

WinSW

WinSW

1)

Windows Service WinSW Task Scheduler.

2)

D:\SMSSEnder\iMaster.ERP.SmsSender.Console.exe

WinSW

D:\SmsService\SmsRunnerService.exe

D:\SmsService\SmsRunnerService.xml

D:\SmsService\run-loop.ps1

D:\SmsService\runner.log

WinSW

D:\SmsService\logs

3)

- exe WinSW xml .
 - :
 - :
SmsRunnerService.exe
 - :
SmsRunnerService.xml
 - exe xml .
 - .
 - \$exePath run-loop.ps1 .
 - \$processName .
-

4)

1:

:
D:\SMSSEnder\
D:\SmsService\
D:\SmsService\logs\
2:

- :

D:\SMSSender\iMaster.ERP.SmsSender.Console.exe

- WinSW :
D:\SmsService\SmsRunnerService.exe
- run-loop.ps1 :
D:\SmsService\run-loop.ps1
- SmsRunnerService.xml :
D:\SmsService\SmsRunnerService.xml

3:

CMD PowerShell **Run as Administrator**

```
cd /d D:\SmsService
SmsRunnerService.exe install
SmsRunnerService.exe start
```

4: Recovery

:

Recovery

```
sc failure SmsRunnerService reset= 86400 actions= restart/5000/restart/10000/restart/60000
sc config SmsRunnerService start= auto
```

5)

run-loop.ps1

PowerShell

```
$exePath = "D:\SMSSender\iMaster.ERP.SmsSender.Console.exe"
$processName = "iMaster.ERP.SmsSender.Console"
$intervalSeconds = 60
$logFile = "D:\SmsService\runner.log"

while ($true) {
    $timestamp = Get-Date -Format "yyyy-MM-dd HH:mm:ss"

    try {
        $alreadyRunning = Get-Process -Name $processName -ErrorAction SilentlyContinue

        if (-not $alreadyRunning) {
            Add-Content $logFile "[${timestamp}] Starting $exePath"

            $process = Start-Process -FilePath $exePath -PassThru
            $process.WaitForExit()

            $exitCode = $process.ExitCode
            Add-Content $logFile "[${timestamp}] Finished with exit code $exitCode"
        }
        else {
            Add-Content $logFile "[${timestamp}] Process already running. Skipped."
        }
    }
    catch {
        Add-Content $logFile "[${timestamp}] ERROR: ${($_.Exception.Message)}"
    }

    Start-Sleep -Seconds $intervalSeconds
}
```

SmsRunnerService.xml

XML

```
<service>
  <id>SmsRunnerService</id>
  <name>SMS Runner Service</name>
  <description>Runs iMaster.ERP.SmsSender.Console.exe every 1 minute</description>

  <executable>C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe</executable>
  <arguments>-NoProfile -ExecutionPolicy Bypass -File "D:\SmsService\run-loop.ps1"</arguments>

  <logpath>D:\SmsService\logs</logpath>
  <log mode="roll-by-size">
    <sizeThreshold>10240</sizeThreshold>
    <keepFiles>8</keepFiles>
  </log>

  <startmode>Automatic</startmode>

  <onfailure action="restart" delay="10 sec"/>
  <onfailure action="restart" delay="30 sec"/>
  <onfailure action="restart" delay="1 min"/>
</service>
```

6)

1: Services

```
:
services.msc
    Running:
SMS Runner Service
```

2:

```
sc query SmsRunnerService
    RUNNING.
```

3:

```
:
D:\SmsService\runner.log
```

4: WinSW

```
:
D:\SmsService\logs
```

7)

```
run-loop.ps1 :
$intervalSeconds = 60
```

- 1:
\$intervalSeconds = 60
 - 2:
\$intervalSeconds = 120
 - 5:
\$intervalSeconds = 300
- :

D:\SmsService\SmsRunnerService.exe restart

8)

Start / Stop / Restart / Uninstall

```
D:\SmsService\SmsRunnerService.exe start
D:\SmsService\SmsRunnerService.exe stop
D:\SmsService\SmsRunnerService.exe restart
D:\SmsService\SmsRunnerService.exe uninstall
```

9)

- .
 - :
 - ◦ \$exePath
 - ◦ \$processName
 - runner.log D:\SmsService .
 - D:\SmsService\logs .
 - :
 - ◦
 - ◦
 - ◦
 - ◦
-

10)

:

- WinSW
- PowerShell
- runner.log
-