

# COMP64803: Introduction to Responsible & Reproducible AI

Mingfei Sun

CDT on Decision-Making for Complex Systems  
Department of Computer Science  
The University of Manchester



# CDT Course Overview

Strands	Area	Year 1: Foundation year with the cohort based in Manchester.			Years 2-4: PhD years with the students located in Manchester and/or Cambridge							
		TB1 (Oct-Jan)	TB2 (Feb-May)	TB3 (Jun-Sep)	Year 2	Year 3	Year 4					
Core Foundation Strand	Machine Learning Foundations	Reasoning and Learning under Uncertainty (UoM, 15cr)	Reinforcement Learning (UoM, 15cr)	Mini research project (30cr)	PhD Project at Manchester and/or Cambridge	PhD Project at Manchester and/or Cambridge	PhD Project at Manchester and/or Cambridge					
	Data-Driven Decision Making	Machine Learning and the Physical World (UoC, 15cr)										
Systems and Applications Strand	Domain Units Real-World Complex Systems	Introductory Domain Topics (UoM, 15cr)	Advanced Domain Topics (UoM, 15cr)	CDT conference	Summer or Winter School (one/year)		Journal Club (throughout the years) CDT Conference (one/year)					
	Team Projects	Team Project 1 (5cr)	Team Project 2 (5cr)									
Enhanced training and EDI Strand	Responsible AI	Responsible AI Courses from The Alan Turing Institute Workshops (5cr)		Responsible Research and Innovation training								
	Research Techniques & Methods	'Starting Your Postgraduate Research', 'Academic writing', 'Writing your Data Management Plan', 'Preparing Papers and Presentations', 'Preparing for your Final Year',										
	Skills Development and EDI training	'Media Training', 'Public Engagement', 'Policy Engagement', 'AI for increasing Business Productivity', 'Entrepreneurship training'										

Figure 1. Overview of CDT Students Training. TB stands for Teaching Block.

## General Information

Title	Introduction to Responsible and Reproducible AI
Unit code	
Credit rating	15
Level	7
Contact hours	Lectures: 6hrs
Other Scheduled teaching and learning activities*	Computer Laboratories: 30hrs
Pre-requisite units	
Co-requisite units	
School responsible	Engineering
Member of staff responsible	Mingfei Sun
ECTS**	7.5
Notional hours of Learning***	150

## Aims and Brief Descriptions

This unit aims to: *provide CDT PhD students with an introductory level of training on the ethics in AI.*

This course has three parts:

## Aims and Brief Descriptions

This unit aims to: *provide CDT PhD students with an introductory level of training on the ethics in AI.*

This course has three parts:

- ▶ **Operationalising Ethics in AI:** to incorporate ethical AI principles into the systems design process (taught by The Alan Turing Institute). The students should enrol this course in the Alan Turing Institute Courses website and obtain a certificate upon completion.

## Aims and Brief Descriptions

This unit aims to: *provide CDT PhD students with an introductory level of training on the ethics in AI.*

This course has three parts:

- ▶ **Operationalising Ethics in AI:** to incorporate ethical AI principles into the systems design process (taught by The Alan Turing Institute). The students should enrol this course in the Alan Turing Institute Courses website and obtain a certificate upon completion.
- ▶ **Team Coding Hackathons:** to bring students together to create, innovate and hack on ideas that inspire them. The students should brainstorm a project idea under a given thematic topic and present their idea through coding demonstrations. It provides students with necessary trainings of basic programming skills, including the use of Git, Docker, efficient Python programming, and the use of Jax/Pytorch framework

## Aims and Brief Descriptions

This unit aims to: *provide CDT PhD students with an introductory level of training on the ethics in AI.*

This course has three parts:

- ▶ **Operationalising Ethics in AI:** to incorporate ethical AI principles into the systems design process (taught by The Alan Turing Institute). The students should enrol this course in the Alan Turing Institute Courses website and obtain a certificate upon completion.
- ▶ **Team Coding Hackathons:** to bring students together to create, innovate and hack on ideas that inspire them. The students should brainstorm a project idea under a given thematic topic and present their idea through coding demonstrations. It provides students with necessary trainings of basic programming skills, including the use of Git, Docker, efficient Python programming, and the use of Jax/Pytorch framework
- ▶ **Paper Reproducibility Challenge:** to investigate reproducibility of papers accepted for publication at top conferences, and verify the empirical results and claims in the paper by reproducing the computational experiments, either via a new implementation or using code/data or other information provided by the authors. The students should submit their results in the form of blogposts.

