

# Metrics

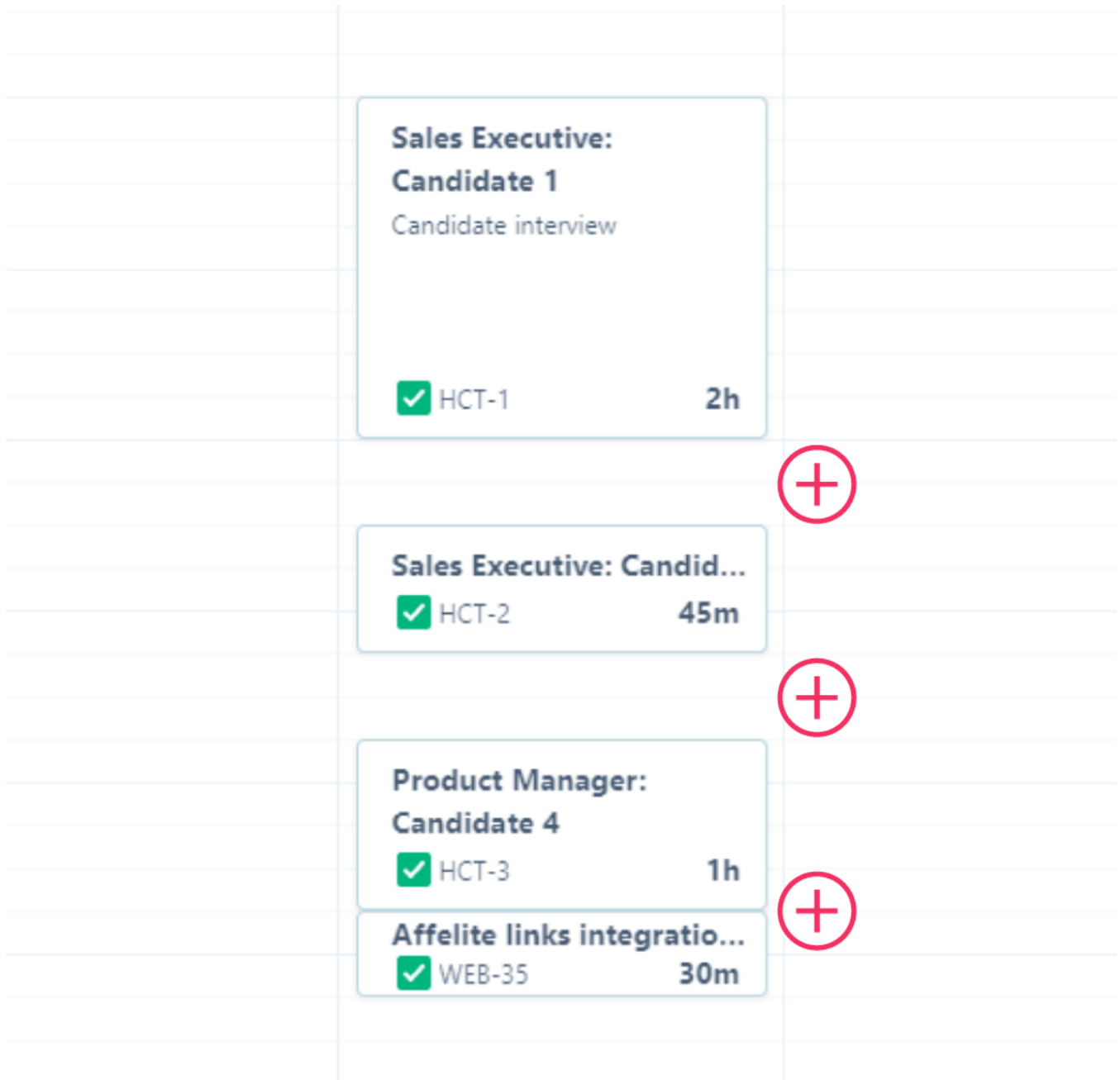
Metrics are the **core** of a rule and can be seen as a framework, or a built-in check that defines what you can configure inside of a rule.

