

DEVELOP AT UBISOFT



The Develop at Ubisoft mentorship aims to attract and develop diverse talent for our studios around the world. With more people playing games than ever before, we have an exciting opportunity to create inclusive entertainment that is truly reflective of our diverse world.

The industry continues to benefit from different backgrounds, fresh perspectives, and new ideas. Develop at Ubisoft supports this evolution by highlighting emerging talent – when they thrive, so does the industry.

Successful applicants receive mentorship from our experts across programming and game design.

Programming: The Challenge

The challenge is to create a **1 button game** with the theme of “**Crowded Spaces**”.

You have complete freedom to be creative. For example, you could create an arcade game where players need to fight the crowds and find their way out. You could also create a space game! **In short, your game can be anything as long as it fits the theme “Crowded Spaces”!**

The game may use directional inputs in addition to one button. Participants may map a single action (e.g. jumping, shooting, flipping gravity), or multiple actions to a single button. Examples of multiple actions could include adding contextual actions based on where the player is or firing/reloading a weapon based on ammo count. In addition to the single button for performing actions we are also allowing participants to use additional buttons or gamepad directional inputs for movement if they choose.

This is a technical challenge, so we will ignore artistic merit in the judging. We’ve intentionally kept the rendering limited because we want to see how well your simulation works, rather than how good you are at drawing. However, it would be nice if you show us some fun and creativity in your game.

This challenge must be done individually, team-based submissions will not be reviewed. All the code you submit must have been done by yourself, however, you are welcome to use the standard library.

Development Environment

Your entry must be a Windows application written entirely in C++. You will need to download Microsoft Visual Studio 2019 or 2022 to successfully complete this project. We recommend Visual Studio Community 2022.

<https://visualstudio.microsoft.com/vs/community/>

If you're a Mac user, please use Boot Camp to install Windows 10.

We will provide you with a simple API which allows you to draw lines, get pad/keyboard inputs and play sounds. You should have no reason for modifying this API. A project can also be found in the package, providing examples of how to use the API.

Tips

- One step at a time: start small to reach something big.
- When you reach a stable state in your work, we advise you to save your project to be able to revert back to it in case you get stuck. For example, you can use a source control solution such as git, or even make a local copy of your code to serve as backup.
- Before starting to code something that seems generic, check in the standard library, if it's not already provided there.
- Check that the release version compiles before submitting your code.

Final project submission:

****All items are mandatory****

1. The link to a YouTube video of a play-through of your game with screen capture software (e.g. OBS) highlighting the gameplay and technical features. Please make sure the video's quality is good. It should not take longer than 5 minutes in length.
2. Your complete code and documentation saved in a .zip file:
 - Your application package must be named as follows:
DAU_Ubisoft_2023_Programming_FirstName_LastName.zip
 - Please do not enter your name anywhere inside of your code or documentation, only in the Zip file name and your resume.
 - **Documentation must be in PDF or PowerPoint format**
3. Submit your final project [here](#) by **Friday, 12th January 11:59pm CET**

Code Structure: (1-15)

- Clear and simple extensible code
- Self-documenting code with clear interfaces
- Well-managed memory and data flow

Technical challenge: (1-10)

- Usage of modern C++
- Applicable algorithms and data structures

Playability: (1-5)

- For example: AI behaviors and character control fluidity, etc.
- Fun game

Reminder: This is a technical challenge, so we will ignore artistic merit in the judging and focus mainly on your programming skills.

Timeline:

October	9-13th	Results week: application confirmation
	16-20th	1st Networking call : « Get to know your mentor»
	23-27th	2nd Networking call : « How does the API work?»
	30-3rd Nov	Mentor Call 1. Intro & Brief
November	13-17th	Mentor Call 2.
	27- 1st Dec	1st version submission
December	4-8th	Mentor Call 3.
January	8-12th	Final project deadline
February	12-16th	Mentor Call 4. Presentation Training
March	8-12th	Mentees presentation & ending ceremonies