

```
// You'll need  
// com.google  
listRef.listAll  
.addOn  
prefixes.fo  
// All  
// You  
}  
it  
} }  
each { item  
the items  
}  
}
```

devfest



Scrum Lego City

 Google Developer Groups



```
Text('Section Title',  
  style: TextStyle(  
    color: Colors.blue[200],  
  ),  
),  
),  
),
```

devfest

```
s.star,  
r: Colors.blue[500],  
Text('23'),
```



Google Developer Groups
Pescara

The team



Lorenza
De
Berardinis



Cesare
De
Sanctis



Martina
Irsuti



Antonio
Villanova



Giorgia
Di
Placido



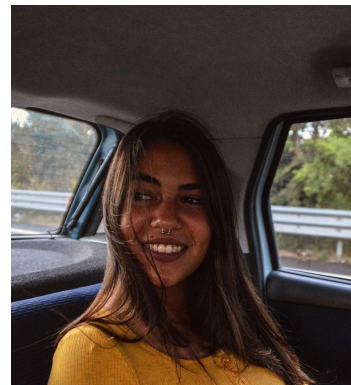
Gregorio
Palamà



Eleonora
Candeloro



Niko
Mennucci



Noemi
Surricchio

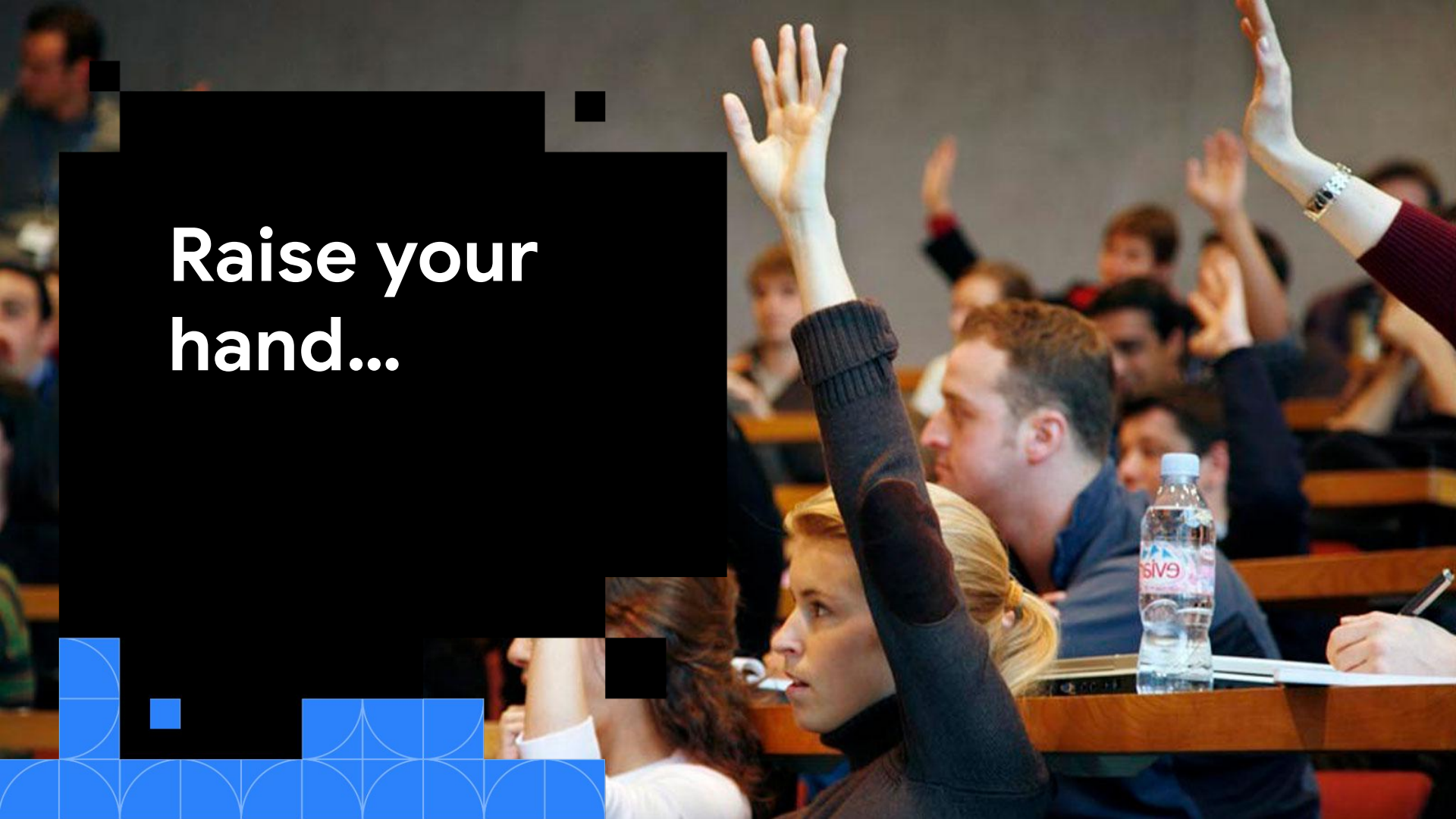


Antonio
Di Marino



Giorgio
Campea

**Raise your
hand...**



**...if you've
ever created
a software**

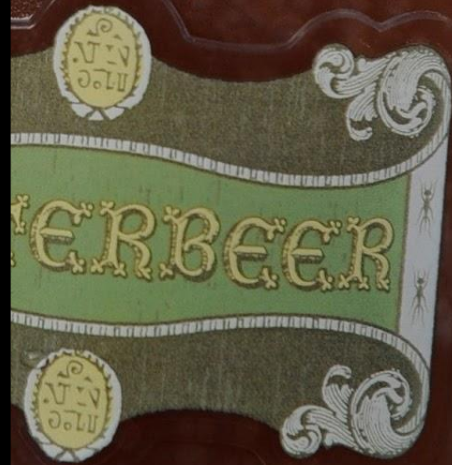
**...if you've ever
done a
complex
project
without any
planning**



A group of five young adults are gathered around a birthday cake with lit candles. They are all wearing colorful party hats and smiling. A large gift box with a red ribbon is visible in the foreground. The background shows a dimly lit room with warm lights and large windows.

