



Lean Agile Software Projects

Chris Riesbeck
Electrical Engineering and Computer Science
Learning Sciences
Northwestern University



Starting a Project?





Feature Presentation: Agile Intervention

Week 1

We've got the green light for our **WorkIt!** app. They love the idea of a personalized audio workout coach and exercise tracker.



WorkIt!



Your personal trainer

hands-free coaching

instant diary

100's of downloadable
routines

Week 1

OK, let's figure out what we need to build.



Week 1

OK, let's figure out what we need to build.

... a page to manage authors, users, and profiles...

A phone app to call out the steps...

We'll need a server, home page, page to define workout routines...

OK, here's a schedule. Let's get going!



User admin module	Workout entry module	User sign up, enter profile	Download custom workout	Workout Caller	Demo!
Week 1	Week 2	Week 3	Week 4	Week 5	Week 6

Week 2

Where's my new user admin page?

I spent 3 hours on it. I uploaded it last night.

Were you working on that? I may have covered it up when I uploaded my code this morning!

Where is the new database?

It's on my laptop ...

that I left on the shuttle



Week 3



Task list

Week 3



Task list

- Add Safari support
- Animate intro logo
- Remove premium link
- Update Ruby gems
- Find div-by-0 bug
- Change to daily logs

Week 3



The logo dances now!

Was that on the list?

What about the Facebook link?

Task list

Add Safari support
Animate intro logo
Remove premium link
Update Ruby gems
Find div-by-0 bug
Change to daily logs

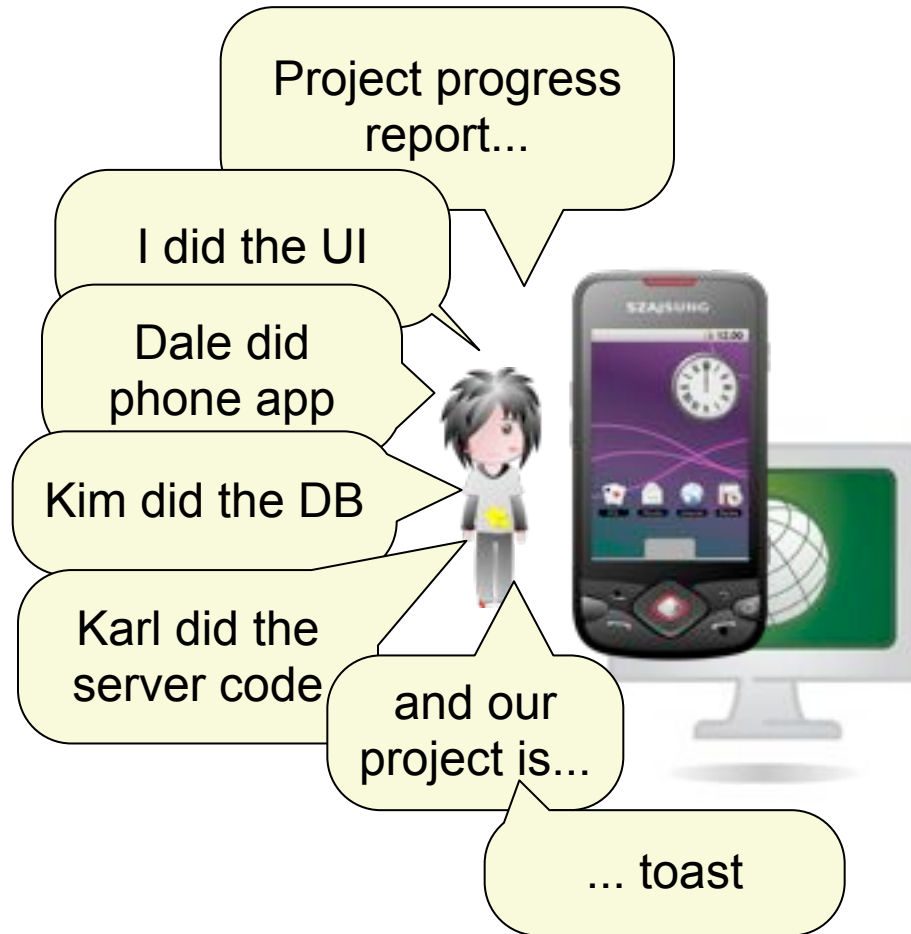


Week 4

Project progress
report...



