```python
"""
This file is a heavily modified version of https://github.com/x4nth055/pythoncode-tutorials/tree/master/ethical-hacking/keylogger

The original code is licensed under the MIT license (https://github.com/x4nth055/pythoncode-tutorials/blob/master/LICENSE).
This means that we can use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the original code, as long as we give credit to the original author and include the license in our final product.
"""

import keyboard
import asyncio
from types import MethodType

class splitWatcher:
  def __init__(self,  splitFunction: MethodType.__func__, endFunction: MethodType.__func__, window, split_key = "=", exit_key = "~"):
    print("Split Key Watcher | Setting Up")
    self.splitFunction = splitFunction
    self.endFunction = endFunction

    # TODO: make this configurable with pysimple GUI
    self.split_key = split_key
    self.exit_key = exit_key

    self.window = window

    self.setup = False
    self.exit_program = False
    print("Split Key Watcher | Setup Complete")

  def callback(self, event):
    name = event.name
    if self.exit_program:
      self.window.write_event_value('Exit', 0)
      self.endFunction()
      self.exit_program = True
      del self
    elif name == self.split_key:
      # TODO: Integrate this with pysimple GUI to save the current time to a csv file
      self.window.write_event_value('-SPLIT-TIMER-', 0)
      print(self.split_key + " | was pressed and you have split")
      self.splitFunction()
    elif name == self.exit_key:
      print(self.exit_key + " | was pressed and you have exited the program")
      self.exit_program = True

  async def endProgram(self):
    while not self.exit_program:
      await asyncio.sleep(0.1)
    return 1

  async def start(self):
    print("Split Key Watcher | Starting")
    # start the keylogger
    keyboard.on_release(callback=self.callback)
    # block the current thread, wait until CTRL+C is pressed
    await self.endProgram()
    print("END")
```