



# **Agile for Software Development**

**SKILLSOFT ASPIRE JOURNEY**

**skillsoft** ▶▶

Głównym wyzwaniem przed którym stają dziś organizacje na całym świecie jest konieczność ciągłego podnoszenia umiejętności i poziomu wiedzy w ślad za gwałtownym rozwojem nowych technologii i zmian na globalnym rynku.

Stały rozwój i podnoszenie kwalifikacji w IT od dawna jest już rzeczą oczywistą, a możliwość zapewnienia wsparcia specjalistom chcącym stale się rozwijać jest jedną z głównych kart przetargowych w walce o pracownika.

Na rynku liczą się dziś ludzie, którzy posiadają konkretne kompetencje i zestaw umiejętności pozwalający im wykonywać zadania efektywnie, a nie Ci z najdłuższym stażem pracy.

Dziś, bardziej niż kiedykolwiek w cenie jest umiejętność budowania ścieżki kariery dla profesjonalistów IT, którzy wciąż chcą się liczyć na rynku pracy.

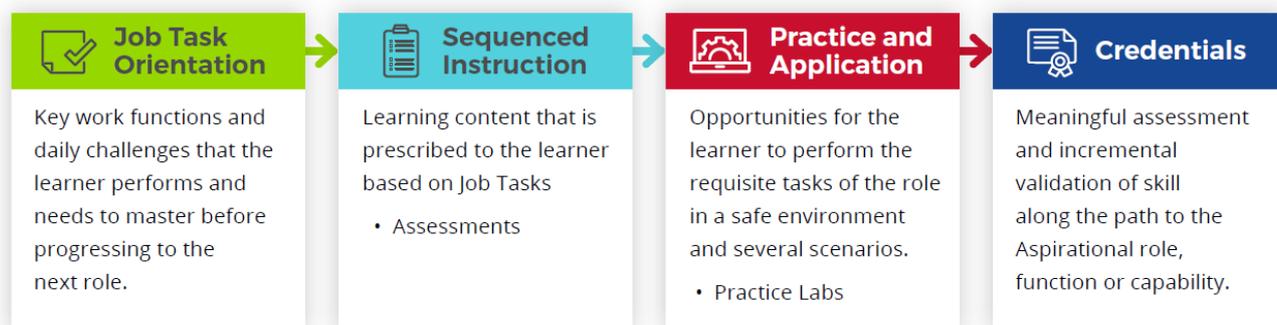
**Skillsoft Aspire Journey** stanowi odpowiedź na pytanie, jakie szkolenia muszą ukończyć, aby być przygotowanym do swojej wymarzonej pracy. Spośród kilkuset kanałów tematycznych dostępnych na naszej platformie szkoleniowej nasi specjaliści wybrali te, które naszym zdaniem najlepiej wyposażą uczących się w narzędzia potrzebne do realizacji zadań w nowej roli.

Skillsoft Aspire Journey to zestawy szkoleń i ćwiczeń w języku angielskim, które metodycznie, krok po kroku pozwalają specjalistom przejść od poziomu podstawowego do zaawansowanego.

Każda ścieżka zawiera szkolenia, laboratoria wirtualne, video i książki, które pomogą uczącym się osiągnąć pożądane kompetencje poświadczone certyfikatem.

