

Reacting to Events



Deborah Kurata

Developer | Content Creator | MVP | GDE

@deborahkurata | https://www.youtube.com/@deborah_kurata

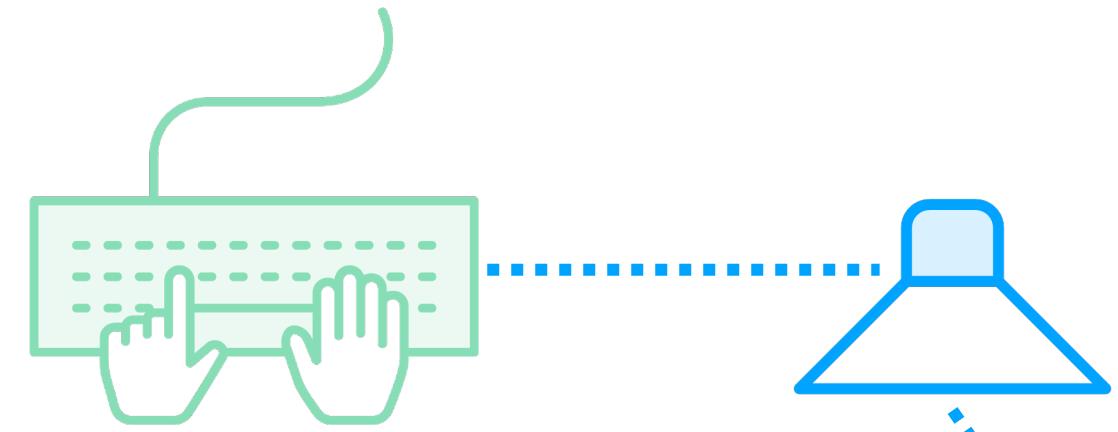
If events are flowing ...

RxJS gets going!



