

340CT Software Quality and Process Management

Case Study – Electronic Coursework Submission (ECS) system

1. Introduction

In this lab, you will be introduced to the planning and tracking of a project based on the given case study, Electronic Coursework Submission (ECS), detailed in Section 2.

1.1 Scrum Planning and Tracking

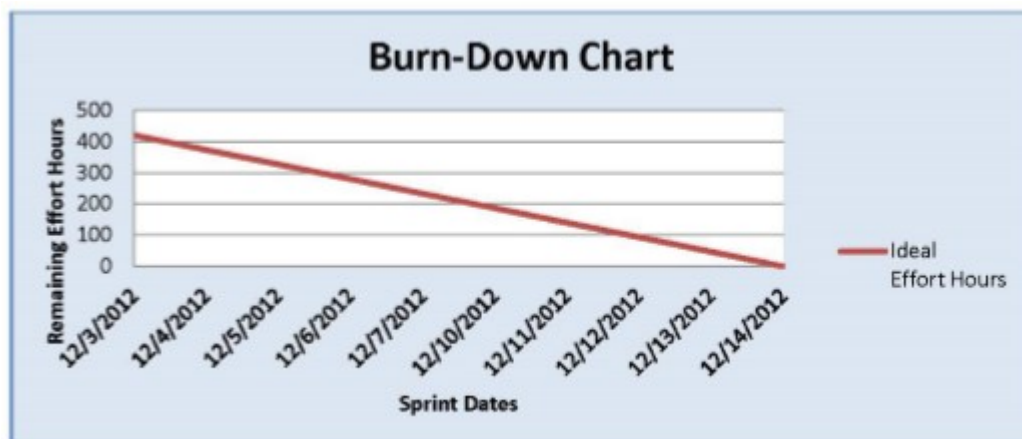
Burn-downs charts provide common sprint tracking mechanisms used by Agile projects. A burn-down chart can be produced using one of the following ways:

- effort remaining
- task count
- story points

This lab firstly focuses on a burn-downs chart based on the approach using effort remaining as follows:

- Step 1: to produce a task breakdown during your sprint planning meeting.
Each task should have associated hours which the team decides on during the planning meeting.
- Step 2: to plot the ideal burn-down chart based on the task breakdown
The ideal burn-down chart reflects progress assuming that all tasks will be completed within the sprint at a uniform rate (as shown in the figure below. A simple burn-down chart can be maintained in a spreadsheet. Dates in the sprint are plotted on the X axis, while remaining efforts are plotted on the Y axis.

The figure below refers to the example for a sprint duration of 2 weeks, a team size of 7 members, 6 working hours per day (assuming 5 working day per week), which amounts to a total of 420 hours.



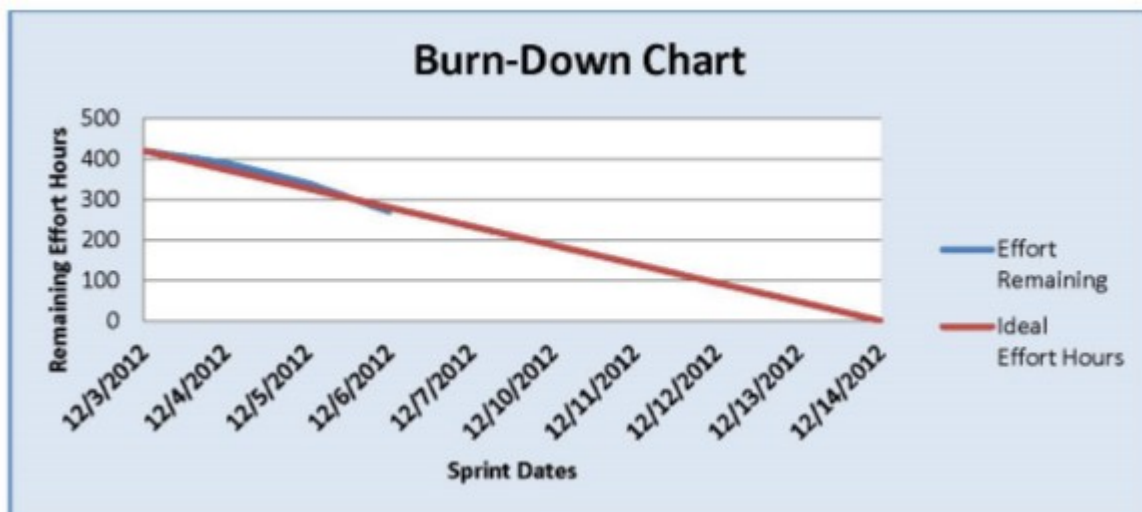
(Source: <https://www.scrumalliance.org/>)

