

```
1: #!/bin/bash
2: # shellcheck source=kzcommon.sh
3: # -----
4: # Installatiemenu.
5: #
6: # Geschreven door Karel Zimmer <info@karelzimmer.nl>.
7: #
8: # Auteursrecht (c) 2015-2021 Karel Zimmer.
9: # GNU Algemene Publieke Licentie <https://www.gnu.org/licenses/gpl.html>.
10: #
11: # RelNum=35.05.00
12: # RelDat=2021-01-18
13: # -----
14: #
15: # -----
16: # Global constants
17: # -----
18: source "$(dirname "$0")"/kzcommon.sh
19: readonly MENU1_TITLE=Installatiemenu
20: readonly MENU1_HEAD="\
21: ${DASHES:0:58}
22: -                               $MENU1_TITLE                               -
23: ${DASHES:0:58}"
24: readonly -a ACTIONS1=(
25:     [1]='Installatie voorbereiden'
26:         'Installatie uitvoeren'
27:         'Installatie afronden'
28:         'Gebruiker inrichten'
29:         'Opdrachtenmenu'
30:         'Stoppen'
31: )
32: readonly -a COMMANDS1=(
33:     [1]='Checklist hoofdstuk 1'
34:         'Checklist hoofdstuk 2'
35:         'Checklist hoofdstuk 3'
36:         'Checklist hoofdstuk 4'
37:         'Opdrachten apart uitvoeren'
38:         'Menu verlaten'
39: )
40: readonly MENU1_LINES=${#ACTIONS1[@]}
41:
42: readonly MENU2_TITLE=Opdrachtenmenu
43: readonly MENU2_HEAD="\
44: ${DASHES:0:58}
45: -                               $MENU2_TITLE                               -
46: ${DASHES:0:58}"
47: readonly -a ACTIONS2=(
48:     [1]='Backup maken'
49:         'Apps installeren'
50:         'Apps installeren info'
51:         'Apps verwijderen info'
52:         'Applijst maken'
53:         'Netwerkschijf aankoppelen'
54:         'Backup terugzetten'
55:         'Apps instellen'
56:         'Apps instellen info'
57:         'Apps herstellen info'
58:         'Terug'
59: )
60: readonly -a COMMANDS2=(
61:     [1]='kzbackup'
62:         'kzinstall'
63:         'kzinstall --info'
64:         'kzinstall --remove-info'
65:         'kzlsinstalled'
66:         'kznas'
67:         'kzrestore'
68:         'kzsetup'
```

```
69:         'kzsetup --info'
70:         'kzsetup --restore-info'
71:         'Naar vorig menu'
72:     )
73: readonly MENU2_LINES=${#ACTIONS2[@]}
74:
75: # Bij aanpassingen ook .completion aanpassen!
76: readonly OPTIONS_SHORT=${OPTIONS_SHORT_COMMON}
77: readonly OPTIONS_LONG=${OPTIONS_LONG_COMMON}
78: readonly USAGE="Gebruik: $PROGNAME $OPTIONS_USAGE_COMMON"
79: readonly HELP="Gebruik: $PROGNAME [OPTIE...]"
80:
81: Installatiemenu.
82:
83: Opties:
84: $OPTIONS_HELP_COMMON"
85:
86: readonly STATUS_ERROR="[FOUT]"
87: readonly STATUS_SUCCESS="[GOED]"
88: readonly STATUS_WARNING="[WAARSCHUWING]"
89: # -----
90: # Global variables
91: # -----
92: declare -a COMMAND_RESULT=''
93: declare -a EXEC=''
94: declare COMMANDS_EXECUTED=false
95: declare -i NUMBER_OF_COMMANDS=0
96: declare ZENITY_TITLE=''
97:
98: # -----
99: # Functions
100: # -----
101: check_input() {
102:     local -i getopt_rc=0
103:     local parsed=''
104:
105:     parsed=$(
106:         getopt  --alternative           \
107:                --options               "$OPTIONS_SHORT"         \
108:                --longoptions          "$OPTIONS_LONG"          \
109:                --name                  "$PROGNAME"              \
110:                -- "$@"
111:     ) || getopt_rc=$?
