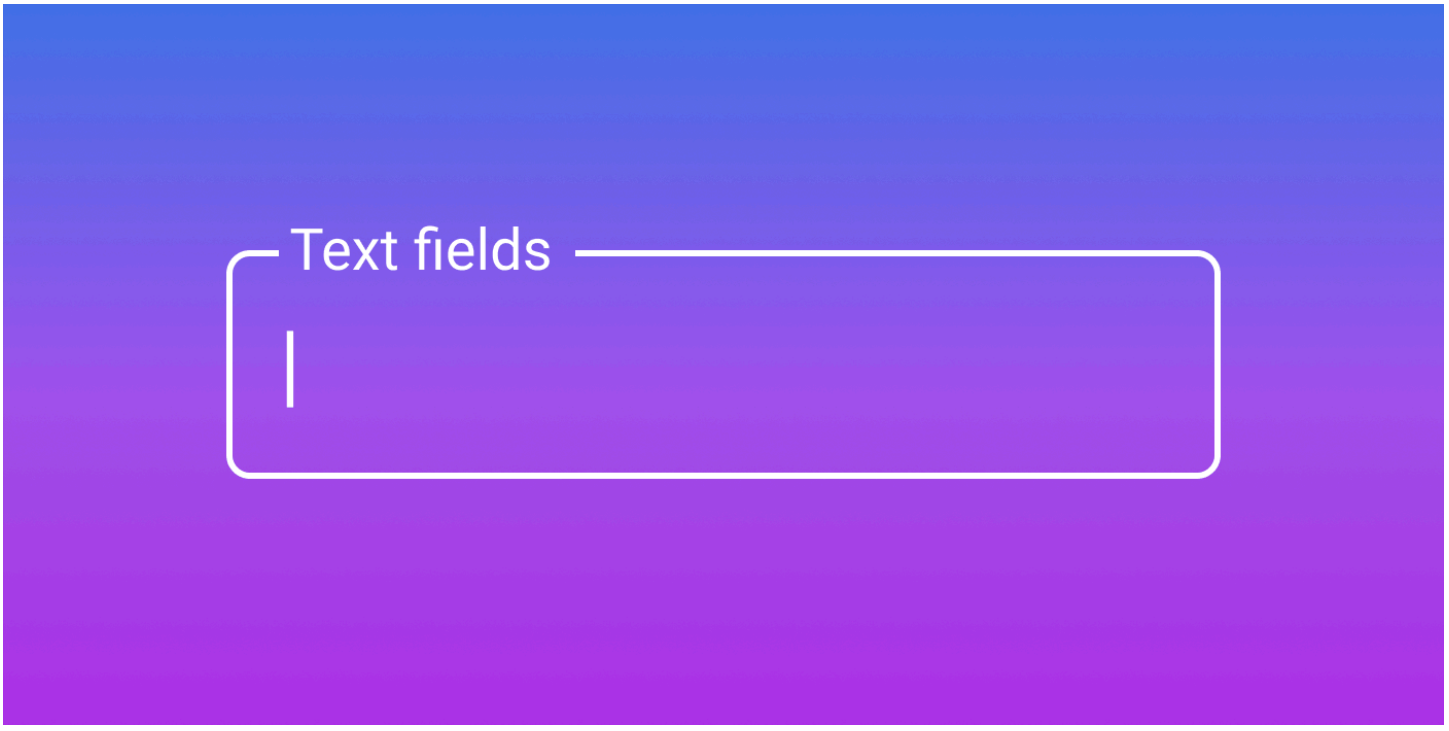
< Go to the original



UI cheat sheet: text fields

Apart from buttons, text fields are probably one of the most used user inputs. When you sign in to Friendface, or order a Nicholas Cage...

Tess Gadd

Follow

UX Collective a11y-light ~18 min read · July 15, 2019 (Updated: December 10, 2021) · Free: No

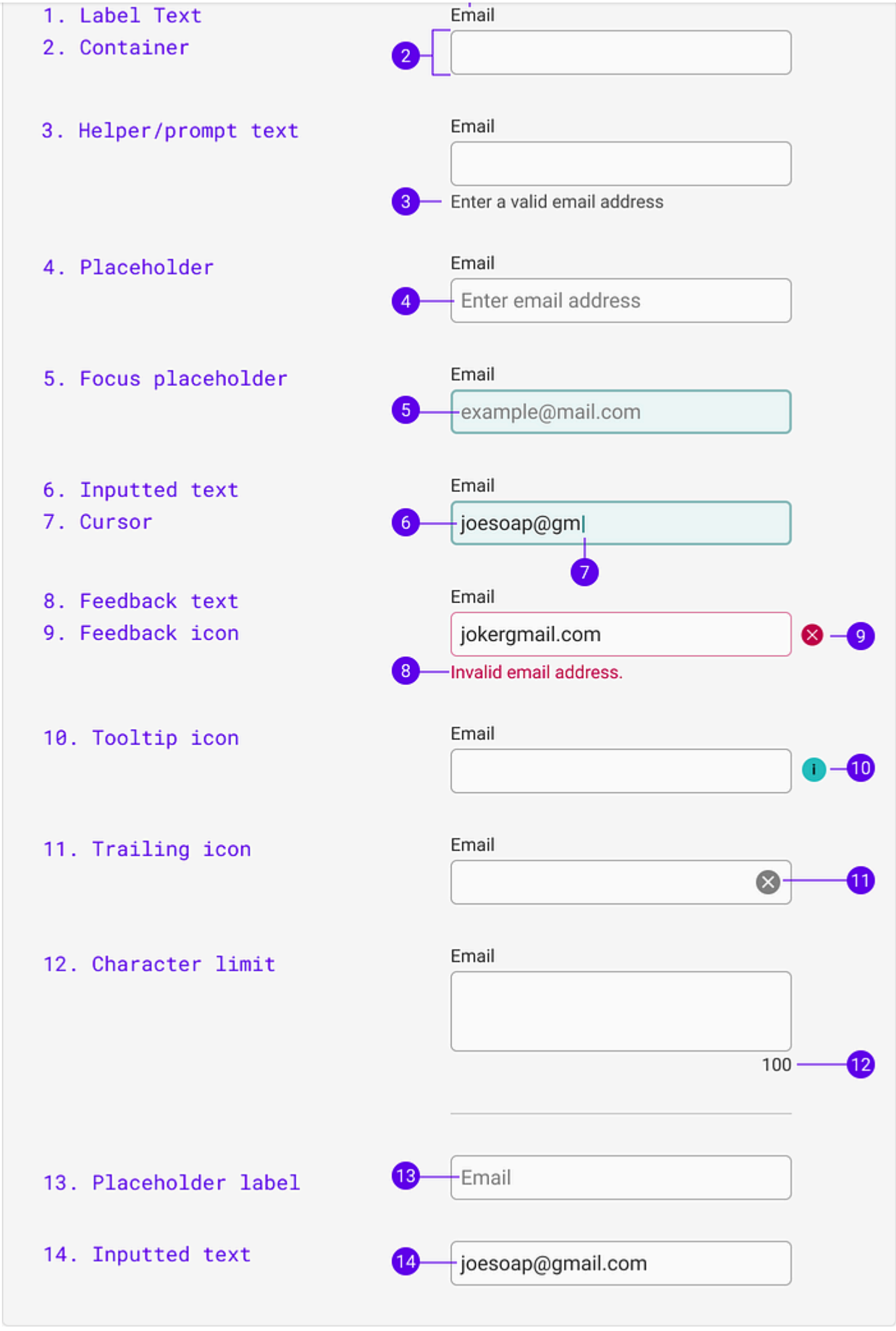
Apart from buttons, text fields are probably one of the most used user inputs. When you sign in to Friendface, or order a Nicholas Cage pillow online, or send your banking details to your new online pen-pal, chances are, you will use a text field.

In later cheat sheets, we will look at pre-determined text fields (aka dropdowns) and how to style forms, but for now, we will be looking at ye old humble text fields and how we should style them.

So what's in this cheat sheet:

1. Text field anatomy

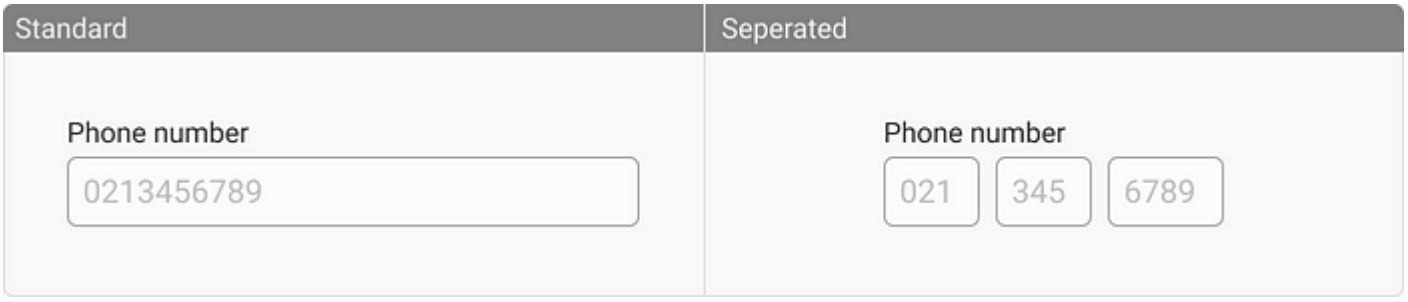
Text fields have multiple bits and pieces, some of which are only visible depending on what state it is in (default, active, inputted, etc). The below image breaks it all down. Bearing in mind that the style of the field may change, chances are that it will still have most of these elements.


