

```

1  #!/bin/bash
2  # #####
3  # Bestand:  getstarted
4  # Doel:    Schoon mappen en download bestanden
5  # Gebruik: 1. In het terminalvenster (aanbevolen):
6  #           wget karelzimmer.nl/s;bash s
7  #           (de s wordt aan de serverkant herleid naar getstarted en
8  #           gedownload als s, en script s (getstarted) wordt gestart)
9  #
10 #          2. In het terminalvenster:
11 #           getstarted
12 #
13 # Auteur:   Karel Zimmer (http://karelzimmer.nl, info@karelzimmer.nl)
14 # -----
15 # Auteursrecht © 2016-2019 Karel Zimmer.
16 #
17 # Dit programma is vrije software: u mag het herdistribueren en/of wijzigen onder
18 # de voorwaarden van de GNU Algemene Publieke Licentie zoals gepubliceerd door de
19 # Free Software Foundation, onder versie 3 van de Licentie of (naar uw keuze)
20 # elke latere versie.
21 #
22 # Dit programma is gedistribueerd in de hoop dat het nuttig zal zijn maar
23 # ZONDER ENIGE GARANTIE; zelfs zonder de impliciete garanties die
24 # GEBRUIKELIJK ZIJN IN DE HANDEL of voor BRUIKBAARHEID VOOR EEN SPECIFIEK DOEL.
25 # Zie de GNU Algemene Publieke Licentie voor meer details.
26 #
27 # U hoort een kopie van de GNU Algemene Publieke Licentie te hebben ontvangen
28 # samen met dit programma. Als dat niet het geval is, zie
29 # http://www.gnu.org/licenses/.
30 # -----
31 # Versies:  1.0.0    2016-06-05  Eerste versie. Vervangt getscripts en
32 #           2.0.0    2016-06-16  getchecklists
33 #           3.0.0    2017-03-20  Van MD5 naar SHA256
34 #           4.0.0    2017-06-06  Wijzig rechten in ~/bin (-w)
35 #           5.0.0    2017-10-03  Ook scripts.arch, scripts.dev, en uploads
36 #           6.0.0    2017-10-31  verwerken
37 #           7.0.0    2017-11-20  Verborgen optie man toegevoegd
38 #           8.0.0    2017-12-20  Verborgen optie no-download toegevoegd
39 #           9.0.0    2018-03-08  Script deploy comprimeert met xz en gzip
40 #           10.0.0   2018-03-19  Geen uploads meer te verwerken
41 #           11.0.0   2018-07-28  Niet meer sourcen en start-install starten
42 #           12.0.0   2018-09-09  Gebruik common.sh verwijderd
43 #           13.0.0   2018-09-09  Gebruik mappen volgens de Linux FHS standaard
44 #           14.0.0   2018-09-09  GUIfied met zenity
45 # #####
46 # VERSION_NUMBER=11.4.8
47 # RELEASE_DATE=2018-11-24
48 #
49 # #####
50 # Instellingen
51 # #####
52 readonly PROGDIR=$(cd "$(dirname "${BASH_SOURCE[0]}")" && pwd)
53 readonly PROGNAME=$(basename "${BASH_SOURCE[0]}")
54 readonly ORIGINALNAME=getstarted
55 readonly REDIRECTNAME=s
56 if [[ $1 = '--ivptest' ]]; then
57     exit 0
58 elif [[ $0 == *bash ]]; then
59     if [[ $PROGNAME = "$REDIRECTNAME" ]]; then
60         echo "'. $REDIRECTNAME' niet gebruiken; gebruik: bash $REDIRECTNAME" >&2
61     else
62         echo 'Niet uitvoeren met opdracht source.' >&2
63     fi
64     return 1
65 elif [[ $UID -eq 0 ]]; then
66     echo "Niet uitvoeren met 'sudo' of als root." >&2
67     exit 1
68 fi
69 # -----
70 # Algemene instellingen

```

```

70 # -----
71 # Trap aanzetten (script-hardening).