## Intended Learning Outcomes

On the successful completion of the course, students will be able to:

- ▶ Explain the concept of ethics and ethical theory in relation to transparency and fairness;
- ▶ Apply the algorithmic tools and technologies to accommodate fairness and transparency;
- ▶ Implement research ideas with basic programming tools;
- ▶ Reproduce the main results from a technical paper and explain it in a short and understandable way.

# Teaching Plan

## ► Fall Term:

- Two lectures (~~Week 1 & 3~~, changed to 30 Sep & 28 Oct): 2 hours each
- Computer Lab Sessions (~~Week 4 & 5~~, changed to Week 6 & 7): 3 hours each

## ► Spring Term:

- One lecture (~~Week 1~~, changed to 10 Mar): 2 hours
- Computer Lab Sessions (~~Week 1, 2, & 3~~, changed to Week 7, 8 & 9): 3 hours each

Month/YEAR	Week Name	S+	Wk									
		No.	M	T	W	T	F	S	S			
Sept 2024	SVAC 14	52	9	10	11	12	13	14	15			
	WW 2024	1	16	17	18	19	20	21	22			
	SEM1W01	2	23	24	25	26	27	28	29			
	SEM1W02	3	30									

Oct 2024	SEM1W02	3	1	2	3	4	5	6				
	SEM1W03	4	7	8	9	10	11	12	13			
	SEM1W04	5	14	15	16	17	18	19	20			
	SEM1W05	6	21	22	23	24	25	26	27			
	SEM1W06	7	28	29	30	31						

Nov 2024	SEM1W06	7	1	2	3							
	SEM1W07	8	4	5	6	7	8	9	10			

Month/YEAR	Week Name	S+	Wk									
		No.	M	T	W	T	F	S	S			
Feb 2025	SEM2W01	20								1	2	
	SEM2W02	21	3	4	5	6	7	8	9			
	SEM2W03	22	10	11	12	13	14	15	16			
	SEM2W04	23	17	18	19	20	21	22	23			
	SEM2W05	24	24	25	26	27	28					

Mar 2025	SEM2W05	24	1	2								
	SEM2W06	25	3	4	5	6	7	8	9			
	SEM2W07	26	10	11	12	13	14	15	16			
	SEM2W08	27	17	18	19	20	21	22	23			
	SEM2W09	28	24	25	26	27	28	29	30			

## Assessment

This unit is assessed as pass/fail only. All elements of assessment must be passed to pass the unit.

Assessment task	Length	How and when feedback is provided	Weighting within unit (if relevant)
Coding assignment	15 hrs	Coding submissions will be assessed with marks on different rubrics and comments/suggestions will be given on what coding skills to improve upon.	0.5
Presentation	12 hrs	Feedback will be given in peer reviews and marks on different rubrics.	0.2
Blogpost report	40 hrs	Reports will be marked on: results coverage, technical clarity, and blogpost understandability.	0.3
Operationalising Ethics in AI Online Course	8 hrs	Online course run and automatically marked by Alan Turing Institute. Upon completion, students must obtain a certificate to demonstrate that they have passed this element.	Pass/Fail*

# In This Lecture (1)

Let's first understand Ethics in AI: to enrol the online course

## ► Operationalising Ethics in AI - Intermediate

The screenshot shows the website of The Alan Turing Institute. At the top left is the institute's logo: 'The Alan Turing Institute'. To its right is a navigation bar with links: Home, Events, News, About us, Research, Skills, People, Opportunities, Partner with us, Contact us, and a magnifying glass icon for search. Below the navigation is a large banner image of a dark, abstract background with vertical streaks of light. Overlaid on this is the course title 'Operationalising Ethics in AI - Intermediate' in large white text. Below the title is a subtitle 'Relating the concepts of ethical AI principles into the systems design process'. A yellow callout box contains the text 'Duration: 11 - 20 hours' and 'Level: Independent User'. At the bottom left of the banner is a white button labeled 'Enroll now'. Below the banner, a horizontal line separates the main content area. The first section is titled 'Course overview' in bold black text. Below it is a paragraph of text describing the course's focus on incorporating ethical AI principles into the systems design process using a multidisciplinary approach, featuring real-world case studies, narratives, worksheets, and coding exercises. It also mentions the learning experience includes provocative thought experiments. At the very bottom of the page, there is a note about the course being commissioned as part of an open funding call for Responsible AI courses, funded by Accenture and the Alan Turing Institute.

**Course overview**

This course focuses on incorporating ethical AI principles into the systems design process and is taught using a multidisciplinary approach. The course features real-world case studies, first-person narratives, self-reflection worksheets and hands-on coding exercises to maximise learner engagement and ensure they can immediately apply what they learn in their work or research context. The learning experience includes provocative thought experiments to challenge and stimulate understanding of ethical considerations in AI.