Week 4



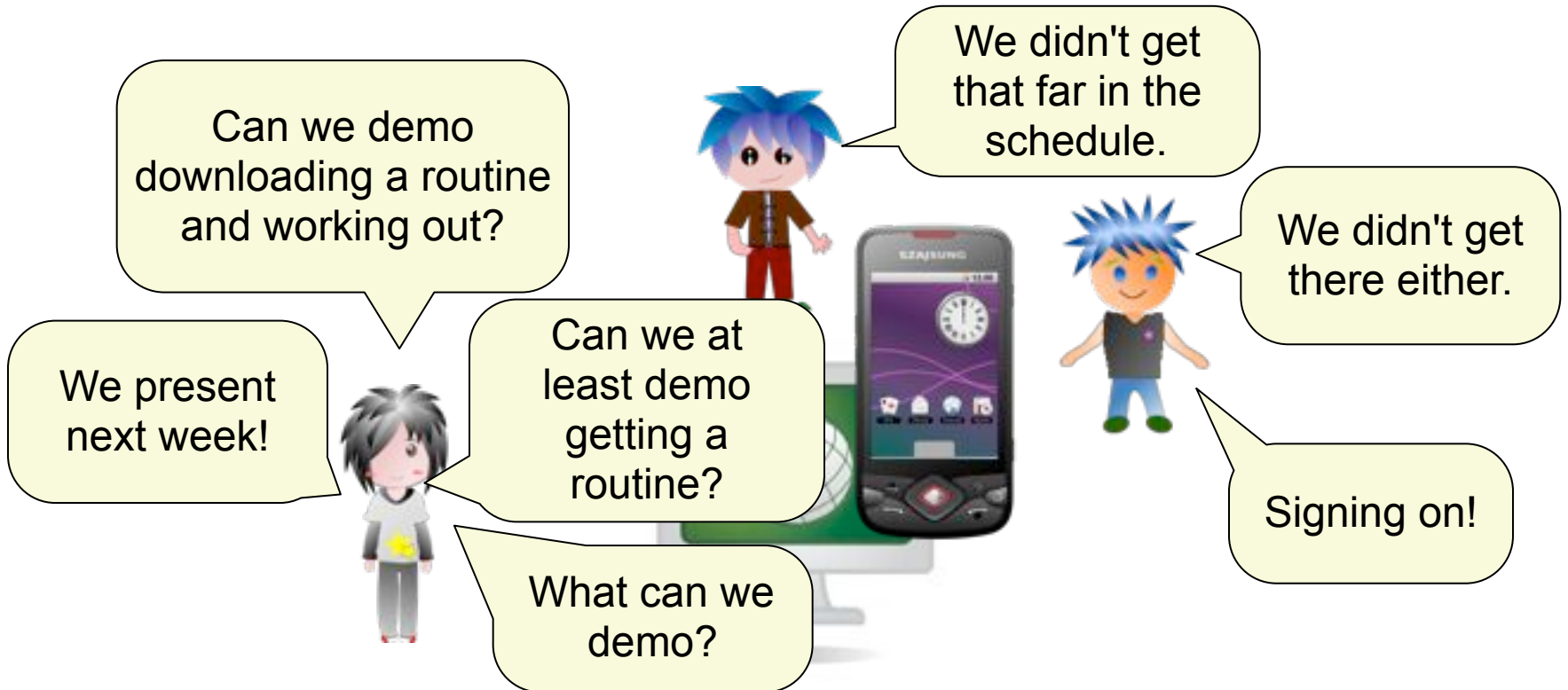
Week 5

We present next week!



