Northeastern University
College of Engineering

CSYE 7200 Big Data System Engineering using Scala

Course Information
Course Title: Big Data System Engineering using Scala
Course Number: CSYE 7200
Term and Year: Spring 2022
Credit Hour: 4
Course Format: On-Ground

Instructor Information
Full Name: Robin Hillyard
Email Address: r.hillyard@northeastern.edu

Course Prerequisites
N/A

Course Description
Covers the fundamentals of functional programming with Scala and seeks to provide a basic, practical foundation for students who want to use it as a language for working with big-data platforms. Scala is one of a new breed of general-purpose functional programming languages that is strongly typed and is object oriented. It runs on the Java virtual machine and is able to share libraries from the vast collection of open-source projects written in Java. For these reasons it is readily accessible by programmers of Java, C++, and similar languages.

Standard Learning Outcomes

Learning outcomes common to all College of Engineering Graduate programs:

1. *An ability to identify, formulate, and solve complex engineering problems.*
2. *An ability to explain and apply engineering design principles, as appropriate to the program's educational objectives.*
3. *An ability to produce solutions that meet specified end-user needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors.*

The Information Systems Program accepts students of different engineering backgrounds with minimum programming skills and produces first class Information Systems engineers that operate at the intersection of real-world complexity, software development, and IT management. Graduating students will be able to construct end-to-end advanced software applications that meet business needs.

Specific Learning Outcomes for the Information Systems program:

1. *Create a strong technical foundation through diverse, high-level courses*
2. *Built crucial interpersonal skills needed to succeed in any industry*

Recommended text(s)

Programming in Scala—**Odersky, Spoon & Venners, Artima** (4th edition)
Functional Programming in Scala—**Chiusano & Bjarnason, Manning**

These are both excellent texts. The first is the definitive guide to Scala co-written by the originator of the language. The second is a beautifully written introduction to the concepts of functional programming, with the advantage that it uses Scala.

Course Objectives

Spark has revolutionized the approach to processing big data, abstracting away the details of map/reduce such that programmers are hardly aware of it. While much work with Spark can be programmed with Java, Python, R, or even plain old SQL, it is often the ETL (ingestion) phase of Big Data work which particularly requires Scala. Why should this be so? Most non-functional languages
are oriented towards doing things as long as everything is working fine. However, real life encounters nulls and occasionally causes exceptions to be thrown. These abnormal situations are very well handled using Scala. Secondly, it is when gathering data that it is most important to be protected by type-safety.

In any case, Spark is implemented in Scala. Thus, programming in Scala helps you not only with best practices, but also enables you to look "under the hood". But functional programming (*fp*) is not only ideal for parallel programming with Spark. *fp* is ideally suited to concurrent programming (all modern computing is potentially concurrent) because side effects and mutable state are either eliminated or carefully encapsulated.

Nevertheless, fp requires a **different way of thinking** from imperative programming (Java, C[++], etc.). This class aims to cover the fundamentals of *fp* (in Scala), and to provide a basic, practical foundation for many different types of programming. Topics to be covered, in addition to all basic programming techniques are: numerical programming, reactive programming (using Akka), parser-combinators and DSLs, testing frameworks and getting the job done. While *fp* has a solid mathematical foundation, the mathematics required is really just basic logic and axioms. You don't need a "higher math" background.

The last third of the class will be largely concerned with projects which will not only test your knowledge of Scala but will give you a great opportunity to tackle something really interesting and, hopefully, useful. Typical projects train and run a machine learning model using Spark.

Grading Breakdown

20% mid, 25% final, 30% project, 25% homework.

Project Information

Projects will normally be worked on in pairs (or trios) and will implement some analysis of Big Data (possibly streaming) typically using Spark and maybe Databricks Notebook. Alternatively, a project might implement a reactive system. Projects must include some significant Scala coding, with unit tests, and

demonstrate scalability. For more detail and ideas, see the Project module on Canvas.

