BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI
WORK INTEGRATED LEARNING PROGRAMMES

COURSE HANDOUT (Flipped)

Part A: Content Design

Course Title	Agile Software Process
Course No(s)	SE ZG544
Credit Units	4 (1 + 1 + 2) (Class Hour + Assignment Hour + Student Preparation Hours) / week (Avg.)
Course Author	K G Krishna
Version No	1.1
Date	23/08/2022

Course Objectives

No	Objective
CO1	To make software engineers trained in legacy software development practices adopt Agile Methodologies for rapid development and deployment of products and services
CO2	To expose various Agile Methods currently in practice and their applicability in various scenarios of software development and testing
CO3	To enable software engineers and project managers decide on the right tools and techniques considering the implementation challenges, pros and cons of each.

Course Contents

Introduction to Agile; Basics of Agile Software Development Approaches; Principles of Agile; Agile Methodologies; Release Planning; Roles and Artifacts in Agile; Agile Requirements; Iteration Planning and Ceremonies; Executing a Sprint; Agile Metrics; Agile Testing and Maintenance; Agile Pitfalls; Ensuring Agile Success

Text/Reference Books

T1	Agile and Iterative Development A Manager's Guide - Craig Larman / Pearson Education - 2004.
T2	Agile Project Management for Dummies - Mark C. Layton, John Wiley & Sons - 2012
T3	Head First Agile, by Andrew Stellman, Jennifer Greene. Released September 2017 Publisher(s): O'Reilly Media, Inc.

T4	Agile Practice Guide AUTHOR: PROJECT MANAGEMENT INSTITUTE
R1	Agile Testing: How to Succeed in an Extreme Testing Environment - John Watkins, Cambridge University Press 2009
R2	Managing Agile Projects , Multi-Media Publications 2004

** Course-code specific to collaborating organization

Glossary of Terms

Module	**M**	Module is a standalone quantum of designed content. A typical course is delivered using a string of modules (typically 10 – 15). M2 means module 2.
Lecture Session	**LS**	A Module consists of several Lecture Sessions (LS) in sequence; Each LS covers a particular topic in its entirety; All Lecture Sessions are video recorded content and are available online for anytime-anywhere viewing by Students; LS1.2 denotes Lecture Session number 2 in Module 1
Video Segment	**VS**	Each Lecture Session (LS) may further be divided into several small (~10-20min) Video Segments (VS) illustrating one sub-topic or concept; LS 1.2 VS 3 (or LS1.2.3) indicates Video Segment number 3 of Lecture Session 2 of Module 1; There may be short-quizzes in between Video Segments to assess Students' understanding of the topic
Contact session	**CS**	Contact sessions refer to physical class-room sessions meant for elaboration of difficult-to-understand concepts, discussions on case-problems, case-studies, and Q&A session with students etc., to be taken up by the course instructor during the contact hours. A Contact Session is built by stringing a bunch of contact session topics. CS3.2 = Contact session sub-module 2 associated with Module 3 CS3.0 = Contact session associated with all sub-modules of Module 3
Case Problem	**CP**	Case problems/topics (experienced by practising Usability Professionals/Product Designers) to be discussed in the class
Self-Study	**SS**	Specific content assigned for self-study by the Student
Homework	**HW**	Specific problems/assignments/lab exercises assigned by Instructor as homework to Students

Teaching Methodology
Flipped Model of Learning (Video Lectures + Case-Studies/Tutorials in Contact Sessions)

Course Organization
Introductory Video/Document: *<< Introduction of Instructors, overview of the course, structure and organization of topics, guidance for navigating the content, and expectations from students>>*
<Module #>.<Lecture Sequence #>.<Video Segment #>

Lecture Sessions: Each of the Lecture Sessions (LSx.y) are delivered via series of several pre-recorded **Video Segments** (VS) of **15 – 20mins** duration followed by:

Contact Sessions: Each of the Contact Session (CSx.y) of 1.5-2Hr each for Illustrating the concepts discussed in the videos with exercises, tutorials and discussion on case-problems (wherever appropriate); contact sessions (CS) may cover more than one Lecture Session (LS). The schedule of Contact Sessions will be prepared and distributed by the Instructor before commencement of the course (Refer Part-B of this Handout.)