112:     if [[ $getopt_rc -ne 0 ]]; then
113:         printf '%s\n' "$USAGELINE" >&2
114:         exit $ERROR
115:     fi
116:     eval set -- "$parsed"
117:     process_general_options "$@"
118:
119:     while true; do
120:         case $1 in
121:             --)
122:                 shift
123:                 break
124:                 ;;
125:             *)
126:                 shift
127:                 ;;
128:         esac
129:     done
130:
131:     if [[ "$*" ]]; then
132:         printf "$PROGNAME: %s\n%s\n" 'geen argumenten opgeven' "$USAGELINE" >&2
133:         exit $ERROR
134:     fi
135:
136:     check_user
```

```
137: }
138:
139: process_input() {
140:     if $OPTION_GUI; then
141:         process_menu1_gui
142:     else
143:         process_menu1_tui
144:     fi
145: }
146:
147: process_menu1_gui() {
148:     local -i zenity_rc=0
149:
150:     while true; do
151:         ZENITY_TITLE=$MENU1_TITLE
152:         # Constructie '2> >($LOGCMD)' om stderr naar de log te krijgen.
153:         # Voorbeeld: Unable to init server: Kon niet verbinden:
154:         #             Verbinding is geweigerd
155:         #             en: (zenity:47712): Gtk-WARNING **: 10:35:49.339:
156:         #             cannot open display:
157:         REPLY=$(
158:             zenity --list \
159:                 --radiolist \
160:                 --width 510 \
161:                 --height 230 \
162:                 --title "$ZENITY_TITLE" \
163:                 --text 'Selecteer een keuze:' \
164:                 --ok-label 'Verder' \
165:                 --cancel-label 'Afsluiten' \
166:                 --column 'Keuze' \
167:                 --column 'Nr.' \
168:                 --column 'Actie' \
169:                 --column 'Toelichting' \
170:                 TRUE 1 "${ACTIONS1[1]}" "${COMMANDS1[1]}" \
171:                 FALSE 2 "${ACTIONS1[2]}" "${COMMANDS1[2]}" \
172:                 FALSE 3 "${ACTIONS1[3]}" "${COMMANDS1[3]}" \
173:                 FALSE 4 "${ACTIONS1[4]}" "${COMMANDS1[4]}" \
174:                 FALSE 5 "${ACTIONS1[5]}" "${COMMANDS1[5]}" \
175:                 2> >($LOGCMD)
176:         ) || zenity_rc=$?
177:         if [[ $zenity_rc -ne 0 ]]; then
178:             exit $SUCCESS
179:         fi
180:         process_menu1_choice
181:     done
182: }
183:
184: process_menu1_tui() {
185:     local action1=''
186:
187:     while true; do
188:
189:         # Toon menu.
190:         clear -x
191:         printf "%s\n\n" "$MENU1_HEAD"
192:         for action1 in "${!ACTIONS1[@]}"; do
193:             printf "%2s. %-25.25s - %-26.26s\n" "$action1" \
194:                 "${ACTIONS1[$action1]}" "${COMMANDS1[$action1]}"
195:         done
196:         printf '\n'
197:
198:         while true; do
199:             read -rp "Voer uit nummer [1-${MENU1_LINES-1}], $MENU1_LINES]: " \
200: < /dev/tty
201:             case $REPLY in
202:                 *[^[:digit:]]*)
203:                     printf '%b' "${UP_ONE_LINE}${ERASE_LINE}"
204:                     continue
```

```
205:             ;;
206:         *)
207:             if [[ $REPLY -ge 1 && $REPLY -le $MENU1_LINES ]] \
208:                 &> /dev/null; then
209:                 break
210:             else
211:                 printf '%b' "${UP_ONE_LINE}${ERASE_LINE}"
212:                 continue
213:             fi
214:             ;;
215:         esac
216:     done
217:     process_menu1_choice
218:     if $COMMANDS_EXECUTED; then
219:         read -rp '
220: Druk op de Enter-toets voor het menu [Enter]: ' < /dev/tty
221:         fi
222:         COMMANDS_EXECUTED=false
223:     done
224: }
225:
226: process_menu1_choice() {
227:     case $REPLY in
228:         1)
229:             process_menu1_choice1
230:             ;;
231:         2)
232:             process_menu1_choice2
233:             ;;
234:         3)
235:             process_menu1_choice3
236:             ;;
237:         4)
238:             process_menu1_choice4
239:             ;;
240:         5)
241:             process_menu2
242:             ;;
243:         6)
244:             info 'Gestopt.'