- Step 3: to update the burn-down chart

Each member carries out tasks from the task breakdown and update effort remaining for the tasks, along with its status. For example, if the total estimated effort for Task 1 is 10 hours. After spending six hours on the task, if the team member believes that it requires another six hours to complete, the "Effort Remaining" should be updated as 6.

Story Name	Task No	Description	Status	Owner	Estimated Effort (in Hours)	Estimated Remaining (in Hours)
Story 1	1	Requirement clarification	In progress	Developer 1	10	6
	2	Design	close	Developer 1	8	0
	3	Develop module	open	Developer 1	12	12

The burn-down chart can then be updated as follows:



(Source: <https://www.scrumalliance.org/>)

1.2 Burn-downs charts using task counts

This section focuses on a burn-downs chart using task counts as follows.

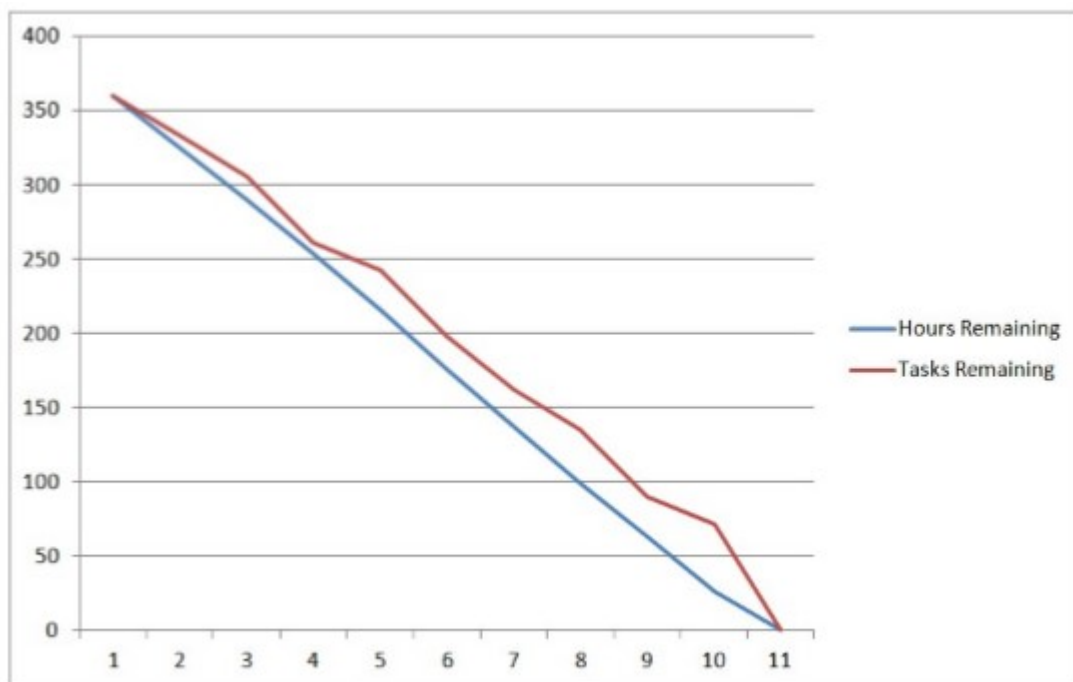
1.2.1 Example: (source: <https://www.scrumalliance.org/>)

Assume that the team members adopt the option to update tasks when a task is completed with the following scenario:

- A 2-week sprint
- 38 tasks
- A 6-person team. Team capacity in hours/day = approx. $6 * 7$ hours /day = 42
- Task estimates created using random numbers between 4 and 16

The burn-down chart is created as follows:

- Line 1: Total Hours Remaining, when task estimates are updated every day.
- Line 2: Number of Tasks Remaining, when the team updates tasks as they complete them.



