

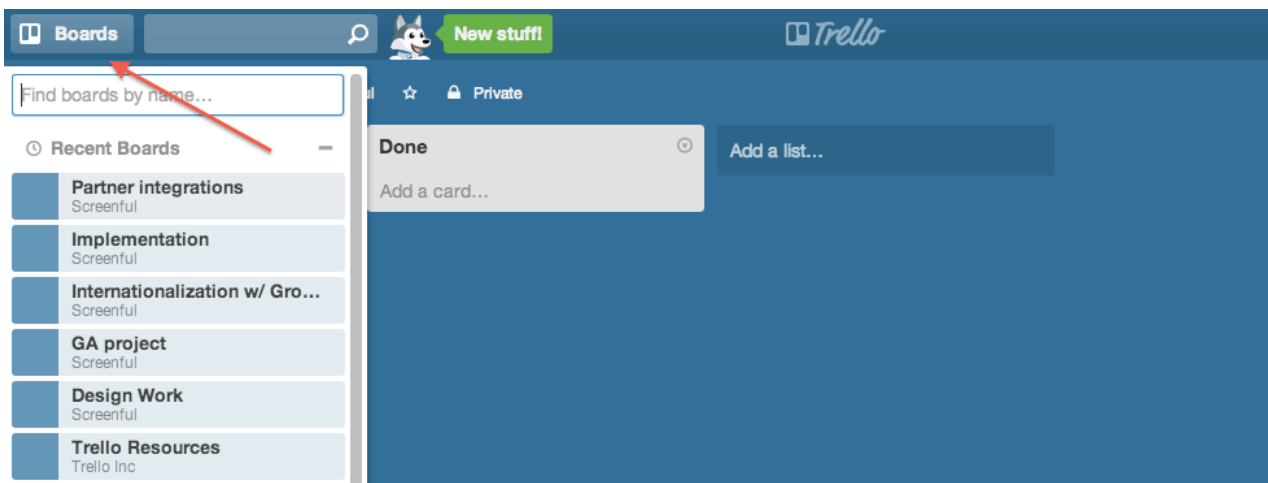
# QUICK SETUP GUIDE (TRELLO)

## Overview

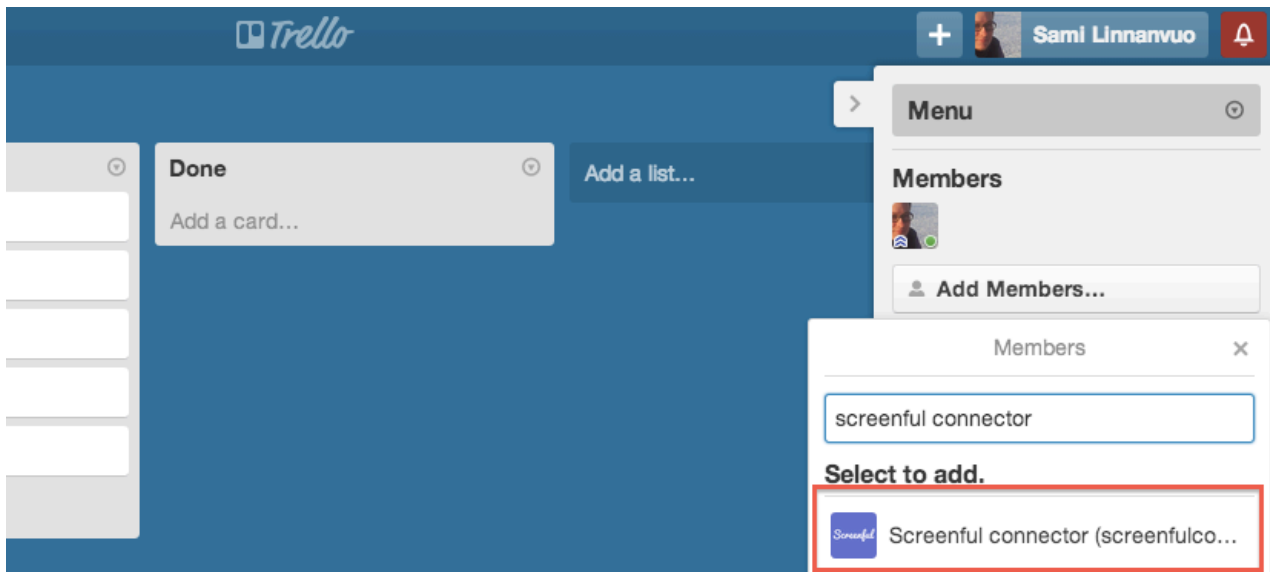
This document outlines the steps needed to setup *Screenful for Agile* with your *Trello*. You need to expose your data to *Screenful connector* that periodically checks for updates in order to keep your dashboard up-to-date. To enable connector to access your task data, you need to grant access to a specific board in your Trello account.