**Events happen
asynchronously over time**

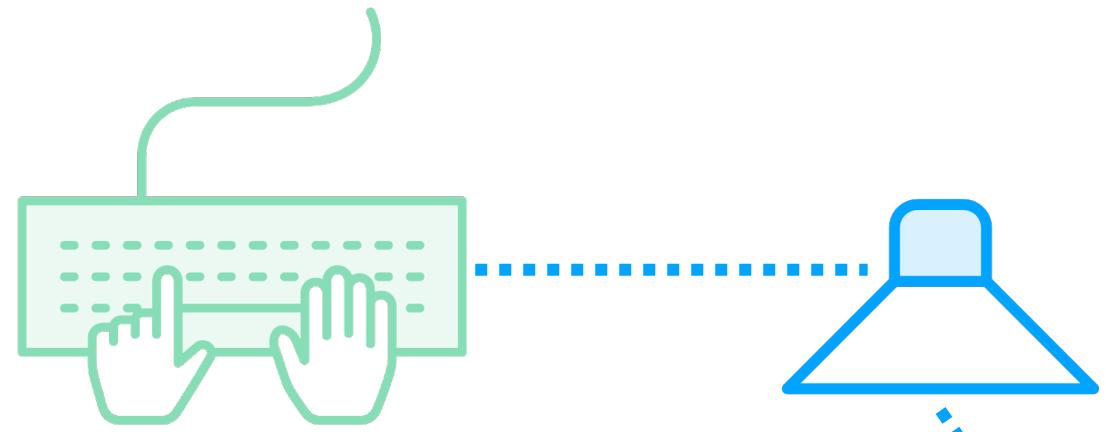
**And at unpredictable
intervals**



Component



Often a stream of events
Not just a single value

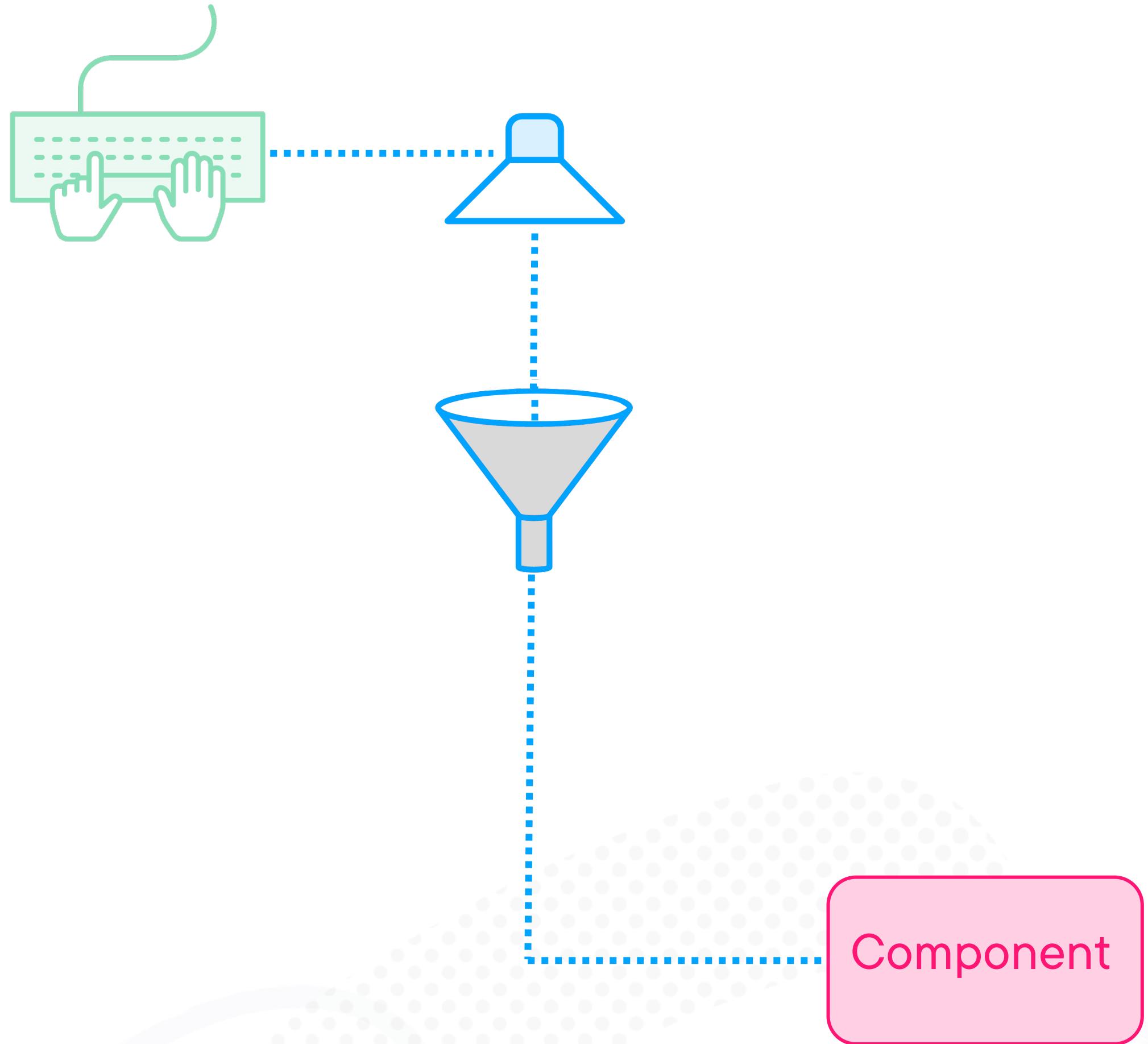


Component

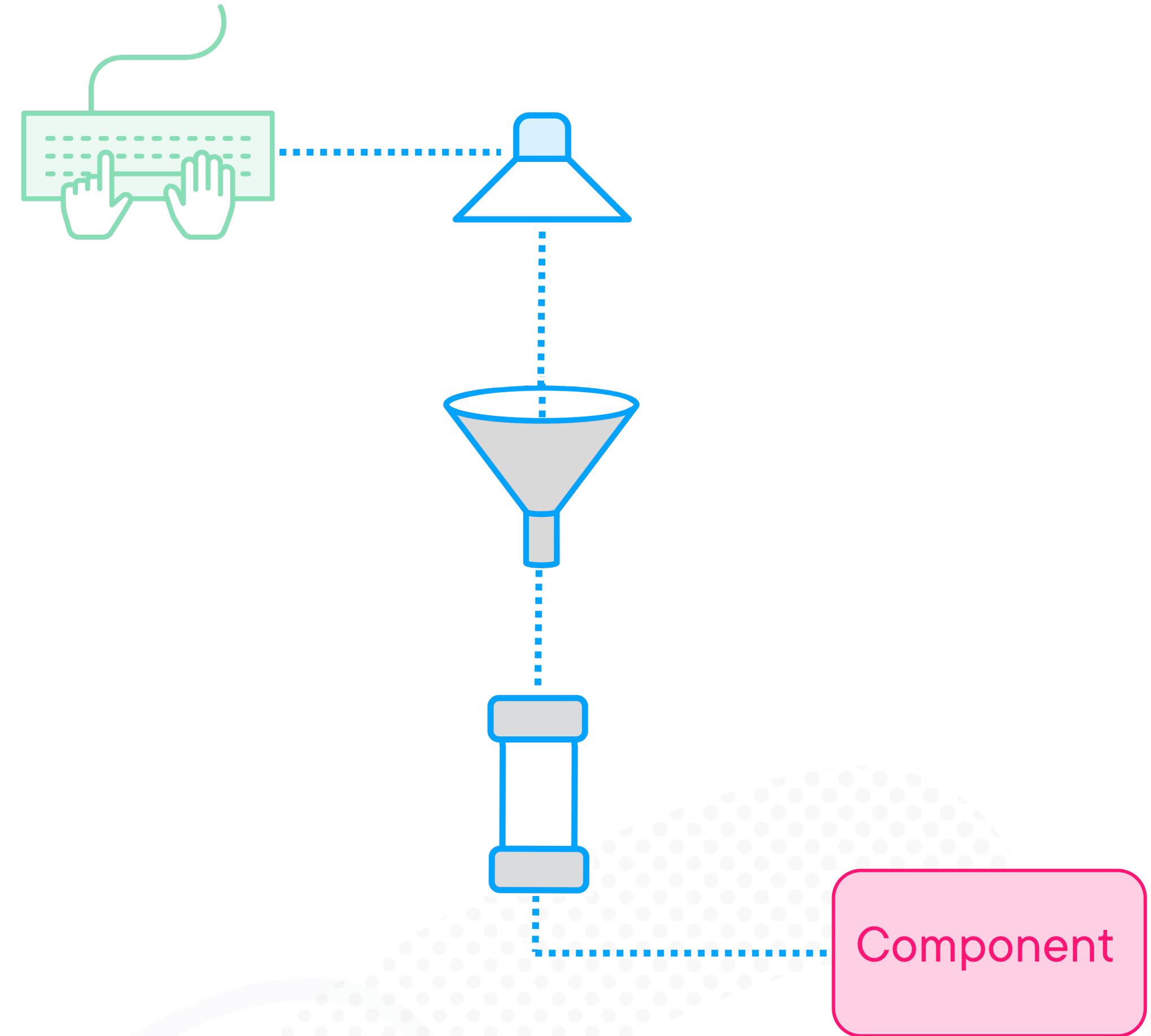


