

# Building The Tomorrow Classroom

[bunsan.io](https://bunsan.io)

#CodeBEAMSTO

# Who are we?



**Pedro  
Hernández**

Software Engineer

@sil mood



**Anayeli  
Malvaez**

Software Engineer

@AnayeliMalvaez

# Bunsan

**Bunsan means  
descentralized**

# How can we take courses?

The start of our journey

**How can we track all the information  
regarding the course?**

**How can we watch it again?**

# A bunch of tools

- They were designed for different purposes
- Moodle is just too much
- You need to use more than one tool
- It could be difficult to improvise





Our virtual classroom



# tich

**Tich** Pedro pedro@lumma.ie

Courses

- + (Add new course)
- Understanding XMPP
- CodeBEAM STO 2019 / Tich Demo  
Building the tomorrow classroom
- BEAM for beginners  
SEAM Sprint

**Phoenix Fundamentals** info sessions

New session

Date	Time	Duration
------	------	----------

**Without students**  
Search for your students by their email or username, add them to your course and we will send them an invite.

**No sessions**  
Schedule a new session to start Tich.

#CodeBEAMSTO

# The main objectives

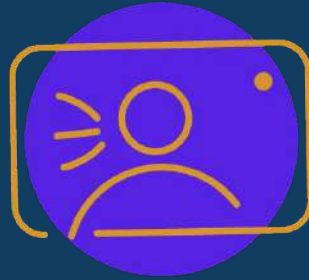
- Real time classes
- Really simple course management
- Always interactive
- Students can explore content during sessions