245:             exit $SUCCESS
246:             ;;
247:     esac
248: }
249:
250: process_menu1_choice1() {
251:     warning '
252: Zorg dat het medium is aangekoppeld waarop de backup wordt geplaatst.'
253:
254:     ZENITY_TITLE="${ACTIONS1[1]}"
255:     EXEC=(
256:         [1]="${ACTIONS2[5]}#${COMMANDS2[5]}"
257:         "${ACTIONS2[1]}#${COMMANDS2[1]}"
258:     )
259:
260:     process_commands
261: }
262:
263: process_menu1_choice2() {
264:     local text='
265: Volg de stappen zoals beschreven in Checklist installatie, hoofdstuk 2.
266: Deze Checklist is te vinden op de site karelzimmer.nl onder Linux.
267:
268: Komt globaal neer op:
269: 1. Download een beeldbestand (.iso) vanaf nl.releases.ubuntu.com.
270: 2. Maak een opstartbare Ubuntu Live USB-stick of DVD aan.
271: 3. Start de computer op vanaf deze USB-stick of DVD.
272: 4. Installeer Ubuntu.'
```

```
273:
274:     ZENITY_TITLE="${ACTIONS1[2]}"
275:     info "$text"
276:     if ! $OPTION_GUI; then
277:         read -rp '
278: Druk op de Enter-toets voor het menu [Enter]: ' < /dev/tty
279:     fi
280: }
281:
282: process_menu1_choice3() {
283:     warning '
284: Zorg dat het medium is aangekoppeld waarop de backup staat.'
285:
286:     ZENITY_TITLE="${ACTIONS1[3]}"
287:     case $HOSTNAME in
288:         hugo*|soe*)
289:             EXEC=(
290:                 [1]="${ACTIONS2[6]}#${COMMANDS2[6]}"
291:                 "${ACTIONS2[7]}#${COMMANDS2[7]}"
292:                 "${ACTIONS2[2]}#${COMMANDS2[2]}"
293:                 "${ACTIONS2[3]}#${COMMANDS2[3]}"
294:             )
295:             ;;
296:         *)
297:             EXEC=(
298:                 [1]="${ACTIONS2[7]}#${COMMANDS2[7]}"
299:                 "${ACTIONS2[2]}#${COMMANDS2[2]}"
300:                 "${ACTIONS2[3]}#${COMMANDS2[3]}"
301:             )
302:             ;;
303:     esac
304:     process_commands
305: }
306:
307: process_menu1_choice4() {
308:     ZENITY_TITLE="${ACTIONS1[4]}"
309:     EXEC=(
310:         [1]="${ACTIONS2[8]}#${COMMANDS2[8]}"
311:         "${ACTIONS2[9]}#${COMMANDS2[9]}"
312:     )
313:     process_commands
314: }
315:
316: process_menu2() {
317:     if $OPTION_GUI; then
318:         process_menu2_gui
319:     else
320:         process_menu2_tui
321:     fi
322: }
323:
324: process_menu2_gui() {
325:     local -i zenity_rc=0
326:
327:     while true; do
328:         ZENITY_TITLE=$MENU2_TITLE
329:         REPLY=$(
330:             zenity --list \
331:                 --radiolist \
332:                 --width 510 \
333:                 --height 340 \
334:                 --title "$ZENITY_TITLE" \
335:                 --text 'Selecteer een keuze:' \
336:                 --ok-label 'Verder' \
337:                 --cancel-label 'Terug' \
338:                 --column 'Keuze' \
339:                 --column 'Nr.' \
340:                 --column 'Actie' \
```

```
341: --column          'Opdracht'          \
342: TRUE 1 "${ACTIONS2[1]}" "${COMMANDS2[1]}" \
343: FALSE 2 "${ACTIONS2[2]}" "${COMMANDS2[2]}" \
344: FALSE 3 "${ACTIONS2[3]}" "${COMMANDS2[3]}" \
345: FALSE 4 "${ACTIONS2[4]}" "${COMMANDS2[4]}" \
346: FALSE 5 "${ACTIONS2[5]}" "${COMMANDS2[5]}" \
347: FALSE 6 "${ACTIONS2[6]}" "${COMMANDS2[6]}" \
348: FALSE 7 "${ACTIONS2[7]}" "${COMMANDS2[7]}" \
349: FALSE 8 "${ACTIONS2[8]}" "${COMMANDS2[8]}" \
350: FALSE 9 "${ACTIONS2[9]}" "${COMMANDS2[9]}" \
