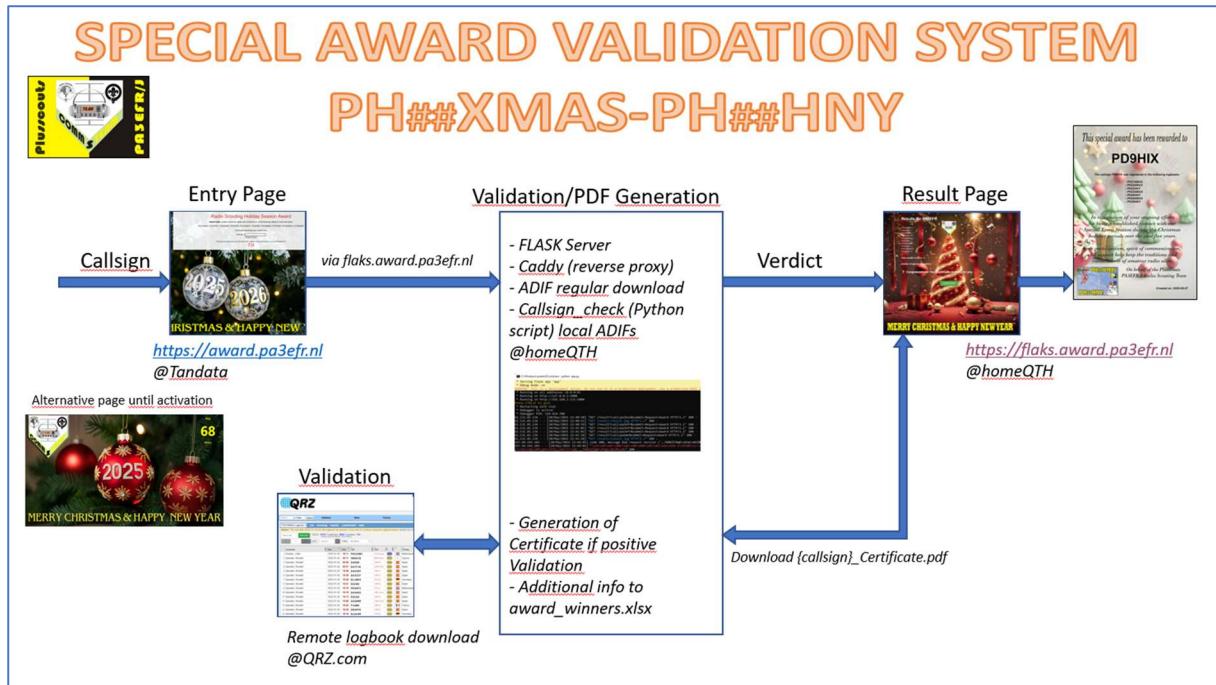


# Special Award Validation System (SAVS)

## voor PH##XMAS en PH##HNY



Rijswijk, september 2025

PA3EFR

## **INLEIDING**

Omdat we in het jubileum jaar van PH##XMAS/PH##HNY iets meer wilden doen dan alleen QSO's maken, bedachten we dat het leuk zou zijn voor trouwe radio operators, die ons in het verleden vaker hebben gewerkt, een award toe te kennen. Maar dan wel via een website en volledig geautomatiseerd.

Met behulp van Python programming skills, AI tools en heel veel geduld is het navolgende systeem tot stand gekomen: het Special Award Validation System (SAVS).

In dit document leg ik de werking vast, geef de verschillende scripts vrij en zijn de verschillende methodieken beschreven dat het SAVS compleet maakt.

In hoofdstuk 1 en 2 worden de verschillende basis componenten stuk voor stuk beschreven. Daarnaast worden de minimale benodigde servers en functionaliteiten beschreven met de bijbehorende applicaties. In hoofdstuk 3 wordt ingegaan op de verschillende uitdagingen en oplossingen die daarvoor gevonden zijn.

In de bijlagen zijn de verschillende batch-files en Python-scripts weergegeven. Deze zijn niet mijn prive eigendom en mogen naar hartenlust her-bruikt worden ter lering ende vermaek.

## HOOFDSTUK 1 BASIS COMPONENTEN

Het SAVS bestaat uit de volgende componenten en hun functie:

### 1. Een Entry Page.

Deze pagina is de eerste entry voor een operator om te checken of hij/zij in aanmerking komt voor een award. De URL naar deze pagina is <https://award.pa3efr.nl> en de pagina draait op een commerciële server als sub domein van pa3efr.nl (in mijn geval Tandata).

Op deze pagina verschijnt een invulveld waar de operator middels zijn callsign de opdracht verstuurd om na te gaan of hij voldoet aan het gestelde aantal QSO's over de afgelopen 5 jaar (inclusief 2025-2026). Deze opdracht wordt gestuurd naar een server die draait op mijn ThinClient, waar ook mijn WIRES-X op draait.



Zolang de datum nog niet bereikt is dat het SAVS actief wordt is op award.pa3efr.nl een alternatieve pagina te zien met een aflopende klok naar 6 december: de datum waarop PH##XMAS als eerste call actief wordt.

### 2. De Flaks server.

```
C:\Windows\system32\cmd.exe - python app.py
* Serving Flask app "app"
* Debug mode: on
* Running on http://127.0.0.1:5000
  (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active
* Debugger PIN: 126-914-700
0.0.0.0:5000 - - [28/May/2025 12:48:50] "GET /result?call=PH##XMAS&submit=Request+Award HTTP/1.1" 200 -
0.0.0.0:5000 - - [28/May/2025 12:48:51] "GET /static/result.jpg HTTP/1.1" 304 -
0.0.0.0:5000 - - [28/May/2025 12:48:59] "GET /result?call=pa3efr&submit=Request+Award HTTP/1.1" 200 -
0.0.0.0:5000 - - [28/May/2025 12:49:01] "GET /result?call=pa3efr&submit=Request+Award HTTP/1.1" 200 -
0.0.0.0:5000 - - [28/May/2025 12:49:09] "GET /result?call=pa3efr&submit=Request+Award HTTP/1.1" 200 -
0.0.0.0:5000 - - [28/May/2025 12:49:18] "GET /result?call=pa3efr&submit=Request+Award HTTP/1.1" 200 -
0.0.0.0:5000 - - [28/May/2025 12:49:19] "GET /static/result.jpg HTTP/1.1" 304 -
167.94.545.103 - - [28/May/2025 12:44:01] code 400, message Bad request version ('1.1.PH##XMAS&submit=Request+Award')
167.94.545.103 - - [28/May/2025 12:44:01] "GET /result?call=PH##XMAS&submit=Request+Award HTTP/1.1" 200 -
167.94.545.103 - - [28/May/2025 12:44:01] "GET /static/result.jpg HTTP/1.1" 304 -
```

om de 15 minuten van het lopende jaar de ADIF binnengehaald van QRZ.zom.

Omdat de Entry Page in een beveiligde omgeving draait ([https](https://)) is ook de Flaks server in een beveiligde omgeving gezet door het activeren van een Reverse Proxy middels het opstarten van een Caddyfile. Deze zorgt voor een valide Let's Encrypt certificate waarmee de mixed environment problems zijn opgelost.

