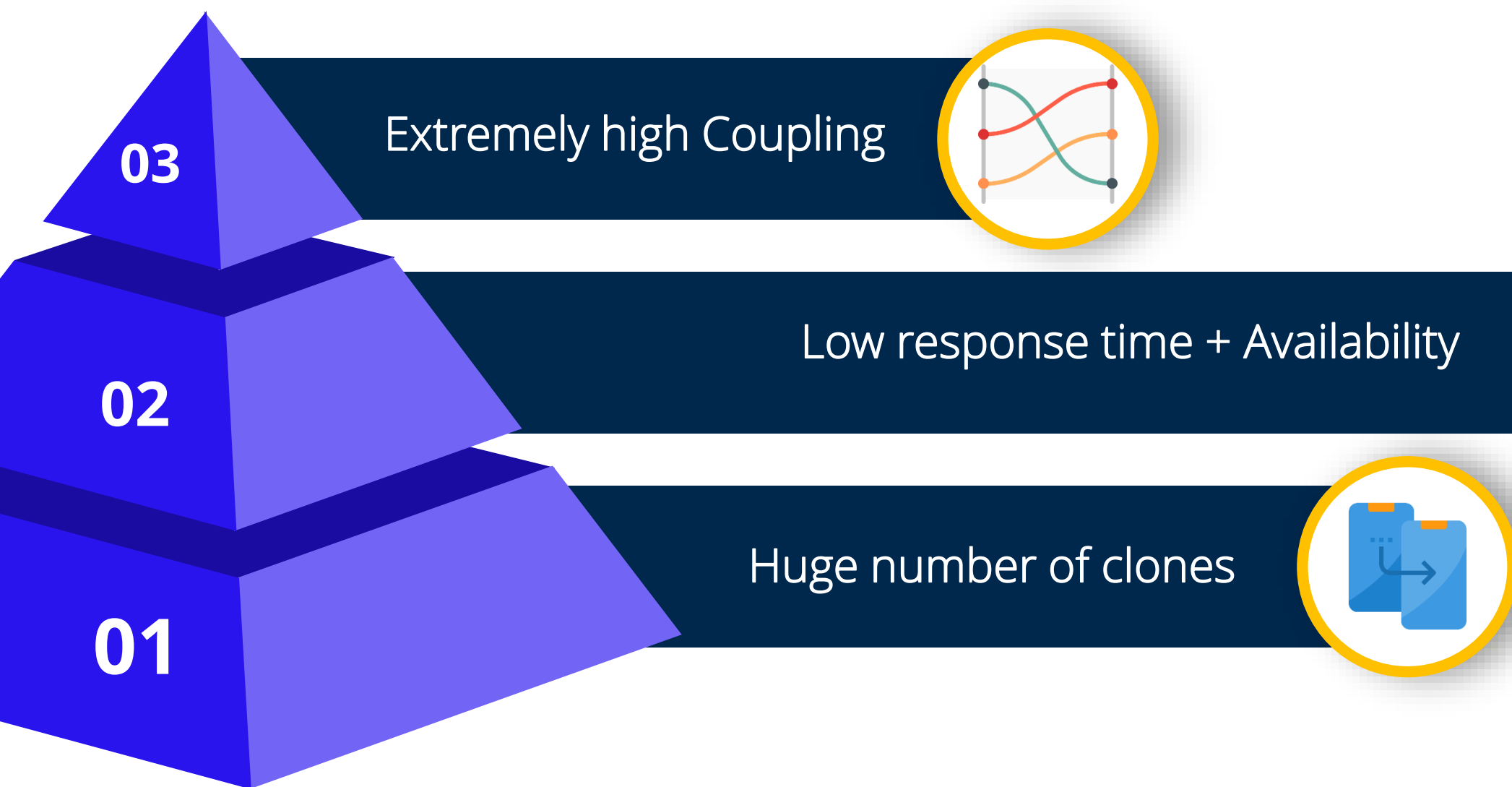
Global Economy  
Costs per Year

# Real-World Example





# Services Architecture





# Context

Software projects evolve and change continuously



## Increases the complexity of project

- 60% of developers' time in understanding the existing software



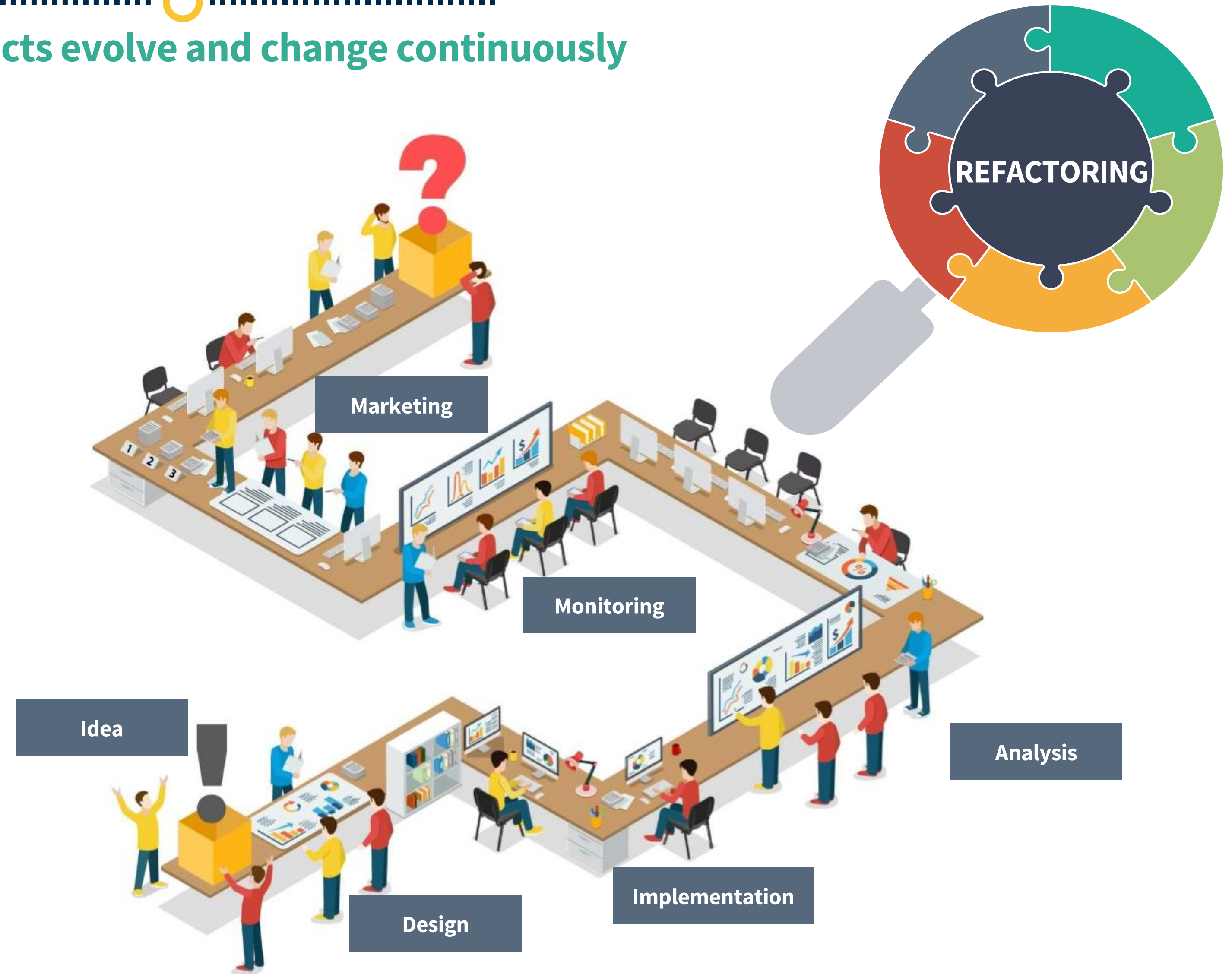
## Reduces the productivity

- 76% of Software Engineers working on maintenance tasks

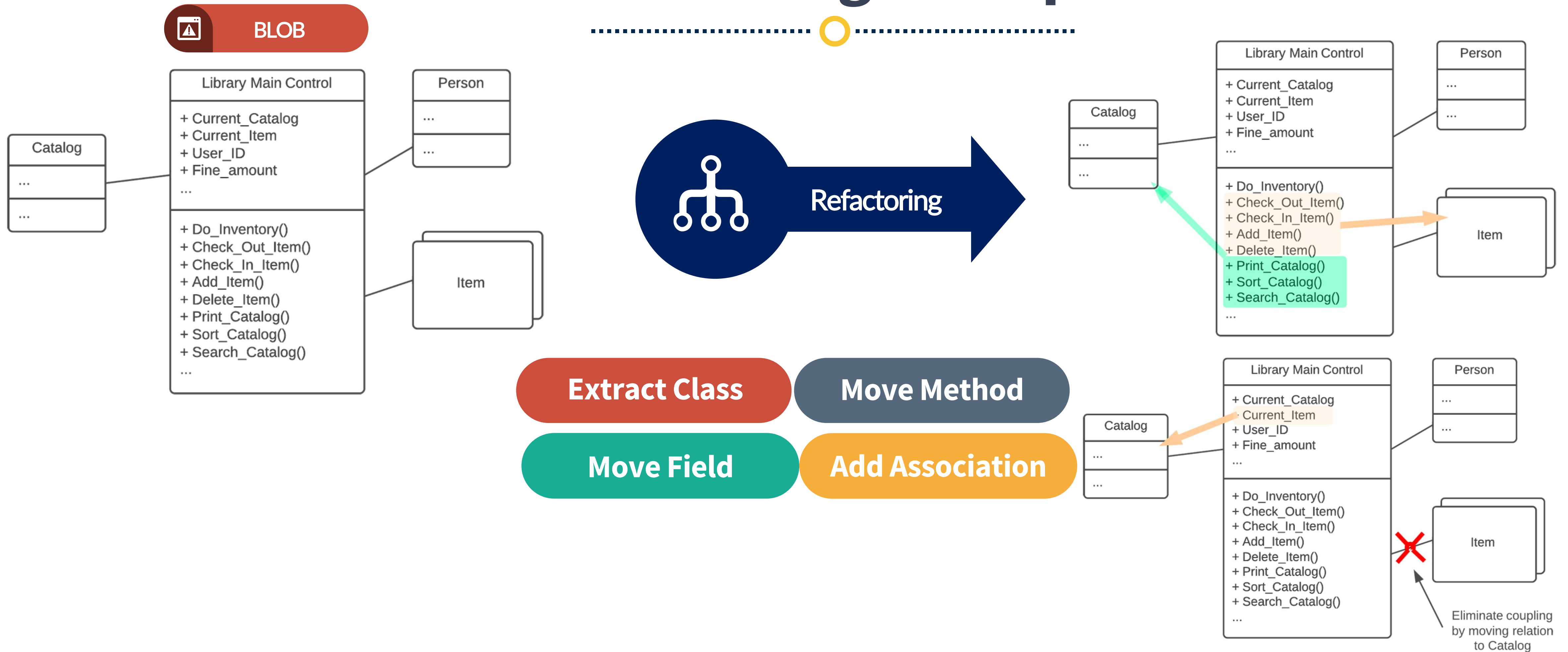


## Increases cost of maintenance

- 67% of Software Budget allocated to maintenance