User admin module	Workout entry module	User sign up, enter profile	Download custom workout	Workout Caller	Demo!
Week 1	Week 2	Week 3	Week 4	Week 5	Week 6

Week 5



User admin module	Workout entry module	User sign up, enter profile	Download custom workout	Workout Caller	Demo!
Week 1	Week 2	Week 3	Week 4	Week 5	Week 6



Suddenly...

Well, that was a disaster. What a run of bad luck.



Well, that was a disaster. What a run of bad luck.



Who are those guys?



We saw you were having problems.

We're your **Guardian Agiles.**



Luck had nothing to do with it— watch!

Week 2



Where's my new user admin page?

Were you working on that? I may have covered it up when I uploaded my code this morning!

Where is the new database?

It took me 3 hours. I uploaded it last night.



It's on my laptop ... that I left on the shuttle

This is classic code chaos.



It gets worse and worse until a project collapses.



Week 3



The logo dances now!

Task list

- ~~Add Safari support~~
- ~~Animate intro logo~~
- ~~Remove premium link~~
- Update Ruby gems
- Find div-by-0 bug
- Change to daily logs

Was that on the list?

What about the Facebook link?

You had too many tasks to keep track of.

And no focus on what really mattered.

Week 4

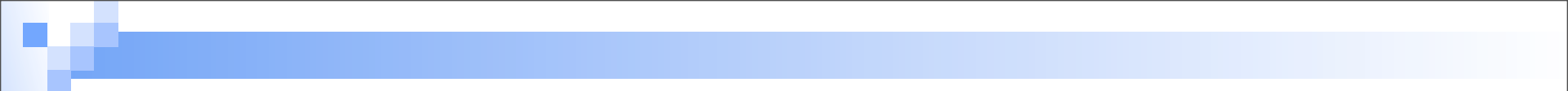


It's just a scratch. I'll be out of traction

Your bus factor was 1



That's how few people need to get hit by a bus for your project to fail.



Week 5

Can we demo downloading a routine and working out?

We present next week!

Can we at least demo getting a routine?

What can we demo?

We didn't get that far in the schedule.

We didn't get there either.

Signing on!

User admin module	Workout entry module	User sign up, enter profile	Download custom workout	Workout Caller	Demo!
Week 1		Week 3	Week 5		

Here's your biggest problem of all.

Your schedule saved the best for last, i.e., never.

I guess we have to try harder.



I guess we have to try harder.



Not harder.
Different. One key
idea—early value—
and a few practices
can help a lot.

You're going to
get a chance to
do this again.



We'll pop
in to help.



Here we
go...

Week 1

... a page to manage authors, users, and profiles...

A phone app to call out the steps...

We'll need a server, home page, page to define workout routines...



Week 1

... a page to manage authors, users, and profiles...

A phone app to call out the steps...