## Aspire Journey Model

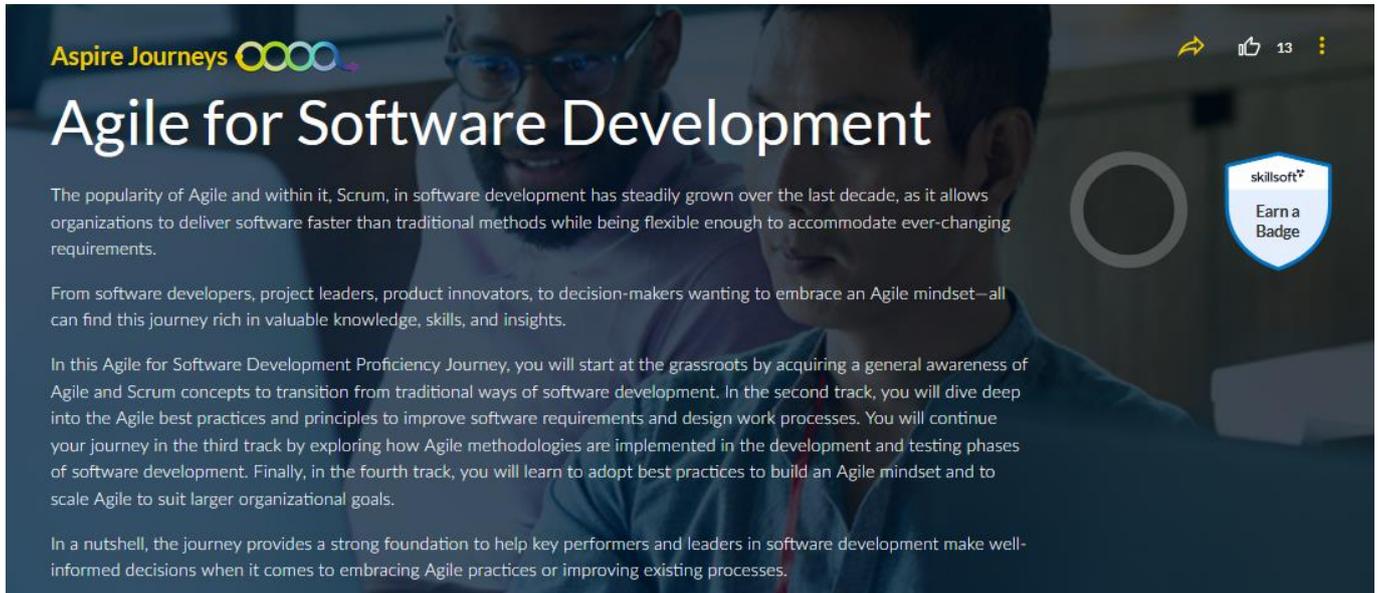
Cała ścieżka opiera się na 4-elementowym cyklu powtarzanym na kolejnych etapach nauki.



1. Określenie kluczowych funkcji i wyzwań, z którymi musi poradzić sobie uczący się w chwili obecnej, jak i tymi, z którymi przyjdzie mu się zmierzyć w nowej pracy.
2. Przejście zaprojektowanych ścieżek w proponowanej kolejności, wykonanie ćwiczeń i zaliczenie testów.
3. Przećwiczenie nowych umiejętności w kontrolowanym środowisku w oparciu o gotowe scenariusze działań. Laboratoria wirtualne Skillsoft
4. Certyfikat – zaliczenie testu końcowego na poziomie co najmniej 70% i uzyskanie certyfikatu potwierdzającego ukończenie danego etapu nauki.

## Aspire Journey – Business Analyst to Data Analyst

Analizując trendy opisujące zachowanie użytkowników na naszych platformach szkoleniowych i współpracując ściśle z naszymi klientami na całym świecie Skillssoft wyselekcjonował najlepsze materiały szkoleniowe i ułożył je w ustrukturalizowaną ścieżkę rozwoju. Ścieżka zawiera około 19 godzin szkoleniowych.



**Aspire Journeys** 

# Agile for Software Development

The popularity of Agile and within it, Scrum, in software development has steadily grown over the last decade, as it allows organizations to deliver software faster than traditional methods while being flexible enough to accommodate ever-changing requirements.

From software developers, project leaders, product innovators, to decision-makers wanting to embrace an Agile mindset—all can find this journey rich in valuable knowledge, skills, and insights.

