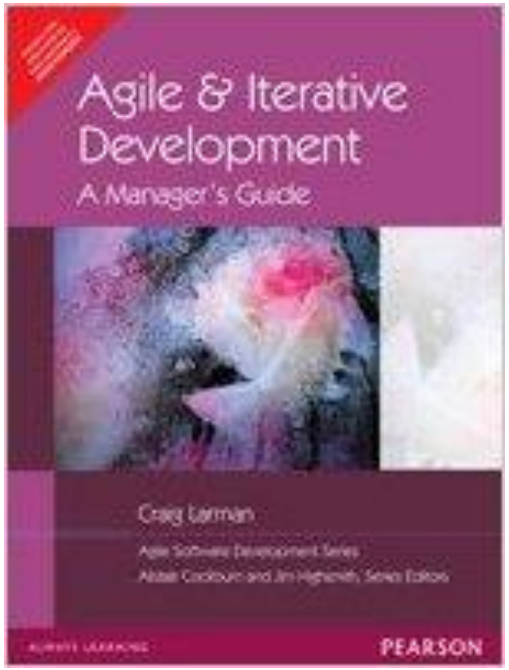
BITS Pilani
Pilani | Dubai | Goa | Hyderabad

Iteration Planning in Agile

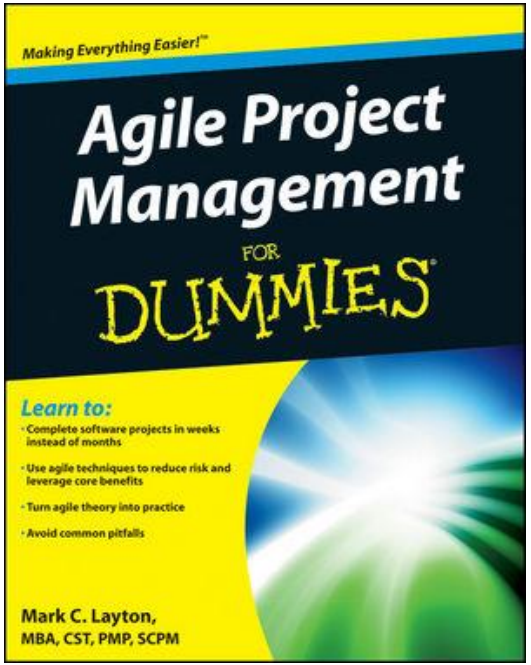
- Prof K G Krishna

Text/Reference Books

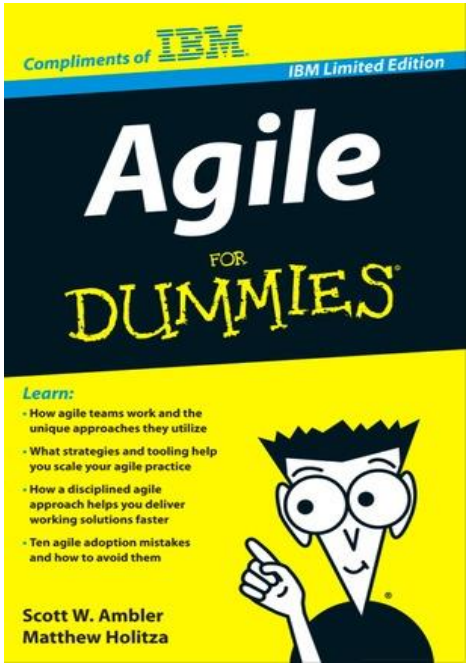
T1



T2



