Read only fields

WIP

Status	COMPLETE
Impact	HIGH
Owner	@ Charles Lee (Unlicensed)
Approver	@ Charles Lee (Unlicensed)
Contributors	@ Ronald Aveling @ Thomas Walker
Informed	@ Mitchell Hamilton @ John Molomby @ Ben Conolly
Due date	
Resources	

📚 Relevant data

A11y for read only web-native fields has long been a contentious problem space. Generally there are a few reasonable approaches to displaying read-only values in a form. The read-only attribute on form fields are applied inconsistently in the browser such that they:

- only apply to inputs of type `text` and `number` (checkbox, radio, and select elements either don't have this attribute or the attribute doesn't work as expected)
- are read in inconsistent ways across the different core screen reader technologies

However the common alternatives such as the `disabled` attribute and `plain text` values all come with various a11y issues in the context of browsing a form such that an obvious superior solution is not self evident.

📘 Background

Users are given the ability to toggle field states in the ItemView of the Keystone Admin UI between `editable` `hidden` and `read`, the latter of which encompasses the read-only functionality we are looking to provide. At the moment, these are all displayed as plain text values, which present with a set of a11y challenges. Namely:

- These values do not have a tab-stop,
- These values are not associable with the field label

However the problem space is made complex by the fact that a variety of keystone fields are not able to ascribe to using a readOnly attribute. We have `input` types with read-only attributes that don't work as expected natively, such as `radio` and `checkbox`, we have form fields that don't have read-only attributes at all such as `select` and we have fields that are not traditional form fields such as `image` , `relationship card`, `relationship list` and `document`. This DACI aims to present and articulate the solutions and problems in this particular problem space clearly so we can come to a decision .

🌈 Options considered

	Option 1	Option 2	Option 3	Option 4 RECOMMENDED
Description	Label and plain text	read only input fields	disabled input fields	read only input field and custom read only fields for non standard keystone form fields.

Pros and cons	➕ Consistent browsing experience for all fields ➖ Does not have tab-stop ➖ Is not associable to a label, therefore values are extricated from their form and label context.	➕ Native browser attribute for flagging a form input as read-only ➕ Keeps tab-stop, so keyboard and screen reader users can access the field and value within the context of the form. ➕ Read only form value is associated with the label. Focusing on the label will also focus the field and announce in the screen reader. ➖ Does not work for checkbox ➖ Does not work for select ➖ Does not work for non-form fields (image, relationship document)	➕ Works for checkbox ➕ Works for select ➖ Ignored by screen readers ➖ Does not have a tab stop (not focusable) ➖ Values are completely unreadable by non-sighted users	➕ Ensures that we rely on good browser defaults for fields where these exist ➕ Gives us the capability to provide a cohesive read-only experience across **all** keystone fields. ➕ Keeps tab-stop, so keyboard and screen reader users can access the field and value within the context of the form. ➖ A **lot** of custom code for differentiated fields that deviate from default browser behaviour. ➖ Requires spiking of potential creative solutions for providing sound accessible read-only behaviour for differentiated fields.
Estimated cost	SMALL	MEDIUM	MEDIUM	LARGE

✅ Action items

- [ ] Spike out the problem space in the context of Keystone Admin UI item view, particularly in reference to differentiated field types:
  - [ ] checkbox
  - [ ] select
  - [ ] document
  - [ ] image
  - [ ] file
  - [ ] date picker
  - [ ] password

🌟 Outcome

We've reached a consensus that, consistently bad is not better than sometimes good.

Engineering should lead the effort to finalise this decision, this is no longer a blocking body of work for design.