In this Agile for Software Development Proficiency Journey, you will start at the grassroots by acquiring a general awareness of Agile and Scrum concepts to transition from traditional ways of software development. In the second track, you will dive deep into the Agile best practices and principles to improve software requirements and design work processes. You will continue your journey in the third track by exploring how Agile methodologies are implemented in the development and testing phases of software development. Finally, in the fourth track, you will learn to adopt best practices to build an Agile mindset and to scale Agile to suit larger organizational goals.

In a nutshell, the journey provides a strong foundation to help key performers and leaders in software development make well-informed decisions when it comes to embracing Agile practices or improving existing processes.

 Earn a Badge



### Track 1: Agile Foundations

3 courses | 2h 55m 43s



### Track 2: Software Requirements & Design with Agile

4 courses | 4h 5m 45s



### Track 3: Software Development & Testing with Agile

6 courses | 7h 17m 58s



### Track 4: Agile Organization

5 courses | 4h 50m 52s

## PREREQUISITES

In order to fully profit from the potential of this Aspire Journey, we recommend the following prerequisite skills:

- Knowledge of Agile
- Knowledge of software development

# Track 1: Agile Foundations

In this track of the Agile for Software Development Aspire journey, the focus will be on Agile foundations. You will explore Agile fundamentals as well as Scrum Methodology.

3 courses | 2h 55m 43s



Agile Foundations:  
Fundamentals

## Objectives:

- identify deliverables in the software development life cycle (SDLC) and recognize frequently used concepts and important terms in software development
- recognize roles and responsibilities on typical software development teams
- identify key concepts related to the software testing process in software development
- identify the similarities in several traditional software development methodologies, including Waterfall
- identify the shortcomings of the Waterfall software development methodology
- describe the Agile software development methodology and recognize its benefits
- identify the steps in the Agile software development life cycle
- identify and describe Agile software development artifacts
- recognize how the Agile methodology differs from traditional waterfall project management
- recognize the practice of continuous integration and continuous delivery (CI/CD) in software development and the Agile process
- recognize best practices when transitioning from traditional software development to Agile



Agile Foundations:  
Scrum Methodology

## Objectives:

- identify the features of Scrum and recognize Scrum as the commonly adapted Agile approach
- compare the Agile and Scrum frameworks and describe how they relate to each other
- illustrate using an example how Scrum is best suited for software development and recognize the benefits of the Scrum approach to Agile development
- describe the Scrum backlog refinement meeting and recognize its importance to the sprint and its relationship to the product backlog
- describe the Scrum planning meeting and recognize its importance to the sprint and its relationship to the sprint and product backlogs
- describe the Scrum sprint review meeting and recognize its importance to the sprint and its relationship to the sprint and product backlogs
- describe the Scrum sprint retrospective meeting and recognize its importance to the sprint and to the team
- describe the product owner role and its responsibilities in a self-organizing Scrum team
- describe the scrum mater role and its responsibilities in a self-organizing Scrum team
- describe the developer role and its responsibilities in a self-organizing Scrum team



## Final Exam: Agile Foundations

### Objectives:

- compare the Agile and Scrum frameworks and describe how they relate to each other
- describe the Agile software development methodology and recognize its benefits
- describe the Developer role and its responsibilities in a self-organizing Scrum team
- describe the Product Owner role and its responsibilities in a self-organizing Scrum team
- describe the Scrum backlog refinement meeting and recognize its importance to the sprint and its relationship to the product backlog
- describe the Scrum Master role and its responsibilities in a self-organizing Scrum team
- describe the Scrum planning meeting and recognize its importance to the sprint and its relationship to the sprint and product backlogs
- describe the Scrum sprint retrospective meeting and recognize its importance to the sprint and the team
- describe the Scrum sprint review meeting and recognize its importance to the sprint and its relationship to the sprint and product backlogs
- identify and describe Agile software development artifacts
- identify deliverables in the software development life cycle and recognize frequently used concepts and important terms in software development
- identify examples of situation that pose challenges to using Scrum
- identify key concepts related to the software testing process in software development
- identify the features of Scrum and recognize Scrum as the commonly adapted Agile approach
- identify the shortcomings of the Waterfall software development methodology
- identify the similarities in several traditional software development methodologies, including Waterfall
- identify the steps in the Agile life cycle
- recognize how the Agile methodology differs from traditional waterfall project management
- recognize roles and responsibilities on typical software development teams
- recognize the practice of continuous integration and continuous delivery (CI/CD) in software development and the Agile process