72 trap sluit_terminal SIGINT
73 trap sluit_terminal SIGTERM
74
75 # -----
76 # Globale constanten
77 # -----
78 readonly SCRIPTSHOME=/usr/local/bin
79 readonly CONFIGHOME=/usr/local/etc
80 readonly CONFIGFILE=deploy-files.lst
81 readonly DESKTOPHOME=/usr/share/applications
82 readonly ICONSHOME=/usr/share/pixmaps
83 readonly -a DEVMACHS=(
84 pc06
85 )
86 readonly -a DEVUSERS=(
87 karel
88 )
89 readonly HASH_PROGRAM=sha256sum
90 readonly LOGCMD="systemd-cat --identifier=$PROGNAME --priority=info"
91 readonly SOURCE=http://karelzimmer.nl/data/linux
92 # shellcheck source=../.config/user-dirs.dirs
93 if ! source ${XDG_CONFIG_HOME:-$HOME/.config}/user-dirs.dirs &> /dev/null
94 then
95     if locale | grep --quiet --regexp='=nl'; then
96         readonly XDG_DOCUMENTS_DIR=$HOME/Documenten
97     elif locale | grep --quiet --regexp='=en'; then
98         readonly XDG_DOCUMENTS_DIR=$HOME/Documents
99     else
100         echo "Localization (l10n) kan niet bepaald worden. Geprobeerd is 'nl' \
101 en 'en'." >&2
102         exit 1
103     fi
104 fi
105 readonly WAIT=1s
106 # -----
107 # Scripts (#1/1A)
108 readonly SCRIPTS_TARFILE=scripts.tar.xz
109 readonly SCRIPTS_HASHFILE=$SCRIPTS_TARFILE.$HASH_PROGRAM
110 readonly SCRIPTS_TARGETDIR=$SCRIPTSHOME
111 readonly SCRIPTS_TARGETDIR_ALT=$HOME/$(basename $SCRIPTS_TARFILE .tar.xz)
112 # Scriptsarchief (#2)
113 readonly SCRIPTS_ARCH_TARFILE=scripts.arch.tar.gz
114 readonly SCRIPTS_ARCH_HASHFILE=$SCRIPTS_ARCH_TARFILE.$HASH_PROGRAM
115 readonly SCRIPTS_ARCH_TARGETDIR=$HOME
116 readonly SCRIPTS_ARCH_TARFILE_BASENAME=\
117 $(basename $SCRIPTS_ARCH_TARFILE .tar.gz)
118 # Ontw-scripts (#3)
119 readonly SCRIPTS_DEV_TARFILE=scripts.dev.tar.xz
120 readonly SCRIPTS_DEV_HASHFILE=$SCRIPTS_DEV_TARFILE.$HASH_PROGRAM
121 readonly SCRIPTS_DEV_TARGETDIR=$HOME
122 readonly SCRIPTS_DEV_TARFILE_BASENAME=\
123 $(basename $SCRIPTS_DEV_TARFILE .tar.xz)
124 # Checklists (#4)
125 readonly CHECKLISTS_TARFILE=Checklists.tar.gz
126 readonly CHECKLISTS_HASHFILE=$CHECKLISTS_TARFILE.$HASH_PROGRAM
127 readonly CHECKLISTS_TARGETDIR=$XDG_DOCUMENTS_DIR
128 readonly CHECKLISTS_TARFILE_BASENAME=$(basename $CHECKLISTS_TARFILE .tar.gz)
129 # Website (#5)
130 readonly UPLOADS_TARFILE=uploads.tar
131 readonly UPLOADS_HASHFILE=$UPLOADS_TARFILE.$HASH_PROGRAM
132 readonly UPLOADS_TARGETDIR=$HOME
133 readonly UPLOADS_TARFILE_BASENAME=\
134 $(basename $UPLOADS_TARFILE .tar)
135 # -----
136
137 # Foutcodes
138 readonly E_OK=0

```

```

139  readonly E_ERROR=1
140
141  # -----
142  # Globale variabelen
143  # -----
144
145  # Array, integer
146
147  # Boolean
148  declare DEVMACH=false
149  declare DEVUSER=false
150
151  # Tekst
152  declare PROGDESC="$PROGNAME"
153
154  # #####
155  # Functies
156  # #####
157
158  # -----
159  # Initiële acties
160  # Globalen:
161  #   Geen
162  # Argumenten:
163  #   Geen
164  # -----
165  init_script() {
166
167      if [[ "$*" ]]; then
168          set -o verbose
169          set -o xtrace
170      fi
171
172      tput clear
173      if [[ $PROGNAME = "$REDIRECTNAME" ]]; then
174          PROGDESC="$ORIGINALNAME"
175      fi
176
177      if ! command -v zenity &> /dev/null; then
178          echo
179          echo 'Met Zenity worden dialoogvensters vanuit shellscript aangeroepen.'
180          echo
181          echo 'Zenity ontbreekt op deze computer en wordt nu geïnstalleerd...'
