MIAMI UNIVERSITY

PROJECT PLANNING

Content from Chapter 3 of "Head First Software Development", Pilone et al.

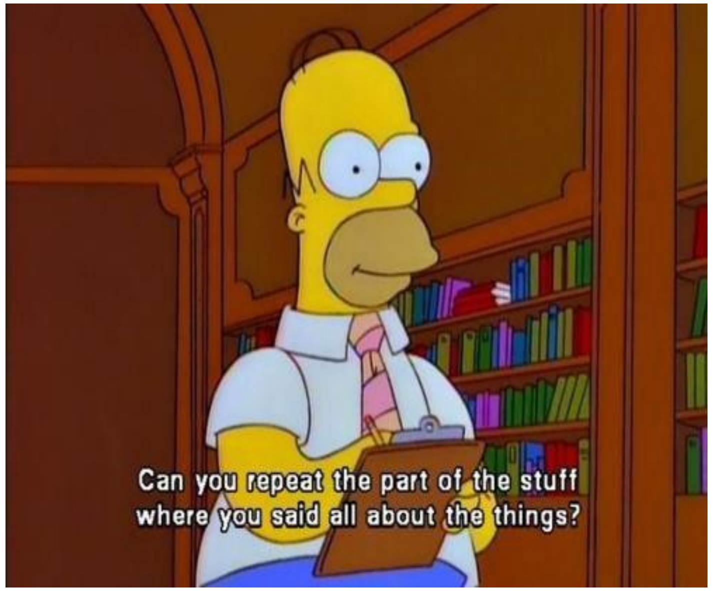
Miami University Software Technology & Analysis Group (MUSTANG)
Computer Science & Software Engineering
Miami University, Oxford, Ohio, USA

Admin

- Read Chapter 3 (Project Planning)
- Read ahead – Chapter 4 (User Stories & Tasks)

AGENDA

- Review
- Fitting User Stories
- MVP/Milestone 1
  - Sanity Check
  - Adjust MVP
  - Exercise
- Velocity
- Big Board

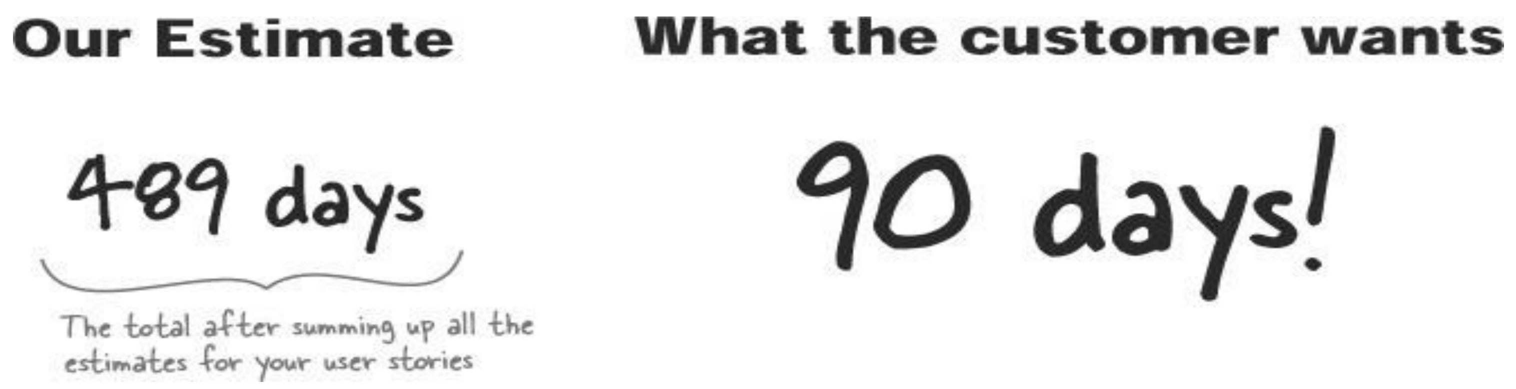

Can you repeat the part of the stuff
where you said all about the things?

Project Planning

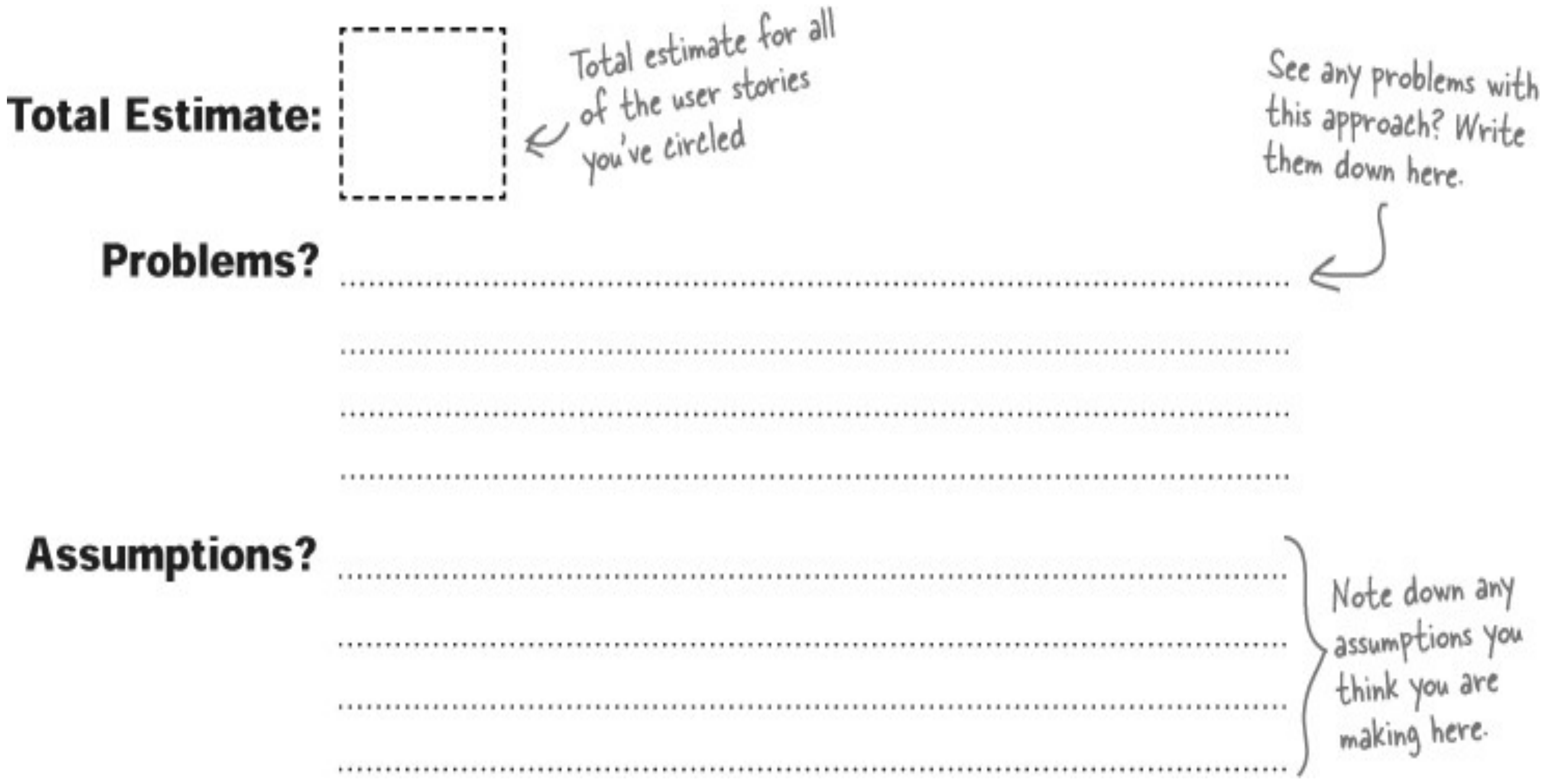
- Prioritizing Requirements
- Defining Appropriate Iterations
- Development Plan

I'll Take My Software Now, Thanks

- Bluesky/Brainstorming yielded lots of ideas and user stories
- Of course, doing everything => Infeasible!
- Likely the customer has a date they want a 'finished' product by.


Our Estimate
What the customer wants
489 days
The total after summing up all the estimates for your user stories
90 days!

Option 1: Fit Stories To Customer's Timeline



Total Estimate:

Total estimate for all of the user stories you've circled

See any problems with this approach? Write them down here.

Problems? ....

....

....

....

Assumptions? ....

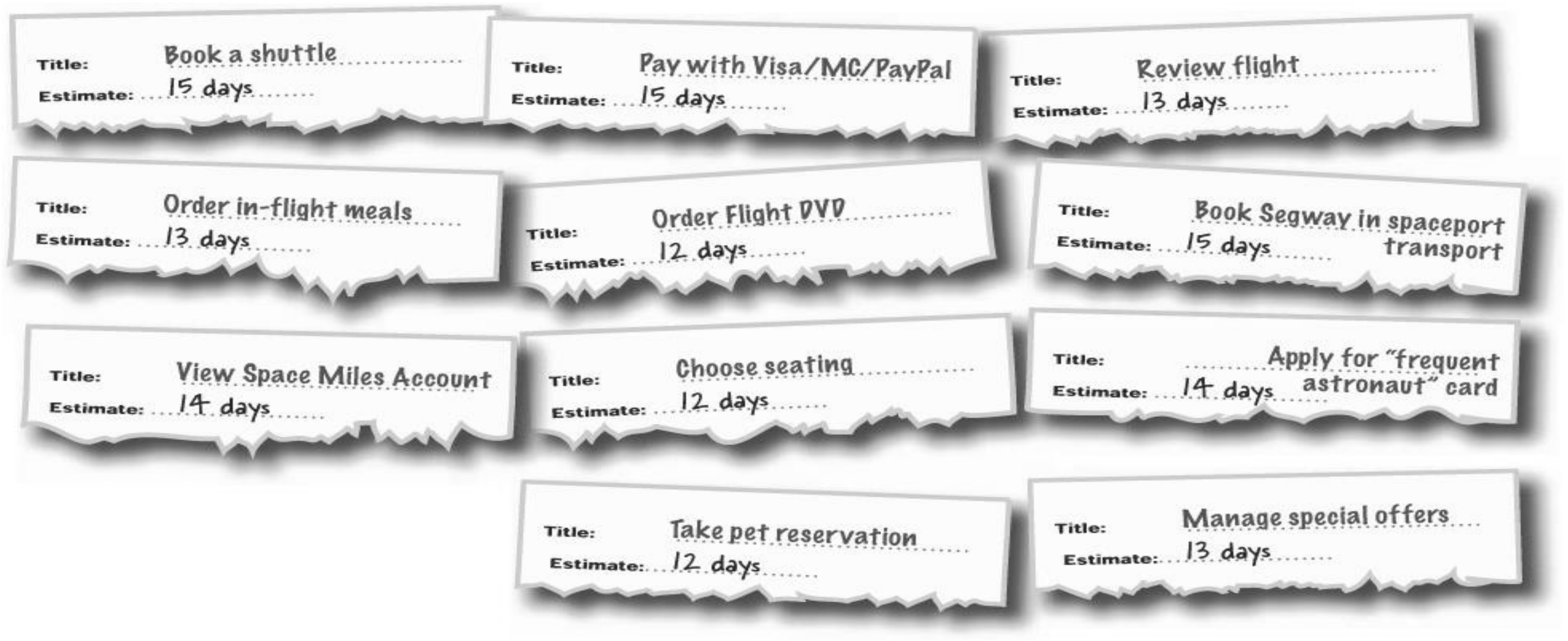
....

....

....

Note down any assumptions you think you are making here.

