Guide contents

Pick a first project

Establish the loading path

Change one existing behavior

Package and maintain

Sources / References

Documentation: In progress **Quick start: In progress**

Offline copy from ModDocs. Source links remain clickable.

Oxygen Not Included modding

Oxygen Not Included mods can change existing behavior, add buildings and UI, or supply art and localization. The game's mod manager loads local and Workshop mods, but community documentation says there is **no stable, officially exposed game API** for much of the code-mod route. Game updates can therefore break patches. Start with a loading-only mod before changing a building. See Cairath's modding guide and the community project template.

> **Quick start:** use the current game build and a minimal C# mod template to log one unmistakable startup message. Verify the mod appears in the in-game list, enable it, restart if requested, and inspect the log. Only after that should you patch a game method.

Pick a first project

Goal	Route	First verification
Change a building value	C# mod and a targeted Harmony patch	One existing building reflects one changed number
Add a building	Code registration, prefab/art and localization	Building appears in the expected menu and works in a fresh colony

Goal	Route	First verification
Change text	Localization files or code localization	Correct language is shown without missing keys
Change animation or art	Game-compatible asset workflow	Image renders at the intended scale and state

The community template walkthrough demonstrates a first DLL and a generator-value patch, but its framework and example build numbers are historical. Use its sequence as a learning aid; check current game assemblies and Klei's DLC metadata discussion before copying any version number or method signature.

Establish the loading path

Make a separate working directory and install a .NET development environment appropriate to the game's current managed assemblies. The community walkthrough explains how to locate `OxygenNotIncluded_Data/Managed` in the installed game and reference the required assemblies in a C# class library. Reference your own installed copy; don't distribute Klei DLLs with your mod. A decompiler such as ILSpy helps confirm that a method named in an old tutorial still exists in your build.

Use the mod structure described by the Cairath guide and place the compiled mod in the game's local or development mod directory. The example template uses the user's `Documents/Klei/OxygenNotIncluded/mods/Dev` folder; verify that your current install recognizes it. Add the metadata files appropriate to the base game or DLC. The `mod_info.yaml` fields determine content compatibility, and copied `lastWorkingBuild` values from old examples should be replaced with values verified against the current game.

Run the game with only this mod enabled. Check three distinct observations: the mod is listed, it is enabled, and its startup message appears after restarting. If the list is empty, check folder nesting and metadata before touching C# code. If the list shows it but the startup message is absent, inspect the mod's load errors and assembly references.

Change one existing behavior

Once loading works, select a single existing building property. The community template's first-mod section demonstrates the overall pattern: locate a building definition, patch its creation point, and change one output property. **Do not copy its exact method names blindly**; inspect your installed game's assembly first. Add logging around the patch, test a new building in a disposable colony, and compare the value with and without the mod. A mod that loads does not prove the patch actually ran.

Avoid changing a global value and then using an old save as the only test. A new object and clean save help distinguish construction-time behavior from saved-state behavior. Test with the Spaced Out! DLC enabled and disabled if you claim both are supported. If a patch works in one context but not another, record the affected game build and content mode.

Package and maintain

Package the DLL, metadata, original art or translations, and only dependencies the user cannot reasonably obtain from the game. State the supported game build, base/DLC compatibility and known mod conflicts. Test the packaged copy from a clean mod folder rather than relying on the development output. The Cairath sample mods repository is educational, but its README itself cautions that the repository is not updated often, so treat its code as historical examples.

This guide is **unfinished**. It documents a safe first-mod path but does not include a freshly compiled, current-build building registration, animation export or Workshop submission test. Those parts need game-version-specific verification.

Sources / References

External project and documentation links cited in this guide. Check the upstream release and version before following version-sensitive steps.

- Cairath's modding guide
- community project template

- Klei's DLC metadata discussion
- Cairath guide
- community template's first-mod section
- Cairath sample mods repository