**...if you've
ever
organized a
party**

**...if you've
ever had
butterbeer**

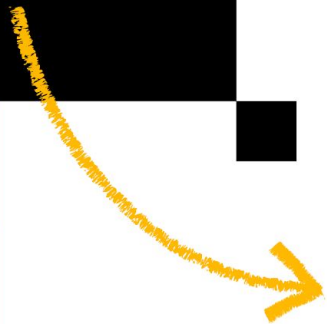


Expectations



```
Text('Section Title',  
style: TextStyle(  
color: Colors.green[200],  
),  
),  
),  
),  
s.star,  
r: Colors.green[500],  
Text('23'),
```

devfest

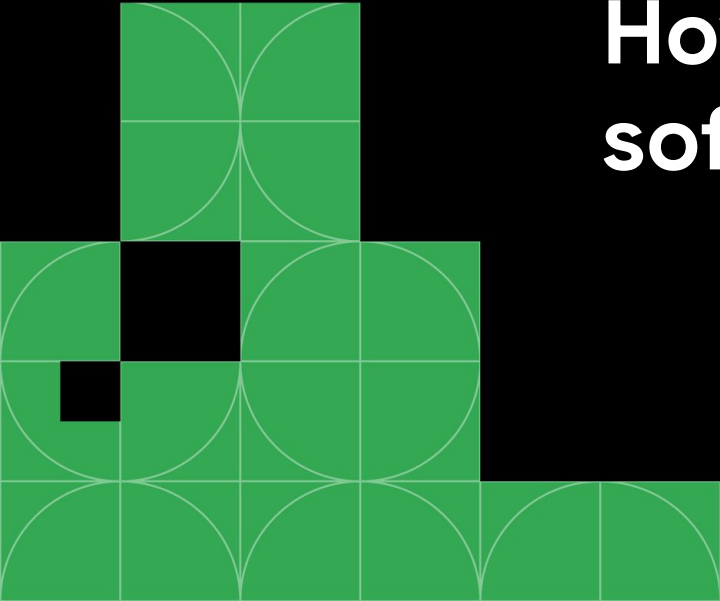


Software development methodologies

```
ext(  
  'Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.green[200],  
  ),  
),  
),
```

devfest

```
s.star,  
r: Colors.green[500],  
Text('23'),
```



How we break down the software creation process



Waterfall

- Very well defined, sequential phases. The output of one phase is the input of the next phase
- Analysis, Design, Development, Testing, Maintenance
- All phases of development are very precisely disciplined

Waterfall

- Estimating activities and anticipating problems in subsequent phases becomes very complex
- Excessively rigid
- Changes in requirements in the middle of the project are not allowed

Waterfall

- Easy to monitor
- Easy to find out errors or problems

**When?
Examples?**



Waterfall

- When the project has a well-defined final objective
- When no budget or time limits

Waterfall

- Building an house
- Building a car
- Organizing a conference
- ...

Spiral

- The phases repeat cyclically
- Identification of objectives, Risk assessment, Development, Planning

Spiral

- More expensive than the Cascade model. The management process is more complex
- It does not eliminate the risk of the proliferation of defects

Spiral

- It's easier to manage risk
- Estimates are simpler, because you work, from time to time, on increments and not on the entire project
- Changes in requirements can be easily absorbed



**When?
Examples?**

Spiral

- When releases are required to be frequent
- When creation of a prototype is applicable
- When we have a medium to high-risk project
- When requirements are unclear and complex

Spiral

- Building an house
- Building a car
- Organizing a conference
- ...

Agile

- Subsequent increments, obtained through iterations or sprints
- Involvement of the customer and stakeholders directly in the development process
- Automation of tests and operations that can be automated

Agile

- Focus on quality. Working software is the main measure of progress
- Team culture, face-to-face communication, daily standup
- High awareness of the team towards the analysis of how to be more efficient

Agile

- It becomes difficult to have an overview
- Care must be taken to avoid inserting new processes into the current iteration
- It forces you to automate at least the tests
- Excessive preparation and planning

Agile

- Stakeholders are not always sufficiently represented
- It can lead to developer burnout more easily than other models

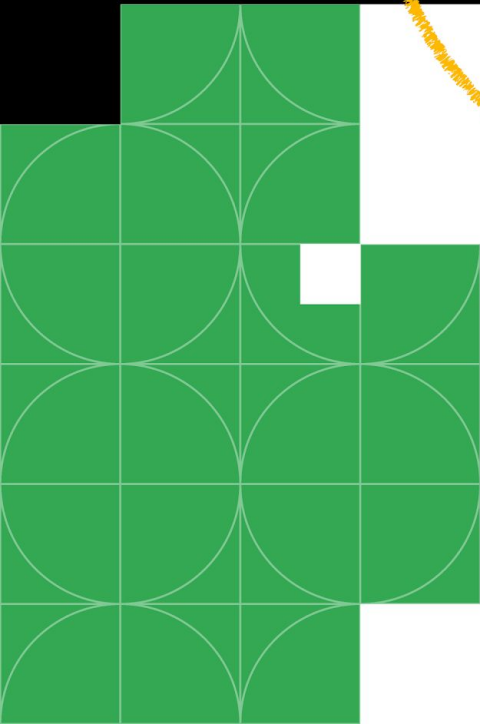
```
ext(
  'Section Title',
  style: TextStyle(
    color: Colors.green[200],
  ),
),
),
```

devfest

```
s.star,
r: Colors.green[500],
Text('23'),
```

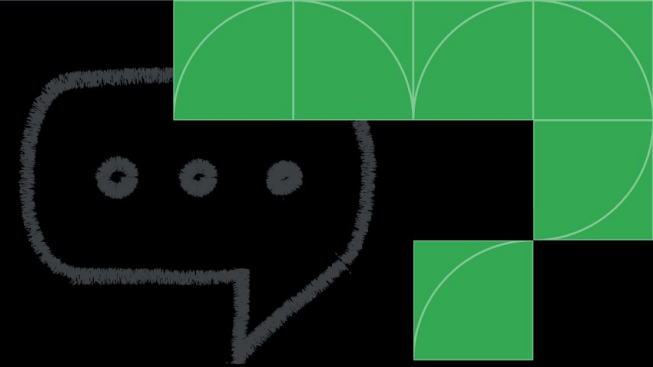


Agile manifesto

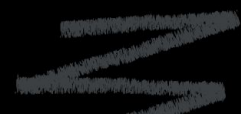
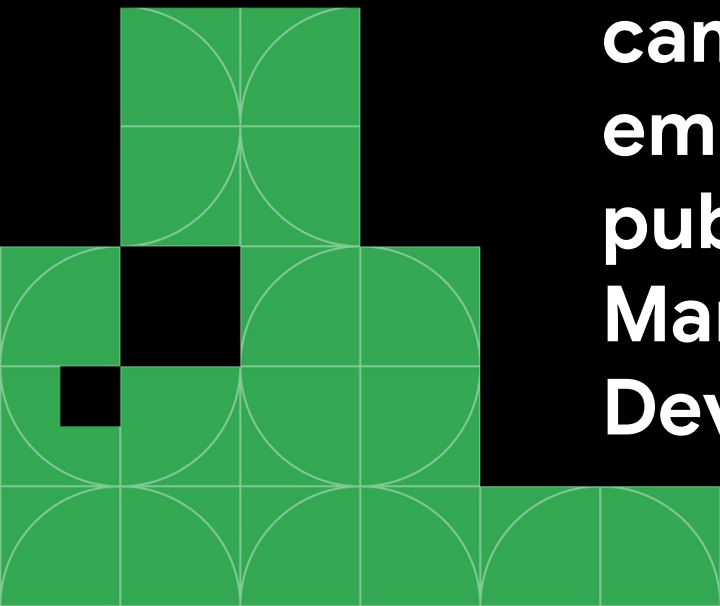


```
Text('Simple Statement or URL',  
style: TextStyle(  
color: Colors.green[200],  
),  
),  
),  
s.star,  
r: Colors.green[500],  
Text('23'),
```

devfest



In 2001, 17 software developers came together to discuss emerging agile methodologies, publishing what they called the Manifesto for Agile Software Development