# Refactoring Example



“A change made to the *internal structure* of software to make it *easier to understand* and *cheaper to modify* without changing its observable behavior”

– M. Fowler [1999]



# Which one is easier to work with?



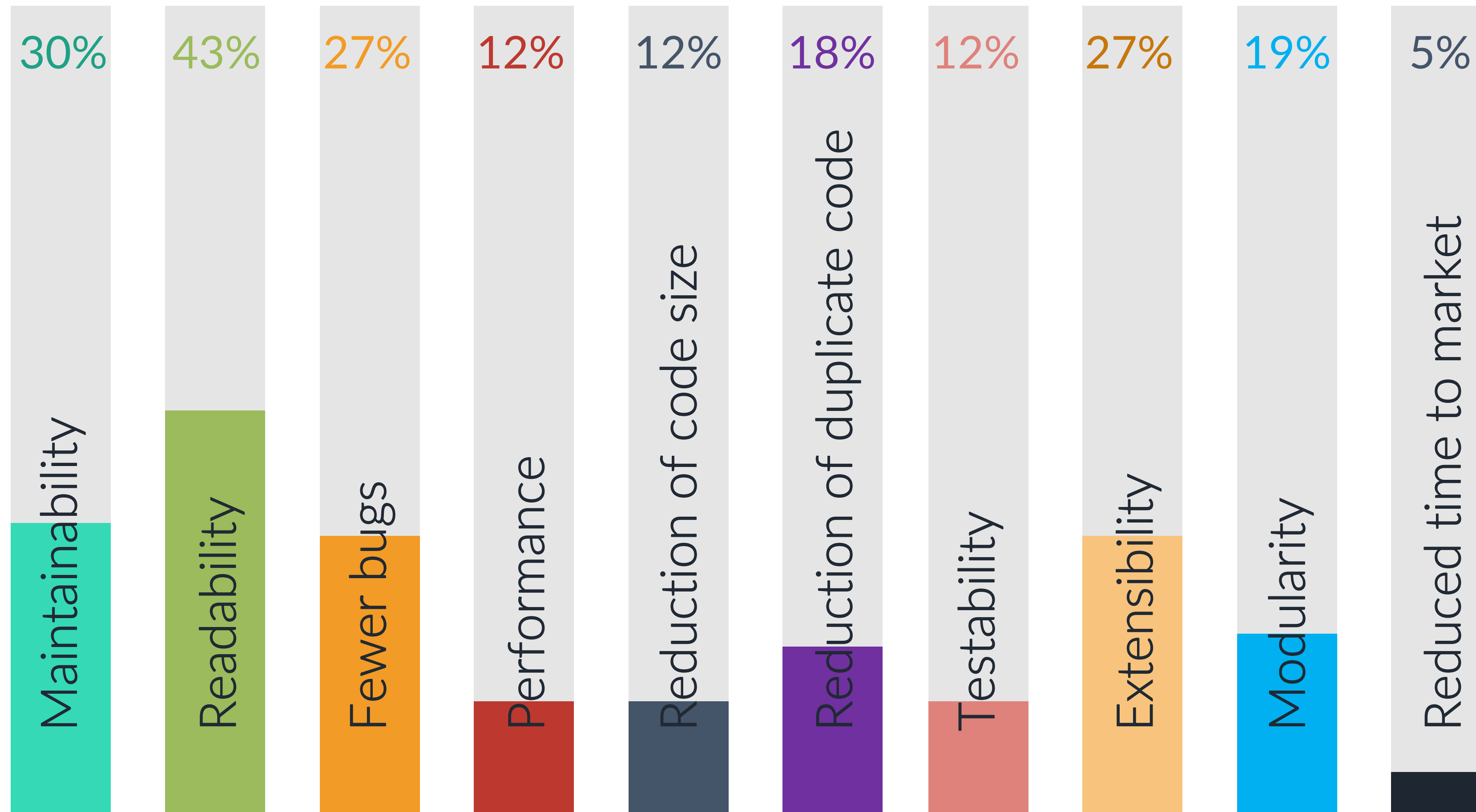


# Refactoring Effect



## An Empirical Study of Refactoring Challenges and Benefits at Microsoft

Miryung et al. "An empirical study of refactoring challenges and benefits at Microsoft." IEEE Transactions on Software Engineering (2014)



