# 2020-11-12 ICT-ACR meeting notes

Action item (for all): Review ACR templates / sample ACRs in GCconnex. Note which common elements should be included in the GC standard report. In 2 meetings (Dec 10th) we will come together and brainstorm the common report format.

## ESDC: Risk assessment table

* Basically took how axe scored issues: Critical, serious, moderate, minor
  + Critical would mean: we cannot release this ICT until fixed
* More specific than per-SC: different scenarios within an SC may be ranked higher risk
* Depending on how long it will take to fix, you can allow non-compliant elements with low risk (as long as they will be fixed soon)
* This will allow launching important apps before fully accessible, if the impact is minor
* HC: “Interim authority to operate”: this fits in here.
* People ask “what do I need to fix?” A: All of them! But clients want prioritization
* PSPC: When we produce reports for ext. vendors and management, how do you weight “should this go live or not?”
  + For each SC, there is a high, medium, low risk.
  + We have DG/ADM sign off on the risk from our branch and client branch
  + This shows a paper trail of liability for if issues are reported by end-users
* ESDC: We report on per-SC, and we have these issues in our tool that can be pre-selected.
* We’ve agreed within ESDC on the severities of each issue in the list.
* CRA: This is important for developers to prioritize fixing issues. In CRA, we first recommend all Level A to be fixed before launch.
* ESDC: Non-text content can fail because of missing alt text (serious), but also because of redundant alt text which isn’t serious at all.
* Some level A criteria like Parsing have no practical impact nowadays – AT is robust enough.
* You have to weigh the risk of delaying launching vs. the risks of inaccessibility

## Report templates

* SSC:
  + Scope of pages, tested against what standard, general & usability issues, then Per SC issue descriptions and links to sufficient tech
* PSPC: Separates issues into risk levels.
  + Summary includes number of pages tested, how many issues, % high, % medium, % low risk.
  + Report starts with PASS/FAIL table per SC
  + For each page, take a screenshot of the page (to identify it)
  + Then for each SC, report each issue
    - Text description (custom text input)
    - Screenshot
    - WCAG ref: could be sufficient technique that should be used, or a failure.
      * Evidence that it fails
    - Audience affected: custom for each issue
  + For risk level of each issue, we choose those based on the issue every time. There is a written risk criteria, but it is not always the same per SC
  + SSC: You set aside “N/A” in your totals – only reporting on % FAIL and % PASS.
  + PSPC: This one just looks more accurate, otherwise because of large amount of N/A’s, the % PASS is much higher than it should be so developers think they’re doing well, when they aren’t.
  + ESDC: For risk levels, it is hard to justify – also depends on the page type (is it a popular page or an unimportant page?) I like the idea but it requires more justification/documentation.
  + SSC: If evaluating multiple pages, how do you combine scores? e.g. if 4/5 pages pass 1.3.1, does it get a “partially supports”, 80% score, etc?
* CSPS:
  + ACR includes axe info on “fix the following” which is helpful for developers.
  + Per issue organization
  + Includes “Historical transactions” (previous reports) and “Resolution” which is filled in by the developer, showing how they fix
  + For each issue, they report the tool /browser/inspection method (e.g. “visual check with IE11”)
  + 2 reports: “noScreenReader” and “screenReader”
* AAFC:
  + Rabab was not present to demo, but report format is similar to other PASS/FAIL WCAG tables (per page, with % scores).

## Wrap-up

* SSC: There are a number of issues to conquer:
  + reporting formats
    - for different audiences
  + testing methods
  + how issues are reported, in terms of wording
* CRA: Even if we have our own tools and internal reports, we should have a common format for GC. I like PSPC’s report format best. One thing I like is including the sufficient techniques. Much like CSPS had the “how to fix” text copied from Axe. You want to provide the info to developers, but you don’t want to have to write this yourself every time.
* We need to figure out exactly what fields need to be in the GC standard report format.
* SSC: There are different sections in the report. Preamble, executive summary, per issue…
* ESDC: The common report format is a good idea, but we also have separate needs and will need our own reports, our own processes etc. But we should also have consensus on what constitutes passes and fails.

Action item (for all): Review ACR templates / sample ACRs in GCconnex. Note which common elements should be included in the GC standard report. In 2 meetings (Dec 10th) we will brainstorm the common report format.