



Flexi-CSV import

There are literally thousands of different time and attendance systems available and many of them can provide useful CSV export. Your perfect attendance system is just waiting for you to find it!

SmoothPay's Flexi-CSV importer lets you identify the layout of the data (*we call it a map*) and import not only attendance data but basic employee data as well.

As long as each row contains a minimum of an employee code (*badge*), preferably a paycode, and some units then it can be imported using Flexi-CSV.

In some cases we may need to create a specific importer (*inflexible timeclock formats, special rules etc.*), however in most cases the Flexi-CSV importer will help you get data into SmoothPay far **faster** than manual keying.

The following sample illustrates the best and most flexible type of CSV output, as it caters for almost any combination of paycode system and can include [optional] date, rate, cost centre, tracking categories, etc. per transaction:

```
badge, paycode, units[, date, rate, cost centre, tracking1, tracking2, comment, etc.]
SMITHJ, T1, 8, 23/08/2020
SMITHJ, T1, 7.5, 24/08/2020
...
```

Other options are available too, such as:

- our [goPayroll API](#) that can be used by external systems (*such as HR, attendance and our own Employee self-service system*) to retrieve from and send data to goPayroll.
- our [Employee self-service](#) is a lightweight tool for staff to lookup leave, history and even use for attendance
- systems and devices already supported by goPayroll's import tools ([see our Timeclock and HR integration section here](#))
- we can also develop bespoke tools for importing from other systems via file import or API (*if available*). There is a cost involved for that service.

Flexi-CSV provides the following import functions

- **Timesheets:** column per time type (*not as flexible as the transaction option*)
- **Pay transactions:** the most powerful and flexible option
- **Employees (new)**
- **Employees (update)**
- **Agency payments:** handy for resetting staff debt balances, amount to pay etc.

The **designated site owner** also has access to **additional import routines** usually available only to our development team to assist with on-boarding employee data when moving to goPayroll. These should only be used if you are confident of the process (***and we strongly advise making a backup before using***):

- **DEV: Leave balances:** imports leave balances from a CSV containing at minimum [badge,leave code,balance]
 - the balance may be the remainder from annual accruals, or a to-date balance
- **DEV: Pay history**
- **DEV: Super funds create/update:** Available for Australia only
- **DEV: Pay codes:** requires [paycode groupID, code, classification rule and optional award rate
 - **groupID codes:** 0=Agency, 1=Allowance, 2=Sundry deduction, 3=Work, 4=Leave, 5=Cost centre, 6=Tracking1, 7=Tracking2, 8=Department, 9=Branch, 10=Award
 - **classification rule:** *contact HelpDesk for a current classification code listing for your country*

Starting the Flexi-CSV importer

Choose **Payrun..Import..Flexi-CSV**

F	Data content	Mapped to
1	123	Skip
2	ORDINARY	*Paycode
3	8	Units
4	22/11/2017	Date (dd/mm/yyyy)
5	example matches ORDINARY payco...	Comment

Example pay transactions CSV file ready for import

Key points	
A	Navigation controls to navigate through records in the CSV data
B	This area displays the record content and lets you select how each item is to be mapped (<i>or skipped</i>) using D (<i>more info below</i>).
C	Selects the type of data being imported: <ul style="list-style-type: none"> • Timesheet data in columns (<i>e.g. a column for badge, ordinary time, another for overtime, etc.</i>) • Timesheet data in rows (<i>each line will identify the employee's badge, a pay code and units and possibly a date, department, cost centre, activity etc</i>) • Employee basic data (<i>each column will contain specific information about the employee</i>) • and so on
D	This lets you choose how the selected field of the record will be used in processing. See below for options available for each type of CSV content.
E	You can drag and drop a CSV file to here for processing, or click to select a file.
F	Easily create a CSV example file in your preferred format
Process	Begins processing

Importing is additive - If you import the same data twice without clearing then the employee will have double entries.

If the employee has no current pay inputs then their standard pay template will be loaded first.

Field mapping

Each field in your CSV data can be mapped as “Skip” (*in which case it is ignored*), or to one of the available options for the type of CSV being imported.

One field in the CSV **must** be mapped to the employee’s Badge.

Employee data

Imports basic employee data from a spreadsheet or CSV file. Bank accounts must be formatted per country payroll guide.

Dates must be in DD/MM/YYYY format.

