The Coding metrics API provides the coding metrics from the Team health insights report for use outside of Flow.

**Note:** The Coding metrics API is not documented in Flow’s interactive API documentation (opens in new tab).

Who can use this?

- Core
- Plus ✓

The Coding metrics API averages **Coding days, Commits per day, Impact, and Efficiency** for the time period selected in the report’s date range filter.

If you haven’t used the Flow customer API before, learn more about how to make a Flow API call.

In this article

Permissions
Calling the API
Supported parameters
Result formats
Examples

**Permissions**

Users calling the Coding metrics API must have Metrics API permissions as well as a valid and unique API key. To enable API permissions for, enable the Manage User Permissions permission. Learn more about administrative permissions and API keys in Flow.

**Important:** If a user does not have both the permission and the API key, they can't use the Coding metrics API. If your user can’t call the Coding metrics API, verify they have both the permission and API key.
To enable the Metrics API permission:

1. In the Settings left pane navigation under User management, click Users.
2. Search for the user you want to give the Metrics API permission to.
3. Click the user’s name.
4. Click the Access tab.
5. In the API access column, check the Metrics API box.
6. Click Save changes.
7. Repeat this process for all you want to access the Coding metrics API.

Note: These changes don't affect existing permissions for other Flow customer APIs.

Calling the API

Call the API endpoint over standard HTTPS. Use any tool or programming language that can add the authorization: bearer <API Key> header.

The base URL for this API is: https://flow.pluralsight.com/v3/customer/metrics/code_fundamentals/period_metrics/.

Supported parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Required or Optional</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>start_date</td>
<td>Required</td>
<td>Specifies the start of the date range.</td>
<td>start_date=2021-05-01</td>
</tr>
<tr>
<td>end_date</td>
<td>Optional</td>
<td>Specifies the end date of the date range. If unspecified, uses the current date.</td>
<td>end_date=2021-05-15</td>
</tr>
<tr>
<td>include_nested_teams</td>
<td>Optional</td>
<td>Determines whether data from nested teams are included.</td>
<td>include_nested_teams=true</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>include_nested_teams=false</td>
</tr>
<tr>
<td>Parameter</td>
<td>Required or Optional</td>
<td>Description</td>
<td>Example</td>
</tr>
<tr>
<td>-----------------</td>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>aggregate</td>
<td>Optional</td>
<td>Matches and teams visible at aggregate depth. Shows anonymized aggregate metrics without individually identifiable details.</td>
<td>aggregate=true, aggregate=false</td>
</tr>
<tr>
<td>atomic</td>
<td>Optional</td>
<td>Matches and teams visible at atomic depth. Shows granular or individually identifiable details in metrics if you have the rights to see them.</td>
<td>atomic=true, atomic=false</td>
</tr>
<tr>
<td>resolution</td>
<td>Required</td>
<td>Specifies whether metrics are averaged over a week or date range.</td>
<td>resolution=week, resolution=period</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- period returns averages for each metric over the given date range.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- week returns details for each week over the given date range.</td>
<td></td>
</tr>
<tr>
<td>apex_user_id</td>
<td>Optional</td>
<td>Matches on the ID of a specific apex user.</td>
<td>apex_user_id=12345</td>
</tr>
<tr>
<td>apex_user_id__in</td>
<td>Optional</td>
<td>Matches on a set of provided apex user IDs.</td>
<td>apex_user_id__in=12,34,56</td>
</tr>
<tr>
<td>team_id</td>
<td>Optional</td>
<td>Matches on an ID of a specific Flow team.</td>
<td>team_id=12345</td>
</tr>
<tr>
<td>team_id__in</td>
<td>Optional</td>
<td>Matches on a set of provided Flow team IDs.</td>
<td>team_id__in=21,34,65,67</td>
</tr>
<tr>
<td>repo_id</td>
<td>Optional</td>
<td>Matches on exact repo ID.</td>
<td>repo_id=12345</td>
</tr>
<tr>
<td>repo_id__in</td>
<td>Optional</td>
<td>Matches on a set of provided repo IDs.</td>
<td>repo_id__in=1,2,3,4</td>
</tr>
<tr>
<td>repo_id_not__in</td>
<td>Optional</td>
<td>Matches on all repo IDs except the provided repo ID set.</td>
<td>repo_id_not__in=67,45,34</td>
</tr>
<tr>
<td>repo_tag_id</td>
<td>Optional</td>
<td>Matches on an exact repo tag ID.</td>
<td>repo_tag_id=5123</td>
</tr>
<tr>
<td>repo_tag_id__in</td>
<td>Optional</td>
<td>Matches on a set of provided repo tag IDs.</td>
<td>repo_tag_id__in=34,21,78,45</td>
</tr>
<tr>
<td>repo_name</td>
<td>Optional</td>
<td>Matches on any repo name which contains the search parameter.</td>
<td>repo_name=ExampleTeam</td>
</tr>
</tbody>
</table>
Result formats