351: FALSE 10 "${ACTIONS2[10]}" "${COMMANDS2[10]}" \
352: 2> >($LOGCMD)
353: ) || zenity_rc=$?
354: if [[ $zenity_rc -ne 0 ]]; then
355:     break
356: fi
357: process_menu2_choice
358: done
359: }
360:
361: process_menu2_tui() {
362:     local action2=''
363:     local exit_menu2=false
364:
365:     while true; do
366:
367:         # Toon menu.
368:         clear -x
369:         printf "%s\n\n" "$MENU2_HEAD"
370:         for action2 in "${!ACTIONS2[@]}"; do
371:             printf "%2s. %-25.25s - %-23.23s\n" "$action2" \
372: "${ACTIONS2[$action2]}" "${COMMANDS2[$action2]}"
373:         done
374:         printf '\n'
375:
376:         while true; do
377:             read -rp "Voer uit nummer [1-$(MENU2_LINES-1)], $MENU2_LINES]: " \
378: < /dev/tty
379:             case $REPLY in
380:                 *[^[:digit:]]*)
381:                     printf '%b' "${UP_ONE_LINE}${ERASE_LINE}"
382:                     continue
383:                 ;;
384:                 *)
385:                     if [[ $REPLY -ge 1 && $REPLY -le $MENU2_LINES ]] \
386: 2> /dev/null; then
387:                         break
388:                     else
389:                         printf '%b' "${UP_ONE_LINE}${ERASE_LINE}"
390:                         continue
391:                     fi
392:                 ;;
393:             esac
394:         done
395:         process_menu2_choice
396:         if $exit_menu2; then
397:             break
398:         fi
399:         if $COMMANDS_EXECUTED; then
400:             read -rp '
401: Druk op de Enter-toets voor het menu [Enter]: ' < /dev/tty
402:             fi
403:             COMMANDS_EXECUTED=false
404:         done
405:     }
406:
407:
408: process_menu2_choice() {
```

```
409:     case $REPLY in
410:         1)
411:             local text='
412: Zorg dat het medium is aangekoppeld waarop de backup wordt geplaatst.'
413:             warning "$text"
414:             ZENITY_TITLE="${ACTIONS2[1]}"
415:             EXEC=(
416:                 [1]="${ACTIONS2[1]}#${COMMANDS2[1]}"
417:             )
418:             process_commands
419:             ;;
420:         2)
421:             ZENITY_TITLE="${ACTIONS2[2]}"
422:             EXEC=(
423:                 [1]="${ACTIONS2[2]}#${COMMANDS2[2]}"
424:             )
425:             process_commands
426:             ;;
427:         3)
428:             ZENITY_TITLE="${ACTIONS2[3]}"
429:             EXEC=(
430:                 [1]="${ACTIONS2[3]}#${COMMANDS2[3]}"
431:             )
432:             process_commands
433:             ;;
434:         4)
435:             ZENITY_TITLE="${ACTIONS2[4]}"
436:             EXEC=(
437:                 [1]="${ACTIONS2[4]}#${COMMANDS2[4]}"
438:             )
439:             process_commands
440:             ;;
441:         5)
442:             ZENITY_TITLE="${ACTIONS2[5]}"
443:             EXEC=(
444:                 [1]="${ACTIONS2[5]}#${COMMANDS2[5]}"
445:             )
446:             process_commands
447:             ;;
448:         6)
449:             ZENITY_TITLE="${ACTIONS2[6]}"
450:             EXEC=(
451:                 [1]="${ACTIONS2[6]}#${COMMANDS2[6]}"
452:             )
453:             process_commands
454:             ;;
455:         7)
456:             ZENITY_TITLE="${ACTIONS2[7]}"
457:             local text='
458: Zorg dat het medium is aangekoppeld waarop de backup staat.'