# Track 2: Software Requirements & Design with Agile

In this track of the Agile for Software Development Aspire journey, the focus will be on Agile Software Projects. Explore software requirements, software design and estimation methods.

4 courses | 4h 5m 45s

skillsoft<sup>2</sup>  
Earn a  
Badge



Bill Brooks

Senior Software Developer

## Agile Software Projects: Software Requirements

### Objectives:

- recognize what software requirements are and their importance in successful software project management
- describe what functional requirements are and how they are used in a software project
- describe what non-functional requirements are and how they are used in a software project
- recognize how functional requirements differ from non-functional requirements
- describe the importance of requirements management and list the four fundamental requirements management processes
- describe in detail the requirements management process
- illustrate through a non-example the shortcomings of traditional requirement specification and management process (ever-changing scope, wasted efforts, demotivated team, etc.)
- compare the benefits and features of traditional project management against Agile project management
- identify how the Agile approach helps define and manage software requirements
- describe the importance of Agile software requirements
- describe a typical software requirements specification document
- describe how the feasibility study tool helps determine whether a solution is practically achievable or not



Bill Brooks

Senior Software Developer

## Agile Software Projects: Software Design

### Objectives:

- describe using examples the best practices of using modularity in Agile software design
- describe using examples the best practices of using coupling concepts in Agile software design
- describe using examples the best practices of using cohesion concepts in Agile software design
- recognize how to build product vision from a requirement or objective of the software product
- illustrate through an example the effectiveness of using product vision board to define the product features
- recognize how Agile tools such as initiatives, themes, epics, and stories help define and manage scope better than traditional methods for scoping
- recognize how to generate user stories from a requirement or objective of the software product
- describe the minimal viable product
- define what the Definition of Done means in the Scrum framework using examples
- describe what UX design is and the principles of UX design
- recognize how to overcome the challenges of integrating Agile with UX design
- describe the best practices for Agile software requirement and design



Bill Brooks  
Senior Software Developer

### Agile Software Projects: Estimation Methods

#### Objectives:

- recognize the challenges faced in transforming to an Agile organization
- describe best practices to be adopted for building an Agile culture in the organization
- recognize how estimation in Agile involves every member of the team
- describe how to perform Agile estimation and some of the popular Agile estimation methods, including Story Points, Planning Poker, T-shirt Sizes, Dot Voting, and Bucket System
- describe how to use the Story Points estimation method in Agile
- describe how to use the Planning Poker estimation method in Agile
- describe how to use the Dot Voting estimation method in Agile
- describe how to use the Bucket System estimation method in Agile
- recognize the benefits offered by Agile estimation models over traditional estimation methods
- describe the key questions to consider when estimating
- describe some of the common pitfalls that occur using Agile estimation techniques
- illustrate through an example how to arrive at an estimate for a story point using Planning Poker



### Final Exam: Software Project Requirements & Design with Agile

#### Objectives:

- define what the Definition of Done means in the Scrum framework using examples
- Describe how to use the bucket system estimation method in Agile
- Describe how to use the dot voting estimation method in Agile
- Describe how to use the planning poker estimation method in Agile
- Describe how to use the story points estimation method in Agile
- describe in detail the requirements management process
- describe the best practices to be adopted for building an Agile culture in the organization
- describe the importance of Agile software requirements
- describe the Minimal Viable Product
- describe using examples the best practices of using coupling concepts in Agile software design
- describe using examples the best practices of using modularity in Agile software design
- describe what functional requirements are and how they are used in a software project
- describe what non-functional requirements are and how they are used in a software project
- discover the benefits offered by Agile estimation models over traditional estimation methods
- recognize how Agile tools such as Initiatives, themes, epics, and stories help define and manage scope better than traditional methods for scoping
- Recognize how functional requirements differ from non-functional requirements
- recognize how to build product vision from a requirement or objective of the software product
- recognize how to generate user stories from a requirement or objective of the software product
- recognize the challenges faced in transforming to an Agile organization
- Recognize what software requirements are and their importance in successful software project management