## Granting Access to your Trello Account

To provide access for Screenful connector to your Trello installation, first select the board that you want to grant access to by clicking [Boards](#) on top left corner and then selecting the board from the select menu



Once on the correct board, click [Add Members](#) on the right side menu and type [Screenful connector](#) into the text box and select *Screenful connector –user* from the select menu.

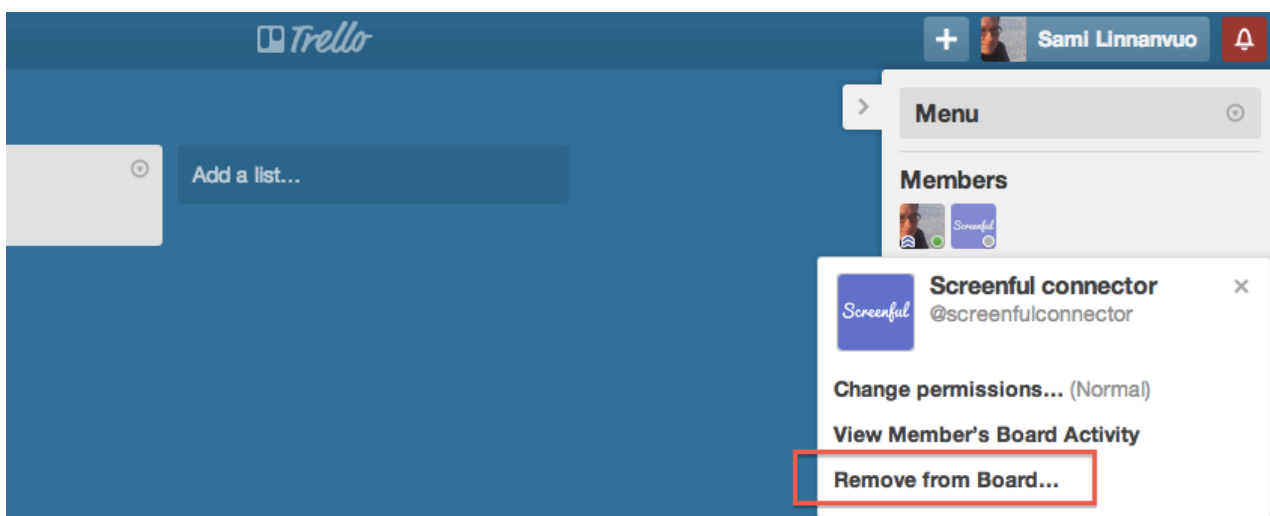


Now Screenful connector can access the data on the Trello board in question.