Call	Name	Grid	Country	Score
1 Operator, Novice	2025-05-28 10:11 PH##XMAS	K8A3D6	United States	0.00
2 Operator, Novice	2025-05-28 10:11 KAS3M0	K8A3D6	United States	0.00
3 Operator, Novice	2025-05-28 10:11 KAS3M1	K8A3D6	United States	0.00
4 Operator, Novice	2025-05-28 10:11 KAS3M2	K8A3D6	United States	0.00
5 Operator, Novice	2025-05-28 10:11 KAS3M3	K8A3D6	United States	0.00
6 Operator, Novice	2025-05-28 10:11 KAS3M4	K8A3D6	United States	0.00
7 Operator, Novice	2025-05-28 10:11 KAS3M5	K8A3D6	United States	0.00
8 Operator, Novice	2025-05-28 10:11 KAS3M6	K8A3D6	United States	0.00
9 Operator, Novice	2025-05-28 10:11 KAS3M7	K8A3D6	United States	0.00
10 Operator, Novice	2025-05-28 10:11 KAS3M8	K8A3D6	United States	0.00
11 Operator, Novice	2025-05-28 10:11 KAS3M9	K8A3D6	United States	0.00
12 Operator, Novice	2025-05-28 10:11 KAS3M0	K8A3D6	United States	0.00
13 Operator, Novice	2025-05-28 10:11 KAS3M1	K8A3D6	United States	0.00
14 Operator, Novice	2025-05-28 10:11 KAS3M2	K8A3D6	United States	0.00
15 Operator, Novice	2025-05-28 10:11 KAS3M3	K8A3D6	United States	0.00
16 Operator, Novice	2025-05-28 10:11 KAS3M4	K8A3D6	United States	0.00
17 Operator, Novice	2025-05-28 10:11 KAS3M5	K8A3D6	United States	0.00
18 Operator, Novice	2025-05-28 10:11 KAS3M6	K8A3D6	United States	0.00
19 Operator, Novice	2025-05-28 10:11 KAS3M7	K8A3D6	United States	0.00
20 Operator, Novice	2025-05-28 10:11 KAS3M8	K8A3D6	United States	0.00
21 Operator, Novice	2025-05-28 10:11 KAS3M9	K8A3D6	United States	0.00
22 Operator, Novice	2025-05-28 10:11 KAS3M0	K8A3D6	United States	0.00
23 Operator, Novice	2025-05-28 10:11 KAS3M1	K8A3D6	United States	0.00
24 Operator, Novice	2025-05-28 10:11 KAS3M2	K8A3D6	United States	0.00
25 Operator, Novice	2025-05-28 10:11 KAS3M3	K8A3D6	United States	0.00
26 Operator, Novice	2025-05-28 10:11 KAS3M4	K8A3D6	United States	0.00
27 Operator, Novice	2025-05-28 10:11 KAS3M5	K8A3D6	United States	0.00
28 Operator, Novice	2025-05-28 10:11 KAS3M6	K8A3D6	United States	0.00
29 Operator, Novice	2025-05-28 10:11 KAS3M7	K8A3D6	United States	0.00
30 Operator, Novice	2025-05-28 10:11 KAS3M8	K8A3D6	United States	0.00
31 Operator, Novice	2025-05-28 10:11 KAS3M9	K8A3D6	United States	0.00
32 Operator, Novice	2025-05-28 10:11 KAS3M0	K8A3D6	United States	0.00
33 Operator, Novice	2025-05-28 10:11 KAS3M1	K8A3D6	United States	0.00
34 Operator, Novice	2025-05-28 10:11 KAS3M2	K8A3D6	United States	0.00
35 Operator, Novice	2025-05-28 10:11 KAS3M3	K8A3D6	United States	0.00
36 Operator, Novice	2025-05-28 10:11 KAS3M4	K8A3D6	United States	0.00
37 Operator, Novice	2025-05-28 10:11 KAS3M5	K8A3D6	United States	0.00
38 Operator, Novice	2025-05-28 10:11 KAS3M6	K8A3D6	United States	0.00
39 Operator, Novice	2025-05-28 10:11 KAS3M7	K8A3D6	United States	0.00
40 Operator, Novice	2025-05-28 10:11 KAS3M8	K8A3D6	United States	0.00
41 Operator, Novice	2025-05-28 10:11 KAS3M9	K8A3D6	United States	0.00
42 Operator, Novice	2025-05-28 10:11 KAS3M0	K8A3D6	United States	0.00
43 Operator, Novice	2025-05-28 10:11 KAS3M1	K8A3D6	United States	0.00
44 Operator, Novice	2025-05-28 10:11 KAS3M2	K8A3D6	United States	0.00
45 Operator, Novice	2025-05-28 10:11 KAS3M3	K8A3D6	United States	0.00
46 Operator, Novice	2025-05-28 10:11 KAS3M4	K8A3D6	United States	0.00
47 Operator, Novice	2025-05-28 10:11 KAS3M5	K8A3D6	United States	0.00
48 Operator, Novice	2025-05-28 10:11 KAS3M6	K8A3D6	United States	0.00
49 Operator, Novice	2025-05-28 10:11 KAS3M7	K8A3D6	United States	0.00
50 Operator, Novice	2025-05-28 10:11 KAS3M8	K8A3D6	United States	0.00
51 Operator, Novice	2025-05-28 10:11 KAS3M9	K8A3D6	United States	0.00
52 Operator, Novice	2025-05-28 10:11 KAS3M0	K8A3D6	United States	0.00
53 Operator, Novice	2025-05-28 10:11 KAS3M1	K8A3D6	United States	0.00
54 Operator, Novice	2025-05-28 10:11 KAS3M2	K8A3D6	United States	0.00
55 Operator, Novice	2025-05-28 10:11 KAS3M3	K8A3D6	United States	0.00
56 Operator, Novice	2025-05-28 10:11 KAS3M4	K8A3D6	United States	0.00
57 Operator, Novice	2025-05-28 10:11 KAS3M5	K8A3D6	United States	0.00
58 Operator, Novice	2025-05-28 10:11 KAS3M6	K8A3D6	United States	0.00
59 Operator, Novice	2025-05-28 10:11 KAS3M7	K8A3D6	United States	0.00
60 Operator, Novice	2025-05-28 10:11 KAS3M8	K8A3D6	United States	0.00
61 Operator, Novice	2025-05-28 10:11 KAS3M9	K8A3D6	United States	0.00
62 Operator, Novice	2025-05-28 10:11 KAS3M0	K8A3D6	United States	0.00
63 Operator, Novice	2025-05-28 10:11 KAS3M1	K8A3D6	United States	0.00
64 Operator, Novice	2025-05-28 10:11 KAS3M2	K8A3D6	United States	0.00
65 Operator, Novice	2025-05-28 10:11 KAS3M3	K8A3D6	United States	0.00
66 Operator, Novice	2025-05-28 10:11 KAS3M4	K8A3D6	United States	0.00
67 Operator, Novice	2025-05-28 10:11 KAS3M5	K8A3D6	United States	0.00
68 Operator, Novice	2025-05-28 10:11 KAS3M6	K8A3D6	United States	0.00
69 Operator, Novice	2025-05-28 10:11 KAS3M7	K8A3D6	United States	0.00
70 Operator, Novice	2025-05-28 10:11 KAS3M8	K8A3D6	United States	0.00
