

Syscon PlantStar 4.0 API

Contents

API Endpoints:.....	2
Overview:.....	2
Job Data Endpoint:.....	2
Overview:.....	2
Endpoint Url:.....	2
Parameters:.....	2
Output Structure:.....	3
Fields Returned:.....	6
Job Event Reason Details Endpoint:.....	8
Overview:.....	8
Endpoint Url:.....	8
Parameters:.....	9
Output Structure:.....	10
Fields Returned:.....	12
Database Layer.....	12
Overview:.....	12
Job Data Endpoint:.....	13
Temporary Table Information:.....	13
get_job_data() Function:.....	13
get_job_data_with_json_parameters() Function:.....	13
Example Usages:.....	14
Job Event Reason Detail Endpoint:.....	14
Temporary Table Information:.....	14
get_job_event_reason_details() Function:.....	15
get_job_event_reason_details_with_json_parameters() Function:.....	15
Example Usages:.....	15

API Endpoints:

Overview:

PlantStar provides multiple API endpoints for retrieving specific data from the system.

All endpoints require an API Token, generated for a User by Syscon PlantStar Support.

All endpoint responses return a JSON object containing:

- Data: Contains the data being returned
- Data Types: Contains Data Types for fields returned by the API in “data”
- Errors: Contains any errors returned by the API

```
{  
  "data": [],  
  "data_types": {},  
  "errors": {}  
}
```

Job Data Endpoint:

Overview:

The Job Data Endpoint returns Job and Sub Job data, with dynamic groupings by Machine, Shift, and/or Job.

Endpoint Url:

- /base_app/get_job_data/

Parameters:

Field	Type	Required	Supported Options	Description
api_token	String	Yes	—	API token generated for the user Required for all requests
jobs	Array[string]	No	—	Filters which jobs to retrieve data for If omitted or empty, all jobs are included
job_identifier_type	String	Yes (if jobs provided)	Name, Unique Identifier, UUID,	Specifies how values in jobs

Field	Type	Required	Supported Options	Description
			Product Name	are interpreted Must match the identifier format used in jobs
machines	Array[string]	No	—	Filters which machines to retrieve data for When combined with jobs, further restricts results to specified Machines
machine_identifier_type	String	Yes (if machines provided)	Name, Unique Identifier, UUID	Specifies how values in machines are interpreted Must match the identifier format used in machines
job_fields_to_return	Array[string]	No	Any job output key from this endpoint's "Fields Returned" section	Limits data to specified job fields If omitted, all available fields are returned
start_datetime	Datetime	No	—	Start of the time range for data retrieval Accepts ISO 8601 strings or JSON datetime object
end_datetime	Datetime	No	—	End of the time range for data retrieval Accepts ISO 8601 strings or JSON datetime object
by_machine	Boolean	No	—	Groups data by machine
by_shift	Boolean	No	—	Groups data by shift
by_job	Boolean	No	—	Controls job-level granularity When False, data will be aggregated at the grouping level from by_machine and by_shift
should_flatten	Boolean	No	—	Flattens nested data

Field	Type	Required	Supported Options	Description
				structure

Output Structure:

The response structure for the data varies depending on the `by_machine`, `by_shift`, and `by_job` grouping flags. When `by_job` is `False`, Job data will be aggregated together for the grouping specified using `by_machine` and `by_shift`. When `should_flatten` is `True`, each row in the returned data represents a single instance of Job/Sub Job data based on the value of `by_machine`, `by_shift`, and `by_job`.