## Working day



The **working day** metric lets you define the **minimum** or **maximum length** of a **regular working day**.

It accumulates the **total** of all time **logged on any given day**.



## Example configuration

One of the [Preset rules](#) uses the working day metric to check whether users logged **more than 10 hours** on any given working day.

## Create new rule - Working day

Name \*

Maximum daily working hours

Description

Check whether the users have worked more than 10 hours on a given day.

Operator \*

The operator compares the value you define in the next step with the metric you selected. 

<=

Working time \*

Configure the number of **hours/minutes** that define a **working day**. 

Hours \*

10


Minutes \*

0

Back

Submit

### Translation


The total of work logged on any given day **must be lower than or equal to (<=)** 10 hours and 0 minutes for the check to pass .

## Working week



Using the **working week** metric lets you check whether your users logged the correct amount of time in a week.

You can either set a maximum or minimum for the week.

 You can e.g. create **two separate rules** using this metric to define a **specific range of work to be logged** in a week (at least x hours but at most y hours per week)

## Example configuration

## Edit - Maximum weekly working hours for students

Name \*

Maximum weekly working hours for students

Description

Check that working students have not worked more than 20 hours per week.

Operator \*

Select the comparison operator to be used. 

<=



Working time \*

Configure the number of **hours/minutes** that define a **working week**.

Hours \*

20

Minutes \*

0

Cancel

Submit

## Working hours



The working hours metric lets you define a **specific timeframe** in which the users only are **allowed to log time**.

This way you can restrict e.g. logging time at night.

## Example configuration

## Create rule - Working hours

Name \*

Only log during working hours

Description

Users are only allowed to log work between 7:30 and 19:30.

Working hours



[Back](#)

[Submit](#)