Compliments of IBM



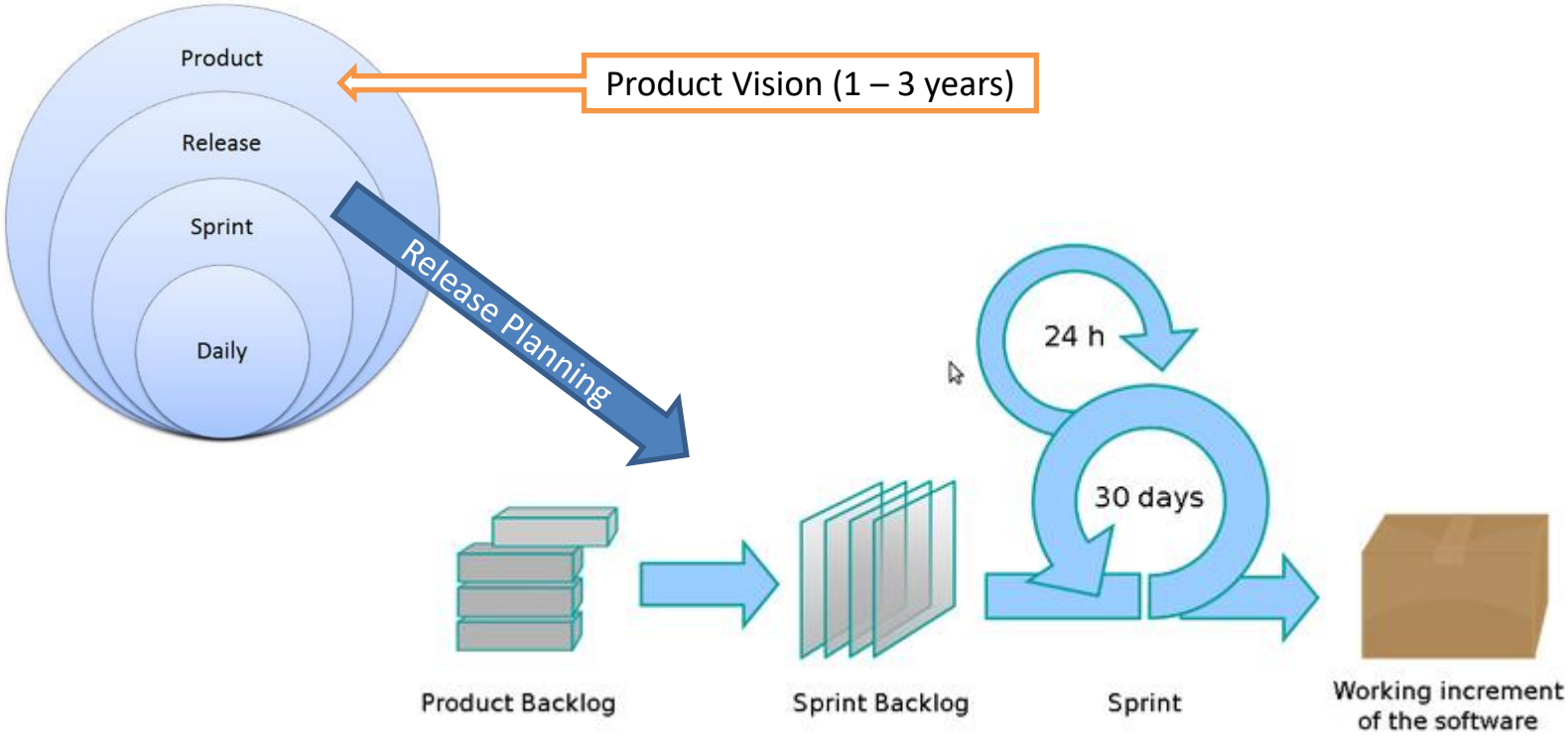
➔ As this field is evolutionary, the student is advised to stay tuned to the current and emerging practices by referring to their own organization's documentation as well as Net sources