Module Description/Topics

Module #	Module Title	LS #	Lecture Title	Duration (~mins)	Text/Chap Ref *
1	Agile Methods - An Introduction	1.1	Traditional Software Development Practices	30	T2-Chap1
		1.2	Need for Agile Methods	20	
		1.3	Benefits of Agile Methods	20	
2	Agile Software Development	2.1	Iterative & Incremental Approaches	35	T1-Chap5 T2-Chap4
		2.2	Popular Agile Methods	35	
3	Agile Principles and Manifesto	3.1	Vision and Principles and Manifesto	50	T1-Chap3
4	Agile Methodologies	4.1	Overview of SCRUM Methodology	40	T1-Chap7,8
		4.2	Extreme Programming (XP)	25	
		4.3	Test-Driven Development (TDD)	25	
5	Agile Requirements	5.1	Requirements Management	30	T1-Chap5, T2-Chap7,11
		5.2	Effort Estimation	30	
6	Release Planning in Agile	6.1	Characteristics of Agile Planning	20	T1-Chap7, T2-Chap7,8
		6.2	Agile Release Planning	40	
7	Iteration Planning	7.1	Sprint as an Iteration	25	T2-Chap8
		7.2	Velocity and Capacity based Planning	15	
		7.3	Release Sprint Planning	20	
8	Executing a Sprint	8.1	Sprint Ceremonies	30	T2-Chap9,10
		8.2	Sprint Reviews and Retrospectives	15	
9	Agile Metrics and Tools	9.1	Overview of Agile Metrics	30	T2-Chap19, T1-Chap14
		9.2	Tools for Agile Project Management	15	
10	Quality Management in Agile	10.1	Managing Quality in Agile Project	30	T2-Chap14
		10.2	Managing Risks in Agile	30	
11	Agile Myths and Pitfalls	11.1	Common Mistakes and Myths in Agile	40	T1-Chap2,7,8,11
		11.2	Predictive Planning vs Adaptive Planning	15	
		11.3	Distributed Agile	20	
12	Ensuring Agile Success	12.1	Managing Change	45	T2-Chap16,17
		12.2	Evolution of Agile with Times	25	

While effort is made to ensure the topics covered in this course are in alignment with referenced text-books, due to changing technologies and emerging practices in this field, it is strongly advised that students refer to their own sources on the net or their own organizations for comprehensive understanding of the concepts.