The API returns results in JSON. The shape of the data depends on the `resolution` flag.

Example results for when the `resolution` flag is set to `week`:

```json
{
    "count": 5,
    "results": [
        {
            "period_start_date": "2021-05-31",
            "active_days": 2.2,
            "commit_count": 4.2,
            "total_impact": 94.1,
            "total_efficiency": 78.1
        },
        {
            "period_start_date": "2021-06-07",
            "active_days": 2.8,
            "commit_count": 5.5,
            "total_impact": 111.1,
            "total_efficiency": 58
        },
        {
            "period_start_date": "2021-06-14",
            "active_days": 2.9,
            "commit_count": 3.7,
            "total_impact": 114.4,
            "total_efficiency": 78.1
        }
    ]
}
```
Note: Weeks with no data are omitted from the results.

Example results for when the resolution flag is set to period:

```json
{
  "count": 1,
  "results": [
    {
      "period_start_date": "2021-06-21",
      "active_days": 2.7,
      "commit_count": 3.9,
      "total_impact": 122.9,
      "total_efficiency": 76.9
    },
    {
      "period_start_date": "2021-06-28",
      "active_days": 1.6,
      "commit_count": 5.7,
      "total_impact": 85.1,
      "total_efficiency": 65.4
    }
  ]
}
```
"active_days": 2.8,
"commit_count": 4.5,
"total_impact": 121.7,
"total_efficiency": 68.2
]
]
}

Error format
If there is an error in any of the request parameters, the request will return an error.

Example error:
{
  "start_date":
  ["Provide a valid date. Date cannot be more than 3 years prior to now."],
  "aggregate":
  ["Provide a valid choice. 'yes' is not one of the available choices: ['true', 'false']"],
  "resolution":
  ["Provide a valid choice. 'month' is not one of the available choices: [period', 'week']"]
}

If there is an error in the request parameters, the system returns a non-200 status code in addition to the response above. This is normally a 400 error for any form of input error, and a 500 error for any back-end exceptions. For back-end exceptions, the error is generic.

Example generic error:
{
  "error": ["Unknown error occurred"]
}
Examples

Get Coding metrics for a team

First, use the Teams API endpoint to find your team ID. In the response, note the team id for use in the next step.

Example query: https://flow.pluralsight.com/v3/customer/core/teams/?name=ExampleTeam.

Example response:

```json
{
  "count":1,
  "next":null,
  "previous":null,
  "results":[
    {
      "id":12345,
      "name":"ExampleTeam",
      "description":"
      "org":1234,
      "avatar":null,
      "vendor":null,
      "created_at":"2020-09-22T15:55:26.482759",
      "parent":54321,
      "path":"11188.99591.52870",
      "ancestors":["ParentTeam","AnotherTeam","ExampleTeam"],
      "visibility":"SHOW",
      "team_level":3,
      "inherited_depth":"atomic",
      "nested_teams":0,
    }
  ]
}
```
To get the overall average of your team's Coding metrics over a specified period, insert the team ID in the team_id parameter of the Coding metrics API endpoint. Set resolution to period.


