Exercise #1
Alternative Clocks

Design Process

By Adrian Crockett

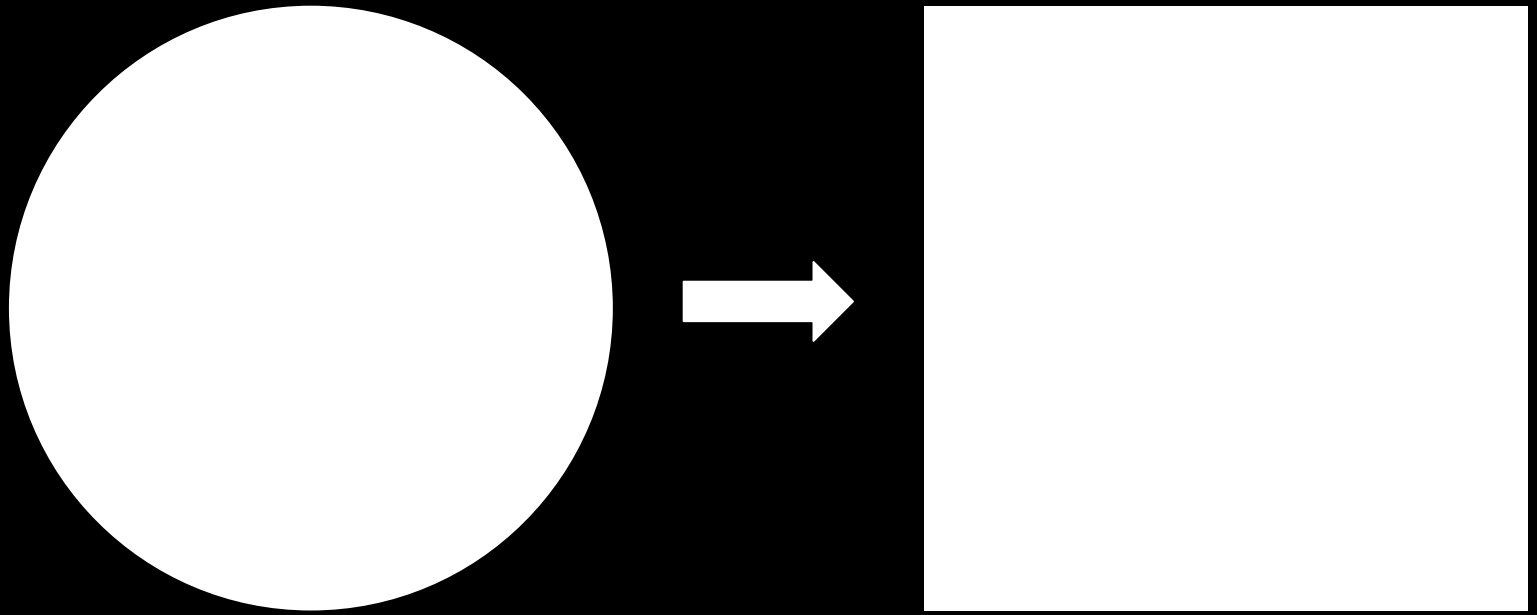
Key design objectives

When we think about representations of time, for most of us the concept is rooted in:

- **A day:**...or where we are within that day
- **Circles:** time is generally visually presented as a circle, representing that at the end of the cycle it will begin again

I wanted to stick within the common unit of a day, but question the circle...additionally, I wanted it to provide the user with some sense of elapsed time that gives visual feedback

Initial thought



Dimensions

There are two common ways of presenting HH, either military (24hr) or civilian (12hr), this resulted in two alternative ways of specifying at the square/grid

- **Military:** HH=24 + MM=60 + SS=60 = 144 units
- **Civilian:** HH=12 + MM=60 + SS=60 = 132 units