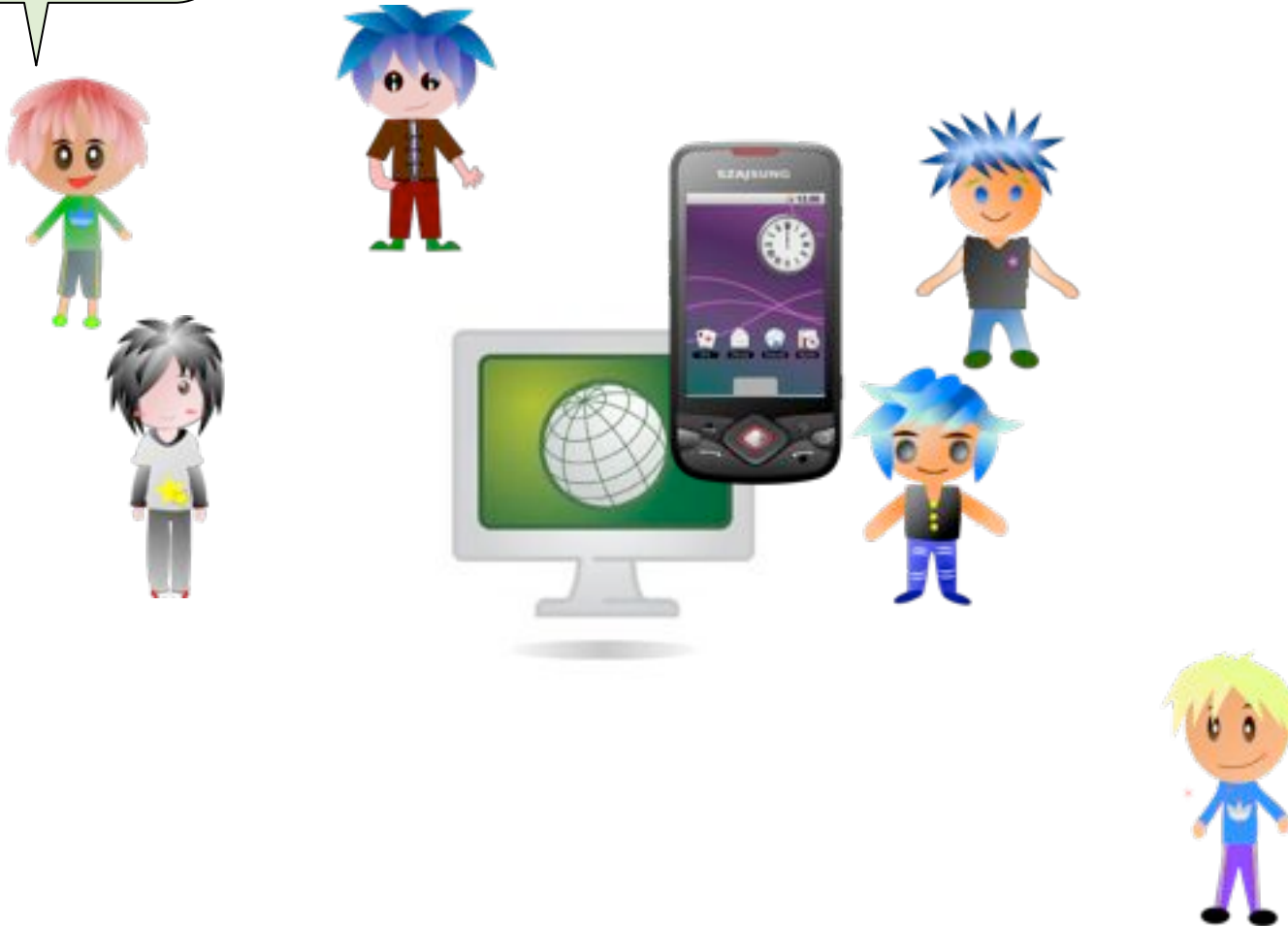
We'll need a server, home page, page to define workout routines...

Your value is getting lost in features and code.



Week 1

Start with a different question.



Week 1

Start with a different question.



What are your **user scenarios**?
The ones you must have in your **MVP**?

An expert authoring a workout routine.

A user entering an exercise goal profile.



A user doing a routine with the phone calling out the steps.



Great! Put those scenarios into a **release backlog**.



A user finding and downloading a routine that fits her profile.



Week 1

OK, we've got our backlog. Let's start coding!

Let's decide what to do first.

Well, clearly we need a page to create users...

...and a way to author new workout routines...



Scenarios

expert authors routine

user adds profile

user downloads routine

app calls out routine steps

Week 1

Let's decide what to do first.

Well, clearly we need a page to create users...

OK, we've got our backlog. Let's start coding!

...and a way to author new workout routines...

None of these test your **core value prop!**



Scenarios

expert authors routine

user adds profile

user downloads routine

app calls out routine steps

Week 1

What's the make or break scenario?



Scenarios

expert authors routine

user adds profile

user downloads routine

app calls out routine steps

Week 1

What's the make or break scenario?



If it doesn't work, nothing else matters?



A user doing a workout with the phone calling out the steps.



Then do that first.



Don't defer value. Build as if each release might be your last.

Scenarios

expert authors routine

user adds profile

user downloads routine

app calls out routine steps

Week 1

OK, let's implement running a workout routine.