## Rest breaks

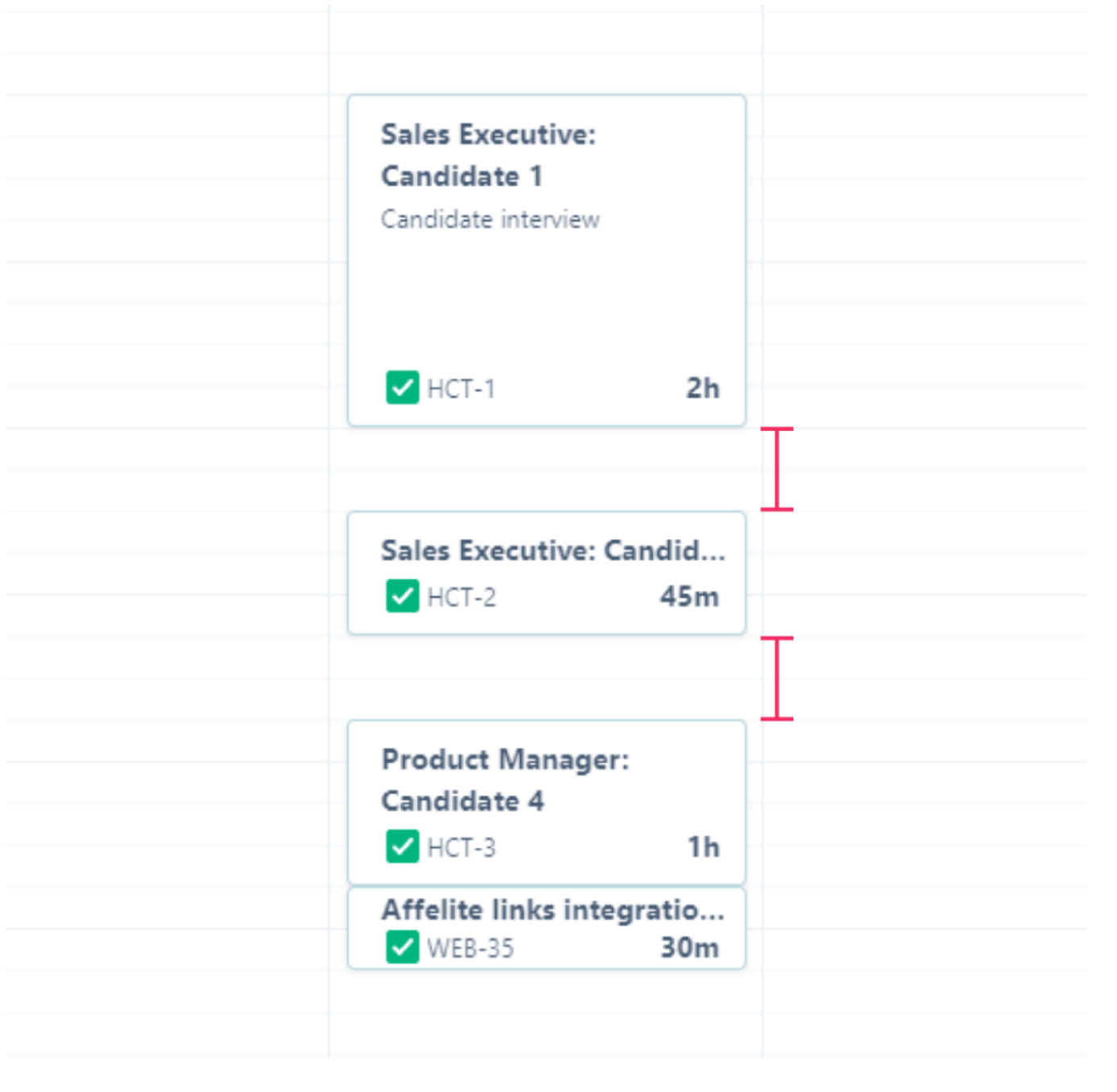


The **rest breaks** metric lets you define the **total duration** of the **rest break(s)** needed on a given working day.

It accumulates the **total duration between** work logs on any given day.

According to some regulations a break must have a minimum duration to be counted as such. The duration can be configured in the global [Settings](#).

The metric also checks whether consecutive work log entries and periods, that cannot be counted as a break, exceed the reference period in total.



## Example configuration

One of the [Preset rules](#) uses the rest break metric to ensure that users have a **minimum of 30 minutes of rest breaks** when their **working day exceeds 6 hours**.

## Edit Minimum daily rest break I

Name \*

Minimum daily rest break I

Description

Check whether users have taken necessary rest breaks (>6 working hours per day).

Operator \*

The operator compares the value you define in the next step with the metric you selected. 

> =

Rest break duration \*

Define the **total duration of the rest break(s)** needed on a given working day as specified below.

Hours \*

0

Minutes \*

30

Reference period (working day) \*

Configure the number of **hours/minutes** that define a **working day**. 

