

/react fait du slack

#project-alpha

☆ | 8 14 | 8 1 | Project Alpha! 🚀

March 13, 2019

Harry Boone 10:55 AM
Hey all - seeing reports in [#alerts-android](#) of multiple 404s from the settings panel.
Can someone look into it?

Arcadio Buendía 10:59 AM
Logging in Jira now

Only visible to you

Jira Cloud APP 5:20 PM
[Create Jira issue](#)

Large list of channels causes front end view to break

Channel picker is breaking the javascript in workspace with 1000+ channels

Jira project Front End (FE)

Issue type Bug

Create **Cancel**

/jira create

ft-care-alerts

https://manomano.atlassian.net/wiki/spaces/CFT/pages/3938419891/order-supervisord+might+be+in+**FATAL**+status

@slack-ft-care-alerts

Less than 1 log events matched in the last 5m against the monitored query:

||@project:order+message env:prd source:supervisord gave up entered **FATAL** status.
too many start retries too quickly (@region:eu-west-3)

Notified

@slack-ft-care-alerts

View in Log Explorer

Triggered: [PRD] (eu-west-3) [ft-css-ops] order-message supervisord number of processes is too low

There are unhealthy processes, which can lead to a degradation or loss of service.

Restart the instances following the instructions below.

See: [https://manomano.atlassian.net/wiki/spaces/CFT/pages/3938419891/order-supervisord+might+be+in+**FATAL**+status](https://manomano.atlassian.net/wiki/spaces/CFT/pages/3938419891/order-supervisord+might+be+in+FATAL+status)

@slack-ft-care-alerts

supervisord.process.count over autoscaling_groupprd-ms-order-message-applications.workers.status.up was < 6.5 on average during the last 15m.

Metric value: 6.484 (15 kB) •

Notified

@slack-ft-care-alerts

Mute Monitor Declare Incident

DataDog 8:05 PM

B I S E M D

Message #ft-care-alerts

The screenshot shows the Slack API developer documentation website. At the top, there's a navigation bar with the Slack logo, a search bar, and links for 'Docs', 'Tutorials', and 'Your apps'. On the left, a sidebar lists several sections: 'Start here: platform overview' (selected), 'Next generation Slack platform', 'Getting started', 'Automation', 'Slack CLI', and 'Deno & TypeScript'. Below this, there are two main columns: 'The next-generation Slack platform' (with 'Functions' under it) and 'Next-generation developer tools' (with 'Slack CLI' under it). A large central section is titled 'Developer docs and guides' and contains the heading 'Examples'. Under 'Examples', there's a code block showing JSON for a 'section' block:

```
1 {  
2   "type": "section",  
3   "text": {  
4     "type": "mrkdwn",  
5     "text": "A message *with some bold text* and _some italicized text_."  
6   }  
7 }
```

A 'Copy' button is located in the top right corner of this code block. Below the code block is a button labeled 'View this block in Block Kit Builder'. The bottom of the page features a footer with sections for 'Workflows for developers', 'App foundations', and 'Communication'.

Developer docs and guides

The next-generation Slack platform

Next-generation developer tools

Examples

A simple text section block:

```
1 {  
2   "type": "section",  
3   "text": {  
4     "type": "mrkdwn",  
5     "text": "A message *with some bold text* and _some italicized text_."  
6   }  
7 }
```

