

# Lisp's Guide to Adventure

I want to create a small interactive text adventure using Common Lisp. It should allow users to create their own rooms, items, actions, etc. using macros and allow them to play the game using their self-created assets.

## Motivation

Common Lisp is very well-suited for creating your own macros. Because you can define new functions in run-time, it makes it easy for players to define new rooms using a template, which will be given by macros.

## Key language features

- Macros: It lets me define templates for rooms, items and actions, so player can more easily add assets to their games.
- Symbolic Data: Because everything is stored as Lists, it allows me to more efficiently represent assets and store them
- Dynamic evaluation: This feature allows the users to define new assets during runtime.

## Design

There will be a world layer where every created room, item and action is stored separately, a macro layer where every macro is stored and a game engine layer where basic actions are defined like inventory, take, go, etc.

## Scope

### Must Haves:

- Working macro definitions for all assets
- Simple commands like go, inventory, etc.
- Persistent player state
- Ability to randomly chain created rooms together and put items in these rooms

### Nice to Haves:

- Ability to save game in a file and load it again
- Scriptable and conditional events
- (Combat, very unlikely)
- Soundtrack (let a simple royalty-free song play in the background)

### Out of Scope:

- Graphics

- Multiplayer

## Challenges

- Scope creep: I think while creating the game I will be very tempted to add new features which I just thought off and forget implementing ones that I need. I will implement features first according to my defined scope.
- Keeping the world persistent: I will need to setup a good system to save the world and all changes within it. If conditional actions like locked doors etc. are too difficult to implement, I may need to cut them
- Macros: I need to keep the finalized macros readable enough so that everybody can create new assets even without understanding the code behind them.
- Randomness: I want the game to work like a rogue-like so it has to add rooms randomly which can be difficult. If this doesn't work, I will either have to do it with a fixed order or ask the user to choose an order.