Build a safe starter menu with DeluxeMenus

DeluxeMenus turns inventory-style menus into buttons, information panels and navigation. A menu can also run privileged actions, so build a read-only example first. A polished item icon does not make the underlying command safe or guarantee that a purchase checks the right player.

Prepare a small test menu

Check the required server version and dependencies for the selected DeluxeMenus release. Its official documentation lists PlaceholderAPI as required. Install compatible releases using the plugin guide, stop normally and keep a backup of existing menus, permissions and any databases involved in purchases.

Start once and read Console. Stop again before manually changing files. Do not import an entire menu pack directly into production: inspect its actions, external links, commands and permissions first. Packs can include console actions that assume different plugins, worlds or rank names.

Register and edit the menu

1. Open `plugins/DeluxeMenus/config.yml`. Under the existing `gui_menus` section, register a uniquely named menu and a unique YAML filename using the documented external-menu format.
2. Use the plugin's supported reload command, `/dm reload`, from an administrator account. Check its output; a reload request alone does not prove the menu parsed.
3. Open the generated file under `plugins/DeluxeMenus/gui_menus/` and work from the example for your installed release.
4. Give the menu a distinct `open_command`, a small valid inventory size and one visible information item.
5. For the first click action, use a harmless message or close action. Do not start with a payment, rank change or console command.

Use spaces consistently in YAML and preserve list indentation. Material names must exist in the Minecraft version you run. Keep the menu ID, filename and opening command distinct in your

notes: they identify different parts of the setup.

Test the player route

The administrative `/dm open` command is not the ordinary player's menu command. Test the configured `open_command` with a non-operator account. Add narrow view or click requirements where appropriate and check that a denied player cannot trigger the protected action by using a different command.

Check placeholders separately through PlaceholderAPI diagnostics. A raw placeholder may mean a missing expansion or unsupported data source, rather than a broken menu slot. Do not grant broad administrator permissions to make a placeholder appear.

Add transactions only after basic validation

Review who executes each action: a player action, a console action and a cross-server action have different effects. Use only the documented player placeholder in trusted commands, and avoid constructing privileged commands from arbitrary user input. Check affordability, permission and success behaviour before promising a reward.

Test insufficient funds, full inventory, rapid repeated clicks and reconnecting. Record balances and received items so duplicated rewards or unexpected charges are visible. If an update breaks a menu, restore its previous configuration and compatible dependency versions together rather than clearing player data.

Further reading: external menus, commands and action types.

Looking for another plugin? Browse the Minecraft plugin guide directory.

Revision #2
Created 2026-11-09 08:57:16 UTC by CloudExa Hosting Admin
Updated 2026-11-09 08:57:17 UTC by CloudExa Hosting Admin