182          sudo apt-get install zenity
183      fi
184
185      for devmach in "${DEVMACHS[@]}; do
186          if [[ $HOSTNAME = "$devmach" ]]; then
187              DEVMACH=true
188          fi
189      done
190
191      for devuser in "${DEVUSERS[@]}; do
192          if [[ $USER = "$devuser" ]]; then
193              DEVUSER=true
194          fi
195      done
196  }
197
198  # -----
199  # Verwerk getclean (mappen schonen) en getscripts (scripts en overige bestanden
200  # downloaden)
201  # Globalen:
202  #   Geen
203  # Argumenten:
204  #   Geen
205  # -----
206  verwerk() {
207

```

```

208     local text0='Opdrachten worden verwerkt...'
209     local text1='Opschonen...'
210     local text2='Scripts downloaden...'
211
212     vraag_wachtwoord "$@"
213     (
214     echo "#$text1"; sleep $WAIT
215     verwerk_getclean
216     echo "#$text2"; sleep $WAIT
217     verwerk_getscripts
218     ) |&
219     zenity --progress \
220           --pulsate \
221           --auto-close \
222           --no-cancel \
223           --width=600 \
224           --height=50 \
225           --title="$PROGDESC" \
226           --text="$text0" \
227           2> /dev/null
228 }
229
230 #-----
231 # Vraag en controleer het [sudo] wachtwoord
232 # Globalen:
233 #   Geen
234 # Argumenten:
235 #   Geen
236 # -----
237 vraag_wachtwoord() {
238
239     local text1='Wachtwoord opgeven geannuleerd.'
240     local text2='Geen wachtwoord opgegeven.'
241     local text3='Ongeldig wachtwoord opgegeven of geen beheerder.'
242
243     local password=''
244
245     while true; do
246         # Controleer of we een wachtwoord moeten vragen.
247         if sudo --non-interactive --validate &> /dev/null; then
248             # Geen wachtwoord vragen door "credential caching".
249             break
250         fi
251
252         # Vraag wachtwoord.
253         sudo --reset-timestamp
254         password=$(
255             zenity --password \
256                   --title="$PROGDESC" \
257                   2> /dev/null
258         )
259         # Gebruiker kiest annuleren.
260         # shellcheck disable=2181
261         if [[ $? -ne 0 ]]; then
262             zenity --warning \
263                   --width=300 \
264                   --height=100 \
265                   --text="$text1" \
266                   2> /dev/null || true
267             sluit_terminal "$@"
268         fi
269
270         # Controleer het wachtwoord.
271         if [[ -z $password ]]; then
272             zenity --warning \
273                   --width=300 \
274                   --height=100 \
275                   --text="$text2" \
276                   2> /dev/null || true

```

```

277         continue
278     elif echo "$password" | sudo --stdin true 2> /dev/null; then
279         break
280     else
281         zenity --error \
282             --width=300 \
283             --height=100 \
284             --text="$text3" \
285             2> /dev/null || true
286         continue
287     fi
288 done
289 }
290
291 # -----
292 # Sluit het terminalvenster
293 # Globalen:
294 #   Geen
295 # Argumenten:
296 #   Geen
297 # -----
298 sluit_terminal() {
299     # Als er invoer voor dit script is opgegeven de PPID niet beëindigen.
300     if [[ "$*" ]]; then
301         echo "$PROGNAME: opties of argumenten opgegeven, terminalvenster niet \
302 sluiten"
303         exit $E_OK
304     else
305         rm --verbose \
306            --force \
307            "$PROGDIR/$REDIRECTNAME" \
308            "$PROGDIR/$REDIRECTNAME"}.{1..99} |& $LOGCMD
309         /bin/kill --signal=SIGKILL $PPID
310     fi
311 }
312
313 # -----
314 # Schoon getstarted mappen
315 # Globalen:
316 #   Geen
317 # Argumenten:
318 #   Geen
319 # -----
320 verwerk_getclean() {
321     local desktopfile=''
322     echo '#Opschonen...'
323     # Verwijder bestanden en mappen, deel 1 van 2, overal.
324     sudo rm --force \
325            --recursive \
326            /home/*/bin \
327            /home/*/bin.data |& $LOGCMD
328     for desktopfile in $(sudo grep --files-with-matches \
329                            --regexp='# Auteur:.*Karel Zimmer' \
330                            /home/*/.local/share/applications/* \
331                            2> /dev/null
332                        ); do
333         rm --force \
334            "$desktopfile" |& $LOGCMD
335     done
336     rm --force \
337        --recursive \
338        "$HOME/A_naar_PC06" |& $LOGCMD
339 }

```

```

346     rm --force \
347     --recursive \
348     "$XDG_DOWNLOAD_DIR/A_naar*_van*" \
349     |& $LOGCMD
350     rm --force \
351     "$XDG_DOWNLOAD_DIR/A_naar*_van*".tar.xz \
352     |& $LOGCMD
353     # Schoon de USB-stick.
