

Professional Learning

Session Notes



MakeCode Arcade

Future Sessions:
bitlink.com.au/newsletter/



Background & Context

These notes and lesson ideas have been prepared as an accompaniment to a professional learning session that took place in Launceston, Tasmania on December 12th, 2022. At that session, James and Troy from Bitlink talked about their experience running Gamesfest, a 13-week course that used the video game industry as a way to engage job seekers in identifying skills through making video games. They also talked about MakeCode Arcade and how it can be used in the classroom. That section of the presentation was the inspiration for this guide.

More Information

This If you would like to access more design and digital technologies supporting materials, or keep an eye out for future events, check out the Bitlink website and sign up for our mailing list.

bitlink.com.au/newsletter/

Why teach game design?

Bringing game design into your classroom can be a way to really engage your students in all manner of learning activities.

Games are a great ice-breaker. Games help students build trust – if you want to play a game together, you have to agree to play by the rules of the game.

There are also plenty of cross-curricular opportunities too. Dice games are a great way to introduce concepts like probability. Board games can be great opportunities to investigate graphic design. Video games involve coding, art, sound, design...

Games can also be a great way to support inquiry – themes can be adjusted to suit your current learning area. Can your students make a game about sustainability? How about a game that teaches financial literacy? Games can be a really engaging way to hook students into the content they might otherwise bounce off.

Games are a great way to teach design and soft skills. Design thinking, prototyping, iteration, communication, and collaboration are all embedded in building games. It's also a great example of project-based learning, and can make a really great portfolio piece that students are proud to show off.



[Australian STEM Video Game Challenge teacher pack](#)



[Games 4 Change Student Challenge teacher resources](#)

Why we like MakeCode Arcade

There are plenty of great tools out there for making video games. Scratch, Game Maker, Godot, Unreal, Unity... But for the simplicity of teaching gaming concepts in the classroom, there are a number of reasons we like Makecode Arcade.

- It runs in a browser – you don't need to ask IT to install anything. We recommend Chrome or Edge as they have some extra support for downloads via USB.
- No accounts required. Students don't need to create an account (or remember a password) to use Makecode Arcade.
- Scratch-style visual programming. It's very easy to start making games with block coding, but you can also transition to JavaScript or Python for advanced students.
- It's strongly related to MakeCode for micro:bit, so there are plenty of skills that will transfer across.
- There are lots of online resources – there's plenty of tutorials and skill maps built right in.
- Integrates with hardware! You can purchase hardware devices to download games to and play them. There's a special sort of magic to seeing your game run on a little device rather than a computer screen.



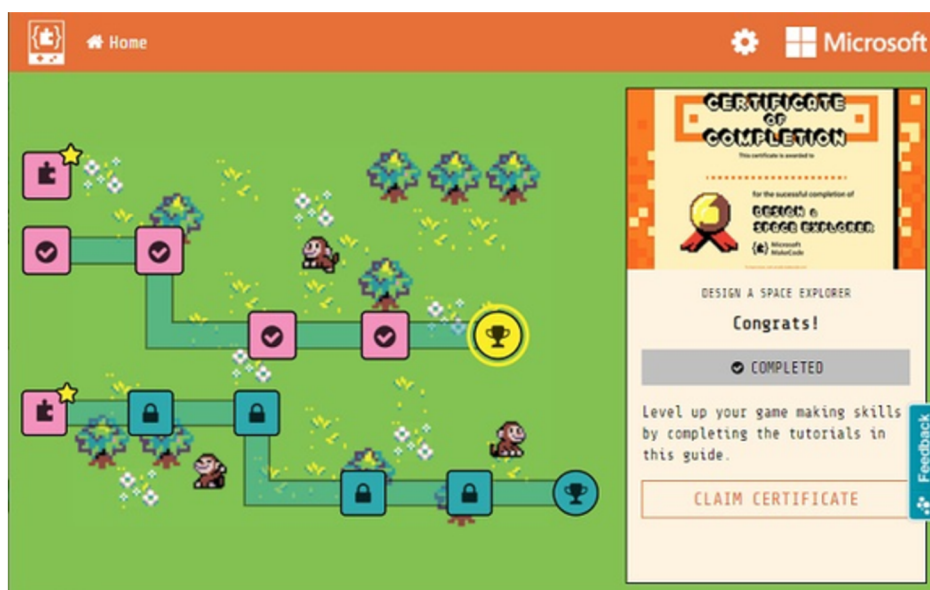
[What is MakeCode Arcade?](#)



[MakeCode Arcade Home Page](#)

Primary School Ideas: Skillmaps

The easiest way to get started with Makecode is to use the built in 'skill maps'. These skill maps are self-guided tutorials that are designed to scaffold the creation of a game by building it through a series of stages. The design of a skill map mimics the over-world screen from many classic video games.