**Note!** Screenful connector will only read data and does not modify any of the data on your Trello board.

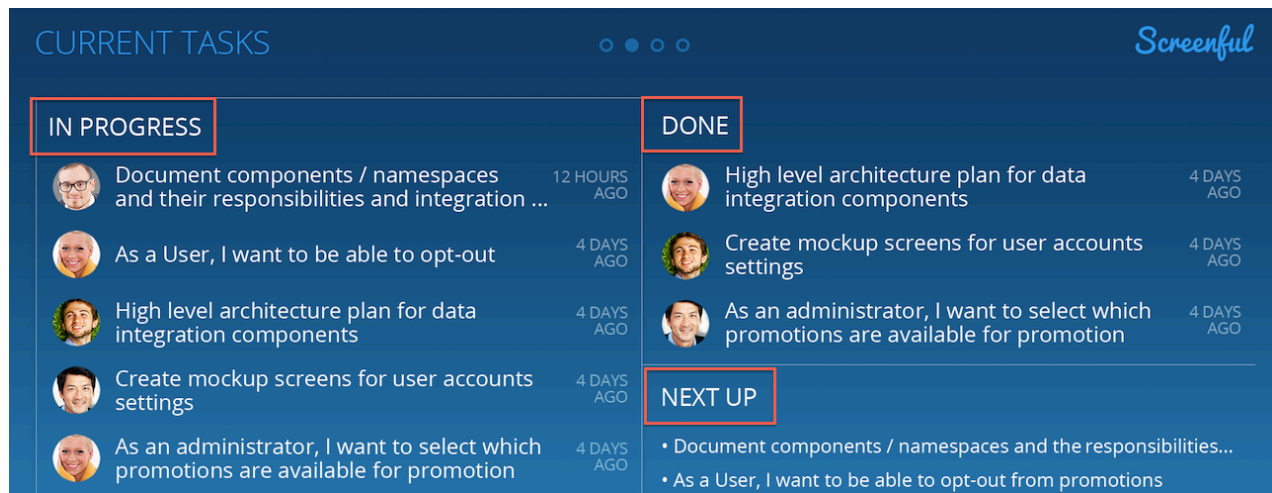
### Disabling Access to your Trello Account

In case you want to disable access from connector, simply click the Avatar on the member list and select *Remove from board*.

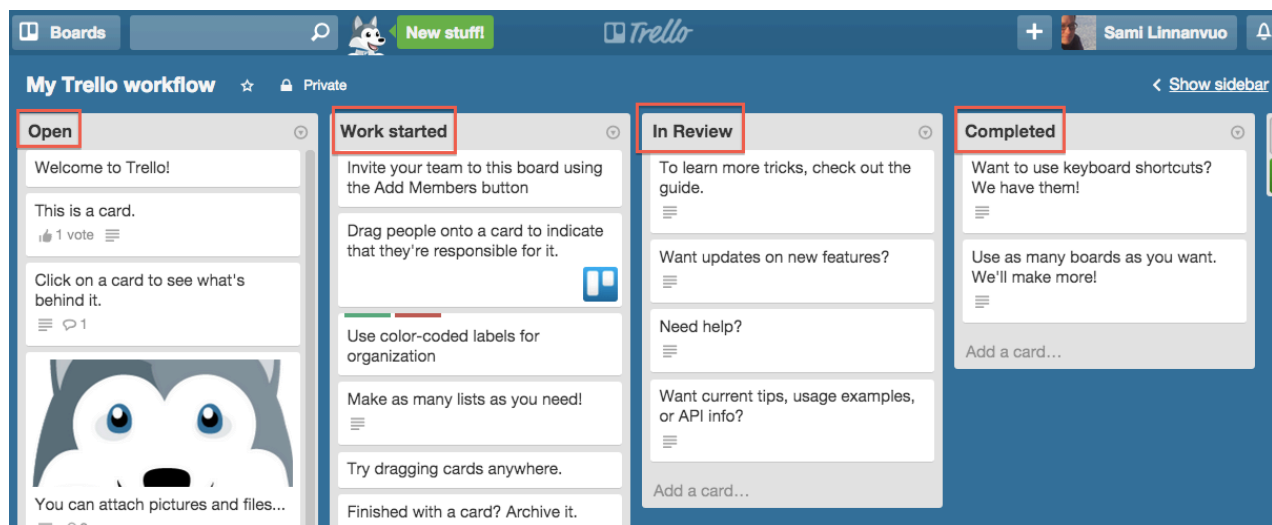


## Mapping Trello columns to Workflow States

Columns in Trello board must mapped with the workflow states represented in the Screenful UI. For example, the *Current Tasks* view has three states: *In Progress*, *Done*, *Next up*:



However, the column names used in Trello may be something entirely different:



In order to determine which columns map to which workflow states, a mapping table is constructed:

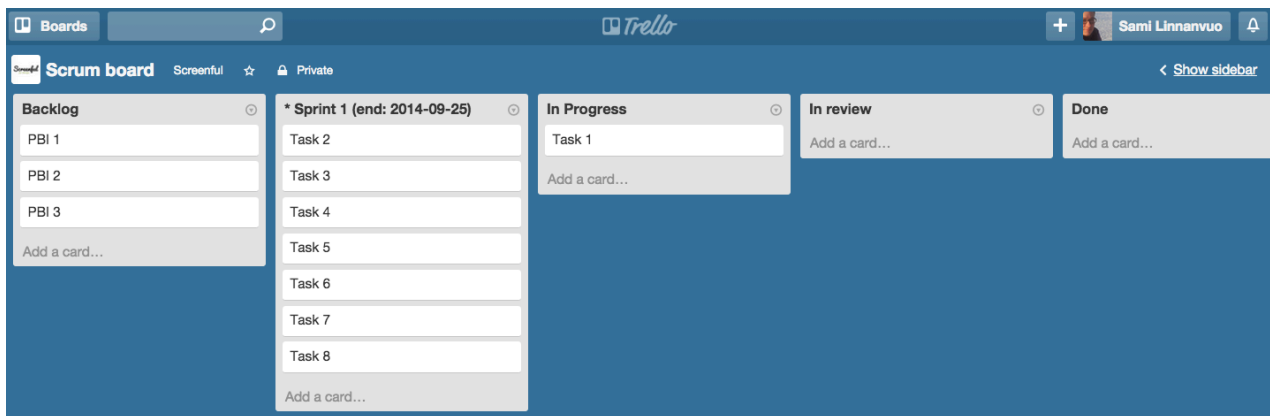
State in Screenful UI	Trello columns
In progress	Work started, In Review
Done	Completed
Next up	Open

**Note!** If you change the column names in your Trello board, the connector process will automatically detect the change and alert Screenful staff to update the mapping table accordingly.

## Setting up a Scrum Board

Technically speaking, Trello does not have built-in support for Scrum process. However, being a flexible tool, you can use its core features to represent Scrum artifacts like product and sprint backlogs, time estimates, and time-boxed sprints if you want to do so.

In order to set up *Connector* to fetch sprint data from Trello, you need to setup the board columns in a certain way so that the *Connector* knows where to look for data. The screenshot below shows an example Scrum board in Trello:



The columns in this board from left-to-right are:

### 1 Backlog (optional)

This is a product backlog and contains all the stories that you may want to implement into your product at some point of future.

### 2 Sprint backlog (mandatory)

Current sprint is the sprint currently in progress. Sprint is set into progress by prefixing the sprint name with a star. In the beginning of the sprint, all the cards are in the sprint backlog (i.e. in the column marked with a star). Cards are progressed to the right during the sprint, and if everything goes as planned, all the cards are in the *Done* column when the sprint ends.

### 3 In progress (mandatory)

There must be least one column to represent the work in progress. Depending on your workflow, there can be multiple in progress columns such as “In review”, “Waiting for deployment” etc.

#### 4 Done (mandatory)

There must be at least one column to represent the work completed. Depending on your workflow, there may be multiple completed columns such as “Done”, “Closed” etc.

**Note!** You can have any number of columns in your board. In the minimum, you must have one column for the cards that are in the current sprint backlog and mark it with star (\*). In addition you need to have at least one *In progress* column and one *Done* column.

**Note!** The names of the columns can be freely chosen and *Connector* can recognize them due to board specific configuration performed by Screenful staff. The only exception is the column representing the current sprint which name must be prefixed by star(\*) and which must be suffixed by the sprint end date.

## Sprint Lifecycle

### Assumptions

- Team may want to end the sprint before its due date
- Team may want to continue the sprint after its due date
- Team may want to estimate tasks using hours or storypoints as a measure, or to not estimate at all.

### 1 Starting a new sprint

A new sprint is started by creating a new backlog column and by adding cards to it. However, before moving cards around the board, the previous sprint must be completed so that the velocity will be recorded correctly. That is achieved by following the steps below (in the numbered order). To start a new sprint

1. Remove asterisk (\*) from the previous sprint backlog
2. Create a new column to act as a new sprint backlog and move cards as needed to the new backlog.
3. Write the sprint end date into the sprint backlog column using a notation (end: yyyy-mm-dd).
4. Archive the old backlog column (from previous sprint)
5. Archive the tasks in the Done column (from previous sprint)
6. Mark the column representing the new sprint backlog with asterisk (\*)

### 2 Completing a sprint

A sprint is considered completed once

- All cards are moved into the *Done* column and sprint end date is passed
- OR

- New sprint is started

## Assigning Estimates for Tasks

A time estimate can be associated with a task by inserting a number in the parenthesis as a prefix to the task name. For example

(6) Task name

Means that the task is estimated to be size of 6. You can record effort already spent on a task by putting it in square brackets, like below

(6) Task name [2]

Now the task has 2/6 units completed, leaving 4 units to be done.