Content Structure *(Pre-recorded Video Modules - Contact-Sessions)*

Module #	ModuleTitle	Contact Session (CS) Topics
1	Agile Methods - An Introduction	**CS1: Discuss** the motivation behind Agile Methods and why the traditional Waterfall model was felt not relevant to modern day computing (post 2000, in particular); Is Agile Process a watered-down method or does it compromise the defintion of Process; Let students reflect on their own practical constraints/issues they face when they have to meet challenges of time-to-market product development effort.
2	Agile Software Development	**CS2**: Let students **debate** on how the Iterative and Incremental methods espoused during 'Waterfall' era (as discussed in the classic Roger Pressman's book on SE) is transformed in the Agile era; **discuss** why customer will feel more comfortable with Agile; Is Agile a quick-and-dirty method or an evolutionary approach to complex software product?
3	Agile Principles and Manifesto	**CS3: Discuss** the vision of Agile champions in the late 90's and how it got acceptance over time; by walking through the Manifesto, make it clear to students that being Agile is more about flexibility, adaptation and customer-orientation rather than down-playing process and its associated documentation overheads.
4	Agile Methodologies	**CS4:** By **discussing** some recent examples of Agile practices (shared by students), classify them into one of the Agile Methods; Demystify Agile as not just one method or 'process', but a set of Techniques/Practices such as Extreme Programming, Pair Programming, TDD, etc.
5	Agile Requirements	**CS5: Hands-on Exercise:** by listing a hypothetical software product (say, mobile app), let students articulate the requirements as a set of 'user experiences' or stories by sketching or narrating the actual end-users's screen navigation; compare each students' stories and how they actually connect to user's pain-points or goal; also illustrate the importance of 'Usability' or 'User Experience' design as part of this exercise
6	Release Planning in Agile	**CS6: Case-study** of one Project following Agile Practices; breaking down the requirements into a set of Releases and the criteria used for prioritizing the Releases; how applying a Game metaphor to software development transforms project into a collaborative and knowledge-sharing endeavour with active participation of all participants.
7	Iteration Planning	**CS7: Discuss** why today's software product can't be developed in one instance or cycle; how to capture real users' intent or pain-points; how do you decide how many iterations are required; what happens when the development is not time-boxed as Sprints
8	Executing a Sprint	**CS8: Discuss and Debate** how one executing one Sprint is differerent or same as executing a project using Waterfall model for that Sprint duration; role of Scrum master vs traditional Project Manager; **Role-Play**: let students identify themselves different roles in the 'Agile Project Game' and describe their activities and how they collaborate with each other towards their common goal.
9	Agile Metrics and Tools	**CS9: Discuss** the role of Metrics in Project Management and how are they different in Agile Methods; let students share the metrics used in their own projects and how they are related to project goals; are metrics unique to organization or universal, discuss.
10	Quality Management in Agile	**CS10: Discuss** how Testing Methodologies they learnt as part of Software Testing course can be tuned to Agile Methods; can Testing and development integrated into one phase, if so, does it imply compromising the quality of the product?
11	Agile Myths and Pitfalls	**CS11: Discuss** when adopting Agile Methods is the best strategy and its potential drawbacks; debate can Agile teams be distributed, if so how to ensure effective communication and collaboration; let students explore the current methods of DevOps, Microservices & Cloud-based app development and use of Low/No-code platforms integrate into Agile philosophy
12	Ensuring Agile Success	**CS12: Discuss** the role of cohesive team and the 'flexible' work environments demanded by today's millenial generation; how to ensure team-work if members are geographically distributed; continuous learning of best practices from industry peers and updating oneself professionally by attending conferences in this field; the role played by organizational culture and incentives, discuss.

Part B: Course Handout

Academic Term	First Semester 2022-2023
Course Title	Agile Software Process
Course No	SS ZG544
Lead Instructor	Ananthraman K
Instructor(s)	

Learning Outcomes

LO1	Students to understand and adopt Agile Methods in their projects by understanding the benefits and challenges involved
LO2	Project Managers to be able to better estimate software projects using Agile Methods and manage changing requirements
LO3	Engineers involved in Testing and Maintenance projects to be able to implement Agile Test-Driven Development (TDD)

Course Introduction & Motivation

This course--consisting of lecture-videos, case-studies and tutorials --aims to introduce Agile Methods for development of software systems. Agile Methods are set of Practices, Techniques and Processes which are based on Iterative model of development with emphasis on continuous collaboration and communication to address the inherent limitations of traditional Waterfall-based software development methodologies. Beginning in the early 2000s, Agile Methodologies have started gaining traction and Scrum—one of the most popular Agile Process—has now become the de facto model of development in many IT organizations.

This course is highly recommended for all software engineers and project managers engaged in the design, development and testing of software products and services meeting time-to-market constraints in today's competitive environment accommodating changing customer requirements and time-to-market constraints. This course covers the follows the topics:

- Traditional Development Methodologies vs. Agile Methods
- Principles of Agile & Agile Manifesto
- Requirements Management in Scrum
- Product and Release Planning in Scrum
- Scrum Ceremonies
- Metrics for Agile Project Management
- Test-Driven Development
- Common Agile Myths and Pitfalls
- Emerging Practices using Agile

Teaching Methodology *(Flipped Learning Model)*

The pedagogy for this course is centred around flipped learning model in which the traditional class-room instruction is replaced with recorded lectures to be watched at home as per the

student's convenience and the erstwhile home-working or tutorials become the focus of classroom contact sessions. In this model, in addition to walking-through the topic-centric exercises and filling the gaps in student's comprehension, the contact sessions are also enriched with discussion on organization-specific practices and case-problems experienced by typical practising professionals in the industry.

