

# Coding Dojo Vienna: (Async) Clock-In-Kata

Session 20160920

Peter Kofler, 'Code Cop'  
@codecopkofler  
[www.code-cop.org](http://www.code-cop.org)

# Peter Kofler

- Ph.D. (Appl. Math.)
- Professional Software Developer for 15+ years
- “fanatic about code quality”
- Freelance Code Mentor



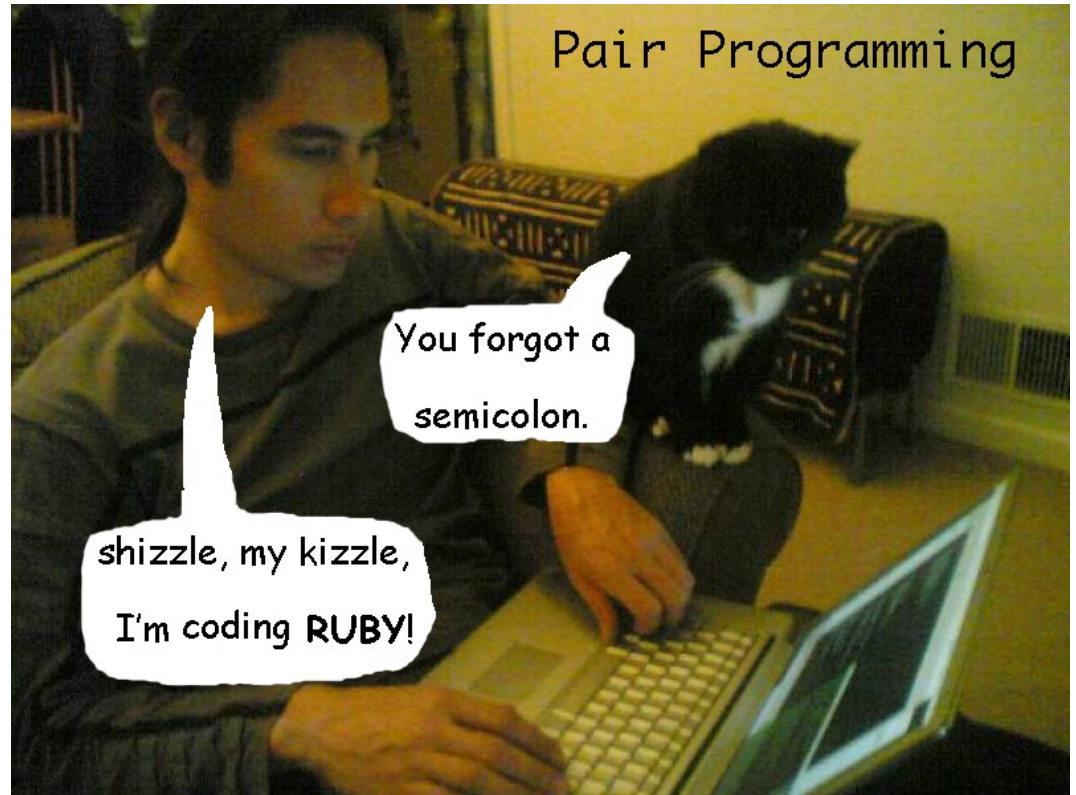
# I help development teams with

- Professionalism
- Quality and Productivity
- Continuous Improvement



# Mentoring

- Pair Programming
- Programming Workshops
- Deliberate Practice, e.g. Coding Dojos



# Developing Quality Software Developers

# Coding Dojo Mindset

- Safe place outside work
- We are here to learn
- Need to slow down
- Focus on doing it right
- Collaborative Game





# Assignment

**Simplex**  
No. 50 vacation  
NAME [Signature]

EXTRA TIME		REGULAR TIME	
IN	OUT	IN	OUT
1st Day		A.M.	2 6 55
		NOON	+45 12:00
		P.M.	9 12:15
		A.M.	5 5:00
		NOON	6 6:50
		P.M.	5 12:20
		A.M.	+20
2nd Day		P.M.	5
		A.M.	