View this block in Block Kit Builder

Workflows for developers

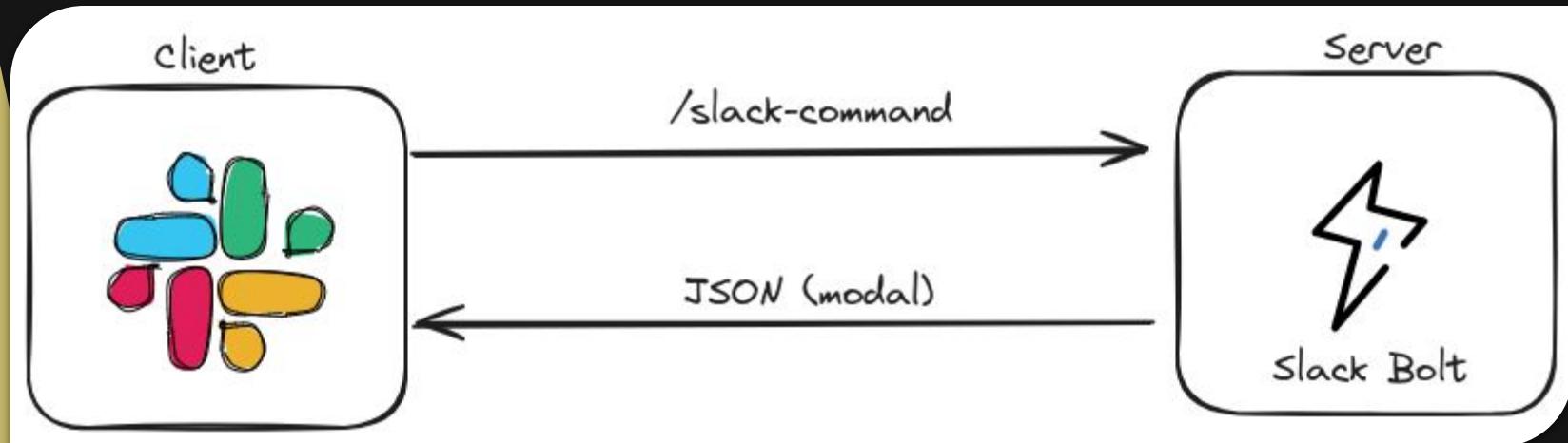
- Reference
- Administration
- Enterprise
- Apps for Admins