Object-oriented Software Development – SE 350 – Spring 2021



# But Refactoring is...



Time-consuming



Boring



Error-prone

## What do experienced developers need:

*"I don't feel a great need for automated refactoring tools, but I would like **code understanding** and **visualization tools** to help me make sure that my manual refactorings are valid."*

*"I'd love a tool that could **estimate the benefits of refactoring**. Also, it'd be awesome to have better tools to help figure out who knows a lot about the existing code"*

# Refactoring Problems



WHAT to Improve in Software Health?



WHEN to Improve Software Health?



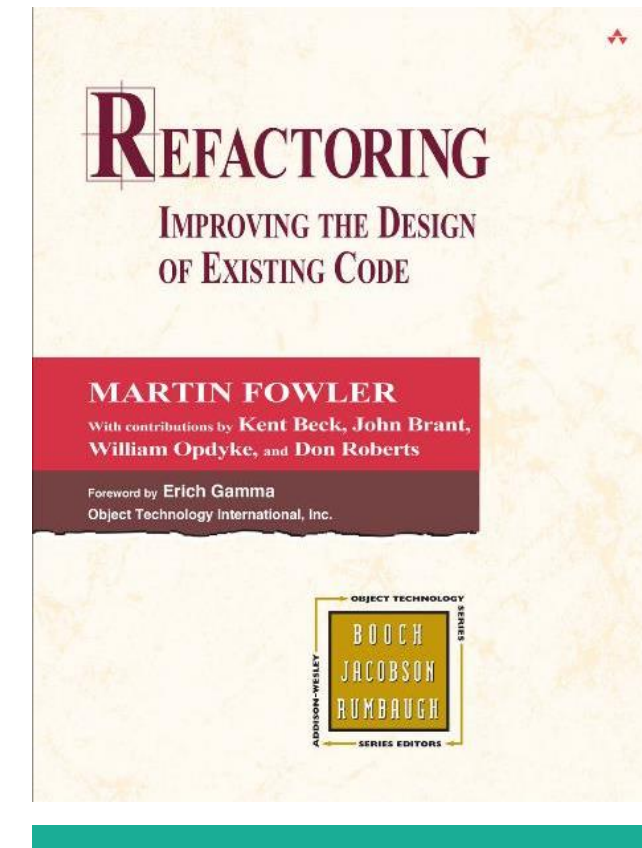
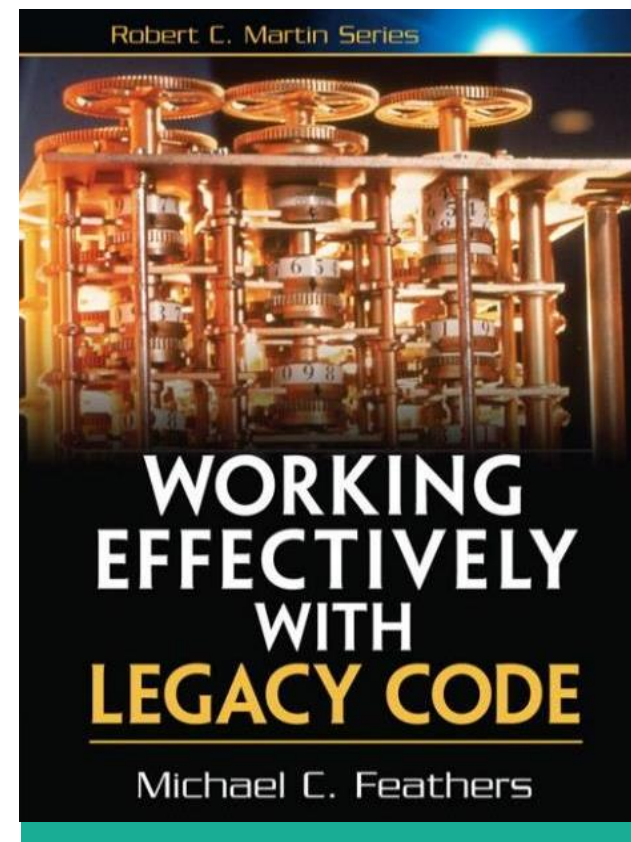
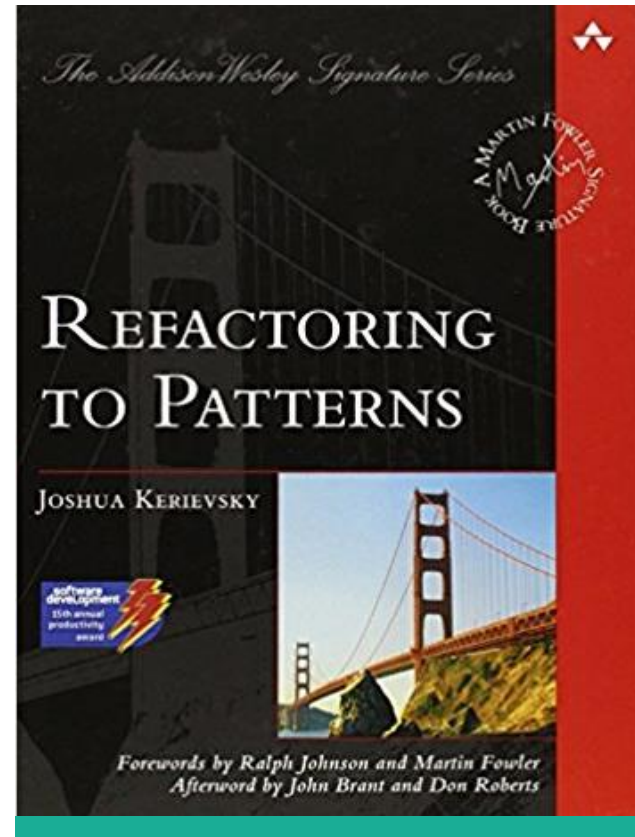
HOW to Improve Software Health?