Example structure when `by_machine = True`, `by_shift = True`, and `by_job = True`:

```
[
  {
    "machine_id": "machine_1",
    "machine_name": "Machine 1",
    "shift_data": [
      {
        "shift_id": "shift_1",
        "shift_name": "Shift 1",
        "shift_start_datetime": "2026-04-20T08:00:00Z",
        "shift_end_datetime": "2026-04-20T16:00:00Z",
        "job_data": [
          {
            "job_id": "job_123",
            "job_name": "Job 123",
            "product_name": "Product For Job 123",
            "start_datetime": "2026-04-20T08:15:00Z",
            "end_datetime": "2026-04-20T12:00:00Z",
            "run_time_in_seconds": 12000,
            "down_time_in_seconds": 3000,
            "idle_time_in_seconds": 1000,
            "assist_time_in_seconds": 500,
            "up_time_in_seconds": 13000,
            "number_of_good_parts": 500,
            "number_of_rejected_parts": 25,
            "number_of_unconfirmed_parts": 10,
            "number_of_parts": 535,
            ...
          },
          "sub_job_data": [
            {
              "sub_job_id": "sub_job_1",
              "sub_job_name": "Sub Job 1",
              "sub_job_number_of_good_parts": 300,
              "sub_job_number_of_rejected_parts": 10,
              "sub_job_number_of_unconfirmed_parts": 5,
              "sub_job_number_of_parts": 315
            },
            {
              "sub_job_id": "sub_job_2",
              "sub_job_name": "Sub Job 2",
              "sub_job_number_of_good_parts": 200,
```

```
        "sub_job_number_of_rejected_parts": 15,  
        "sub_job_number_of_unconfirmed_parts": 5,  
        "sub_job_number_of_parts": 220  
    }  
  ]  
}  
]  
]
```

Example of flattened structure when `should_flatten = True`:

```
[  
  {  
    "machine_id": "machine_1",  
    "machine_name": "Machine 1",  
    "shift_id": "shift_1",  
    "shift_name": "Shift 1",  
    "shift_start_datetime": "2026-04-20T08:00:00Z",  
    "shift_end_datetime": "2026-04-20T16:00:00Z",  
    "job_id": "job_123",  
    "job_name": "Job 123",  
    "product_name": "Product For Job 123",  
    "start_datetime": "2026-04-20T08:15:00Z",  
    "end_datetime": "2026-04-20T12:00:00Z",  
    "run_time_in_seconds": 12000,  
    "down_time_in_seconds": 3000,  
    "idle_time_in_seconds": 1000,  
    "assist_time_in_seconds": 500,  
    "up_time_in_seconds": 13000,  
    "number_of_good_parts": 500,  
    "number_of_rejected_parts": 25,  
    "number_of_unconfirmed_parts": 10,  
    "number_of_parts": 535,  
    "sub_job_id": "sub_job_1",  
    "sub_job_name": "Sub Job 1",  
    "sub_job_number_of_good_parts": 300,  
    "sub_job_number_of_rejected_parts": 10,  
    "sub_job_number_of_unconfirmed_parts": 5,  
    "sub_job_number_of_parts": 315  
  },  
  {  
    "machine_id": "machine_1",  
    "machine_name": "Machine 1",  
    "shift_id": "shift_1",  
    "shift_name": "Shift 1",  
    "shift_start_datetime": "2026-04-20T08:00:00Z",  
    "shift_end_datetime": "2026-04-20T16:00:00Z",  
    "job_id": "job_123",  
    "job_name": "Job 123",  
    "product_name": "Product For Job 123",  
    "start_datetime": "2026-04-20T08:15:00Z",
```

```

"end_datetime": "2026-04-20T12:00:00Z",
"run_time_in_seconds": 12000,
"down_time_in_seconds": 3000,
"idle_time_in_seconds": 1000,
"assist_time_in_seconds": 500,
"up_time_in_seconds": 13000,
"number_of_good_parts": 500,
"number_of_rejected_parts": 25,
"number_of_unconfirmed_parts": 10,
"number_of_parts": 535,
"sub_job_id": "sub_job_2",
"sub_job_name": "Sub Job 2",
"sub_job_number_of_good_parts": 200,
"sub_job_number_of_rejected_parts": 15,
"sub_job_number_of_unconfirmed_parts": 5,
"sub_job_number_of_parts": 220
}
]

```

Fields Returned:

Display Name	Output Key	Data Type
Machine ID	machine_id	string
Machine Name	machine_name	string
Shift ID	shift_id	string
Shift Name	shift_name	string
Shift Start Datetime	shift_start_datetime	datetime
Shift End Datetime	shift_end_datetime	datetime
Job ID	job_id	string
Job Name	job_name	string
Job Unique Identifier	unique_identifier	string
Job Comments	comments	string
Job Product Name	product_name	string
Job Product Description	product_description	string
Job Production Type	production_type	string
Job Duration Type	duration_type	string
Job Start Datetime	start_datetime	datetime
Job End Datetime	end_datetime	datetime
Job Required By Datetime	required_by_datetime	datetime

Display Name	Output Key	Data Type
Job Run Time In Seconds	run_time_in_seconds	float
Job Idle Time In Seconds	idle_time_in_seconds	float
Job Down Time In Seconds	down_time_in_seconds	float
Job Assist Time In Seconds	assist_time_in_seconds	float
Job Up Time In Seconds	up_time_in_seconds	float
Job Number Of Good Parts	number_of_good_parts	float
Job Number Of Rejected Parts	number_of_rejected_parts	float
Job Number Of Unconfirmed Parts	number_of_unconfirmed_parts	float
Job Number Of Parts	number_of_parts	float
Job Number Of Parts Remaining	number_of_parts_remaining	float
Job Number Of Assists	number_of_assists	integer
Job Number Of Downs	number_of_downs	integer
Job Number Of Totes	number_of_totes	integer
Job OEE	oee	float
Job Performance	performance	float
Job Quality	quality	float
Job Availability	availability	float
Job Scrap Percent	scrap_percent	float
Job Good Parts Percent	good_parts_percent	float
Job Average Cycle Time	average_cycle_time	float
Job Average Cycle Time (Past Cycles)	average_cycle_for_time_past_cycles	float
Job Most Recent Cycle Time	most_recent_cycle_time_in_seconds	float
Job Total Cycle Count	total_cycle_count	float
Job Normal Cycle Count	normal_cycle_count	float
Job Fast Cycle Count	fast_cycle_count	float
Job Slow Cycle Count	slow_cycle_count	float
Job Down Cycle Count	down_cycle_count	float

Display Name	Output Key	Data Type
Job Idle Cycle Count	idle_cycle_count	float
Job Good Parts / Second	good_parts_per_second	float
Job Good Parts / Minute	good_parts_per_minute	float
Job Good Parts / Hour	good_parts_per_hour	float
Job Estimated Good Parts / Second	estimated_good_number_of_parts_per_second	float
Job Estimated Good Parts / Minute	estimated_good_number_of_parts_per_minute	float
Job Estimated Good Parts / Hour	estimated_good_number_of_parts_per_hour	float
Job Estimated Time Remaining	estimated_time_remaining	float
Job Standard Cycle Time	standard_cycle_time	float
Job Standard Parts Per Cycle	standard_number_of_parts_per_cycle	float
Job Standard Rate (Seconds)	standard_rate_in_seconds	float
Job Setup Time (Seconds)	standard_setup_time_in_seconds	float
Job Down Threshold (Seconds)	down_threshold_in_seconds	float
Job Down Threshold Multiplier	down_threshold_multiplier	float
Job Cycle Time Tolerance %	cycle_time_tolerance_percent	float
Job Tool Types	tool_types	string
Job Material Types	material_types	string
Job Cycle Weight	cycle_weight	string
Job Runner Weight	runner_weight	string
Job Total Standard Part Weight	total_standard_part_weight	string
Job Total Actual Part Weight	total_actual_part_weight	string
Job Is Refeeding Runner	is_refeeding_runner	boolean
Job Parts Must Be Confirmed	parts_must_be_confirmed	boolean
Job Auto Complete Tote At Size	should_auto_complete_tote_at_tote_size	boolean
Job Print On Tote Completion	should_print_on_tote_completion	boolean
Sub Job ID	sub_job_id	string
Sub Job Name	sub_job_name	string