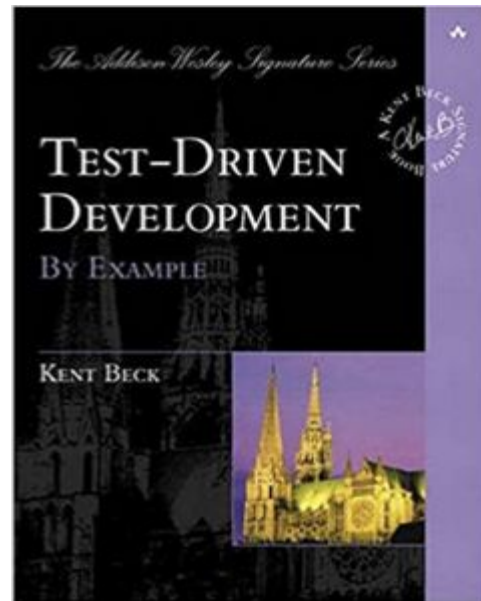
 Google Developer Groups
Pescara

JUV 1-0 BAR





Kent Beck





Kent Beck



Kent Beck



Ward Cunningham





Kent Beck

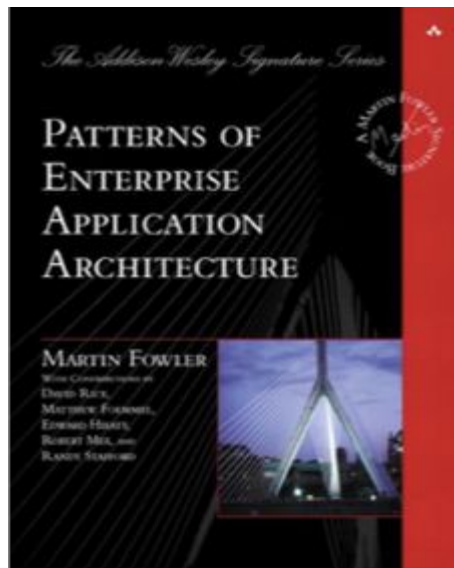
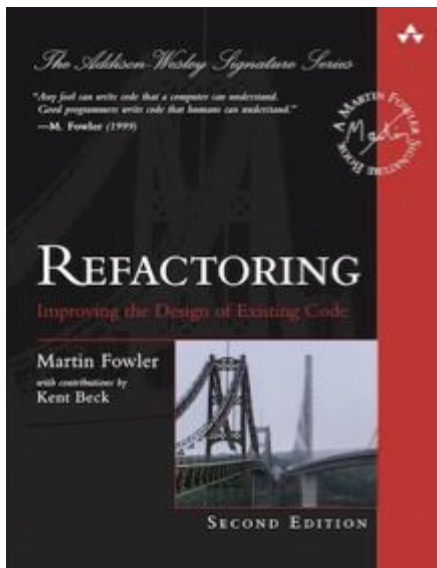


