SmoothPay goPayroll API

SmoothPay goPayroll online API access

Your site manager can obtain your unique access token for your online payroll database from the *Profile menu* (*top-right*) by selecting *API Details*.

Your access token must be provided in <u>every request</u> to the online API.

You MUST keep your token (API key) private - if you think it has been compromised then contact the HelpDesk for a new API token to be issued.

goPayroll API endpoints

- **https://app.gopayroll.net/api/staff** *for all employee-related requests*
- **https://app.gopayroll.net/api/time** *for all related timesheet requests*
- **https://app.gopayroll.net/api/bas** *for PAYG Activity Statement totals*

NOTE: THE API ENDPOINTS CHANGE FROM 4 DEC 2022 to:
- **https://gopayroll.paysauce.com/api/staff** *for all employee-related requests*
- **https://gopayroll.paysauce.com/api/time** *for all related timesheet requests*
- **https://gopayroll.paysauce.com/api/bas** *for PAYG Activity Statement totals*

Parameters

The API has only one parameter, called "payload".

Payload is a JSON item containing:

- token (*provided by SmoothPay helpdesk - must be kept confidential*)
- method (*so the API knows what you need*)
- other key:value pairs (*as required by the method*)

NOTE: *Using parameters generally restricts the amount of data that can be sent via a browser or URL+parameters to the API, so the payload can instead be included in the* **body** *of a POST instead.*

Usage example

This simple example retrieves basic employee details:

```
https://app.gopayroll.net/api/staff?payload={"token":"YourTokenGoes
Here","method":"QueryStaff","id":"1"}
```
or
```
https://app.gopayroll.net/api/staff?payload={"token":"YourTokenGoes
Here","method":"QueryStaff","badge":"FLIFR"}
```
and if employee 1 (or badge FLIFR) exists you will get a JSON response similar to this:

```
{"data":[{"id":
1,"badge":"FLIFR","surname":"Flintstone","forename":"Fred","activity":"","addres
s":"16 Rubble Heights,Rockville
9999","branch":"1","contract":"Permanent","department":"5","dob":"","email":"fre
d@flinstone.com","employment":"Piecework","glcode":"64110","jobcode":"","occupat
ion":"Dino
operator","payrate":"0.000010","phone":"","role":"","start":"2018-11-14","tracki
ng1":"","tracking2":""}]}
```
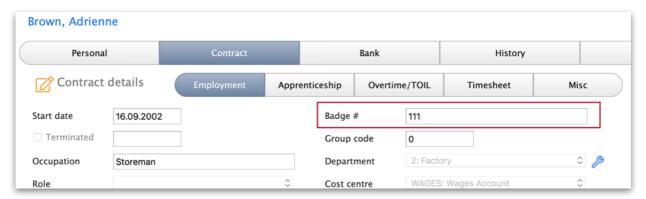
Query operations can be used or tested with simple GET operations (as above).

Add and Update operations can only be performed using POST

API calls may be throttled, capped or blocked to prevent degradation of web-services.

Employee badge

The employee's badge is the common value that identifies an employee and is established here:

Staff..Contract..Employment - example Badge

If you re-use badge numbers as staff come and go (*not generally a good idea*), then the following rules apply:

- the first active employee record having that badge number will be selected
- if no active employee can be identified, terminated staff are checked and the most recently departed employee record with a matching badge will be selected and automatically duplicated to active status

Employee ID

This unique ID is automatically generated when an employee record is added to goPayroll.

You can view the ID in the employee listings, any of the staff API response data, or in any edit view, such as:

Staff..Contract..Employment edit view showing Unique ID and Badge

API Example call and response

- All requests and responses are JSON and follow very simple *key:value* format.

- The client sends a suitably formatted JSON string containing the request and associated information - and hopefully receives some useful data in return.

This example illustrates the common QueryStaff method payload:

```
{"token":"gopayrolltoken","method":"QueryStaff","id":"123"}
```

 or

```
{"token":"gopayrolltoken","method":"QueryStaff","badge":"FLIFR"}
```

responds with (*assuming the employee actually exists*):

```
{"data":[
  {"id":"123","badge":FLIFR,"surname":"Flintstone","forename":"Fred",...}
 ]
}
```

otherwise:

```
{"Status":"Fail","ErrorMessage":"No match"}
```

Each API request payload may contain additional parameter values to control data range or other specific selection criteria - *see the table below for options available for each method*.

All successful requests will present the data element as a JSON array (*as there could be one or more records matching your request*)

All Add or Update requests must have data presented as a JSON array (*as there may be multiple entries to add or update*)

API methods

AddStaff

Provides for adding one or more staff records (**badge and surname are mandatory**).

