



Front-end task: Email composer

What do you have to build

An application for composing emails. It has to contain fields for entering the recipients, the subject and the body of the email. It also has to be possible to attach images to the email. The application will not actually send any email at all.

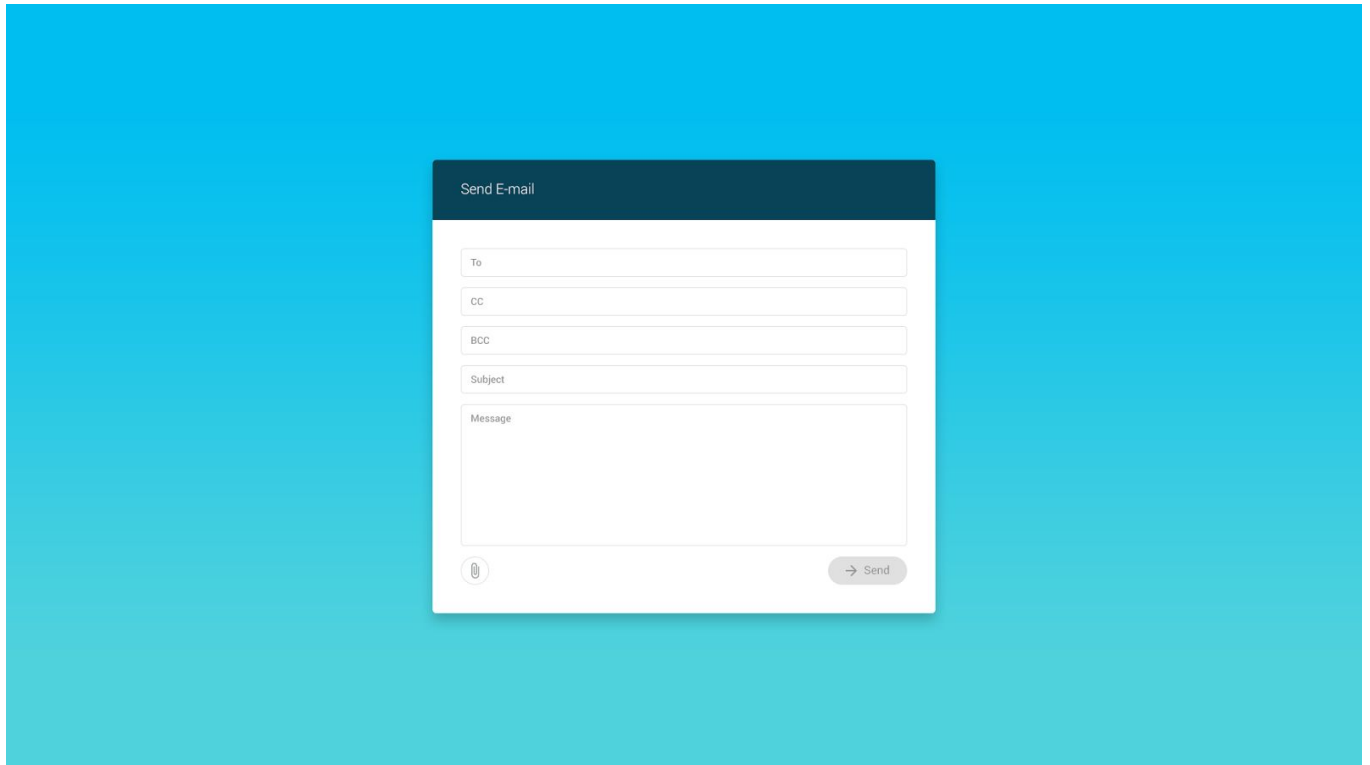
Requirements

- The component has to be built using only HTML, CSS and JS.
- Include documentation explaining how to run the project (see the 'How to submit our task' section below).
- The component must look like the attached mockups (see the 'Mockups' section below). Try to follow a 'pixel perfect' philosophy.
- The component has to be rendered correctly in the following browsers: Chrome, Firefox, Internet Explorer 11, Edge and Safari.
- You have to use pre-processors for building the CSS files (i.e. SASS, LESS, ...).

