**Intro**

**Stand: 19.04.2021 (beginn des neuen Branches** devSLC\_2021**)**

In diesem Dokument möchte ich kurz die Funktionialität des VSim’s aus Sicht «Coding» beschreiben. Aufgrund der Komplexität aufgrund vieler Verknüpfungen ist es mir wichtig, zuerst eine Übersicht zu gewinnen, bevor grössere Umbauten gemacht werden.

**Funktionalität Shiny-App (server & UI)**

1. Vorarbeit (server.R)

* Laden aller Funktionen (colors, icons, presentation, rules, sims, import, plots)
* Laden der Excels (verwendet das Excel, dass in Funktion «import\_from\_excel» als Default gesetzt ist)

|  |  |  |  |
| --- | --- | --- | --- |
| Namen / Tabs im Excel | Bedeutung | Dimension (19.04.2021) | Spalten (19.04.2021) |
| Qlist\_1 🡨 Q1 | (Fragen 1) | 29 x 7 | Qnum, Question, Answer 1 - 5 |
| Qlist\_2a 🡨 Q2a  Wird später überschrieben | (Fragen 2a) | 12 x 4 | Qnum, Question, Massnahme, Column1 |
| Qlist\_2b 🡨 Q2b | (Fragen 2b) | 6 x 2 | Qnum, Question |
| Qlist\_2c 🡨 Q2c | (Fragen 2c) | 8 x 7 | Qnum, Question, Answer 1 – 5 |
| Glist 🡨 G1 | (Gaps) | 29 x 6 | Qnum, Question, Gap1, Ind1, Gap2, Ind2 |
| Gtypelist 🡨 Gtype | (Gap-types) | 29 x 2 | Gap1, Gap1\_type |

Redundanz: Qlist, Alist, Glist & Gtypelist ev. alles in eine Tabelle nehmen… 🡪 Macht es irgendwo sinn diese

Getrennt zu lassen?

* Definition der choices für Panel 2b (Zeitverwendung)
  + "viel weniger als bisher"
  + "weniger als bisher"
  + "gleich viel wie bisher"
  + "mehr als bisher"
  + "viel mehr als bisher"
* Definition weiterer Variablen für die kommenden Auswertungen

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Type | Bedeutung |  |
| result\_coll | list | «Collection der Resultate» |  |
| results | Named character (29) | «Collection der Antworten auf Fragen im Teil 1» |  |
| Qlist\_2a | data.frame(1:6) | Nötig? |  |
| relevant\_gaps | NULL |  |  |
| results2 | NULL | Nötig? 🡪 wird in postProcessing von mainPanel1 erstellt |  |
| kommentare2 | NULL | Nötig? 🡪 wird in postProcessing von mainPanel1 erstellt |  |
| results2b | NULL |  |  |
| kommentare2b | NULL |  |  |
| results2c | NULL |  |  |
| kommentare2c | NULL |  |  |
|  |  |  |  |
| question\_id | 0 |  |  |
| Survey\_Sections | List | Übersicht, wo welche Teile anfangen & enden |  |
|  |  |  |  |

2. Kern-Funktionalitäten (server.R)

* Navigation
  + observeEvent “prev\_button” & “next\_button” 🡪 increase & decrease “question\_id”
* renderMainPanel()
  + render\_main\_panel\_1\_2a\_2b\_2c\_3
* renderProgressbar()
  + renders the progress bar
* inclRmd()
  + Function to render reports (Zwischenauswertung & Schlussauswertung)

render\_main\_panel\_1.R

* helper-functionalities 🡪 questionaryPostProcessing() / option\_list() / output$question / output$save\_results
* output$mainPanel
  + Teil1\_intro 🡪 Text
  + Teil1\_fragen beantworten
    - Holt vorgespeicherte Antworten aus results
    - output$question holt die Frage aus der Qlist\_1
    - option\_list() holt die Antwortmöglichkeiten aus Qlist\_1
    - output$save\_results speichert gegebene Antworten in results
  + Teil1\_endstatement 🡪 Text
    - output$save\_results speichert alle gegebenen Antworten von results in neuer Alist und diese in result\_coll$Alist
  + Teil1\_summary
    - questionaryPostProcessing()
      * hole Alist aus result\_coll$Alist
      * speichere Antworten-Liste als Excel in «data/outputs/»
      * Qlist\_2a: Berechne relevante Fragen für Teil 2a mit calc\_relevant\_questions\_2a(Alist)
      * Calc\_survery\_question\_ids()
      * Erneute Erstellung von results2 & kommentare2
    - inclRmd("rmds/gaps\_user.rmdy")

render\_main\_panel\_2a.R

* helper-functionalities 🡪 questionaryPostProcessing() / option\_list() / output$question / output$save\_results
* output$mainPanel
  + Teil2\_intro: 🡪 Text
  + Teil2\_fragen beantworten:
    - Holt vorgespeicherte Antworten aus results2 und kommentare2
    - output$question holt die Frage aus der Qlist\_2a
    - option\_list() holt die Antwortmöglichkeiten aus Qlist\_2a
    - Input für Kommentar wird erzeugt
    - output$save\_results speichert gegebene Antworten & Komm. in results2 und kommentare2
  + Teil2\_ent\_stetement 🡪 Text
    - output$save\_results speichert gegebene Antworten von results2 und kommentare2 in result\_coll$Alist\_2a
    - questionaryPostProcessing()
      * hole Alist aus result\_coll$Alist
      * relevant\_gaps berechne relevante Gaps für 2b und speichere dieses in result\_coll$relevant\_gaps (ev. schon im postProcessing von Teil1 möglich)
      * Calc\_survery\_question\_ids()
      * Erstellung von results2b & kommentare2b (falls noch auf NULL)

Render\_main\_panel\_2b.R

* helper-functionalities 🡪 questionaryPostProcessing() / output$save\_results
* output$mainPanel
  + Teil2b\_intro: 🡪 text
  + Teil2b\_first\_question:
    - Text unter Verwendung der relevant\_gaps / choices / results2b / kommentare2b
    - output$save\_results speichert gegebene Antworten von results2b und kommentare2b in
  + Teil2b\_ent-statement:
    - Text
    - output$save\_results speichert gegebene Antworten von results2b und kommentare2b in result\_coll$Alist\_2b
    - questionaryPostProcessing()
      * calc\_survey\_question\_ids()
      * Erstellung von results2c & kommentare2c (falls noch auf NULL)

Render\_main\_panel\_2c.R

* helper-functionalities 🡪 questionaryPostProcessing() / output$save\_results
* output$mainPanel
  + Teil2c\_intro: 🡪 text
  + Teil2c\_fragen\_beantworten:
    - Text unter Verwendung der results2c / kommentare2c
    - output$save\_results speichert gegebene Antworten von results2c und kommentare2c in
  + Teil2c\_ent\_statement:
    - Text
    - output$save\_results speichert gegebene Antworten von results2c und kommentare2c in result\_coll$Alist\_2c
    - questionaryPostProcessing() = leer

Render\_main\_panel\_3.R

* Render Rmd

**Achtung:**

* rule\_identify\_unzufriedenheiten() in rule\_extract\_unzufriedenheiten() sucht nur nach dem String "Unzufrieden" in den Gaps
* analog for rule\_identify\_...() 🡪 7 Regeln