1. Label Text
2. Container
Email
2
3. Helper/prompt text
Email
3
Enter a valid email address
4. Placeholder
Email
4
Enter email address
5. Focus placeholder
Email
5
example@mail.com
6. Inputted text
7. Cursor
Email
6
joesoap@gm
7
8. Feedback text
9. Feedback icon
Email
jokergmail.com
9
8
Invalid email address.
10. Tooltip icon
Email
i
10
11. Trailing icon
Email
11
12. Character limit
Email
100
12
13. Placeholder label
13
Email
14. Inputted text
14
joesoap@gmail.com

2. Text field structure

There are two main text field structures: standard and separated.

The separated text field communicates what kind of information is required and in what format. Common examples of fields that should be separated are phone numbers, dates, and some codes. If you decide to go with standard fields, then I would definitely recommend making them autoformatted (see section 8).



Example of a separated and standard text field structure.

As you can see in the above example, the separated version communicates the desired number format a lot better and it makes it easier for the user to double-check what they have written, as people tend to remember numbers in groups.

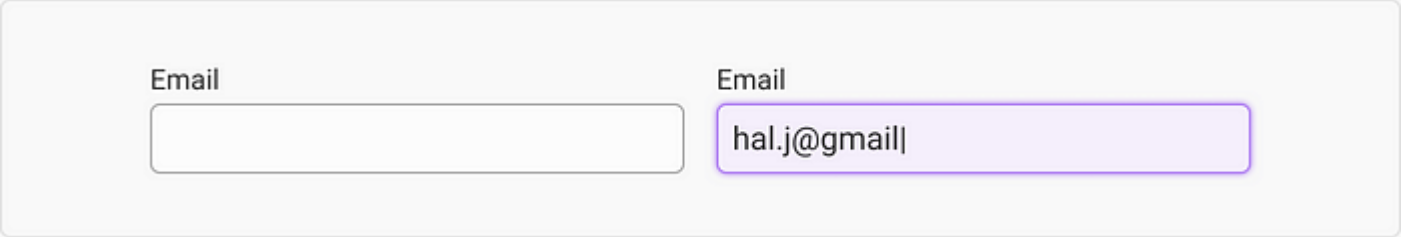
You may also want to hint to your user how long an input must be by using the length of the desired input as the length of the container.

Example of a zip code text field with two different lengths.

3. Common text field styles

There are a couple of different variations of text fields that appear all over the web, but, in this section, I will just highlight the few most used styles.

These text fields are what we have been using in interface design for years. While they may not be revolutionary, they do the job — which at the end of the day is all that users care about.



Standard text field

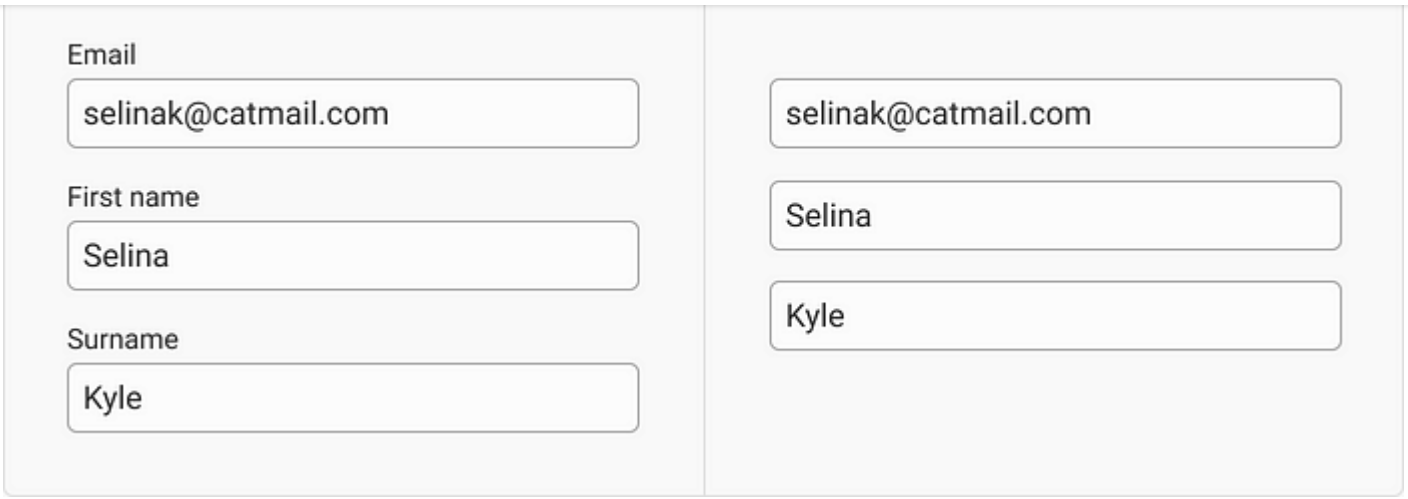
Two of the things I like about these field types are that they look like an input field and that the label is always visible.

Text field with a placeholder label

These text fields look very similar to the standard type except that they use placeholder text as a label.

Text field with a placeholder label

While this is a great solution for space saving, it isn't great for usability. The classic example is if you forget what you are supposed to be typing once you click the input field. (Full disclosure — this has never happened to me, but it was something that everyone else seems very concerned about 🤷‍♀️).



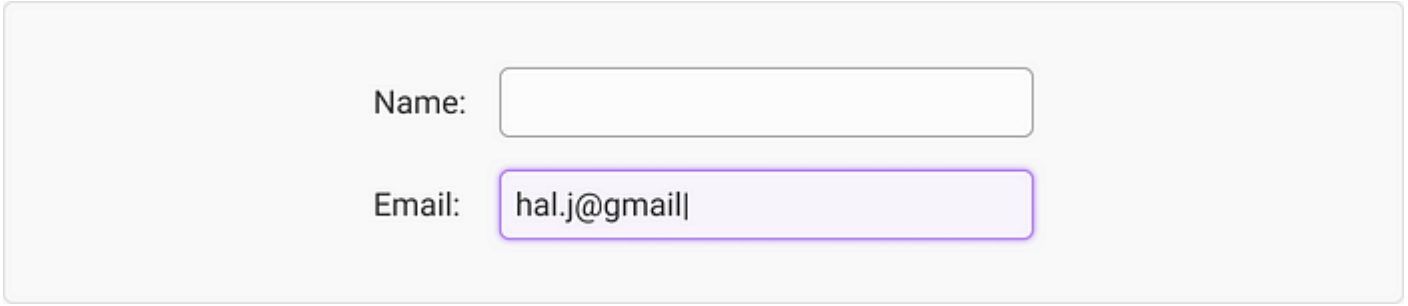
Comparison of the standard text field and text fields with placeholder labels.

What does happen to me, however, is this ^. If you look at the example on the right in the image above, you can't be sure that the first name isn't in the surname field without removing the type to check the placeholder label.

Pro tip: I generally advocate for avoiding using input fields without labels at all costs. However, I don't mind them so much for login screen as most people online are used to the classic email & password layout.