LET'S TRY THAT – PICK SOME STORIES



Title: Book a shuttle
Estimate: 15 days

Title: Pay with Visa/MC/PayPal
Estimate: 15 days

Title: Review flight
Estimate: 13 days

Title: Order in-flight meals
Estimate: 13 days

Title: Order Flight DVD
Estimate: 12 days

Title: Book Segway in spaceport transport
Estimate: 15 days

Title: View Space Miles Account
Estimate: 14 days

Title: Choose seating
Estimate: 12 days

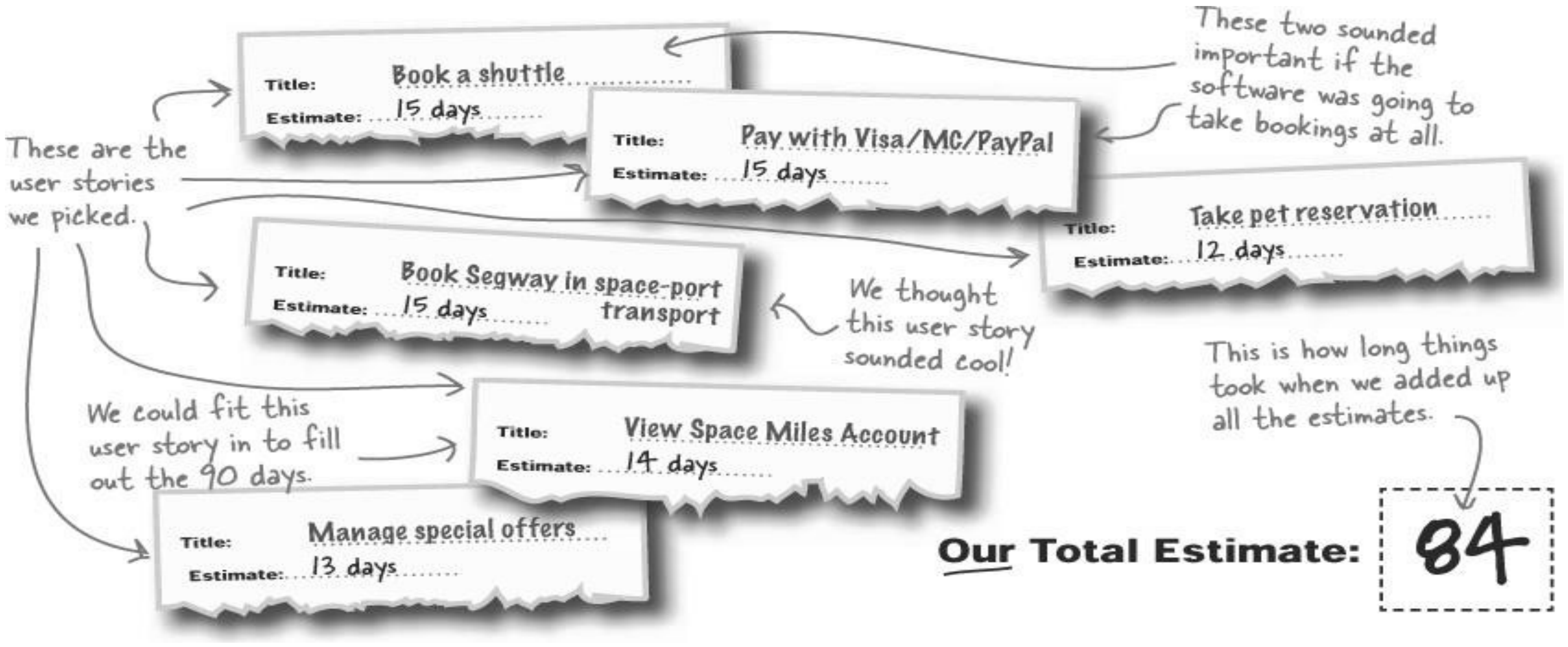
Title: Apply for "frequent astronaut" card
Estimate: 14 days

Title: Take pet reservation
Estimate: 12 days

Title: Manage special offers
Estimate: 13 days

MIAMI UNIVERSITY

Simple Solution


These are the user stories we picked.
Title: Book a shuttle
Estimate: 15 days
Title: Pay with Visa/MC/PayPal
Estimate: 15 days
These two sounded important if the software was going to take bookings at all.
Title: Take pet reservation
Estimate: 12 days
Title: Book Segway in space-port transport
Estimate: 15 days
We thought this user story sounded cool!
Title: View Space Miles Account
Estimate: 14 days
We could fit this user story in to fill out the 90 days.
Title: Manage special offers
Estimate: 13 days
This is how long things took when we added up all the estimates.
Our Total Estimate: 84

MIAMI UNIVERSITY

Pick Stories That "FIT"

There, I told you we could all fit...

What Went Wrong?

- Two problems with this:
  - ?
  - ?

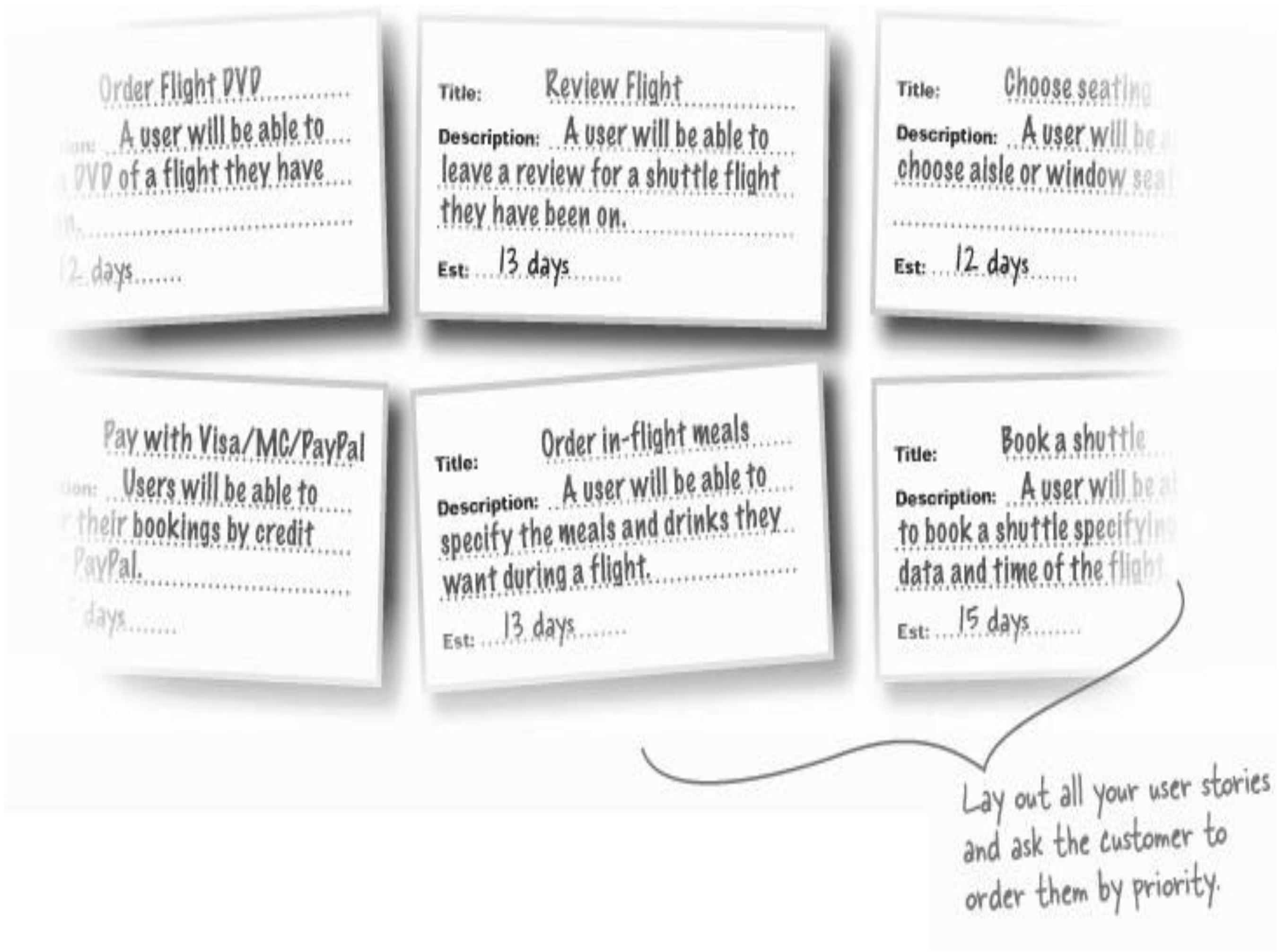
SHOW OUR PLAN TO THE CUSTOMER AND ..

- We ignored the customer completely
- No priority!
  - Customer-focused
- They choose what's needed
  - What's in and out
- You can provide feedback
  - But it's THEIR choice.

We showed the stories we chose to the customer.
But that's not what I wanted at all!

LAY IT ALL OUT

- Lay them ALL out on a table
  - Have customer order them.



Order Flight DVD
ion: A user will be able to
DVD of a flight they have
2 days

Title: Review Flight
Description: A user will be able to leave a review for a shuttle flight they have been on.
Est: 13 days

Title: Choose seatin
Description: A user will be
choose aisle or window sea
Est: 12 days

Pay with Visa/MC/PayPal
ion: Users will be able to
their bookings by credit
PayPal.
days

Title: Order in-flight meals
Description: A user will be able to specify the meals and drinks they want during a flight.
Est: 13 days

Title: Book a shuttle
Description: A user will be a
to book a shuttle specifyin
data and time of the fligh
Est: 15 days

Lay out all your user stories and ask the customer to order them by priority.

PRIORITIZE WITH THE CUSTOMER

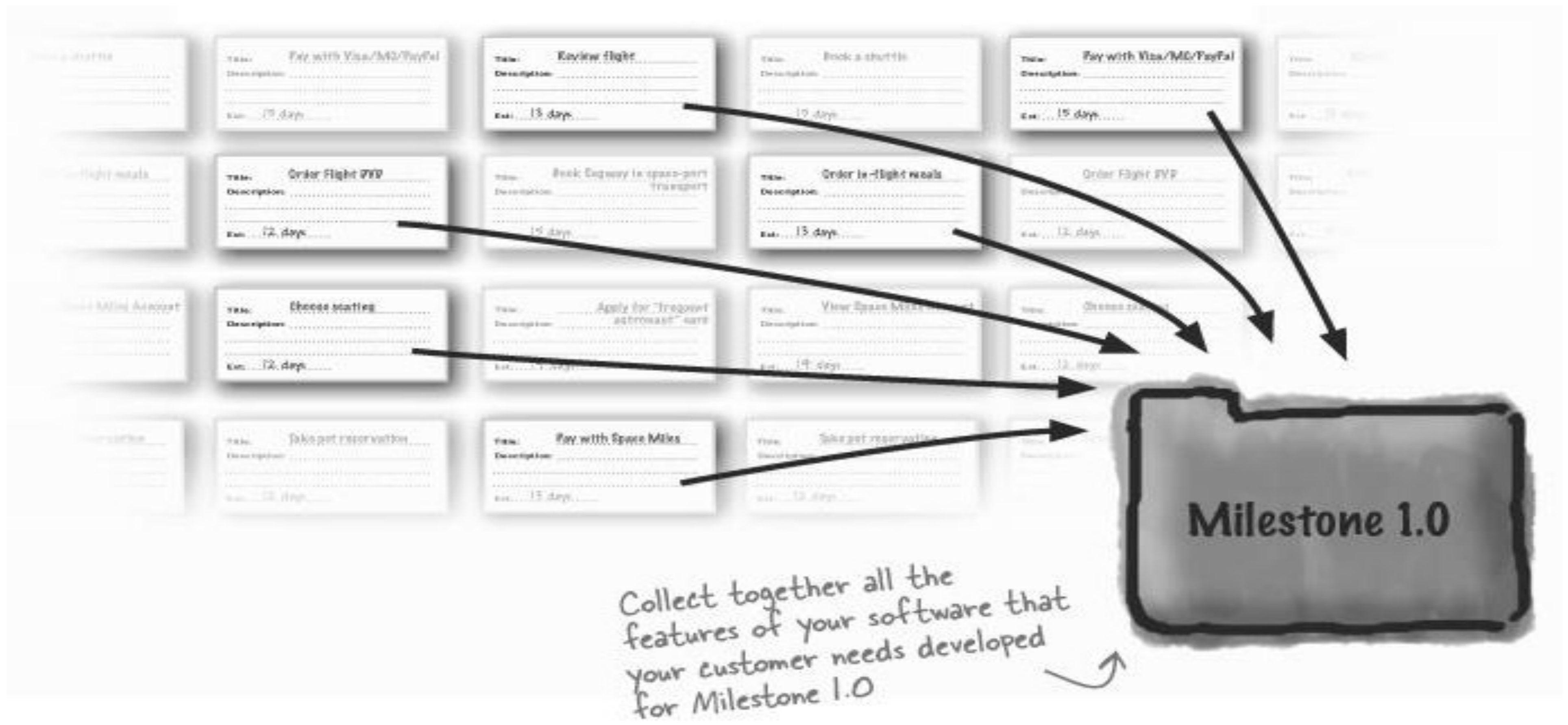
- Have the customer SORT user stories by importance
- Indicate "required" features for MVP, or "Milestone 1.0" as referred to in book
- What is MVP?