# The main features



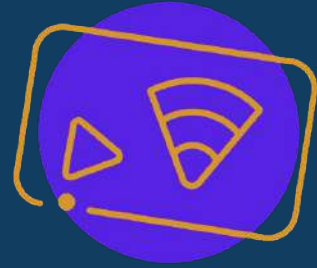
---

Real time whiteboard



---

Videoconferences



---

Slides sharing

# How did we build it?

# Guess what?

# Elixir



#CodeBEAMSTO



Swift



Vue.js



RabbitMQ



**JANUS**  
WebRTC Gateway





iPad  
app

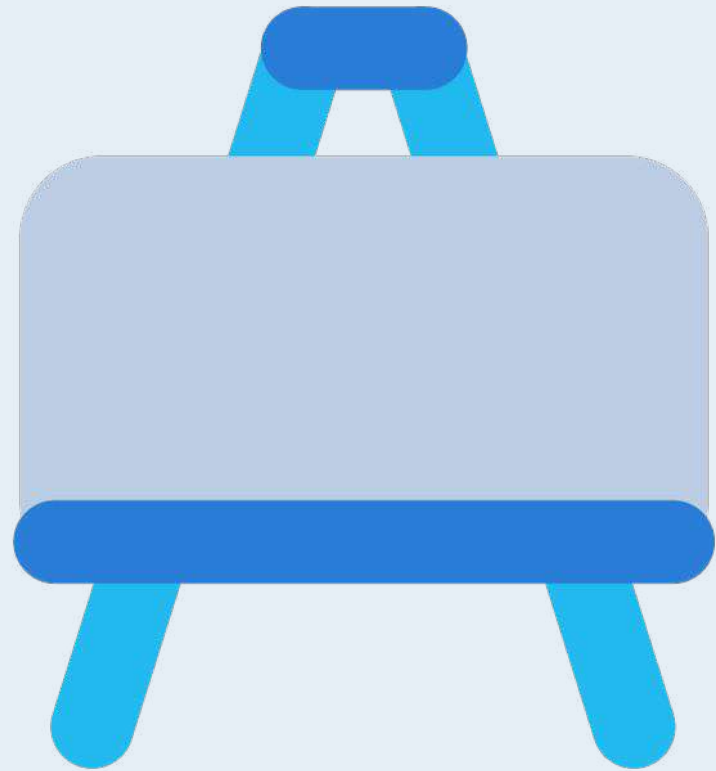
Web Virtual  
Classroom

Admin

# Tech challenges

#CodeBEAMSTO

# Whiteboard



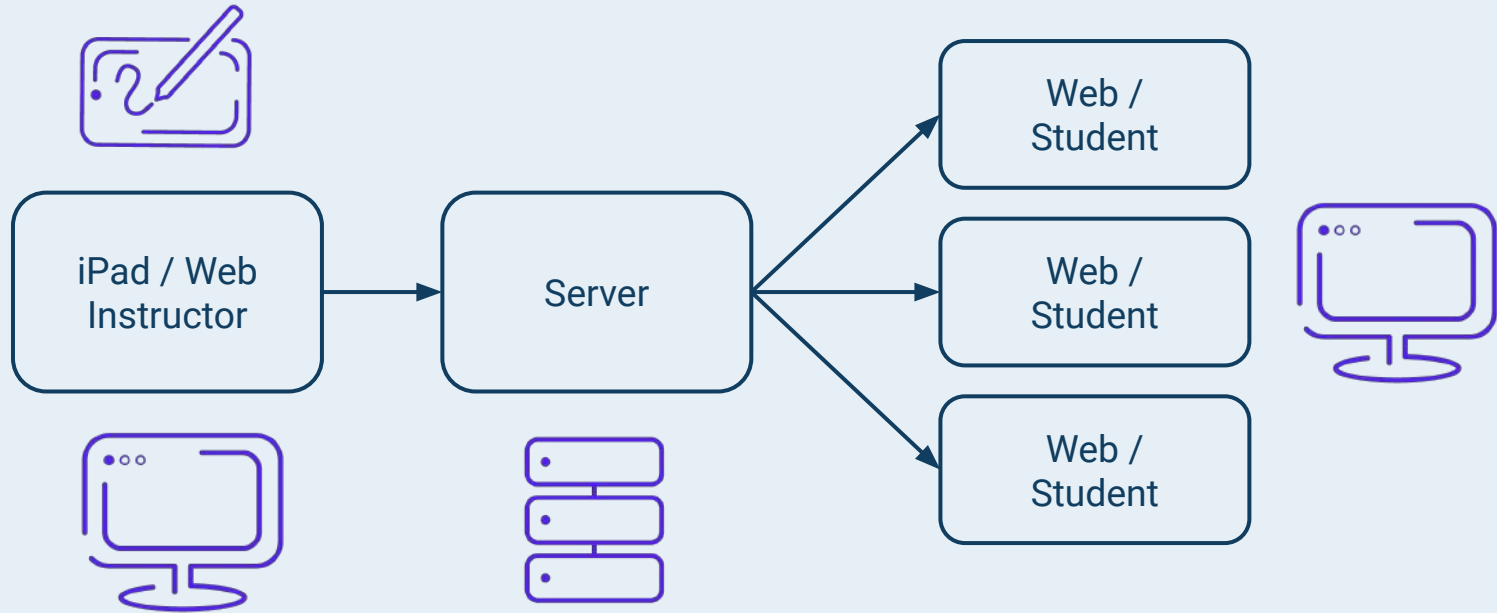
#CodeBEAMSTO

# Apple Pencil

Why?



# Broadcasting a draw



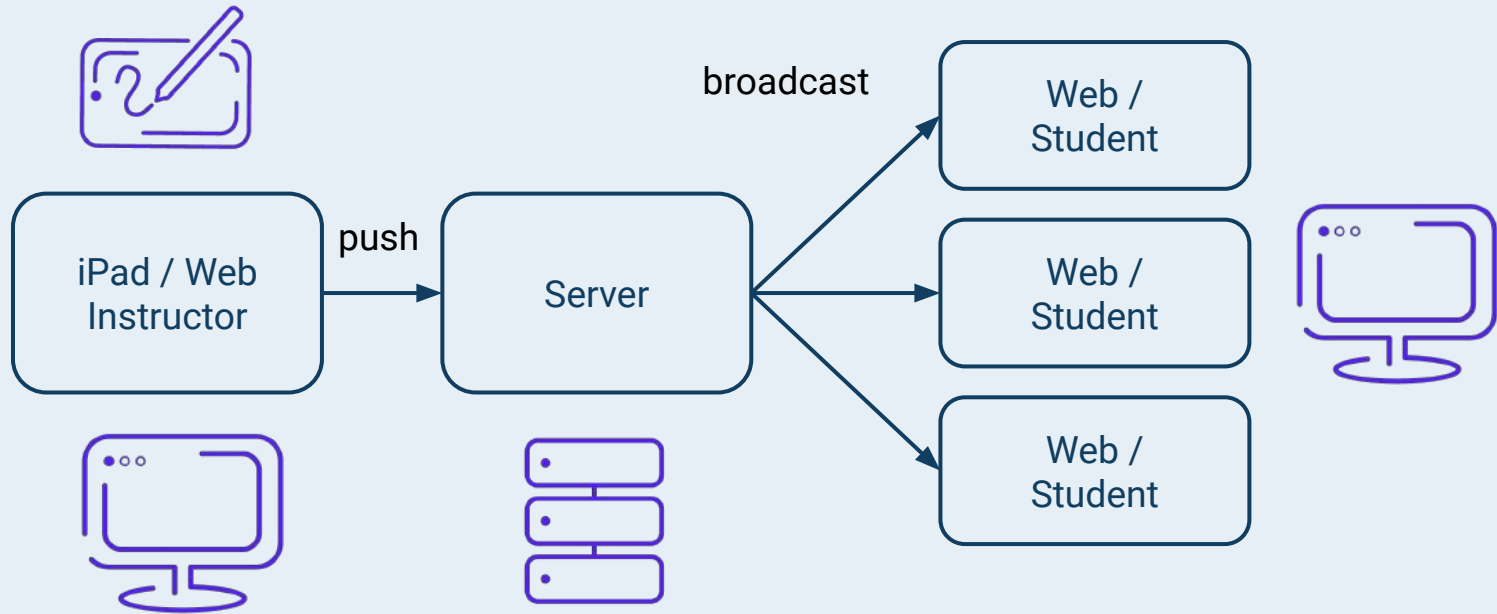
# Phoenix Channels

#CodeBEAMSTO



```
1 defmodule MyApp.BoardChannel do
2   use Phoenix.Channel
3
4
5   @spec join(String.t(), any(), Phoenix.Socket.t()) :: {:ok, Phoenix.Socket.t()}
6   def join("board:" <> _board_id, _params, socket) do
7     send(self(), :after_join)
8     {:ok, socket}
9   end
10
11   @spec handle_in(String.t(), map(), Phoenix.Socket.t()) :: {:noreply, Phoenix.Socket.t()}
12   def handle_in("board:new_page", payload, socket) do
13     broadcast!(socket, "board:new_page", payload)
14     {:noreply, socket}
15   end
```

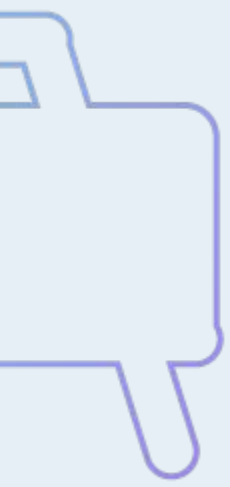
# Broadcasting a draw





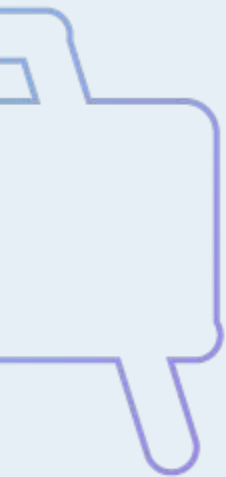
# Demo

Let's draw!



```
{
  user_id: 1
  path: {
    d: "L135.671875 141.34375,L134.671875 142.34375,L130.671875
      146.34375,L123.671875 151.34375,L114.671875 155.34375,L106.671875
      159.34375,L99.671875 161.34375,L93.671875 162.34375,L89.671875
      163.34375,L86.671875 164.34375",
    id: "f8a057ac-93c4-4ade-b5ac-f025ab1c4173",
    opacity: 1,
    color: "#585858",
    width: 2
  }
}
```