App foundations

- Installation and permissions
- Slack APIs

Communication

- Sending and viewing messages
- Files API



```
const callback_id = "view_1";
app.command("/define-word", async ({ ack, client, body }) => {
| await ack();

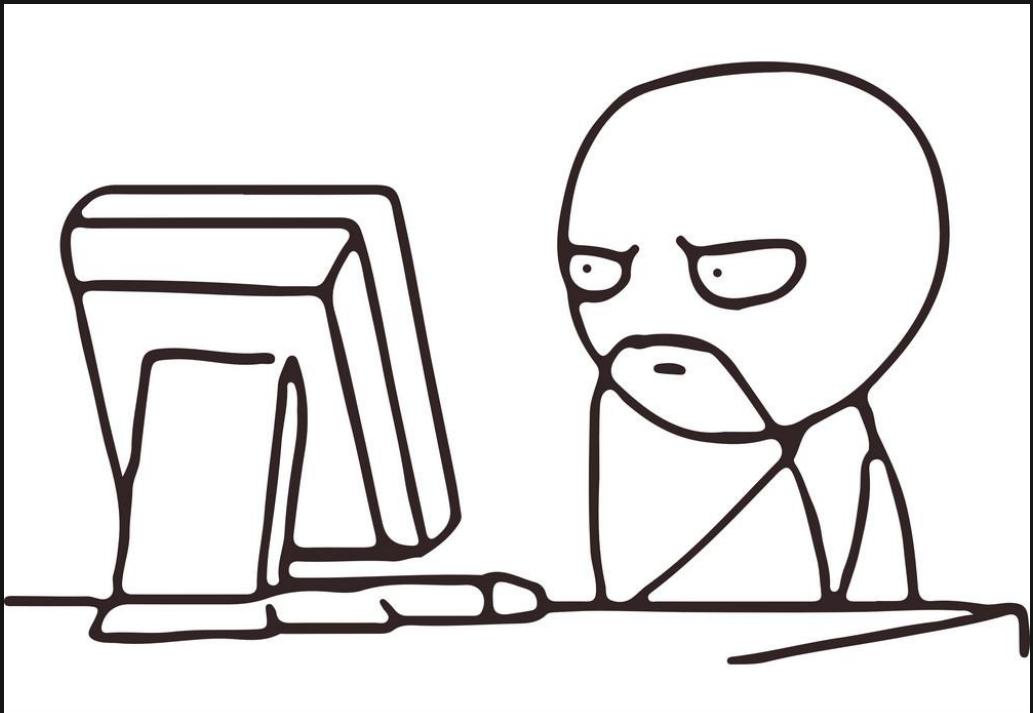
await client.views.open({
  trigger_id: body.trigger_id,
  view: [
    type: "modal",
    callback_id,
    title: { type: "plain_text", text: "Vanilla Slack Modal" },
    submit: { type: "plain_text", text: "Submit" },
    blocks: [
      {
        type: "image",
        image_url: "https://images.unsplash.com/photo-1571934811356-5c",
        alt_text: "Tea time",
        title: {
          type: "plain_text",
          text: "Tea time",
        },
      },
      {
        type: "section",
        text: {
          type: "mrkdwn",
          text: "Type below *the word* you want to _define_.",
        },
      },
      {
        type: "input",
        block_id: "my_prompt",
        label: {
          type: "plain_text",
          text: "Word",
        },
        element: {
          type: "plain_text_input",
          action_id: "my_input",
        },
      },
    ],
  ],
});
```

```
app.view(callback_id) async ({ ack, view, client, body }) => {
  await ack();
  const {
    values: {
      my_prompt: { my_input: myInput },
    },
  } = view.state;
  await client.chat.postMessage({
    channel: body.user.id,
    text: `I am processing your request... :hourglass:`,
  });

  const definition = await defineWord(myInput.value);

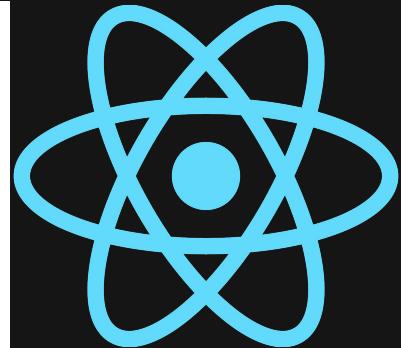
  await client.chat.postMessage({
    channel: body.user.id,
    text: `${myInput.value}: ${definition}`,
  });
};
```

1 You, 4 weeks ago • feat(vanilla): add app using plain slack



VectorStock®

VectorStock.com/11495678



Pascal Fong Kye

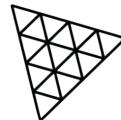
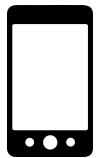
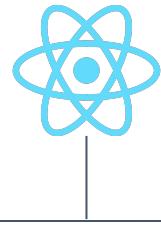