Mapping option (employee)	Comment
Skip	Ignores field
Badge	Mandatory - record ignored if same badge already exists
First name	
Surname	
Address	
Town	
State	
Postcode	
Country	
DOB	dd/mm/yyyy
Tax#	will auto-correct 11223456 to 011-223-456 etc.
Start date	dd/mm/yyyy
Finish date	dd/mm/yyyy
Phone	
Email	
Bank (code or abbrev)	e.g. ANZ
Bank account#	correctly formatted per country guide, otherwise will store what it's given
Super fund	not required for NZ
Super member#	not required for NZ
Tax code	
Tax rate (if a flat rate applies)	e.g. 0.20 for 20% flat tax rate

Mapping option (employee)	Comment
Paypoint (department)	
Position	
Employment (Full/Part/Casual)	F, P or C (NZ: if importing piece-workers you will need to edit after import)
Pay rate (hourly)	
Salary (annual)	
Pay cycle (W/F/M)	W, F or M

Time (columnar format)

Each row contains all time/leave types for period (e.g. *Ordinary total in one column, Overtime in another etc.*).

This is a convenient format for sites having a single row entry per employee with ordinary, overtime and various leave totals in a specific cell of each row. No paycodes are required.

There is no provision for user coding or sundry allowances or deductions (see the next section for individually coded line transactions for more flexibility).

May have multiple dated lines (otherwise current period end date is used for all transactions).

Importing is additive - If you import the same data twice without clearing then the employee will have double entries.

If the employee has no current pay inputs then their standard pay template will be loaded first.

Mapping option	Comment
Skip	Ignores field
Badge	Mandatory
Ordinary hours	
Overtime hours	
Double time hours	
Pay rate	
Days paid	
Department code	
Cost centre	
Job code	
Activity code	
Comment	
Date (dd/mm/yyyy)	
Date (yyyy-mm-dd)	
NZ - additional options	
Annual leave (hrs)	Determines proportion of week based on Hours x Pay rate \$value as a proportion of best weekly value
Annual leave (days)	Determines proportion of week based on contract settings (daily hours x pay rate \$value as proportion of best weekly value)

Mapping option	Comment
Annual leave (weeks)	Paid at best weekly value as per Holidays Act
Annual leave comment	
Sick leave (hrs)	Makes no sense under NZ law and will be treated as 1 day
Sick leave (days)	Paid using Average Daily Pay
Sick leave comment	
Public taken (hrs)	Makes no sense under NZ law and will be treated as 1 day
Public taken (days)	Paid using Average Daily Pay
Public taken comment	
Public worked (hrs)	
Public worked comment	
Paid rest breaks (hrs)	Generally used for Piece-workers (NZ)
Total hours worked (hrs)	Can accumulate or replace last THW value (setting in Configure) - used to ensure piece-worker receives topup to minimum hourly rate (NZ)
International - additional options	
Annual leave (hrs)	
Annual leave comment	
Sick leave (hrs)	
Sick leave comment	
Public taken (hrs)	
Public taken comment	
Public worked (hrs)	
Public worked comment	

Example columnar format spreadsheet

	A	B	C	D
1	badge	ordinary	overtime	double
2	123	40		
3	124	40	4.5	1.5
4	125	19.5		
5				

Time (transaction format)

Import timesheet data from a spreadsheet or CSV file where pay entry types occur once per row, for as many rows as required.

This is the most flexible option, and is typical of most attendance and time-clocking systems.

It caters for any Work, Leave, Allowance or Sundry Deduction code as well as the special codes required by piece-workers in NZ (THW for Total Hours Worked, and PRB for Paid Rest Breaks) and "Lauranka" codes.

Requires employee badge, pay code and units as a minimum.

We detect "Lauranka" alias-codes T1, T2, THW, PRB, SLD, SLH etc if not discovered in code lookup (*refer alias-code guide below*) to cater (especially) for leave units rendered in hours, days or weeks (esp. important in NZ)

Importing is additive - If you import the same data twice without clearing then the employee will have double entries.

If the employee has no current pay inputs then their standard pay template will be loaded first.

