After this email, detail design of all US in Palier 2 must follow below structure:

1.       Create 3 Subtasks in US JIRA.

a.       Front-end implementation

b.      Back-end implementation

c.       Clarification (optional) only create it is not existed.

2.       Front-end implementation structure need 4 parts:

a.       Creation:

                                                               i.      Which component/service/pipe/guard are created? If it requires 3rd party libraries or components (Kendo, ngx-spinner …), please note it in the design too.

                                                             ii.      Where to put that components?

                                                            iii.      What should be done to make application recognize new components? E.g: register with Angular modules, add routing, add provider, add clarification…

b.      Reusable:

                                                               i.      Do it have any point which can be reused by another US in previous Sprints? E.g: reuse structure from component A, B, C…

                                                             ii.      Do it need to customize existed component?

c.       Major requirements:

                                                               i.      Are there any point that only defined for current US? E.g:

1.       Which configuration variables will be added?

2.       How long does it take to load the screen?

3.       How many item need to be displayed on screen?

4.       How security will be configured for current screen?

d.      Clarification:

                                                               i.      Question about front-end implementation need to be clarification with BA? => This will be moved to Clarification JIRA after detail design is reviewed.

3.       Back-end implementation structure:

a.       Creation:

                                                               i.      How many API/Service/Repository are created?

b.      Reusable:

                                                               i.      Do it have any services/method can be reused?

c.       Major requirements:

                                                               i.      Security

                                                             ii.      Transactional/Non-transactional

                                                            iii.      Special query condition. E.g: fuzzy search, wild card search…

d.      Clarification:

                                                               i.      Question about back-end implementation need to be clarification with BA? => This will be moved to Clarification JIRA after detail design is reviewed.

Important Note: When create a new detail design JIRA, it must be following above structure. Any incorrect creation will be forbidden and need to be corrected.

For example for Front End:

Detail design should be written as bellow:

1. Create new component **affaires** (1) in app/affaires/components/ directory.
2. Register (1) with affaires module.
3. Define routing for (1).
4. Layout search screen (try to reuse structure, class of personnes.component.html).
5. Check droit "Liste seulement" before search.
6. Add new configuration value for limitted 100 search result in siel.web-front.properties.
7. Use new custom \_card \_ layout to render affaires search result item.
8. Define new warning message with parameter in fr\_CH.json for *"Seuls les $LIMITE$ premiers résultats sont affichés. Veuillez affiner vos critères de recherche si vous n'avez pas trouvé l'affaire que vous cherchiez."* + Add check to display this message.
9. Define properties mapping for Trier dropdown list.

Question to clarify:

1. Should trier dropdown list place outside the scroll of search result?
2. What is the purpose of star button in card item? Is it favourite?

For Example for Backend:

Detail design should be written as below:

1. Mapping Affaires entity by mdc-gen.
2. Create AffairesSearchCriterion.
3. Create API with mapping api/affairs for method findAllWithCriterion(AffairesSearchCriterion affairesSearchCriterion):PageResult<AffairesDto>
4. Check user droit "Liste seulement".
5. Create AffairesService inherited from GenericDomainService + create AffairesRepository + AffairesRepositoryImpl implement findAllWithCriterion(AffairesSearchCriterion affairesSearchCriterion):PageResult<AffairesDto>
6. Search query must follow definition in **Règles d'évaluation des critères**.