

# Project Integration Technical Design

Jochem Arends  
495637  
Group 1

Academic Year: 2023-2024

```

procedure FALL_DETECTION(name)
   $\vec{v} \leftarrow \text{READ\_ACCELEROMETER}$ 
   $\text{angular\_v} \leftarrow \text{READ\_GYROSCOPE}$ 

  if  $|\vec{v}| \geq \text{threshold}$  and IN_LYING_POSE( $\text{angular\_v}$ ) then
     $\text{location} \leftarrow \text{GET\_LOCATION}$ 
     $\text{time} \leftarrow \text{GET\_TIME}$ 
     $\text{event\_type} \leftarrow \text{fall}$ 

    while no wireless connection do
      TRY_TO_CONNECT
      SLEEP ▷ non-blocking delay
    end while

     $\text{msg} \leftarrow \{\text{location}, \text{time}, \text{event\_type}\}$ 
    NOTIFY_THIRD_PARTY( $\text{msg}$ )
    INSERT_INTO_DATABASE( $\text{msg}$ )
  end if
end procedure

```

```

procedure APP_MAIN
  if not logged in then
    PROMPT_FOR_CREDENTIALS
  end if

   $\vec{data} \leftarrow \text{FETCH\_DATA}$ 

  repeat
    if database updated then
       $\vec{data} \leftarrow \text{FETCH\_DATA}$ 
    end if

    for all  $row \in \vec{data}$  do
      RENDER( $row$ )
    end for
  until forever
end procedure

```

```

procedure STARTUP
  INIT_COMPONENTS

  if not logged in then
    username  $\leftarrow$  PROMPT_FOR_USERNAME
    password  $\leftarrow$  PROMPT_FOR_PASSWORD

    if has account then
      LOGIN(username, password)
    else
      contact person  $\leftarrow$  PROMPT_FOR_PHONE_NUMBER

      WRITE_NVM(username)            $\triangleright$  to non-volatile memory
      WRITE_NVM(password)
      WRITE_NVM(contact person)

      ADD_TO_DATABASE(username, password, contact person)
    end if
  end if
end procedure

```

```

procedure OBSERVE_HEART_RATE
   $bpm \leftarrow \text{READ\_BPM}$ 

  if  $bpm < \text{threshold}_{min}$  or  $bpm > \text{threshold}_{max}$  then
     $location \leftarrow \text{GET\_LOCATION}$ 
     $time \leftarrow \text{GET\_TIME}$ 
     $event\_type \leftarrow \text{heartfailure}$ 

    while no wireless connection do
      TRY_TO_CONNECT
      SLEEP ▷ non-blocking delay
    end while

     $msg \leftarrow \{location, time, event\_type\}$ 
    NOTIFY_THIRD_PARTY( $msg$ )
    INSERT_INTO_DATABASE( $msg$ )
  end if
end procedure

```

```
procedure MAIN_LOOP
  FALL_DETECTION
  OBVERSE_HEART_RATE

  time  $\leftarrow$  GET_CURRENT_TIME
  RENDER(time) MAIN_LOOP
end procedure
```

- ▷ Also acts on events.
- ▷ Also acts on events.