



NANODEGREE PROGRAM SYLLABUS

Agile Software Development



Overview

Learn how to build products that deliver continuous value to customers using an Agile approach to software development. By the end of this program, students will be able to differentiate between Scrum, Kanban and XP, create an environment that fosters high performing teams, and manage iteration planning using Agile techniques. Students will also be able to create a release plan for a Minimum Viable Product, create metrics that show project status, and effectively communicate progress both within and outside of the development team.

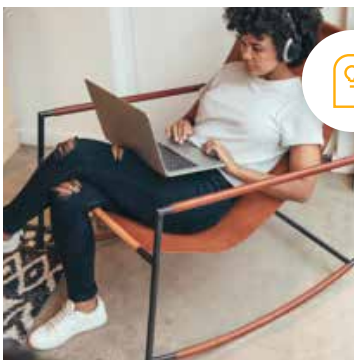
Prerequisites: No prior experience is required, but it is recommended that students are comfortable with basic computer skills, such as managing files, using third-party online programs, and navigating the Internet through an online browser.



Estimated Time:
3 months at
10hrs/week



Instructional Tools Available:
Video lectures,
mentor-led student
community, forums,
project reviews



Flexible Learning:
Self-paced



Need Help?
udacity.com/advisor

*The length of this program is an estimation of total hours the average student may take to complete all required coursework, including lecture and project time. If you spend about 10 hours per week working through the program, you should finish within the time provided. Actual hours may vary.

Course 1: Foundations of Agile and Agile Frameworks

In this course, students will be introduced to The Agile Mindset and how it sets the tone for “Being” Agile versus just “Doing” Agile. Students will learn to leverage The Agile Manifesto as the foundation for all Agile Frameworks, as well as identify the practical differences between Agile and Waterfall approaches. Students will then take a deep dive into Agile Teams and Governance and apply best practices of both in order to deliver immense business value. By the end of this course, students will master characteristics of three of the more popular Agile Frameworks being utilized across all industries, which are Scrum, Kanban and XP, as well as apply The Agile Manifesto to deliver practical value in Agile teams and organizations.

Course 1 Project

WorldVisitz Mobile App
Agile Delivery Launch

In this project, students will step in as an Agile consultant to help launch WorldVisitz’s Agile journey. Students will recommend and define an Agile delivery solution for WorldVisitz executives to replace their current inefficient traditional product development processes. Based on an assessment of the organization, students will prepare a presentation to convince WorldVisitz leaders of the business value and rationale for adopting an Agile framework. Students will also prepare an Agile onboarding presentation to get the team started on their Agile journey.

LEARNING OUTCOMES

LESSON ONE

Why Agile?

- Explain the Agile Mindset and how it sets the tone for ‘Being’ Agile versus just ‘Doing’ Agile
- Identify how the Agile Manifesto sets the foundation for all Agile Frameworks
- Compare Agile versus the more traditional Waterfall approach to product development
- Evaluate common misconceptions about Agile

LESSON TWO

Build and Evolve Agile Teams

- Recognize the characteristics of a high performing Agile Team
- Sustain and enable high performing teams
- Identify an Agile team’s core roles, optimal size, structure, and cross-functional skills
- Apply best practices of Agile Governance

LESSON THREE

Agile Frameworks

- Compare and contrast Scrum, Kanban, and XP
- Evaluate the unique characteristics of the Scrum framework and appropriate uses
- Evaluate the advantages of the Kanban Framework and its appropriate uses
- Evaluate the advantages of the XP Framework and its appropriate uses



Course 2: Delivering Value with Agile Planning and Prioritization

The Agile approach to planning is an iterative process that focuses on delivering value to the customer. In this course, students will be introduced to the high level aspects of agile planning, including product vision and roadmaps. Students will also learn how to manage requirements at the tactical level, including gathering, writing, and prioritizing requirements using agile techniques from each of the major frameworks. Finally, students will learn different approaches to estimating work and how to build release and iteration plans that help the team continuously deliver value. At the end of the course, students will be equipped with the tools and techniques they need to apply Agile Planning and Prioritization within a team or organization to deliver value more efficiently.

