Game lesson title	**Coding with AI Assistance**		
Brief intro	This lesson introduces students to using AI tools, specifically Microsoft Copilot and ChatGPT, to assist in coding game elements. Students will learn how to use these tools to generate and modify Python code for text-based adventures and HTML for interactive stories.		
Lesson hook	What if you had an AI coding partner that could help you bring your game to life? Today, we'll explore how AI can assist us in writing code for our games!		
Suggested year level	**9/10**	**Suggested age level**	14-16
Suggested duration	45 minutes	**Costs involved?** (detail below)	N/A
Author	You are free to copy, communicate and adapt this lesson plan which was created by Leon Furze and ACMI, licensed under a Creative Commons Attribution 2.0		
Subject/s			
☒ **[subject] Digital Technologies**			
Curriculum/Capabilities Alignment (VIC/AC) and Skills			
Please find the relevant VIC and AC Curriculum points			
VC2TDI10C04: Implement, modify and debug modular programs, applying selected algorithms and data structures, including in an object-oriented programming language			

Game Used	**NA**
Game play required?	**No**

CLASSIFICATION	DEVICE/s	INTERNET REQUIRED?	COST (RRP)
Note: Terms and conditions for AI tools are generally 13+ but Adobe Firefly is K-12	Cloud based – any device	Yes	School subscriptions vary. Access to Copilot and ChatGPT free for limited use.

Important note about Game Classification

As classifications can change, teachers are responsible for checking the latest videogame classification and suitability for their class age group. Please visit https://www.classification.gov.au/ and https://www.commonsense.org/education/ to guide you.

How are games used by students in the lesson?

☐ **Watchers** – observing, analysing and evaluating. Learning about the world and ourselves through understanding the impact of games culture and industry.

☐ **Players** - learning by playing videogames - learning things applicable to life outside of (and in) the game e.g. flight simulators, esports, etc.

☒ **Makers** - learning through making games (coding, creative production, teamwork, leadership)

☐ **Explorers** (Minecraft) – imaginative, self-directed, exploratory/sandbox learning.

eSafety Considerations

Ensure students are aware of responsible internet use and privacy considerations when using AI tools.

Prior knowledge/skills (Required/Recommended/References)

Basic understanding of programming concepts. Familiarity with the game concepts developed in previous lessons.

Student outcomes

Learning goals	KNOW: How AI can assist in coding tasks; The capabilities and limitations of AI coding tools

	BE ABLE TO: Use Microsoft Copilot and ChatGPT to generate and modify code for games; Debug and customise AI-generated code
	IMPROVE: Programming skills in Python and HTML; Problem-solving skills in a coding context

LESSON SEQUENCE	
Prep & introduction activities	1. Brief introduction to AI-assisted coding and its applications in game development 2. Demonstration of Microsoft Copilot and ChatGPT for code generation
Main lesson activities	1. Guided practice: As a class, use AI to generate a simple game element (e.g., a character dialogue system) in Python (5 minutes) 2. Individual work: Students use AI tools to generate code for their games (15 minutes) a. Text-based adventure element in Python OR b. Interactive story element in HTML 3. Pair programming: Students work with a partner to debug and customise their code (5 minutes)
Reflection activity	1. Class discussion: How did AI assist you in coding? What challenges did you face? 2. Quick write: How might AI-assisted coding change the game development process?
Differentiation: modification, extension and inclusion notes	Modification: Provide code templates for students to modify with AI assistance. Extension: Challenge advanced students to implement more complex game mechanics using AI-generated code. Inclusion: Ensure coding environments are accessible for students with visual or motor impairments.
Student Homework/Further Work	
Continue developing game code, focusing on integrating multiple elements into a cohesive game structure.	
Assessment	
Formative assessment based on students' ability to effectively use AI tools to generate and modify code, and their understanding of the code produced. Collect code samples for review.	

ACMI Game Lessons acknowledges the support of the Department of Education, Victoria, through the Strategic Partnerships Program.