**Now you can draw!**

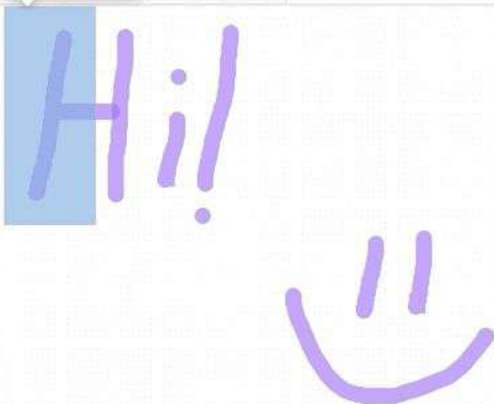


#CodeBEAMSTO

**How can we store a lot of paths?**



path 21 x 116



1/1



```
1094970703125,1072.9439697265625C1559
.1094970703125,1072.9439697265625,155
9.1094970703125,1072.9439697265625,15
59.1094970703125,1072.9439697265625L1
559.1094970703125,1072.9439697265625"
></path> == $0
```

```
<path fill="none" stroke-linecap=
"round" stroke="#a269ff" stroke-
width="27" stroke-opacity="0.7" d=
"M1707.6138916015625,813.615417480468
8L1707.6138916015625,813.615417480468
8C1707.6138916015625,813.615417480468
8,1707.6138916015625,813.615417480468
8,1707.6138916015625,813.615417480468
8C1707.6138916015625,813.615417480468
8,1707.6138916015625,813.615417480468
8,1707.6138916015625,813.984832763671
9C1707.6138916015625,814.354248046875
```

... div div div #board #svg g #canvas g path

Styles Computed Event Listeners DOM Breakpoints >>

Filter

:hov .cls +

```
element.style {
}
```

path[Attributes Style] {

```
fill: none;
stroke-linecap: round;
stroke: ■ rgb(162, 105, 255);
stroke-width: 27;
stroke-opacity: 0.7;
```

```
d: path("M 1605.66 815.832 L 1605.66 816.201 C 1605.66
816.571 1605.66 817.31 1605.66 817.679 C 1605.66
```

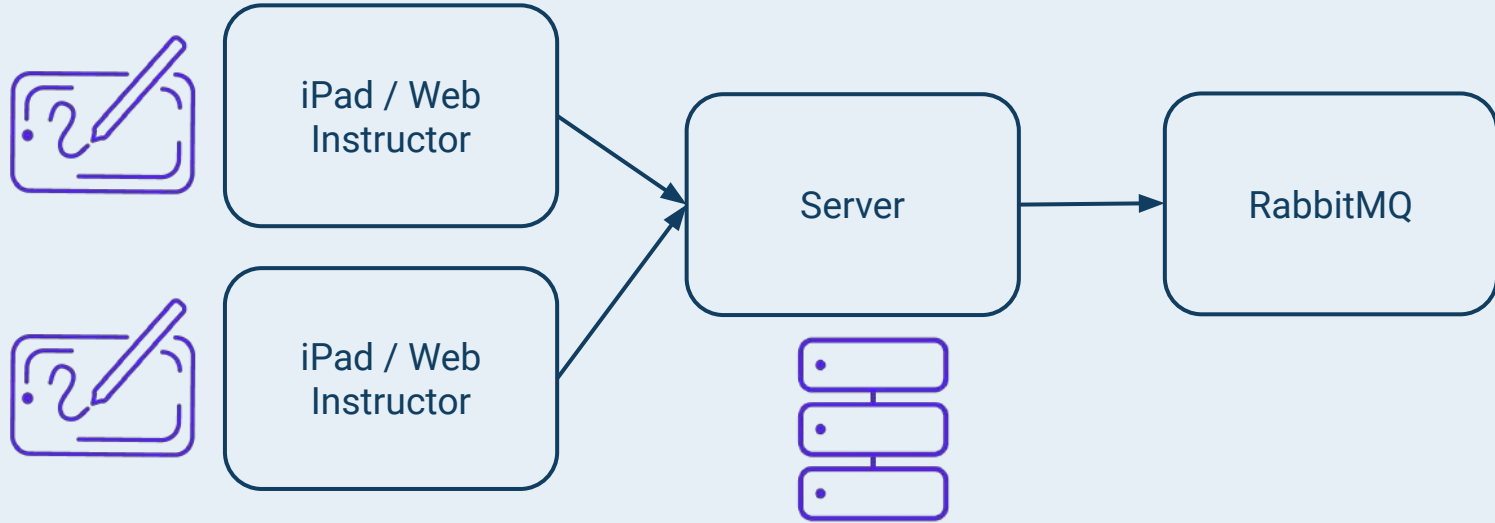
**This happens when we  
just insert the paths**