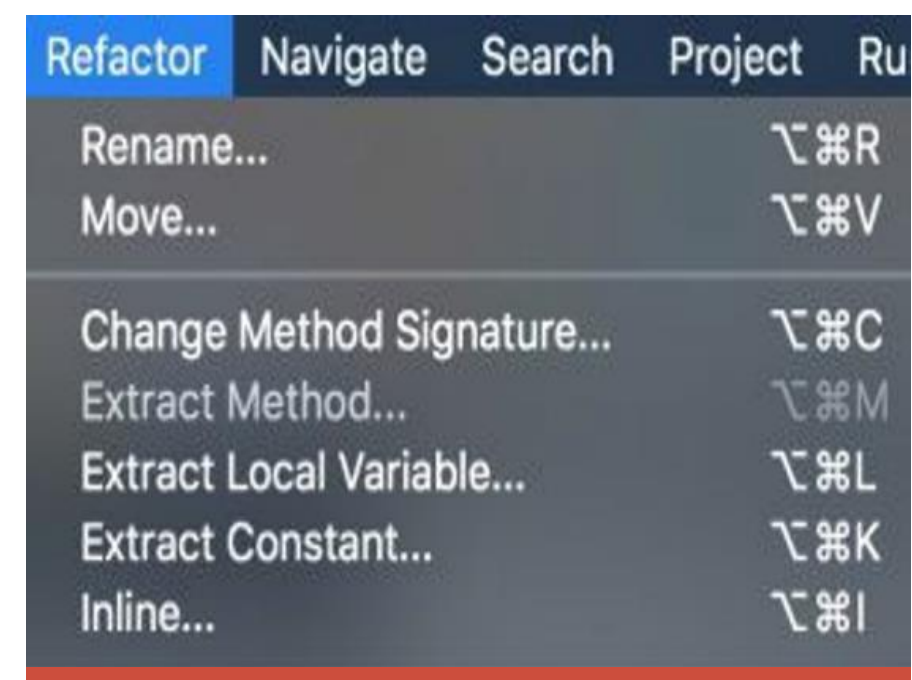


# Current State of Refactoring

## Approaches / Challenges



1999

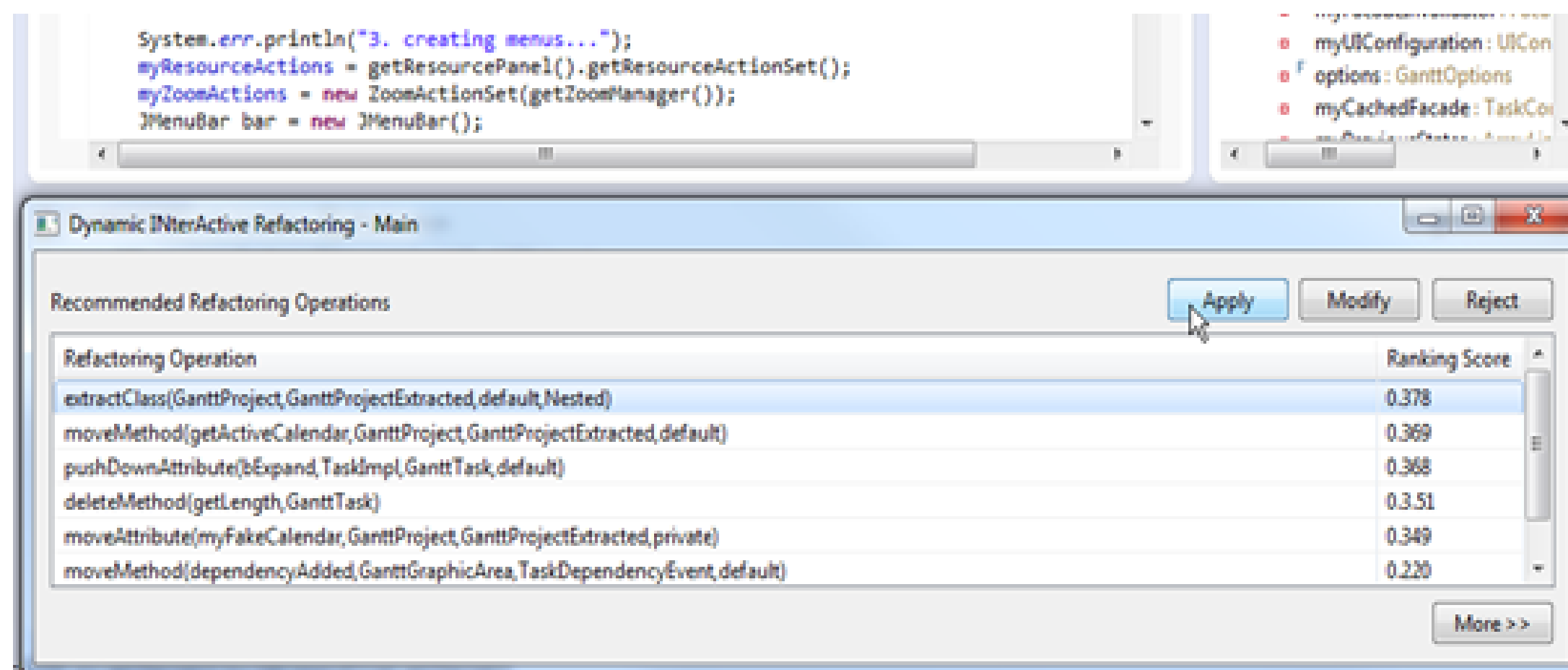


### Manual

- error-prone
- time consuming
- not scalable
- not useful for radical refactoring

## Fully-automated

- lacks flexibility
- fails to consider developer perspective
- proposes a long static list of Refactorings
- not always lead to the desired architecture



## Semi-automated/Interactive

- Expensive process
- Evaluation of many refactorings
- Have to consider many quality attributes



Refactoring Tools/Techniques are not aligned with  
*Continuous Integration*



# Your turn!

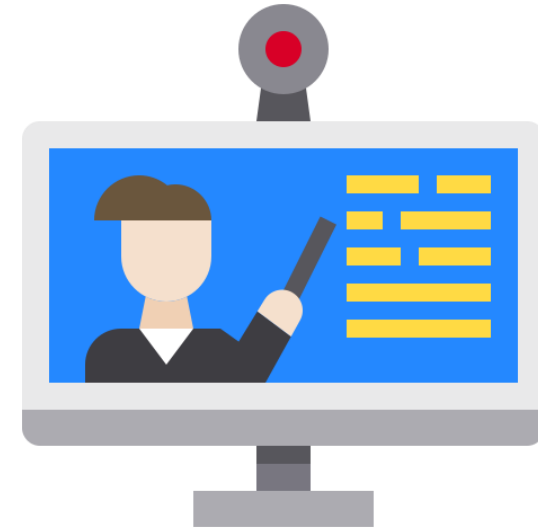


## ■ Introduce yourself

- Your background / experiences
- Future plan (Dream job)
- Your interests
- Why did you take this course?
- What is your course load this semester?
- How do you feel about Online course modality?
- Any fun fact about yourself?

## ■ Any questions about this course?





# Object-oriented Software Development

## Course Overview



# Class Schedule

## ▪ Object-oriented Software Development – SE 350

- Spring 2021
- Online Zoom Sessions
  - **Required** for **Sync** (601-602) sections
  - **Optional** for **Async** (610-611) sections
  - Recordings will be available for all sections in D2L
  - You can find the schedule and links in D2L
- **Section 601-610:**
  - **Tuesdays - Thursdays** 1:30 PM – 3:00 PM
- **Section 602-611:**
  - **Tuesdays - Thursdays** 3:10 PM – 4:40 PM



# My Contact



## ▪ Email

- [V.alizadeh@depaul.edu](mailto:V.alizadeh@depaul.edu)

## ▪ Microsoft Team

- Private message
- Or mention me in Discussion channel

## ▪ Office

- #829 CDM
- Gated community (offices behind locked doors)
- Need to call for access (phone next to reception table)
- Office Phone
  - 312-362-6248





# Office Hours



## ■ Zoom Office Hours

- Wednesdays
  - 2:00 – 5:00 PM
  - Join URL:
    - <https://depaul.zoom.us/j/7240757601?pwd=Tjl3djJ0aWY3R244R09uZ1VMWG1Fdz09>
    - There is waiting room.
- Better to schedule an appointment
  - You can contact me (email or Teams) to schedule a time.
  - Or use BlueStar
  - We can have individual or group meeting upon your request.