1.3 Burn-downs charts using story points

This section focuses on a burn-downs chart using story points as follows:

- Step 1: to commits to a finite number of user stories that must be completed during each sprint by the scrum team

Step 2: to generate child tasks for each user story during sprint planning, and time estimation for each child task to be assigned estimated hours

- Step 3: to generate a burn-down charts based on hours after the estimation
(Please note that it is advised not to model the story points in detail but to define them at a higher level. The main function of a burn-down chart is to reflect how the team performs during the sprint. What the team commits to in the sprint are the selected user stories and the total number of committed story points.)

1.1.1 Example: (source: <https://www.scrumalliance.org/>)

Assume that a sprint duration is two weeks (ten workdays). A scrum team has committed to completing three user stories (S) for the total of 16 story points (SP). The team also defined child tasks for each S and split the estimated SPs to their child tasks. The scrum team consists of four team members (TM):

- TMA works six hours per day
- TMB works six hours per day
- TMC can only work three hours per day
- TMD works six hours per day, but TMD will work on the project except for the last two days of the first week.

The following table summarises the above scenario:

US_01 (3 SP)	US_02 (5 SP)	US_03 (8 SP)
CT_US01_01 (3 SP)	CT_US_02_01 (2 SP) CT_US_02_02 (3 SP)	CT_US_03_01 (0.5 SP) CT_US_03_02 (1.5 SP) CT_US_03_03 (2.5 SP) CT_US_03_04 (3.5 SP)

The teams will work for 198 hours in total and complete 16 SPs. Each working hour is equal to 16/198 SP under ideal conditions. The following table summarises the sprint planning by hours and the ideal remainder SPs for the ten working days.

Note: Daily SP (expected SP per day) = Daily Hours * Sprint SP Total [16] / Sprint Working Hours Total [198]
For example, Daily SP (for DAY_01) = 21 * 16 /198 = 1.70

	DAY_01	DAY_02	DAY_03	DAY_04	DAY_05	DAY_06	DAY_07	DAY_08	DAY_09	DAY_10	END
TM_A	6	6	6	6	6	6	6	6	6	6	60
TM_B	6	6	6	6	6	6	6	6	6	6	60
TM_C	3	3	3	3	3	3	3	3	3	3	30
TM_D	6	6	6	0	0	6	6	6	6	6	48
TOTAL	21	21	21	15	15	21	21	21	21	21	198
DAILY	1,70	1,70	1,70	1,21	1,21	1,70	1,70	1,70	1,70	1,70	16,00
BEGINNING	16,00	14,30	12,61	10,91	9,70	8,48	6,79	5,09	3,39	1,70	0,00

The following burn-down chart is generated for the above sprint plan.



2. Case Study – ECS

A local college is planning to have an Electronic Coursework Submission (ECS) system designed to automate its assessment handling process. Your team is assigned to develop and manage this project for the college.

The college Assessment Office has provided the following requirements for the ECS system.

The staff of each module should be able to enter the details for a piece of coursework into the system as below:

- module code
- module title
- module tutor responsible for the coursework
- coursework no.
- coursework title
- issue date
- due date and time
- assessment type: individual or group
- % of module mark

The system would generate a cover sheet which contains the above information about the coursework. The students would then be able to download the cover sheet from the system and enter their details (show below) ready for submission:

- student id
- student name

Once a student has upload the work with the cover sheet on the system through a submission link, the system would generate a unique reference code to confirm the submission of the work:

Staff and students should be able to log onto the ECS system to view the status of a coursework submission, which includes: 'awaiting submission', 'submitted', 'absence'. The system should also be able to generate reports of submissions on request by module leaders.

3. Lab Activities

In this lab, your team should use the guide given in Section 1 to produce the following:

- A draft project plan with a task breakdown for your project in sprints
- A burn-down chart using effort remaining or story points to monitor your project

Please note that you are not required to submit the above lab activities for marking, but they would be useful for the coursework assessment.