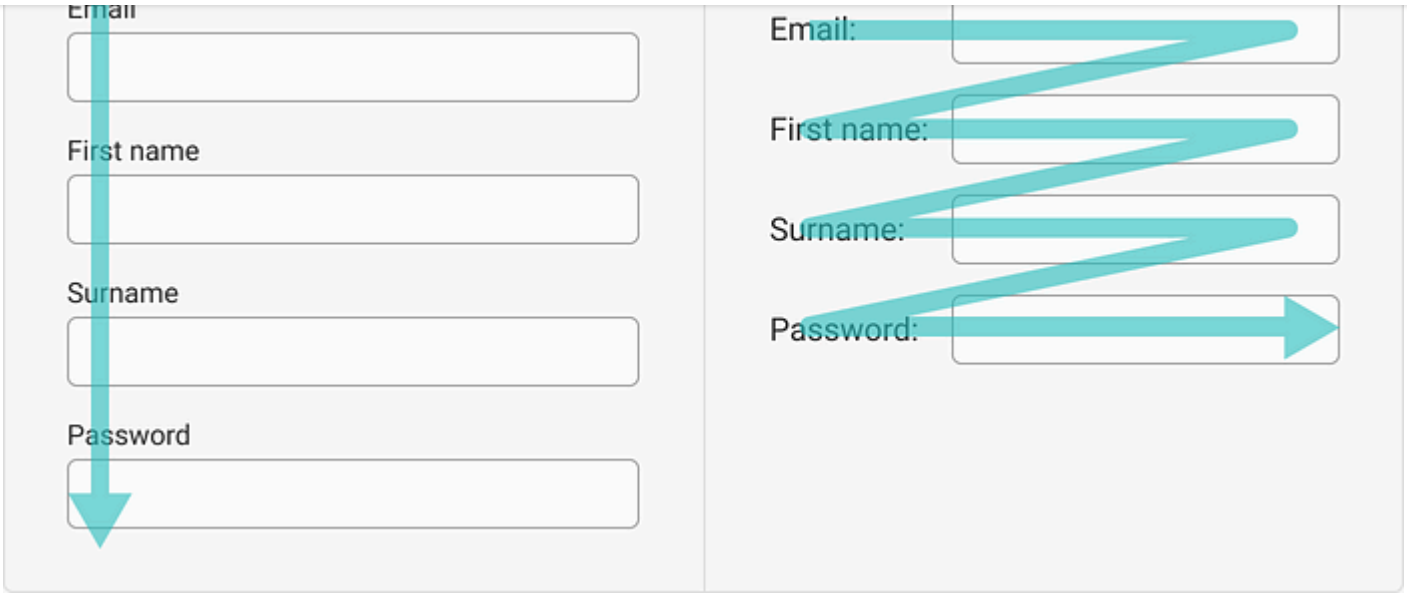
Horizontally aligned labels

Horizontal text labels are great for saving vertical space. Because of this, you will see them used more often in software applications as opposed to online forms.



Horizontally aligned text field

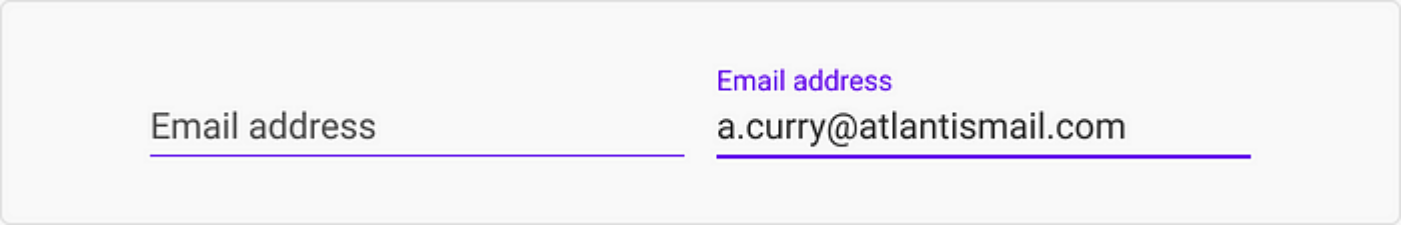
However, I tend to avoid using them because they create a weird flow for the user. As opposed to reading down, your eye has to zig-zag down the screen, which is a bit tedious. And even worse: on really long, tightly spaced forms, it can sometimes be difficult to tell which label connects to which input box.



Comparison of horizontally and vertically aligned forms.

Material Design's line text field (old) and filled text field (new)

Material Design no longer talks about the 'line only' style of text field, but it is still used widely across the internet. It's a sleek and elegant field without any of the frills, making it incredibly popular during its time.

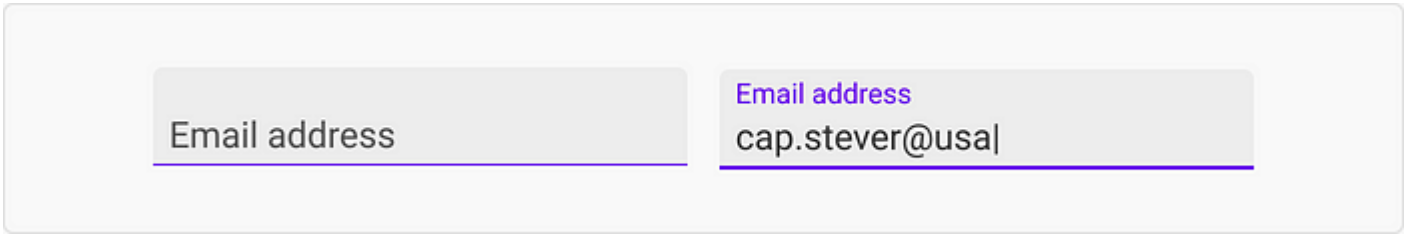


Line text field

I am not sure exactly why it is no longer in their pattern library, but I remember an ex-colleague of mine say that he found users didn't know it was an input field because it didn't look like a traditional input or clickable. **I would love to know if anyone else has had the same issue or knows anything more about this.**

Post publish update: *Dave Chui has confirmed that Material Design made changes to the text field to improve the scannability and discoverability. Their findings will be shared soon!*

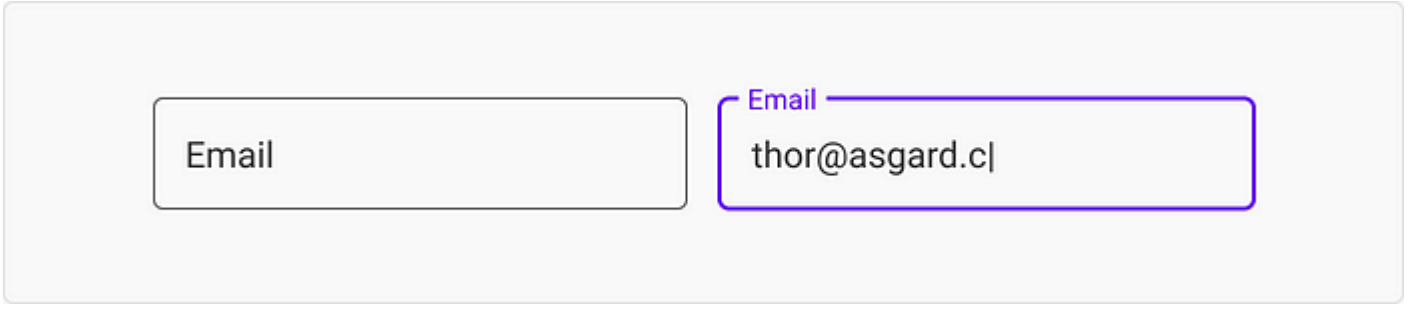
Regardless of the reason, the filled text field seems to have taken over. And if my suspicions are true, this would make sense as they look a little more like



Filled text field

Material Design's outlined text field

Material Design's outlined text fields are a modern take on the standard text field. This clever design saves space and keeps the label always visible. I am a big fan of this elegant little text field.



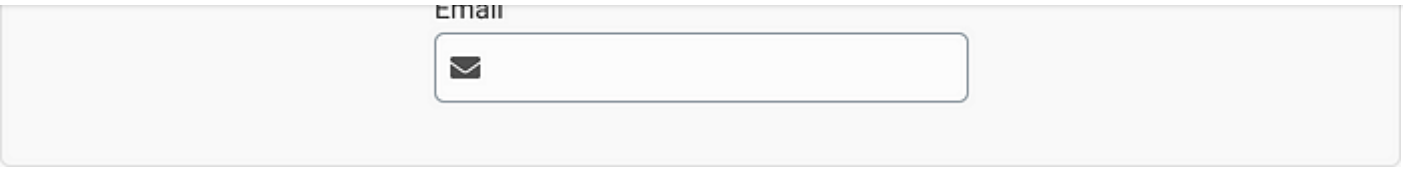
Outlined text field

4. Icons

Icons are often used in text fields but can have very different meanings depending on the icon itself and it's position. Below are the most common cases for icons in form fields.

Standard icon usage

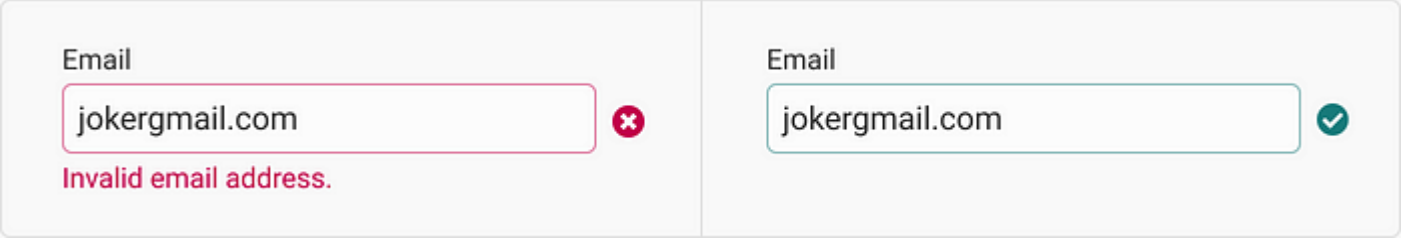
Most icons that you encounter while using forms will be there purely for aesthetic reasons. These icons will indicate what the user needs to input; for example, an envelope will indicate an email address, a geo-pin will indicate an address, a phone will indicate a phone number, etc.



Aesthetic icon usage

Feedback icons

Feedback icons are great to indicate to the user that they were successful/unsuccessful while trying to complete a form. You may say, "But we can just use colour to indicate success," but alas, my friends, people with red-green colour blindness won't be able to tell the difference. Hence, icons are a great way to indicate success or fail feedback.