# Course Forum: Microsoft Teams



Microsoft Teams

- **Everyone is required** to join the Microsoft Teams group

- **Team Code:**



- **Where do I enter the code?**

- Click Join or create a team below your teams list and look for the Join a team with a code card

- **We use MS Teams as a Discussion forum, Sharing files, Announcements, Scheduling the meetings, Communications, ....**

- **Some materials and activities will be replicated in D2L as well.**





# What is this course about?



- This course studies **Principle, techniques and tools** of object-oriented **modeling, design**, and **implementation** of large-scale software systems. The students will learn the real-world aspects of object orientation by putting the concepts into practice.
- **Topics include:**
  - Principles of object-oriented programming languages
  - Principles of object-oriented design
  - UML diagrams (class, object, sequence,...)
  - Quality metrics and Refactoring of object-oriented design
  - Anti-patterns
  - Testing methodologies
  - Design Patterns



# What is this course about?



## ▪ Not just programming

- Designing programming solutions
- What does it mean?

## ▪ Common approaches

- Patterns

## ▪ School vs. Real World

## ▪ Writing maintainable, extensible, high quality code





# Course Objectives & Learning Outcomes



## ▪ Object-Oriented Design Principles

- You will have a deeper understanding of object-oriented concepts and how to use them, and will be able to design and develop software applications using object-oriented design principles.

## ▪ Visual Modeling

- You will be able to model a software solution visually using UML sequence and class diagrams.

## ▪ Design Patterns

- You will be able to design and implement an executable solution to a given problem in a programming language using the most suitable set of common software and architectural design patterns.

## ▪ Object-Oriented Programming Paradigm Principles

- You will be able to effectively translate design patterns and object-oriented design principles into an object-oriented programming language.

## ▪ Implementation

- You will be able to apply advanced object-oriented programming language features/capabilities to design and implement a given software application using object-oriented best practices. You will be proficient with incremental/iterative development, refactoring, and writing higher-quality codes.



# Prerequisite



## ■ What do you need to have for this class?

- Understanding how to create and manage a software project.
- **Java** and **UML** will be used for source code examples, assignments, and the exams.
- Some experience programming in **Java** or another C-like language is required. This is **NOT** an introductory Java course.

## ■ Courses:

- **Required:**
- **CSC 301** | DATA STRUCTURES II
  - (OR equivalent courses on data structures)
- **Optional and Useful:**
- **SE 333** | SOFTWARE TESTING
- **SE 330** | OBJECT ORIENTED MODELING





# Grading



## ■ Assignments (50%)

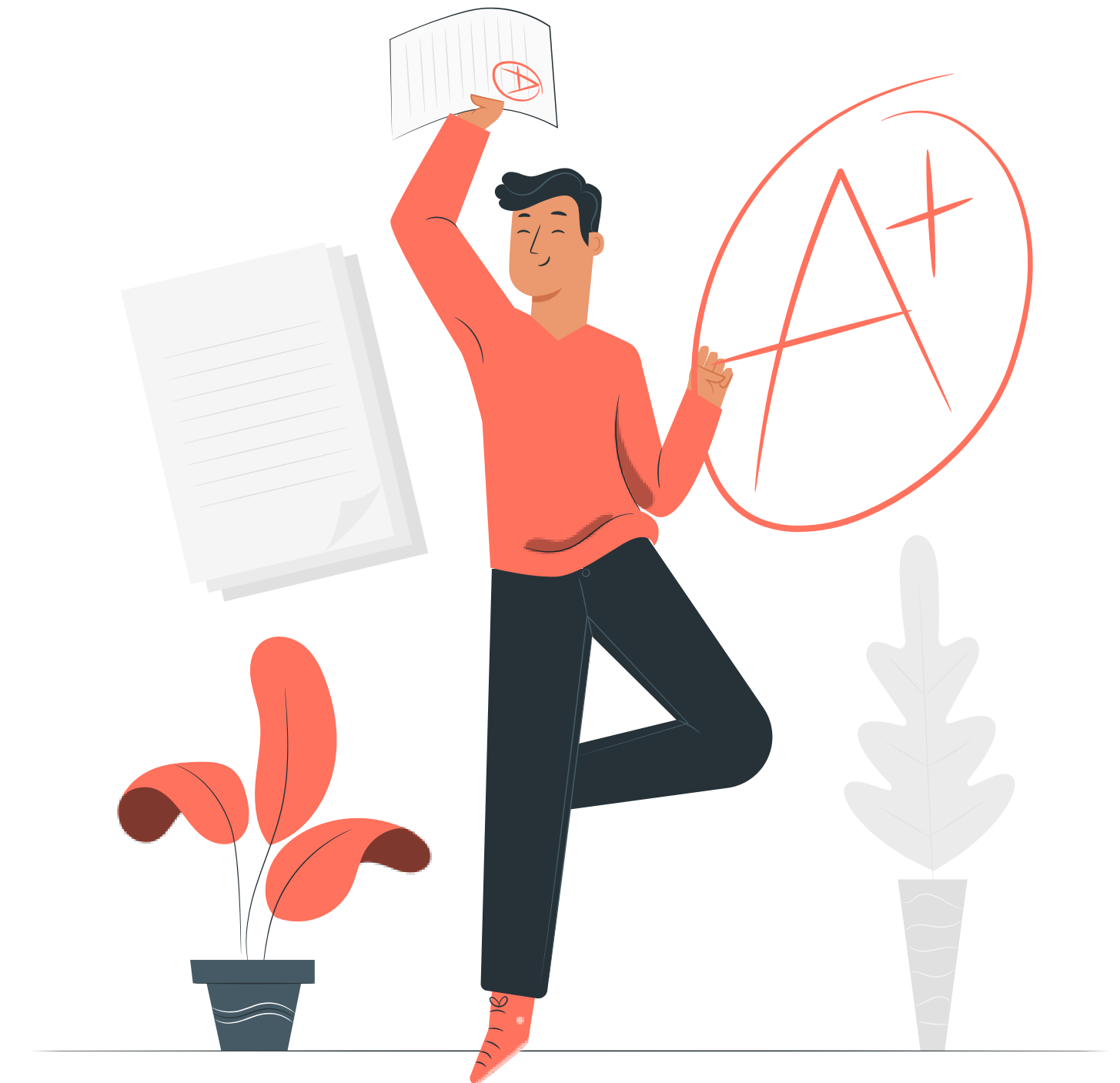
- **5%** - In-class and Forum Participation
- **5%** - Quizzes
- **40%** - Home works and Programming Projects
  - Must be submitted via D2L. No email or cloud sharing submissions is allowed.