TO THE PERSON WHO PAYS FOR THE NETFLIX ACCOUNT THAT EVERYONE ELSE USES
NBA
YOU THE REAL MVP

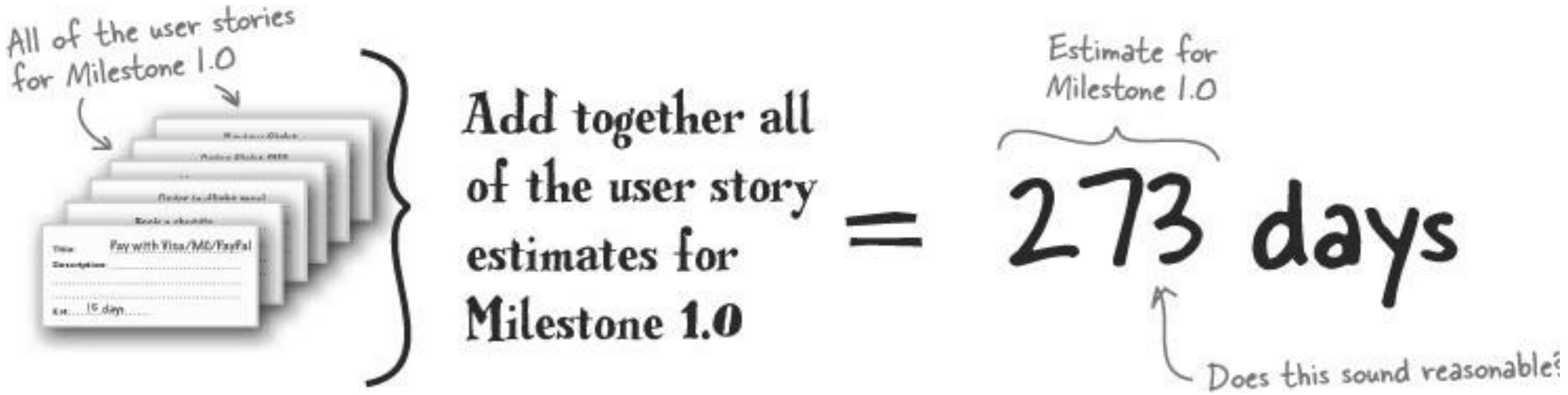
MVP/Milestone 1.0

- Minimum Viable Product
  - MVP is the smallest product that you can build to demonstrate the power of your idea. This should be usable product.
- First major release
  - Includes many small iterations where we will show the customer results for feedback
- There are three reasons for building an MVP:
  - Identify the minimum feature set that can demonstrate the power of the idea
  - Show it around to see what responses it evokes (early customer feedback)
  - Figure out the next iteration from the feedback

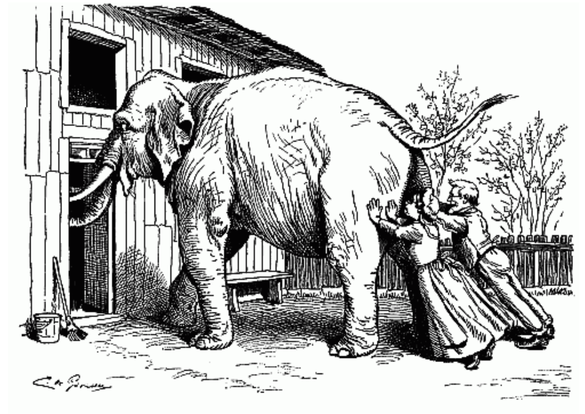
Milestone 1.0


Title: Review flight
Est: 13 days
Title: Pay with Visa/MC/PayPal
Est: 15 days
Title: Order Flight DVD
Est: 12 days
Title: Order in-flight meals
Est: 13 days
Title: Choose seating
Est: 12 days
Title: Pay with Space Miles
Est: 15 days
Milestone 1.0
Collect together all the features of your software that your customer needs developed for Milestone 1.0

MILESTONE 1.0 – SANITY CHECK


All of the user stories for Milestone 1.0
Pay with Visa/MC/PayPal
15 days
Add together all of the user story estimates for Milestone 1.0
=
Estimate for Milestone 1.0
273 days
Does this sound reasonable?

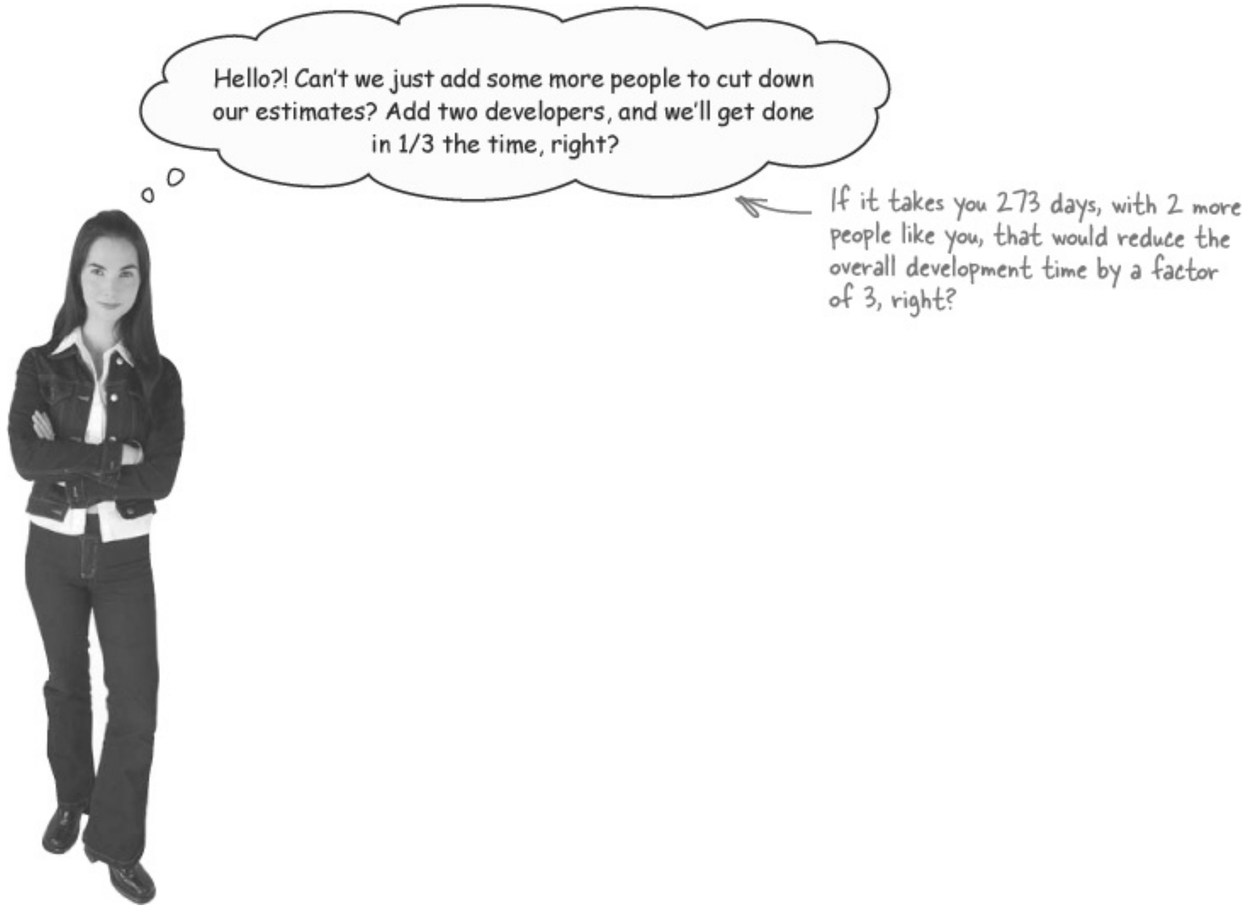
FEATURES FOR MVP DO NOT FIT 90 DAYS?



- 273 Days >>> 90 days!
- (common) REPRIORTIZE

1. Cut out more functionality
   - Establish what is absolutely critical
   - Explain schedule -> Customers will admit they need less
2. Ship a milestone build ASAP
   - Keeps up momentum and focuses team on deadline
   - Don't let customers talk you into > development cycles
3. Focus on baseline functionality
   - All you need is a functioning product…
   - Future features can be scheduled for later milestones

ADD MORE PEOPLE?


Hello?! Can't we just add some more people to cut down our estimates? Add two developers, and we'll get done in 1/3 the time, right?
If it takes you 273 days, with 2 more people like you, that would reduce the overall development time by a factor of 3, right?