Course 2 Project

Create an MVP Release Plan

In this project, students will create a plan to develop a software product for the Centers for Disease Control that will help prevent the spread of a deadly virus. Students will be given a specific set of criteria to create a plan for the customer. The plan will include a vision, roadmap, user stories, and a release plan that outlines the minimum set of features that are needed to make the product successful and align with business requirements for an MVP.

LEARNING OUTCOMES

LESSON ONE

Agile Planning

- Describe the benefits of agile planning
- Explain the MVP concept and understand how it's used in software development
- Construct a product roadmap
- Create user stories that describe product requirements
- Understand the difference between features, epics, and user stories
- Identify acceptance criteria for user stories

LESSON TWO

Prioritization

- Define the product backlog and explain why it exists
- Explain how each team role uses and interacts with the backlog
- Manage a backlog and organize it using progressive elaboration
- Identify and apply different prioritization techniques
- Prioritize a product backlog

LESSON THREE

Scoping

- Understand how to control the scope of user stories
- Refine the Definition of Done for user stories
- Estimate user stories
- Explain and apply various estimation techniques
- Apply ideal time to estimates

LESSON FOUR

Release and Iteration Planning

- Explain the relationship between release and iteration planning
- Identify the outcomes of release and iteration planning
- Plan an MVP that delivers value incrementally by using techniques such as continuous integration and continuous delivery
- Explain and apply timeboxes
- Apply the Scrum framework to plan a release and a sprint



Course 3: Progress, Communication, and Organizational Agility

In this course, students will learn how to communicate project progress and status through information radiators, guide and direct organizational agility through the use of metrics and how to create those metrics, and differentiate between patterns to mimic and anti-patterns to avoid. Students will also learn how to identify and mitigate risk, and build an internal team communication strategy and an external communication strategy that work effectively for the relevant stakeholder audience.

Course 3 Project

Report Agile Project Status with a Big Visual Information Radiator

In this project, students will learn how to plan, organize, monitor, and display project information at a glance. Students will create a Big Visual Information Radiator (BVIR) that can be used by anyone involved in an Agile project to review project goals, work completion status, metrics, risks and target completion date. Students will learn how to use techniques such as appropriate use of burn-up and burn-down charts, prioritization of user stories, identifying risks, and calculating velocity. Students will also learn how to effectively communicate project status and appropriate key metrics to a senior management audience within a corporation.

LEARNING OUTCOMES

LESSON ONE

Agile Metrics

- Explain the importance of using metrics in Agile
- Differentiate between outputs and outcomes
- Calculate a Velocity
- Determine the Lead Time and Cycle Time
- Monitor the status of Work in Progress (WIP)
- Estimate when work should be completed
- Identify escaped defects and how to handle them appropriately

LESSON TWO

Measuring Progress and Impact

- Identify the different parts of the continuous improvement process that lead to a sustainable development level
- Differentiate between patterns and antipatterns of the continuous improvement process
- Effectively apply Agile thinking to reach sustainable development
- Run a retrospective to effectively identify what went well, what didn't go well, and what can be improved
- Effectively prioritize the next steps to improve on the lessons learned and add it to the backlog
- Use the appropriate chart type to build an IR to communicate a specific metric
- Create a BVIR to effectively communicate project status

LESSON THREE

Identifying Risks

- Determine the impact risk would have on a project
- Effectively communicate how technical debt impacts a project
- Apply Agile techniques to mitigate technical debt effectively
- Identify failure patterns and implement mitigation strategies
- Use testing techniques early to keep an Agile project on schedule