## ■ Exams (50%)

- **20%** - Mid-term Exam
- **30%** - Final Exam
- **Optional Bonus (10%)**
  - **10%** - Research OR Implementation Project

## ■ Grading Scale

A 93-100	B- 80-82	D+ 65-68
A- 90-92	C+ 77-79	D 60-64
B+ 87-89	C 73-76	F <60
B 83-86	C- 69-72	



# Tentative Schedule

Date	Activity	Lecture
<b>Week 1</b> Mar30 & Apr1	<b>Environment Setup</b>	<ul style="list-style-type: none"> <li>• Introduction to OO</li> <li>• UML Diagrams</li> <li>• OO Principles                             <ul style="list-style-type: none"> <li>• Abstraction</li> <li>• Encapsulation</li> <li>• Inheritance</li> <li>• Polymorphism</li> <li>• Composition</li> <li>• Interface</li> <li>• Delegation</li> </ul> </li> </ul>
<b>Week 2</b> Apr6 & Apr8		
<b>Week 3</b> Apr13 & Apr15	<b>Assignment 1</b>	<ul style="list-style-type: none"> <li>• Design with Exceptions</li> <li>• Principles and Foundations of OO Design                             <ul style="list-style-type: none"> <li>• SOLID Principles                                     <ul style="list-style-type: none"> <li>• Single Responsibility</li> <li>• Open-Closed</li> <li>• Liskov Substitution</li> <li>• Dependency Inversion</li> <li>• Interface Segregation</li> </ul> </li> </ul> </li> </ul>
<b>Week 4</b> Apr20 & Apr22	<b>Assignment 2</b>	
<b>Week 5</b> Apr27 & Apr29	<b>Mid-term Exam</b>	<ul style="list-style-type: none"> <li>• Design Patterns                             <ul style="list-style-type: none"> <li>• Creational</li> <li>• Structural</li> <li>• Behavioral</li> </ul> </li> <li>• Anti-patterns</li> <li>• Refactoring OO Design</li> <li>• JUnit</li> <li>• OO Design &amp; Development Research Trends</li> </ul>
<b>Week 6</b> May4 & May6		
<b>Week 7</b> May11 & May13	<b>Assignment 3</b>	
<b>Week 8</b> May18 & May20		
<b>Week 9</b> May25 & May27	<b>Assignment 4</b>	
<b>Week 10</b> Jun1 & Jun3	<b>Bonus Projects Due</b>	
<b>Week 11 (Finals)</b> Jun8 & Jun10	<b>Final Exam</b>	