Développeur Web
chez ManoMano ~ 3 ans
 pfongkye

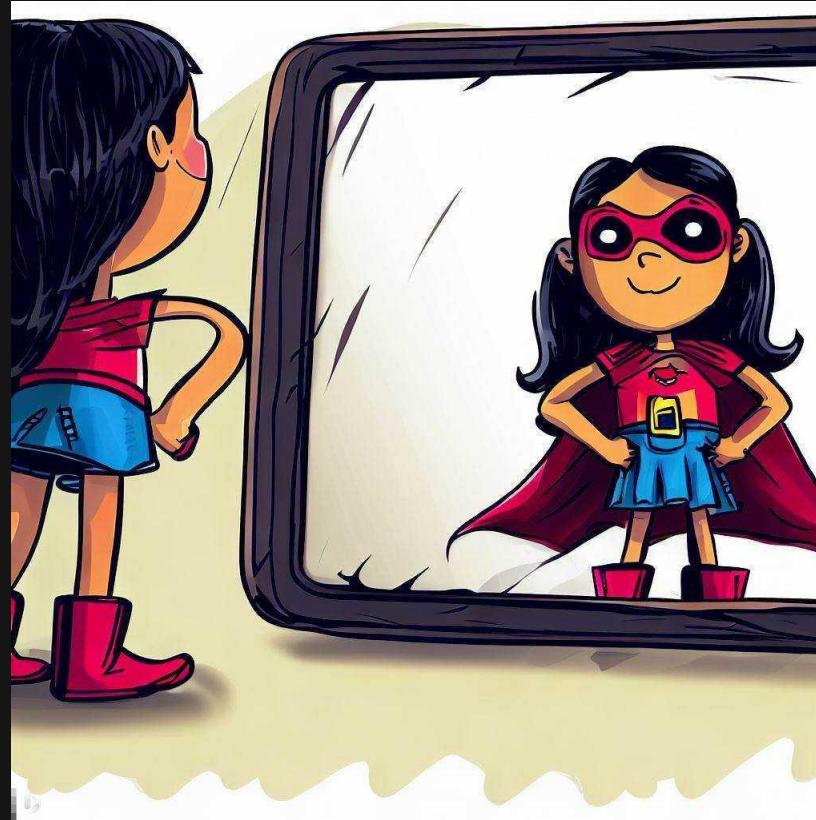
React



images unsplash



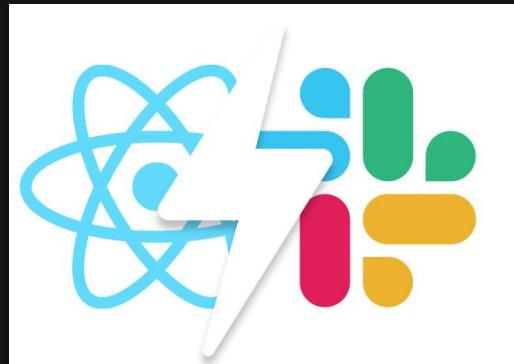
Moteur de rendu



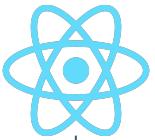
IA générative

```
<div>
  <h1>Bienvenue sur mon site !</h1>
</div>
```

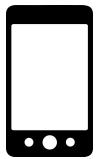
JSX



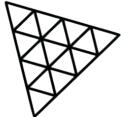
**React
Reconciler
Slack Bolt**



react-dom



react-native



react-three-fiber



react-pdf



react-slack

```
{  
  "type": "section",  
  "text": {  
    "type": "mrkdwn",  
    "text": "A message *with some bold text* and _some italicized text_.  
  }  
}
```

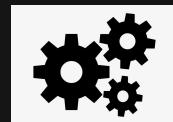
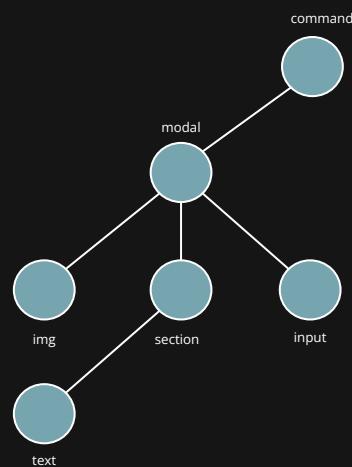
```
<section>  
| <text markdown>A message *with some bold text* and _some italicized text_.</text>  
</section>
```

- Syntaxe proche de React
- Le projet React-slack-renderer ne convenait pas
- Principe de fonctionnement de react-dom

