CS – 19 Open Source Tools

Objectives:

- Understanding Open Source Philosophy
- Identify and explore a range of open source tools.
- Understand the collaborative nature of open source development and the role of communities.
- Learn how to contribute to open source projects through code contrications, documentation, bug reporting etc.

Prerequisites:

- Basic Computer Skills
- Basic knowledge of Version Control

No	Topics	Details

1	**Open Source Softwares**	• Understanding Open Source Software
		  • Definition
		  • Principles
		  • History and evolution
		• Open-Source Licensing
		  • Overview
		  • Rights and responsibilities of users and developers under open source licenses
		  • Contracts & licenses and related issues
		• Application of Open sources
		• Open Sources Operating System:
		  • FEDORA
		  • UBUNTU

2	**Open Source Development and Collaboration**	• Version Control with Git
		  • Introduction to version control systems.
		  • Git fundamentals, repositories, commits, branches and merges
		• Open Source Project Management
		  • Overview of Project Management Methodologies (Agile)
		  • Tools for Project Planning, Task Tracking and Team Collaboration (Trello)
		  • Contributing to open source projects: Issue Tracking, Pull Requests, Code Reviews.

3	**Case Studies**	• Apache • Linux Operating System

Seminar - 5 Lectures
Expert Talk - 5 Lectures
Test - 5 Lectures

Total Lectures 30 + 15 = 45

Reference Books:

- "Producing Open Source Software: How to Run a Successful Free Software Project" by Karl Fogel
- "Git Pocket Guide: A Working Introduction" by Richard E. Silverman
- "The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win" by Gene Kim, Kevin Behr, and George Spafford
- KailashVadera, Bhavyesh Gandhi, “Open Source Technology”, Laxmi Publications Pvt. Ltd 2012, 1st Edition.
- Fadi P. Deek and James A. M. McHugh, “Open Source: Technology and Policy”, Cambridge Universities Press 2007.

Course Outcomes:

- Recognize the benefits and features of Open Source Technology and to interpret, contrast and compare open source products among themselves
- Use appropriate open source tools based on the nature of the problem
- Write code and compile different open-source software.