Example response:

{
  "count": 1,
  "results": [
    {
      "active_days": 2.4,
      "commit_count": 3.5,
      "total_impact": 97.7,
      "total_efficiency": 83.1
    }
  ]
}

To get the weekly averages of your team's Coding metrics for a specified period, insert the team ID in the team_id parameter of the Coding metrics API endpoint. Set resolution to week.

Example query: https://flow.pluralsight.com/v3/customer/metrics/code_fundamentals/period_metrics/?start_date=2021-06-01&end_date=2021-06-30&team_id=12345&include_nested_teams=true&resolution=week.
Example response:

```json
{
    "count": 5,
    "results": [
        {
            "period_start_date": "2021-05-31",
            "active_days": 1.0,
            "commit_count": 4.0,
            "total_impact": 18.5,
            "total_efficiency": 52.6
        },
        {
            "period_start_date": "2021-06-07",
            "active_days": 1.0,
            "commit_count": 4.0,
            "total_impact": 33.8,
            "total_efficiency": 94.9
        },
        {
            "period_start_date": "2021-06-14",
            "active_days": 3.0,
            "commit_count": 3.5,
            "total_impact": 118.0,
```

Note: Weeks with no commit activity are omitted from the response.
Get Coding metrics for an individual

To get Coding metrics for a single user, find the user’s Apex user ID in the user’s endpoint of the Flow customer API. See Users API to learn more about retrieving a user’s Apex user ID. Note the id for the user you want to see the coding metrics for.


**Note:** This example uses the team ID from the previous example to find the Apex user ID. Your request may look different if you use a different parameter to find the user's Apex user ID.
Example response:
{
    "count": 4,
    "next": null,
    "previous": null,
    "results": [

        {
            "id": 1234567,
            "name": "Stephanie Ventura",
            "email": "stephanie-ventura@examplecompany.com",
            "hidden_from_reports": false,
            "org_id": 1234,
            "created_at": "2019-06-24T21:50:40.880538",
            "teams": [

                {
                    "id": 12345,
                    "name": "ExampleTeam",
                    "description": "",
                    "org": 1234,
                    "avatar": null,
                    "vendor": null,
                    "created_at": "2020-09-22T15:55:26.482759",
                    "parent": 54321,
                    "path": "11188.99591.52870",
                    "visibility": "SHOW",
                }

            ]

        }

    ]

}
null,

0,

4,

4,

"depth":"inherit"

},

{

... false,

null,

true,

true,

false,

false,

null,

"2020-11-17T04:07:57.948366",

"login": {
Once you have the Apex user ID, insert it into the `apex_user_id` parameter and set `resolution` to `period` to return the Coding metrics' overall averages across a specified period.


Example response:

```json
{
    "count": 1,
    "results": [
        {
            "active_days": 3.3,
            "commit_count": 4.2,
            "total_impact": 154.9,
            ...
        },
        {
            ...
        },
        {
            ...
        },
        ...
    ]
}
```
To get weekly averages, use the apex_user_id parameter and set resolution to week.


Example response:

```json
{
    "count": 5,
    "results": [
        {
            "period_start_date": "2021-05-31",
            "active_days": 2.8,
            "commit_count": 3.9,
            "total_impact": 123.8,
            "total_efficiency": 76.2
        },
        {
            "period_start_date": "2021-06-07",
            "active_days": 3.0,
            "commit_count": 5.1,
            "total_impact": 136.6,
            "total_efficiency": 60.3
        },
        {
            "total_efficiency": 71.1
        }
    ]
}
```
"period_start_date": "2021-06-14",

"active_days": 3.2,

"commit_count": 3.9,

"total_impact": 141.2,

"total_efficiency": 67.1
}

{

"period_start_date": "2021-06-21",

"active_days": 3.2,

"commit_count": 3.6,

"total_impact": 168.4,

"total_efficiency": 79.3
}

{

"period_start_date": "2021-06-28",

"active_days": 2.1,

"commit_count": 4.4,

"total_impact": 99.3,

"total_efficiency": 74.1
}

}

If you need help, please contact Pluralsight Support.