71 Operator, Novice	2025-05-28 10:11 KAS3M9	K8A3D6	United States	0.00
72 Operator, Novice	2025-05-28 10:11 KAS3M0	K8A3D6	United States	0.00
73 Operator, Novice	2025-05-28 10:11 KAS3M1	K8A3D6	United States	0.00
74 Operator, Novice	2025-05-28 10:11 KAS3M2	K8A3D6	United States	0.00
75 Operator, Novice	2025-05-28 10:11 KAS3M3	K8A3D6	United States	0.00
76 Operator, Novice	2025-05-28 10:11 KAS3M4	K8A3D6	United States	0.00
77 Operator, Novice	2025-05-28 10:11 KAS3M5	K8A3D6	United States	0.00
78 Operator, Novice	2025-05-28 10:11 KAS3M6	K8A3D6	United States	0.00
79 Operator, Novice	2025-05-28 10:11 KAS3M7	K8A3D6	United States	0.00
80 Operator, Novice	2025-05-28 10:11 KAS3M8	K8A3D6	United States	0.00
81 Operator, Novice	2025-05-28 10:11 KAS3M9	K8A3D6	United States	0.00
82 Operator, Novice	2025-05-28 10:11 KAS3M0	K8A3D6	United States	0.00
83 Operator, Novice	2025-05-28 10:11 KAS3M1	K8A3D6	United States	0.00
84 Operator, Novice	2025-05-28 10:11 KAS3M2	K8A3D6	United States	0.00
85 Operator, Novice	2025-05-28 10:11 KAS3M3	K8A3D6	United States	0.00
86 Operator, Novice	2025-05-28 10:11 KAS3M4	K8A3D6	United States	0.00
87 Operator, Novice	2025-05-28 10:11 KAS3M5	K8A3D6	United States	0.00
88 Operator, Novice	2025-05-28 10:11 KAS3M6	K8A3D6	United States	0.00
89 Operator, Novice	2025-05-28 10:11 KAS3M7	K8A3D6	United States	0.00
90 Operator, Novice	2025-05-28 10:11 KAS3M8	K8A3D6	United States	0.00
91 Operator, Novice	2025-05-28 10:11 KAS3M9	K8A3D6	United States	0.00
92 Operator, Novice	2025-05-28 10:11 KAS3M0	K8A3D6	United States	0.00
93 Operator, Novice	2025-05-28 10:11 KAS3M1	K8A3D6	United States	0.00
94 Operator, Novice	2025-05-28 10:11 KAS3M2	K8A3D6	United States	0.00
95 Operator, Novice	2025-05-28 10:11 KAS3M3	K8A3D6	United States	0.00
96 Operator, Novice	2025-05-28 10:11 KAS3M4	K8A3D6	United States	0.00
97 Operator, Novice	2025-05-28 10:11 KAS3M5	K8A3D6	United States	0.00
98 Operator, Novice	2025-05-28 10:11 KAS3M6	K8A3D6	United States	0.00
99 Operator, Novice	2025-05-28 10:11 KAS3M7	K8A3D6	United States	0.00
100 Operator, Novice	2025-05-28 10:11 KAS3M8	K8A3D6	United States	0.00
101 Operator, Novice	2025-05-28 10:11 KAS3M9	K8A3D6	United States	0.00
102 Operator, Novice	2025-05-28 10:11 KAS3M0	K8A3D6	United States	0.00
103 Operator, Novice	2025-05-28 10:11 KAS3M1	K8A3D6	United States	0.00
104 Operator, Novice	2025-05-28 10:11 KAS3M2	K8A3D6	United States	0.00
105 Operator, Novice	2025-05-28 10:11 KAS3M3	K8A3D6	United States	0.00
106 Operator, Novice	2025-05-28 10:11 KAS3M4	K8A3D6	United States	0.00
107 Operator, Novice	2025-05-28 10:11 KAS3M5	K8A3D6	United States	0.00
108 Operator, Novice	2025-05-28 10:11 KAS3M6	K8A3D6	United States	0.00
109 Operator, Novice	2025-05-28 10:11 KAS3M7	K8A3D6	United States	0.00
110 Operator, Novice	2025-05-28 10:11 KAS3M8	K8A3D6	United States	0.00
111 Operator, Novice	2025-05-28 10:11 KAS3M9	K8A3D6	United States	0.00
112 Operator, Novice	2025-05-28 10:11 KAS3M0	K8A3D6	United States	0.00
113 Operator, Novice	2025-05-28 10:11 KAS3M1	K8A3D6	United States	0.00
114 Operator, Novice	2025-05-28 10:11 KAS3M2	K8A3D6	United States	0.00
115 Operator, Novice	2025-05-28 10:11 KAS3M3	K8A3D6	United States	0.00
116 Operator, Novice	2025-05-28 10:11 KAS3M4	K8A3D6	United States	0.00
117 Operator, Novice	2025-05-28 10:11 KAS3M5	K8A3D6	United States	0.00
118 Operator, Novice	2025-05-28 10:11 KAS3M6	K8A3D6	United States	0.00
119 Operator, Novice	2025-05-28 10:11 KAS3M7	K8A3D6	United States	0.00
120 Operator, Novice	2025-05-28 10:11 KAS3M8	K8A3D6	United States	0.00
121 Operator, Novice	2025-05-28 10:11 KAS3M9	K8A3D6	United States	0.00
122 Operator, Novice	2025-05-28 10:11 KAS3M0	K8A3D6	United States	0.00
123 Operator, Novice	2025-05-28 10:11 KAS3M1	K8A3D6	United States	0.00
124 Operator, Novice	2025-05-28 10:11 KAS3M2	K8A3D6	United States	0.00
125 Operator, Novice	2025-05-28 10:11 KAS3M3	K8A3D6	United States	0.00
126 Operator, Novice	2025-05-28 10:11 KAS3M4	K8A3D6	United States	0.00
127 Operator, Novice	2025-05-28 10:11 KAS3M5	K8A3D6	United States	0.00
128 Operator, Novice	2025-05-28 10:11 KAS3M6	K8A3D6	United States	0.00
129 Operator, Novice	2025-05-28 10:11 KAS3M7	K8A3D6	United States	0.00
130 Operator, Novice	2025-05-28 10:11 KAS3M8	K8A3D6	United States	0.00
131 Operator, Novice	2025-05-28 10:11 KAS3M9	K8A3D6	United States	0.00
132 Operator, Novice	2025-05-28 10:11 KAS3M0	K8A3D6	United States	0.00
133 Operator, Novice	2025-05-28 10:11 KAS3M1	K8A3D6	United States	0.00
134 Operator, Novice	2025-05-28 10:11 KAS3M2	K8A3D6	United States	0.00
135 Operator, Novice	2025-05-28 10:11 KAS3M3	K8A3D6	United States	0.00
136 Operator, Novice	2025-05-28 10:11 KAS3M4	K8A3D6	United States	0.00
137 Operator, Novice	2025-05-28 10:11 KAS3M5	K8A3D6	United States	0.00
138 Operator, Novice	2025-05-28 10:11 KAS3M6	K8A3D6	United States	0.00
139 Operator, Novice	2025-05-28 10:11 KAS3M7	K8A3D6	United States	0

