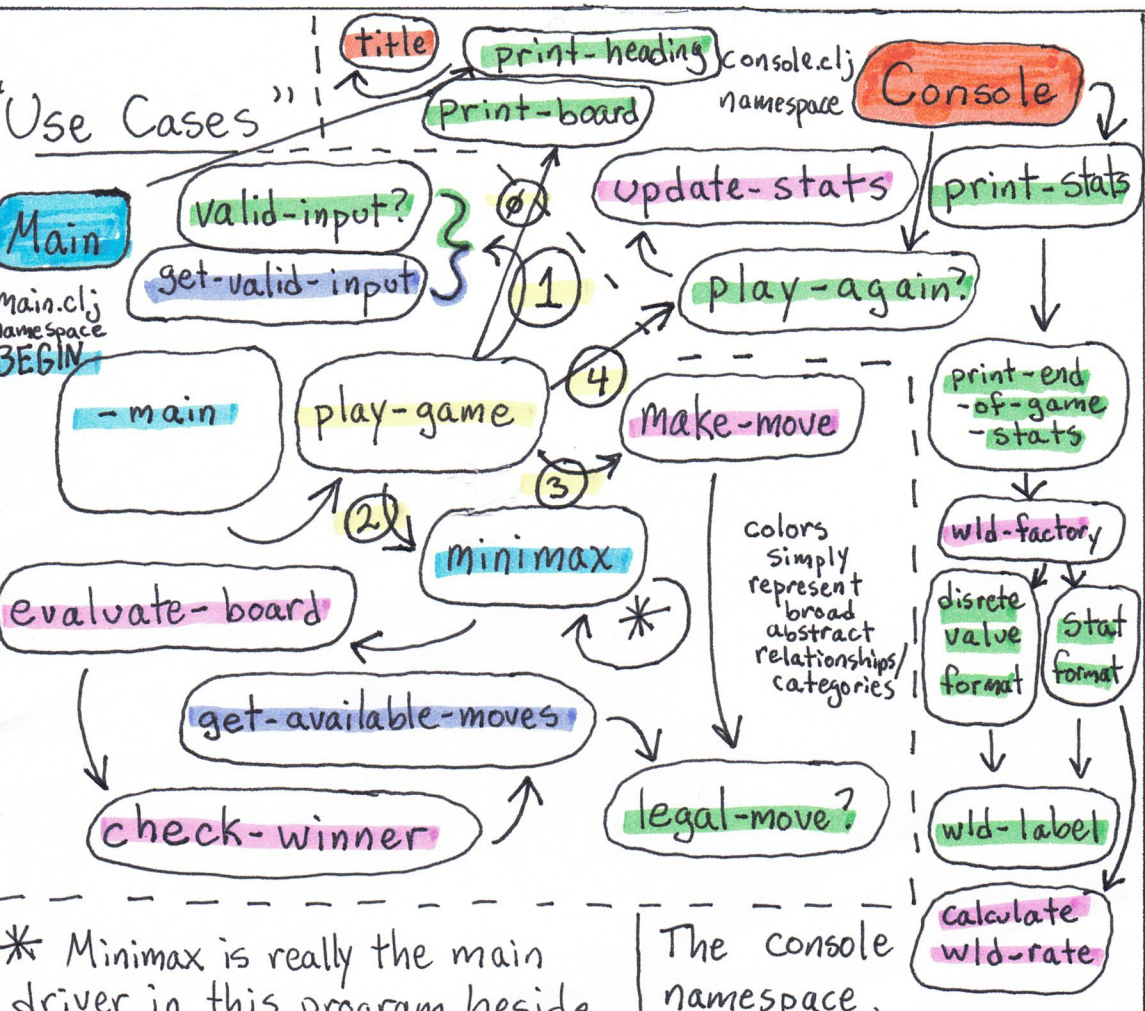
Tic Tac Toe — Refined Word Cloud / Domain Model



* Minimax is really the main driver in this program beside play-game. It invokes most of the more 'logically-precarious" functions, and is recursive; so most time was spent on this

The console namespace, in keeping with its name, just contains the io related fn's.

"objects"

Since Clojure (clj) i a functional language there are few things exactly comprable to objects in a tradition sense. I think the 'board' itsef occupies a space akin to sta however it isn't truly th same, as board and all the other local variables are boond rather than assigned. Thus most emphasis is on functions, which I suppose fall under the category of "use cases"

"actors"

Player/User is the only actor. [cpu] may be considered one, but it really isn't sinc it is a clear algorithm.

In the future, multiplayer or stakeholders may be considered.

Potentially analysts for future records and statistics

"data attributes"

title

label

draw

score [-1, 0, 1]

w/l/d

minimax return [score [x y]] → the next move to take.

core.main.clj fns — Console ns

State/printing — play-game

"active-computation/logic" — "set"

"records"

currently this game keeps track of wi losses, and draws i local storage during the application operat

Long term storage is stretch goal that could manifest multiple ways.

Detailed stats like average number o moves and "favorit