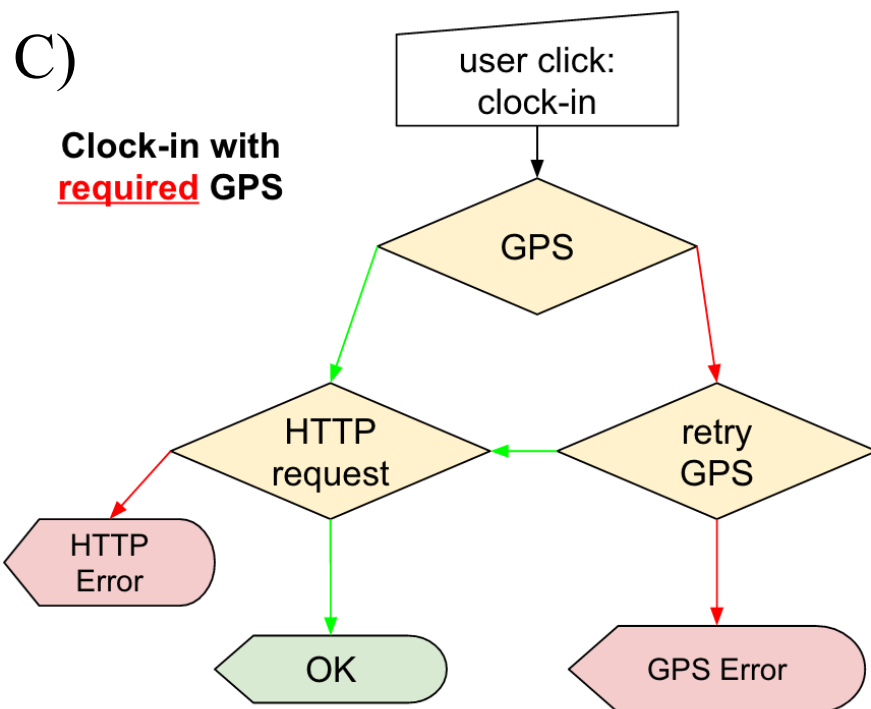
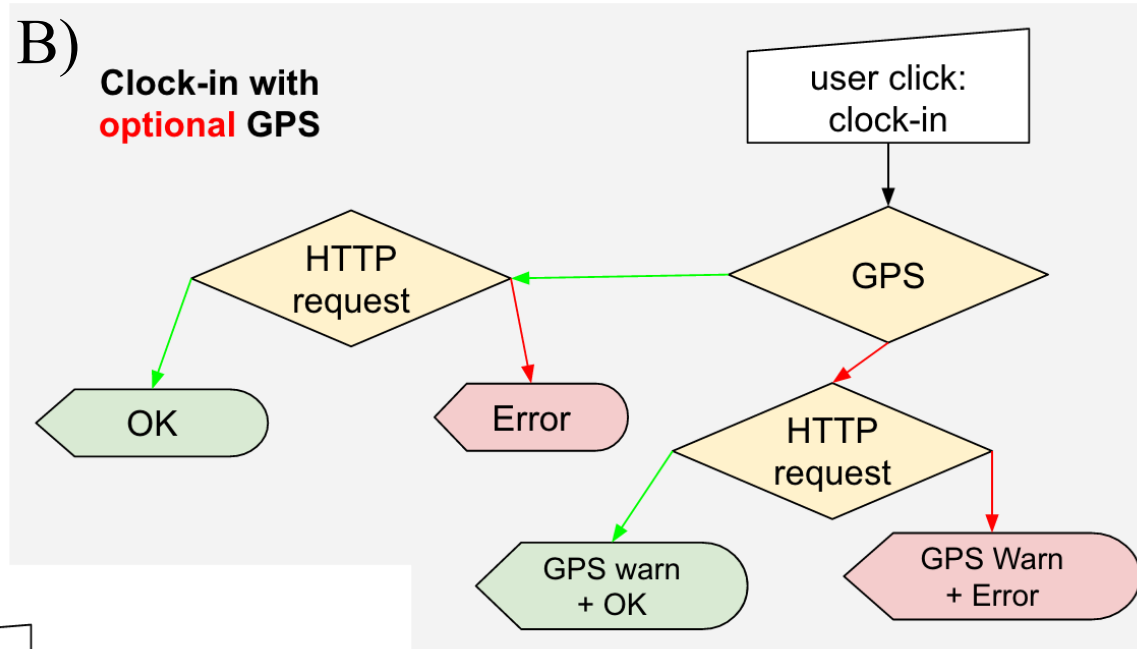
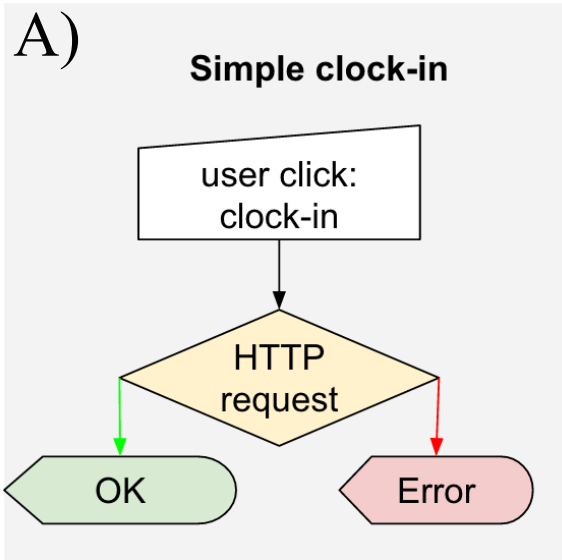
**Simplex**  
No. 52  
NAME [Signature]