Ward Cunningham





SO LILLO



Martin Fowler



Kent Beck



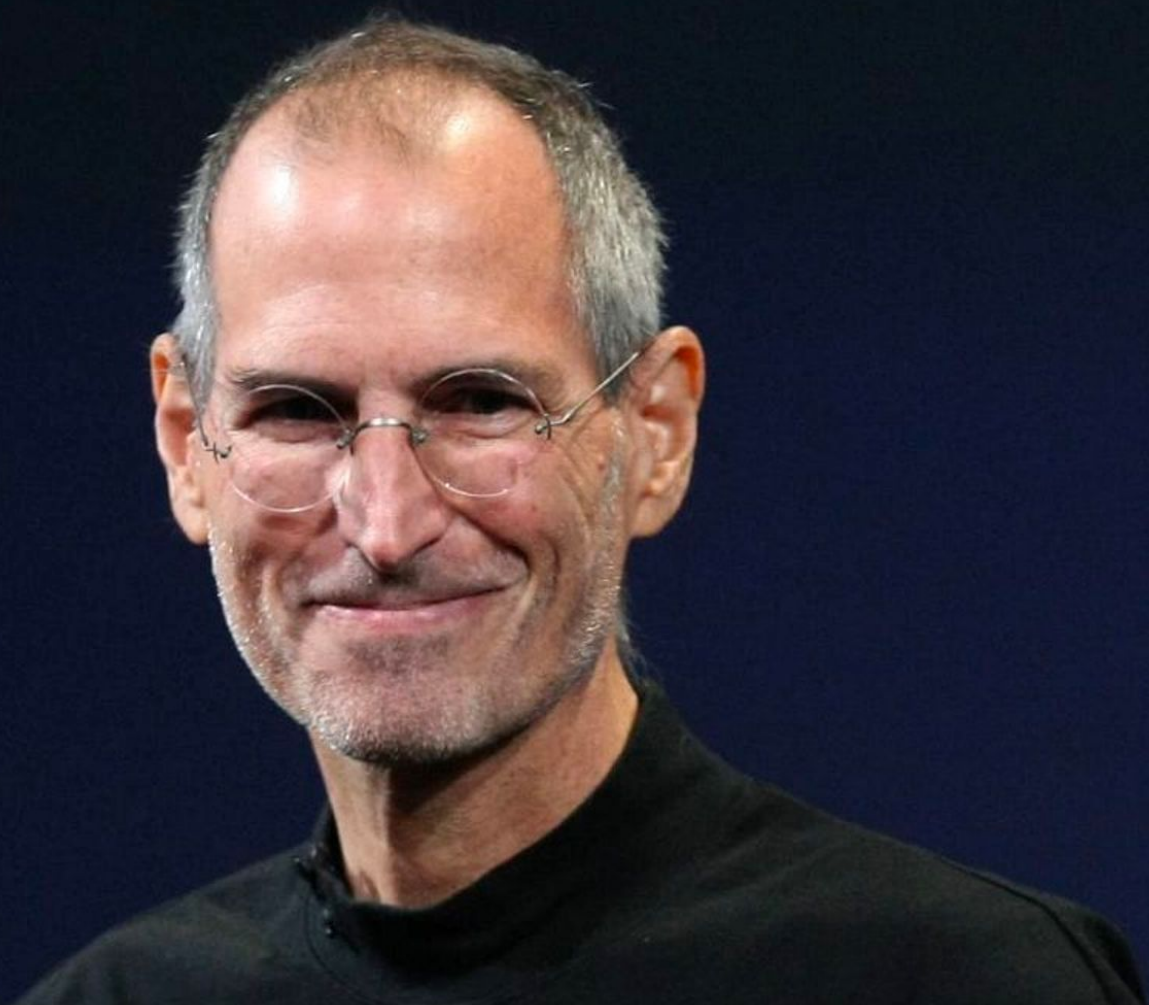
Ward Cunningham



Martin Fowler

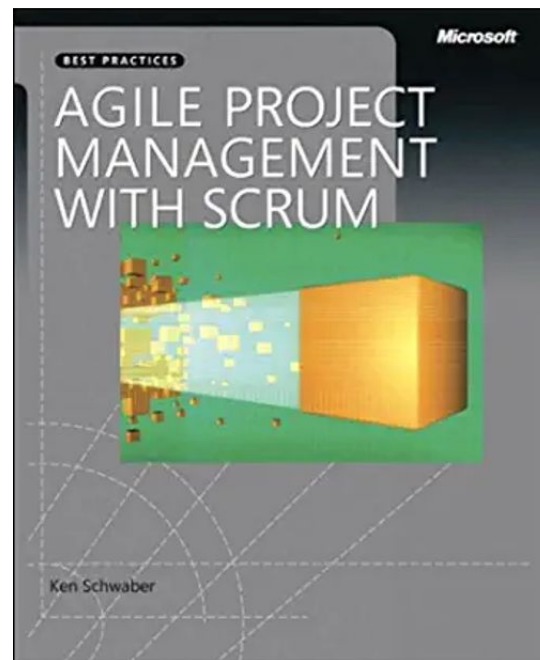


 Google Developer Groups
Pescara





Ken Schwaber





Ken Schwaber





Robert C. Martin



Ken Schwaber

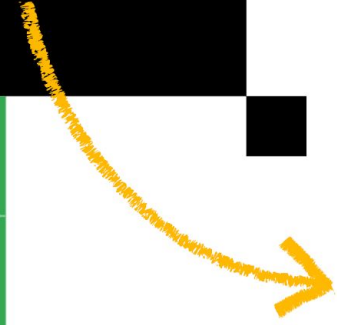


Robert C. Martin

```
Text(
  'Section Title',
  style: TextStyle(
    color: Colors.green[200],
  ),
),
),
```

devfest

```
s.star,
r: Colors.green[500],
Text('23'),
```



Google Developer Groups

Pescara

Agile manifesto

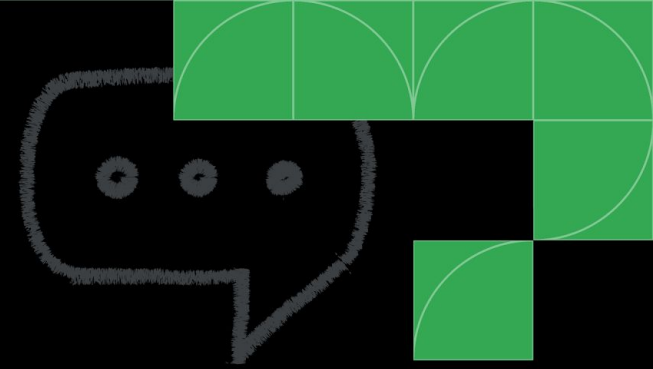


Values

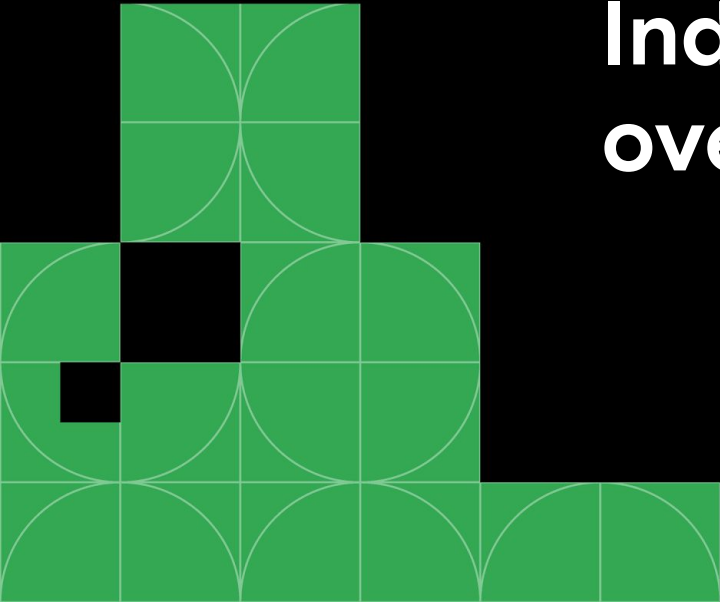

```
text(
  'Simple Statement or URL',
  style: TextStyle(
    color: Colors.green[200],
  ),
),
),
```

devfest

```
s.star,
r: Colors.green[500],
Text('23'),
```



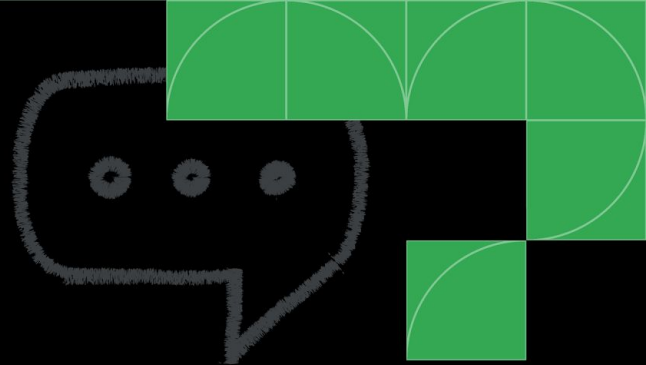
Individuals and interactions over processes and tools