As many defined *key:value* pairs as required may be included in staff item (*note "id" is ignored as these are allocated by the payroll - define "badge" instead*).

Request example:
```
{...,
 "method":"AddStaff",
 "data":[
  {"badge":"value","key":"value",...},
  {...}
  ]
 }
```

Response:
```
{"data":[
  {"rowid":"value","id":"123","badge":FLIFR,"status":"OK|FAIL","message":"if
an error occurred"}
  ]
 }
```

Response will contain an array of responses for each record supplied (*so you can map your external "badge" to SmoothPay's employee "id" or store the "id" as you need*).

field name	comment
acccode	NZ ACC classification code, defaults to company setting
activity	Activity code, defaults to company setting
address	Comma-separated lines will be converted to multi-line, e.g. "Shed 7,123 Main Street" will import as: 　　　Shed 7 　　　123 Main Street Don't include town or postcode.
agreeddailyvalue	NZ agreed daily value
apphrs	Apprenticeship hours completed
appps	true:false show apprenticeship hours on payslip
apprentice	true:false
appstart	YYYY-MM-DD
award	award code
agreedweeklyvalue	NZ agreed weekly value
badge*	**Mandatory for AddStaff**. Code used to identify employee from your external system
branch	branch code

field name	comment
contract	0=Permanent [default], 1=Temporary, 2=Casual, 3=Contractor/Agent
daysperweek	Typical days per week (default 5)
department	department code
dependants	Number of dependants
dob	YYYY-MM-DD date of birth
email	email address
employment	0=Full time [default], 1=Part time, 2=Piece worker
esctrate	NZ ESCT rate applicable to employee's anticipated earnings, e.g. 0.30=30%
forename	first name/s
gender	M [default] or F
glcode	default GL expense cost centre, defaults to company setting
groupcode	allocates the employee to a "group" having a common code, e.g. "A", "B" etc. to provide optional group selection in staff lists and payrun selection
gstopt	0=None [default], 1=Payslip with GST, 2=Buyer created tax invoice
hecs	AUS HECS debt truelfalse
hoursperday	Typical hours per day (default 8)
hoursperweek	Typical hours per week (default 40)
irregular	truelfalse particularly important for NZ Holidays Act interpretation of Ordinary Weekly Earnings
jobcode	default job code, defaults to company setting
kiwisaver	NZ truelfalse enrolled in KiwiSaver
ksholiday	NZ truelfalse if employee is on KiwiSaver holiday
leaveloading	AUS truelfalse employee entitled to leave loading
medicare	AUS 0=No exemption claimed [default], 1=Half, 2=Full
mcreduct	AUS truelfalse if claiming a medicare reduction (see medicare)
mcspouse	AUS truelfalse
nok	Next of kin (multiple lines will be created from comma-separated values)
payment	0=Cash, 1=Cheque, 2=Direct credit [default]
payslip	0=Normal (print/email) [default], 1=no payslip, 2=email only, 3=print only
occupation	
payrate	
paycycle	0=weekly [default], 1=fortnightly, 2=monthly, 3=4-weekly, 4=half-monthly

field name	comment
phone	
postcode	
pretaxec	NZ truelfalse allows employee to pay own tax on employer contribution, potentially saving more in KiwiSaver
resident	truelfalse
role	role code
sacrate	AUS: Salary sacrifice rate entered as cents in the dollar for percentage, or dollar amount for amount per pay PNG: Nasfund fund housing % entered as Toia per Kina, e.g. 0.1=10%
sacrifice	0=none [default], 1=Amount per pay, 2=percentage
salary	
sfss	AUS student financial supplement truelfalse
start	YYYY-MM-DD start date, defaults to today
state	AUS state: 0=ACT, 1=NSW, 2=NT, 3=QLD, 4=SA, 5=TAS, 6=VIC, 7=WA, 8=Other
slrate	NZ override flat Student Loan repayment percentage, e.g. 0.12=12%
superrate1	employee's superannuation percentage, e.g. 3.0=3%, default company setting
superrate2	employer's superannuation percentage, e.g. 9.5=9.5%, default company setting
superref	super member number (if any)
surname*	**Mandatory for AddStaff**
taxfree	AUS truelfalse claiming tax free threshold for this job
taxrate	flat tax rate overrides normal tax calculation, e.g. 0.2=20%
taxrule	NZ: CAE=13, EDW=14, M=2, MSL=6, ME=3, MESL=4, ND=1 [default], NSW=15, S=7, SSL=10, SB=18, SBSL=19, SH=8, SHSL=9, ST=11, STSL=12, STC=16, WT=17 AUS: Scale 1=1, 2=2, 3=3, 4=4, 5=5, 6=6, 7=7 PNG: 1=No declaration, 2=Non-resident, 3=Resident Solomon Islands: 1=Resident, 2=Non-resident Fiji and Cook Islands: 1=Primary, 2=Secondary Otherwise: 1=Standard
taxnumber	Properly formatted tax number
tfnexempt	AUS exemption for not having TFN: 0=Not exempt [default], 1=applied to ATO, 2=low earner/under 16, 3=pensioner
town	