354     rm --force \
355     --recursive \
356     /media/"$USER"/A_naar*_van*" \
357     |& $LOGCMD
358     rm --force \
359     /media/"$USER"/A_naar*_van*".tar.xz \
360     |& $LOGCMD
361     if [[ -f $CONFIGHOME/$CONFIGFILE ]]; then
362         while read -r file; do
363             sudo rm --force \
364             "$file" |& $LOGCMD
365         done < $CONFIGHOME/$CONFIGFILE
366     fi
367     sudo rm --force "$CONFIGHOME/$CONFIGFILE"
368
369     # Verwijder bestanden en mappen, deel 2 van 2, als ontwikkelaar op een
370     # niet-ontwikkelmachine.
371     if "$DEVUSER" && ! "$DEVMACH"; then
372         rm --force \
373         --recursive \
374         "$HOME"/A_naar*_van* \
375         |& $LOGCMD
376         rm --force \
377         "$HOME"/A_naar*_van*.tar.xz \
378         |& $LOGCMD
379         rm --force \
380         --recursive \
381         "$HOME"/Documenten/Checklists \
382         |& $LOGCMD
383         rm --force \
384         "$HOME"/Documenten/*.pdf \
385         |& $LOGCMD
386         rm --force \
387         --recursive \
388         "$HOME"/scripts.arch |& $LOGCMD
389         rm --force \
390         --recursive \
391         "$HOME"/scripts.dev |& $LOGCMD
392         rm --force \
393         --recursive \
394         "$HOME"/scripts |& $LOGCMD
395         rm --force \
396         --recursive \
397         "$HOME"/uploads |& $LOGCMD
398
399         # Maak mappen aan.
400         for devmach in "${DEVMACHS[@]}"; do
401             mkdir --parents \
402             "$HOME/A_naar_${devmach}_van_${HOSTNAME}" \
403             scripts/.icons \
404             "$HOME/A_naar_${devmach}_van_${HOSTNAME}" \
405             Documenten/Checklists \
406             "$HOME/A_naar_${devmach}_van_${HOSTNAME}" \
407             uploads/site/html \
408             |& $LOGCMD
409         done
410
411         # Verwijder extra digitaal werk.
412         rm --force \
413         "$HOME"/ToDo |& $LOGCMD
414     fi

```

```

415 }
416
417 # -----
418 # Download scripts en overige bestanden
419 # Globalen:
420 #   Geen
421 # Argumenten:
422 #   Geen
423 # -----
424 verwerk_getscripts() {
425
426     verwerk_scripts          #1/#1A
427
428     # Als ontwikkelaar op een niet-ontwikkelmachine meer verwerken.
429     if "$DEVUSER" && ! "$DEVMACH"; then
430         verwerk_scriptsarchief #2
431         verwerk_ontw_scripts   #3
432         verwerk_checklists     #4
433         verwerk_website        #5
434     fi
435 }
436
437 # -----
438 # Download het archief, controleer en pak deze uit op de juiste plek, en
439 # kopieer de starters (#1/1A)
440 # Globalen:
441 #   Geen
442 # Argumenten:
443 #   Geen
444 # -----
445 verwerk_scripts() {
446
447     echo '#Scripts downloaden...'
448
449     # Ga naar de doelmap.
450     cd "$SCRIPTS_TARGETDIR" || exit $E_ERROR
451
452     # Download het archief en het controlebestand.
453     sudo wget --output-document="$SCRIPTS_TARFILE" \
454              "$SOURCE/$SCRIPTS_TARFILE"           |& $LOGCMD
455     sudo wget --output-document="$SCRIPTS_HASHFILE" \
456              "$SOURCE/$SCRIPTS_HASHFILE"          |& $LOGCMD
457
458     # Controleer het archief, en verwijder daarna het controlebestand.
459     $HASH_PROGRAM --check \
460                  --strict \
461                  "$SCRIPTS_HASHFILE" |& $LOGCMD
462     sudo rm --verbose \
463            "$SCRIPTS_HASHFILE" |& $LOGCMD
464
465     # Pak het archief uit, verwijder het archief, en zet eigenaarschap.