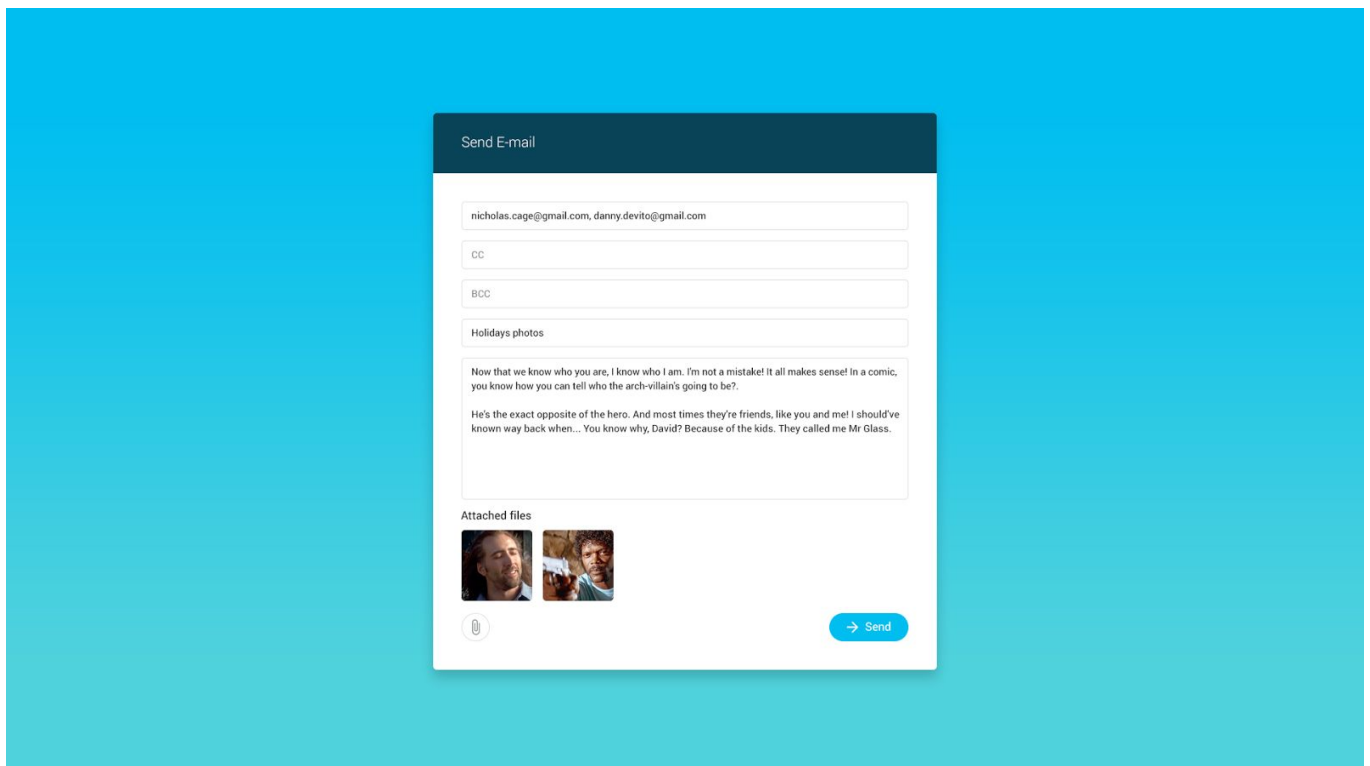
Allowed tools

- You can use CSS framework (i.e. Bootstrap, Material Design, Foundation, etc.).
- You can use vanilla javascript but you will get extra points if you use a modern framework such as Vue, React, Marionette, etc.
- You can use trans-compilers for building the JS files (i.e. Coffeescript, Typescript, etc.).
- You can write your JS files following the ECMAScript 2015 standard, but you have to use a compiler (i.e. Babel) for generating the distributable code.
- You can use any tool for bundling your code (i.e. Webpack, Browserify, etc.).
- You can use any task runner like gulp or Grunt.

Mockups



Full size: <https://drive.google.com/open?id=0B0SARhMwOfbTWjBwWVZHQjIBUkk>



Full size: <https://drive.google.com/open?id=0B0SARhMwOfbTb2pHb1N2T2hyYk0>

Send E-mail

nicholas.cage@

CC


BCC


Holidays photos

Now that we know who you are, I know who I am. I'm not a mistake! It all makes sense! In a comic, you know how you can tell who the arch-villain's going to be?.

He's the exact opposite of the hero. And most times they're friends, like you and me! I should've known way back when... You know why, David? Because of the kids. They called me Mr Glass.

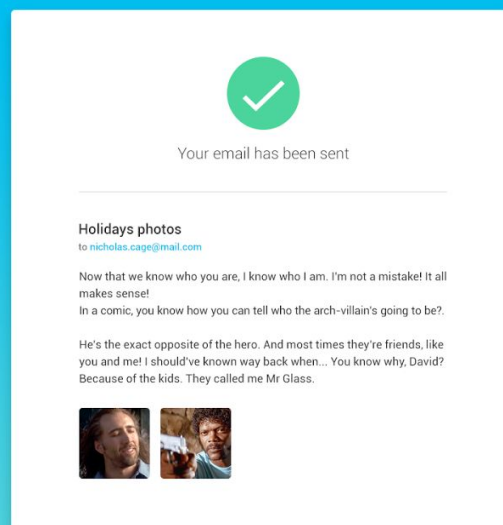
Attached files





→ Send

Full size: <https://drive.google.com/open?id=0B0SARhMwOfbTbVFWR18zTEhDRUk>



Full size: <https://drive.google.com/open?id=1YKq3N6CHof2dhx2GVN7mu50oKrn2TXe7>

Acceptance criteria

Recipients

- There are three fields for entering the recipients: TO, CC and BCC.