#CodeBEAMSTO



# Storing multiples draws

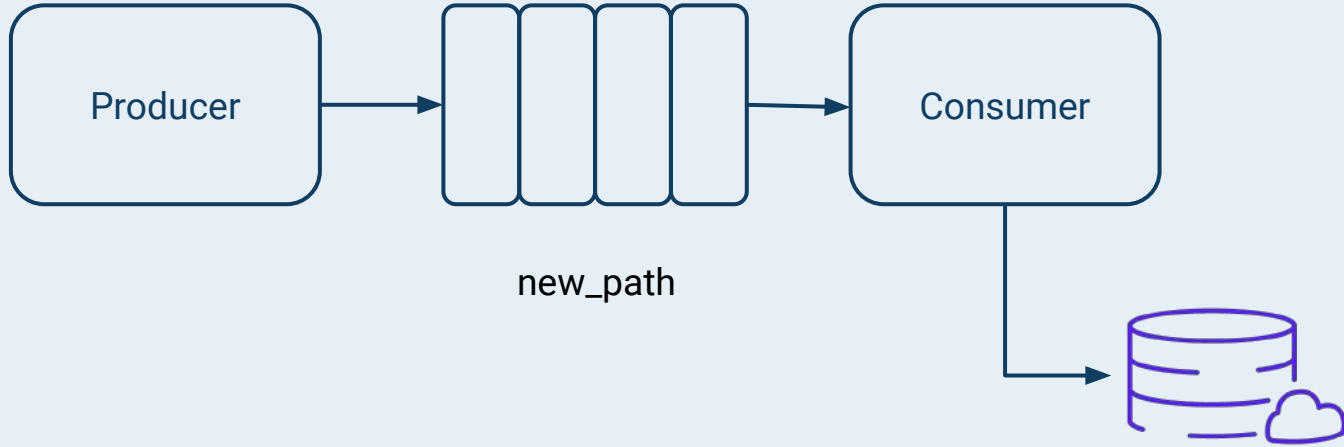


# Integration with Elixir



```
defp deps() do
  [
    {:amqp, "~> 1.0"},
  ]
end
```

RabbitMQ





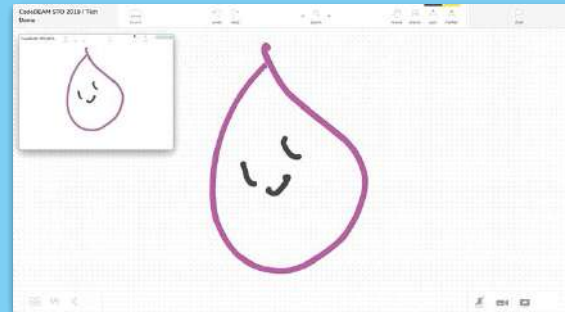
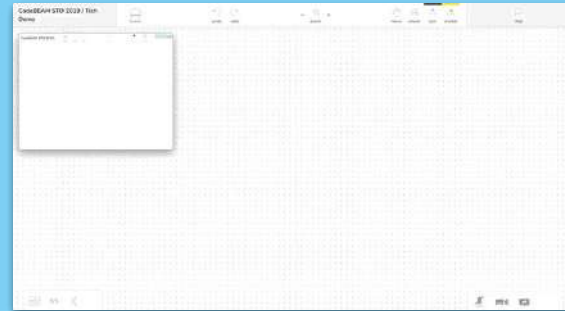
#CodeBEAMSTO