## Important Dates

**Monday April 5, 2021** Last day to add classes to SQ2021 schedule

**Friday April 9, 2021** Last day to drop SQ2021 classes with no penalty

**Monday April 12, 2021** Last day to select pass/fail option for SQ2021 classes

**Friday May 14, 2021** Last day to withdraw from SQ2021 classes



**Note** Please visit *DePaul Academic Calendar-Spring Term* for the complete list.

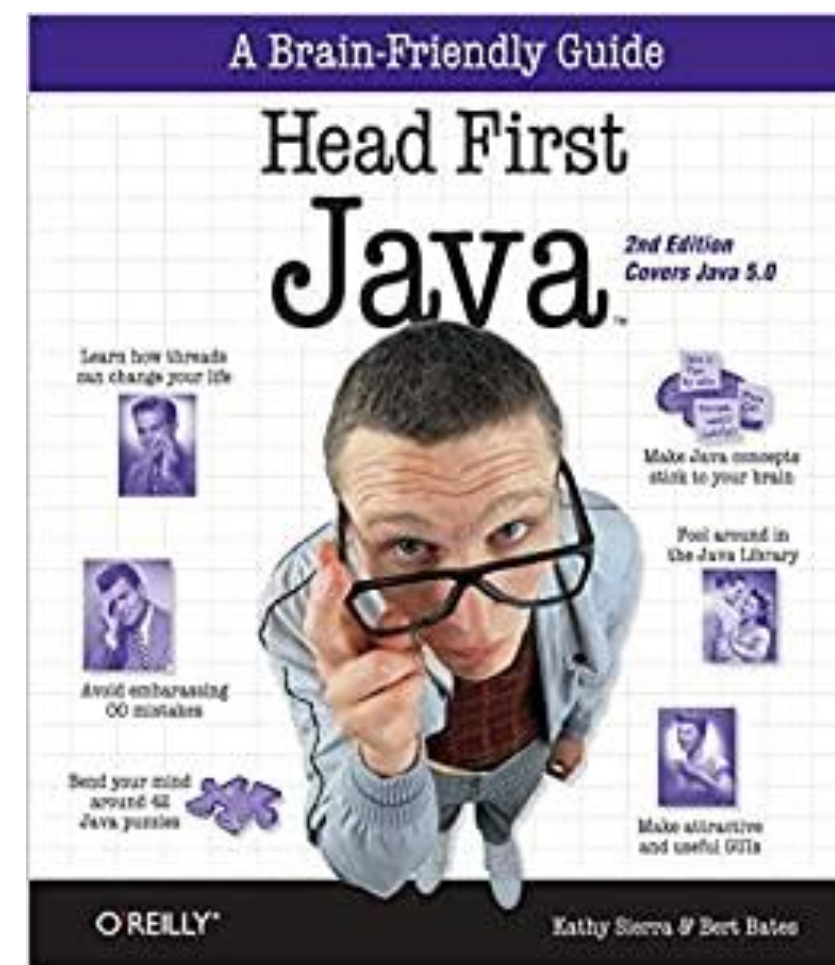
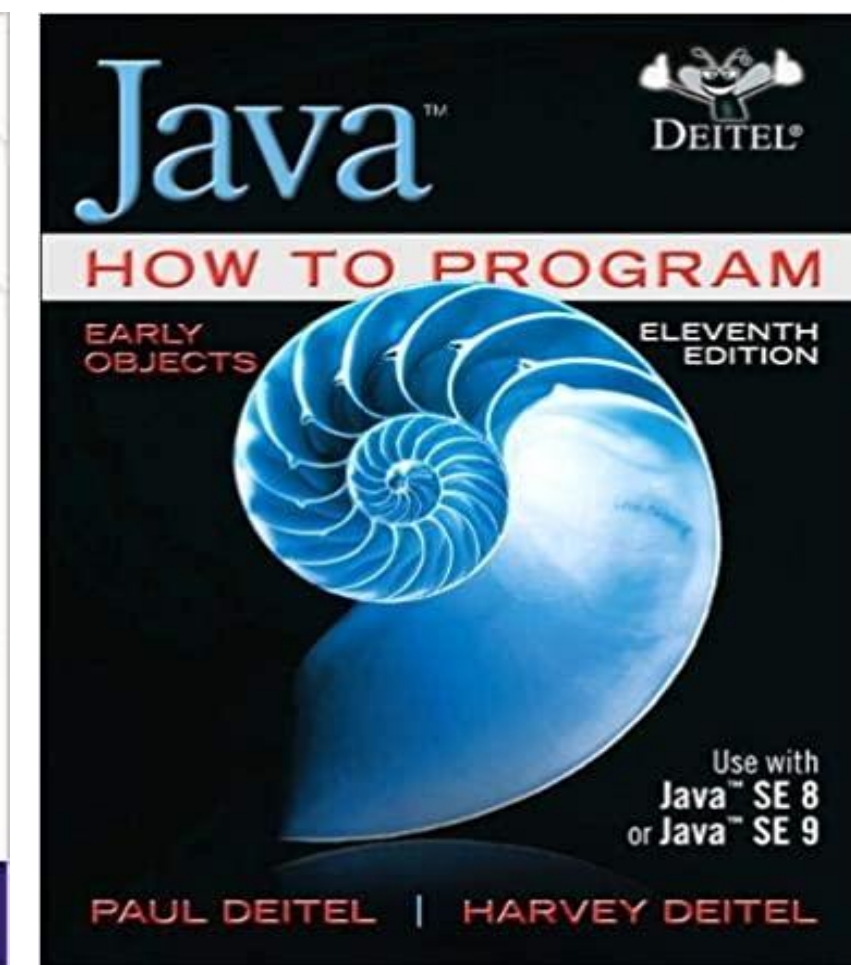
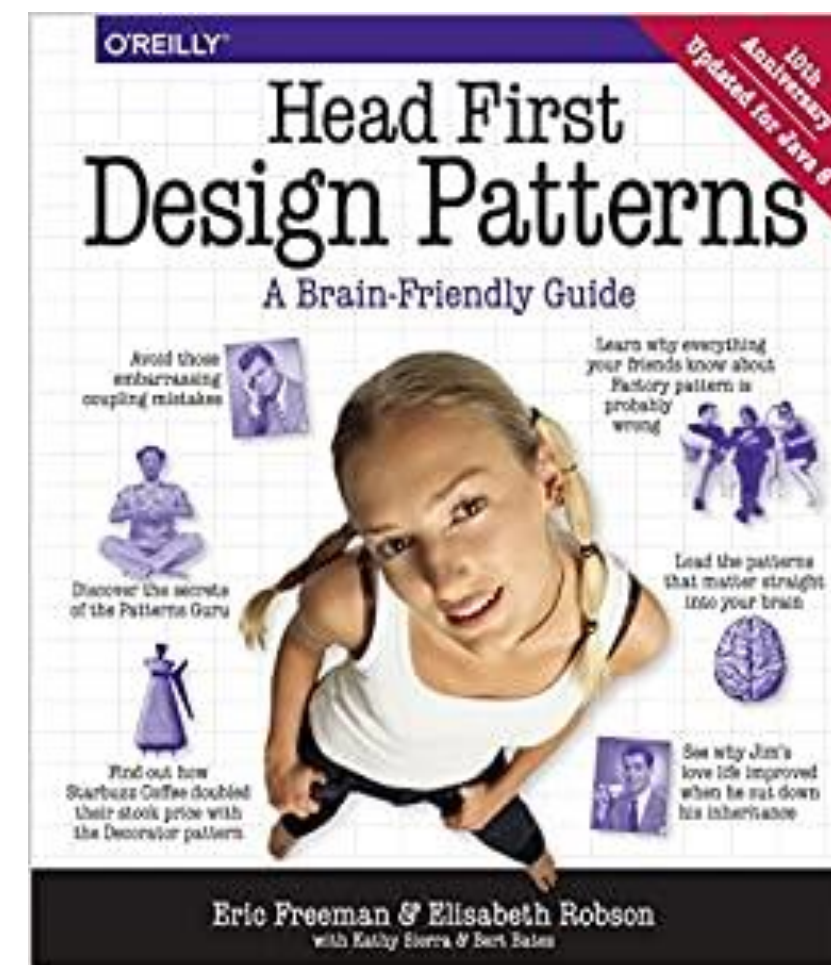
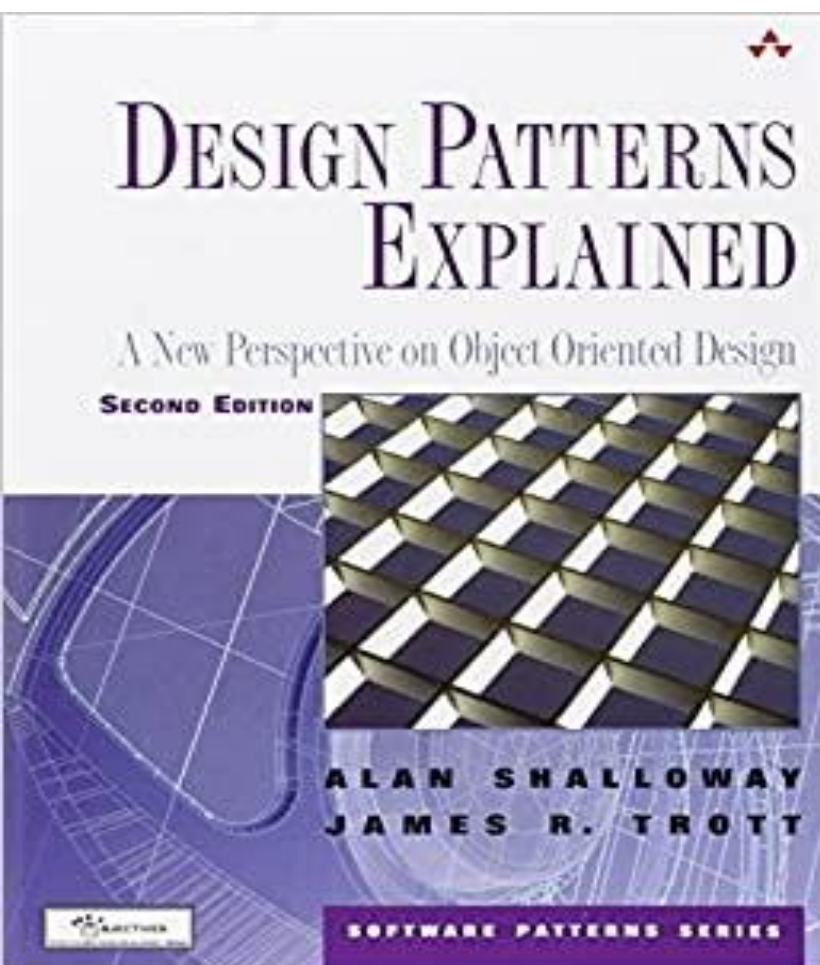
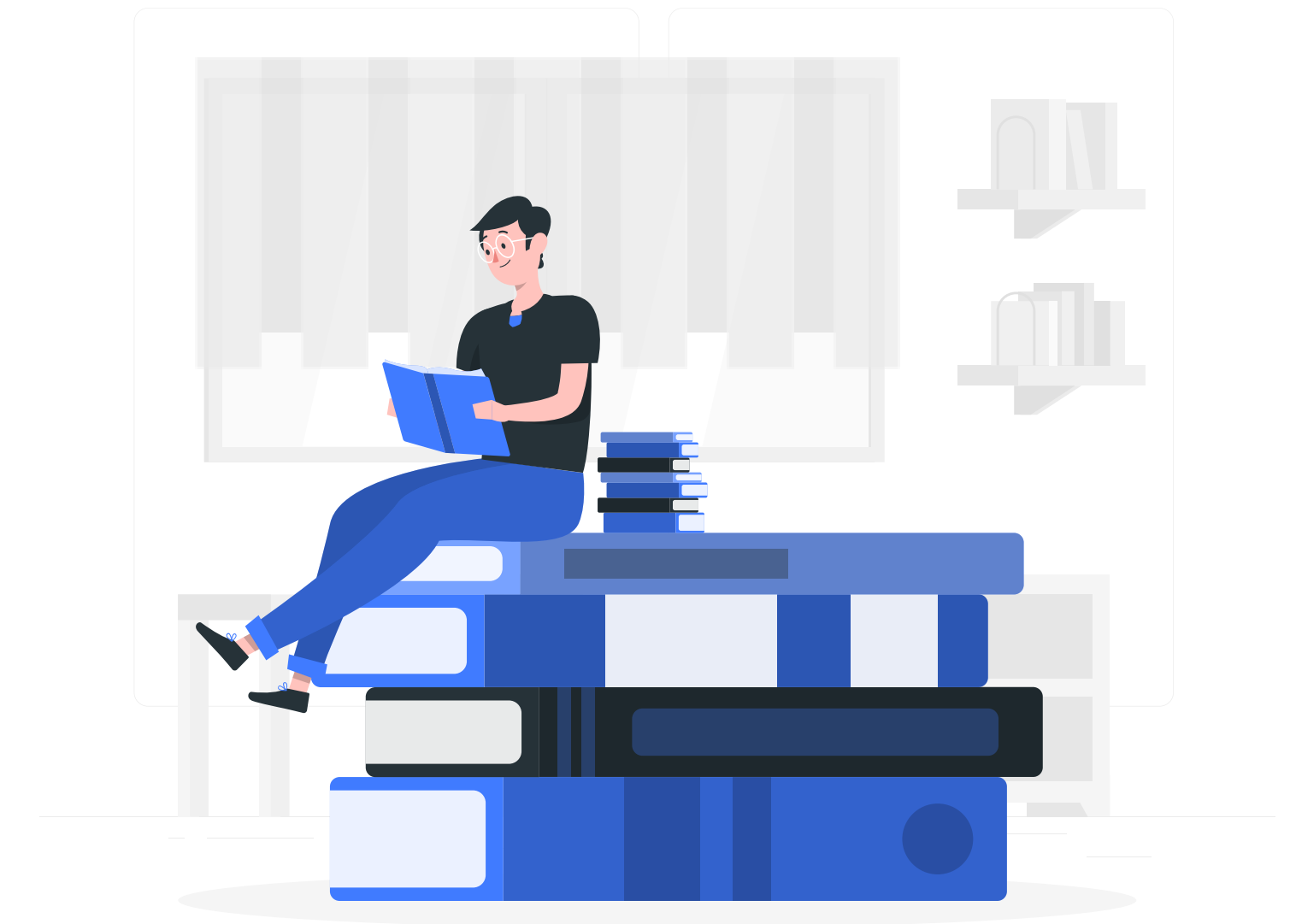