A catalogue of such case-studies with continuous update based on inputs from collaborating organizations is maintained by the lead instructor. In addition, **problem-sets** are made available by the instructor as take-home exercises for student's practice.

As part of evaluation, 2-4 week long **take-home assignments** of interest to individual student organizations are provided and at the end of which students are expected to prepare a detailed report or make a presentation to the class.

Course Delivery (by Instructor *via Contact Sessions*)

- There are 12 Contact Sessions (of 1.5 - 2 hours each)--6 before mid-semester and 6 post-mid-semester over a period of 22 weeks with alternate weeks for home-watching of Recorded Lecture Sessions (LS)

- The 6th & 12th Contact Sessions are planned for review of topics pre-mid-semester and pre-end-semester examinations

- The students are expected to watch the prescribed recorded Lecture Sessions (LS) before attending the above Contact Sessions

Instruction Delivery *(via Recorded Lecture Sessions)*

- Students are expected to watch the pre-recorded digital content (below Modules/Lecture Sessions) at their own pace before attending scheduled Contact Sessions (CS)

- Minimum of one Contact Session (topics listed under CSx) is to be planned for each core module (M1…M12)

Pre-recorded Modules/Lecture Sessions

Module #	Module Title	LS #	Lecture Title	Duration (~mins)	Text/Chap Ref *
1	Agile Methods - An Introduction	1.1	Traditional Software Development Practices	30	T2-Chap1
		1.2	Need for Agile Methods	20	
		1.3	Benefits of Agile Methods	20	
2	Agile Software Development	2.1	Iterative & Incremental Approaches	35	T1-Chap5 T2-Chap4
		2.2	Popular Agile Methods	35	
3	Agile Principles and Manifesto	3.1	Vision and Principles and Manifesto	50	T1-Chap3
4	Agile Methodologies	4.1	Overview of SCRUM Methodology	40	T1-Chap7,8
		4.2	Extreme Programming (XP)	25	
		4.3	Test-Driven Development (TDD)	25	
		4.4	Lean Software Development	20	T3
5	Agile Requirements	4.5	Kanban	20	T3
		5.2	Effort Estimation	30	
6	Release Planning in Agile	6.1	Characteristics of Agile Planning	20	T1-Chap7, T2-Chap7,8
		6.2	Agile Release Planning	40	
7	Iteration Planning	7.1	Sprint as an Iteration	25	T2-Chap8
		7.2	Velocity and Capacity based Planning	15	
		7.3	Release Sprint Planning	20	
8	Executing a Sprint	8.1	Sprint Ceremonies	30	T2-Chap9,10
		8.2	Sprint Reviews and Retrospectives	15	
9	Agile Metrics and Tools	9.1	Overview of Agile Metrics	30	T2-Chap19, T1-Chap14
		9.2	Tools for Agile Project Management	15	
10	Quality Management in Agile	10.1	Managing Quality in Agile Project	30	T2-Chap14
		10.2	Managing Risks in Agile	30	
11	Agile Myths and Pitfalls	11.1	Common Mistakes and Myths in Agile	40	T1-Chap2,7,8,11
		11.2	Predictive Planning vs Adaptive Planning	15	
		11.3	Distributed Agile	20	
12	Ensuring Agile Success	12.1	Managing Change	45	T2-Chap16,17
		12.2	Evolution of Agile with Times	25	

Contact Session Plan

- o Each Module (M#) covers an independent topic and module may encompass more than one Recorded Lecture Session (LS).