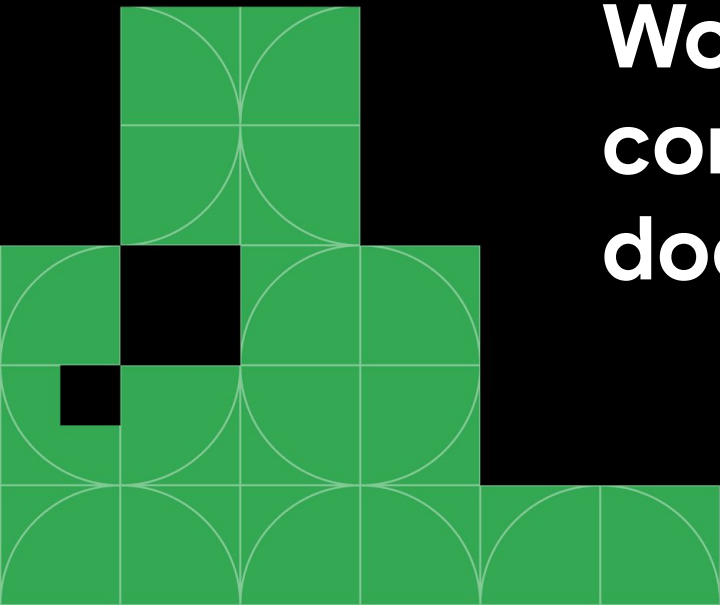
```
Text('Simple Statement or URL',  
style: TextStyle(  
  color: Colors.green[200],  
),  
),  
)
```

devfest

```
s.star,  
r: Colors.green[500],  
Text('23'),
```



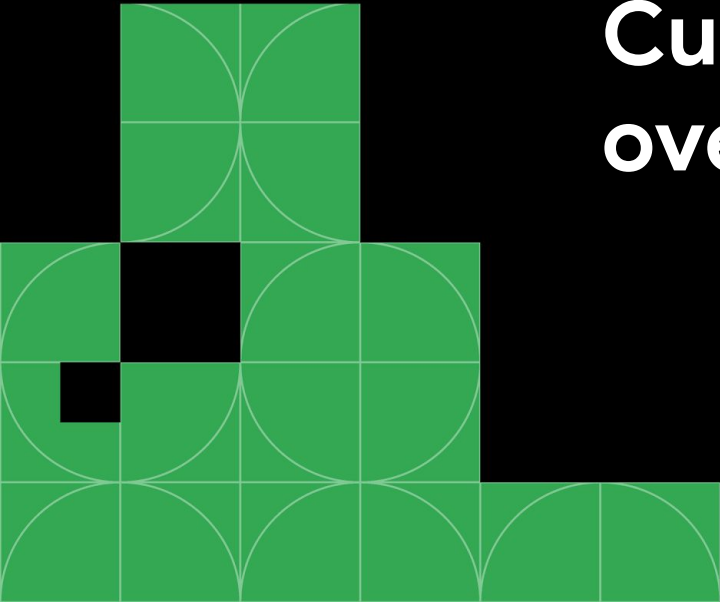
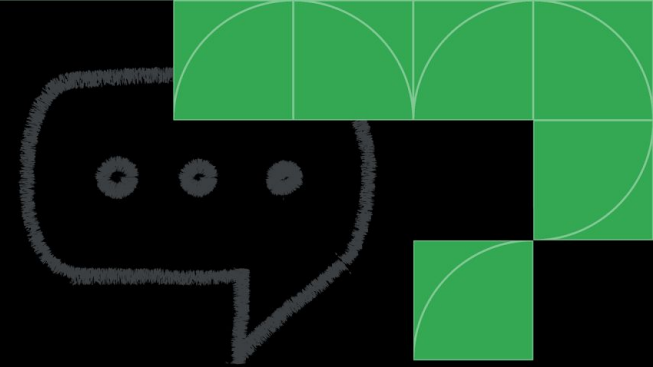
Working software over comprehensive documentation



```
Text('Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.green[200],  
  ),  
)  
)  
)
```

devfest

```
s.star,  
r: Colors.green[500],  
Text('23'),
```



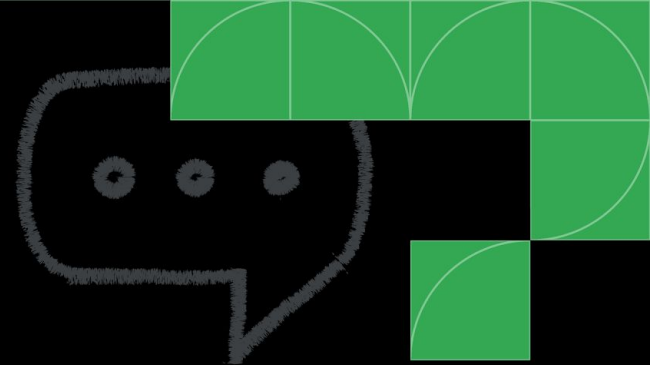
Customer collaboration over contract negotiation



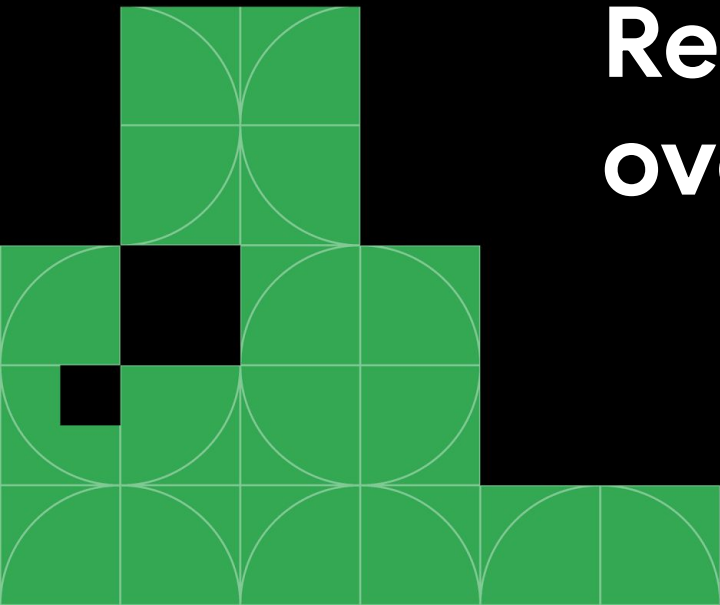
```
Text('Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.green[200],  
  ),  
),  
,
```

devfest

```
s.star,  
r: Colors.green[500],  
Text('23'),
```



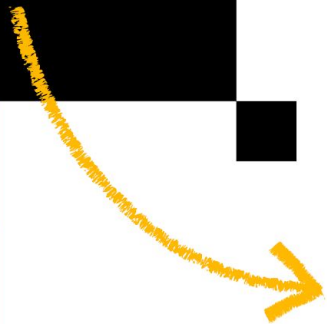
Responding to change over following a plan



```
Text(
  'Section Title',
  style: TextStyle(
    color: Colors.green[200],
  ),
),
),
```