### 3. Result Page

Op de Result Page kan de operator zien in welke jaren er QSO's zijn gemaakt die valide zijn voor het award. Deze pagina draait op de server van Flaks bij mij thuis.



Een knop onderaan de pagina biedt de operator de mogelijkheid om het certificaat te downloaden als PDF met de ingevulde resultaten duidelijk zichtbaar.

### 4. Award/Certificate



Het certificaat laat zien in welke jaren de callsign voorkwam in de gecheckte logboeken. Dit draagt bij aan de betrokkenheid van de operators bij het station en toont eervol aan welke inspanningen er gepleegd zijn om in het jubileum jaar het award te behalen.

Op verzoek kan het certificaat ook na het evenement nog worden opgevraagd omdat een exemplaar (in PDF vorm) ook wordt opgeslagen op de ThinClient, evenals een registratie in de Excel file.

## HOOFDSTUK 2 FUNCTIONALITEITEN

In dit hoofdstuk gaan we in detail in op de verschillende applicaties van de verschillende componenten. In de bijlagen zijn de batch files en scripts opgenomen voor hergebruik.

In dit document worden achtereenvolgende de volgende secties beschreven:

1. Entry Page
2. ThinClient met:
  - a. Flaks Server
  - b. Caddyfile
  - c. Logboek Download
  - d. Callsign Check Script
  - e. Certificate Generator
  - f. Award Winner Registration
3. Result Page

Ik probeer daar waar mogelijk via bookmarks de mogelijkheid te bieden om te schakelen tussen de hoofdstukken/secties en de bijbehorende bijlagen.

### Sectie 1: Entry Page

De Entry Page voor 6 december ziet er anders uit als dat ie er na 6 december uitziet. Dat heeft ermee te maken dat tot 6 december de validatie server niet actief is. Vanaf 6 december gaan PH##XMAS geactiveerd worden en kunnen de operators ook checken of zij in aanmerking komen voor een award.



Deze operationele award pagina (<https://award.pa3efr.nl>) is het eerste gezicht van het award-proces. De operator voert hier zijn callsign in om na te laten gaan of hij in aanmerking komt voor een award.

De tekst geeft aan dat minimaal drie van de aangegeven stations moet zijn gelogd in de afgelopen jaren.

Er wordt op deze pagina (zij het in kleine letters) wel gewaarschuwd voor het feit dat het enige seconden kan duren voordat een resultaat zichtbaar wordt, maar doorgaans is dat geen belemmering om het toch te proberen.

De structuur en inhoudelijke index.html die op de commerciële server van Tandata draait is te vinden in [bijlage A](#).

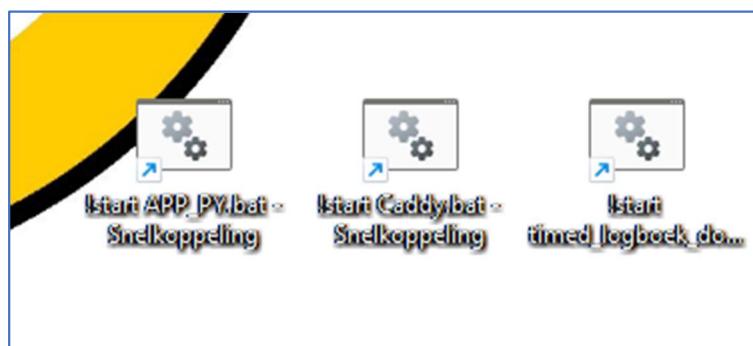
## Sectie 2: ThinClient (TC)

De ThinClient (TC) die voor dit award is ingericht draait permanent in mijn radio shack ten behoeve van WIRES-X. Het is daarom dat gekozen is voor deze low-power-consumptie optie om tijdens de feestdagen de operators hun award te kunnen laten checken.

De TC draait met een harde schijf aan boord en het OS is Windows11Pro.

Op deze TC draait dus het WIRES-X programma (gekoppeld aan externe hardware) en draait de Flaks Server, samen met een Caddyfile aan aanvullende software voor de verwerking van het award-request.

Naam	Status	Gewijzigd op
_pycache_	✓	30-9-2025 10:16
Caddy	✓	30-9-2025 11:21
callsigns	✓	30-9-2025 10:17
certificates	✓	30-9-2025 10:17
nginx	✓	30-9-2025 10:19
static	✓	30-9-2025 10:19
templates	✓	30-9-2025 10:19
!start APP_PY.bat	✓	27-5-2025 15:26
!start timed_logboek_download.bat	✓	28-5-2025 15:48
app.py	✓	20-5-2025 12:12
award_winners.xlsx	✓	29-9-2025 20:37
check_callsign.py	✓	28-5-2025 17:08
start_flask.bat	✓	29-9-2025 15:14
test_port_access.bat	✓	29-9-2025 15:33
timed_logboek_download.py	✓	28-5-2025 16:57



## Sectie 2a: Flaks Server

Een Flaks Server verwerkt binnenkomende verzoeken (requests) van gebruikers en geeft daarop gestructureerde antwoorden (responses). Hij vormt dus de schakel tussen een front-end (Entry Page) en het Callsign Check script die op de TC draait.

Het opstarten van deze Flaks Server dient als eerste te geschieden en staat daarom vooraan in het python script App.py welke als eerste moet worden uitgevoerd via de batchfile !start APP\_PY.bat.

Het script van de APP\_PY.bat en van de !start APP\_PY.bat staan in [bijlage B](#) en [bijlage C](#).

Het script checkt of een ingegeven callsign in aanmerking komt voor een award (middels het script check\_callsign.py, zie sectie 2d) en geeft het resultaat terug in aan de result.html file (zie sectie 2e). Deze html is te vinden in de map templates, die plaatjes gebruikt uit de map static.

Tenslotte wordt in dit script een certificaat aangemaakt met de routine download\_certificate en opgeslagen als PDF met de ingevoerde callsign. Zie hiervoor sectie 2f.

### Sectie 2b: Caddyfile

Een Caddyfile kan als reverse proxy optreden, wat betekent dat het inkomende verzoeken van clients (Entry Page) ontvangt en deze doorstuurt naar een achterliggende server (de Flaks Server).

In een mixed security environment (waar de diensten via HTTPS en andere alleen via HTTP bereikbaar zijn) zorgt de Caddyfile ervoor dat alles naar buiten toe veilig via HTTPS loopt, terwijl Caddy intern flexibel kan doorschakelen naar zowel veilige (HTTPS) als onveilige (HTTP) back-ends. Zo vormt het een beveiligde “tussenlaag” die je omgeving consistent en veilig presenteert aan gebruikers.

De veilige Entry Page kan zo overweg met de onveilige Result Page.

De Caddy File haalt op gezette momenten de Let's Encrypt licentie op om deze te kunnen tonen aan de Entry Page bij een request.

De Caddy gerelateerde informatie staat opgeslagen in de map Caddy en is verwerkt (inhoudelijk) in [bijlage D](#).

De Caddy File wordt als tweede gestart met de routine !start Caddy.bat en is ook te vinden in dezelfde bijlage.

### Sectie 2c: Logboek Download

Voordat een callsign check kan plaatsvinden zijn er een aantal logboeken nodig:

- a. De logboeken van de afgelopen (en afgesloten) jaren.  
Deze worden vooraf gedownload en als ADIF opgeslagen in de map callsigns.
  
- b. De logboeken van de lopende jaren.  
Deze worden op gezette (zelf aan te geven) tijden opgehaald van QRZ.com met de bijbehorende API keys. Het script hiervoor heet timed\_logboek\_download.py en wordt als derde proces opgestart vanaf de desktop met !start timed\_logboek\_download.bat.

Zodra de getimedde logboeken zijn opgehaald loopt er een timer mee die aangeeft hoe lang het duurt voor de volgende getimedde download. Standaard is de interval ingesteld op 15 minuten.

De genoemde scripts zijn uitgewerkt in [bijlage E](#).

### Sectie 2d: Callsign Check Script

Of een callsign voorkomt in de logboeken van de afgelopen en de lopen jaren wordt gecheckt door een script genaamd `check_callsign.py`. Op zich een vrij eenvoudige check in de beschikbare ADIF logboeken en de meest recente gedownloade logboeken van QRZ.com.

Het script is opgenomen in [bijlage F](#).