- o Contact Sessions **(1.5 - 2hrs each week)** are scheduled **alternate weeks** after the student watches all Recorded Lecture Sessions (LS) of the specified Modules during the previous week

- o In the flipped learning model, Contact Sessions (CSx) are meant for in-classroom discussions of case-studies, tutorials/exercises to be done by students or responding to student's questions on topics--may cover more than one Module/LS as per the pace of Instructor/Students' learning

Date / Week#	Contact Session(CS)	Topics# for CS (by Instructor)	Pre-CS Preparation (by Students)
Week 1	1	CS1	M1
Week 3	2	CS2	M2
Week 5	3	CS3	M3
Week 7	4	CS4	M4
Week 9	5	CS5	M5
Week 11	6	**CS6**	**M6**
<<Mid-Sem Examination / Syllabus: M1 → M6>>			
Week 13	7	CS7	M7
Week 15	8	CS8	M8
Week 17	9	CS9	M9
Week 19	10	CS10	M10
Week 21	11	CS11	M11
Week 23	12	CS12	M12
<<End-Sem Examination / Syllabus All Modules (M1→M12)>>			

#: In addition to the listed topics under CSx, refer the below list of topics for experiential learning, they can be given as assignments to students or taken up for discussion in contact sessions

Experiential Learning Components:

The ELCs can be used by the Instructor for discussion in the Contact Sessions or used by Students as examples of self-study or given as take-home assignments by Instructors. Typical examples of ELCs can be:

- Case study of actual Project Implementation using any one of the Agile Methods (SCRUM, Pair-Programming, TDD, etc.)
- Let Students share their own experiences in adopting Agile Methods (such as Pair-Programming) and how QA is ensured without compromising on schedule or cost
- The Instructor to illustrate the role of user-stories in managing Requirements for today's application development in mobile/cloud infrastructure environment.

Assignments

- Each student is given an individual assignment on any of the topics discussed in the class; Assignment Topics are based on practical problems experienced or part of work-items or tools used by collaborating organizations
- Assignments are take-home and deadline-driven (typically of 2-4 weeks duration) announced post Mid-semester examination
- Students to spend at least 16 hours of work in study, research, building prototypes, discussion and preparation of the model/report and presentation.
- As part of deliverables for evaluation, the student is expected to demonstrate or prepare a report and make a short-presentation in the class

Evaluation Scheme

EC #	Name	Type	Weight	Duration	Schedule
EC-1A	Quiz-1	Online	5%	1 Week	August 16-30, 2022
EC-1B	Quiz-2	Online	5%	1 Week	September 16-30, 2022
EC–1C	Assignment	Take-home	15%	2-4 Weeks	October 16-30, 2022
EC-2	Mid-Sem Exam.	Open Book	30%	2 Hrs.	Saturday, 24/09/2022 (AN)
EC-3	End-Sem Exam.	Open Book	45%	2 Hrs.	Saturday, 26/11/2022 (AN)

EC: Evaluation Component   <TBA>: To Be Announced in the class/online

BITS-Pilani eLearn Site: All the recorded Lecture Sessions (LS) are accessible via BITS-Pilani eLearn.site (http://elearn.bits-pilani.ac.in/user/) for all registered students; All materials and communication regarding the course (announcements, assignment submission, online quizzes, and Instructor materials, Session Presentations, etc.) are provided on eLearn site only.

Communication Policy: Students are expected to use Q/A and Discussion Forums in the above eLearn site for all student-instructor communication; No personal emails or mobile calls to instructors will be entertained; Issues related to site access and other administrative issues to be directed to the specified WILP contacts on BITS-Pilani website

Closed Book: No reference material of any kind will be permitted inside the exam hall.

Open Book: Use of any printed Text/Ref Books and hand-written material (notebooks) will be permitted inside the exam hall. Loose sheets, Photocopies and Laser printouts of any material will not be permitted. Computers of any kind will not be allowed inside the exam hall. Use of calculators will be allowed in all exams. No exchange of any material will be allowed.

Self-Study: It shall be the responsibility of the individual student to be regular in maintaining the self-study schedule (watching of Recorded Lectures before the scheduled contact sessions) as given in the course handout.

Instructor