This course has been commissioned as part of our open funding call for Responsible AI courses, with funding from Accenture and the Alan Turing Institute.

<https://www.turing.ac.uk/courses/operationalising-ethics-ai-intermediate>

# Operationalising Ethics in AI - Intermediate

Overall progress %

6



Course introduction



Fairness Intermediate 1  
- Responsible AI



Fairness Intermediate 2  
- Bias within  
Algorithms



Fairness Intermediate 3 -  
Legal Interpretations of  
Fairness and ...



Fairness Intermediate 4 -  
Technical Interpretations of  
Fairness & Bias



Fairness Intermediate 5  
- Gameplay exercise



Transparency Intermediate  
1 - Philosophical concepts  
of Transparency



Transparency  
Intermediate 2 -  
Explainability



Transparency  
Intermediate 3 -  
Applying Technological  
Concepts



Feedback

# Operationalising Ethics in AI - Intermediate

You can skip Reflection Tasks:

## Reflection tasks

**Reflection questions**

• If you had to make a list of intrinsic values, which ones would you choose, and why?  
• Do you think fairness is an intrinsic or instrumental value? Why?  
• You read articles about using AI in policing. Is it possible to create fair and non-discriminatory AI for predictive policing? Why, or why not?

**Written task**

**Task 1**  
Write a three-paragraph pitch to persuade a client to consider procuring an AI tool because your development team has mitigated bias.  
Show that you understand what fairness is, how it might conflict with other values, but how it can still be achieved.

**Task 2**  
Re-read the Amazon case study and answer:

1. How could this have been avoided?
2. Using the principles learned in this module, make an argument that Amazon is culpable for neglecting bias

Done

You'll need to complete all quiz:

## Activities

Fairness Intermediate 2 - Quiz

# Operationalising Ethics in AI - Intermediate

User report ▾						
MS Mingfei Sun						
Grade item	Calculated weight	Grade	Range	Percentage	Feedback	Contribution to course total
▼ Operationalising Ethics in AI - Intermediate						
QUIZ Fairness Intermediate 1 - Quiz	-	10.00	0-10	100.00 %	-	
QUIZ Fairness Intermediate 2 - Quiz	-	10.00	0-10	100.00 %	-	
QUIZ Fairness Intermediate 3 - Quiz	-	10.00	0-10	100.00 %	-	
QUIZ Fairness Intermediate 4 - Quiz	-	10.00	0-10	100.00 %	-	
QUIZ Transparency Intermediate 2 - Quiz	-	10.00	0-10	100.00 %	-	
QUIZ Transparency Intermediate 3 - Quiz	-	10.00	0-10	100.00 %	-	
QUIZ Transparency Intermediate 1 - Quiz	-	10.00	0-10	100.00 %	-	
Σ AGGREGATION Course total	-	-	0-70	-	-	

Submit your grades to me by **15th Dec** (passing grades: 75% each)

## In This Lecture (2)

How can we be responsible for our codes?

## In This Lecture (2)

How can we be responsible for our codes?

- ▶ Version controls: Git
- ▶ Reproducible development environments: Docker
- ▶ Good coding practice: Effective Python
- ▶ More ...

## What is Version Control?



# Git Basics: What is Version Control?

CC BY-NC-SA

Episode #1

# Version controls: How to use Git

## What is Git?



## Git Hands-on Exercise

- ▶ Create a GitHub account (if you don't have one)
- ▶ Fork the repository from <https://github.com/mingfeisun/COMP64803-Responsible-Reproducible-AI>
- ▶ Clone locally the repo from your own forked version
- ▶ Add your readme markdown file to cohort-2024
- ▶ Git add and commit your changes
- ▶ Git push your changes and make a pull request

# Docker: Most-Used Development Tool



# Docker

Build, ship, and run anywhere.