DE
MO

La Transformation

```
<command value="/react-slack" ack>
  <modal title="Simple Modal" onSubmit={handleSubmit}>
    <img
      title="Coffee time"
      url={url}
    />
    <section>
      <text markdown>
        | Type below *the word* you want to _define_.
      </text>
      You, now + uncommitted changes
    </section>
    <input label="Word" actionId="my_input" blockId="my_prompt" />
  </modal>
</command>
```



react-reconciler

```
await client.views.open({
  trigger_id: body.trigger_id,
  view: {
    type: "modal",
    callback_id,
    title: { type: "plain_text", text: "Simple Modal" },
    submit: { type: "plain_text", text: "Submit" },
    blocks: [
      {
        type: "image",
        image_url: url,
        alt_text: "Coffee time",
        title: { ... }
      },
      {
        type: "section",
        text: { ... }
      },
      {
        type: "input",
        block_id: "my_prompt",
        label: { ... },
        element: { ... }
      }
    ],
  }
});
```

```
app.command("/define-word", async ({ ack, client, body }) => {
  await ack();

  await client.views.open({
    trigger_id: body.trigger_id,
    view: {
      type: "modal",
      callback_id,
      title: { type: "plain_text", text: "Vanilla Slack Modal" },
      submit: { type: "plain_text", text: "Submit" },
      blocks: [
        {
          type: "image",
          image_url: "https://images.unsplash.com/photo-1571934811356-5cc061b6821f",
          alt_text: "Tea time",
          title: {
            type: "plain_text",
            text: "Tea time",
          },
        },
        {
          type: "section",
          text: {
            type: "mrkdwn",
            text: "Type below *the word* you want to _define_.",
          },
        },
        {
          type: "input",
          block_id: "my_prompt",
          label: {
            type: "plain_text",
            text: "Word",
          },
          element: {
            type: "plain_text_input",
            action_id: "my_input",
          },
        },
      ],
    });
  });

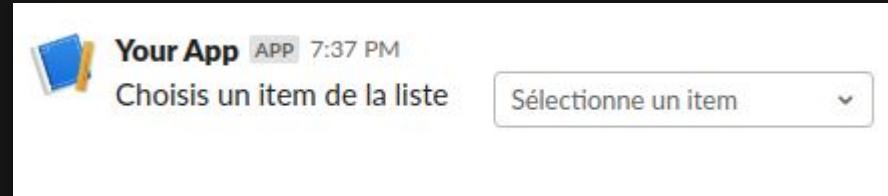
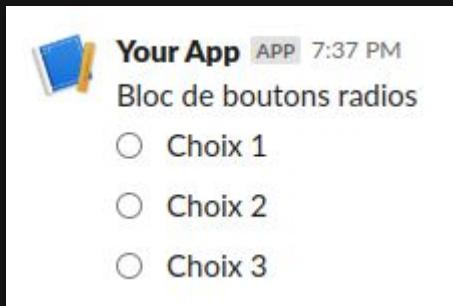
  app.view(callback_id, async ({ ack, view, client, body }) => {
    });
});
```

```
async function handleSubmit({ ack, view, client, body }) {
}

reactSlack.render(
  <command value="/define-word" ack>
    <modal title="React Modal" onSubmit={handleSubmit}>
      <img
        title="My Dictionary"
        url="https://nextjs-blog-pfongkye.vercel.app/images/dictionary_lookup.jpg"
      />
      <MySectionWithText>
        Type below *the word* you want to _define_.
      </MySectionWithText>
      <input label="Word" actionId="my_input" blockId="my_prompt" />
    </modal>
  </command>
);
```

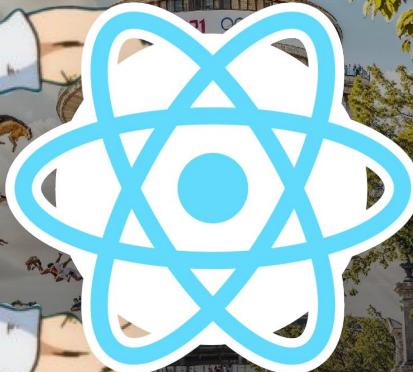
```
function MySectionWithText({ children }: React.PropsWithChildren) {
  return (
    <section>
      <text markdown>{children}</text>
    </section>
  );
}
```

La suite



- Enrichir les composants
- Intégrer dans un cas d'usage chez ManoMano
- Open source

Merci



/react fait du slack



<https://github.com/pfongkye/react-slack>