devfest

```
s.star,
r: Colors.green[500],
Text('23'),
```



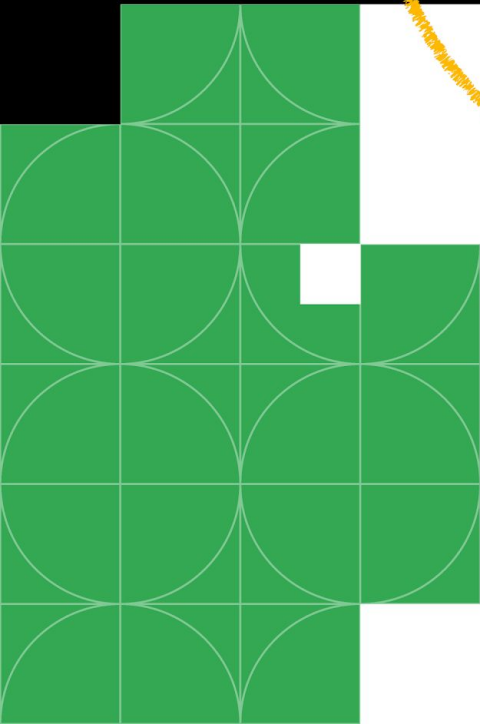
Google Developer Groups

Pescara

Agile manifesto

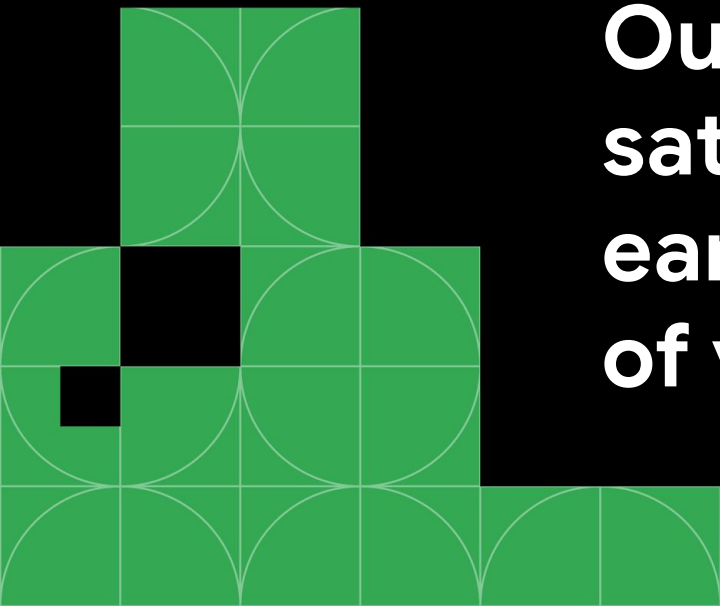
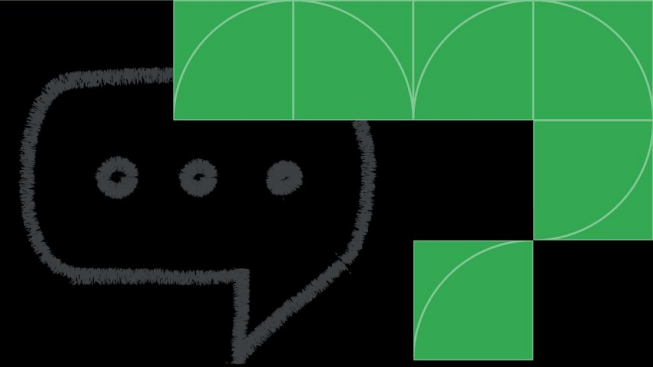
•

Principles



```
Text('Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.green[200],  
  ),  
),  
),  
s.star,  
r: Colors.green[500],  
Text('23'),
```

devfest

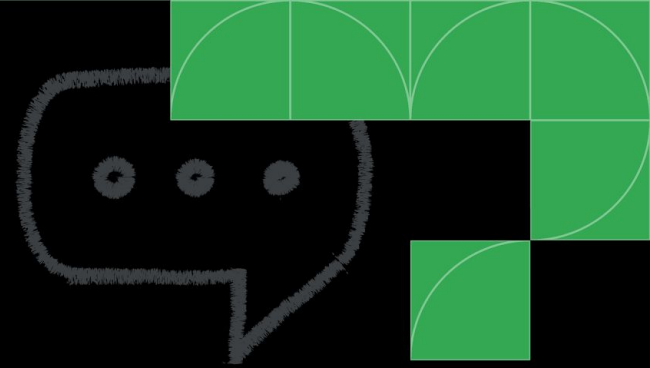


Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.



```
Text('Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.green[200],  
  ),  
),  
),  
s.star,  
r: Colors.green[500],  
Text('23'),
```

devfest



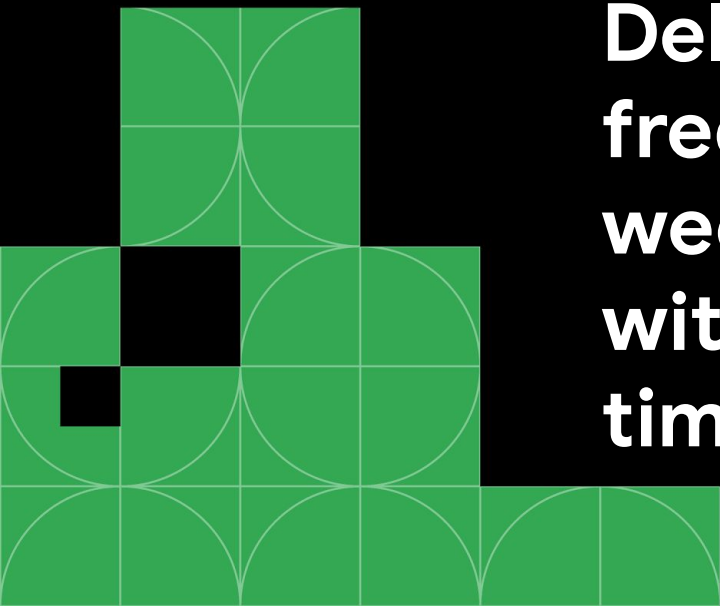
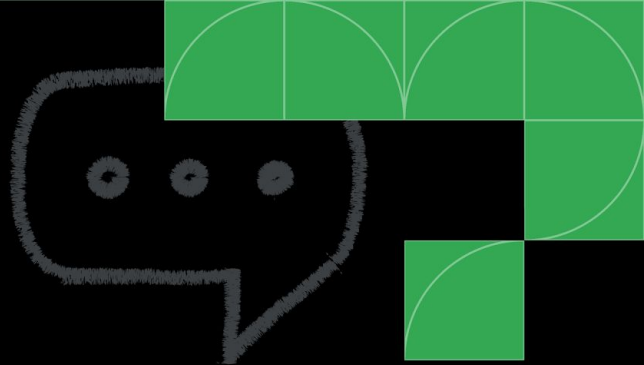
**Welcome changing requirements,
even late in development. Agile
processes harness change for the
customer's competitive
advantage.**



```
Text('Simple Statement or URL',  
style: TextStyle(  
color: Colors.green[200],  
),  
),  
),
```