Aspire Journeys: Agile for Software Development

# Track 3: Software Development & Testing with Agile

In this track of the Agile for Software Development Aspire journey, the focus will be on development & testing with Agile. You will explore Agile methodologies, extreme programming, Agile teams, and project management.

6 courses | 7h 17m 58s



Development & Testing with Agile: Agile Methodologies in Software Development

## Objectives:

- recognize Agile methodologies and provide a brief overview of Agile methodologies in software development
- define the Crystal Agile framework and its features as a main Agile methodology
- describe the concept of feature-driven development as an Agile software development methodology
- define the use of feature-driven development and its components by applying it to Agile software development
- define Extreme Programming (XP) and its features
- describe the principles and steps of the Lean development methodology
- describe the use of Kanban, including the Just-In-Time (JIT) approach and its benefits
- compare and contrast the various Agile methodologies
- describe use case examples using Crystal and feature-driven development



Development & Testing with Agile: Extreme Programming

## Objectives:

- describe Extreme Programming (XP) concepts, including its roles, values, and benefits
- describe extreme programming using a software development use case
- recognize the planning game practice and its purpose
- recognize the pair programming practice and its purpose
- describe the effectiveness of pair programming as an Agile tool to develop software
- recognize the test-driven development practice and its benefits
- define the test-driven development cycle from beginning to end
- describe the customer's role in XP
- provide an overview of source control management and tools
- describe source control tools and how they can be used in an Agile project
- recognize the practice of continuous integration
- demonstrate how continuous integration tools help Agile software development
- describe the coding standard practice and its benefits to Agile software development
- describe the concept of collective code ownership and its benefits
- identify the practice of code refactoring
- describe how code refactoring helps Agile software development based on examples
- recognize the small releases practice and its use for iterative releases in Agile software development
- recognize the system metaphor practice and its elements
- describe the concepts behind the 40-hour week



Development &  
Testing with Agile:  
Agile Self-organizing  
Teams

Objectives:

- discover the key concepts covered in this course
- define self-organizing teams and how they work in Agile software development environment
- describe tips and effective practices for developing an Agile collaborative environment
- define concepts used working with teams
- illustrate how to overcome challenges of working with distributed Agile teams through an example
- describe Agile team metrics and how to effectively use them for self-improvement
- define Agile team metrics in the concept of a software development environment
- identify best practices and guidelines for removing impediments to a successful sprint
- identify skills to becoming an effective product owner
- identify skills to becoming an effective Scrum master
- recognize how to build a strong Agile team and effective techniques for maintaining it



Development &  
Testing with Agile:  
Team Meetings

Objectives:

- describe what Agile team meetings are and the various types
- describe a sprint planning meeting, including goals and objectives
- recognize the benefits and best practices for an effective sprint planning meeting
- describe goals and objectives for an effective stand-up meeting
- recognize through a work example, best practices for daily stand-up meetings
- describe the goals and objectives of the sprint review meeting
- recognize best practices for sprint review meetings through the use of an example
- recognize how to document the minutes of a meeting through the use of an example
- demonstrate how to run a great virtual meeting



Development &  
Testing with Agile:  
Project Management

Objectives:

