In NvChad v2.5+, the folder structure is very clean. You no longer use a "custom" folder; instead, you put your files directly in the lua/ directory.
Here is how to add a plugin and a custom keybinding.

1. How to Add a Plugin

NvChad uses **Lazy.nvim** to manage plugins. To add a new one, you should create or edit the file lua/plugins/init.lua.
Example: Adding "Todo-Comments"
(This plugin highlights comments like TODO: or FIXME:)

1. Open lua/plugins/init.lua in Neovim.
2. Add the plugin block to the returned table:

Lua

```
local plugins = {
  -- Your new plugin goes here:
  {
    "folke/todo-comments.nvim",
    dependencies = { "nvim-lua/plenary.nvim" },
    lazy = false, -- Load it on startup
    config = function()
      require("todo-comments").setup()
    end,
  },
}

return plugins
```

3. Save the file (:w). NvChad will detect the change and prompt you to install it (or just run :Lazy install).

2. How to Add Custom Keybindings

Mappings are handled in lua/mappings.lua. NvChad uses the standard Neovim function vim.keymap.set.
Example: Adding a shortcut to Save with Ctrl + s

1. Open lua/mappings.lua.
2. Add your custom maps at the bottom of the file:

Lua

```
local map = vim.keymap.set

-- Standard Neovim mapping syntax:
-- map("Mode", "Shortcut", "Action", { desc = "Description" })

-- Save file with Ctrl + s in Normal mode
map("n", "<C-s>", "<cmd> w <cr>", { desc = "Save file" })

-- Open a new vertical split with Leader + v
map("n", "<leader>v", "<cmd> vsplit <cr>", { desc = "Vertical Split" })
```

Note: In Neovim, <leader> is usually the **Spacebar**. So <leader>v means pressing Space then v.

---

3. The "Master Switch": chadrc.lua

The lua/chadrc.lua file is the heart of your UI. This is where you change themes or toggle UI features.

If you want to...	Look in this file:
Change the Theme	lua/chadrc.lua (look for theme = "...")
Add/Disable Plugins	lua/plugins/init.lua
Change Keys	lua/mappings.lua
Change Editor Options	lua/options.lua (e.g., line numbers, indent size)

Pro Tip: The "Cheatsheet"

NvChad has a built-in interactive cheatsheet. If you ever forget your keybindings, just type:
<leader>ch (Space + c + h)
Would you like me to help you set up a **Language Server (LSP)** for a specific language (like Python or JavaScript) so you get auto-completion and error checking?