



Certificate of Achievement

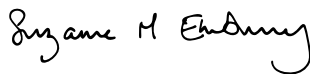
Matthias Giger

has completed the following course:

COLLABORATIVE CODING WITH GIT
THE UNIVERSITY OF MANCHESTER AND INSTITUTE OF CODING

This online course taught Git as a version control system for collaborative software development.

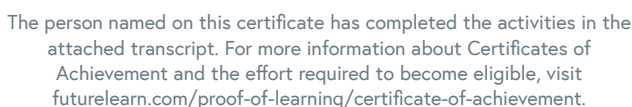
6 weeks, 2 hours per week



Dr Suzanne Embury
Reader in Software Engineering
The University of Manchester



Dr Fardeen Mackenzie
Research Associate
The University of Manchester



The person named on this certificate has completed the activities in the attached transcript. For more information about Certificates of Achievement and the effort required to become eligible, visit futurelearn.com/proof-of-learning/certificate-of-achievement.

This certificate represents proof of learning. It is not a formal qualification, degree, or part of a degree.

Matthias Giger

has completed the following course:

COLLABORATIVE CODING WITH GIT THE UNIVERSITY OF MANCHESTER AND INSTITUTE OF CODING

70%
OVERALL
SCORE

This online course taught the underlying structure of the Git version control system and how it can be used to facilitate collaborative software development. Topics covered included onboarding with Git, creating and maintaining Git repositories, best practice, Git as a QA tool, and modern Git workflows.

- 3. • Working collaboratively with colleagues on projects.
- 4. • Git as a Quality Assurance tool and as a facilitator for team learning.
- 5. • Making Git work for you and your team using Git Workflows.

STUDY REQUIREMENT

6 weeks, 2 hours per week

LEARNING OUTCOMES

- Produce and maintain your own Git repository which can be used privately or shared with an unlimited number of collaborators.
- Assess the current state of a project when you join a new team and quickly orientate yourself with the codebase using Git.
- Contribute to software development projects safely, professionally and following practices which are common within industry.
- Collaborate within teams of various sizes on a single software project using methods such as code review to ensure safe code changes, maintain high code quality, maintain consistent coding styles and facilitate intra-team knowledge transfer.
- Demonstrate your understanding, not just of Git commands, but of the underlying structure which powers Git in order to solve problems when things go wrong.

SYLLABUS

1. • Visualising and exploring a project's history using Git.
2. • Using Git to manage safe and effective project changes.