field name	comment
tracking1	default Xero tracking category 1 option
tracking2	default Xero tracking category 2 option
wagetype	0=waged [default], 1=variable salary, 2=fixed salary, 3=net salary (NZ only)

UpdateStaff

Provides for updating one or more staff records.

As many defined *key:value* pairs as required may be included in staff item (*if both "id" and "badge" are provided, then "id" will always take precedence. If only "badge" is provided then the first active employee having that badge will be updated*).

Request example:

```
{...,
 "method":"UpdateStaff",
 "data":[
  {"id":"value","key":"value",...},
  {...}
 ]
}
```

Response:

```
{"data":[
  {"rowid":"value","id":"123","status":OK|FAIL","message":"if an error
occurred"}
 ]
}
```

Response will contain an array of responses for each record supplied.

Fields are as defined for AddStaff method.

Badge can only be changed if id is passed in the payload data.

Start date change will trigger anniversary date updates if the employee has no pay history (*change should really be made in payroll to ensure associated records are updated and for more thorough integrity checks*)

Contract change will only be effected if the employee has no pay history (*otherwise ignored*)

QueryStaff

Provides for retrieving one or more staff records

Request example for one ID:
```
{...,"method":"QueryStaff",“id”:"123"}
```

or multiple IDs:
```
{...,"method":"QueryStaff",“id”:"123,176,177,182"}
```

or badge:
```
{...,"method":"QueryStaff",“badge”:"FLIFR"}
```

Response:
```
{"data":[
  {"id":"123","badge":FLIFR,"surname":"Flintstone","forename":"Fred",...}
 ]
}
```

QueryStaff options	
id	Preferred method if seeking a single employee record (id may also contain a comma-separated list of IDs).
badge	will retrieve all current staff having matching badge (*WARNING: SmoothPay does NOT prevent duplicate badge numbers as these are often re-used*)
department	retrieve all current staff belonging to a particular department code (as defined in SmoothPay)
branch	retrieve all current staff belonging to a particular branch code (as defined in SmoothPay)
tracking1	staff belonging to tracking category 1
tracking2	staff belonging to tracking category 2
start	all staff who started on a particular date (SQLDate format YYYY-MM-DD)
finish	all staff who finished on a particular date
start + finish	if both keys are provided then all staff who started or finished between the two dates (*empty(start) will be regarded as 1900-01-01, empty(finish) will be regarded as today*) All current staff can be retrieved by simply specifying empty entries for start and finish.

QueryLeave

Provides for retrieving leave balances for one or more current staff and returns an array of employees, each containing an array of leave balance records (*similar to the leave balance enquiry in SmoothPay*).

Request example:

```
{...,"method":"QueryLeave","id":"123"}
```

Response:

```
{"data":[
  {"id":"123","badge":FLIFR,"surname":"Flintstone","forename":"Fred",...,
   "record":[
     {"rowid":"value","paycode":"ANNUAL","name":"Annual
leave","anniversarybalance":"3.2","uom":"Weeks","lastanniversary":"2018-02-15
","accrual":"4","todatebalance":"6.4","value":"6400.00","comment":"a
comment"},
     {...leave...}
   ]
  },
  {...employee...}
 ]
}
```

Additional items may be appended to each employee's record to cater for averages that may be effective at that point in time (e.g. NZ currently requires OWE, AWE, ADP and so on which affect leave valuation)

QueryLeave options	
id	Preferred method if seeking a single employee record (id may also contain a comma-separated list of IDs).
badge	will retrieve all current staff having matching badge (*WARNING: SmoothPay does NOT prevent duplicate badge numbers as these are often re-used*)
department	retrieve all current staff belonging to a particular department code (as defined in SmoothPay)
branch	retrieve all current staff belonging to a particular branch code (as defined in SmoothPay)
tracking1	staff belonging to tracking category 1
tracking2	staff belonging to tracking category 2
paycode	all current staff with entitlements to the specified leave code (e.g. just ANNUAL). Can be used in combination with other options to restrict the data set. Staff not having an entitlement to the specified code are skipped.