**Filter to only events that
meet specific criteria**

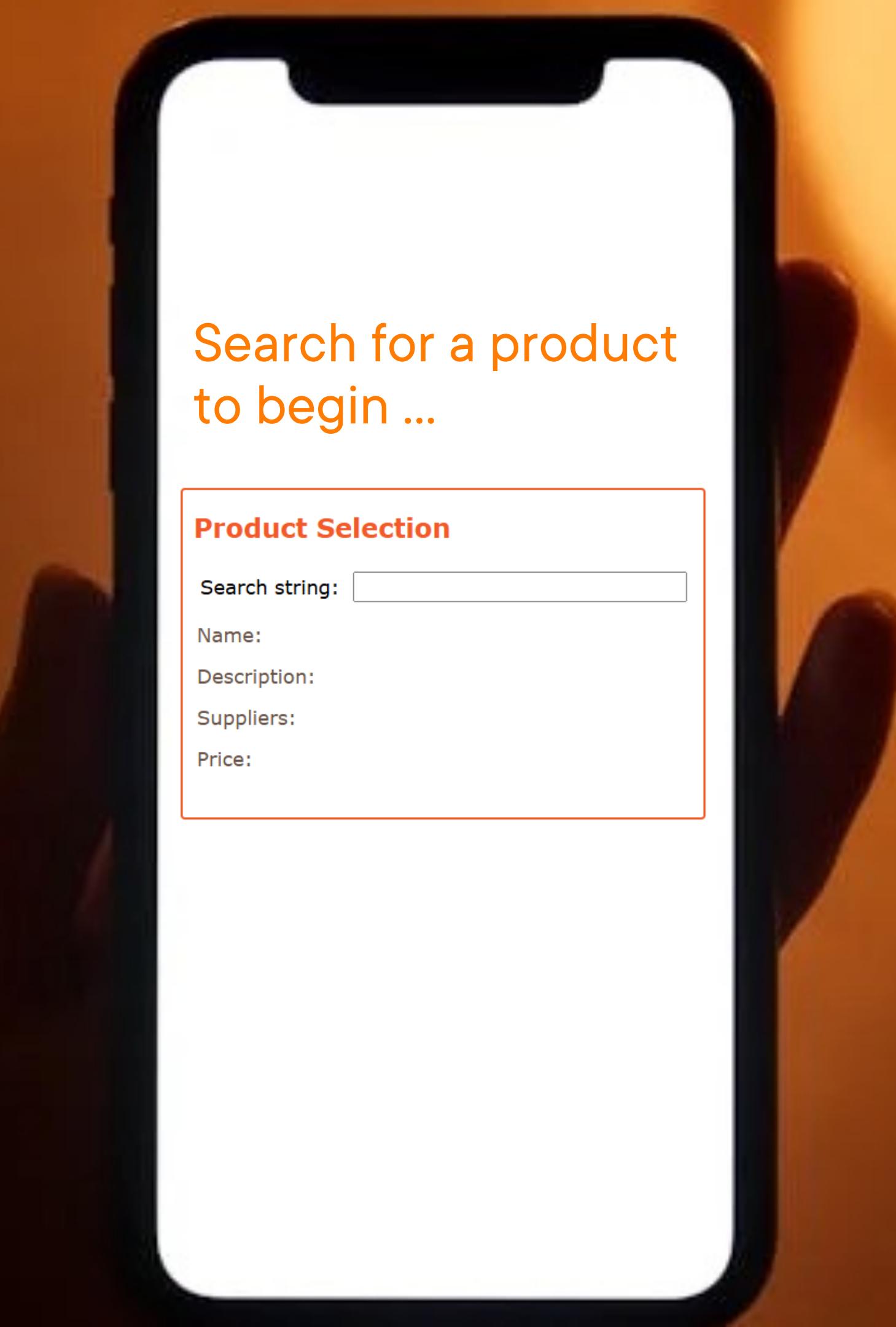
**For example:
specific keys**



**Pipe it through a set of operators
To modify or process the event before emitting it**



Global Keyboard Shortcut



Search for a product
to begin ...

Product Selection

Search string:

Name:

Description:

Suppliers:

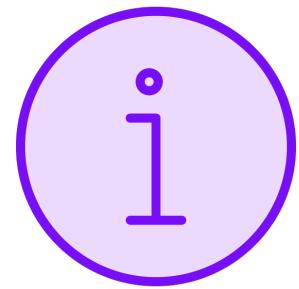
Price:

Press ?, a help message should appear

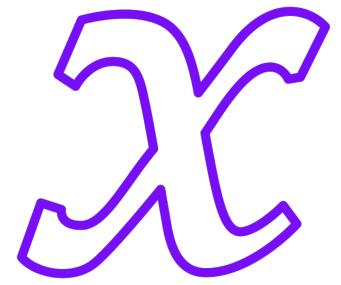
Press ESC, the help message should disappear