Topics

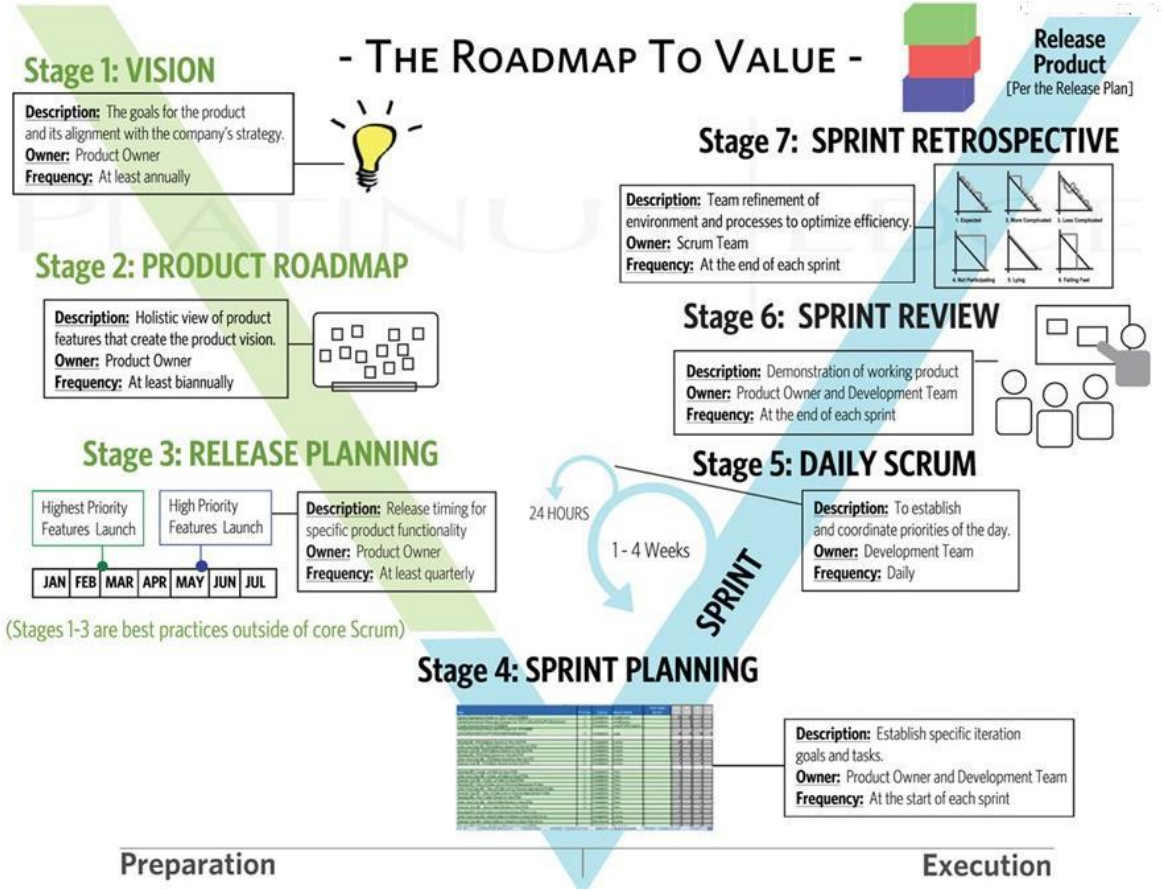
Iteration Planning in Agile

- Sprint as an Iteration
- Velocity based Planning
- Capacity based Planning
- Sprint Planning/Backlog

Levels of Planning in Agile

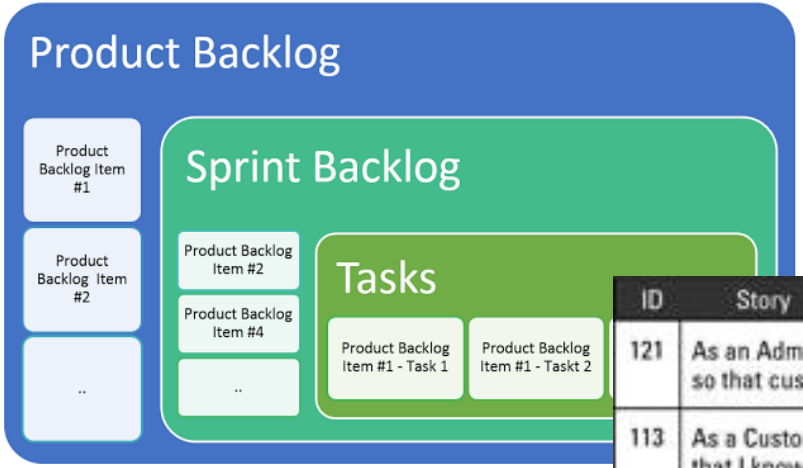


Planning at *Every* Stage in Agile (SCRUM)