- describe how Jira can be used for effective Agile project management
- demonstrate how Scrum boards in Jira can be used to track an Agile software project
- recognize Agile project communication methods and challenges, and compare them with traditional communication models in software development
- describe useful Agile project management data analysis metrics and how they can be effectively used
- recognize the use of data analysis for Agile PM
- describe the use of continuous development and delivery for Agile in a software development environment
- demonstrate the use of key value metrics using examples
- recognize best practices for Agile software development and testing
- summarize the key concepts covered in this course



Final Exam:  
Software  
Development and  
Testing with Agile

Objectives:

- define Extreme Programming or (XP) and its features
- define self-organizing teams and how they work in an Agile software development environment
- define the Crystal Agile framework and its features as a main Agile methodology
- describe goals and objectives for an effective stand-up meeting
- describe the coding standard practice and its benefits to Agile software development
- describe the concept of feature-driven development as an Agile software development methodology
- describe the concepts of Extreme Programming, including its roles, values, and benefits
- describe the roles of Extreme Programming
- describe the use of Agile metrics to manage an effective team
- describe the use of continuous development and delivery for Agile in a software development environment
- describe tips and effective practices for developing an Agile, collaborative environment
- describe useful Agile project management data analysis metrics and how they can be effectively used
- describe what Agile team meetings and the various types are
- identify skills to becoming an effective product owner
- identify small releases practice and its use for iterative releases in Agile software development
- identify the pair programming practice and its purpose
- identify the test-driven development practice and its benefits
- recognize Agile methodologies providing a brief overview of Agile methodologies in software development
- recognize the best practices for Agile software development and testing
- recognize using an example how to document the minutes of a meeting

# Track 4: Agile Organization

In this track of the Agile for Software Development Aspire journey, the focus will be on Agile organizations. Explore Cloud for Agile software development, DevOps for Agile, Agile models, and building an Agile culture.

5 courses | 4h 50m 52s

skillssoft<sup>2</sup>  
Earn a  
Badge



**Sven Batalla**  
Product Owner/Scrum Master  
Agile Organizations:  
Cloud for Agile  
Software Development

## Objectives:

- discover the key concepts covered in this course
- describe the benefits of using cloud computing for Agile
- recognize considerations related to cloud management
- list IAM solutions such as federated identity, single in, and multi-factor authentication
- describe the components of cloud infrastructure
- recognize different privacy concerns such as private data and jurisdictional concerns
- recognize the challenges introduced by different types of cloud platforms such as public, private, hybrid, and community
- list cloud application architecture specifics such as supplementary security components, cryptography, and sandboxing
- describe the various cloud deployment models such as public, private, hybrid, community, and virtual private clouds, as well as multi-cloud and multitenancy environments
- describe the key components and the pros and cons of using the Infrastructure as a Service (IaaS) service model
- describe the key components and the pros and cons of using the Platform as a Service (PaaS) service model
- describe the key components and the pros and cons of using the Software as a Service (SaaS) service model
- describe the Internet of Things (IoT) and how it applies to cloud computing



**Colin Calnan**  
Senior Web Developer  
Agile Organizations:  
DevOps & Agile

## Objectives:

- discover the key concepts covered in this course
- describe the features of DevOps and its benefits
- describe the benefits of using DevOps and Agile together
- describe the components and features of the Agile development process
- identify the steps in the Agile life cycle
- identify the significance of continuous integration using the Agile process
- describe how the cloud can be used with DevOps
- compare continuous integration, continuous delivery and continuous deployment and how they are related to each other
- describe the purpose of continuous integration and why it is important for software development
- describe the benefits of using continuous integration
- describe scalability and methods for performing scaling using auto-scaling, horizontal scaling, vertical scaling, and cloud bursting
- describe Agile, DevOps, and cloud interaction
- describe the Agile cloud architecture environment



Agile Organizations:  
Agile Models & Scaling  
Agile

Objectives:

- compare and contrast the different Agile frameworks available for Agile transformation of the organization
- describe the features of the Hybrid Agile model
- describe the features of the Bimodal Agile model
- recognize the seven wastes to eliminate through the Lean model of Agile
- describe guidelines to identify the best fit Agile framework for the organization
- recognize the benefits offered by Agile estimation models over traditional estimation methods
- describe the methods to use to increase the benefits of Agile from a project level to an organizational level
- describe the various scaling models available for integrating Agile
- recognize the SAFe framework available for scaling Agile
- describe the characteristics, benefits, and challenges of using the LeSS framework
- identify the challenges in scaling Agile
- describe the Scrum of Scrums and the roles and responsibilities of the team members, as well as how to conduct this meeting



Agile Organizations:  
Building an Agile  
Culture

Objectives:

- recognize the challenges faced in transforming to an Agile organization
- recognize how to establish a governance framework for adopting Agile at an organizational level
- describe the best practices to be adopted for building an Agile culture in the organization
- describe Agile management best practices and how to create effective teams
- recognize the best practices to be adopted by an Agile leader to build and mentor Agile teams for a software project
- recognize the best practices to adopt for overcoming the challenges of working with distributed teams in an Agile environment
- describe the best practices for managing and working with distributed teams in Agile
- identify how to effectively transition a distributed team to Scrum
- describe Agile team dynamics and how to ensure teams are effective and productive
- describe conflict resolution techniques for the Agile coach
- identify the parameters with which to assess an organization in order to determine its Agile maturity
- describe how Agile senior management can, through effective Agile leaders and teams, successfully maximize the benefits of Agile practices for software development



## Final Exam: Agile Organization

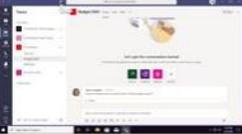
### Objectives:

- compare and contrast the different Agile frameworks available for Agile transformation of the organization
- describe agile management best practices and how to create effective teams
- describe conflict resolution techniques for the agile coach
- describe the benefits of using cloud computing for Agile
- describe the benefits of using continuous integration
- describe the best practices to be adopted for building an Agile culture in the organization
- describe the components and features of the agile development process
- describe the features of DevOps and its benefits
- describe the features of the Bimodal Agile model
- describe the features of the Hybrid Agile model
- describe the various cloud deployment models such as public clouds, private clouds, hybrid clouds, community clouds, virtual private clouds, multi-cloud environments, and multitenancy environments
- discover the best practices to be adopted by an Agile Coach
- discuss the Scrum of Scrums and the roles and responsibilities of the team members, as well as how to conduct this meeting
- identify the challenges in scaling Agile
- identify the disadvantages of vertical scaling
- identify the steps in the Agile life cycle
- recognize considerations related to cloud management
- recognize different privacy concerns such as private data and jurisdictional concerns
- recognize the challenges faced in transforming to an Agile organization
- understand challenges introduced by different types of a cloud platform such as public, private, hybrid, community

## Business & Leadership with Agile for Software Development Optional

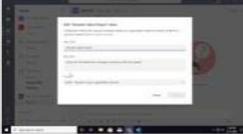
 <p>COURSE</p> <p><b>Innovating with Lean Product Management</b></p> <p>216</p>	 <p>COURSE</p> <p><b>Cultivating Cross-functional Team Collaboration</b></p> <p>180</p>	 <p>COURSE</p> <p><b>The Essential Role of the Agile Product Owner</b></p> <p>247</p>	 <p>COURSE</p> <p><b>Navigating through Changes and Conflicts in Projects</b></p> <p>293</p>	 <p>COURSE</p> <p><b>Effective Team Communication</b></p> <p>1018</p>
 <p>COURSE</p> <p><b>Developing a Growth Mindset</b></p> <p>2124</p>	 <p>COURSE</p> <p><b>Knowing When to Take Strategic Risks</b></p> <p>461</p>	 <p>COURSE</p> <p><b>Building the Foundation for an Effective Team</b></p> <p>516</p>	 <p>COURSE</p> <p><b>Managing for Operational Excellence</b></p> <p>378</p>	