Display Name	Output Key	Data Type
Sub Job Good Parts	sub_job_number_of_good_parts	float
Sub Job Unconfirmed Parts	sub_job_number_of_unconfirmed_parts	float
Sub Job Rejected Parts	sub_job_number_of_rejected_parts	float
Sub Job Total Parts	sub_job_number_of_parts	float

Job Event Reason Details Endpoint:

Overview:

The Job Event Reason Details Endpoint returns Job event reason data for Downs, Assists, and Reject Parts, with dynamic groupings by Machine, Shift, and/or Job

Endpoint Url:

- /base_app/get_job_event_reason_details/

Parameters:

Field	Type	Required	Supported Options	Description
api_token	String	Yes	—	API token generated for the user Required for all requests
jobs	Array[string]	No	—	Filters which jobs to retrieve data for If omitted or empty, all jobs are included
job_identifier_type	String	Yes (if jobs provided)	Name, Unique Identifier, UUID, Product Name	Specifies how values in jobs are interpreted Must match the identifier format used in jobs
machines	Array[string]	No	—	Filters which machines to retrieve data for When combined with jobs, further restricts results to specified Machines
machine_identifier_type	String	Yes (if	Name, Unique	Specifies how values in

Field	Type	Required	Supported Options	Description
		machines provided)	Identifier, UUID	machines are interpreted Must match the identifier format used in machines
reason_types_to_return	Array[string]	No	Down, Assist, Reject	Filters data to specified Job Event Reason Types If omitted, all Job Event Reason Types will be returned
start_datetime	Datetime	No	—	Start of the time range for data retrieval Accepts ISO 8601 strings or JSON datetime object
end_datetime	Datetime	No	—	End of the time range for data retrieval Accepts ISO 8601 strings or JSON datetime object
by_machine	Boolean	No	—	Groups data by machine
by_shift	Boolean	No	—	Groups data by shift
by_job	Boolean	No	—	Controls job-level granularity When False, data will be aggregated at the grouping level from by_machine and by_shift
should_flatten	Boolean	No	—	Flattens nested response structure

Output Structure:

The structure for the data varies depending on the by_machine, by_shift, and by_job grouping flags. When by_job is False, Job Event Reason data will be aggregated together for the grouping specified using by_machine and by_shift. When should_flatten is True, each row in the returned data represents a single reason instance.

Example structure when by_machine = True, by_shift = True, and by_job = True:

```
[
  {
```

```

"machine_id": "machine_1",
"machine_name": "Machine 1",
"shift_data": [
  {
    "shift_id": "shift_1",
    "shift_name": "Shift 1",
    "shift_start_datetime": "2026-04-20T06:00:00Z",
    "shift_end_datetime": "2026-04-20T14:00:00Z",
    "job_data": [
      {
        "job_id": "job_123",
        "job_name": "Job 123",
        "job_start_datetime": "2026-04-20T06:30:00Z",
        "job_end_datetime": "2026-04-20T10:00:00Z",
        "product_name": "Product For Job 1",
        "down_reasons": {
          "Maintenance": {
            "reason": "Maintenance",
            "job_event_reason_type": "Down",
            "total_time_in_seconds": 1200,
            "total_count": 2
          }
        },
        "assist_reasons": {
          "Operator Needed": {
            "reason": "Operator Needed",
            "job_event_reason_type": "Assist",
            "total_time_in_seconds": 300,
            "total_count": 1
          }
        },
        "reject_reasons": {
          "sub_job_uuid": {
            "sub_job_name": "Sub Job 1",
            "sub_job_reject_reasons": {
              "Defect": {
                "reason": "Defect",
                "job_event_reason_type": "Reject",
                "total_count": 5
              }
            }
          }
        }
      }
    ]
  }
]

```

Example of flattened structure when should_flatten = True:

```

[
  {
    "machine_id": "machine_1",
    "machine_name": "Machine 1",
    "shift_id": "shift_1",

