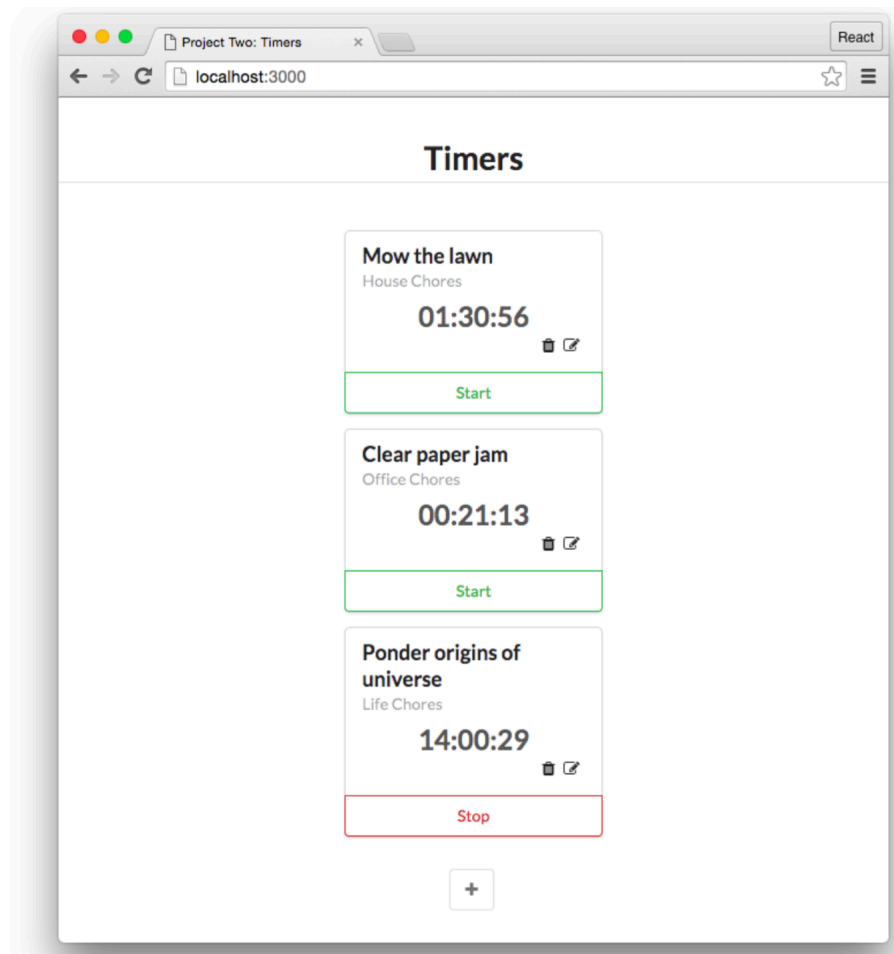


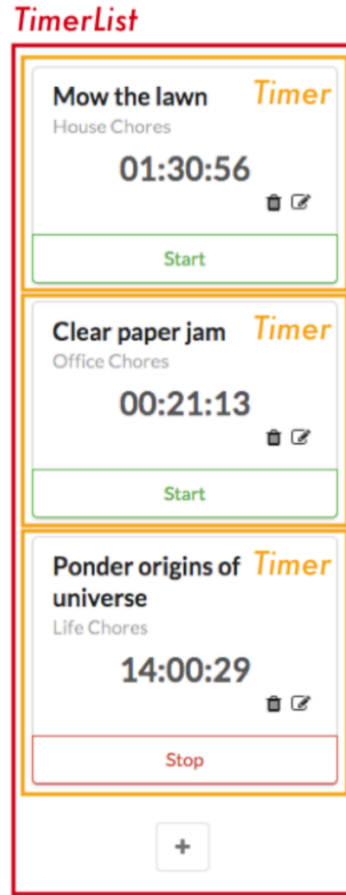
Timer – Logging App using React

development of React app- pattern:

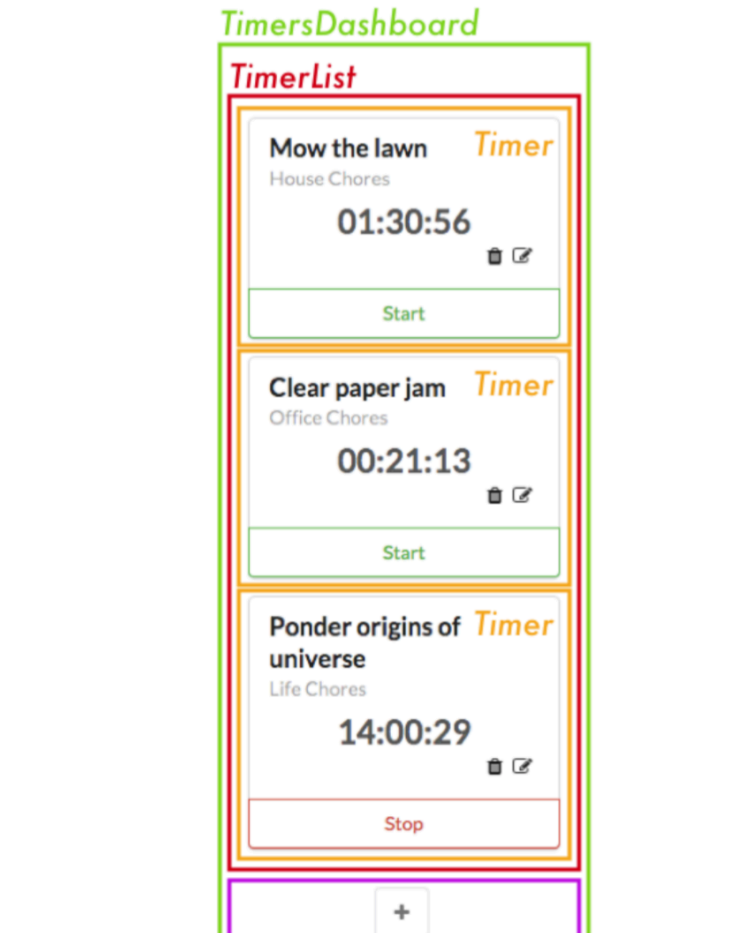
1. Break the app into components
2. Build a static version of the app
3. Determine what should be stateful
4. Determine in which component each piece of state should live
5. Hard-code initial states
6. Add inverse data flow
7. Add server communication