Source: (T2)

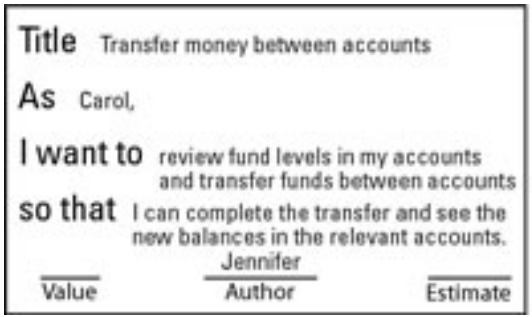
Product-backlog vs. Sprint-backlog vs. Tasks/User Stories

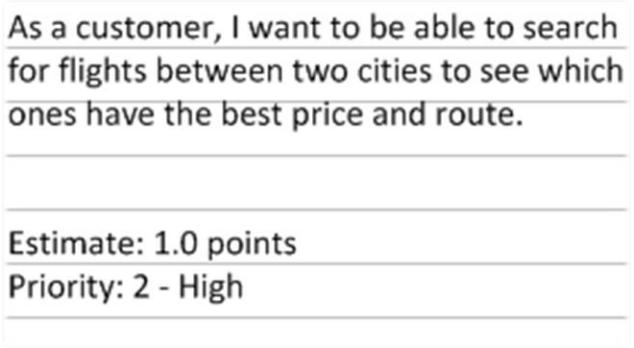


ID	Story	Type	Status	Value
121	As an Administrator, I want to link accounts to profiles, so that customers can access new accounts.	Feature	Not started	5
113	As a Customer, I want to view my account balances, so that I know how much money is currently in each account.	Feature	Not started	3
403	As a Customer, I want to transfer money between my active accounts, so that I can adjust each account's balance.	Feature	Not started	1
97	As a Site Visitor, I want to contact the bank, so that I can ask questions and raise issues.	Feature	Not started	2
68	As a Site Visitor, I want to find locations, so that I can use bank services.	Feature	Not started	8

User Stories = Requirements for the Developer/Tester

1. Identify the project stakeholders.
2. Identify who will use the product.
3. Working with the stakeholders, write down the requirements that the product will need in a story format (User Story Cards)





➔ The best changes often come at the end of the project, when you know the most about your product and its customers

Source: (T2-Chap8)

Sprint Planning (work backwards from Goal)

1. Discuss and set a Sprint Goal.
2. Review the User Stories from the Product-backlog that support the Sprint Goal and revisit their relative estimates
3. Determine what the team can commit (Stories) to in the current Sprint.
4. Create Task-list for each Committed Story
5. Always Plan One Sprint at a Time (Just-in-Time Planning)

Example: Sprint Goal

"As a mobile banking customer, I want to log in to my account so that I can view my account balances and pending and prior transactions."

Sprint Stories (for the above Goal)

- Log in and access my accounts.
- View account balances.
- View pending transactions.
- View prior transactions