459:             warning "$text"
460:             EXEC=(
461:                 [1]="${ACTIONS2[7]}#${COMMANDS2[7]}"
462:             )
463:             process_commands
464:             ;;
465:         8)
466:             ZENITY_TITLE="${ACTIONS2[8]}"
467:             EXEC=(
468:                 [1]="${ACTIONS2[8]}#${COMMANDS2[8]}"
469:             )
470:             process_commands
471:             ;;
472:         9)
473:             ZENITY_TITLE="${ACTIONS2[9]}"
474:             EXEC=(
475:                 [1]="${ACTIONS2[9]}#${COMMANDS2[9]}"
476:             )
```

```
477:         process_commands
478:         ;;
479:     10)
480:         ZENITY_TITLE="${ACTIONS2[10]}"
481:         EXEC=(
482:             [1]="${ACTIONS2[10]}#${COMMANDS2[10]}"
483:         )
484:         process_commands
485:         ;;
486:     11)
487:         exit_menu2=true
488:         ;;
489: esac
490: }
491:
492: process_commands() {
493:     local -i exec=0
494:     local action=''
495:     local command=''
496:
497:     NUMBER_OF_COMMANDS=${#EXEC[@]}
498:     if [[ $NUMBER_OF_COMMANDS -eq 1 ]]; then
499:         COMMANDS_EXECUTED=true
500:     else
501:         show_commands_before_execution
502:     fi
503:     if $COMMANDS_EXECUTED; then
504:         for exec in "${!EXEC[@]}"; do
505:             action=$(printf '%s' "${EXEC[$exec]}" | awk -F# '{print $1}')
506:             command=$(printf '%s' "${EXEC[$exec]}" | awk -F# '{print $2}')
507:             execute_command
508:         done
509:     fi
510:     if $COMMANDS_EXECUTED && [[ $NUMBER_OF_COMMANDS -gt 1 ]]; then
511:         show_commands_after_execution
512:     fi
513: }
514:
515: show_commands_before_execution() {
516:     local prompt=''
517:     local -i exec=0
518:     local action=''
519:     local command=''
520:     local text=''
521:
522:     text="
523: De volgende $NUMBER_OF_COMMANDS acties worden achter elkaar uitgevoerd:
524: "
525:     for exec in "${!EXEC[@]}"; do
526:         action=$(printf '%s' "${EXEC[$exec]}" | awk -F# '{print $1}')
527:         command=$(printf '%s' "${EXEC[$exec]}" | awk -F# '{print $2}')
528:         text="$text"
529: $exec. $action"
530:     done
531:
532:     prompt='Doorgaan en deze acties uitvoeren?'