Breaking the app into components



Breaking the app into components



Breaking the app into components

The diagram illustrates the decomposition of a timer application into two distinct components, separated by a vertical line.

Left Component (Display):

- Title:** Mow the lawn
- Category:** House Chores
- Time:** 01:30:56
- Actions:** Delete (trash icon) and Edit (pencil icon)
- Start Button:** A green button labeled "Start" at the bottom.

Right Component (Edit Form):

- Title:** A text input field containing "Mow the lawn".
- Project:** A text input field containing "House Chores".
- Buttons:** Two buttons at the bottom: "Update" (blue border) and "Cancel" (red border).

A single timer: Displaying time (left) vs. edit form (right)



Breaking the app into components

EditableTimer

Mow the lawn *Timer*

House Chores

01:30:56

Start

EditableTimer

Title *TimerForm*

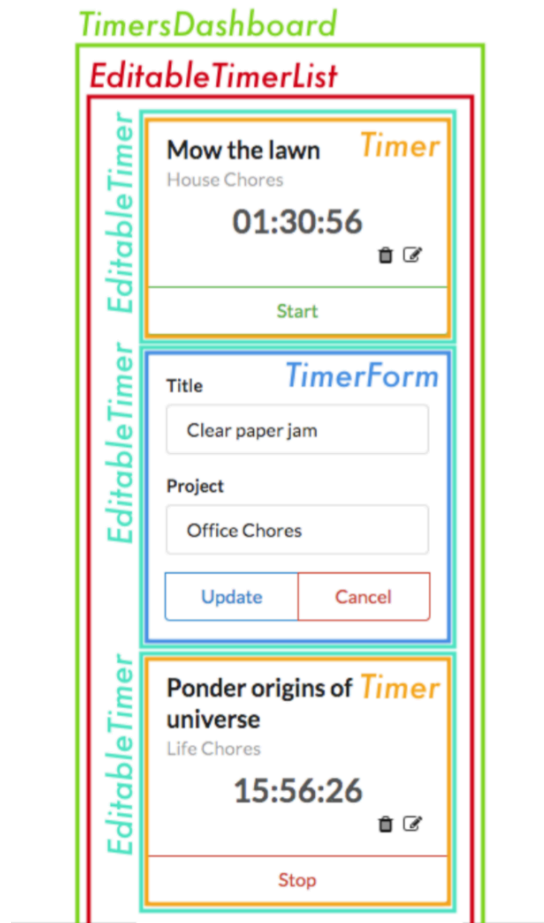
Clear paper jam

Project

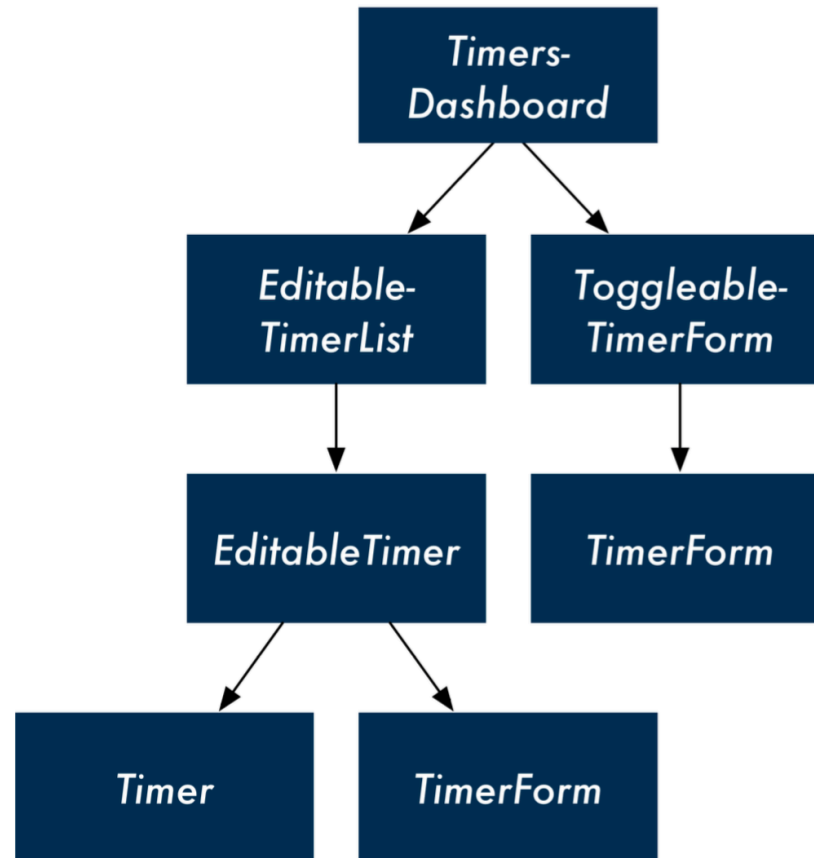
Office Chores

Update Cancel

Breaking the app into components



Component Tree



The steps for building React apps