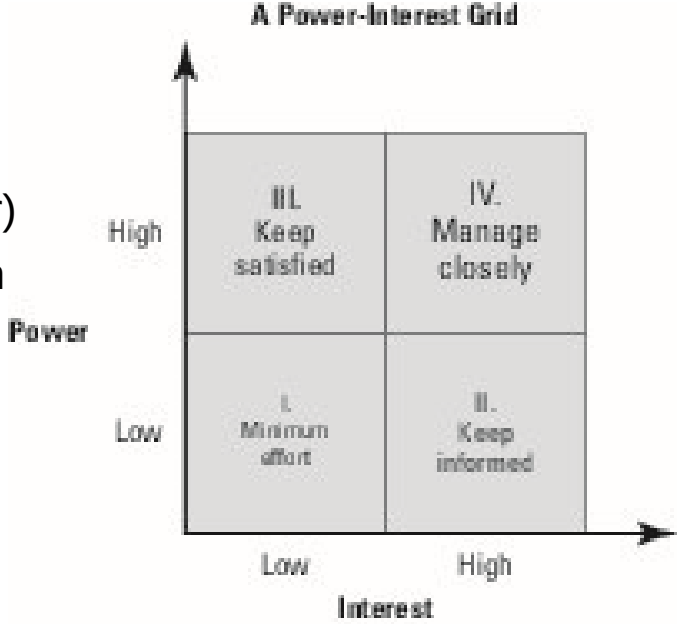
Tasks (for the above Stories)

- ✓ Create an authentication screen for a username and password, with a Submit button.
- ✓ Create an error screen for the user to reenter credentials.
- ✓ Create a logged-in screen (includes list of accounts — to be completed in next user story).
- ✓ Using authentication code from the online banking application, rewrite code for an iPhone/iPad application.
- ✓ Create calls to the database to verify the username and password. Refactor code for mobile devices.

Source: (T2-Chap8)

Who are the *Stakeholders* to Contribute User Stories?

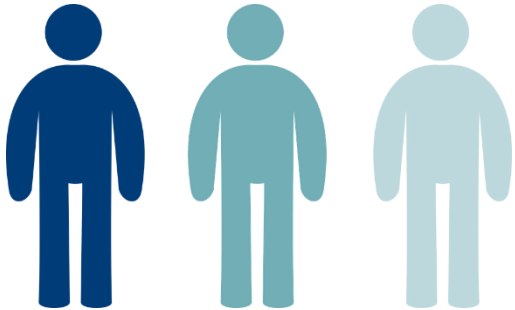
- Who Interact with End-users / Customers on regular basis (Customer Service, Marketing, Sales,…)
- Business Domain Experts
- Actual End-users
- Technical Experts (who developed such Products earlier)
- Technical Experts (when the Product need interface with other systems)
- …



Source: (T2-Chap8)

Define *Personas* (Classes of End-users)

- Persona = Representative of End-user Group (with an identified set of similar profile, demographics, behaviours and attitudes)
- Business Domain Experts help identify Personas
- Identify Patterns-of-use through Field Study (Ethnography)
- Personas are key part of Product Vision Statement
- Ensure Manageable Personas (not too many, not too few)
- Not only existing Customers, but also Potential Customers



Source: (T2-Chap8)

Velocity / Capacity Planning ➔➔

Creating Tasks for Sprint-backlog

1. Create the sprint backlog tasks associated with each user story. Break the user stories into discrete individual tasks (can be completed within few hours or less than a day) and allocate a number of hours to each task. Make sure that the tasks encompass each part of the definition of *done*: developed, integrated, tested, and documented.
2. Ensure no over-committing occurs at the beginning of a sprint, especially in the project's first few sprints. The development team should target being able to complete a task in a day or less, for a couple reasons: Short-term goals promote productivity; and a short timeframe brings problems to the forefront quickly. If a team member is working on a task for more than a day or two, that task or that team member may need special attention. If the tasks exceed the hours available, seek the project owner's advice on which user stories to remove from this sprint.
3. Each member chooses a first task to accomplish. Development team members should work on only one task on one user story at a time to enable *swarming* — the practice of having the whole development team work on one requirement until it's completed. Swarming can be a very efficient way to complete work in a short amount of time.