# Restored!



# More interactions

- undo/redo
- move canvas
- zoom canvas

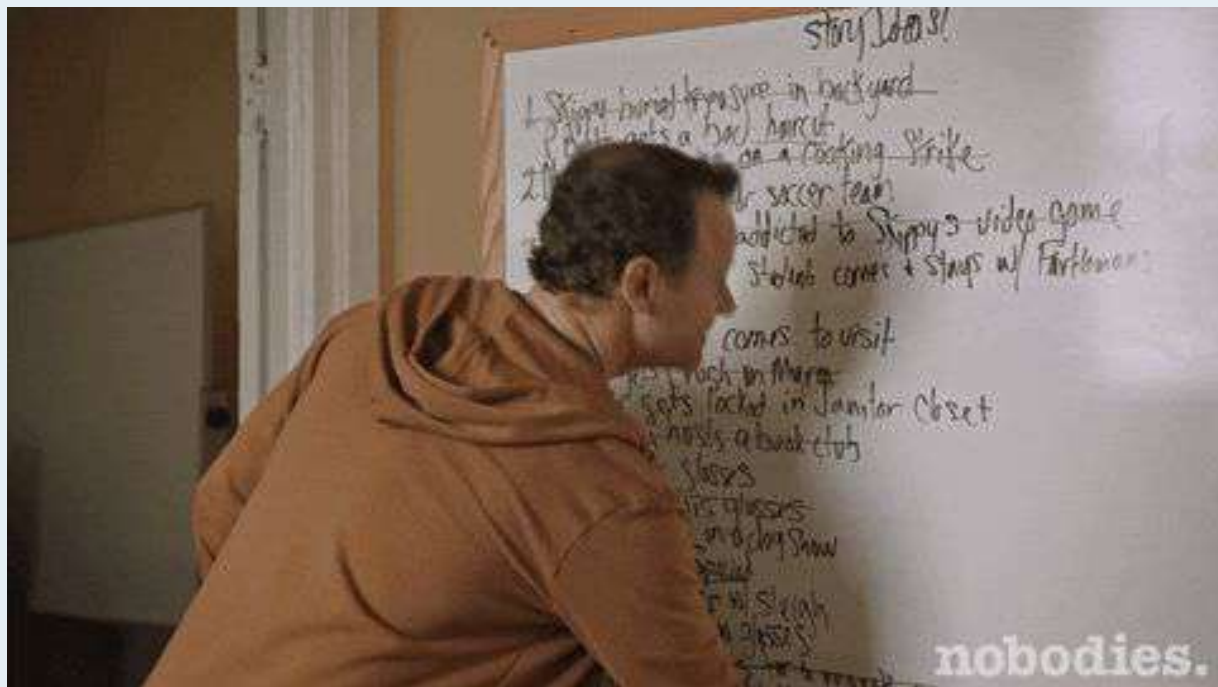




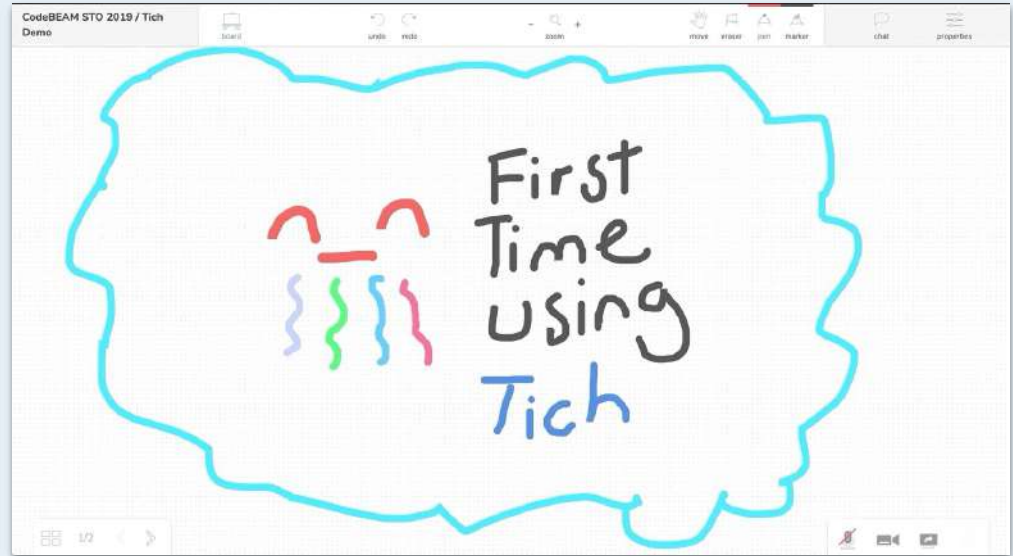
#CodeBEAMSTO

**What happened when  
you need more space?**





# Multiple pages



# Using multiple channels

## Board Channel

- `page:new`
- `page:delete`
- `page:set`

## Page Channel

- `path:web`
- `path:ipad`
- `path:delete`
- `path:restore`
- `canvas:drag`

# Sync board process

## Board Channel

join  
board:id

board:restore

setupLastPage

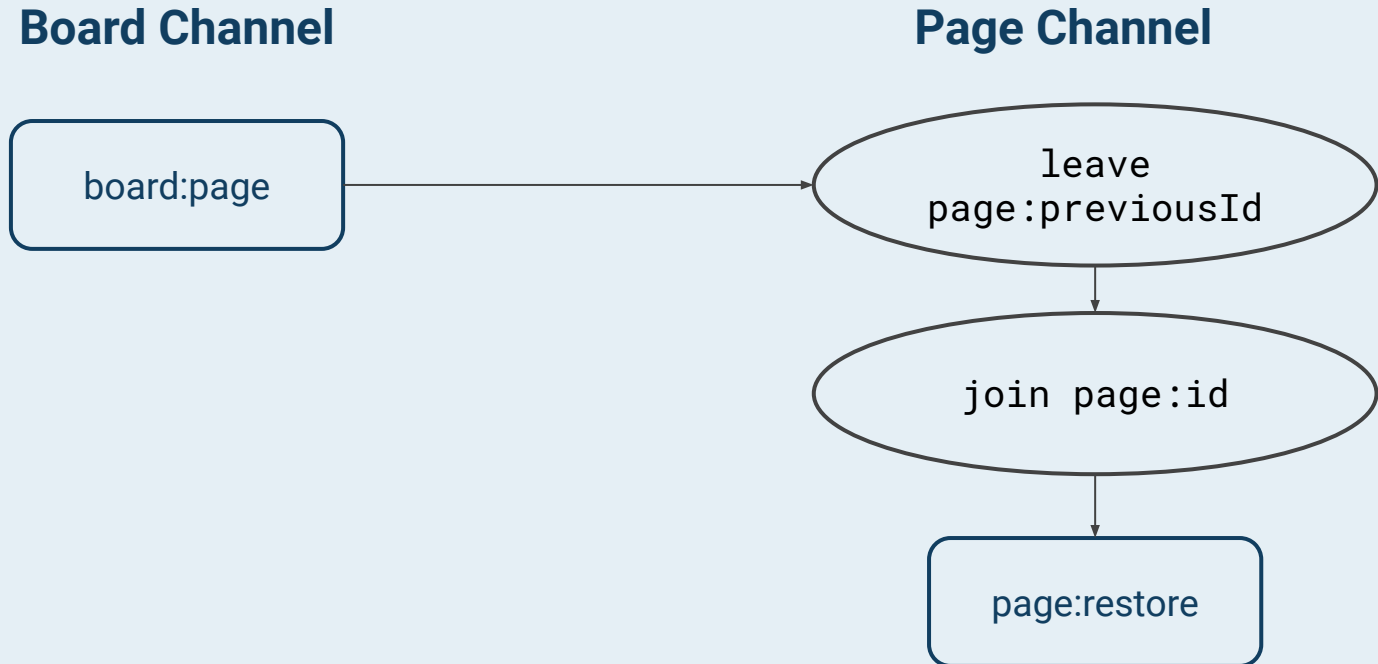
## Page Channel

join page:id

page:restore



# Change page process

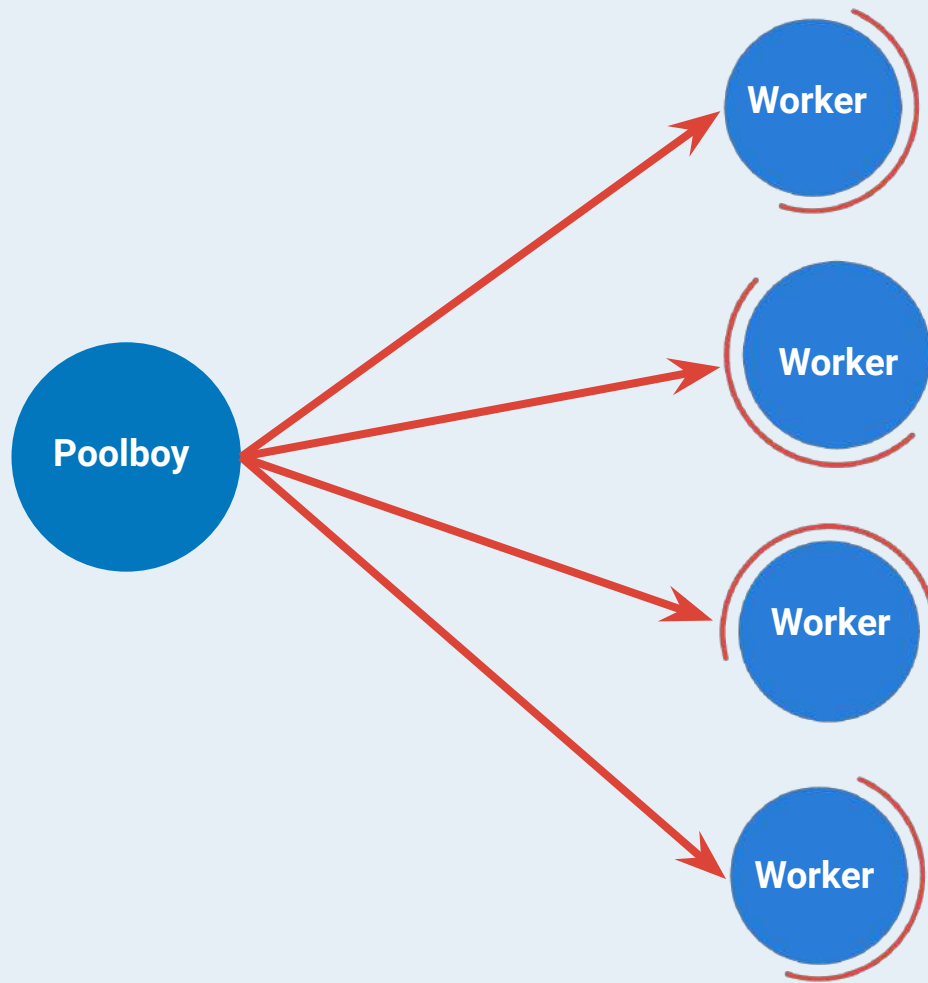
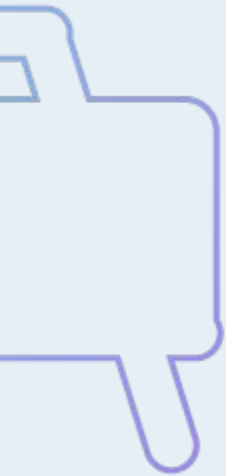


# Avoid clients validations

# Generating thumbnails



```