Course Schedule (may vary somewhat)

Week 1		**Introduction; Big Data systems; Spark overview; Looking under the hood: Scala; Scala and Functional Programming.**
Week 2		**Scala (continued); Important concepts. Parallel Processing and Mutable State.**
Week 3		**More Advanced Scala Concepts (REPL, substitution, type inference, lazy functions, lists and streams, generics and variance); Dealing with Exceptional Conditions.**
Week 4		**Collections; Lazy Lists; Managing State; Types.**
Week 5		**Functional composition and for comprehensions; recursion.**
Week 6		**Syntax; Type Declarations; Functions, methods & operators; Specifications & Unit Tests.**
Week 7		**Implicits; Serialization/de-serialization; Parallel Processing and Futures; Monoids, functors, and monads.**
Week 8		**Mid-term exam; Syntactic sugar; Repositories.**
Week 9		**Enumerated Types; Actors; Syntactic sugar (continued) and pattern matching; Tour of the API; Parsing and DSLs.**
Week 10		**Spark details; GraphX, MLlib; Spark Streaming and Spark SQL.**
Week 11		**Play/Activator; Numerical Computing.**
Week 12 thru 13		**Projects and other topics not already covered.**
Week 14		**Project presentations**
Week 15		**Final exam and remaining Project presentations.**

A word on the difficulty of the course

Ideally, this should not be a particularly difficult course. The concepts are all quite simple. However, it's my experience that students often struggle with the different idioms used in functional programming. In my opinion, the hard work that you must put in to feel really comfortable with these concepts is well worth it, and will reward you throughout your career, particularly as functional programming becomes, more and more, the "norm." **But**, if you are looking for an easy class that you can snooze your way through, please don't choose this class. You will inevitably be disappointed.

End-of-Course Evaluation Surveys

Your feedback regarding your educational experience in this class is very important to the College of Professional Studies. Your comments will make a difference in the future planning and presentation of our curriculum.

At the end of this course, please take the time to complete the evaluation survey at **https://neu.evaluationkit.com**. Your survey responses are **completely anonymous and confidential**. For courses 6 weeks in length or shorter, surveys will be open one week prior to the end of the courses; for courses greater than 6 weeks in length, surveys will be open for two weeks. An email will be sent to your HuskyMail account notifying you when surveys are available.

Academic Integrity

A commitment to the principles of academic integrity is essential to the mission of Northeastern University. The promotion of independent and original scholarship ensures that students derive the most from their educational experience and their pursuit of knowledge. Academic dishonesty violates the most fundamental values of an intellectual community and undermines the achievements of the entire University.

As members of the academic community, students must become familiar with their rights and responsibilities. In each course, they are responsible for knowing the requirements and restrictions regarding research and writing, examinations of whatever kind, collaborative work, the use of study aids, the appropriateness of assistance, and other issues. Students are responsible for learning the conventions of documentation and acknowledgment of sources in their fields. Northeastern University expects students to complete all examinations, tests, papers, creative projects, and assignments of any kind according to the highest ethical standards, as set forth either explicitly or implicitly in this Code or by the direction of instructors.

Go to http://www.northeastern.edu/osccr/academic-integrity-policy/ to access the full academic integrity policy.

Student Accommodations

Northeastern University and the Disability Resource Center (DRC) are committed to providing disability services that enable students who qualify under Section 504 of the Rehabilitation Act and the Americans with Disabilities Act Amendments Act (ADAAA) to participate fully in the activities of the university. To receive accommodations through the DRC, students must provide appropriate documentation that demonstrates a current substantially limiting disability.

For more information, visit http://www.northeastern.edu/drc/getting-started-with-the-drc/.

Library Services

The Northeastern University Library is at the hub of campus intellectual life. Resources include over 900,000 print volumes, 206,500 e-books, and 70,225 electronic journals.

For more information and for Education specific resources, visit http://subjectguides.lib.neu.edu/edresearch.

24/7 Blackboard Technical Help

For immediate technical support for Blackboard, call 617-373-4357 or emailhelp@northeastern.edu

Within Blackboard, open a support case via the red support button on the right side of the screen, click Create Case

myNortheastern, e-mail, and basic technical support
Visit the Information Technology Services (ITS) Support Portal
Email: help@northeastern.edu
ITS Customer Service Desk: 617-373-4357

Diversity and Inclusion

Northeastern University is committed to equal opportunity, affirmative action, diversity and social justice while building a climate of inclusion on and beyond campus. In the classroom, member of the University community work to cultivate an inclusive environment that denounces discrimination through innovation, collaboration and an awareness of global perspectives on social justice.

Please visit http://www.northeastern.edu/oidi/ for complete information on Diversity and Inclusion

TITLE IX

Title IX of the Education Amendments of 1972 protects individuals from sex or gender-based discrimination, including discrimination based on gender-identity, in educational programs and activities that receive federal financial assistance.

Northeastern's Title IX Policy prohibits Prohibited Offenses, which are defined as sexual harassment, sexual assault, relationship or domestic violence, and stalking. The Title IX Policy applies to the entire community, including male, female, transgender students, faculty and staff.

In case of an emergency, please call 911.

Please visit www.northeastern.edu/titleix for a complete list of reporting options and resources both on- and off-campus.