## Sectie 2e: Certificate Generator

In hetzelfde script check\_callsign.py, maar dan de tweede functie, wordt een certificaat aangemaakt als de callsign voldoet aan de gestelde eisen voor het behalen van het certificaat.

Klik voor het script op de link in de vorige paragraaf.

De gegenereerde certificaten worden opgeslagen in de map certificates en worden weggeschreven als <callsign>\_certificate.pdf.

## **Sectie 2f: Award Winner Registration**

Als een operator voldoet aan het verkrijgen van een award, dan wordt deze bijgeschreven in de file award\_winners.xlsx. Dit is een deel van het script check\_callsign.py. Het is verstandig om voorafgaand aan het evenement deze file op te schonen en te ontdoen van alle test-award\_winners.

### Sectie 3: Result Page

Voor de operator die zijn callsign heeft ingevoerd (Entry Page) krijgt in een nieuw venster zijn resultaten te zien: het aantal keren dat hij voorkomt in een logboek van 1 van de evenementen calls en of hij zijn certificaat kan downloaden.

In deze pagina zit een routine gebakken dat als hij het certificaat wilt downloaden deze PDF wordt opgehaald uit de map certificates en wordt geupload naar zijn/haar systeem.



De result.html staat in de map templates, de images die hiervoor nodig zijn staan in de map static. De inhoud van result.html is opgenomen in [bijlage G](#).

Mochten er onvoldoende contactmomenten geweest zijn dan de vereiste drie, dan wordt dat ook op deze pagina vermeld en kan het certificaat niet gedownload worden.

### HOOFDSTUK 3 UITDAGINGEN EN OPLOSSINGEN

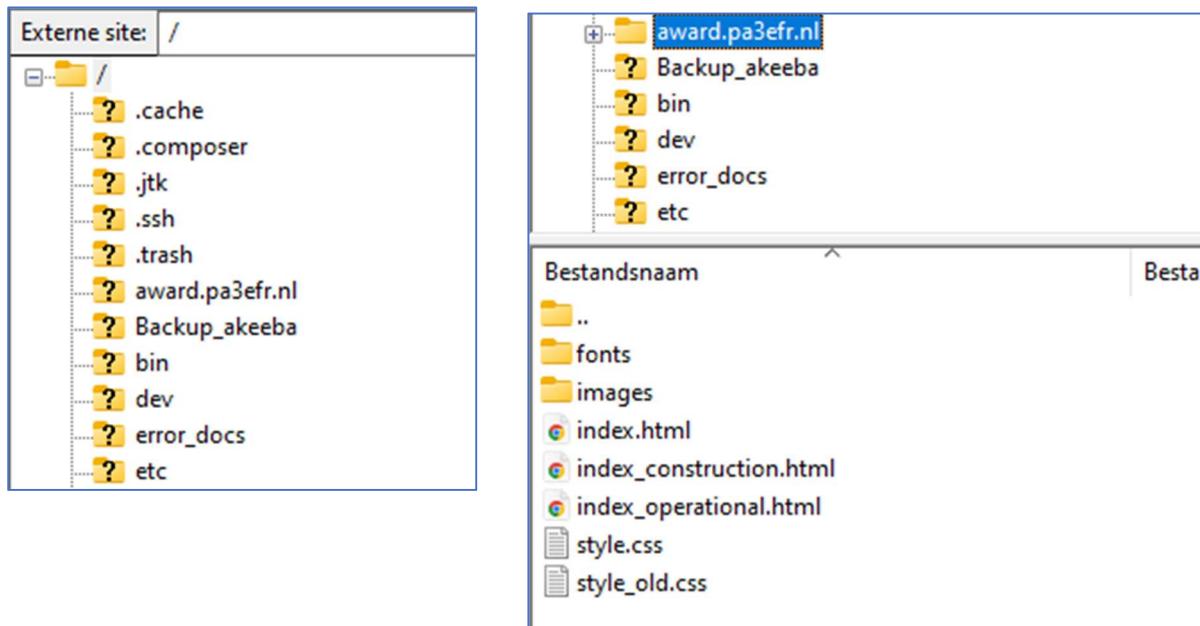
Tijdens de ontwikkel fase van het SAVS heb ik de volgende uitdagingen onderkend:

1. Joomla versus html
2. HTTPS versus HTTP
3. Verwerken data op statische website en statische PDF

Ik zal proberen hier vast te leggen welke oplossingen ik heb gevonden voor de uitdagingen die ik tegenkwam. Grootste ondersteuning in deze waren Artificial Intelligence tools in de vorm van [ChatGPT](#) en [agent.minimax.io](#).

#### 1. Joomla versus html

Mijn website pa3efr.nl is gebouwd in Joomla. En dat terwijl ik mijn award systeem wilde bouwen met html gecodeerde pagina's. Via de Plesk applicatie heb ik eerst een sub domein bij Tandata aangemaakt, genaamd award.pa3efr.nl. Via mijn Filezilla FTP applicatie kon ik daarna gefabriceerde html files daarheen overzetten en was deze extern ook bereikbaar.



#### 2. HTTPS versus HTTP.

Deze uitdaging heeft me de meeste tijd gekost. Met behulp van AI, waarbij ik de wens aangaf, en bij aangereikte oplossingen de Error Codes en respons pagina's terugvoerde, is het me uiteindelijk gelukt om via Caddy de uitdaging opgelost te krijgen.

#### 3. Verwerken data op statische website en statische PDF

Nadat de check\_callsign script het werk heeft gedaan om te zien of een operator in aanmerking komt voor een certificaat, is het vervolgens aan de software om zijn callsign, het

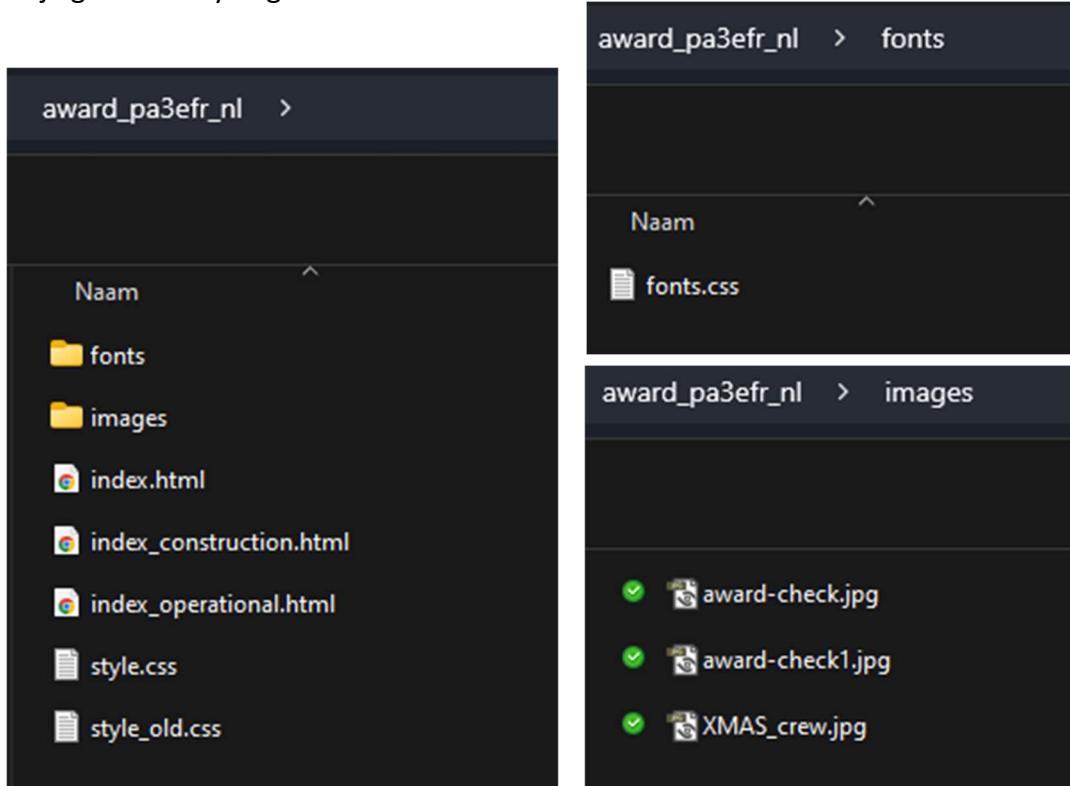
aantal gewerkte stations (met het aantal geregistreerde QSO's) terug te geven op een browserpagina én in een PDF.