def thumbnail_gen(page_id) do
  page_id
  |> get_all_paths
  |> create_the_svg
  |> convert_in_png
  |> resize
  |> upload to S3
end
```



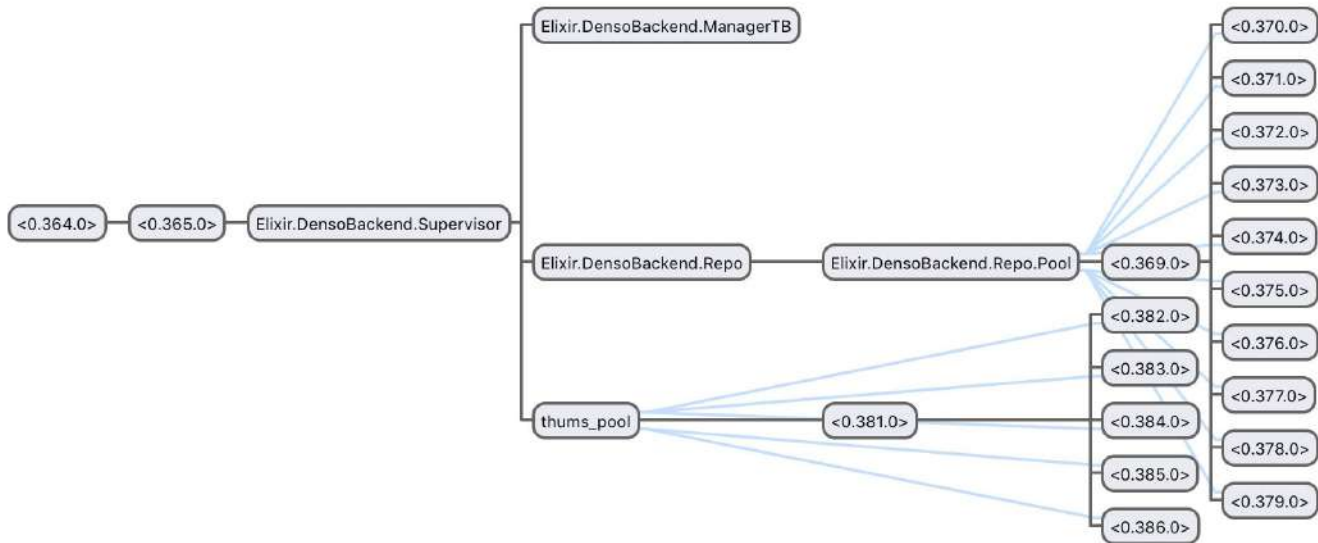




```
1 defp poolboy_config do
2   pool_options = [
3     name: {:local, :thumbs_pool},
4     worker_module: MyApp.GenerateThumbnails,
5     size: 5,
6     max_overflow: 10
7   ]
8
9   :poolboy.child_spec(:thumbs_pool, pool_options, [])
10 end
```

#

- amqp\_client
- briefly
- cowboy
- db\_connection
- denso\_backend**
- denso\_web
- ecto
- elixir
- ex\_aws
- gettext
- hackney
- hex
- iex
- inets
- jose
- kernel
- logger
- mix
- phoenix
- phoenix\_ecto
- phoenix\_live\_reload
- phoenix\_pubsub
- plug
- postgrex
- ranch
- runtime\_tools
- ssl
- timex
- tzdata



