**A Sprint** fixed period or duration in which a team works to complete a set of tasks

An **Epic** is a **big task or project** that is too large to complete in one sprint. It is broken down into **smaller tasks (stories)** that can be completed over multiple sprints.

A **Story** is a small task . It is part of an **Epic**.

A **Story Point** is a number that represents how much effort a story takes to complete. (usually in form of Fibonacci series)

1. Very Easy task
2. Easy task
3. Moderate task
4. Difficult task

**Here is your** Sprint Planning and Velocity Calculation **for the Butterfly Identification Project, using the same structure and estimation logic as your example:**

## Key Agile Terms

* Sprint: Fixed period (e.g., 5 days) in which the team completes a set of tasks.
* Epic: Large feature (e.g., "Species Identification") broken into smaller stories.
* Story: Small, actionable task (e.g., "Upload butterfly image").
* Story Point: Effort estimate for a story, often using Fibonacci series (1, 2, 3, 5, 8, ...).

Story Point Estimation Reference:

* Very Easy task – 1 point
* Easy task – 2 points
* Moderate task – 3 points
* Difficult task – 5 points

## Sprint 1 (5 Days)

Epic: Data Preparation & User Authentication

| Story | Story Points |
| --- | --- |
| Collect butterfly image datasets | 2 |
| Load data into system | 1 |
| Handle missing values | 3 |
| Encode categorical labels | 2 |

Total Story Points (Sprint 1): 8

## Sprint 2 (5 Days)

Epic: Model & Interface Development

| Story | Story Points |
| --- | --- |
| Build image classification model | 5 |
| Test model accuracy | 3 |
| Develop user registration/login pages | 3 |
| Deploy model with Flask API | 5 |

Total Story Points (Sprint 2): 16

## Velocity Calculation

* Total Story Points Completed: 8 (Sprint 1) + 16 (Sprint 2) = 24
* Number of Sprints: 2

Velocity = Total Story Points Completed / Number of Sprints = 24 / 2 = 12

Your team’s velocity is 12 Story Points per Sprint.

## Example Table for Reference

| Sprint | Epic | Story | Story Points |
| --- | --- | --- | --- |
| Sprint 1 | Data Preparation & Authentication | Collect butterfly image datasets | 2 |
|  |  | Load data into system | 1 |
|  |  | Handle missing values | 3 |
|  |  | Encode categorical labels | 2 |
| Sprint 2 | Model & Interface Development | Build image classification model | 5 |
|  |  | Test model accuracy | 3 |
|  |  | Develop user registration/login pages | 3 |
|  |  | Deploy model with Flask API | 5 |

You can use this format for your project documentation and sprint reviews.  
This approach aligns with standard Agile velocity tracking and estimation practices[1](https://www.atlassian.com/agile/project-management/velocity-scrum)[2](https://www.hatica.io/blog/calculate-sprint-velocity/)[3](https://www.lucidchart.com/blog/how-to-estimate-sprint-velocity)[6](https://www.resolution.de/post/sprint-velocity-formula/).

1. <https://www.atlassian.com/agile/project-management/velocity-scrum>
2. <https://www.hatica.io/blog/calculate-sprint-velocity/>
3. <https://www.lucidchart.com/blog/how-to-estimate-sprint-velocity>
4. <https://business.adobe.com/blog/basics/velocity>
5. <https://dev.to/hatica/how-to-calculate-sprint-velocity-for-agile-planning-5akj>
6. <https://www.resolution.de/post/sprint-velocity-formula/>
7. <https://www.itsdart.com/blog/how-to-calculate-velocity-for-first-sprint>
8. <https://asana.com/resources/sprint-velocity>