With 144 units, an idea came to mind

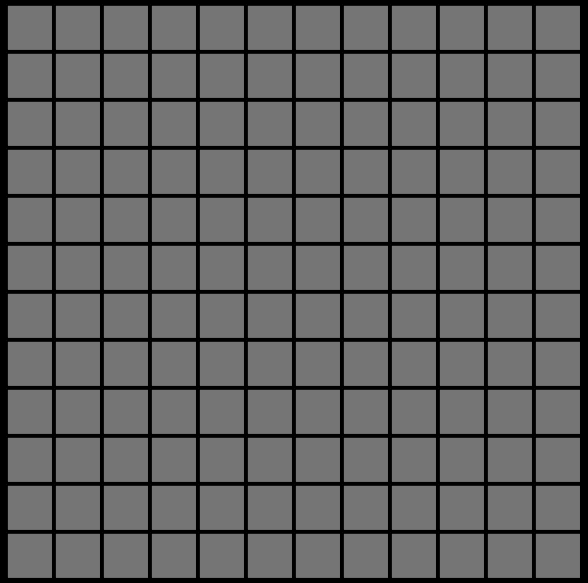
- **12x12:** The square root of 144 is 12, which results in 12 rows by 12 columns
- **12 is the magic number for time:** HH, MM, and SS are all divisible by 12
- **Rows:** Rows 1-2 = HH, Rows 3-8 = MM, and Rows 9=12 = SS

General concept...

HH: 24 hours

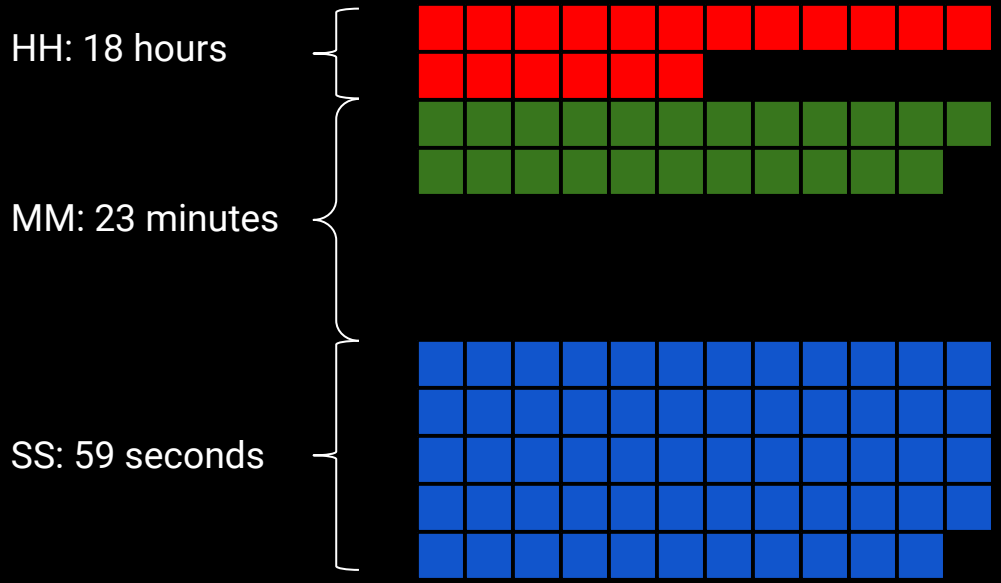
MM: 60 minutes

SS: 60 seconds



144 units of time represented on a 12x12 matrix

In action...18:23:59

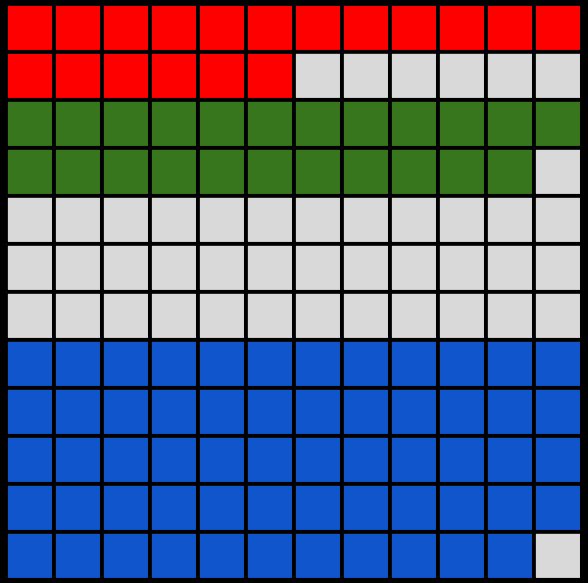


In user testing I tried out two alternatives ideas:

- Flipping rows & columns
- Contiguous vs. segmented

While I only tested with 4 people the results were in favor of this version - which I'm also assuming is significantly easier to program than the contiguous