devfest

```
s.star,  
r: Colors.green[500],  
Text('23'),
```

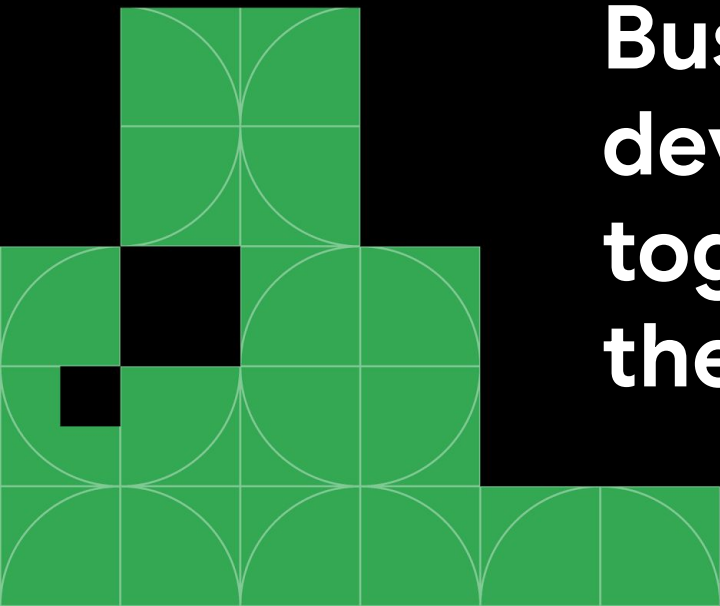
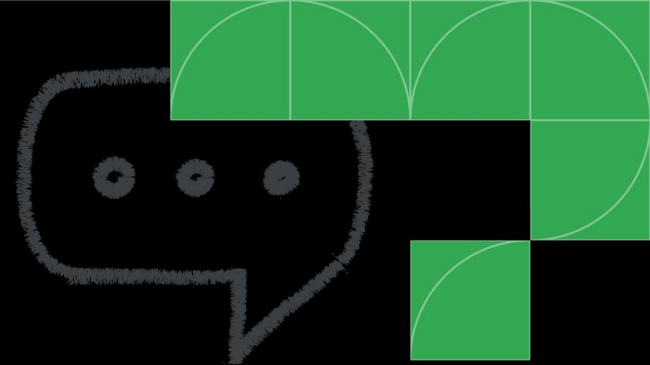


Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.




```
Text('Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.green[200],  
  ),  
)  
)  
)  
  
s.star,  
r: Colors.green[500],  
Text('23'),
```

devfest



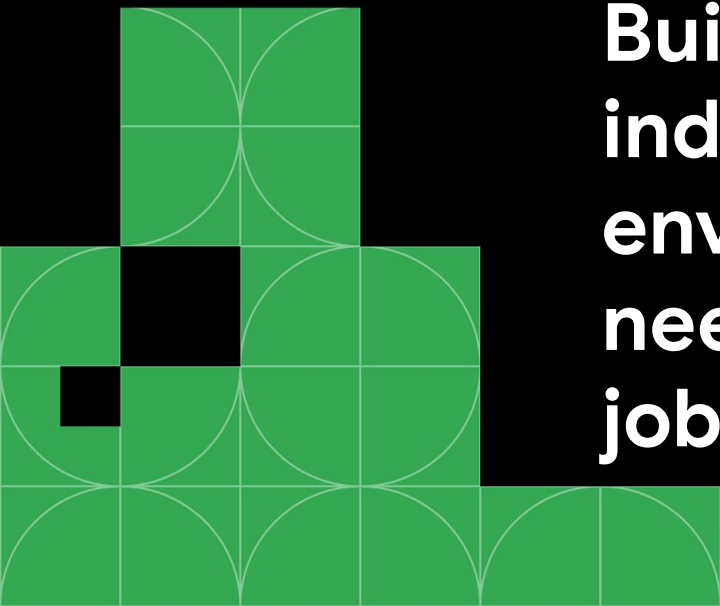
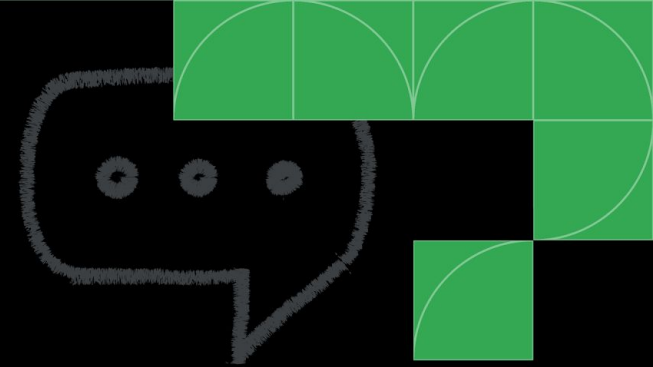
**Business people and
developers must work
together daily throughout
the project.**



```
Text(
  'Simple Statement or URL',
  style: TextStyle(
    color: Colors.green[200],
  ),
),
),
```

devfest

```
s.star,
r: Colors.green[500],
Text('23'),
```



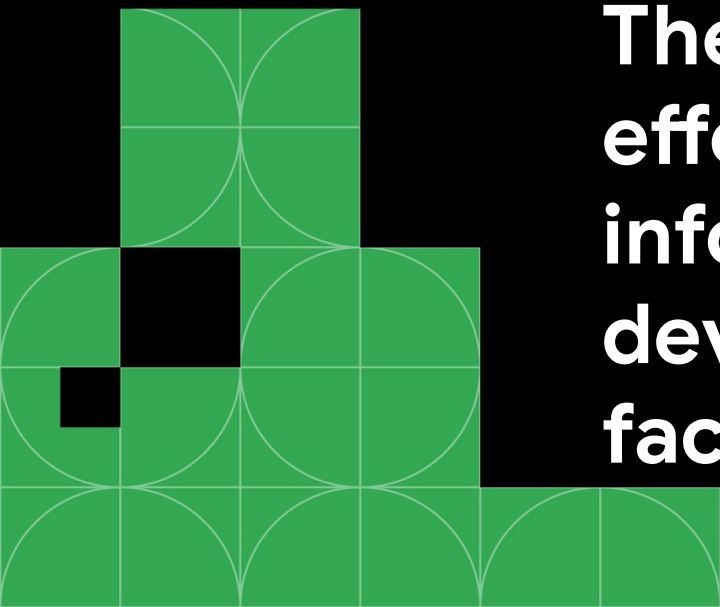
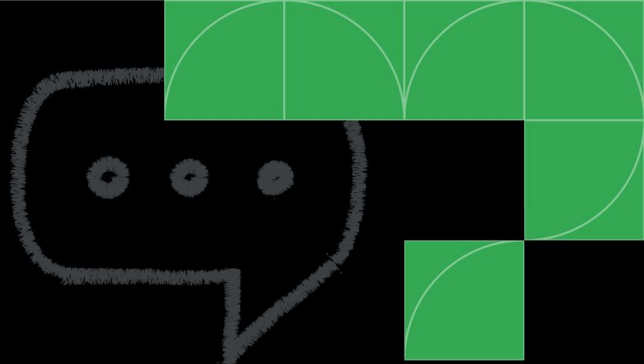
Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.




```
Text('Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.green[200],  
  ),  
)  
)
```