EXTRA TIME		REGULAR TIME	
IN	OUT	IN	OUT
1st Day		A.M.	2 6 55
		NOON	+45 12:00
		P.M.	9 12:15
		A.M.	5 5:00
		NOON	6 6:50
		P.M.	5 12:20
		A.M.	+20
2nd Day		P.M.	5
		A.M.	

# Clock-In-Kata

- Build a client interface for time tracking.
  - The time punch (data) is sent to back end.
  - It accepts optional GPS positions.
- 
- Call to server back end is asynchronous.
  - Collecting GPS data is asynchronous, too.
  - Asking user for GPS permission is even more asynchronous. (optional)





#### Bonus

- 3 x retry GPS
- network error
- timeout
- unify all of them

#### Legend

Success →

Error →

asynchronous action

# Focus on “Asynchronous”



# Async Language Features

- Futures / Promises
  - Chaining promises returns new promise.
  - `Future#then` (Dart)
  - `Promise#then` (EcmaScript 6)
  - `CompletableFuture#then*` (Java 8)
- `async` `await` reserved words (Dart, C#)
  - Short cuts to Future API

# How to Start

- Find a pair.
- Create new project.
- Add a testing framework.
- Start with Case A)
- Mock the asynchronous backend call.
- Keep the async code clear and readable.
- What is executed sequential/parallel?

Don't Focus on  
Getting it Done.  
Focus on Doing  
It Perfectly.

→ Practice



# Closing Circle

- What did you learn today?
- What surprised you today?
- What will you do differently in the future?





# Peter Kofler

@codecopkofler

[www.code-cop.org](http://www.code-cop.org)

Kata by

# Wolfram Kriesing

@wolframkriesing

<https://github.com/wolframkriesing/clock-in-kata>

# CC Images

- Bruce <http://www.flickr.com/photos/sherpas428/4350620602/>
- pairing <http://www.flickr.com/photos/dav/94735395/>
- dojo <http://www.flickr.com/photos/49715404@N00/3267627038/>
- time punches  
<https://www.flickr.com/photos/qchristopher/3057895390/>
- async [https://www.flickr.com/photos/ilo\\_oli/187670793/](https://www.flickr.com/photos/ilo_oli/187670793/)
- wants you <http://www.flickr.com/photos/shutter/105497713/>