# Slides Sharing

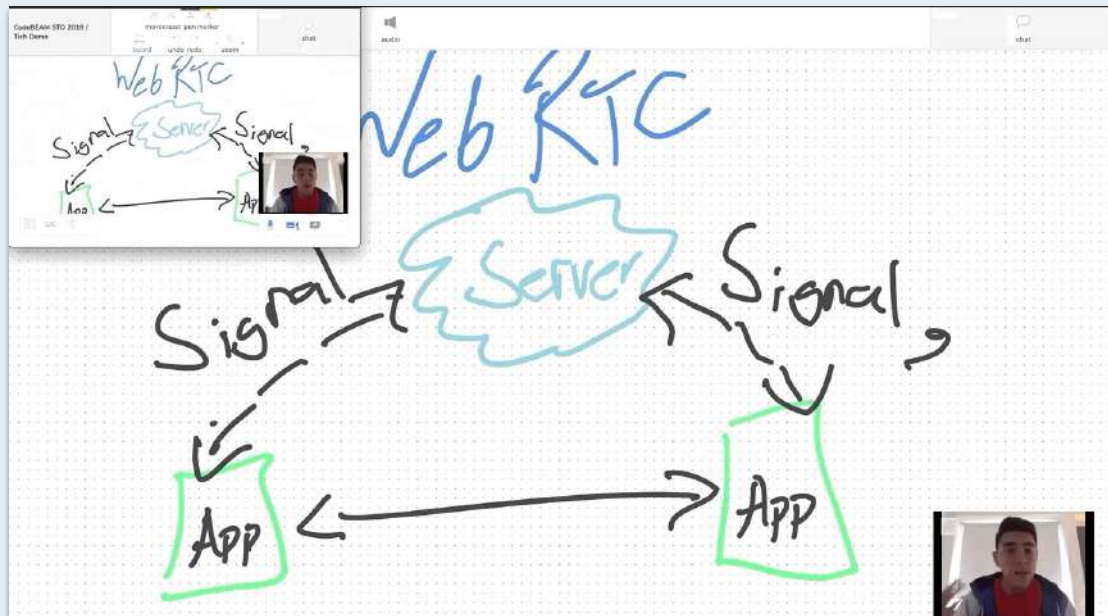


#CodeBEAMSTO

# Slides and content events

- `slides:url`
- `slides:page`
- `content:set`

# Video Conferences

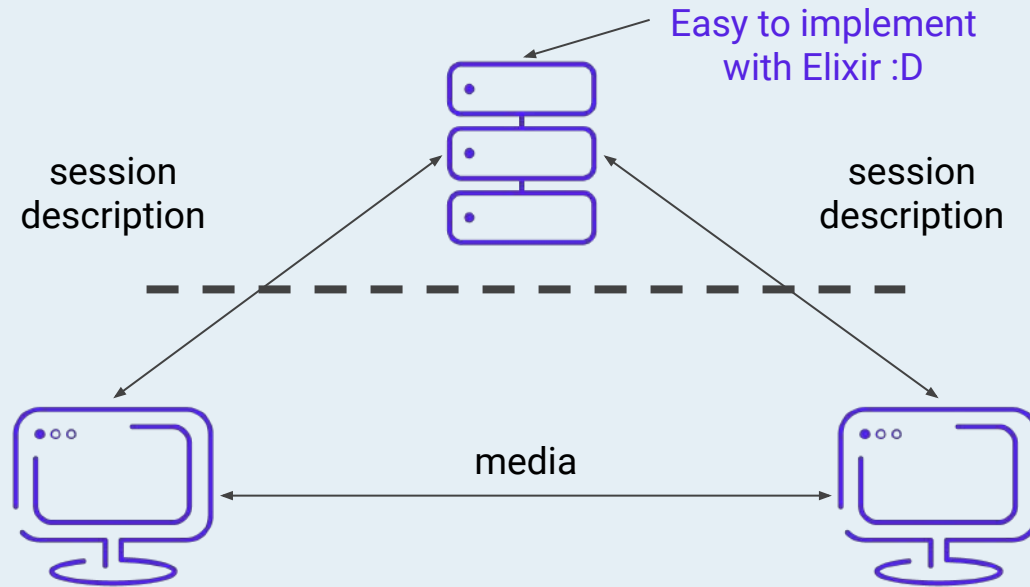


# Go WebRTC

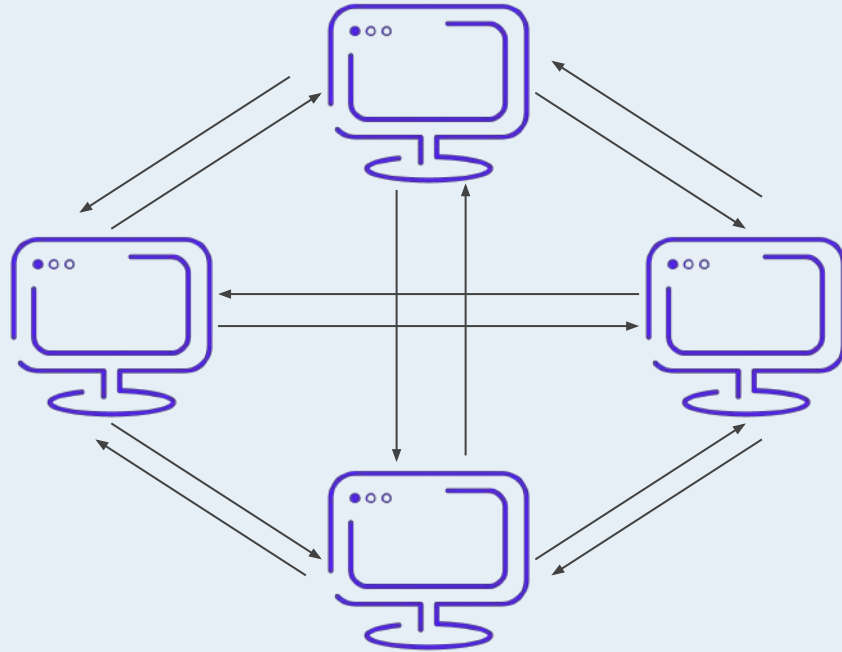
- Real Time Communications
- Peer to peer
- Signaling