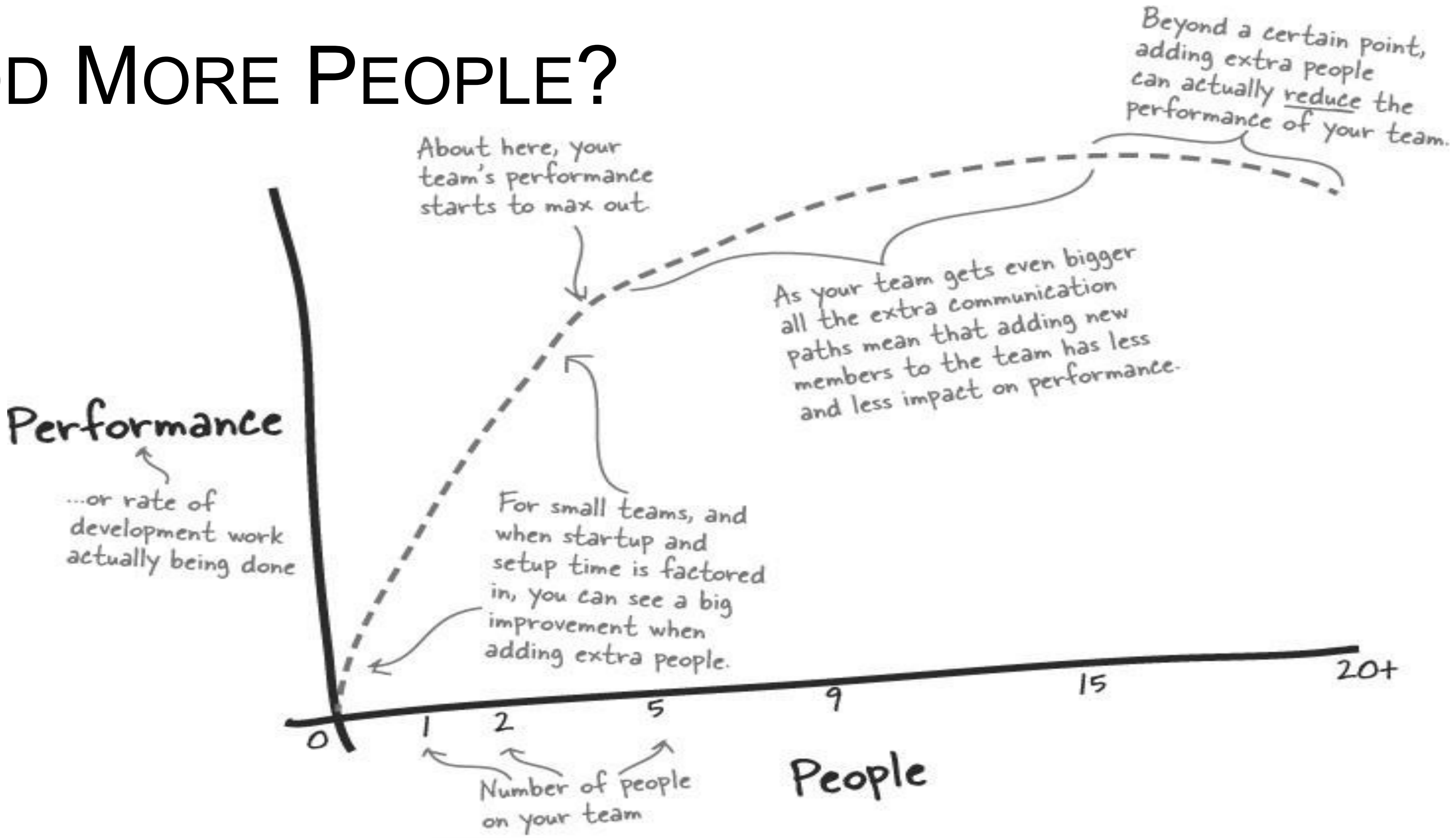
Add More People?

- E.g., If it takes you 273 days, with 2 more people like you, that would reduce the overall development time by a factor of 3, right?
- No. Why?
  - 1.?
  - 2.?
  - 3.?
- The best approach is to monitor your team, and if you start to see your team actually get less productive, even though you have more people, then it's time to re-evaluate the amount of work you have to do or the amount of time in which you have to do it.

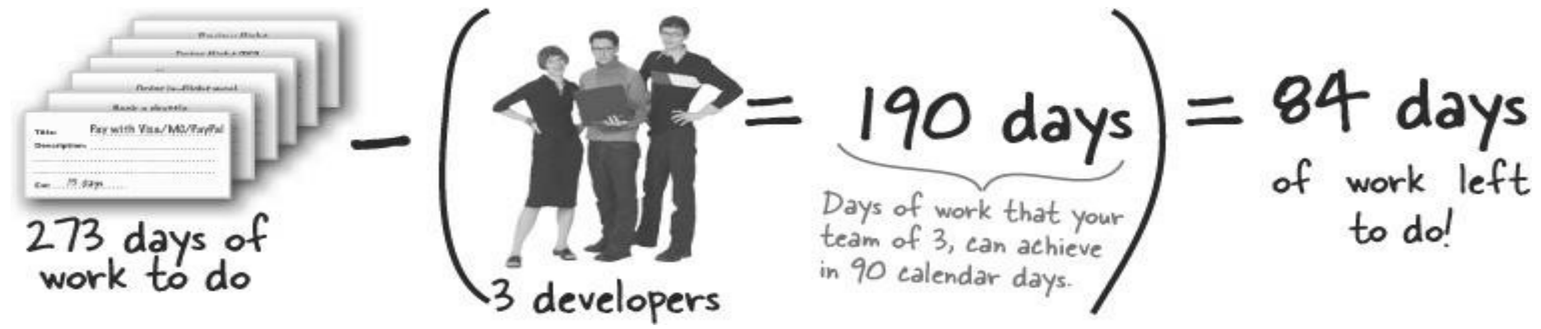
IT'S ABOUT MORE THAN DEVELOPMENT TIME

- Every new team member needs to get up to speed on the project; they need to:
  - Understand the software, the technical decisions, and how everything fits together
  - Get the new person set up with the right tools and equipment to work with the team
  - Every person you add to your team makes the job of keeping everyone focused and knowing what they are doing harder
- Adding more people to your team doesn't always work as you'd expect.
  - If 1 person takes 273 days to complete Milestone 1.0, then 3 people ***won't*** necessarily take 91. In fact they could actually take much longer!
  - Performance doesn't always increase with the size of your team

ADD MORE PEOPLE?

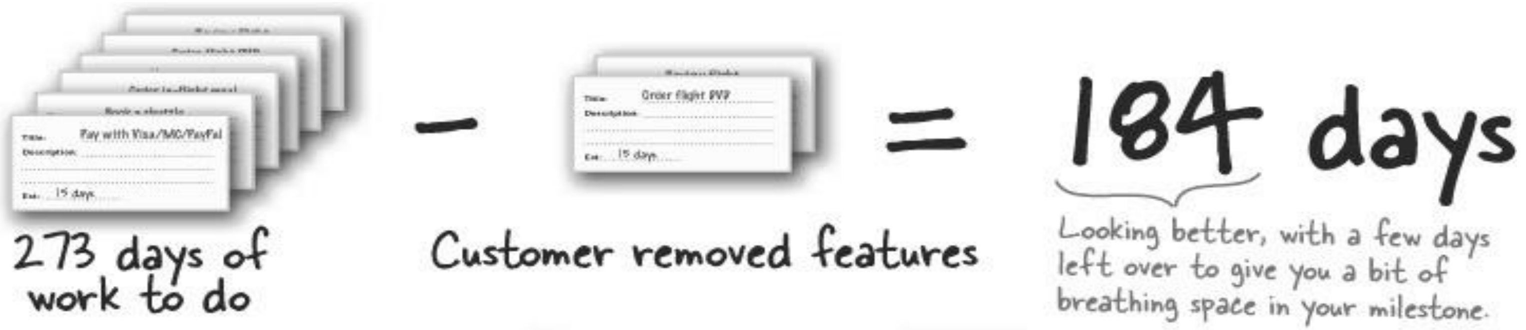


ADJUSTING THE MVP BASED ON DEADLINE



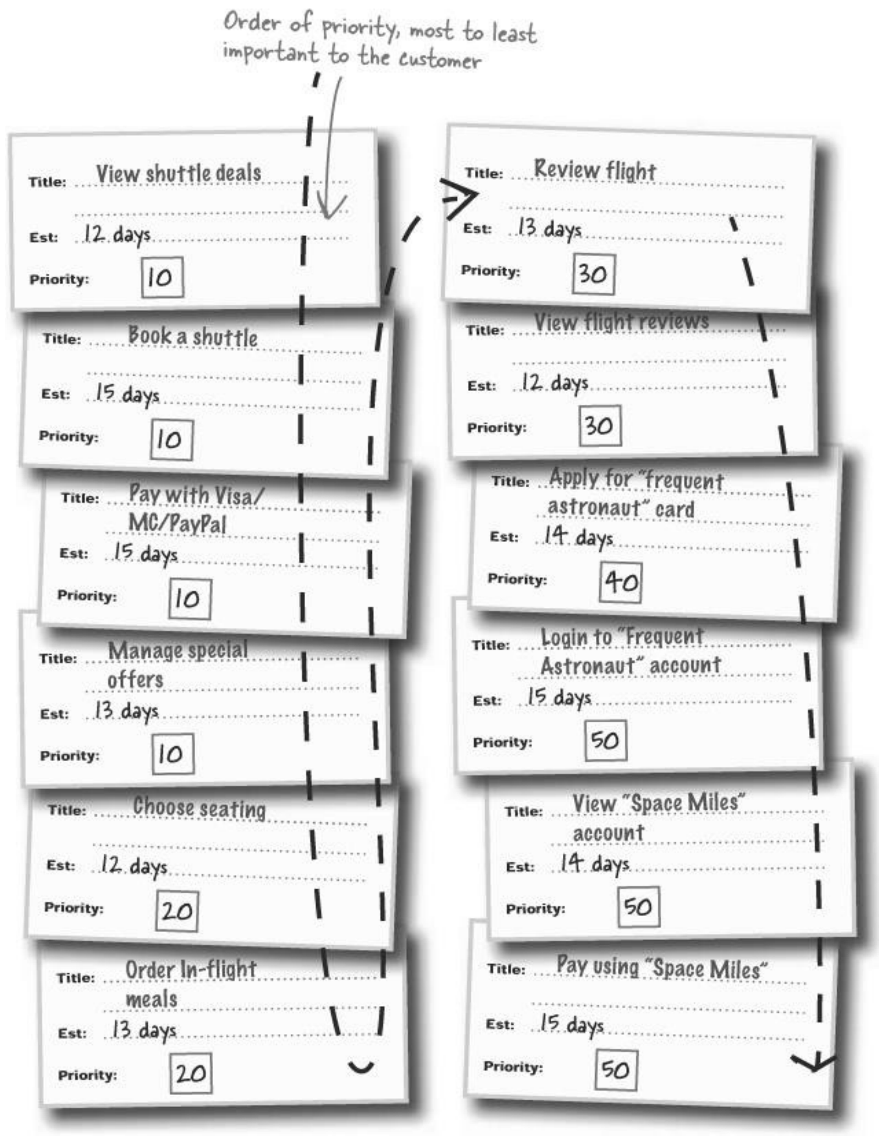
Pay with Visa/MC/PayPal
15 days
273 days of work to do
–
3 developers
= 190 days
Days of work that your team of 3, can achieve in 90 calendar days.
= 84 days of work left to do!

After Adjustment



Pay with Visa/MC/PayPal
15 days
273 days of work to do
–
Order flight DVD
15 days
Customer removed features
= 184 days
Looking better, with a few days left over to give you a bit of breathing space in your milestone.

