

# Reactive Programming

**Duration: 2 days (9 hours each day with working lunch)**

Reactive programming is gaining a lot of attention. A combination of some fundamental approaches—functional programming, event based programming, loose coupling, and error resilience—to mention a few, have come together as a set of tools to create highly responsive and performant applications. This course will help developers get up to speed with creating reactive applications.

The course has a good balance of interactive lectures and hands-on exercises. The attendees are expected to pair-up and work on the lab exercises. The instructor will assist the attendees as they work on the labs. The objective of the course is for the attendees to gain an in depth practical knowledge of the concepts so they can put them to immediate use on their real projects.

The course content will be customized to meet your teams' specific needs. Please review this detailed outline and suggest changes (additions, deletions, modifications) as you feel fit.

## Topics

### Reactive Applications

- \* Why a new/different programming model?
- \* What's reactive programming?
- \* Reactive Manifesto
- \* Characteristics of Reactive Applications
- \* State of software technologies
- \* The next logical step
- \* Exercise

### Functional Programming and Reactive

- \* Functional Style
- \* Function Composition
- \* Lazy Evaluations
- \* Thinking of Streams
- \* Streams of data
- \* Relationship to Reactive Programming
- \* Exercise

### APIs for Reactive Programming

- \* Observable
- \* Beyond the observer pattern
- \* Features of observables
- \* Iterators vs. observables
- \* Propagating Signals and errors
- \* Encapsulation of services

- \* Decoupling of clients
- \* Exercise

#### Client-Server Interactions

- \* Synchronous
- \* Asynchronous
- \* Working with data
- \* Unsubscribing
- \* Exercise

#### Graceful Error Handling

- \* Failure as first class citizen
- \* Communicating Failures
- \* Graceful termination
- \* Handling Failures
- \* Resuming service on Failure
- \* Communication model
- \* Exercise

#### Working with Streams

- \* Starting
- \* Filters
- \* Skips and Takes
- \* Supporting different client needs
- \* Exercise

#### Programming Asynchrony

- \* Subscription options
- \* Threading and threading options
- \* Schedulers
- \* Working with schedulers
- \* Exercise

#### Configuring for Speed

- \* Stream Generation
- \* Varied Speed of Clients
- \* Back pressures
- \* Using Producers
- \* Exercise

#### Reactive at Large

- \* Reactive Systems
- \* Reactive Languages
- \* Architecting Reactive Systems
- \* Exercise

## About the Instructor

Dr. Venkat Subramaniam is an award-winning author, founder of Agile Developer, Inc., creator of agilelearner.com, and an instructional professor at the University of Houston.

He has trained and mentored thousands of software developers in the US, Canada, Europe, and Asia, and is a regularly-invited speaker at several international conferences. Venkat helps his clients effectively apply and succeed with sustainable agile practices on their software projects.

Venkat is a (co)author of multiple technical books, including the 2007 Jolt Productivity award winning book Practices of an Agile Developer. You can find a list of his books at agiledeveloper.com. You may read more about Venkat and Agile Developer, Inc. at <http://agiledeveloper.com>.