Source: (T2-Chap8)

Velocity Planning (of Sprint)

- Scrum teams use velocity to plan how much work they can take on in a release and sprint. Velocity is the sum of all user story points completed within a sprint. So, if a scrum team completed six user stories during its first sprint with sizes 8, 5, 5, 3, 2, 1, their velocity for the first sprint is 24. The scrum team would plan its second sprint keeping in mind that it completed 24 story points during the first sprint.
- After multiple sprints, scrum teams can use their running average velocity as an input to determine how much work they can take on in a sprint, as well as to extrapolate their release schedule by dividing the total number of story points in the release by their average velocity.

Source: (T2-Chap8)

Capacity Planning in Scrum

- In Agile, the Team (not Individual) is the persistent Group (Unit) which has a combination of skills required to do the project
- **Velocity = Amount of work (No. of Story Points) a Team can do in one Sprint**
- **Capacity Planning = Estimating/Calculating the Capacity of Agile Team**
- Units of Measurement: Story Points, Person-Hours (recommended)
- Number of workdays in the period (at five days per week)
- Factors to consider calculating Capacity (as FTE):
  - Number of Team members; Known meetings or activities which involve the whole team, during which no one can contribute hands-on work; Planned time off for each person in the period; Fractional availability of each person for work when not in known meetings
- State Assumptions clearly (Avg complexity of Work, Specialized Task if any)
- Capacity Planning (Team-oriented Estimation) is effective so long as there is no dominance of Specialized Tasks

Ref: https://www.cprime.com

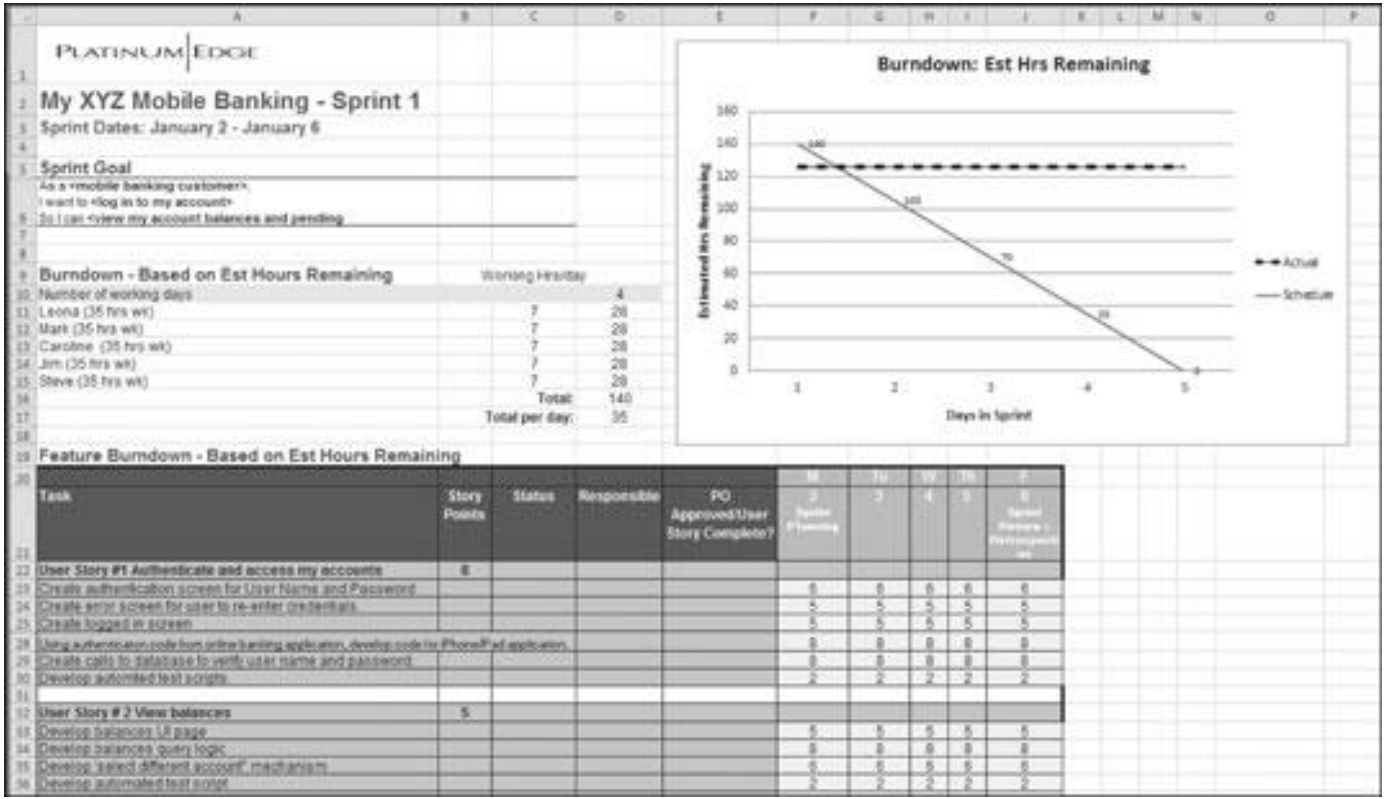
Capacity Planning (Be *Realistic*!)