Exercise


Order of priority, most to least important to the customer

Title: View shuttle deals
Est: 12 days
Priority: 10

Title: Book a shuttle
Est: 15 days
Priority: 10

Title: Pay with Visa/MC/PayPal
Est: 15 days
Priority: 10

Title: Manage special offers
Est: 13 days
Priority: 10

Title: Choose seating
Est: 12 days
Priority: 20

Title: Order In-flight meals
Est: 13 days
Priority: 20

Title: Review flight
Est: 13 days
Priority: 30

Title: View flight reviews
Est: 12 days
Priority: 30

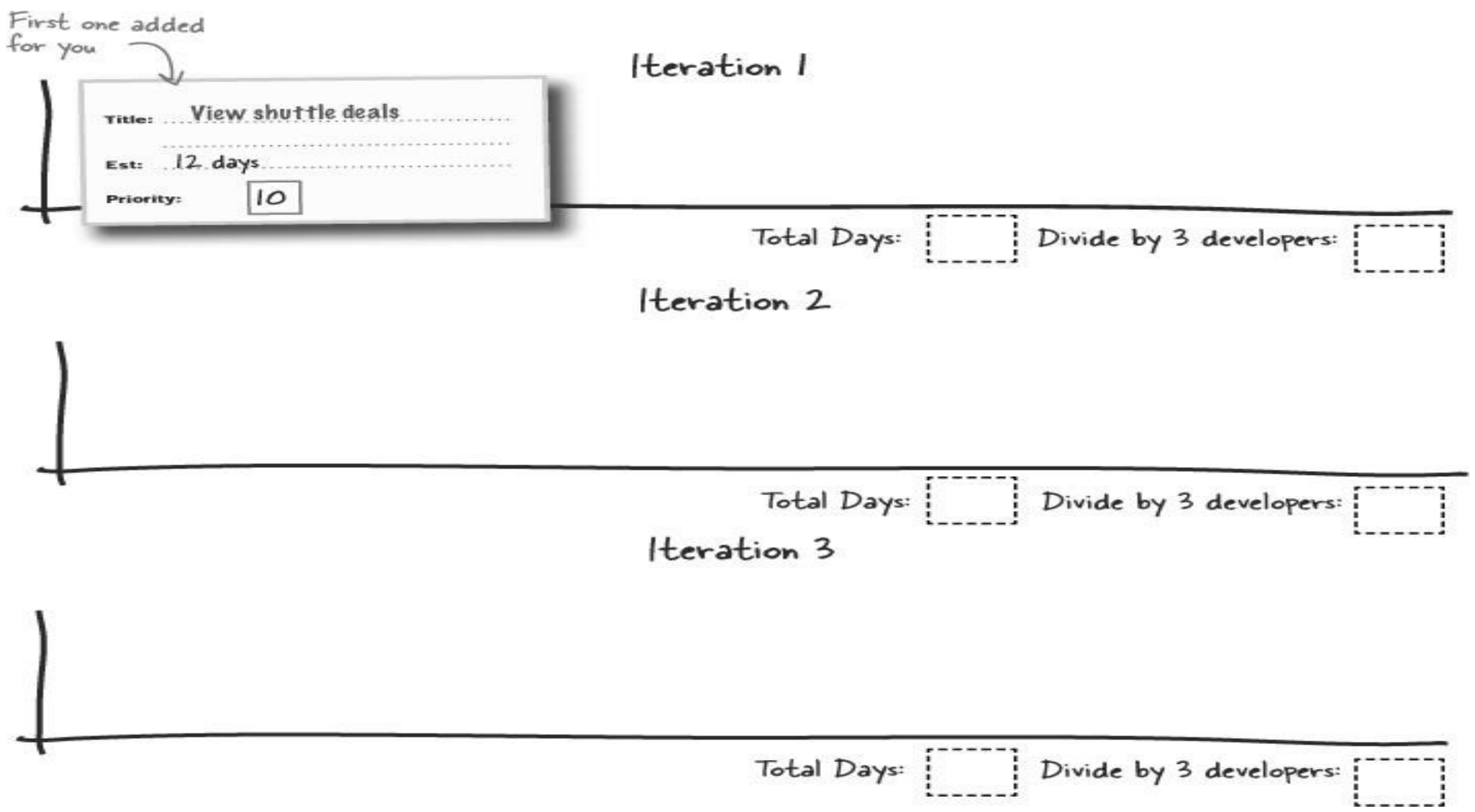
Title: Apply for "frequent astronaut" card
Est: 14 days
Priority: 40

Title: Login to "Frequent Astronaut" account
Est: 15 days
Priority: 50

Title: View "Space Miles" account
Est: 14 days
Priority: 50

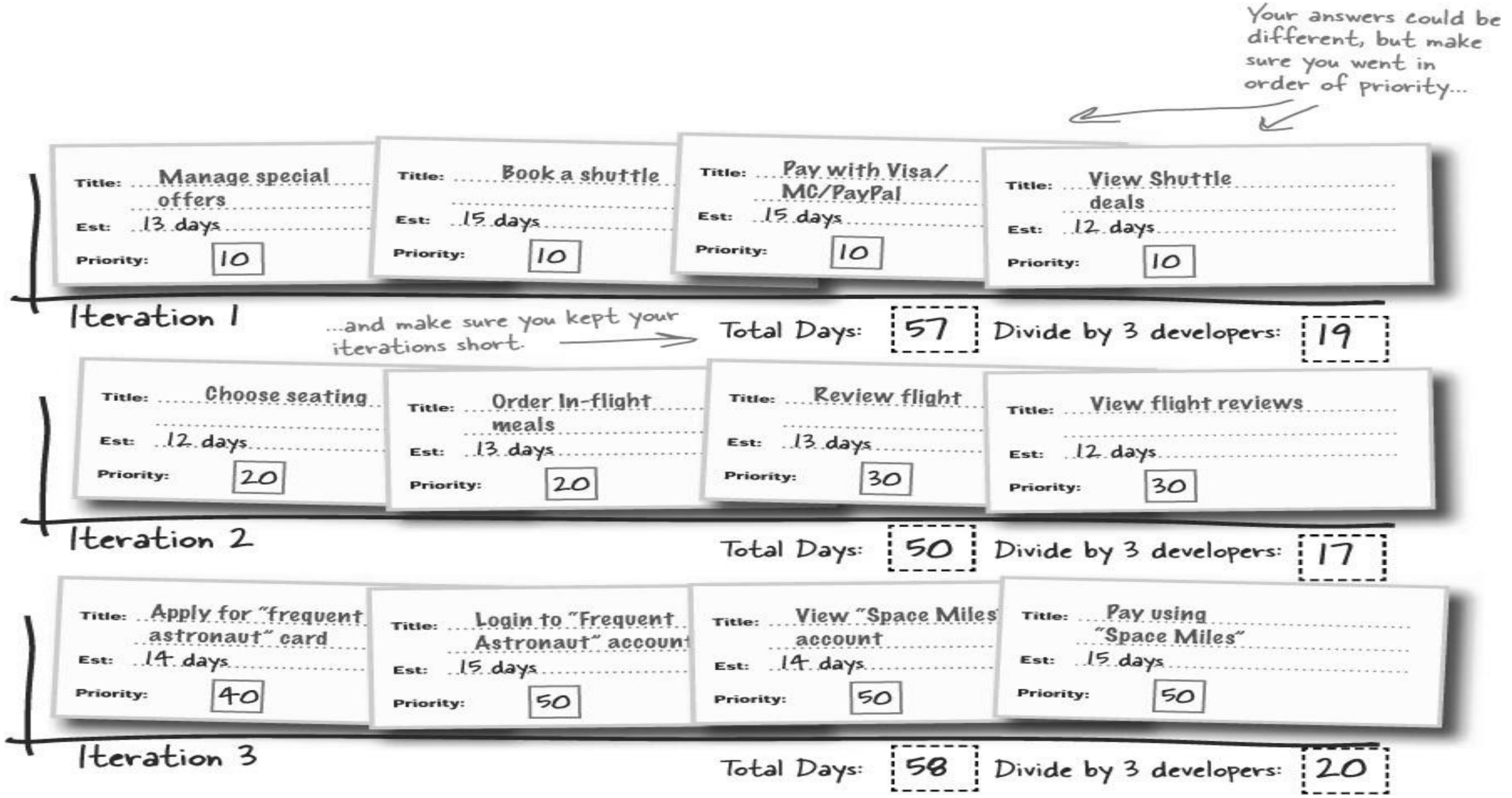
Title: Pay using "Space Miles"
Est: 15 days
Priority: 50

{ Aim For 20 Working Day Iterations }



MIAMI UNIVERSITY

Exercise Potential Solution



Your answers could be different, but make sure you went in order of priority...

Title: Manage special offers
Est: 13 days
Priority: 10

Title: Book a shuttle
Est: 15 days
Priority: 10

Title: Pay with Visa/MC/PayPal
Est: 15 days
Priority: 10

Title: View Shuttle deals
Est: 12 days
Priority: 10

Iteration 1

...and make sure you kept your iterations short.

Total Days: 57 Divide by 3 developers: 19

Title: Choose seating
Est: 12 days
Priority: 20

Title: Order In-flight meals
Est: 13 days
Priority: 20

Title: Review flight
Est: 13 days
Priority: 30

Title: View flight reviews
Est: 12 days
Priority: 30

Iteration 2

Total Days: 50 Divide by 3 developers: 17

Title: Apply for "frequent astronaut" card
Est: 14 days
Priority: 40

Title: Login to "Frequent Astronaut" account
Est: 15 days
Priority: 50

Title: View "Space Miles" account
Est: 14 days
Priority: 50

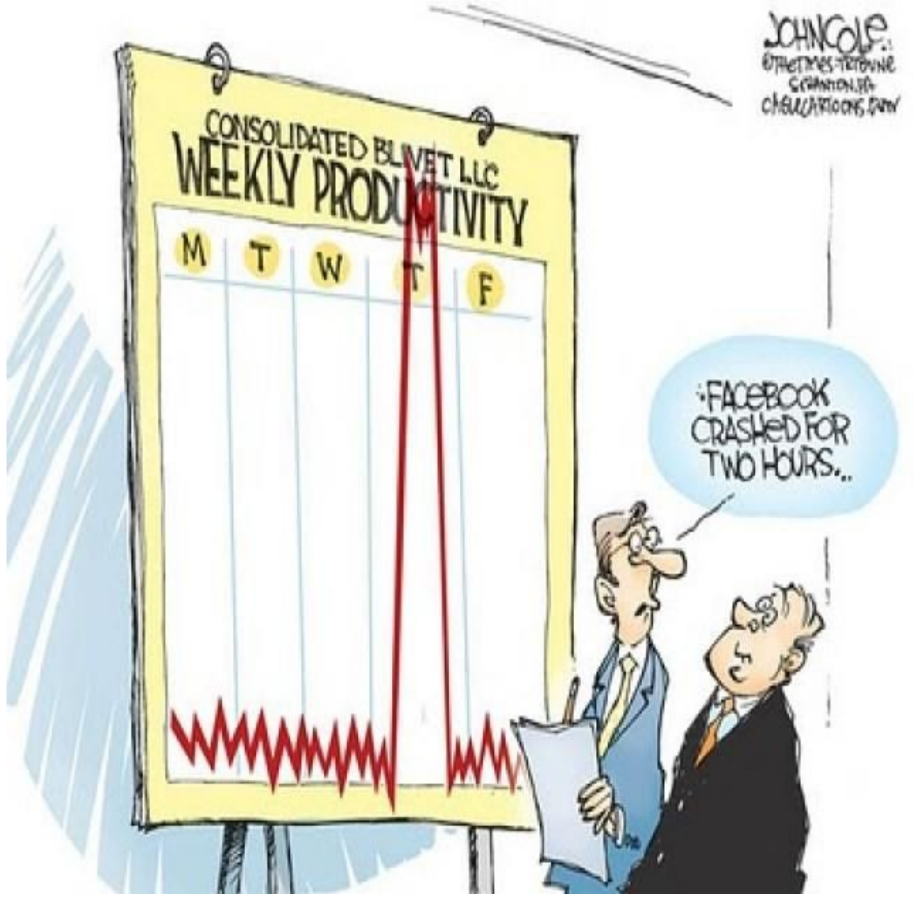
Title: Pay using "Space Miles"
Est: 15 days
Priority: 50