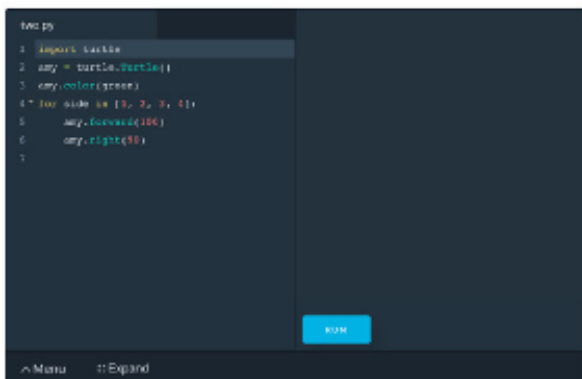
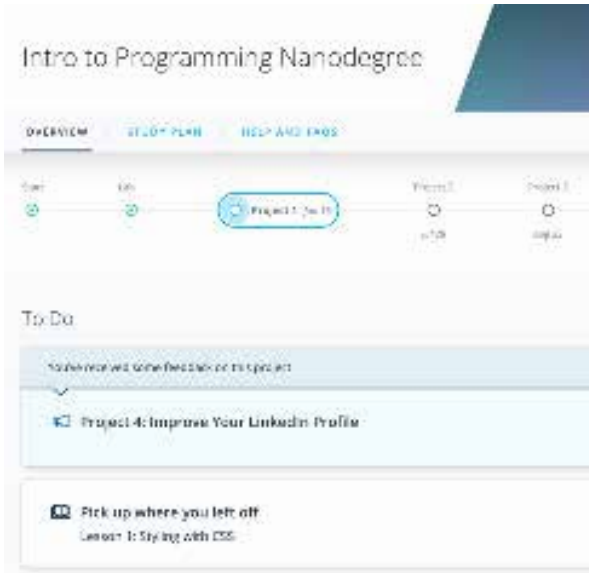
LESSON FOUR

Agile Communication

- Differentiate between metrics that should be shared internally vs. externally
- Create a BVIR to communicate project status to the relevant team/stakeholders effectively
- Correctly determine the status of the current in-progress project
- Effectively implement tools to communicate across remote teams
- Build a communication strategy for an internal team and for external parties
- Build trust using Agile techniques between team members to encourage transparency in communication



Our Classroom Experience



REAL-WORLD PROJECTS

Build your skills through industry-relevant projects. Get personalized feedback from our network of 900+ project reviewers. Our simple interface makes it easy to submit your projects as often as you need and receive unlimited feedback on your work.

KNOWLEDGE

Find answers to your questions with Knowledge, our proprietary wiki. Search questions asked by other students and discover in real-time how to solve the challenges that you encounter.

STUDENT HUB

Leverage the power of community through a simple, yet powerful chat interface built within the classroom. Use Student Hub to connect with your technical mentor and fellow students in your Nanodegree program.

WORKSPACES

See your code in action. Check the output and quality of your code by running them on workspaces that are a part of our classroom.

QUIZZES

Check your understanding of concepts learned in the program by answering simple and auto-graded quizzes. Easily go back to the lessons to brush up on concepts anytime you get an answer wrong.

CUSTOM STUDY PLANS

Work with a mentor to create a custom study plan to suit your personal needs. Use this plan to keep track of your progress toward your goal.

PROGRESS TRACKER

Stay on track to complete your Nanodegree program with useful milestone reminders.

Learn with the Best



Hasnain Rizvi

SENIOR AGILE COACH

Dr. Hasnain Rizvi is an Agile Transformation thought leader and has trained over 25,000 professionals globally. He is a trainer and adjunct professor for Global 2000 clients, universities, and organizations including The University of British Columbia, Southern Alberta Institute of Technology, and Steinbeis School of International Business and Entrepreneurship.



Mark Ginise

AGILE ENGINEER AND COACH

Mark Ginise leads Agility training for the federal government. He has taught Agility to DoD programs, and worked as an internal change agent for Federal Government agencies. His specialties include Agile transformations, DevSecOps, cloud migrations, and technology education. He enjoys the beach, his children, and traveling.



Vincent High

AGILE DELIVERY & TRANSFORMATION LEAD

Vincent is a Scrum Master, Agile Instructor, and currently serves as an Agile Delivery Lead at a top US bank. Throughout his career he has served as a Scrum Master and Agile Coach within start-ups, large corporations, and non-profit organizations. In his spare time he enjoys watching old movies with family.