Formula = (Total Sprint Hours - Known Spent Hours) * Effective Hours on Task = capacity

Where:

Total Sprint Hours =	Sprint duration (weeks) * Work hours per week
Known Spent Hours =	Hours needed fpr the ceremonies, vacation, statutory holidays, planned training days, recurrent meetings or other things
Effective hours on task =	Compensation factor for actual effective hours in a day (factors in e-mail, coffee or bathroom breaks, impromptu hallway banter)

Sprint-backlog can be an Excel Sheet



PLATINUM EDGE

My XYZ Mobile Banking - Sprint 1

Sprint Dates: January 2 - January 6

Sprint Goal

As a <mobile banking customer>,
I want to <log in to my account>
So I can <view my account balances and pending

Burndown - Based on Est Hours Remaining

	Working Hrs/day	
Number of working days		4
Leona (35 hrs wk)	7	28
Mark (35 hrs wk)	7	28
Caroline (35 hrs wk)	7	28
Jim (35 hrs wk)	7	28
Steve (35 hrs wk)	7	28
	Total:	140
	Total per day:	35

Burndown: Est Hrs Remaining

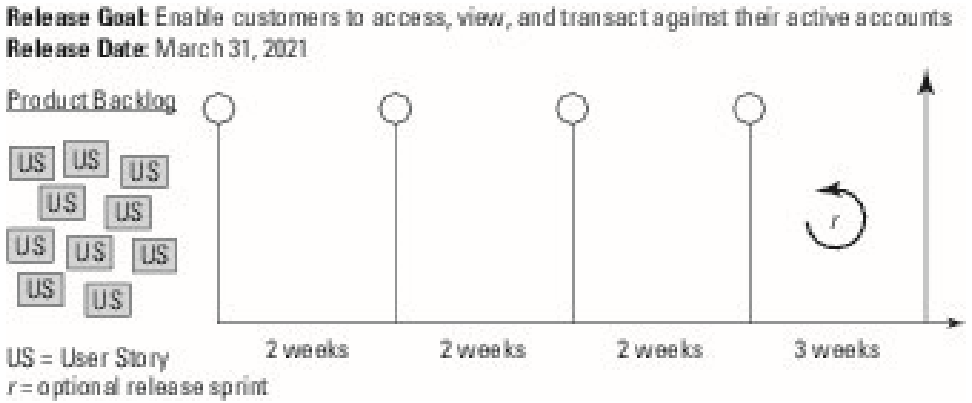
Feature Burndown - Based on Est Hours Remaining