```

```

    "shift_name": "Shift 1",
    "shift_start_datetime": "2026-04-20T06:00:00Z",
    "shift_end_datetime": "2026-04-20T14:00:00Z",
    "job_id": "job_123",
    "job_name": "Job 123",
    "job_start_datetime": "2026-04-20T06:30:00Z",
    "job_end_datetime": "2026-04-20T10:00:00Z",
    "product_name": "Product For Job 123",
    "reason": "Maintenance",
    "job_event_reason_type": "Down",
    "total_time_in_seconds": 1200,
    "total_count": 2
  },
  {
    "machine_id": "machine_1",
    "machine_name": "Machine 1",
    "shift_id": "shift_1",
    "shift_name": "Shift 1",
    "shift_start_datetime": "2026-04-20T06:00:00Z",
    "shift_end_datetime": "2026-04-20T14:00:00Z",
    "job_id": "job_123",
    "job_name": "Job 123",
    "job_start_datetime": "2026-04-20T06:30:00Z",
    "job_end_datetime": "2026-04-20T10:00:00Z",
    "product_name": "Product For Job 123",
    "reason": "Operator Needed",
    "job_event_reason_type": "Assist",
    "total_time_in_seconds": 300,
    "total_count": 1
  },
]

```

Fields Returned:

Display Name	Field Name	Data Type
Machine ID	machine_id	string
Machine Name	machine_name	string
Shift ID	shift_id	string
Shift Name	shift_name	string
Shift Start Datetime	shift_start_datetime	datetime
Shift End Datetime	shift_end_datetime	datetime
Job ID	job_id	string
Job Name	job_name	string
Start Datetime	job_start_datetime	datetime
End Datetime	job_end_datetime	datetime

Display Name	Field Name	Data Type
Product Name	product_name	string
Job Event Reason Type	job_event_reason_type	string
Job Event Reason Total Time In Seconds	total_time_in_seconds	float
Job Event Reason Total Count	total_count	float
Sub Job ID	sub_job_id	string
Sub Job Name	sub_job_name	string
Sub Job Event Reason Count	sub_job_event_reason_count	float

Database Layer

Overview:

Each endpoint has a corresponding set of functions within the PlantStar API database that can be used to retrieve data from the API using SQL.

These functions take a set of parameters similar to the endpoints parameters, then create a Temporary Table and store the retrieved data within that table, to be accessed or queried upon within the same database session. The name for the temporary table can be passed into each function; if one is not provided then a default table name is used.

If the temporary table already exists in the session, it will be dropped and re-made with new data.

Job Data Endpoint:

Temporary Table Information:

- Default Name: **temp_job_data**
- Column Schema:

- o All columns and their data types are derived from the fields returned by the Job data endpoint. The table column schema will dynamically change depending on the parameters provided.
 - By_machine – These Machine Fields will be included:
 - Machine ID, Machine Name
 - By_shift – These Shift fields will be included
 - Shift ID, Shift Name, Shift Start Datetime, Shift End Datetime
 - By_job – These Job fields will be included
 - Job ID, Job Name
 - job_fields_to_return – If fields are specified, only those fields plus any above from the grouping parameters, will be in the table. If it is left empty, all Job fields will be included.

get_job_data() Function:

Parameters:

```
get_job_data(
  api_token text,
  jobs text[] default array[]::text[],
  job_identifier_type text default '',
  machines text[] default array[]::text[],
  machine_identifier_type text default '',
  job_fields_to_return text[] default array[]::text[],
  start_datetime text default '',
  end_datetime text default '',
  by_machine bool default false,
  by_shift bool default false,
  by_job bool default false,
  temp_table_name text default 'temp_job_data'
)
```

get_job_data_with_json_parameters() Function:

Parameters:

```
get_job_data_with_json_parameters(
  endpoint_parameters JSONB,
  temp_table_name TEXT default 'temp_job_data'
)
```