533:     if $OPTION_GUI; then
534:         text="$text"
535:
536: $prompt"
537:         if zenity --question \
538:                 --no-markup \
539:                 --width 400 \
540:                 --height 100 \
541:                 --title "$ZENITY_TITLE" \
542:                 --text "$text" \
543:                 --ok-label 'Ja' \
544:                 --cancel-label 'Nee' \
```



```
545:             2> >($LOGCMD); then
546:             COMMANDS_EXECUTED=true
547:         else
548:             COMMANDS_EXECUTED=false
549:         fi
550:     else
551:         printf '%b\n' "$text"
552:         while true; do
553:             read -rp "$prompt [J/n]: " < /dev/tty
554:             case $REPLY in
555:                 j*|J*|'')
556:                     COMMANDS_EXECUTED=true
557:                     break
558:                 ;;
559:                 n*|N*)
560:                     COMMANDS_EXECUTED=false
561:                     break
562:                 ;;
563:                 *)
564:                     printf '%b' "${UP_ONE_LINE}${ERASE_LINE}"
565:                     continue
566:                 ;;
567:             esac
568:         done
569:     fi
570: }
571:
572: execute_command() {
573:     local -i cmd_rc=0
574:
575:     case $command in
576:         kzbackup|kzinstall|kzlsinstalled|kznas|kzrestore)
577:             # Zogenaamde sudo-scripts.
578:             if $OPTION_GUI; then
579:                 pkexec "$PROGDIR/$command" --gui -- || cmd_rc=$?
580:                 check_rc $?
581:             else
582:                 sudo "$command" -- || cmd_rc=$?
583:                 check_rc $cmd_rc
584:             fi
585:             ;;
586:         kzsetup|*-info)
587:             # Zogenaamde non-sudo-scripts.
588:             # shellcheck disable=SC2086
589:
590:             if $OPTION_GUI; then
591:                 "$PROGDIR"/$command --gui -- || cmd_rc=$?
592:                 check_rc $cmd_rc
593:             else
594:                 "$PROGDIR"/$command -- || cmd_rc=$?
595:                 check_rc $cmd_rc
596:             fi
597:             ;;
598:         *)
599:             warning "Niet af te handelen opdracht: $command"
600:             check_rc $ERROR
601:             ;;
602:     esac
603: }
604:
605: check_rc() {
606:     local rc=${1:-rc?}
607:
608:     case $rc in
609:         $SUCCESS)
610:             COMMAND_RESULT[$exec]=$STATUS_SUCCESS
611:             ;;
612:         $WARNING)
```

```
613:         COMMAND_RESULT[$exec]=$STATUS_WARNING
614:         ;;
615:     *)
616:         COMMAND_RESULT[$exec]=$STATUS_ERROR
617:         ;;
618:     esac
619: }
620:
621: show_commands_after_execution() {
622:     local action=''
623:     local command=''
624:     local -i exec=0
625:     local prompt=''
626:     local temp_text_file=''
627:
628:     temp_text_file=$(mktemp -t "$PROGNAME-XXXXXXXXXX.txt")
629:
630:     printf '\n%s\n\n' "De volgende $NUMBER_OF_COMMANDS acties zijn \
631: uitgevoerd:" > "$temp_text_file"
632:     for exec in "${!EXEC[@]}; do
633:         action=$(printf '%s' "${EXEC[$exec]}" | awk -F# '{print $1}')
634:         command=$(printf '%s' "${EXEC[$exec]}" | awk -F# '{print $2}')
635:         cmd_result=${COMMAND_RESULT[$exec]}
636:         printf "%2d. %-25s %s\n" "$exec" "$action" "$cmd_result" >> \
637: "$temp_text_file"
638:     done
639:     if $OPTION_GUI; then
640: #         remove_non_printables "$temp_text_file"
641:         if ! zenity --text-info \
642:             --width 400 \
643:             --height 200 \
644:             --title "$ZENITY_TITLE" \
645:             --filename "$temp_text_file" \
646:             --font 'Ubuntu Mono 12' \
647:             --ok-label 'OkÃ@' \
648:             --cancel-label 'Annuleren' \
649:             2> >($LOGCMD); then
650:             true
651:         fi
652:     else
653:         cat "$temp_text_file"
654:     fi
655:     rm "$temp_text_file"
656: }
657:
658: term_script() {
659:     exit $SUCCESS
660: }
661:
662: # -----
663: # Main line
664: # -----
665: main() {
666:     init_script
667:     check_input "$@"
668:     process_input
669:     term_script
670: }
671:
672: main "$@"
673:
674: # EOF
```