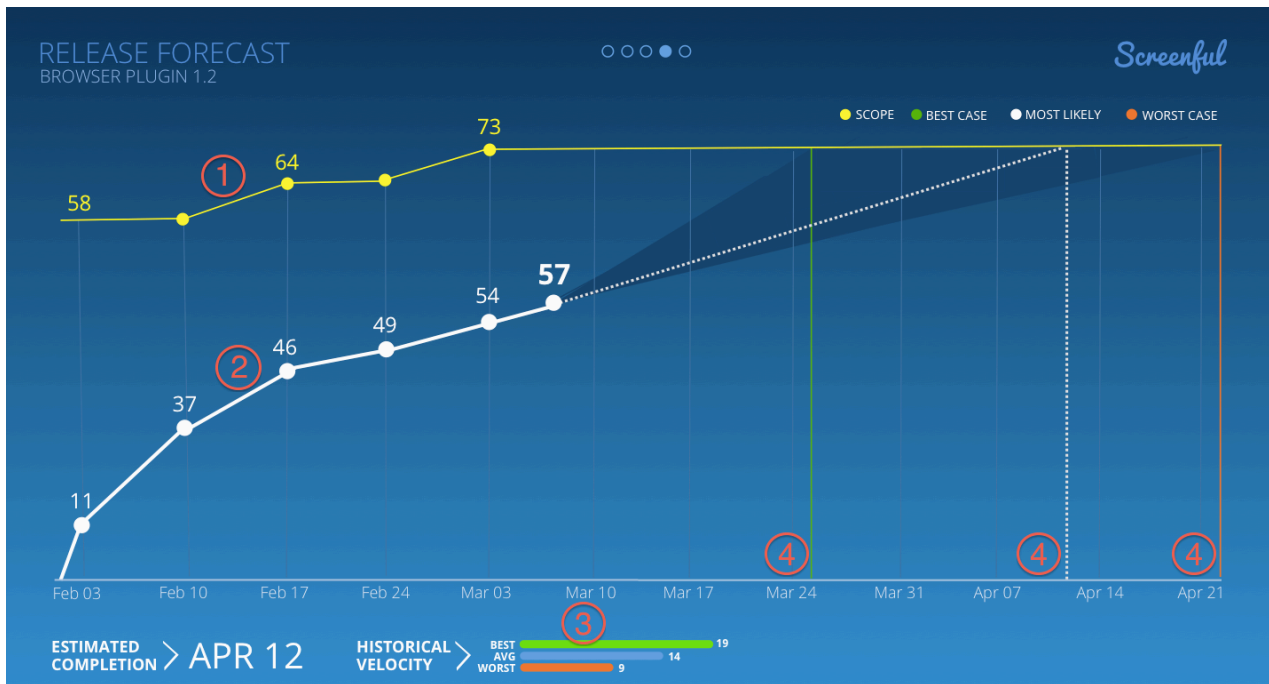
**Note!** You can record effort to a task whenever you want. If you change the time estimate of a task after the sprint is started, it will affect the sprint scope.

## Setting up a Release Forecast

**Note!** While there may be varying definitions, by *release* we mean any chunk of work that you want to treat as a whole. In your vocabulary it might as well be called *milestone*, *version*, or a *project*. The key characteristic of a release is that it has a scope (which may or may not be fixed) and its progress needs to be communicated to the interested parties.

A release forecast helps you to estimate your most likely release date based on your team's historical velocity as well as best- and worst-case scenarios.