We should be able to get that scenario done in two weeks



Scenarios

expert authors routine

user adds profile

user downloads routine

app calls out routine steps

Week 1

OK, let's implement running a workout routine.



We should be able to get that scenario done in two weeks



That's too vague. It's going to slip.

Scenarios

expert authors routine

user adds profile

user downloads routine

app calls out routine steps

Week 1

Set hard deadlines.
One-week **timeboxes**
work well.

ZFR: zero feature release (MVP skeleton).
All key parts (db, server, app, ...) running
and talking to each other.



Each week is
an internal
release you can
user test.

only if needed



Week 1

In a week?
How can we
possibly...



Week 1

In a week?
How can we
possibly...

DTSTTCPW: Do the simplest
thing that could possibly work.

Focus on
one new user
function.

DTSTTCPW!

Don't build a
database
editor if a text
file will work.

Backlog anything
not essential for
validated learning.

Do you really
need user login
just now?

Week 1

OK, let's get started. I'll do the UI, Kim the DB, ...



Week 1

OK, let's get started. I'll do the UI, Kim the DB, ...



Don't deal out tasks.



And don't silo!



Silos: Separating developers by specialized skills and functionality.

Week 1



Make just one queue, most important stories first. Take the next one in line when you're ready, no matter what it is.



Swarm! Work together to get one story done before starting another.



Story	To Do	In Process	Done
User can upload game	file upload	game parser	
Users can team up		user join team	create team
Users can edit avatars	avatar edit page avatar list page		



This will reduce **work in progress**, and cross-train to improve your bus factor.

Agile Task Board: An project information radiator.

Week 1

At last, time to code!



I put the web server on my dorm desktop with FTP.



Everyone, be sure email the team when you're working on a file!



Scenarios

expert authors routine

user adds profile

user downloads routine

app calls out routine steps

Week 1

At last, time to code!



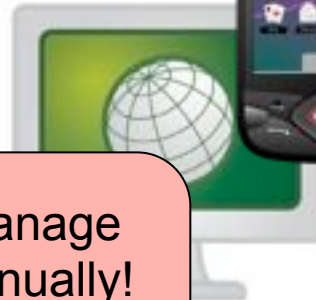
I put the web server on my dorm desktop with FTP.



Everyone, be sure email the team when you're working on a file!



Don't manage code manually!



Repeat after me

Scenarios

expert authors routine

user adds profile

user downloads routine

app calls out routine steps



Source control



Put everything, code, documentation, sketches, ... in a source control repository.

A cloud solution like github works well.

Make sure everyone checks out a copy at least daily.





Week 1... 2...