Deze scripts waren veelal tijdrovend omdat via trial en error bepaald moest worden of en zo ja waar de data zichtbaar werd. Ook weer dankzij AI is me dit uiteindelijk gelukt. Ook zonder programmeer-kennis is dit dankzij AI prima uit te voeren.

Het is achteraf een leuk project geweest. Het bestaande SAVS is eenvoudig om te bouwen naar een volgend evenement en wellicht leuk voor andere mensen om hierin ook een uitdaging te zien.

Erwin, PA3EFR  
30 september 2025

## Bijlage A – Entry Page



### ➤ Index.html (index\_construction.html)

```
<!doctype html>
<html lang="nl">
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>XMAS countdown</title>
<style>
/* Full-page fixed background from images/XMAS_crew.jpg */
html, body{height:100%;margin:0}
body{
background-image: url('https://award.pa3efr.nl/images/XMAS_crew.jpg');
background-size: cover;
background-position: center;
background-repeat: no-repeat;
background-attachment: fixed;
-webkit-font-smoothing:antialiased;
-moz-osx-font-smoothing:grayscale;
user-select:none;
}

/* Position countdown top right but moved left between ornaments */
#container{
position:fixed;
top:20px;
right:100px; /* push it left from the edge */
text-align:center;
color: yellow;
font-family: system-ui, -apple-system, "Segoe UI", Roboto, "Helvetica
Neue", Arial;
text-shadow: 0 2px 12px rgba(0,0,0,0.7);
}
```

```

#label-top, #label-bottom {
    font-size: 1.5rem;
    font-weight:600;
    display:block;
}

#countdown{
    font-weight:700;
    font-size: calc(48px + 4vw);
    line-height:1;
    margin:0.25em 0;
}

/* Hide other accidental elements */
a, img, p, h1, h2, h3, h4, h5, h6 { display: none !important; }

</style>
</head>
<body>
<div id="container" aria-label="Dagen tot 6 december 2025">
    <span id="label-top">Only</span>
    <div id="countdown"></div>
    <span id="label-bottom">days 2 go</span>
</div>

<script>
    (function(){
        // Target date: 6 December 2025 at 00:00:00 local time
        const target = new Date(2025, 11, 6, 0, 0, 0); // month is 0-based: 11 = December
        const el = document.getElementById('countdown');

        function update(){
            const now = new Date();
            const msPerDay = 24*60*60*1000;
            const diff = target - now;
            let days = Math.ceil(diff / msPerDay);
            if (days < 0) days = 0;
            el.textContent = String(days);
        }

        // initial render and then update every 30 seconds
        update();
        setInterval(update, 30_000);
    })();
</script>
</body>
</html>

```

## ➤ Index\_operational.html

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta charset="utf-8">
    <title>Download Award</title>
    <link rel="icon" href="../favicon.ico">
    <link rel="stylesheet" href="style.css">
    <link rel="stylesheet" href="../fonts/fonts.css">

    <!-- Toegevoegde achtergrondstijl (enige wijziging) -->
    <style>
        html, body {
            height: 100%;
            margin: 0;

```

```

        padding: 0;
    }

    body {
        background-image: url('images/award-check.jpg'); /* Zorg dat dit pad
klopt */
        background-size: cover;
        background-position: center;
        background-repeat: no-repeat;
        min-height: 100vh;
    }

```

</style>

</head>

<body>

<div id="content">

<p>
 <span class="h">Radio Scouting Holiday Season Award</span><br>
 </p>
 <p><b>Award rules:</b> contact should be made with at least three of the
following stations in the past years:</p>
 <p>
 PH21XMAS, PH22HNY, PH22XMAS, PH23HNY, PH23XMAS, PH24HNY, PH24XMAS,
PH25HNY, PH25XMAS or PH26HNY.
 <br><br>
 Check and download your award here:
 </p>
 <p>
 <form action="https://Flaks.award.pa3efr.nl:8443/result" method="GET"
target="\_blank">
 <label for="call">Callsign:</label>
 <input type="text" id="call" name="call" pattern="[A-Za-z0-9/]+"
title="Only uppercase letters, numbers, and / are allowed." required>
 <br>
 <input type="submit" name="submit" value="Request Award">
 </form>
 </p>
 <p>
 <small><i>Checking logbooks takes up to 30 seconds for validation and
is executed on a private RaspberryPi.</i></small>
 <br>
 <span class="big">73!</span>
 </p>
 </div>
</body>
</html>

## ➤ Styles.css

```

body {
    background: url(images/award-check.jpg) repeat;
    text-align: center;
    margin: 5%;
    background-size: cover;
}

.message {
    padding: 3px;
    font-weight: 700;
}

.big {
    font-size: 55px;
    margin: 0
}

#content{

```

```
border: 1px solid #fcfcfc;
/*background-color: white;*/
/*opacity:0.9;*/
/* use background instead of opacity, so images will show without opacity */
background: rgba(255, 255, 255, 0.90);
border-radius: 5px;
justify-content: center;
/*https://getcssscan.com/css-box-shadow-examples*/
box-shadow: rgba(0, 0, 0, 0.12) 0px 1px 3px, rgba(0, 0, 0, 0.24) 0px 1px 2px;
overflow:hidden; /*iphone*/
margin-left:28vw; max-width:66vw;
padding-left:15px;
}

.thumbnail {
max-width:350px;
height:auto;
}

.thumbnail-test {
border: 1px solid black;
}

/*SSTV iframe*/
#sstv, #qso, #qrz, #awards{
max-width:1280px;
height:520px;
width:100%;
overflow:hidden !important;
}

#qrz{
height:552px;
}
.internetnl{
max-height:25px;
width:auto;
}

iframe { border: none; }

a {
text-decoration:none;
}

big {
font-size:55px;
}

.xiegu {
float:left;
padding:25px;
}

.left {
text-align: left;
}

/*SSTV iframe*/
@media only screen and (max-width: 800px) {
#content{
margin-left:0; max-width:90vw;
}
.sstv, #sstv {
max-width:80vw;
}
}
```

```

.xiegu {
    display: none;
}
.left {
    text-align: center;
}
body {
    background-size: 100%; /*remove after easter*/
}
}

.pc1kshack {
width:90%;
max-width:700px;
height:auto;
border: 1px solid #cccccc;
border-radius: 5px;
}

.qrzaward {
width:130px;
height:auto;
border-radius: 5px;
background-color: white;
margin:3px;
border: 1px solid #cccccc;
}

.korenaer {
max-width:90%;
max-height: 500px;
width: auto;
}

```

## ➤ Fonts.css