Example Usages:

get_job_data() with default table name:

```
SELECT get_job_data(
  api_token := 'your_api_token_here',
  start_datetime := '2026-04-20T00:00:00Z',
  end_datetime := '2026-04-21T00:00:00Z',
  job_fields_to_return := ARRAY[
```

```

        'number_of_good_parts', 'run_time_in_seconds', 'oee'
    ],
    by_machine := true,
    by_shift := true,
    by_job := true
);
SELECT * FROM temp_job_data;

```

get_job_data() with custom table name:

```

SELECT get_job_data(
    api_token := 'your_api_token_here',
    job_fields_to_return := ARRAY[
        'number_of_good_parts', 'run_time_in_seconds', 'oee'
    ],
    by_machine := true,
    by_shift := true,
    by_job := true,
    temp_table_name := 'custom_job_data_table'
);
SELECT * FROM custom_job_data_table;

```

get_job_data_with_json_parameters():

```

SELECT get_job_data_with_json_parameters(
    '{
        "api_token": "your_api_token_here",
        "by_machine": true,
        "by_shift": true,
        "by_job": true,
        "start_datetime": "2026-04-20T00:00:00Z",
        "end_datetime": "2026-04-21T00:00:00Z"
    }'::jsonb
);
SELECT job_name, down_time_in_seconds FROM temp_job_data;

```

Job Event Reason Detail Endpoint:

Temporary Table Information:

- Default Name: **temp_job_event_reason_details**
- Column Schema:
 - o All columns and their data types are derived from the fields returned by the Job Event Reason Details endpoint. The table column schema will dynamically change depending on the parameters provided.
 - By_machine – These Machine Fields will be included:
 - Machine ID, Machine Name
 - By_shift – These Shift fields will be included
 - Shift ID, Shift Name, Shift Start Datetime, Shift End Datetime
 - By_job – These Job fields will be included

- Job ID, Job Name

get_job_event_reason_details() Function:

Parameters:

```
get_job_event_reason_details(  
  api_token text,  
  jobs text[] default array[]::text[],  
  job_identifier_type text default '',  
  machines text[] default array[]::text[],  
  machine_identifier_type text default '',  
  reason_types_to_return text[] default array[]::text[],  
  start_datetime text default '',  
  end_datetime text default '',  
  by_machine bool default false,  
  by_shift bool default false,  
  by_job bool default false,  
  temp_table_name text default 'temp_job_event_reason_details'  
)
```

get_job_event_reason_details_with_json_parameters() Function:

Parameters:

```
get_job_event_reason_details_with_json_parameters(  
  endpoint_parameters JSONB,  
  temp_table_name TEXT default 'temp_job_event_reason_details'  
)
```

Example Usages:

get_job_event_reason_details() with default table name:

```
SELECT get_job_event_reason_details(  
  api_token := 'your_api_token_here',  
  start_datetime := '2026-04-20T00:00:00Z',  
  end_datetime := '2026-04-21T00:00:00Z',  
  reason_types_to_return := ARRAY['Down', 'Assist'],  
  by_machine := true,  
  by_shift := true,  
  by_job := true  
);  
SELECT * FROM temp_job_event_reason_details;
```

get_job_event_reason_details() with custom table name:

```
SELECT get_job_event_reason_details(  
  api_token := 'your_api_token_here',  
  by_machine := true,  
  by_shift := true,  
  by_job := true,  
  temp_table_name := 'custom_job_event_reason_details_table'
```

```
);  
SELECT * FROM custom_job_event_reason_details_table;
```

get_job_event_reason_details_with_json_parameters():

```
SELECT get_job_event_reason_details_with_json_parameters (  
  '{  
    "api_token": "your_api_token_here",  
    "by_machine": true,  
    "by_shift": true,  
    "by_job": true,  
    "reason_types_to_return": ["Down", "Assist"]  
  }'::jsonb  
);  
SELECT * FROM temp_job_event_reason_details;
```