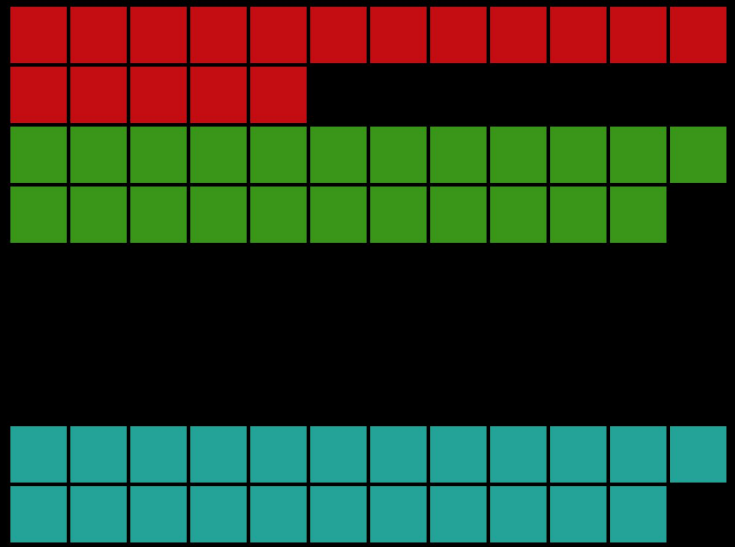
In action...18:23:59 (Alternative)



The final thing I tested, was if the empty grids should be the same color as the background and grid borders…

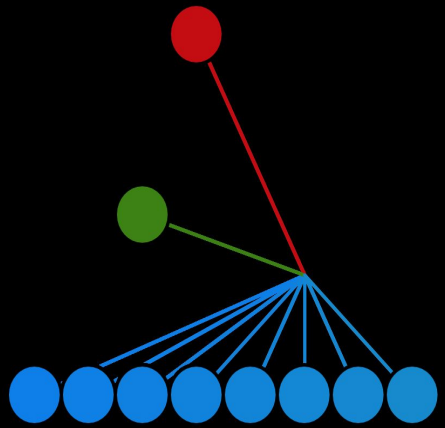
...this area was split down the middle, so I gave myself the golden vote and went with the previous page

Prototype 1



In my first live proptype the code was inefficient (140 lines of code) albeit functional...however it was missing some degree of “wow” so based on my peers comments I played around with some alternatives

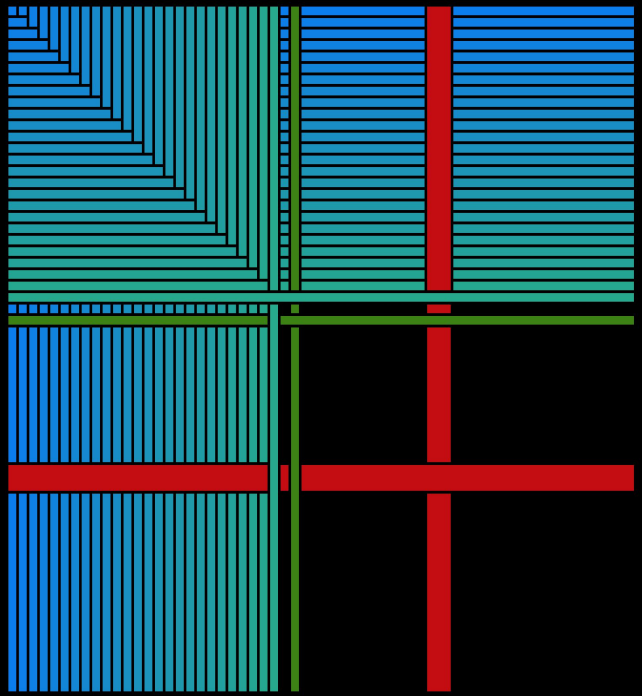
Prototype 2



Building on the same code base and thought process I took the project in a slight detour. I changed RECT to ELLIPSE and also added a connector lines from the center of the clocks grid

Again while this worked it was exceptionally unstable and a lot of playing around with ordering had to go into it

Prototype 3



In my third prototype I completely changed directions, moving away from the design objective of being able to tell an exact time and the continued desire to get closer to a wow factor

One thing to note is the different width of the HH (5 pixels) lines vs. the MM and SS (2 pixels) lines

Decision & Direction

Taking in account the “wow” factor and my daughter's vote, I ended up focusing on the third prototype. Key extensions and reductions are

- **Large:** Added Month and Day as VARs to the project and created a TRANSLATE function on top of the combination of those two results - interestingly this ended up using two coordinate systems on the clock
- **Medium:** Small, plus the dynamic colors that depend on the values of HH, MM and SS respectively
- **Small:** Base version with Simplified color to static values