```

@font-face{font-family:'alata';src:url('alata-regular-ham.woff2')
format('woff2');}

@font-face{font-family:'dejavusans';src:url('NotoSans-Regular.woff2')
format('woff2');}

@font-face{font-family:'NotoSans';src:url('NotoSans-Regular.woff2')
format('woff2');}

@font-face{font-family:'NotoSansBold';src:url('NotoSans-Bold.woff2')
format('woff2');}

@font-face{font-family:'NotoSansMono';src:url('NotoSansMono-
VariableFont_wdth,wght.woff2') format('woff2');}

/* Flag font `TwemojiCountryFlags.woff2` generated by using `npm install country-
flag-emoji-polyfill` copied woff2 file into this folder. */

@font-face{font-family:'twemoji';src:url('TwemojiCountryFlags.woff2')
format('woff2');}

body {
font-family:NotoSans,sans-serif;
}

.bold, th{
font-family:NotoSansBold,sans-serif;;
}
```

```
}
```

```
.h, .big {
    font-family:alata,sans-serif;
    font-size:39px;
    color:#cb2821;
}
```

```
.red {
    color:#cb2821;
}
```

```
.flag {
    font-family:twemoji;
}
```

```
/* Media query to exclude Apple devices (Safari on iOS and macOS) */
@supports (-webkit-touch-callout: none) {
    .flag {
        font-family: sans-serif; /* Fallback font for Apple devices, iOS rendering bug */
    }
}
```

[Return to text.](#)

## Bijlage B, APP\_PY.bat

```
from flask import Flask, request, render_template, send_from_directory
from check_callsign import check_callsign
import os

app = Flask(__name__)

@app.route('/')
def index():
    return '☒ Flask backend draait. Gebruik <code>/result?call=PA3EFR</code> om een award op te vragen.'

@app.route('/result', methods=['GET'])
def result():
    call = request.args.get('call', '').strip().upper()
    if not call:
        return "☒ Geen geldige callsign opgegeven."

    data = check_callsign(call)
    return render_template('result.html', data=data)

@app.route('/certificate/<callsign>')
def download_certificate(callsign):
    cert_dir = 'certificates'
    filename = f"{callsign.upper()}_certificate.pdf"
    filepath = os.path.join(cert_dir, filename)
    if os.path.exists(filepath):
        return send_from_directory(cert_dir, filename, as_attachment=True)
    return f"☒ Certificate not found for {callsign}", 404

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=5000, debug=True)
```

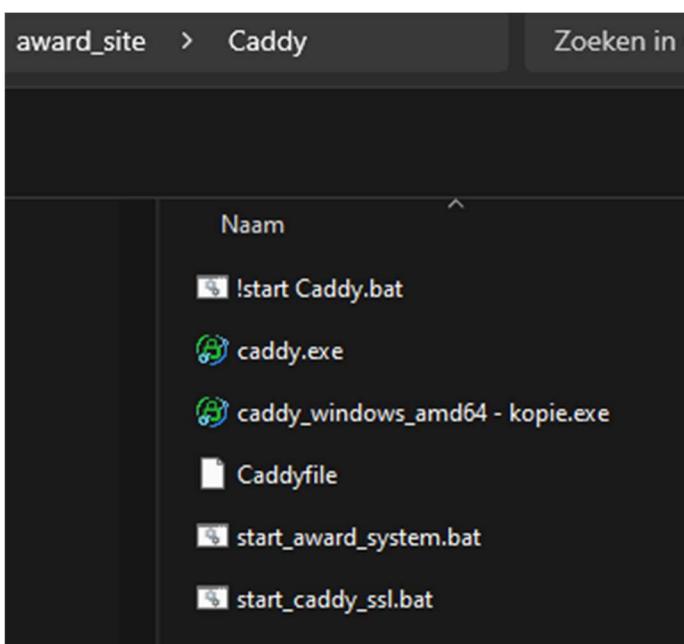
[Return to text](#)

## Bijlage C: !start APP\_PY.bat

```
@echo off
START CMD /T:e0 /K "CD D:\award_site & python app.py"
```

[Return to text](#)

## Bijlage D: Caddy Files



De files die hieronder niet beschreven zijn werden aangemaakt in de ontwikkel fase van het project. Deze worden ook niet verder uitgelegd.

### ➤ !start Caddy.bat

```
@echo off
title Award Systeem - Caddy op Poort 8443
echo =====
echo      Award Systeem - HTTPS Poort 8443
echo =====
echo.
echo Controleren of Flask server draait...
netstat -an | findstr ":5000" > nul
if errorlevel 1 (
    echo [FOUT] Flask server draait niet op poort 5000!
    echo.
    echo Start eerst Flask in een ander venster:
    echo     python app.py
    echo.
    pause
    exit /b 1
)
echo [OK] Flask server gevonden op poort 5000
echo.
echo Starten van Caddy op poort 8443...
echo.
echo De URL wordt: https://flask.award.pa3efr.nl:8443
echo.
caddy run --config Caddyfile
echo.
echo Caddy gestopt.
pause
```

### ➤ Caddyfile

```
flask.award.pa3efr.nl:8443 {
    reverse_proxy localhost:5000
}
```

[Return to text](#)

## Bijlage E: ophalen logboeken lopend jaar

### ➤ !start timed\_logboek\_download.bat

```
@echo off
START CMD /T:e0 /K "CD D:\award_site & python timed_logboek_download.py"
```

### ➤ timed\_logboek\_download.py

```
import os
import time
import requests
import html
import urllib.parse
from datetime import datetime, timedelta

def update_remote_logbooks():
    remote_logbooks = {
        "PH25XMAS": "4DB9-....-579F",
        "PH26HNY": "C822-....-0906"
    }

    adif_directory = r"D:\award_site\callsigns"

    for logbook_name, api_key in remote_logbooks.items():
        params = {
            "KEY": api_key,
            "ACTION": "FETCH",
            "ADIF": 1
        }

        try:
            response = requests.get("https://logbook.qrz.com/api", params=params)
            if response.status_code != 200:
                print(f"☒ Fout bij ophalen {logbook_name}: {response.status_code}")
                continue

            decoded = html.unescape(response.text)
            parsed = urllib.parse.parse_qs(decoded)
            adif_data = parsed.get("ADIF", [""])[0]

            filename = os.path.join(adif_directory, f"{logbook_name}.adi")
            with open(filename, "w", encoding="utf-8") as f:
                f.write(adif_data)
            print(f"☑ {logbook_name} opgeslagen als lokaal .adi bestand")

        except Exception as e:
            print(f"☒ Fout bij {logbook_name}: {e}")

if __name__ == "__main__":
    try:
        interval_minutes = input("Hoe vaak moet de update uitgevoerd worden (in minuten)? [standaard: 15]: ")
        interval_minutes = int(interval_minutes) if interval_minutes.strip() else 15
    except ValueError:
        print("⚠ Ongeldige invoer, standaard wordt 15 minuten gebruikt.")
        interval_minutes = 15

    interval = timedelta(minutes=interval_minutes)
```

```
next_run = datetime.now()

print(f"\n❷ Script gestart. Updates elke {interval_minutes} minuten. Druk op
CTRL+C om te stoppen.")

try:
    while True:
        if datetime.now() >= next_run:
            print(f"\n❸ {datetime.now().strftime('%H:%M:%S')} - Update
gestart")
            update_remote_logbooks()
            next_run = datetime.now() + interval