# Textbooks

- None of the following books are required. You can use an alternate resource as a reference.



Additional course material (Articles, Lectures, Links, Videos, etc.) will be distributed in class.





# Review the Syllabus Carefully



## Contents

<b>Course Syllabus</b>	<b>1</b>	<b>COVID Social Distance and Mask</b>	
Course Description	1	Policy	6
Prerequisite	1	Zoom - Guidelines and Expectations	6
Course Objectives	2	General Guidelines	6
Grading	2	Participation Expectations	7
Resources	3	Students with Disabilities	7
<b>Tentative Schedule</b>	<b>4</b>	Academic Integrity	8
Important Dates	4	Academic Policies	8
<b>Course Policies</b>	<b>5</b>	Incomplete Grades	8
Communication Policy	5	Online Course Evaluations	9
Assignments and Exams Policy	5	Diversity and Inclusion	9
Changes to Syllabus	6	Mental Health and Academic Assistance	9

## Object-oriented Software Development

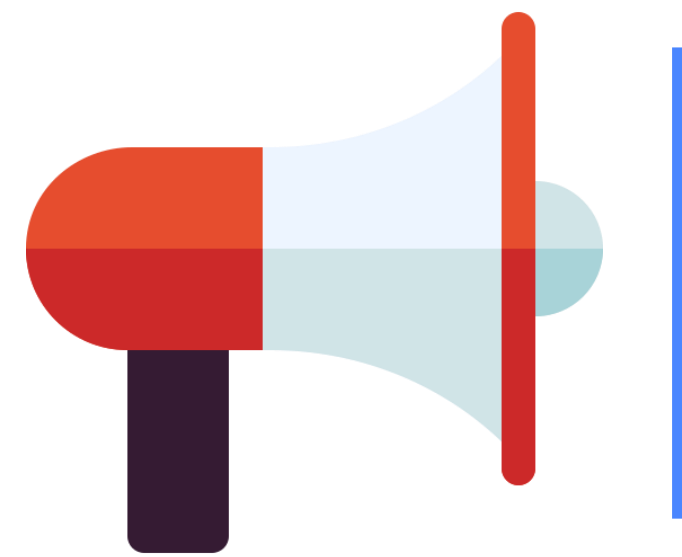
SE 350 (601-610)

**Instructor:** Vahid Alizadeh  
**Email:** [v.alizadeh@depaul.edu](mailto:v.alizadeh@depaul.edu)  
**Office:** CDM 829  
**Office Hours:** Wednesdays 2-5 PM - via Zoom / BlueStar Appointment  
**Lecture:** Tuesdays & Thursdays, 1:30-3:00 PM - via Zoom  
**Course Forum:** MS Teams  
**Quarter:** Spring 2021



*Last update: March 30, 2021*





# Announcements

# Jobs for Bootcamp Grads and Junior Level Programmers

Jr.DevJobsAll JobsPost a JobInsightsCompaniesArticlesCommunitySign UpSign In

## Find your First Developer Job

Jobs for Bootcamp Grads and Junior Level Programmers

Search Jobs by Title, Skills, City/State, Company...Search

Featured Jobs

### 2021 Software Engineer Paid Internship

The Church of Jesus Christ of Latter-day SaintsRiverton, UT

Posting Dates: 01/01/2021 - 04/01/2021 Job Family: Human Resources Department: Information and Co...

View

18 hours ago

### Software Intern

Eastern Research Group, Inc.Lexington, MA

Software Intern to support the Digital Solutions group with a variety of supporting tasks includi...

View

18 hours ago

### Junior Software Engineer

GoFormzSan Diego, CA

GoFormz – San Diego, CA Job Description GoFormz is a San Diego-based startup leading the Mobile D...

View

18 hours ago

### Software Development Engineering Intern - Heterogeneous Computing

Baidu USASunnyvale, CA

We are looking for an AI/ML software intern to participate in building industry-leading AI comput...

View

1 day ago

### Co op/Intern: Packet Software

CienaSan Jose, CA 95134 (North San Jose area)

### Software Engineering Intern - Summer 2021

NXP SemiconductorsSan Diego, CA 92121 (Sorrento Valley area)

An isometric illustration of a person in a white shirt and dark pants standing next to a large, stylized 'JOBS' board. The board is white with a yellow 'JOBS' title and a grid of job listings. A magnifying glass with a yellow handle is positioned in front of the board. To the right of the board, there are three circular icons: a dollar sign, a person icon, and a briefcase icon. The entire scene is set on a light gray circular base.

<https://www.jrdevjobs.com/>

A circular icon with a yellow border. Inside the circle, there is a blue folder icon with a white arrow pointing to the right.

DEPAUL  
UNIVERSITY

Object-oriented Software Development – SE 350 – Spring 2021

32