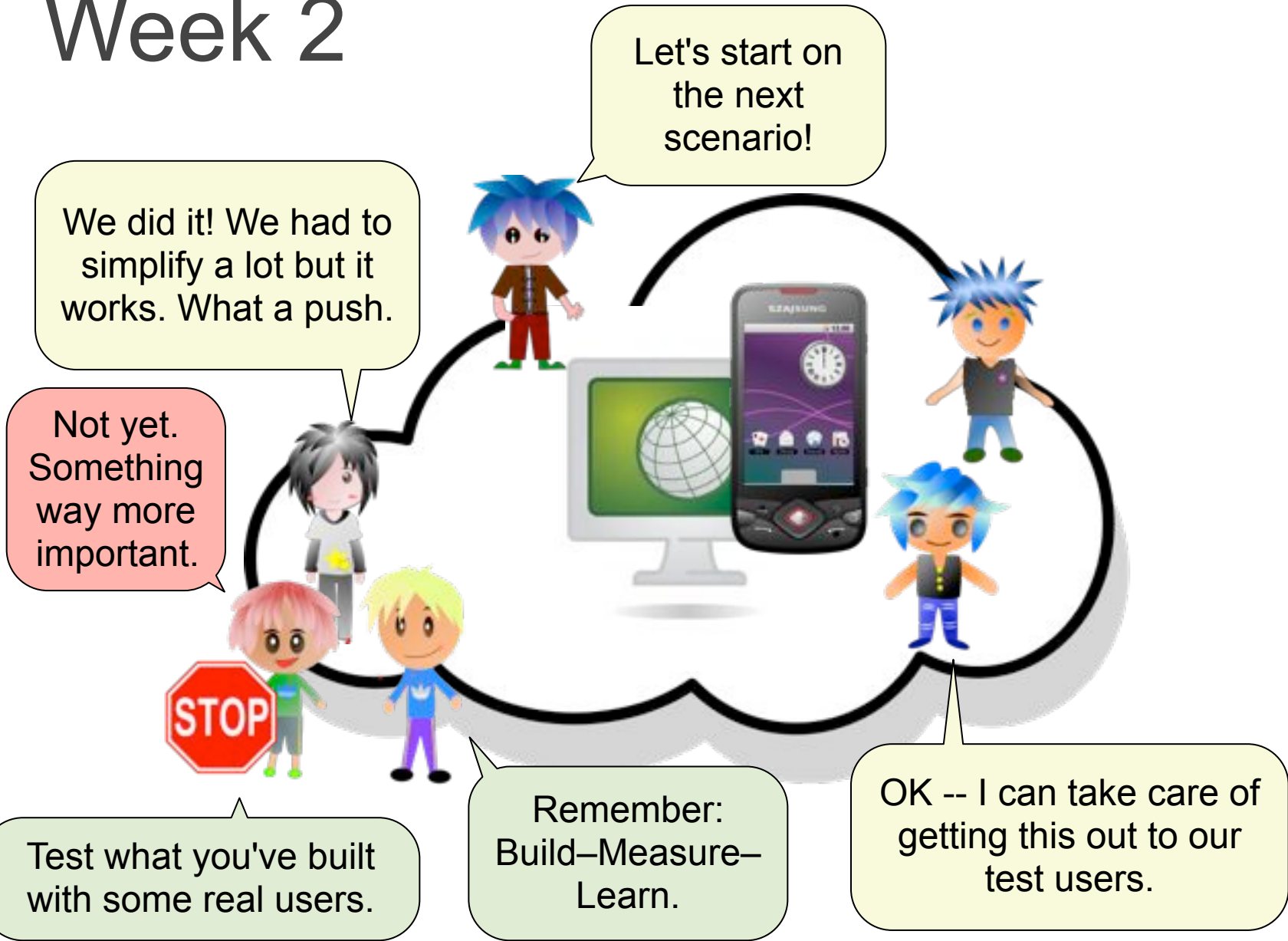
Week 2

We did it! We had to simplify a lot but it works. What a push.

Let's start on the next scenario!



Week 2



Week 2

OK, some users are using Release 1.

Now for the next scenario.



Week 2

OK, some users are using Release 1.

Now for the next scenario.

Not yet. Reflect first!



What didn't work out so well this first iteration?

Why not? What change might help?



Week 2

One of us didn't use the source control system.