All Our Nanodegree Programs Include:



EXPERIENCED PROJECT REVIEWERS

REVIEWER SERVICES

- Personalized feedback & line by line code reviews
- 1600+ Reviewers with a 4.85/5 average rating
- 3 hour average project review turnaround time
- Unlimited submissions and feedback loops
- Practical tips and industry best practices
- Additional suggested resources to improve



TECHNICAL MENTOR SUPPORT

MENTORSHIP SERVICES

- Questions answered quickly by our team of technical mentors
- 1000+ Mentors with a 4.7/5 average rating
- Support for all your technical questions



PERSONAL CAREER SERVICES

CAREER SUPPORT

- Resume support
- Github portfolio review
- LinkedIn profile optimization

Frequently Asked Questions

PROGRAM OVERVIEW

WHY SHOULD I ENROLL?

Companies are looking to deliver better products faster and more efficiently, and they're looking to professionals with the Agile skillset to lead that transformation -- according to KPMG, 68% of organizations surveyed indicated "faster product delivery" as their reason for adopting Agile practices. That's why jobs that involve Agile development skills are projected to grow 7.3 percent over the next 10 years, with a median salary of around \$103,000, according to Burning Glass.

In the Agile Software Development Nanodegree program, you'll build the skills you need to guide the delivery of high value products with top notch software development teams. By the end of the program, you'll be able to differentiate between popular Agile frameworks, create an environment that fosters high performing teams, manage iteration planning using Agile techniques, and more. Graduates of the program will be in-demand at a wide range of companies looking to adopt or improve their use of Agile development practices.

WHAT JOBS WILL THIS PROGRAM PREPARE ME FOR?

Students who graduate from the Agile Software Development Nanodegree program can find success across a variety of different roles, including:

- Delivery Lead
- Agile Delivery Lead
- Engineering Manager
- Project Manager or Agile Project Manager
- Scrum master
- Product Owner
- Product Manager
- and many others...

HOW DO I KNOW IF THIS PROGRAM IS RIGHT FOR ME?

Agile team leaders are a critical piece of many high performing development operations - they plan and prioritize work, protect the team from obstacles, and ensure consistent delivery. If you're someone who likes to identify and solve problems so your team can focus on execution, or someone who likes to create and refine the structures and processes that enable a team to succeed, or just someone who is interested in learning how to add more rigor and organization to your project management, then the Agile Software Development Nanodegree program is a great fit.



FAQs Continued

ENROLLMENT AND ADMISSION

DO I NEED TO APPLY? WHAT ARE THE ADMISSION CRITERIA?

There is no application. This Nanodegree program accepts everyone, regardless of experience and specific background.

WHAT ARE THE PREREQUISITES FOR ENROLLMENT?

No prior experience with Agile Software Development is required. You will need to be comfortable with basic computer skills, such as managing files, using third-party online programs, and navigating the Internet through an online browser.

TUITION AND TERM OF PROGRAM

HOW IS THIS NANODEGREE PROGRAM STRUCTURED?

The Agile Software Development Nanodegree program is comprised of content and curriculum to support three projects. Once you subscribe to a Nanodegree program, you will have access to the content and services for the length of time specified by your subscription. We estimate that students can complete the program in three (3) months, working 5-10 hours per week.

Each project will be reviewed by the Udacity reviewer network. Feedback will be provided and if you do not pass the project, you will be asked to resubmit the project until it passes.

HOW LONG IS THIS NANODEGREE PROGRAM?

Access to this Nanodegree program runs for the length of time specified in the payment card above. If you do not graduate within that time period, you will continue learning with month to month payments. See the [Terms of Use](#) and [FAQs](#) for other policies regarding the terms of access to our Nanodegree programs.

CAN I SWITCH MY START DATE? CAN I GET A REFUND?

Please see the Udacity Program [Terms of Use](#) and [FAQs](#) for policies on enrollment in our programs.

WHAT SOFTWARE AND VERSIONS WILL I NEED FOR THIS PROGRAM?

WHAT SOFTWARE AND VERSIONS WILL I NEED IN THIS PROGRAM?

There are no specific hardware or software requirements for this program, other than those outlined on Udacity's general [Technology Requirements page](#).