Feedback icons

Clear icon

Clear icons appear in some input fields and allow the user to reset the field. While this may seem unnecessary, it is actually super helpful on touch devices where it isn't as easy to delete inputted text as it is on a desktop.

Clear icon

Voice input

This icon indicates to the user that they can enter characters using their voice. While I am opposed to talking to my phone, this feature is really helpful for

Voice input icon

Show/hide icon

For password fields, hiding the characters helps to avoid prying eyes, but you still may want to check what you wrote, hence the little show/hide button is very useful. By default keep the password hidden.

Show/hide icon

5. Text field states

Text fields have to change their state/appearance so that users know what to do. We need to add these little visual cues to nudge the user in the right direction.

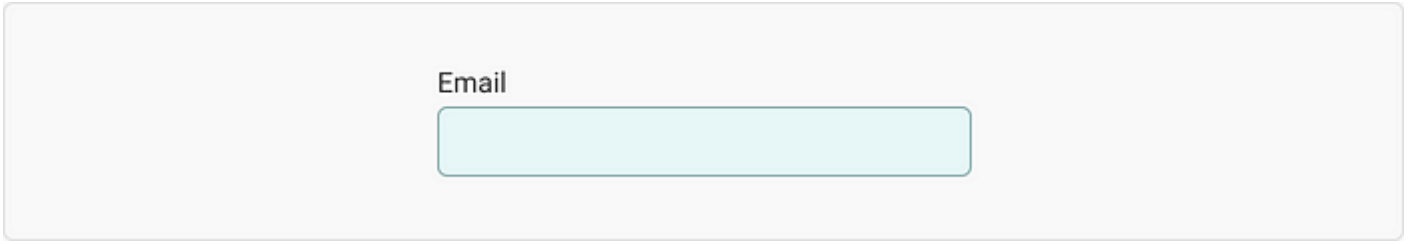
Default & disabled

The default or active state is when you haven't clicked an input field and it is ready and waiting for action. An input field may be disabled when the user should know that it is there, but can't interact with it because of business or system requirements.

Default and disabled text field

Hover

Like buttons, text fields should indicate to the users that they are interactive or clickable. Most fields change colour or increase their line thickness on hover. If you hover off it, it should revert to its original state.



Don't be a n00b tip: Touch devices don't have a hover state.

Focus

The focus state is when you click the form field and it indicates that it is selected so that you can start to type. We need to indicate this state so that users know that they can start typing and the information will appear in that box and not in another.

Focus state

a prompt. Not recommended as it isn't 'obvious' enough in my opinion, but worth knowing about if the need ever arises.

On focus placeholder

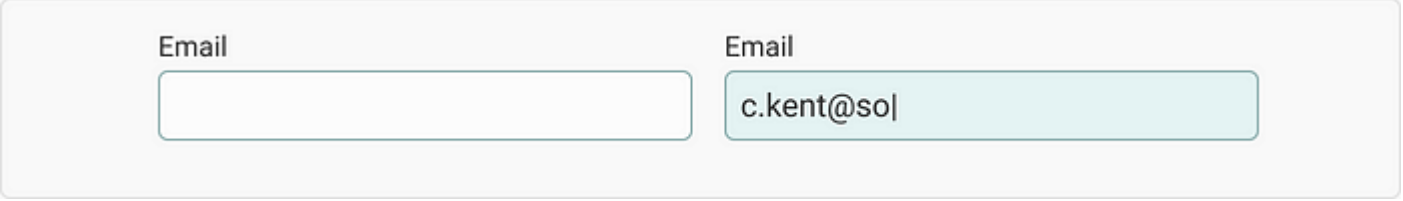
I have also heard of the focus state being called highlighted and selected.

6. Text field labelling & prompt text

In this section, we will just look at all the different text & labelling elements described in section one.

Label

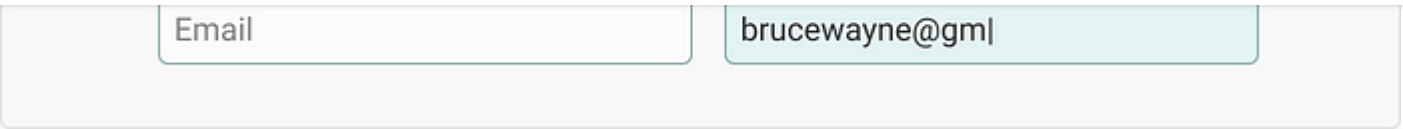
Text fields with a constant label are the most accessible as the label makes it easier for users to double-check that the information they have added is correct.



Text field label

Placeholder label

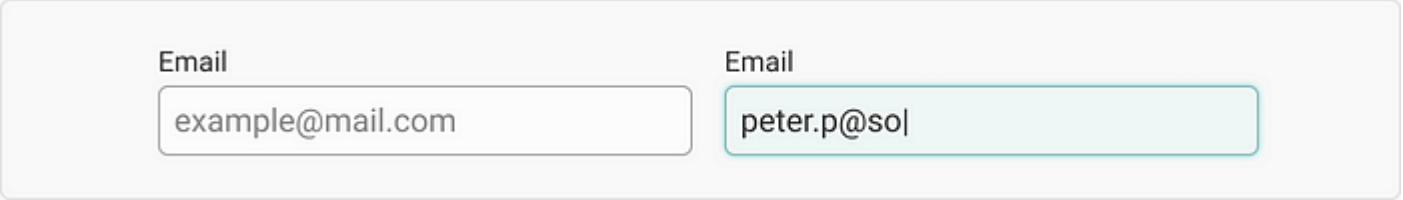
As mentioned before, we can also use the placeholder as a label. I personally avoid using them as it is difficult for users to doubled-check that they have inputted the right information into the correct box.



Text field placeholder label

Label and prompt text

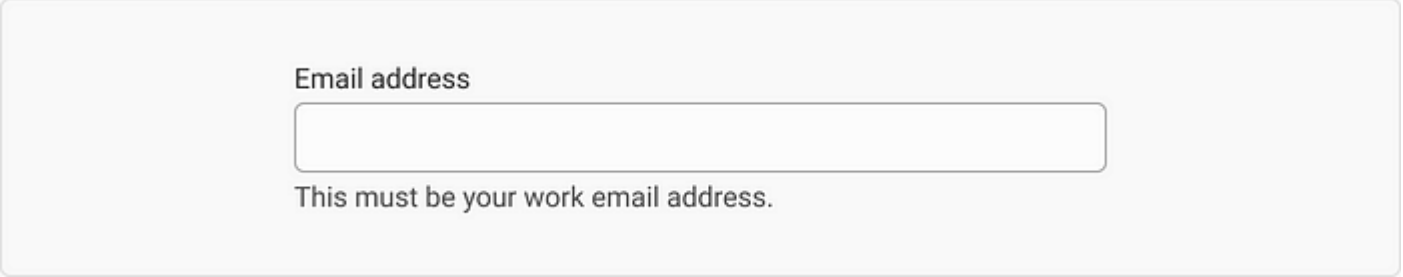
Placeholder text gives a user more context as to what information they need to input. Sometimes it seems a bit superfluous, but it can help hint as to what is needed.



Label and prompt text combination

Helper text

Helper text is useful to give the user a better idea of what kind of information they need to fill in. This extra information will often be dictated by your business rules and may be very specific to your system and product requirements.

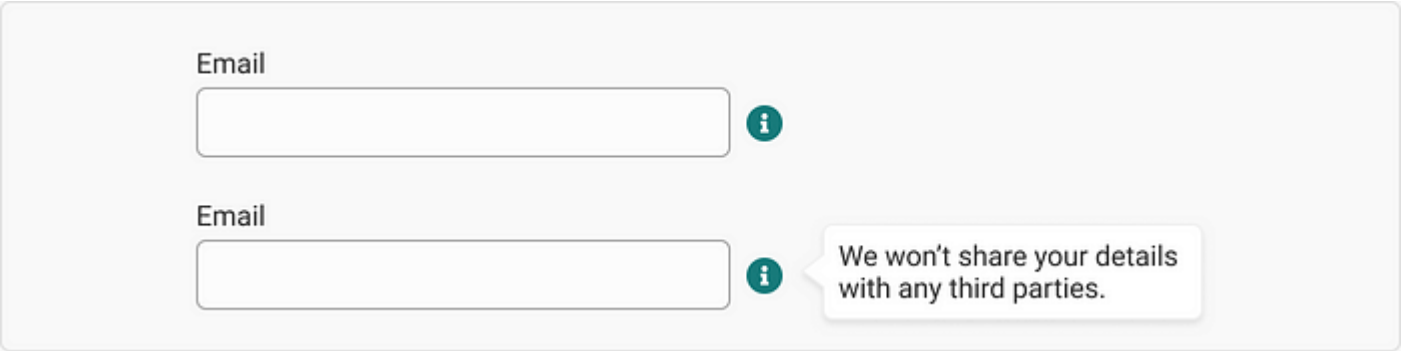


Text field with helper text

Tooltip icon

With the updates that we have gotten around privacy laws, it is critical to disclose to your user exactly how their information is going to be used. While there are a couple of ways to do this, tooltips keep things neat.

format, etc.).