Each location on the map is a tutorial that covers how to build part of the game. These tutorials are scaffolded, so each one build on the previous tutorial. Once a skill map is completed, students get a certificate to show they have completed it, as well as a playable game.

Skill maps are easy to use and require no real prep – just point students at the website and they can move at their own pace. If you really like skill maps, you can build your own, but they do require a fair bit of work to implement. It's also important to consider literacy, as skill maps tutorials involve a lot of text.

Secondary School Ideas: Classic Arcade Games

For older students, we think a great way to get started is to try to remake classic arcade games! Games like Frogger and Gradius are great examples that teach a lot about game mechanics like sprites, movement, collisions, projectiles, and win conditions. Note that some games like Pacman might seem simple, but are actually pretty difficult to create – so choose carefully!



Even though there are not a lot of design decisions in recreating a classic, there is still a lot of creativity to explore, like creating new artwork to change the game's theme. You might like to start with a play session, letting students make notes about how a classic game works. We encourage students to make their game in small sections that let them test each new mechanic. For an example of how we built games in playable stages, see the next page.



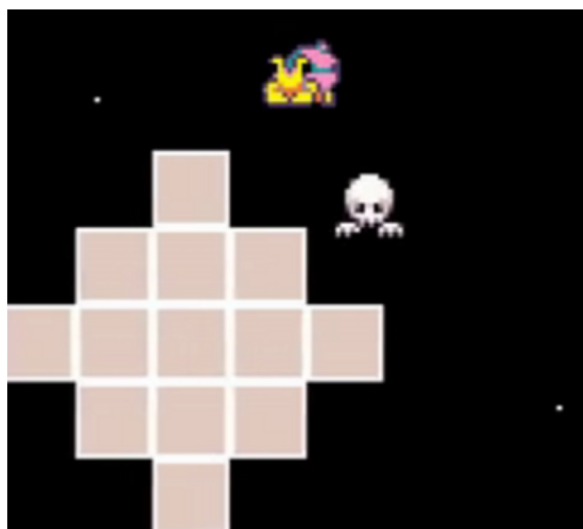
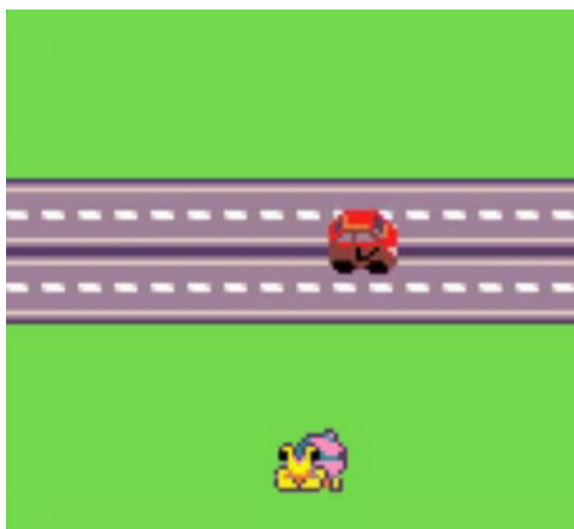
[Play Frogger online \(may contain ads\)](#)



[Play Gradius online \(may contain ads\)](#)

Our structure for building classic games

- Get your character moving. Create a sprite and enable it to move when you press the buttons.
- Create a goal, and make the game winnable when you touch it.
- Add a background to your game. You can use tile maps to create obstacles with the level.
- Add enemies to avoid. Make the game end if you hit them.
- Make the enemies move. Add projectiles if needed.
- Add music. And sound effects. And other 'nice to haves', like a scoring system, or a title screen.
- Add a second level and make it loadable.
- Repeat as often as you like!



Stay in touch:
bitlink.com.au/newsletter/