# Signaling server / JSEP

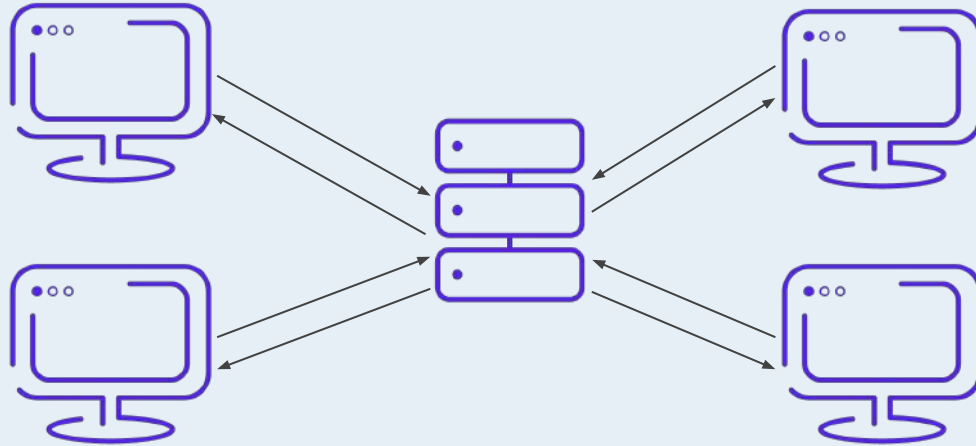


# Peer to peer won't scale :(





# Use a media server

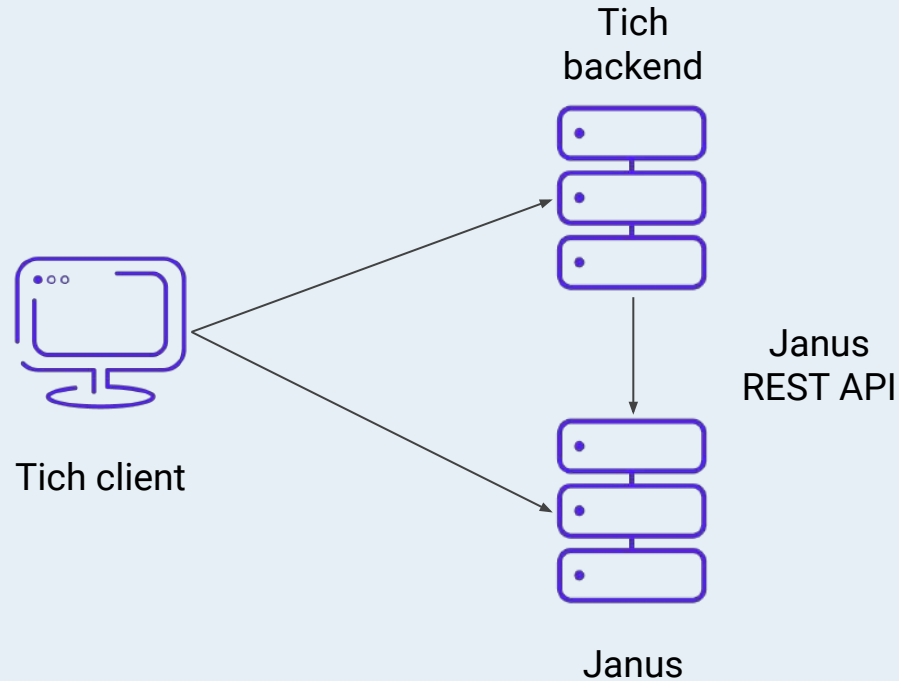


# Janus Media Server

- SFU & MCU
- Janus Plugins
- REST API

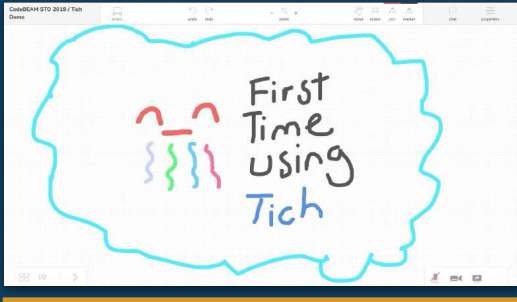


# Tich Setup

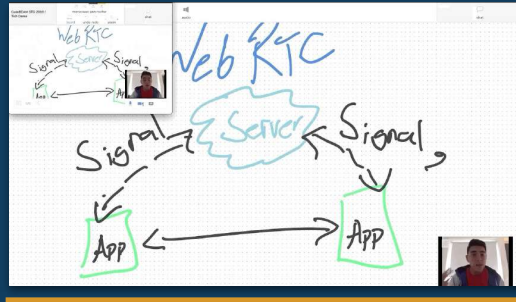




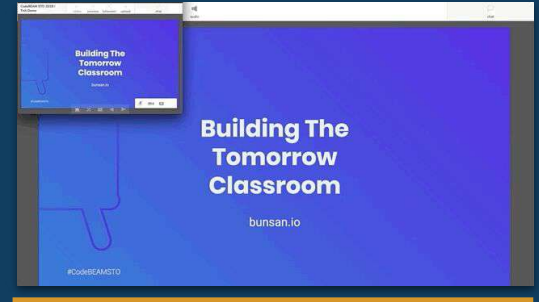
# is alive!



Whiteboard



Videoconference



Slides sharing



#CodeBEAMSTO



Are you interested in Tich?

**tich@bunsan.io**



# Building The Tomorrow Classroom

[bunsan.io](https://bunsan.io)

#CodeBEAMSTO