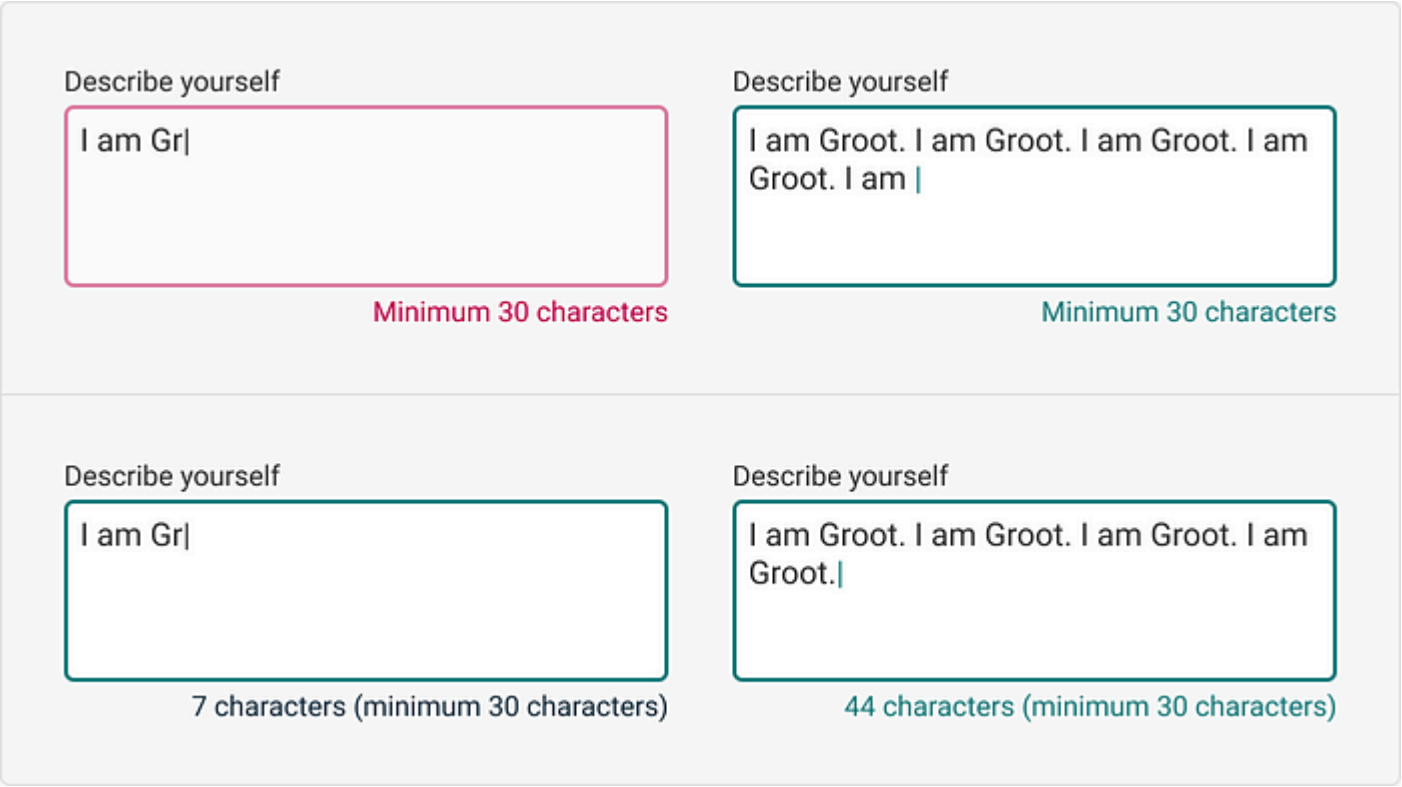


Text field with tooltip

When it comes to choosing what should go in a tooltip and what shouldn't, ask yourself 'Would not knowing this information make the form harder to fill out?' if the answer is 'no', make it a tooltip, otherwise consider using helper text.

Minimum character limit

There are no set rules for minimum character limits, so long as it is communicated to your users whether they have said enough.



Minimum characters text field

start.

Maximum character limit

Maximum character limits are the opposite of minimum character limits, as the name implies. So as opposed to making sure that there are enough characters, in this pattern we have to let users know that they have written too much.

Maximum character text field

My all-time favourite maximum character limit text field is Twitter's. To see what I am talking about, scroll down to the hall of fame.

Post publish pro tip: *Sometimes the / in 'x/100' is a bit difficult to read, so it is better to use the 'characters left' pattern. Thank you for this tip Justine Sundaram!*

Alternate maximum character text field

7. Feedback

password is incorrect, or if you have entered a valid birthday date, etc.

There are two approaches to giving feedback; both can be useful, depending on the context. In the first approach, your aunt Mildred tells you that you should wear more make-up. In the second approach, you ask your aunt Mildred if you look okay, and she tells you to wear more make-up.

So, moving on from that metaphor:

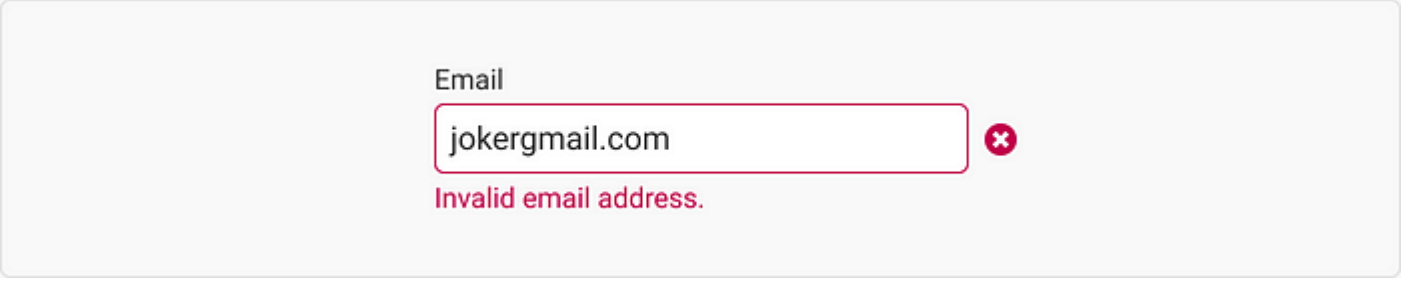
Approach 1: Tell the user that they made a mistake while they are filling out the form.

Approach 2: Tell the user that there is a mistake on submission.

For our purposes right now, the results can both look the same.

Fail feedback

Fail feedback can be used to highlight mistakes such as empty required fields, wrong passwords, invalid email (e.g. when the email doesn't contain an '@'), wrong email (when the email isn't on the database), etc.



Fail feedback

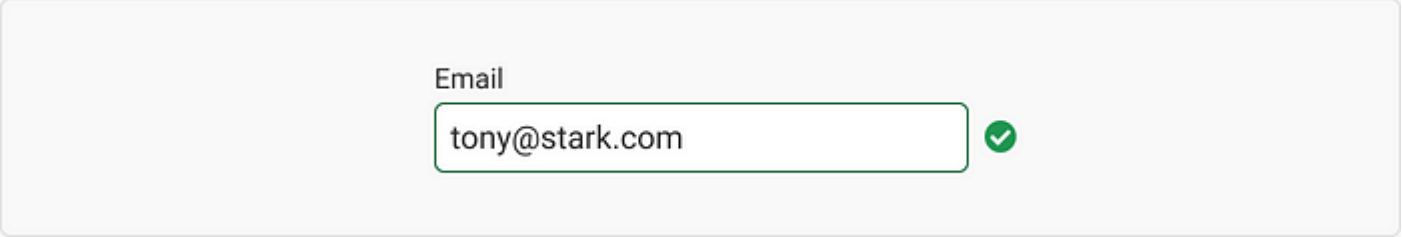
Tips for styling fail feedback:

- Make use of feedback text to let the user know why there is an issue.
- Make sure the feedback text is legible (AAA standards).

Positive feedback

Positive feedback isn't given as often as negative feedback. We can have long philosophical debates about this. ('Maybe traffic cops should tell us when we are driving well and not only when we aren't.') In this case, however, we expect most users to be able to fill out their details correctly and there being a mistake should only be the exception.

However, you may consider adding positive feedback to places that the systems can check and that would make the rest of the application process easier if the user knew that a certain piece of information was correct.