- If an invalid email is entered in any of them, it should be highlighted in red.

- Multiple emails can be entered separated by a comma (i.e. [email1@example.com](#), [email2@example.com](#)).

- There is no limit in the number of emails the composer can allow (so, infinite emails are allowed to be entered).
- If no valid email is entered in any of them, the 'Send' button has to be disabled.



Subject

- There is one field for entering the subject.

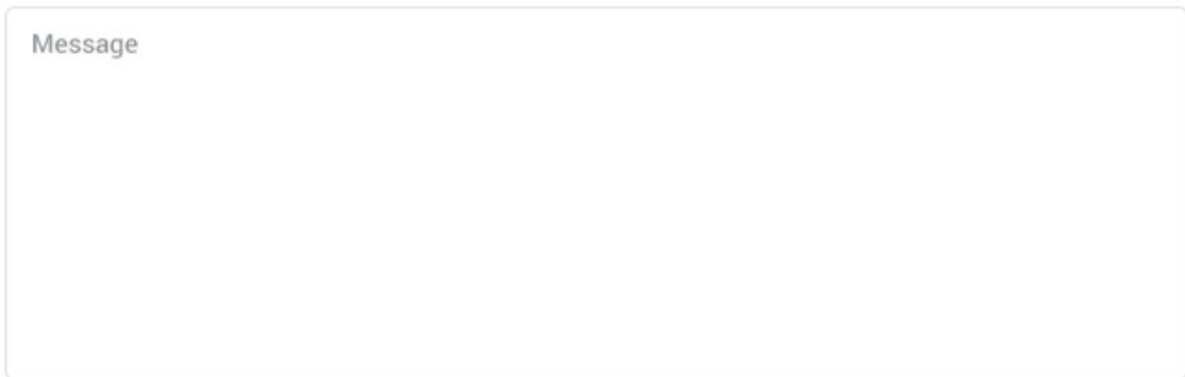
- Any character is allowed.
- The max length of the subject is 100 characters.
- If no subject is entered, the 'Send' button has to be disabled.



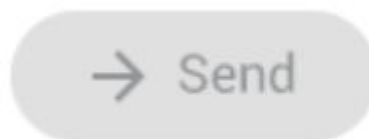
Body

- There is one field for entering the email message.

Message



- Any character is allowed.
- There is no max length.
- If no message is entered, the 'Send' button has to be disabled.



Attachments

- There is one button for uploading images.



- It is not required to apply any file validation (this means that any file can be attached, but the composer will assume that only images will be uploaded).
- Multiple images can be attached at the same time.
- After adding one or more images, it is still possible to add more images without removing the attached ones.
- When an image is attached, a thumbnail of that image has to be displayed.

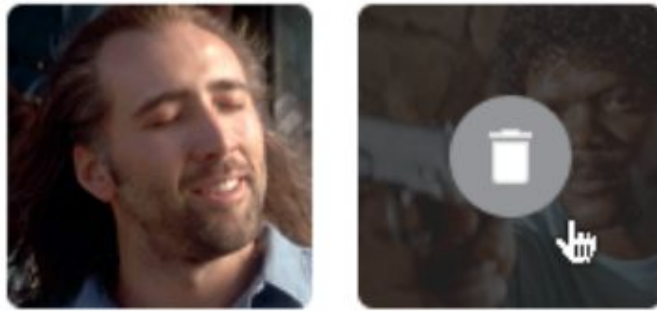
Attached files



- There is no limit in the number of images the composer can allow (so, infinite images are allowed to be attached).

- Any attached image can be removed from the composer.

Attached files



Send

- There is one button for sending the email.
- The button has to be disabled if there is no subject, body and at least one recipient.



- When a subject with at least one character, a message with at least one character and at least one valid recipient have been entered, the button will change to an 'enabled' status.



- After clicking in the button, if it is enabled, the application will display a confirmation that contains: a success message and a copy of the email sent (subject, recipients, message and attachments).



Your email has been sent

Holidays photos

to nicholas.cage@mail.com

Now that we know who you are, I know who I am. I'm not a mistake! It all makes sense!

In a comic, you know how you can tell who the arch-villain's going to be?.

He's the exact opposite of the hero. And most times they're friends, like you and me! I should've known way back when... You know why, David? Because of the kids. They called me Mr Glass.



- The application won't actually send an email, it just needs to change the interface.

Improvements

We are looking for someone who is always willing to add value, so feel free for developing features not listed in the acceptance criteria that will enhance the composer, but document them when submitting your task.

How to submit your task

- You have 1 week for submitting your task since the moment you received these instructions.
- Upload the code to a public repository and email us the URL of that repository.
- The repository should include a README.md file explaining how to run the project.
- If you use pre-processors/compiler, you don't need to commit the compiled code as long as you provide tasks and instructions for building the component.
- If a demo of the project is publicly accessible (i.e. in GitHub Pages), send us an URL for seeing it live.