Products ▾ Developers ▾ Pricing Support Blog Company ▾

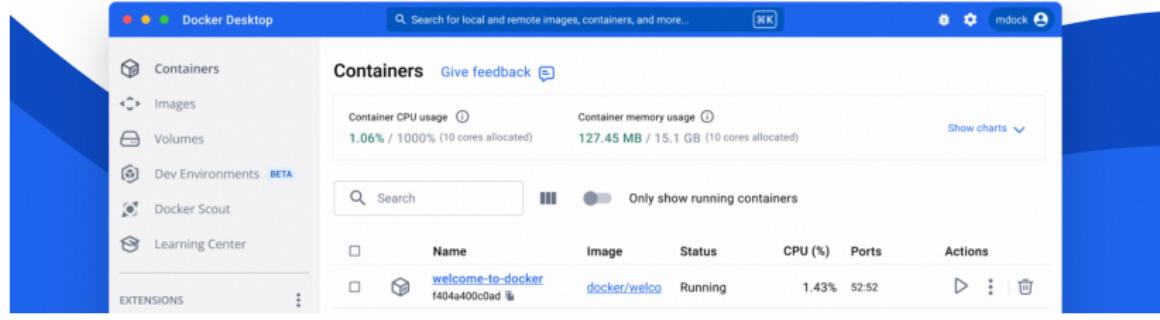
Sign In Get started

# Develop faster. Run anywhere.

Build with the #1 most-used developer tool

 Download for Windows - AMD64 ▾

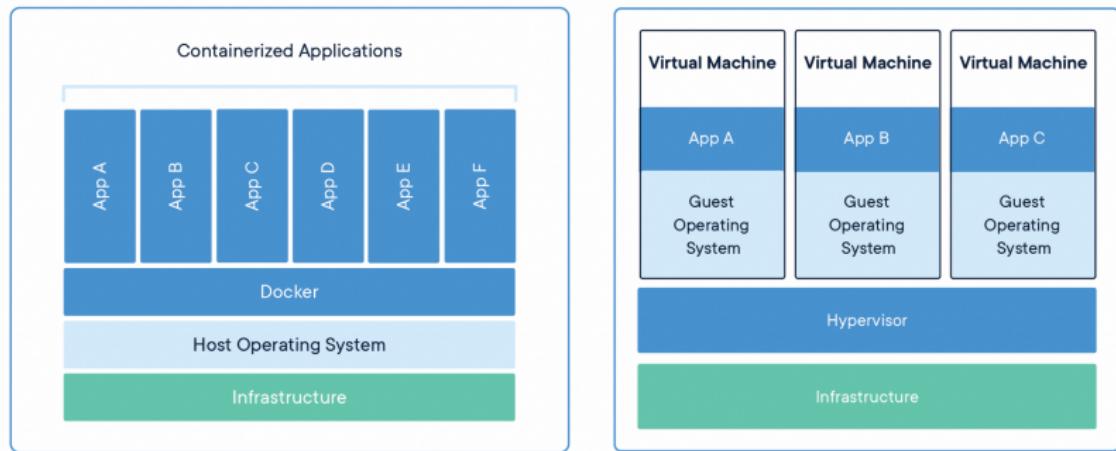
Learn more about Docker



## Comparing Containers and Virtual Machines

Containers and virtual machines have similar resource isolation and allocation benefits, but function differently because containers virtualize the operating system instead of hardware.

Containers are more portable and efficient.



# Docker: What Is Docker?

A woman with long dark hair, wearing a black top, is gesturing with her right hand towards a large blue whale icon on the left side of the slide. The whale icon has a stack of shipping containers on its back. The background is a light blue gradient. At the top left, there is a blue bar with the text "Docker Course". At the top right, there is a small circular icon with a question mark. The main title "What is Docker?" is displayed in large, bold, black font in the center. The slide is part of a presentation, as indicated by the navigation icons at the bottom.

Docker Course

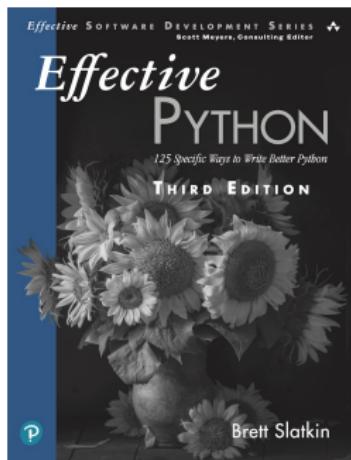
What is Docker?

## Docker Hands-on Tutorial

Go to <https://github.com/mingfeisun/COMP64803-Responsible-Reproducible-AI/wiki/Docker-tutorial> and follow the instructions

**Pythonic Thinking:** The Python community has come to use the adjective Pythonic to describe code that follows a particular style. The idioms of Python have emerged over time through experience using the language and working with others.

- ▶ “Know Which Version of Python You’re Using”
- ▶ Unit tests: “You should always test your code, regardless of what language it’s written in”
  - ▶ pytest: <https://docs.pytest.org/en/stable/>
- ▶ Try to read and understand source codes as much as possible



## For Next Lecture

You are supposed to:

- ▶ Pick **5 specific ways** to write better Python (discuss with your supervisor to determine which ones can be crucial to know for your future research)
- ▶ Prepare a **10-min talk** on your selected effective Python ways
- ▶ Give a **live demo or executable code snippets** in a docker environment
- ▶ **Git push your codes** to our course unit repository via pull request (before the lecture)

END LECTURE