Success/positive feedback

Tips for designing positive feedback:

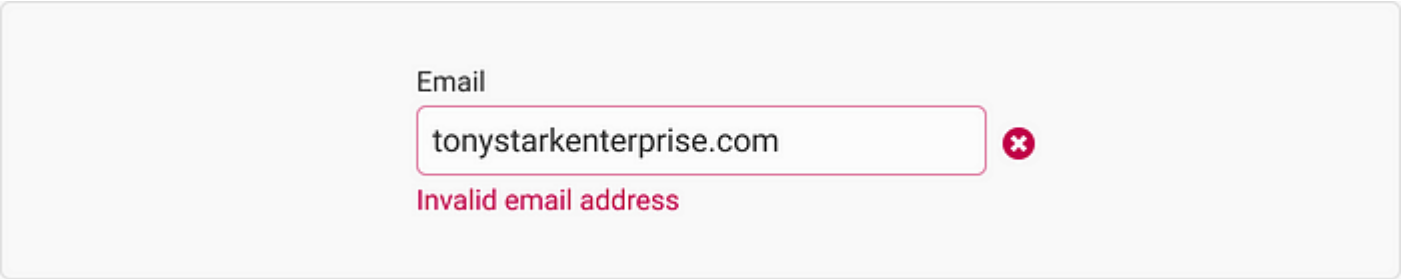
- Use an icon, as colourblind people may not be able to differentiate the colour green.
- You should only see positive feedback while you are completing the form and not on submit.

Invalid email address feedback (register)

When a user registers, the system has to make sure that the email is valid. We want users to input the correct address so that we can contact them and also so that they can log in later. This is vital for any sign-up process.

Non-optional: Check to see if the email follows a pattern

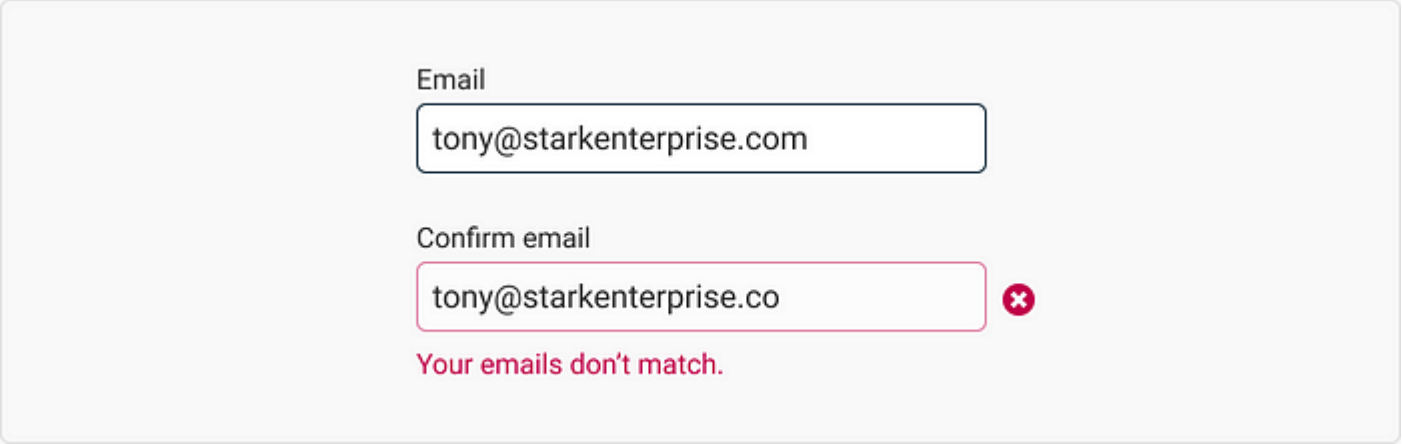
the pattern is correct.



Invalid email address feedback

Optional: Double-check the email address with a confirm field

While we don't see this that often, some forms ask you to confirm the email address to avoid mistypes and mistakes.



Email addresses don't match feedback

After all, if the user makes a mistake while typing their email into a sign-up form, it is incredibly difficult to reset it.

Optional: Send a validation email to confirm the address

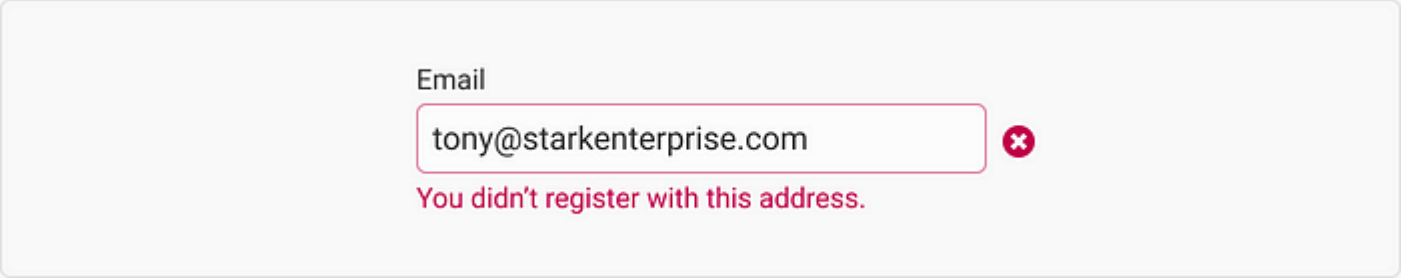
This method of confirmation is becoming more and more popular for an obvious reason: you can know if an address is real because the user is able to respond to an email you send them. Here is the standard flow of email confirmation validation.

1. The user signs up and enters their email address.

4. They follow the link in the email and it opens up a browser page and alerts them to the fact that their email had been validated.
5. They can now continue to the rest of the site.

Wrong email address feedback (login)

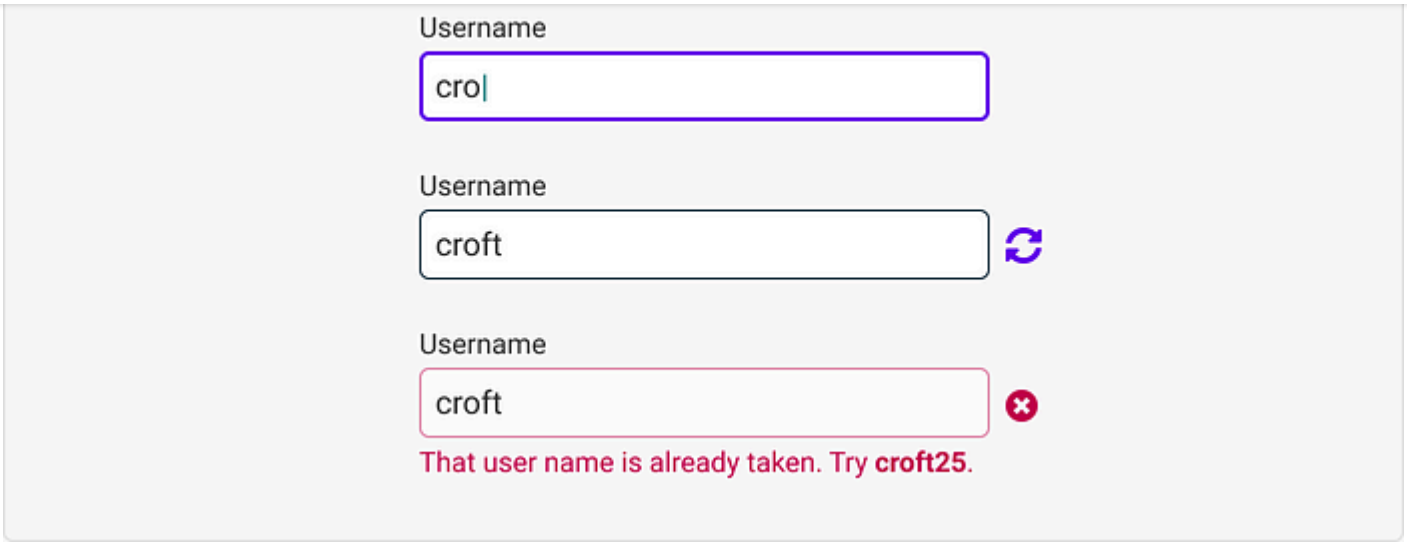
If the user enters the wrong email address, or it isn't on the database, your system has to let them know. Usually, there isn't a penalty (such as limited tries) for entering an email wrong. In some cases where the user has entered the wrong email address, you may want to suggest that they re-register with that address.



Wrong email address feedback

Unavailable username feedback (register)

Creating a username is the worst because someone out there always got there first, and all the cool ones are already taken. (Maybe one day I will register for a game before someone steals the username 'croft', but it has yet to happen.) So you sit there trying all sorts of random combinations until one of them is available. (croft, lcroft, laracroft, 1aracroft, 14racroft, 14racr0ft — success! Wait, why the hell is my name 14racr0ft?)

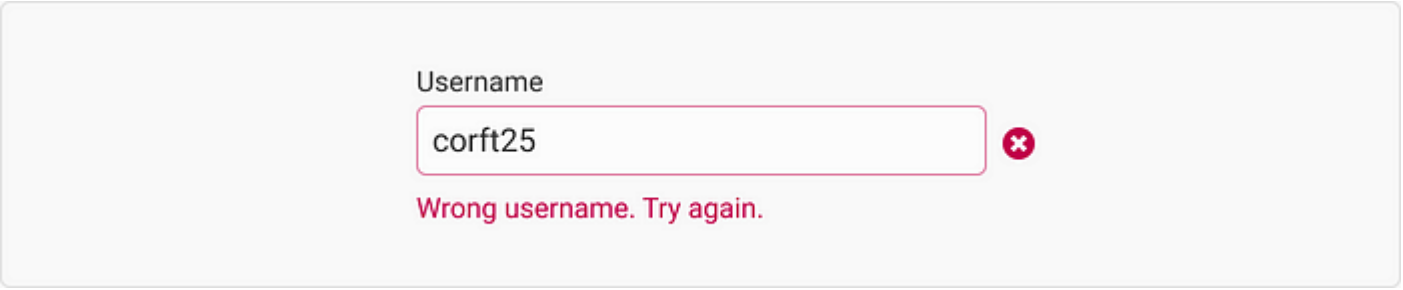


Registering a username feedback

Moving on from my username rant, you need to always communicate instantly with your user if their desired username is available before they select the confirm button, otherwise it becomes a tedious process. And while it may be a bit old school, it is sometimes nice for the system to recommend usernames that are similar and available.