Week 2

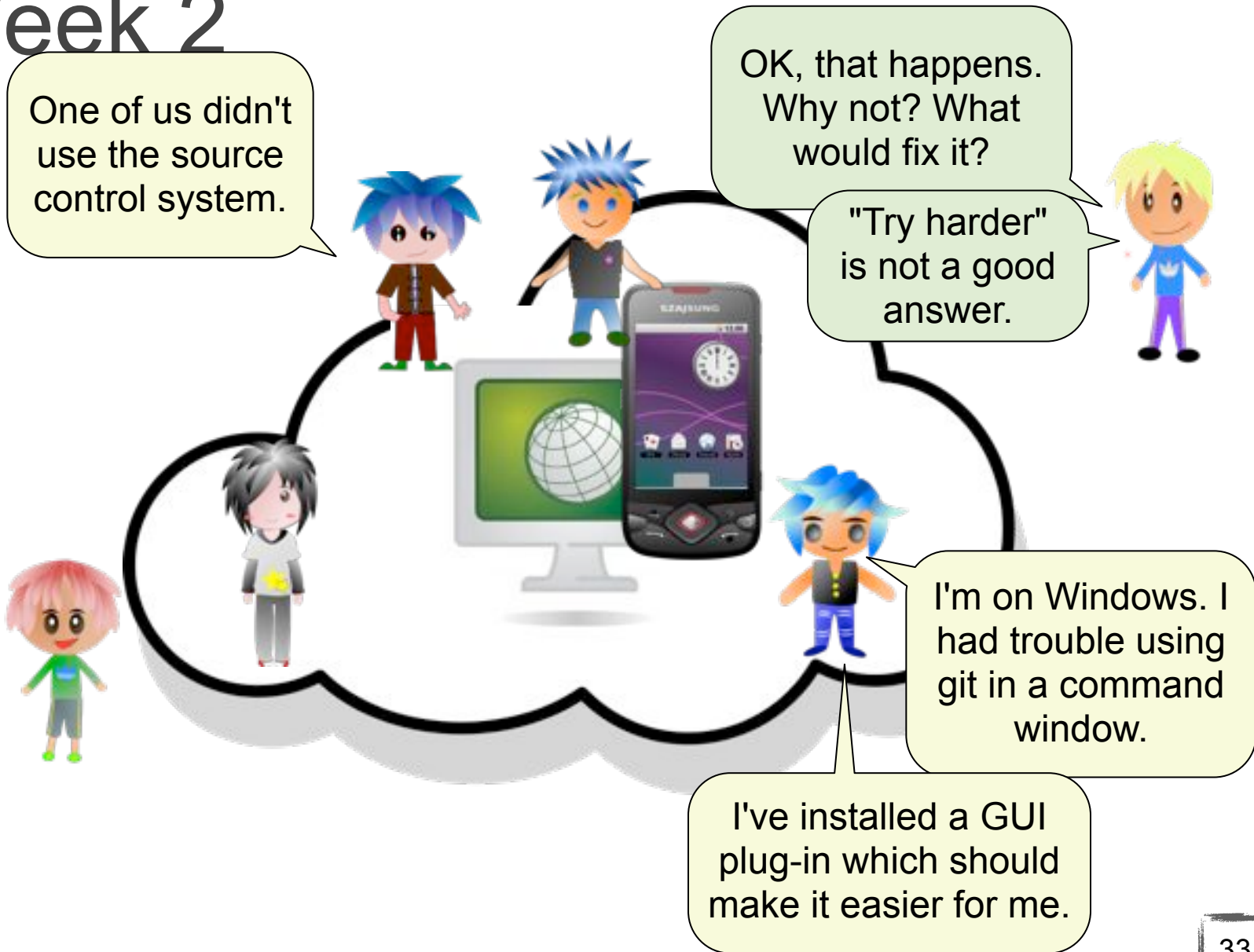
One of us didn't use the source control system.

OK, that happens. Why not? What would fix it?

"Try harder" is not a good answer.

I'm on Windows. I had trouble using git in a command window.

I've installed a GUI plug-in which should make it easier for me.



Week 2

OK, we're testing,
we've changed
some processes...

Now for the
next scenario.

Let's start
coding!



Scenarios

expert authors routine

user adds profile

user downloads routine

app calls out routine steps



Week 3... 4...

We did it. Every week, we picked the next most important scenarios and made it work.

Thanks, **Guardian Agiles!!!**

Remember:

- early value
- backlogs
- swarm
- scenarios
- timeboxes
- source control
- reflect and improve

Good luck!

Often it wasn't what we originally planned to do at all.



What about you?
Could you use some
Guardian Agiles?



Readings

- *The Agile Samurai*. Jonathan Rasmussen. Pragmatic Bookshelf, 2010, especially Parts I and IV
- Jim Murphy's Agile For Startups slides
- Eric Ries' Startup Lessons Learned blog
- Mike Cohn's Succeeding with Agile blog
- Elisabeth Hendrickson's Agile Acid Test
- Intro to Extreme Programming