Hours \*

6


Minutes \*

0

Cancel

Submit

### Translation

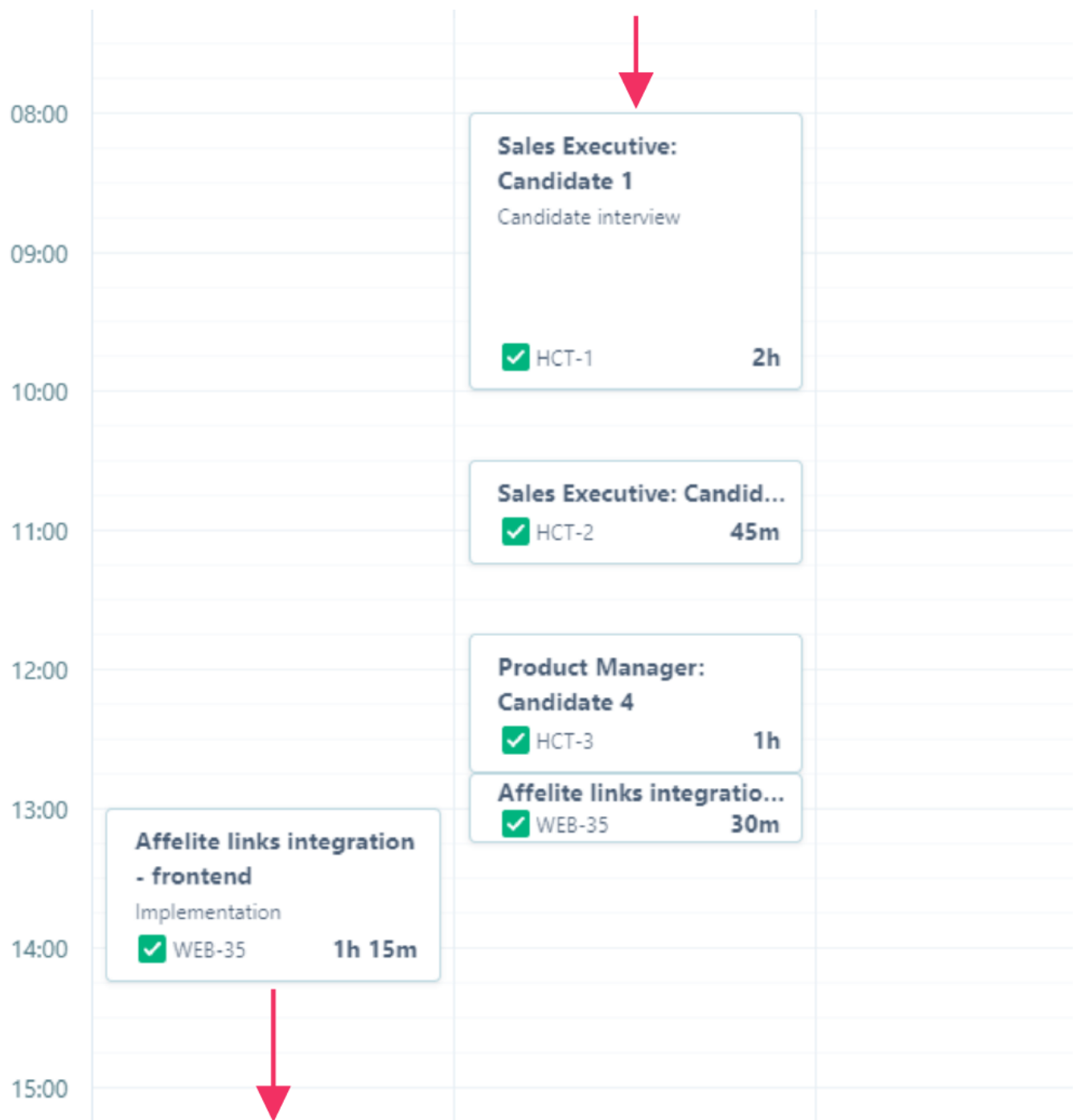
The total time **between** work logs **must be greater than or equal to** ( $\geq$ ) 0 hours and 30 minutes if a user has **logged more than 6 hours** on any given working day for the check to pass .

## Daily rest period



The **daily rest period** metric lets you define the **minimum length** of a **daily rest period**.

It measures the interval between the **End time** of the **last** worklog of any given day and the **Start time** of the **first** workload of the next working day.



## Example configuration

One of the [Preset rules](#) uses the daily rest period to ensure that users have **rested for a minimum of 11 consecutive hours** between two working days.



## Edit Minimum daily rest period

Name \*

Description

Duration \*

Define the **consecutive duration of the daily rest period** per 24-hour period needed on a given working day.