Iteration 3

Total Days: 58 Divide by 3 developers: 20

KEEP ITERATIONS SHORT & BALANCED

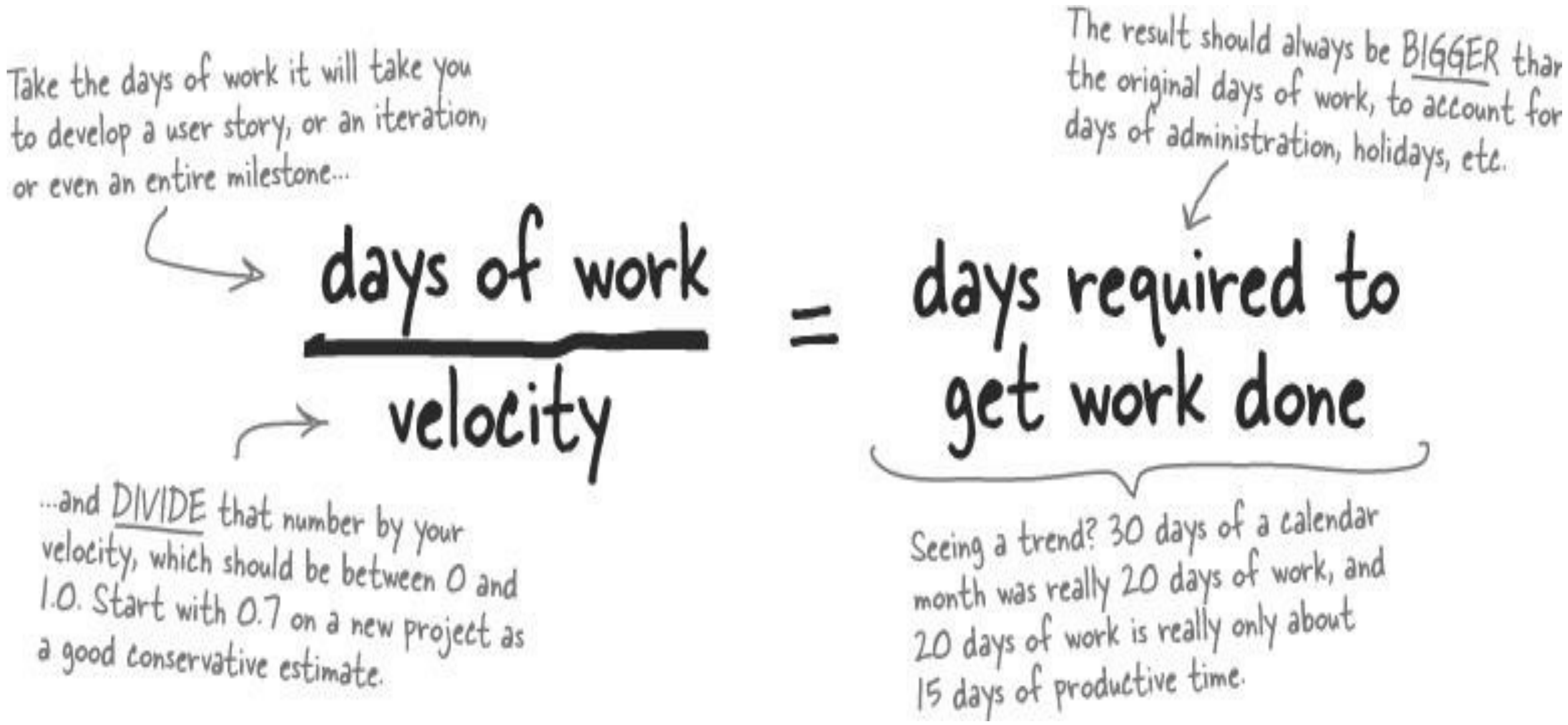
- 30-day iterations become 20 working days of “Productive” development.
  - Weekends
  - Vacation, bugs, and things that came up along the way
- Helps you deal with change and keeps you motivated and focused.



VELOCITY

- Velocity is a percentage: given X number of days, how much of that time is productive work?
  - Start with a velocity of 0.7.
  - This means your team has a velocity value of 0.7. (conservative)
  - For every 10 days of work time, about 3 of those days will be taken up
    - holidays, software installation, paperwork, phone calls, and other non-development tasks.
- Apply velocity to your amount of work, and get a realistic estimate of how long that work will actually take.

VELOCITY



Take the days of work it will take you to develop a user story, or an iteration, or even an entire milestone...

The result should always be BIGGER than the original days of work, to account for days of administration, holidays, etc.

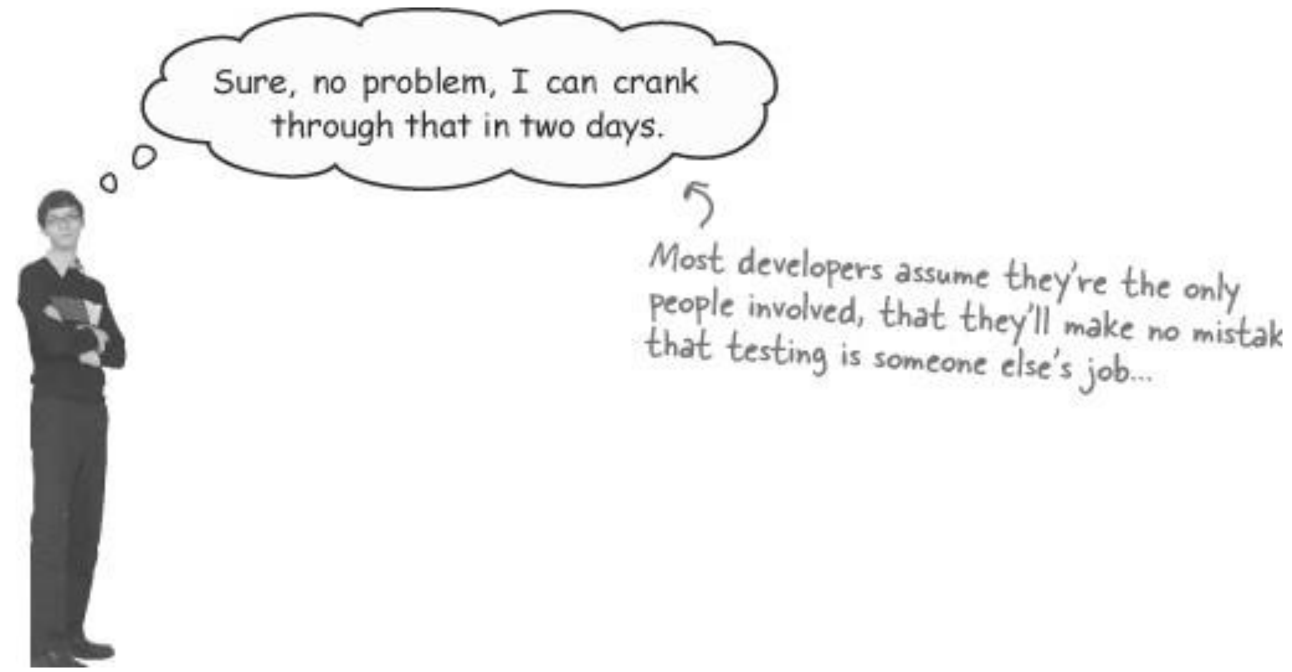
$$\frac{\text{days of work}}{\text{velocity}} = \underbrace{\text{days required to get work done}}$$

...and DIVIDE that number by your velocity, which should be between 0 and 1.0. Start with 0.7 on a new project as a good conservative estimate.

Seeing a trend? 30 days of a calendar month was really 20 days of work, and 20 days of work is really only about 15 days of productive time.

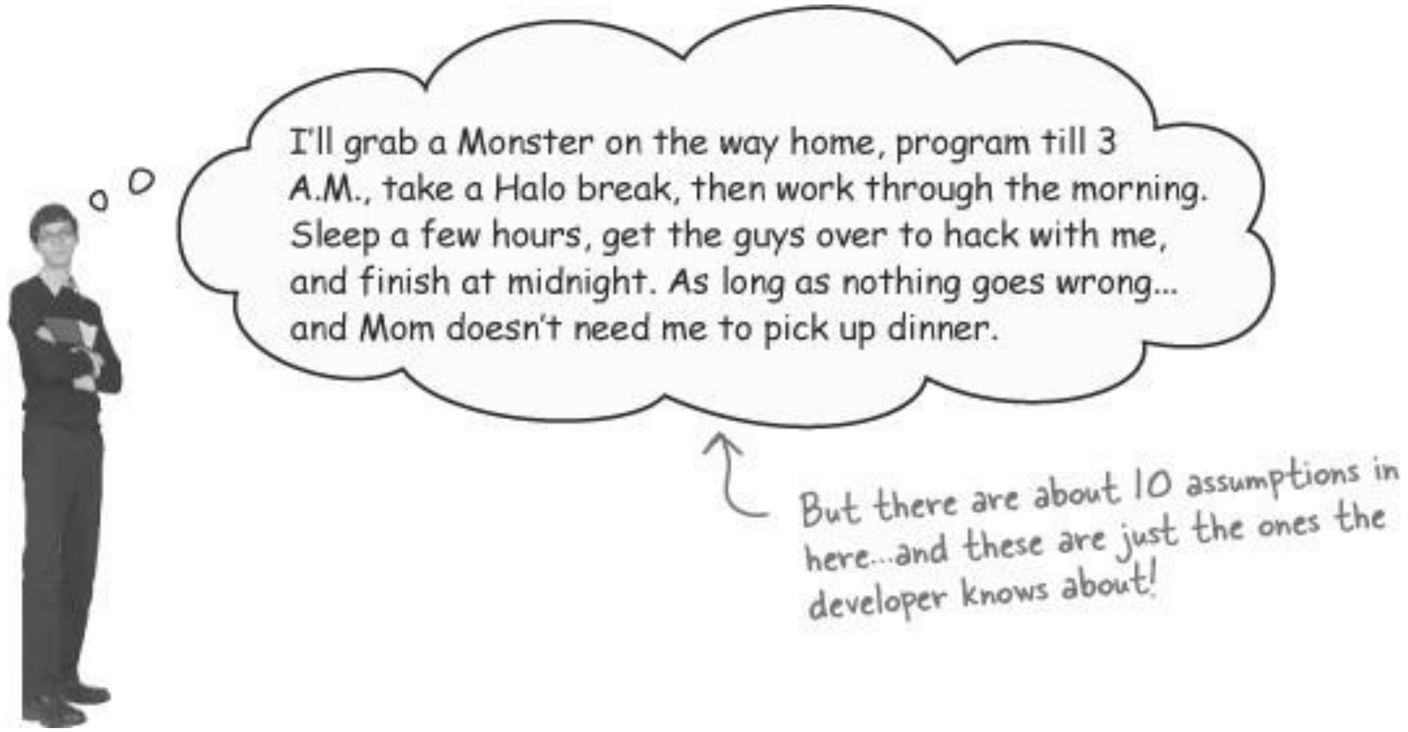
PROGRAMMERS – IDEALIST

- Give you a **better-than-best-case estimate**


Sure, no problem, I can crank through that in two days.
Most developers assume they're the only people involved, that they'll make no mistak that testing is someone else's job...