QueryLeave results

Each item in the **data** array contains basic employee information plus a **record** array containing entries for each accruing leave type (*or just the selected leave code if specified in the request*)

Leave record results	(in order of appearance)
paycode	Leave paycode
accrualrate	e.g. 4 or 7.692308 etc.
accrualmethod	e.g. Annual weeks, Percentage hours per pay, etc.
lastanniversary	Last anniversary date for this leave record
maximum	Accrual cap
balance	Balance as recorded in payroll
uom	Unit of measure, e.g. weeks, days, hours

> New Zealand has a unique Holidays Act and the following fields are added to the employee information (except contractor/agent), as they have relevance to correct calculation of leave value at the time it is consumed.

NZ-specific results	(in order of appearance)
lastpe	YYYY-MM-DD last period ending date recorded (or start date if never paid)
serviceweeks	weeks between start and lastpe
daysperweek	days per week (default 5)
irregular	truelfalse indicates if employee's work pattern is so irregular that an ordinary week cannot be determined (*4-week average applies instead of Ordinary weekly Earnings from contract*)
parental	truelfalse indicates if employee is within the Parental Leave entitlement period for annual leave accrual (*any leave accrued then consumed whilst in the shadow period is valued using AWE52 only*)
parentalbal	balance of annual leave accrued then consumed whilst in the parental leave shadow period
owe	Ordinary weekly pay determined from contract settings
gross4	Gross earnings for Holidays Act calculations for the last 4 weeks
weeks4divisor	Generally 4 weeks (or less if employee started sooner)
awe4	Average weekly earnings for the last 4 weeks (gross4 / weeks4divisor)
gross52	Gross earnings for Holidays Act calculations for the last 52 weeks
weeks52divisor	Generally 52 weeks (or less if employee started sooner)
awe52	Average weekly earnings for the last 52 weeks (gross52 / weeks52divisor)
agreedweeklyvalue	Agreed weekly value

NZ-specific results	(in order of appearance)
adpcontract	Average daily pay determined from contract settings (owe / daysperweek)
dayspaid52	Total days paid as recorded in pay history for the last 52 weeks
adp52	Average daily pay for the last 52 weeks (gross52 / dayspaid52)
agreeddailyvalue	Agreed daily value

AddTime

Provides for adding one or more timesheet records.

Timesheet records are retrieved on demand from SmoothPay's timesheet import tool (*this provides an added layer of security and control*).

As many defined *key:value* pairs as required may be included in timesheet item.

> Timesheet "paycode" values may also use the **alias codes** understood by SmoothPay (refer Flexi-CSV guide for alias details). This is particularly useful in NZ where you can specify annual leave consumed in hours (AL, ALH or ALC) and the value will determine what proportion of a week it represents).

Request example:

```
{...,
 "method":"AddTime",
 "data":[
  {"id|badge":"value","paycode":"value","units:"value","rate":"value",...},
  {...}
  ]
}
```

Response:

```
{"data":[
  {"rowid":"value","refid:"value","timeid":"value","status":"OK|
FAIL","message":"error description"}
  ]
}
```

> Response will contain an array of responses for each record supplied.

Key values	(**either one of these is mandatory)
id**	employee id
badge**	will match the FIRST current staff record having matching badge (*WARNING: SmoothPay does NOT prevent duplicate badge numbers as these are often re-used*)
refid	If supplied, refid will be returned in the result set so you can identify which entries succeeded and which failed (*and assign the timeID as you see fit*). A rowid will always be returned.
paycode	**MANDATORY**: matches paycode in Smoothpay (*and you can use accepted alias codes as defined in the Alias Paycodes defined below*)
units	**MANDATORY**: units - typically decimal hours (e.g. 1.5 = 1 hour 30 minutes), however it depends on the context (*if an alias paycode has been used, and it is measured in days, then the units will represent a number of days*)
rate	rate (*defaults to employee payrate for time/leave, or lastrate for other pay codes*)
date	YYYY-MM-DD: if not supplied will default to current period end date. if invalid format (e.g. dd/mm/yy etc.) then record will be rejected.