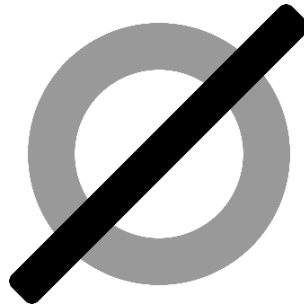
Hours \*

Minutes \*

Cancel

Submit

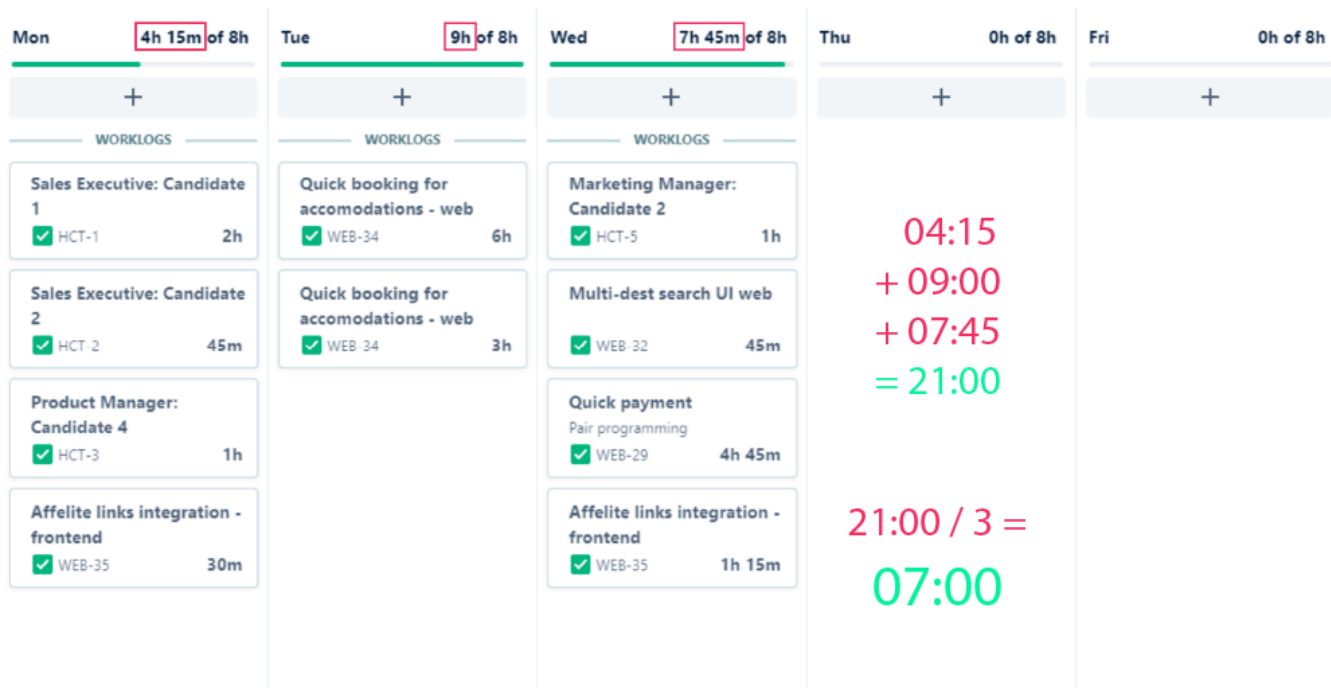
## Average of work logged



The **average of work logged** metric lets you define the **minimum** or **maximum average working time** of a user.

It calculates the **average** daily working time (**total of time logged on any day**) over a given [Tempo period](#). The Tempo period will be selected once you execute this rule in [Reports](#).

In case of a failed check, all involved worklogs will be reported as failed, even if the daily working time might be valid for that particular day.



### Example configuration


One of the [Preset rules](#) uses the average of work logged metric to ensure that the **average daily working time** does **not exceed 8.5 hours**.

#### Edit Maximum average working time

Name \*

Description

Operator \*
 

The operator compares the value you define in the next step with the metric you selected. 

Total work logged \*
 

Configure the average number of **hours/minutes** that a user must have logged in the **reference period** specified below.


Hours \*

Minutes \*

Cancel

Submit

#### Translation

The total of work logged on any given day **must be lower than or equal to (<=)** 8 hours and 30 minutes **on average** in a given Tempo period for the check to pass .