# Productivity Tools with Agile for Software Development Optional



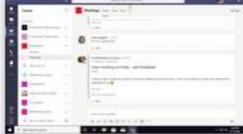
COURSE  
Getting to know the application

978



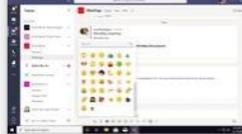
COURSE  
Using Teams & Channels

731



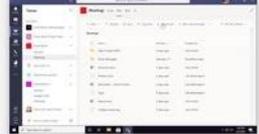
COURSE  
Communicating via the App

704



COURSE  
Formatting, Illustrating & Reacting to Messages

531



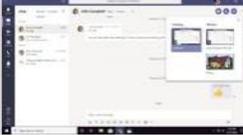
COURSE  
Creating, Finding & Organizing Files

520



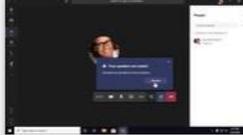
COURSE  
Working with Apps, Tabs & Wiki

458



COURSE  
Making calls, Organizing Contacts & Using Voicemail

446



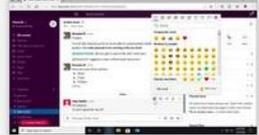
COURSE  
Creating, Joining & Managing Meetings

455



COURSE  
Signing in & Setting Up Slack

22



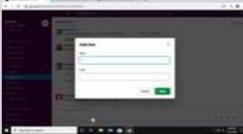
COURSE  
Using Channels in Slack

10



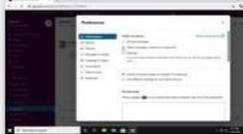
COURSE  
Using Private Messaging & Communication Tools in...

12



COURSE  
Creating, Finding & Sharing Information in Slack

6



COURSE  
Configuring Slack

4



COURSE  
Creating & Setting Up Projects in Jira Cloud

141



COURSE  
Configuring & Managing Boards in Jira Cloud

97



COURSE  
Planning & Working on a Software Project in Jira...

72



COURSE  
Reporting in Jira Software

71



COURSE  
Signing in & Navigating within Spaces

37



COURSE  
Setting Up & Managing Spaces

30



COURSE  
Working with Space

26



COURSE  
Working with Team Members

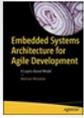
69



COURSE  
Configuring Spaces

18

## Bookshelf (i) Optional

 <p>BOOK</p> <p>Emerging Innovations in Agile Software Development</p> <p>14</p>	 <p>BOOK</p> <p>Agile Software Architecture: Aligning Agile Processes an...</p> <p>54</p>	 <p>BOOK</p> <p>Agile Software Development: Best Practices for Large...</p> <p>2</p>	 <p>BOOK</p> <p>Embedded Systems Architecture for Agile...</p> <p>33</p>	 <p>BOOK</p> <p>Agile Development in the Real World</p> <p>29</p>
 <p>BOOK</p> <p>Agile Foundation: Principles, Practices and Frameworks</p> <p>3</p>	 <p>BOOK</p> <p>Software Project Estimation: Intelligent Forecasting,...</p> <p>45</p>	 <p>BOOK</p> <p>Agile Testing Foundations: An ISTQB Foundation Level...</p> <p>158</p>	 <p>BOOK</p> <p>Agile Testing: How to Succeed in an Extreme...</p> <p>6</p>	 <p>BOOK</p> <p>An Executive's Guide to Software Quality in an Agile Organization</p> <p>2</p>
 <p>BOOK</p> <p>The Agile Organization: How to Build an Engaged,...</p> <p>6</p>	 <p>BOOK</p> <p>Performance Management for Agile Organizations,...</p> <p>2</p>			

**FOLLOW US ON:**



[www.skilltech.pl](http://www.skilltech.pl)

email: [biuro@skilltech.pl](mailto:biuro@skilltech.pl)

tel. +48 22 44 88 827

**SkillTech**  
Technology hired for excellence