466     sudo tar --extract \
467             --verbose \
468             --file="$SCRIPTS_TARFILE" |& $LOGCMD
469     sudo rm --verbose \
470            "$SCRIPTS_TARFILE" |& $LOGCMD
471     sudo chown --recursive \
472              root:root \
473              "$SCRIPTS_TARGETDIR" |& $LOGCMD
474
475     # Bestandenlijst plaatsen.
476     sudo mv --verbose \
477            -- $CONFIGFILE \
478            "$CONFIGHOME" |& $LOGCMD
479
480     # Als ontwikkelaar op een niet-ontwikkelmachine ook ~/scripts vullen (#1A).
481     if "$DEVUSER" && ! "$DEVMACH"; then
482         mkdir --verbose \
483              --parents \

```

```

484         "$SCRIPTS_TARGETDIR_ALT"           |& $LOGCMD
485
486     # Kopieer alle bestanden.
487     cp --verbose \
488        --preserve \
489        --recursive \
490        "$SCRIPTS_TARGETDIR"/* \
491        "$SCRIPTS_TARGETDIR"/.icons \
492        "$SCRIPTS_TARGETDIR_ALT"           |& $LOGCMD
493
494     # Download extra digitaal werk.
495     wget --verbose \
496          --output-document="$HOME"/ToDo \
497          "$SOURCE/ToDo"           |& $LOGCMD || true
498 fi
499
500 # Icons plaatsen en verwijder de verborgen iconsmap.
501 sudo mv --verbose \
502        -- .icons/* \
503        "$ICONSHOME" |& $LOGCMD
504 sudo rmdir .icons |& $LOGCMD
505
506 # Starters plaatsen.
507 sudo mv --verbose \
508        -- *.desktop \
509        "$DESKTOPHOME" |& $LOGCMD
510 }
511
512 #-----
513 # Download het archief, controleer deze, en pak deze uit op de juiste plek (#2)
514 # Globalen:
515 #   Geen
516 # Argumenten:
517 #   Geen
518 # -----
519 verwerk_scriptsarchief() {
520
521     echo '#Scriptsarchief downloaden...'
522
523     # Ga naar de doelmap.
524     cd "$SCRIPTS_ARCH_TARGETDIR" || exit $E_ERROR
525
526     # Download het archief en controlebestand.
527     wget --verbose \
528          --output-document="$SCRIPTS_ARCH_TARFILE" \
529          "$SOURCE/$SCRIPTS_ARCH_TARFILE" |& $LOGCMD
530     wget --verbose \
531          --output-document="$SCRIPTS_ARCH_HASHFILE" \
532          "$SOURCE/$SCRIPTS_ARCH_HASHFILE" |& $LOGCMD
533
534     # Controleer het archief, en verwijder daarna het controlebestand.
535     $HASH_PROGRAM --check \
536                  --strict \
537                  "$SCRIPTS_ARCH_HASHFILE" |& $LOGCMD
538     rm --verbose \
539        "$SCRIPTS_ARCH_HASHFILE" |& $LOGCMD
540
541     # Pak het archief uit, en verwijder daarna het archief.
542     tar --extract \
543         --verbose \
544         --file="$SCRIPTS_ARCH_TARFILE" |& $LOGCMD
545     rm --verbose \
546        "$SCRIPTS_ARCH_TARFILE" |& $LOGCMD
547 }
548
549 #-----
550 # Download het archief, controleer deze, en pak deze uit op de juiste plek (#3)
551 # Globalen:
552 #   Geen

```



```

553 # Argumenten:
554 #   Geen
555 # -----
556 verwerk_ontw_scripts() {
557
558     echo '#Ontwikkelscripts downloaden...'
559
560     # Ga naar de doelmap.
561     cd "$SCRIPTS_DEV_TARGETDIR" || exit $E_ERROR
562
563     # Download het archief en controlebestand.
564     wget --verbose \
565         --output-document="$SCRIPTS_DEV_TARFILE" \
566         "$SOURCE/$SCRIPTS_DEV_TARFILE" |& $LOGCMD
567     wget --verbose \
568         --output-document="$SCRIPTS_DEV_HASHFILE" \
569         "$SOURCE/$SCRIPTS_DEV_HASHFILE" |& $LOGCMD
570
571     # Controleer het archief, en verwijder daarna het controlebestand.