Key values	(**either one of these is mandatory)
department	department code in SmoothPay (otherwise defaults to employee's contract department setting)
branch	retrieve all current staff belonging to a particular branch code (as defined in SmoothPay)
glcode	cost centre, default employee cost centre
jobcode	job code, default employee job code
activity	activity code, default employee activity code
tracking1	Xero tracking category 1 option code
tracking2	Xero tracking category 2 option code
comment	free text

BAS (PAYG Activity Statement totals)

https://app.gopayroll.net/api/bas endpoint provides PAYG Activity Statement totals (Australia)

The payload key:values available are:

- token (required in all API calls)
- range - can be "month" otherwise defaults to "quarter"
- ending - any SQLDate in the month you wish to report to

Request example:
```
{"token": "yourtoken",
 "range": "month",
 "ending": "2022-08-31"
}
```

Response:
```
{"range":"month","ending":"2022-08-31","W1":"1172044","W2":"289654","W3":
0,"W4":0,"W5":"289654"}
```

PHP (Curl) example for AddTime

```php
<?php

// get cURL resource
$ch = curl_init();

// set url
curl_setopt($ch, CURLOPT_URL, 'https://app.gopayroll.net/api/time?
payload=%7B%22token%22:%22_Your_API_Token_here_%22,%22method%22:%22AddTime%22,%2
2data%22:%5B%7B%22id%22:%221902%22,%22paycode%22:%22T1%22,%22units%22:20.5,%22jo
bcode%22:%22ABCJOB%22%7D%5D%7D');

// set method
curl_setopt($ch, CURLOPT_CUSTOMREQUEST, 'POST');

// return the transfer as a string
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);

// set headers
curl_setopt($ch, CURLOPT_HTTPHEADER, [
  'Content-Type: application/json; charset=utf-8',
  'Cookie: serverid=node8418',
]);

// json body
$json_array = [

];
$body = json_encode($json_array);

// set body
curl_setopt($ch, CURLOPT_POST, 1);
curl_setopt($ch, CURLOPT_POSTFIELDS, $body);

// send the request and save response to $response
$response = curl_exec($ch);

// stop if fails
if (!$response) {
  die('Error: "' . curl_error($ch) . '" - Code: ' . curl_errno($ch));
}

echo 'HTTP Status Code: ' . curl_getinfo($ch, CURLINFO_HTTP_CODE) . PHP_EOL;
echo 'Response Body: ' . $response . PHP_EOL;

// close curl resource to free up system resources
curl_close($ch);
```

CURL alternative:

```
curl -X "POST" "https://app.gopayroll.net/api/time" \
     -H 'Content-Type: text/plain' \
     -d "{payload string}"
```

Alias paycodes

Alias paycodes are used to map units represented a certain way to (typically) a leave code that uses different units in payroll.

For example, NZ law requires annual leave to be represented in weeks. A code **ALH** will map units provided in hours to weeks in SmoothPay by calculating the value of those hours at the employee's hourly rate and determines from that what proportion that value is of the best of ordinary weekly earnings, average weekly earnings, agreed weekly value. So, rather than using the "ANNUAL" code defined in SmoothPay, use ALH instead if you want to render annual leave taken in hours in your attendance system.

Assuming these alias-codes are not set up in SmoothPay they'll work as follows:

T1=Ordinary time	Uses employee's ordinary pay rate for account/job combination (if set)
T2=Overtime	Uses 1.5 multiplier automatically - units should be actual hours worked
T3=Double time	Uses 2x multiplier automatically - units should be actual hours worked
AL or ALH=Annual leave (hours)	Portion of week determined from $value (NZ) otherwise hours x rate
ALD=Annual leave in days	Portion of week and value determined from employee's days per week (NZ)
ALW=Annual leave in weeks	Pays best weekly rate (NZ)
BL or BLH=Bereavement Leave (hours)	Pays units @ payrate (NZ: guesses 1 day format does not provide days used)
BLD=Bereavement Leave (days)	Pays units @ average daily pay (NZ)
PTH or 99=Public taken (hours)	Pays units @ payrate (NZ: guesses 1 day format does not provide days used)
PTD=Public taken in days	Pays units @ average daily pay (NZ) otherwise hours x rate
PWH=Public worked (hours)	Pays hours at penal rate (NZ) otherwise hours x rate
SL or SLH=Sick Leave (hours)	Guesses 1 day (NZ: format does not provide days used)
SLD=Sick leave in days	Pays units @ average daily pay (NZ) otherwise units x hours per day x rate
DAYS=Days paid	Sets days paid from value in Days column (required in NZ for average daily pay)
98=Accrue Alternative Leave Day	Accrues units as alternative days accrued
ALT=Consume Alternative day/s	Alternative days consumed
ALTH=ALT in hours	Pays units @ payrate (NZ: guesses 1 day format does not provide days used)
ACC=1st week ACC hours	ACC hours paid
UPL=Unpaid leave	Unpaid leave hours
TERMINATE	Sets the **finalpay** flag in the current payrun record for the selected employee *The import process to goPayroll checks for any staff having the finalpay flag set and calculates any termination entitlements automatically.*

Feedback is welcome.

end