Wrong username feedback (login)

Here the user has to enter their username to log in to their account. If a user forgets their username, suggest they retrieve it with their email address.

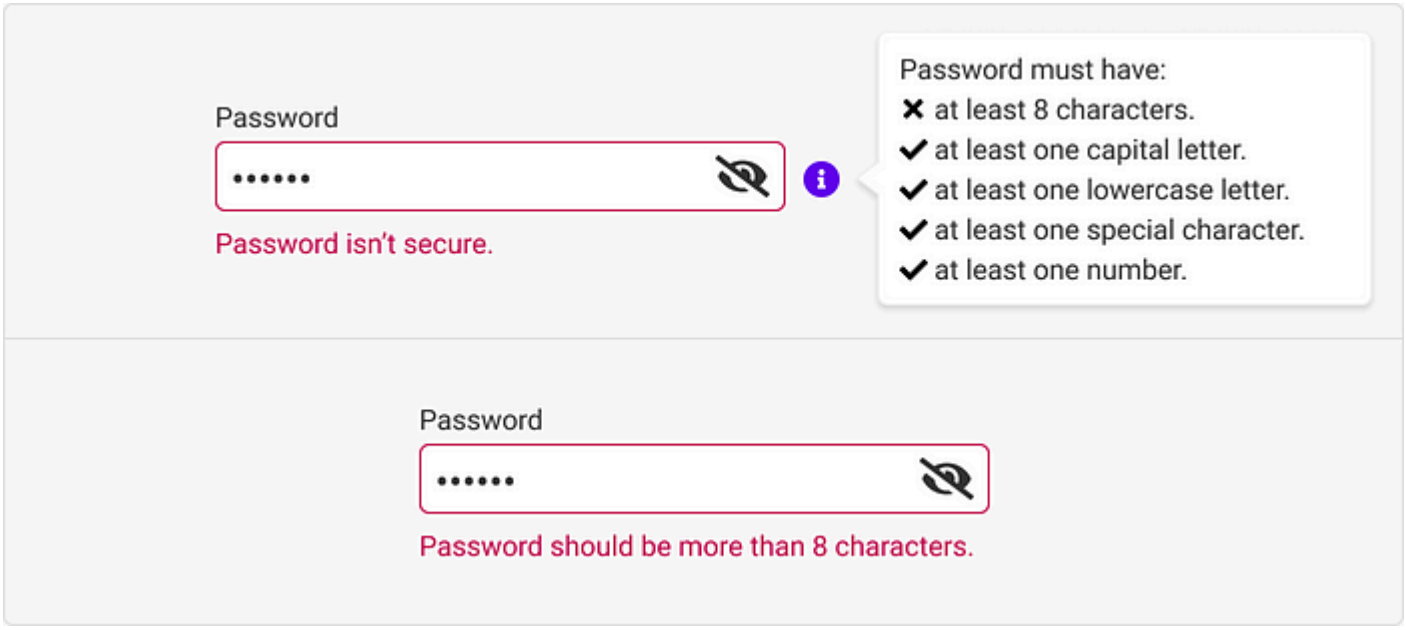


Wrong username feedback.

Weak password feedback (register)

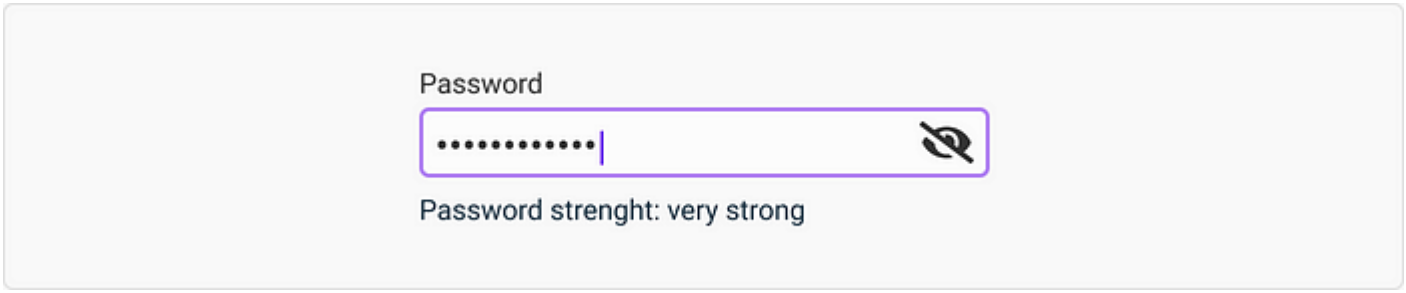
If the password that the user provides isn't strong enough, the interface will have to tell them. It will also have to tell them why it isn't strong enough and how they can make it stronger.

online forum, due to the importance of the information it contains. However, the standard password pattern is: 8 characters long, with at least one capital letter, one lowercase letter, one special character, and a number. That being said, if you make it any more complicated than that your users will get annoyed. After all, everyone just uses one password right?



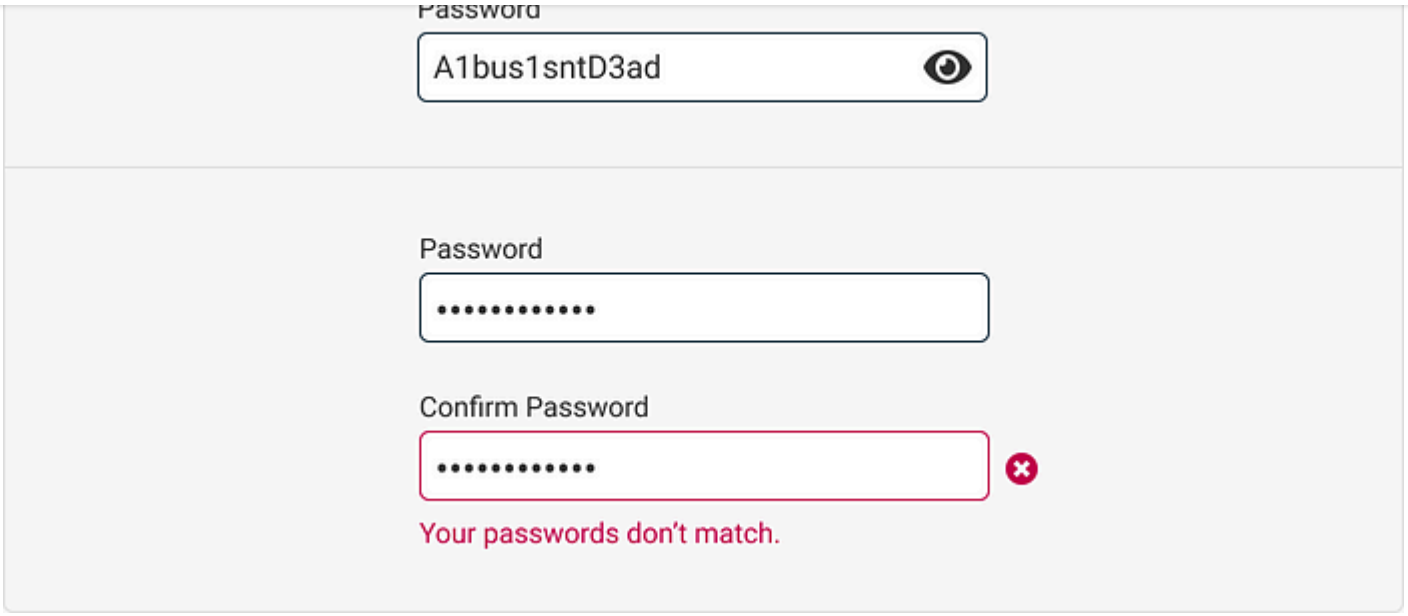
Feedback showing the user how they can improve their passwords.

If your user is struggling to improve their password you should very clearly tell them how they can strengthen it. Some do this in helper text, others in tooltips — regardless of the solution, let the user know.



Password strength indicator

Some sites indicate to their users how strong their password is depending on the complexity of the password. For a cool approach, look at Dropbox's solution in the hall of fame section below.