        # Toon afteller
        remaining = (next_run - datetime.now()).total_seconds()
        minutes = int(remaining) // 60
        seconds = int(remaining) % 60
        print(f"❹ Nog {minutes:02d}:{seconds:02d} tot volgende update",
end="\r")
        time.sleep(1)

    except KeyboardInterrupt:
        print("\n❺ Script gestopt door gebruiker.")
```

[Return to text](#)

## Bijlage F: controle van de callsign

### ➤ check\_callsign.py

```
import os
import re
import html
import requests
import urllib.parse
from openpyxl import Workbook, load_workbook
from datetime import datetime
from reportlab.pdfgen import canvas
from reportlab.lib.pagesizes import A4
from reportlab.lib.utils import ImageReader
from PIL import Image

# === CONFIG ===
adif_directory = r"D:\award_site\callsigns"

allowed_local_logbooks = [
    "PH21XMAS", "PH22HNY", "PH22XMAS", "PH23HNY", "PH23XMAS",
    "PH24HNY", "PH24XMAS", "PH25HNY", "PH25XMAS", "PH26HNY"
]
call_regex = re.compile(r"<call:>\d+([A-Z0-9/]+)", re.IGNORECASE)

def check_callsign(input_call):
    input_call = input_call.strip().upper()
    results = {}

    # Lokale ADIF-bestanden
    adif_files = [f for f in os.listdir(adif_directory) if
f.lower().endswith(".adi")]
    for adif_file in adif_files:
        logbook_name = adif_file.split('.')[0].upper()
        if logbook_name not in allowed_local_logbooks:
            continue

        file_path = os.path.join(adif_directory, adif_file)

        try:
            with open(file_path, 'r', encoding='utf-8', errors='ignore') as f:
                contents = f.read()
                calls_found = call_regex.findall(contents)
                match_count = [c.upper() for c in calls_found].count(input_call)
                results[logbook_name] = match_count

        except Exception as e:
            results[logbook_name] = f"FOUT: {e}"

    total_qsos = sum(count for count in results.values() if isinstance(count,
int))
    logbooks_with_qsos = sum(1 for count in results.values() if isinstance(count,
int) and count > 0)
    qualified = logbooks_with_qsos >= 3

    # Excel-output
    if qualified:
        excel_path = "award_winners.xlsx"
        if os.path.exists(excel_path):
            wb = load_workbook(excel_path)
            ws = wb.active
        else:
            wb = Workbook()
```

```

        ws = wb.active
        ws.append(["Callsign", "Logboeken", "Datum", "Aantal logboeken",
        "Aantal QSO's"])

        logboeken_met_match = [log for log, count in results.items() if
        isinstance(count, int) and count > 0]
        logboeken_str = ", ".join(logboeken_met_match)
        vandaag = datetime.today().strftime('%Y-%m-%d')

        ws.append([input_call, logboeken_str, vandaag, logbooks_with_qsos,
        total_qsos])
        wb.save(excel_path)

        # Maak het certificaat
        certificate_dir = "certificates"
        os.makedirs(certificate_dir, exist_ok=True)
        certificate_filename = os.path.join(certificate_dir,
        f"{input_call}_certificate.pdf")
        create_certificate(input_call, logboeken_met_match, certificate_filename)

    return {
        "callsign": input_call,
        "results": results,
        "total_qsos": total_qsos,
        "logbooks_with_qsos": logbooks_with_qsos,
        "qualified": qualified
    }

def create_certificate(callsign, matched_logbooks,
output_path="callsign_certificate.pdf"):
    print(callsign)
    print(matched_logbooks)
    background_path = os.path.join("static", "certificate.jpg")
    if not os.path.exists(background_path):
        print("X Achtergrondbestand ontbreekt:", background_path)
        return

    background = Image.open(background_path)
    width, height = A4

    c = canvas.Canvas(output_path, pagesize=A4)
    c.drawImage(ImageReader(background), 0, 0, width=width, height=height)

    font_name = "Helvetica-Bold"
    font_size = 64
    text = f"{callsign}"
    x = width / 2
    y = height - 180

    c.setFont(font_name, font_size)
    c.setFillColorRGB(1, 1, 1)
    for dx in [-1, 0, 1]:
        for dy in [-1, 0, 1]:
            if dx != 0 or dy != 0:
                c.drawCentredString(x + dx, y + dy, text)

    c.setFillColorRGB(0, 0, 0)
    c.drawCentredString(x, y, text)

    c.setFont("Helvetica-BoldOblique", 14)
    y = height - 240
    c.drawString(100, y, f"The callsign {callsign} was registered in the
following logbooks:")

    y -= 36
    for logbook in matched_logbooks:
        c.drawString(250, y, f"- {logbook}")
        y -= 16

```

```
c.drawString(400, 50, f"Created on: {datetime.today().strftime('%Y-%m-%d')}")  
c.save()
```

[Return to text](#)

## Bijlage G: resultaat van de controle

### ➤ results.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Award Results for {{ data.callsign }}</title>
    <style>
        html, body {
            height: 100%;
            margin: 0;
            padding: 0;
        }

        body {
            background-image: url('{{ url_for("static", filename="result.jpg") }}');
            background-size: cover;
            background-position: center;
            background-repeat: no-repeat;
            min-height: 100vh;
            color: white;
            font-family: sans-serif;
        }

        h1, h2, p, li {
            text-shadow: 1px 1px 3px rgba(0, 0, 0, 0.7);
        }

        ul {
            list-style-type: none;
            padding-left: 0;
        }

        li {
            margin-bottom: 4px;
        }

        body > * {
            max-width: 800px;
            margin: 0 auto;
            padding: 1em;
        }

        .certificate-button {
            display: inline-block;
            background-color: #4CAF50;
            color: white;
            padding: 12px 24px;
            text-align: center;
            font-size: 16px;
            border: none;
            border-radius: 8px;
            text-decoration: none;
            margin-top: 20px;
            box-shadow: 1px 1px 4px rgba(0,0,0,0.3);
        }
    </style>
</head>
<body>
    <h1>Results for {{ data.callsign }}</h1>
    <ul>
```

```

{%
    for logbook, count in data.results.items() %
        <li>{{ logbook }}: {{ count }}</li>
    {% endfor %}
</ul>

<p><strong>Total number of QSO's:</strong> {{ data.total_qsos }}</p>
<p><strong>Number of logbook registrations:</strong> {{ data.logbooks_with_qsos }}</p>

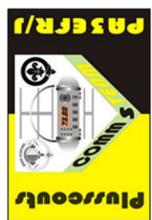
{%
    if data.qualified %
        <h2>❷ Congratulations! Your certificate is ready for you.</h2>
        <div style="text-align: center;">
            <a class="certificate-button" href="{{ url_for('download_certificate', callsign=data.callsign) }}">
                CERTIFICATE
            </a>
        </div>
    {% else %
        <h2>❸ Unfortunately, inadequate number of QSO's for the award.</h2>
        <p>At least 1 registration is required.</p>
    {% endif %}
</body>
</html>

```

[Return to text](#)

# SPECIAL AWARD VALIDATION SYSTEM

## PH##XMAS-PH##HNY



Entry Page



<https://award.pa3efr.nl>

@Tandata



MERRY CHRISTMAS & HAPPY NEW YEAR



Remote logbook download  
@QRZ.com

Validation/PDF Generation

- FLASK Server
- Caddy (reverse proxy)
- ADIF regular download
- Callsign\_check (Python script) local ADIFs

@homeQTH

Validation



- Generation of Certificate if positive Validation
- Additional info to award\_winners.xlsx

Result Page



<https://flaks.award.pa3efr.nl>

@homeQTH

Verdict