Below is a Release forecast based on four weeks of history:



The release forecast char consists of (as numbered in the screenshot)

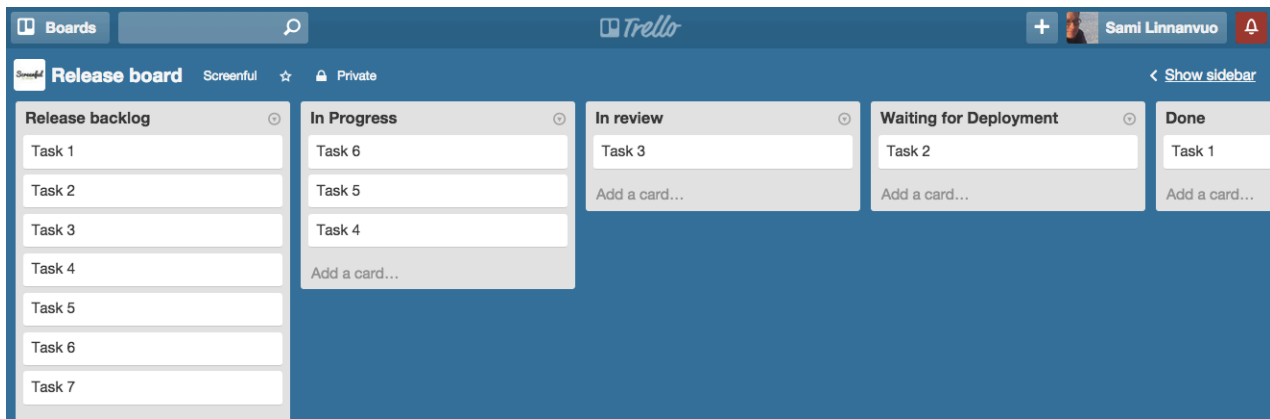
1. A Scope curve representing the total work assigned for the release
2. A Completed curve representing the work done so far
3. Historical velocity of the team
4. Completion scenarios based on the historical velocity

The accuracy of this forecast depends on the stage of your project and the variation of the velocity. In short, the fewer tasks there are left and the more stable your velocity (i.e. small variation) the more reliable the forecast.

In order to get a release forecast for your project, you need to construct your board so that it has

- One or more columns to represent release backlog
- One or more columns to represent work in progress
- One column to represent the work done

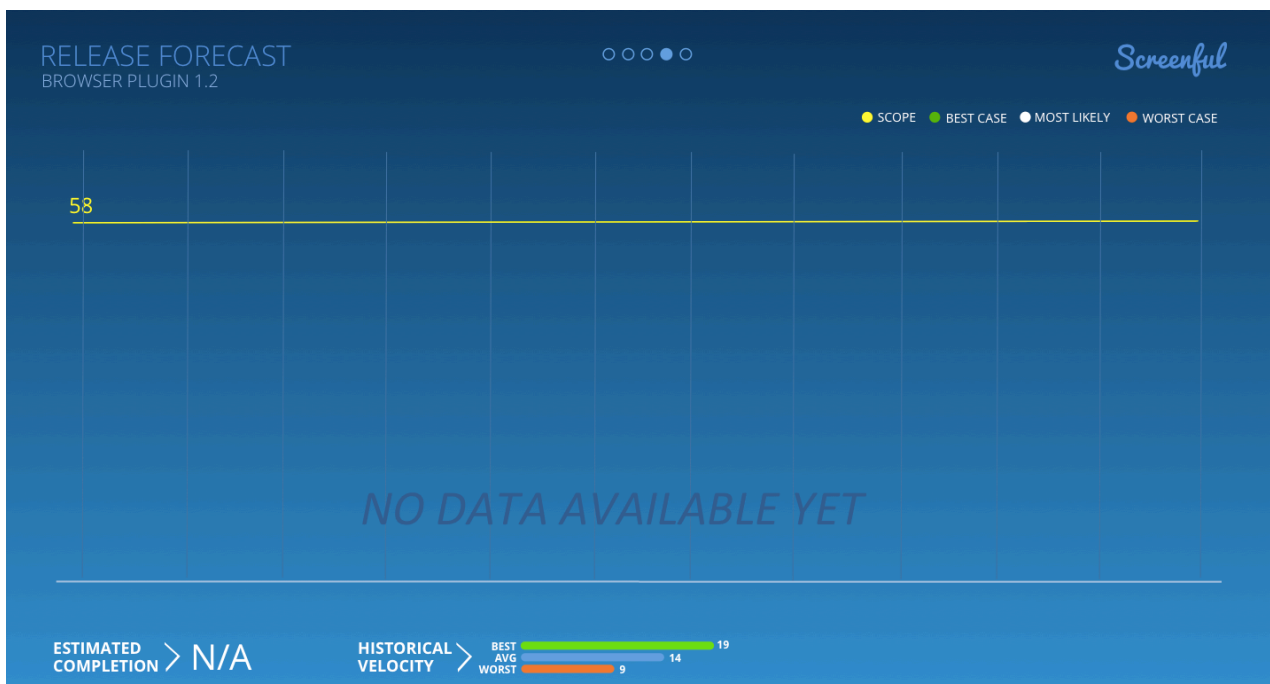
Below is an example of a Trello board representing a release:



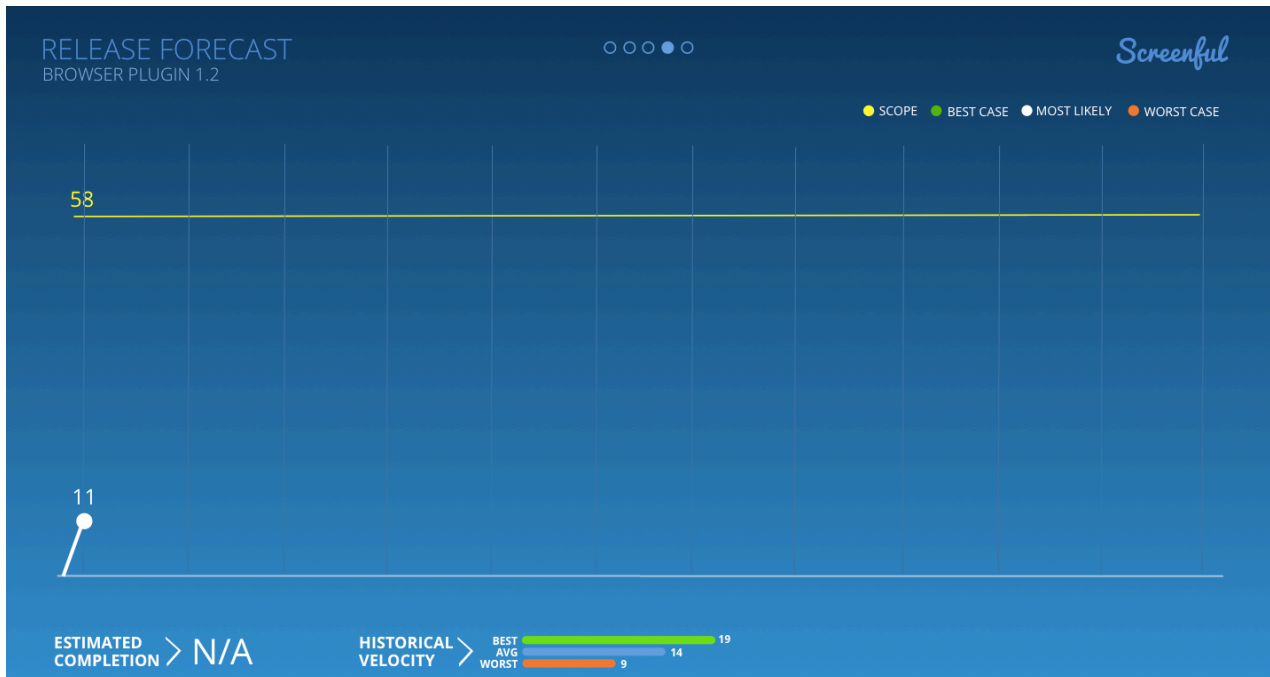
The leftmost column contains the backlog of all items that needs to be developed for the release. When a task is taken under work, it is moved from the backlog to *In Progress* column. There can be several *In Progress* columns depending on your workflow such as *In Review*, *Waiting for Deployment* etc. Finally, the task is moved to *Done* column and at that point it is considered completed.

### Starting a new release

Release history starts from the time when first card was moved to *Done* column. Therefore, you can initiate a new tracking any time by archiving the cards in the done column. After doing so, the chart shows only the scope of the work as a yellow trend line on top along with the text “no data available yet”.



As soon as you move one or more tasks into the *Done* column, the dotted white line representing the work done so far, becomes visible.



**Note!** While scope and work done is immediately available in the chart, a minimum of one week of data is required in order to provide forecast scenarios for the release.

### Completing a release

A release is completed once all the tasks are moved to the *Done* column.