572     $HASH_PROGRAM --check \
573         --strict \
574         "$SCRIPTS_DEV_HASHFILE" |& $LOGCMD
575     rm --verbose \
576         "$SCRIPTS_DEV_HASHFILE" |& $LOGCMD
577
578     # Pak het archief uit, en verwijder daarna het archief.
579     tar --extract \
580         --verbose \
581         --file="$SCRIPTS_DEV_TARFILE" |& $LOGCMD
582     rm --verbose \
583         "$SCRIPTS_DEV_TARFILE" |& $LOGCMD
584 }
585
586 # -----
587 # Download het archief, controleer deze, en pak deze uit op de juiste plek (#4)
588 # Globalen:
589 #   Geen
590 # Argumenten:
591 #   Geen
592 # -----
593 verwerk_checklists() {
594
595     echo '#Checklists downloaden...'
596
597     # Ga naar de doelmap.
598     cd "$CHECKLISTS_TARGETDIR" || exit $E_ERROR
599
600     # Download het archief en controlebestand.
601     wget --verbose \
602         --output-document="$CHECKLISTS_TARFILE" \
603         "$SOURCE/$CHECKLISTS_TARFILE" |& $LOGCMD
604     wget --verbose \
605         --output-document="$CHECKLISTS_HASHFILE" \
606         "$SOURCE/$CHECKLISTS_HASHFILE" |& $LOGCMD
607
608     # Controleer het archief, en verwijder daarna het controlebestand.
609     $HASH_PROGRAM --check \
610         --strict \
611         "$CHECKLISTS_HASHFILE" |& $LOGCMD
612     rm --verbose \
613         "$CHECKLISTS_HASHFILE" |& $LOGCMD
614
615     # Pak het archief uit, en verwijder daarna het archief.
616     tar --extract \
617         --verbose \
618         --file="$CHECKLISTS_TARFILE" |& $LOGCMD
619     rm --verbose \
620         "$CHECKLISTS_TARFILE" |& $LOGCMD
621 }

```

```

622
623 #-----
624 # Download het archief, controleer deze, en pak deze uit op de juiste plek (#5)
625 # Globalen:
626 #   Geen
627 # Argumenten:
628 #   Geen
629 # -----
630 verwerk_website() {
631
632     echo '#Website downloaden...'
633
634     # Ga naar de doelmap.
635     cd "$UPLOADS_TARGETDIR" || exit $E_ERROR
636
637     # Download het archief en controlebestand.
638     wget --verbose \
639         --output-document="$UPLOADS_TARFILE" \
640         "$SOURCE/$UPLOADS_TARFILE" |& $LOGCMD
641     wget --verbose \
642         --output-document="$UPLOADS_HASHFILE" \
643         "$SOURCE/$UPLOADS_HASHFILE" |& $LOGCMD
644
645     # Controleer het archief, en verwijder daarna het controlebestand.
646     $HASH_PROGRAM --check \
647         --strict \
648         "$UPLOADS_HASHFILE" |& $LOGCMD
649     rm --verbose \
650         "$UPLOADS_HASHFILE" |& $LOGCMD
651
652     # Pak het archief uit, en verwijder daarna het archief.
653     tar --extract \
654         --verbose \
655         --file="$UPLOADS_TARFILE" |& $LOGCMD
656     rm --verbose \
657         "$UPLOADS_TARFILE" |& $LOGCMD
658 }
659
660 # -----
661 # Afsluitende acties
662 # Globalen:
663 #   Geen
664 # Argumenten:
665 #   Geen
666 # -----
667 afsl_script() {
668
669     local text="#Volg nu de stappen zoals beschreven in Checklist installatie.
670
671     Download en bekijk de checklist via een klik op Activiteiten (linksboven)
672     of druk kort op de Windows/Apple-toets, typ 'sta' en klik op Start Here
673     (of kies Start Here via menu Overig).
674
675     Kies 1 'Checklist installatie' op Menu Installatie.
676     Volg de aanwijzingen op het scherm en de stappen in de checklist.
677
678     N.B.
679     De checklist is ook te vinden op de site
680     http://karelzimmer.nl onder Linux."
681
682     zenity --info \
683         --width=550 \
684         --height=150 \
685         --text="$text" \
686         2> /dev/null || true
687
688     sluit_terminal "$@"
689 }
690

```

```
691 # #####
692 # Hoofdlĳn
693 # #####
694
695 # init_script
696 {
697     init_script "$@"
698 }
699
700 # verwerk
701 {
702     verwerk "$@"
703 }
704
705 # afsl_script
706 {
707     afsl_script "$@"
708 }
709
710 # Einde script
```