Task	Story Points	Status	Responsible	PO Approved/User Story Complete?	1 Sprint Planning	2	3	4	5 Sprint Review / Retrospective
User Story #1 Authenticate and access my accounts	8								
Create authentication screen for User Name and Password					6	6	6	6	6
Create error screen for user to re-enter credentials					5	5	5	5	5
Create logged in screen					5	5	5	5	5
Using authentication code from online banking application, develop code for iPhone/iPad application					8	8	8	8	8
Create calls to database to verify user name and password					8	8	8	8	8
Develop automated test scripts					2	2	2	2	2
User Story # 2 View balances	5								
Develop balances UI page					5	5	5	5	5
Develop balances query logic					8	8	8	8	8
Develop 'select different account' mechanism					5	5	5	5	5
Develop automated test script					2	2	2	2	2

Source: (T2-Chap8)

Release Sprint Planning ➔➔

Release Planning



Source: (T2-Chap8)

No Separate ‘Release-backlog’ Recommended

- Not all agile projects use release planning. Some scrum teams release functionality for customer use with every sprint, or even every day. The development team, product, organization, customers, stakeholders, and the project’s technological complexity can all help determine your approach to product releases.
- Don’t create a new, separate backlog during release planning. The task is unnecessary and reduces the product owner’s flexibility. Prioritizing the existing product backlog based on the release goal is sufficient and enables the product owner to have the latest information when he or she commits to the scope during sprint planning.

➔ The product backlog and release plan are some of the most important communication channels between the product owner and the development team.

Source: (T2-Chap8)

Release Sprint? Consider Tasks Too *Heavy* for one Regular Sprint…

- Some tasks, such as security testing or load testing a software project, can't be completed within a sprint, because the security or load testing environments take time to set up and request. Although release sprints allow scrum teams to plan for these types of activities, doing so is an anti-pattern, or the opposite of being agile. Your goal should be to complete all work required for functionality to be shippable at the end of each sprint.
- Some project teams add a release sprint to some releases to conduct activities that are unrelated to product development but necessary to release the product to customers. If you need a release sprint, be sure to factor that into the date you choose.

Source: (T2-Chap8)

Release Plan, Release Schedule

- The Product-backlog and Release Plan are some of the most important communication channels between the product owner and the development team
- The Release Plan contains a Release Schedule for a specific set of Features. The Product Owner creates a Release Plan at the start of each Release. Creating a release plan involves:

1. Establish the release goal. The release goal is an overall business goal for the product features in your release.
2. Identify a target release date. Some scrum teams determine release dates based on the completion of functionality; others may have hard dates, such as March 31 or September 1.
3. Review the product backlog and the product roadmap to determine the highest-priority user stories that support your release goal (the minimum marketable features). These user stories will make up your first release.
4. Refine the user stories in your release goal. During release planning, dependencies, gaps, or new details are often identified that affect estimates and prioritization. This is the time to make sure the portion of the product-backlog supporting your release is sized appropriately. The development team helps the product owner by updating estimates for any added or revised user stories, and commits to the release goal and scope with the product owner.
5. Estimate the number of sprints needed, based on the scrum team's velocity.
6. Identify work necessary to release that can't be completed within a sprint. Plan a release sprint, if necessary, and determine how long it should be.

Source: (T2-Chap8)

Just-in-Time Planning of Releases

- Use Just-in-Time Planning, Do not get into microscopic detailing early: commit to the plan for the first release, but anything beyond the first release is tentative (subject to change).
- It's a good idea to achieve releases with about 80 percent of the user stories, using the final 20 percent to add robust features that will meet the release goal (to add to "wow" factor)

Source: (T2-Chap8)

Summary: Iteration Planning

- Sprint is a fixed-duration (~30 days) Iteration of Development Activity in the SCRUM Process
- Planning starts with Product Visioning ➔ Product-backlog ➔ Sprint-backlog
- Sprint Planning to involve all Stakeholders
- Identify Personas for Requirements Analysis (creation of Product/Sprint-backlog)
- Story is the Unit of Requirements in Sprint
- Releases (to Customer) can be planned as part of Sprint planning (Development Sprint vs. Release Sprint); no separate Release-backlog may be required
- Estimation/Capacity is Team-centric (not Individual) – Velocity/Capacity Planning

Thank You