



SSH TINY.CHRISTMAS

a tiny terminal singalong

Dec 24, 2025

ssh tiny.christmas should take you here:

0:00

Hitting space advances the lyrics for all connected carolers.

A little more about the project

I'm using [wish](#) to serve this over ssh and using [bubbletea](#) to render the interface. I've been working on a several of ssh games so this was all familiar territory for me.

Wish and bubbletea are really neat, and I think ssh is a fun delivery mechanism. Some of the other ssh stuff I've been working on has required forking bubbletea

(stay tuned!) but it was nice to use it as intended this time.

Drawing

I think the trickiest thing I'm doing here is using [Unicode Braille](#) for drawing.

Unicode Braille symbols are a 2x4 grid of dots, where any combination of dots can be “raised” – like ⠠, ⠡, or ⠢. This is really handy for drawing with higher fidelity than typical ascii art.

Unicode Braille is also really cool! You can basically think of a Unicode Braille symbol as an 8-bit binary representation of a number. To access a Braille symbol, you take the bits that you want to “raise” and add that to the starting offset for the Unicode Braille codepoint.

I pre-rendered 5 different trees via a Python script. The server picks an appropriate tree for your screen size.

Wrapping up

Honestly the hardest part of this project was realizing that there are like 30 versions of “O Christmas Tree” and none seems to be the canonical version. I just a version with lyrics that I liked. I hope I picked the right one?

Anyway. Merry Christmas!

Thanks for reading!



Keep up with me on my socials 👉

Or sub to my newsletter here! 📧