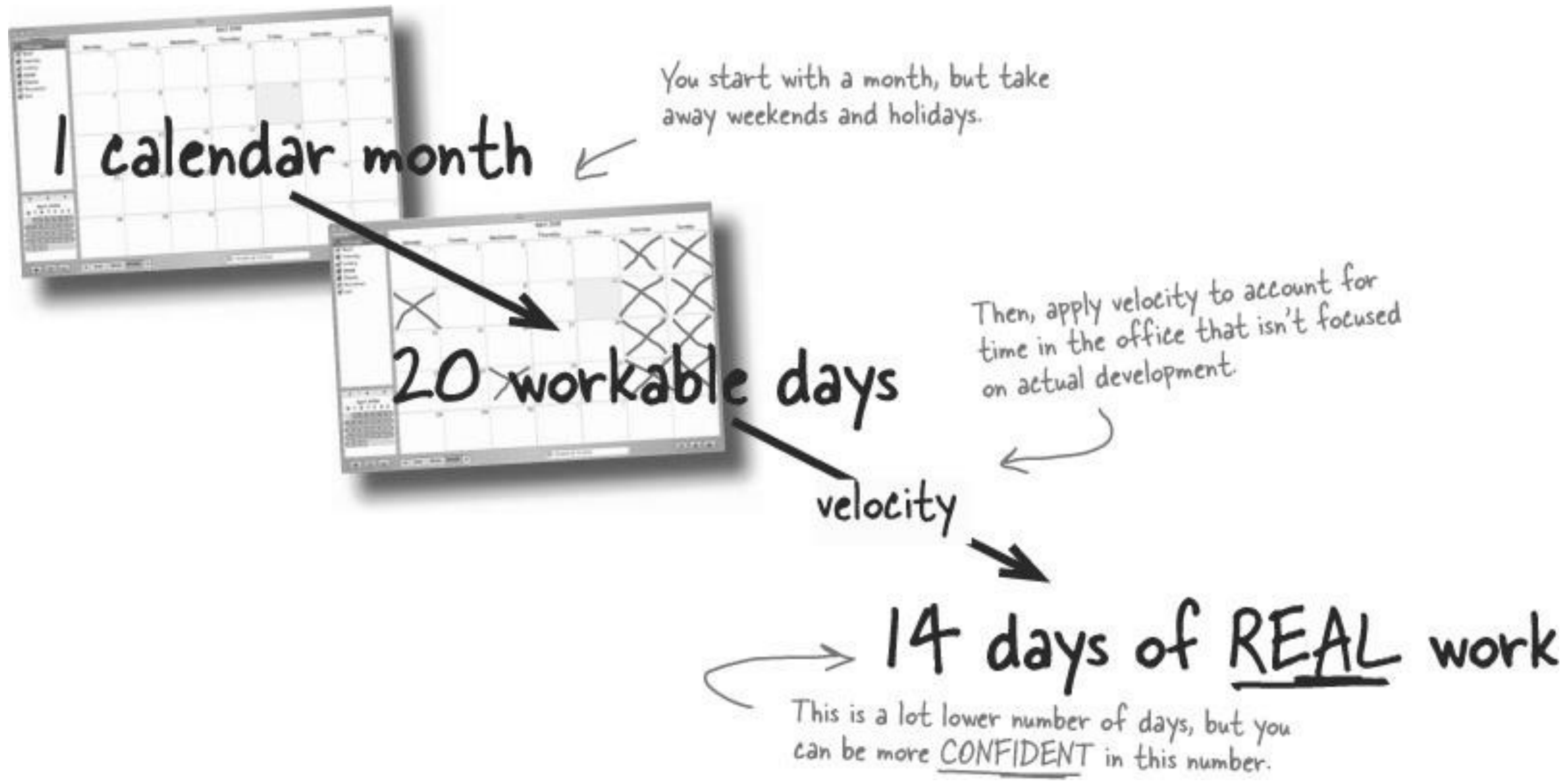
PROGRAMMERS – REALITY

- What they are really thinking…

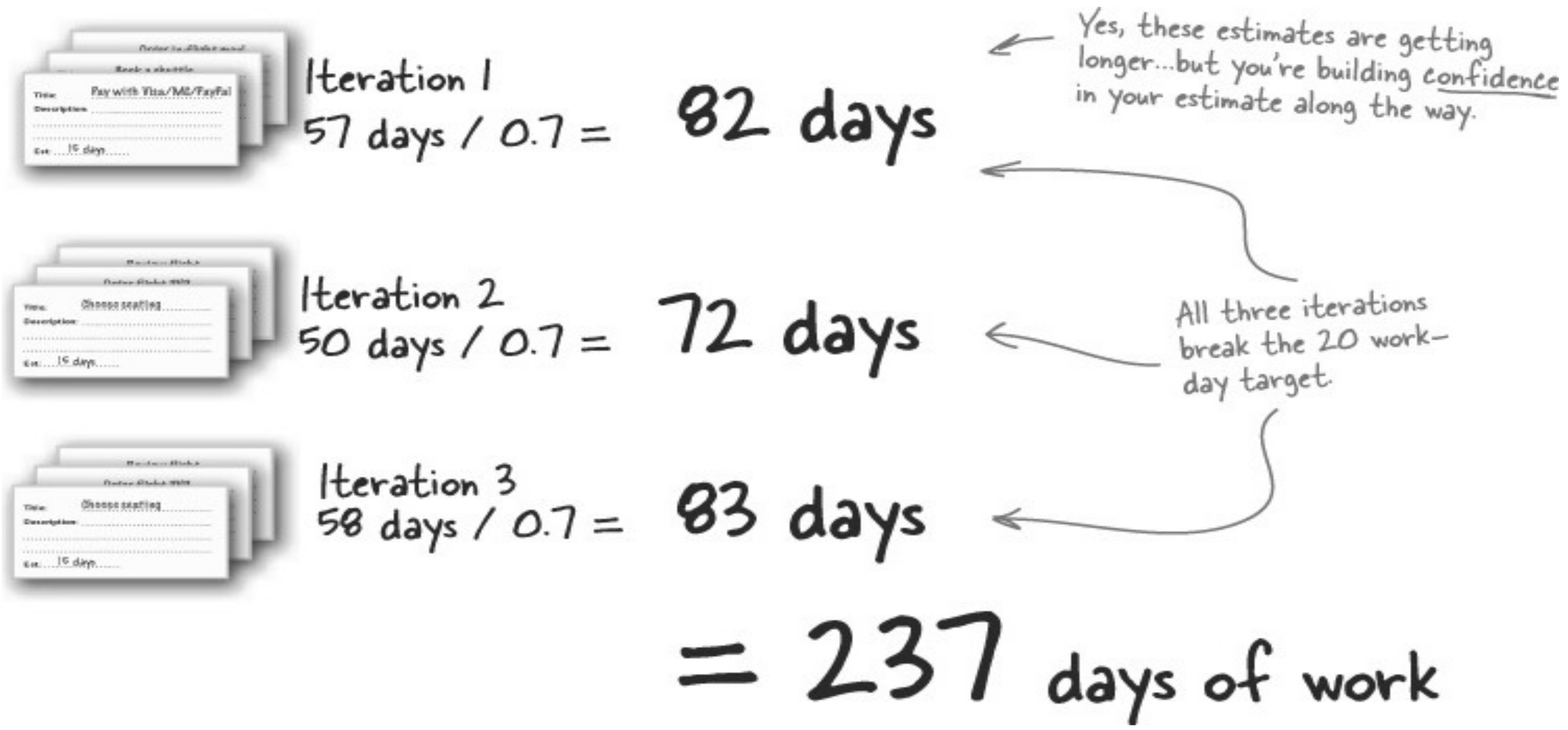

I'll grab a Monster on the way home, program till 3 A.M., take a Halo break, then work through the morning. Sleep a few hours, get the guys over to hack with me, and finish at midnight. As long as nothing goes wrong... and Mom doesn't need me to pick up dinner.
But there are about 10 assumptions in here...and these are just the ones the developer knows about!

DEVELOPERS

- We aim to be Software Developers
  - More conservative and realistic estimates


1 calendar month
You start with a month, but take away weekends and holidays.
20 workable days
Then, apply velocity to account for time in the office that isn't focused on actual development.
velocity
14 days of REAL work
This is a lot lower number of days, but you can be more CONFIDENT in this number.

Realistic Iteration Estimates



Iteration 1
57 days / 0.7 = 82 days

Iteration 2
50 days / 0.7 = 72 days

Iteration 3
58 days / 0.7 = 83 days

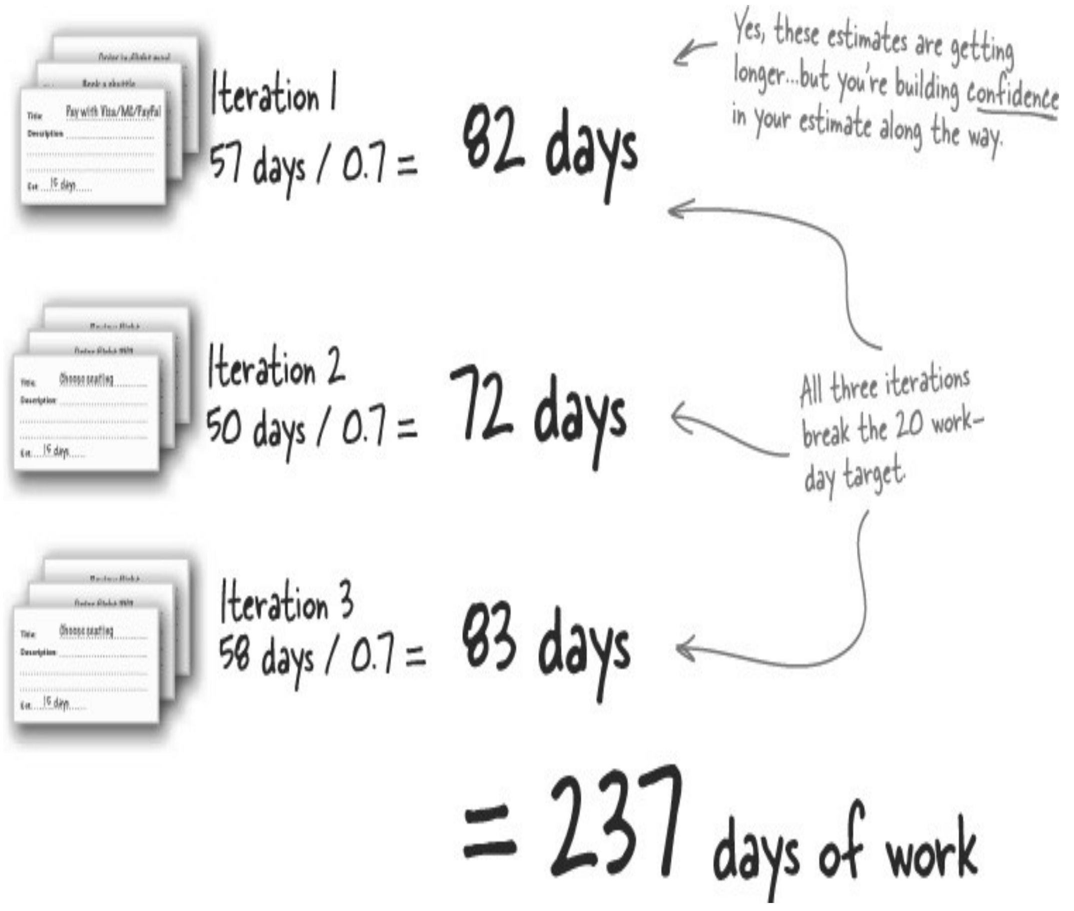
= 237 days of work

Yes, these estimates are getting longer...but you're building confidence in your estimate along the way.

All three iterations break the 20 work-day target.