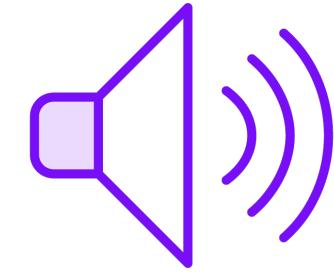
Coding Steps



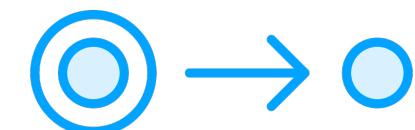
Write help message



Declare state



**Create and subscribe
to an observable**



**Transform event to
key value**



Filter the key values



Show/hide the help



GitHub

<https://github.com/DeborahK/angular-rxjs-ps-course>

Beginning sample application files:

apm-begin

Final (completed) sample application files:

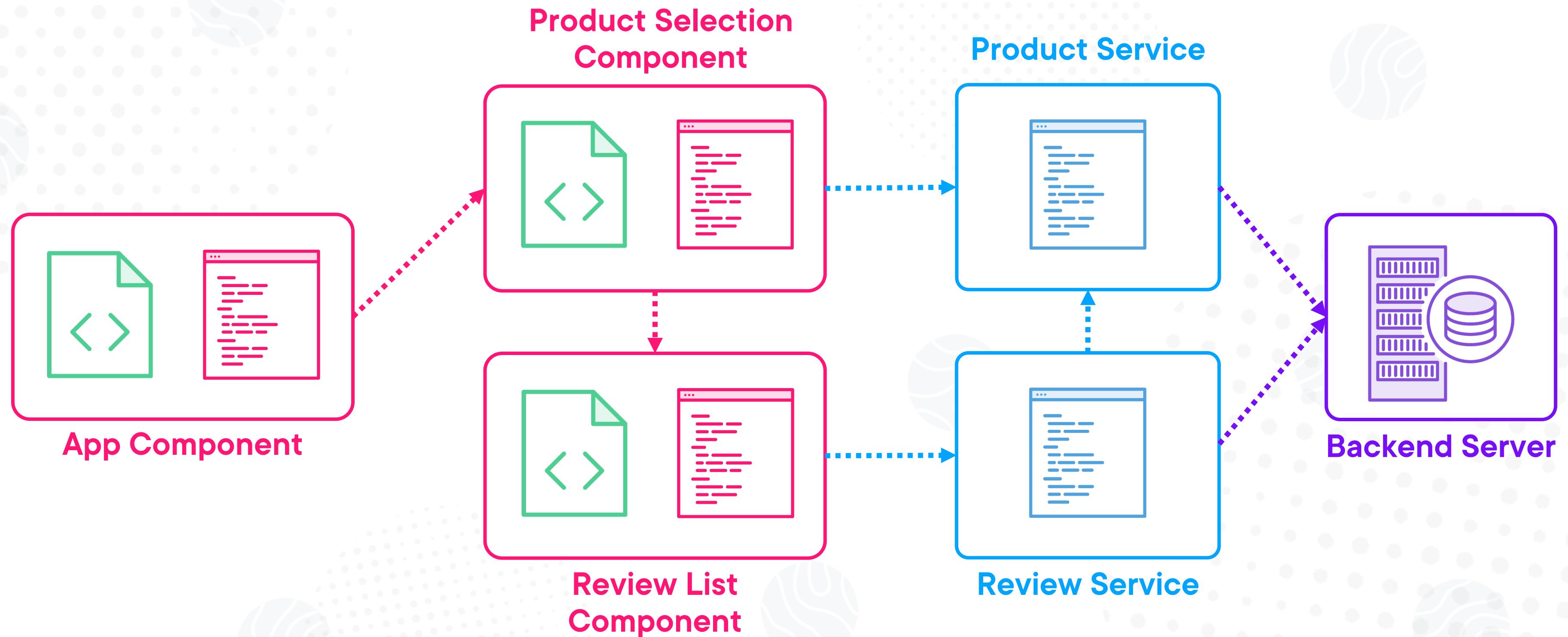
apm-end

File with clickable links to additional information:

MOREINFO.md



Sample Application Structure



Observable Creation Functions

```
const apples$ = of('Apple1', 'Apple2');  
// 'Apple1' 'Apple2'
```

```
const apples$ = from(['Apple1', 'Apple2']);  
// 'Apple1' 'Apple2'
```

```
const clicks$ = fromEvent(document, 'click');  
// PointerEvent {isTrusted: true, pointerId: 1,  
// width: 1, height: 1, pressure: 0, ...}
```