Allowance and Deduction codes **MUST** be established as Units x Rate (*as a "Per pay period" code will always overwrite units with 1*)

Mapping option (transaction)	Comment
Skip	Ignores field
Badge	Mandatory - must match badge of a current staff record
Paycode	Mandatory - "ORDINARY", "T1" etc. Paycode must match one of Work, Leave, Allowance or Sundry Deduction codes (case-insensitive), or failing that then it must match one of the "Lauranka" codes
Units	Mandatory
Rate	Optional - will use payrate (for work codes) or last used rate if not specified
Department	Match on code (auto-create if not found)
Cost centre	Match on code (auto-create if not found)
Job code	Match on code (auto-create if not found)
Activity	Match on code (auto-create if not found)
Comment	Optional
Date (dd/mm/yyyy)	Date in dd/mm/yyyy format
Date (yyyy-mm-dd)	Date in yyyy-mm-dd format

Example Transactional format spreadsheet

Badge	Paycode	Units	Rate	Cost centre	Job	Activity
143	Thinning	36.5	5.556	SPR	Thin	BBN
143	Thinning	34	2.315	SPR	This	LIR
143	Thinning	35	4.5	YOR	Thin	CH
143	Thinning	36	4.5	YOR	Thin	CH
143	Harvest	3	50	MT	Harv	Pear
143	Harvest	1.5	50	MT	Harv	Pear
143	Thinning	0	0.926	SPR	Thin	BBN
143	THW	38				
143	PRB	1.2				

Badge: must match an employee's badge

Paycode: must match one of Work, Leave, Allowance, Deduction paycode (or failing that then it must match one of the Alias paycodes)

Agency payments

Use the Flexi-CSV Agency payments option to import new and update existing Agency payments (*recurring staff debt, loan etc payments*) from a simple CSV file.

Requires employee badge, pay code and one or more of total due (replace), total due (increase), repayment per pay (replace).

As this routine updates existing agency payments we recommend you make a fresh backup before testing to ensure you get the required results.

Mapping option (Agency payments)	Comment
Skip	Ignores field
Badge	Mandatory - must match badge of a current staff record
Agency code	Mandatory - "DEBT" etc. Paycode must match one of your Agency codes (case-insensitive)
Bank statement particulars	Optional - will set the particulars to appear on the recipient agency's bank statement transaction
Add this amount to total due	Optional - Adds to the total amount due and increases the balance due by this amount (e.g. increase a \$250 debt by another \$100)
Replace total amount due with this amount	Optional - Replaces the total amount due
Minimum payment (amount per pay)	Optional - Sets the amount to be deducted each pay period

Example Agency payment format spreadsheet

	A	B	C	D	E
1	Badge	Paycode	Increase	Minimum	
2	123	Debt	100	20	
3	124	Debt	25		
4	125	Loan	1300	130	
5					

Badge: must match an employee's badge

Paycode: must match one of your Agency paycodes

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:

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

Special Provisions

Data from CSV imports is expected to be complete and to take into account any special rules regarding shift allowances, automatic overtime etc (*SmoothPay's auto-overtime and time conversion rules are ignored for imports from this import tool unless the **Allow auto-overtime rules** option is ticked*).

If you have a special requirement (*non-standard file, special processing rules, etc.*) please contact the HelpDesk and we'll see what we can achieve.

Please note

- Imported data will be matched against an Employee's *Badge Number* (see *Contract tab*)
- *An employee will be re-activated if the only match is against a terminated employee record - DO NOT leave terminated staff in your import files!*
- Timesheet data in rows should match the codes you have set up in SmoothPay (*make them the same to suit your import*). If a code cannot be found then the fallback is to use the Alias Paycodes standard (*refer Alias Paycodes above*)
- If UNITS = zero the transaction will not be imported
- *If no RATE is supplied, the employee's default (or last used) rate will be applied - this is great for time entries and standard allowances or deductions but not so great if you're importing sundry deductions with UNITS=1 and RATE=0 (these should be set to UNITS=0, or discarded from the CSV file).*
- For **annual leave**, NZ law requires accrual, value and consumption to be in weeks. Codes that map to hours will produce a proportion of a week used based on best weekly value, daily codes will produce a proportion of a week based on the employee's Days Per Week setting.
- For **sick leave** (and other daily leave types) NZ law requires accrual, value and consumption in days. If the line code maps to a non-compliant setting (e.g. in hours) the entry will be regarded as 1 day consumed and may require manual alteration. If the line code maps to days then the payment will be based on Average Daily Pay for that employee (it is critical that days paid has been correctly recorded in all history for the last 12 months).
- Leave for other countries will be processed as hours

Feedback

We're always keen to do better!

Any and all feedback is appreciated and if you feel we could include better examples, provide more explanation, provide references to additional information, make a process easier to use, or you spot something that isn't working the way it's supposed to - please let us know.