Realistic Iteration Estimates

- So if you have 3 developers, each of them has to work 79 days in 3 months
- But there are only 60 working days per month


Iteration 1
57 days / 0.7 = 82 days
Yes, these estimates are getting longer...but you're building confidence in your estimate along the way.
Iteration 2
50 days / 0.7 = 72 days
All three iterations break the 20 work-day target.
Iteration 3
58 days / 0.7 = 83 days
= 237 days of work

Consider Velocity At The Start

{Before Breaking into Iterations}

- Apply velocity to each iteration

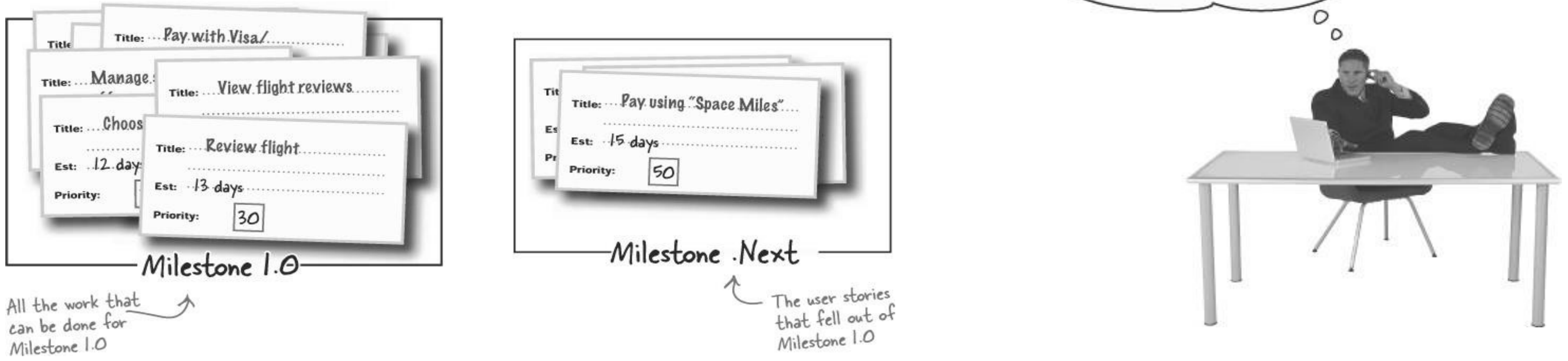
$$\underset{\text{The number of people on your team.}}{3} \times \underset{\text{20 working days in your iteration}}{20} \times \underbrace{0.7}_{\text{Your team's first pass velocity.}} = \underset{\text{The amount of work, in person-days, that your team can handle in one iteration.}}{42}$$

- Add your iterations up to get a total milestone

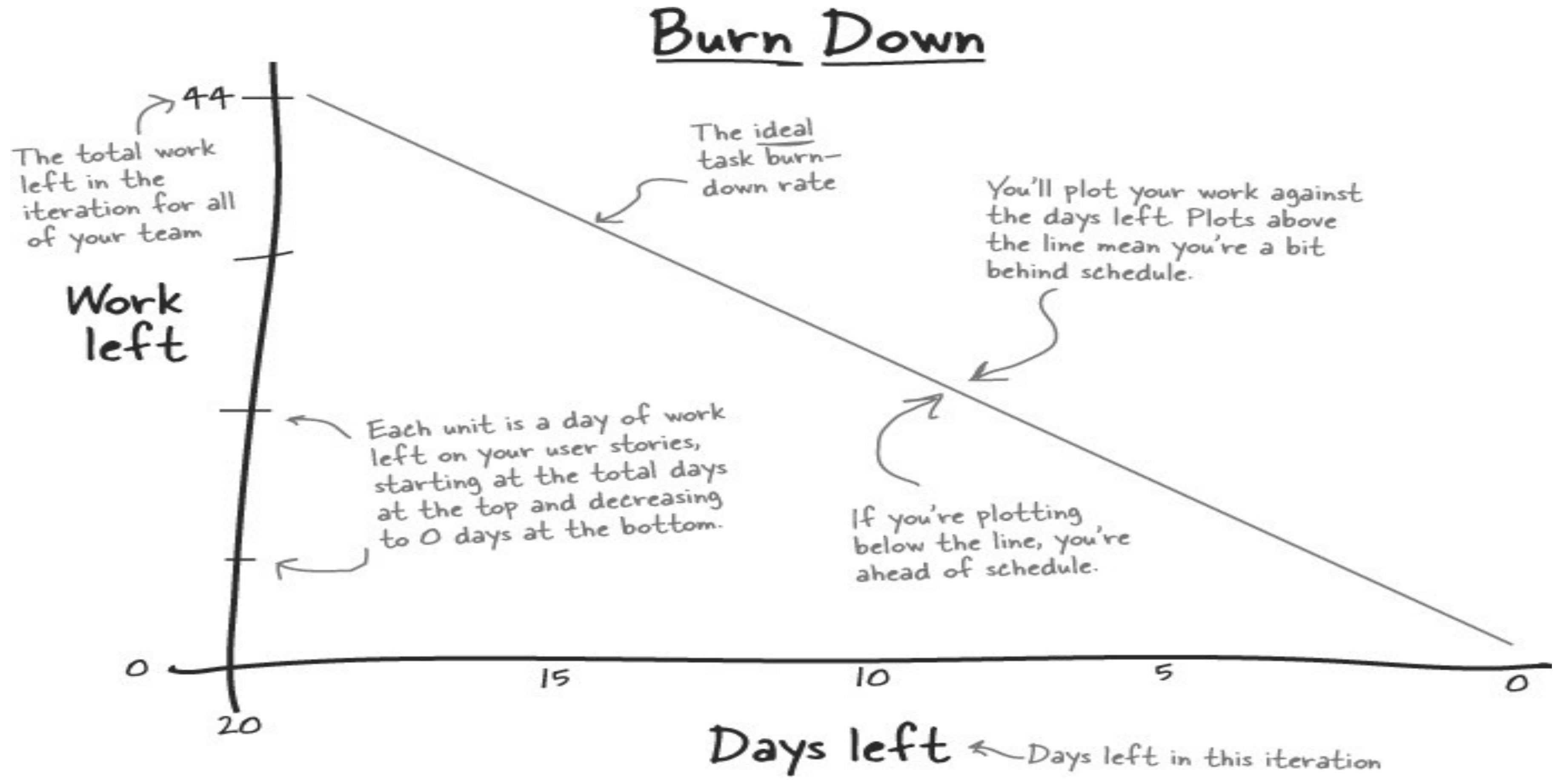
$$42 \times \underset{\text{Number of iterations in Milestone 1.0.}}{3} = \underbrace{126}_{\text{Amount of work in days that you and your team can do before Milestone 1.0 needs to be shipped.}}$$

The Customer Still Wants More!

- What do you do?
  1. Add an iteration to Milestone 1
  2. Explain that the overflow work is not lost, just postponed
  3. Show the customer the math, if they will listen
     - Be Transparent(?)!


Title: Pay with Visa/
Title: Manage
Title: View flight reviews
Title: Choos
Est: 12 day
Priority:
Title: Review flight
Est: 13 days
Priority: 30
Milestone 1.0
All the work that can be done for Milestone 1.0
Title: Pay using "Space Miles"
Est: 15 days
Priority: 50
Milestone Next
The user stories that fell out of Milestone 1.0
Alright, it's worth it to me to lose space miles in Milestone 1.0 to keep things moving. Let's do it.

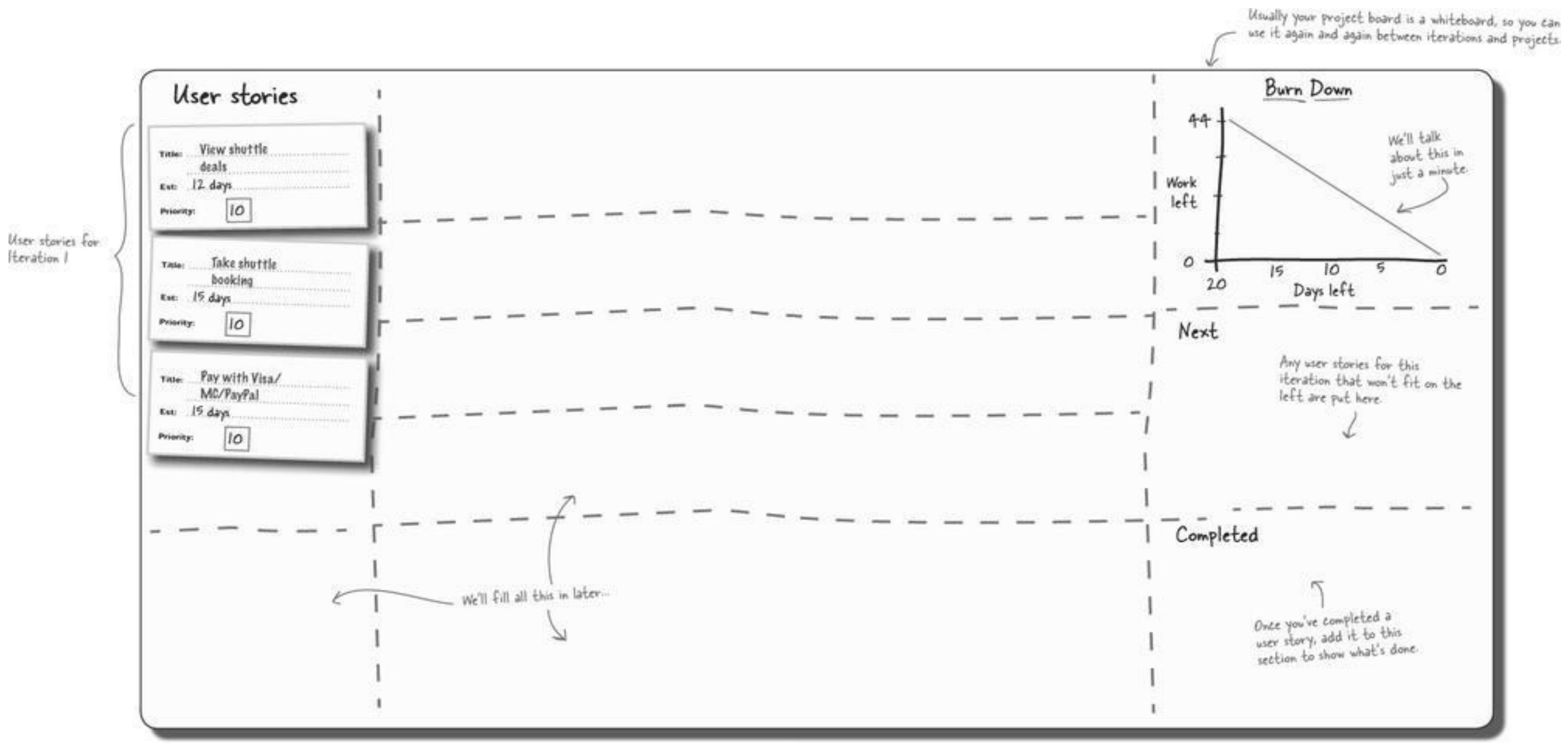
BE AWARE OF BURN DOWN (?)



Be Aware Of Burn Out

- Don’t overschedule or over work
- Personal lives are a factor
  - A happier team is a more productive team.
- Fatigue affects productivity
  - Lots of studies suggest that developers are really only incredibly productive for about three hours a day
  - The more tired your developers are, the less likely they’ll even get to that three hours of really productive time
  - Fatigue makes cowards of us all

THE BIG BOARD


Usually your project board is a whiteboard, so you can use it again and again between iterations and projects.
User stories
User stories for Iteration 1
Title: View shuttle deals
Est: 12 days
Priority: 10
Title: Take shuttle booking
Est: 15 days
Priority: 10
Title: Pay with Visa/MC/PayPal
Est: 15 days
Priority: 10
Burn Down
44
Work left
0
20
15
10
5
0
Days left
We'll talk about this in just a minute.
Next
Any user stories for this iteration that won't fit on the left are put here.
Completed
Once you've completed a user story, add it to this section to show what's done.
We'll fill all this in later...

Next Class – User Stories

EXTREME PROGRAMMING
I CAN'T GIVE YOU ALL OF THESE FEATURES IN THE FIRST VERSION.
AND EACH FEATURE NEEDS TO HAVE WHAT WE CALL A "USER STORY."
OKAY, HERE'S A STORY: YOU GIVE ME ALL OF MY FEATURES OR I'LL RUIN YOUR LIFE.
www.dilbert.com scottadams@aol.com
© 2002 United Feature Syndicate, Inc.