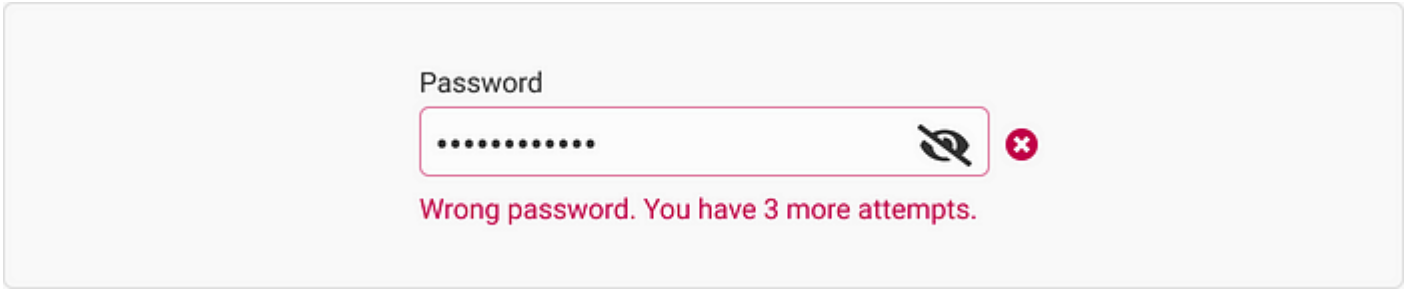
Text fields allowing users to check their passwords

Your user may also want to confirm their password — especially seeing that it is hidden. One way is to add a show/hide symbol. Another is to have them write out their password twice. While it is important to make sure that a password is correct, I personally think it is more important to make sure an email address is correct, as you can always reset a password while you can't do anything with an incorrect email address.

Wrong password feedback (login)

Password fields typically display the entered text as a non-alphanumeric symbol. (This is to stop your nosy spouse peering over your shoulder to see what your Facebook login details are.)

A login password should only show that it is wrong once the user has clicked submit. You don't want to tell someone that they have the wrong password before this point because you would just be making it easy for baddies (or your spouse).



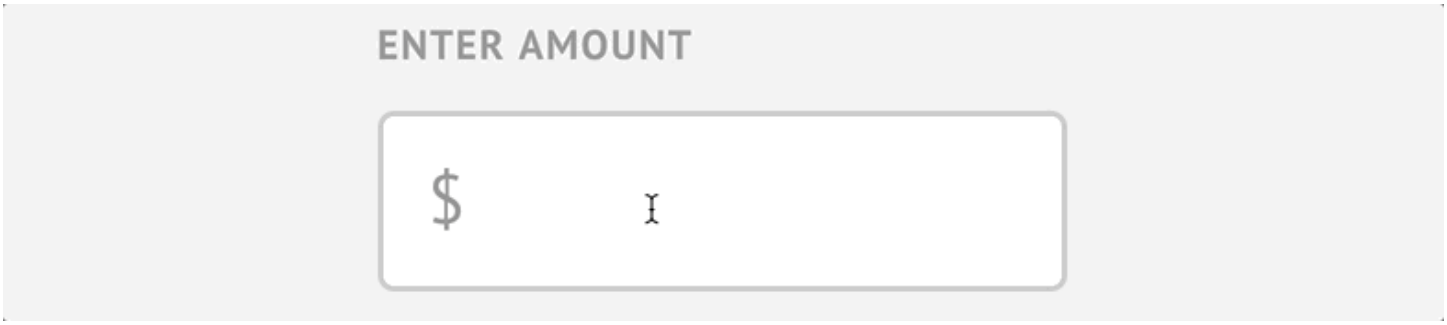
You may want to limit the number of wrong attempts that a user can enter — especially on an information-sensitive site.

8. Autoformat

Autoformatting your text boxes helps the user input the correct values as well as helps them check that what they entered is correct by grouping characters correctly.

Currency

Depending on your currency and country you may write out long numbers differently (1 000.00 vs 1,000.00 vs 1000,00) so it is helpful to create an input that standardises the format of long numbers — especially when dealing with currency.



Text field that autoformats currency | Captured from https://webdesign.tutsplus.com/tutorials/auto-formatting-input-value--cms-26745 (7 July 2019)

These input fields should also stop you from using letters of the alphabet and punctuation.

Phone number

Depending on where you are from, you may format phone numbers differently — and even if all your users are from the same country, the user may not be sure whether to add international or area codes — so having an auto-formatted phone number really helps.

Text field that autoformats phone numbers | Captured from https://nosir.github.io/cleave.js/ (7 July 2019)

For something like a phone number, you can also use separate fields to force the user to enter the number in the correct format, as mentioned in the first section of this story.

Date

Different countries write the date in different ways (DD/MM/YYYY vs MM/DD/YYYY vs YYYY/MM/DD) so in the below example it really helps to have the pattern written out on the focus placeholder text.

Text field with autoformatting and focus prompt text | Captured from https://nosir.github.io/cleave.js/ (7 July 2019)

Text masking

Text masking is similar to autoformatting but with a plus: not only does it format values correctly, but it also gives you an indication of how much more you need to input.

Field with text masking | Image from Josh Morony

I would definitely recommend checking out Josh Morony's site to learn more about text masking.

Depending on your needs, you may need autoformatted text for different reasons, e.g. card numbers, license codes, etc. Having these autoformatted just makes it all *soooo* much easier.

9. Touch targets

Input field sizing for touch screen (tap)

An MIT Touch Lab study showed that the part of the finger used for touching screens is 8–10mm, therefore the minimum target size needs to be 10mm or larger if you want to avoid users making fat finger mistakes.

Material Design suggests that the touch target should be 48dp x 48dp with 8dp between different touchpoints.

While I can't find any documentation on **iOS'** design system's target size, the general understanding is that its minimum target size is 44 x 44pts.

Input field sizing for desktop (click)

Because a cursor on a desktop is smaller than a finger on a touch screen, you can make your touch target a lot smaller. But would you want to?

10. Accessibility checklist

You've done it! You have completed designing your text field babies, and it is time to ship it off to development. But wait!... have you checked that your text fields are accessible first?

1. Does your text field meet the WCAG AAA colour contrast standards? Some designers rather use the AA standards, but I'm far too paranoid for that. My current favourite contrast checker is WebAIM.
2. Are your text fields (labels included) bigger than 44px for touch screens? I like to include labels as part of the touch area because if you click the label, the container should still switch to the focus state.
3. Is the label always visible?
4. Do the relevant fields have helpful feedback text ("error" vs "Your email isn't in the correct format")?

If you are interested in development, it is good to understand how screen readers work. Learn more here.

This section was created post publishing. Thank you T for suggesting it!

11. Text field hall of fame

In this section, I swoon over lovely interactions and text field mastery. If you have any recommendations or shout outs, let a girl know, right?

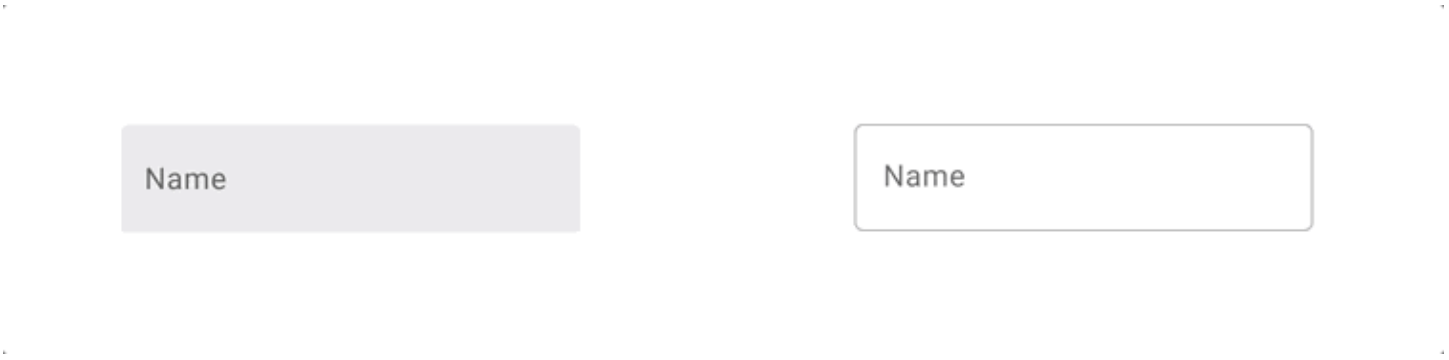
Twitter

highlights the number of characters over. Brilliant. Love it. High-fives all around.

Twitter's max character limit input | Captured from Twitter (6 July 2019)

Material Design's floating labels

Material's floating label is just so good looking. Not only does it always keep the label, but it also is such an elegant animation. Sexiness.



Material Design's input fields with floating labels | Captured on Material Design (7 July 2019)

Dropbox

Dropbox's 'create password' field gives the user an indicator of how secure their password is. And, dear reader of this story, they have also structured it so that people with colour blindness can still read it. I also like how they give their users a chance to have a less secure password if that is what they want. Such a simple and human solution.

Dropbox's password field | Image via https://littlebigdetails.com | via rammionline

12. Closing thoughts

Thank you for reading this - I know it was a long one. As always, if you have any thoughts, suggestions, corrections, questions, or recommendations for the hall of fame, please don't hesitate to leave me a comment below.

#ux-design #ui-design #ui #ux #usability