devfest

```
  s.star,  
  r: Colors.green[500],  
  Text('23'),
```



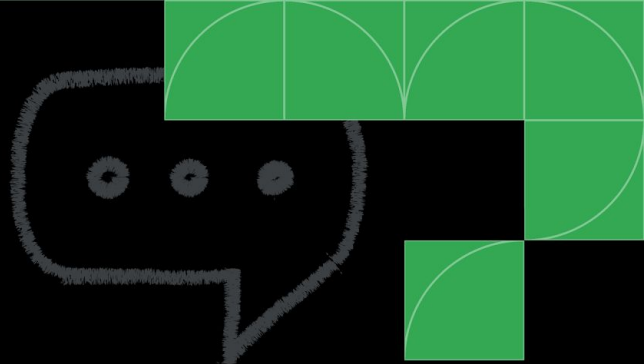
The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.



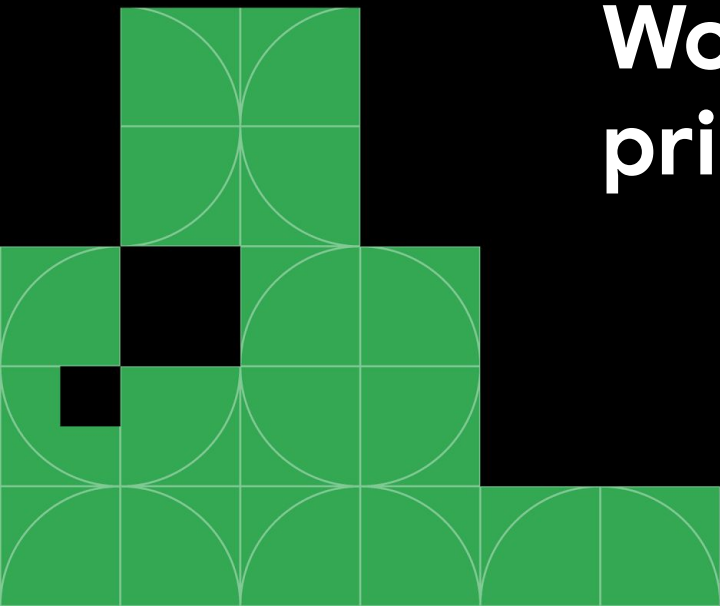
```
Text('Simple Statement or URL',  
style: TextStyle(  
color: Colors.green[200],  
),  
),  
),  
),
```

devfest

```
s.star,  
r: Colors.green[500],  
Text('23'),
```



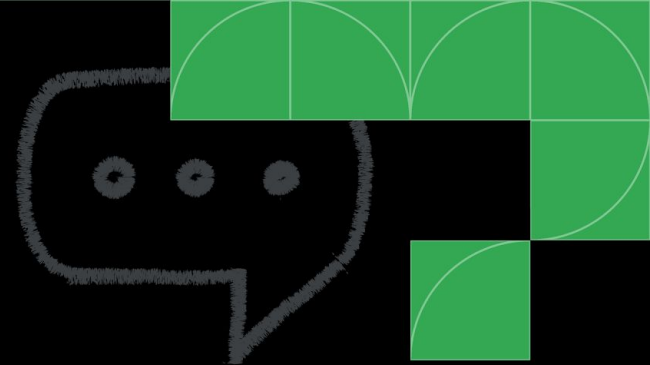
**Working software is the
primary measure of progress.**



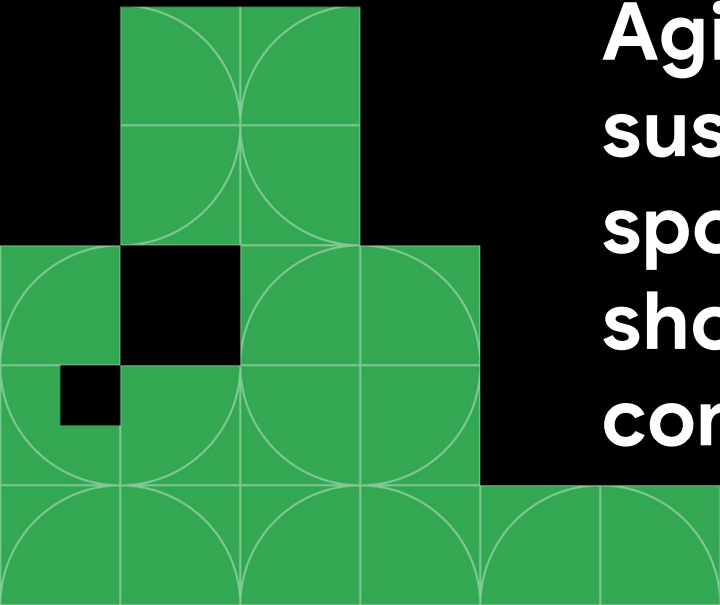
```
Text('Simple Statement or URL',  
    style: TextStyle(  
      color: Colors.green[200],  
    ),  
),  
),
```

devfest

```
s.star,  
r: Colors.green[500],  
Text('23'),
```

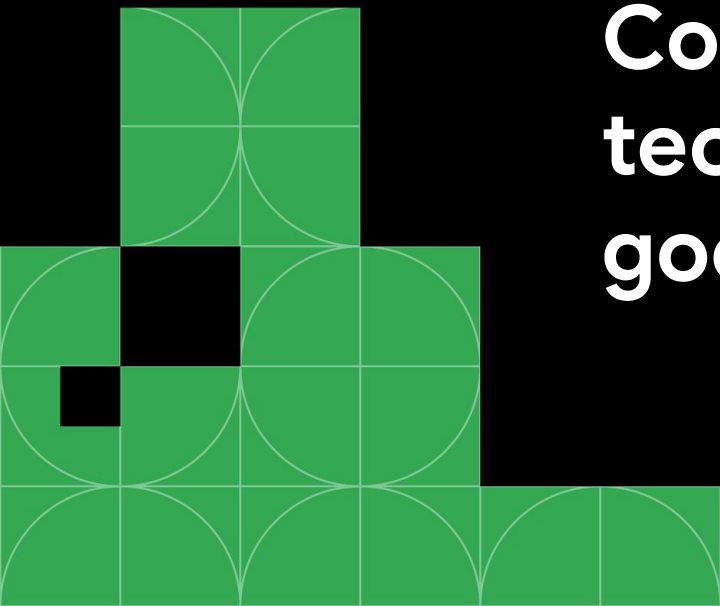
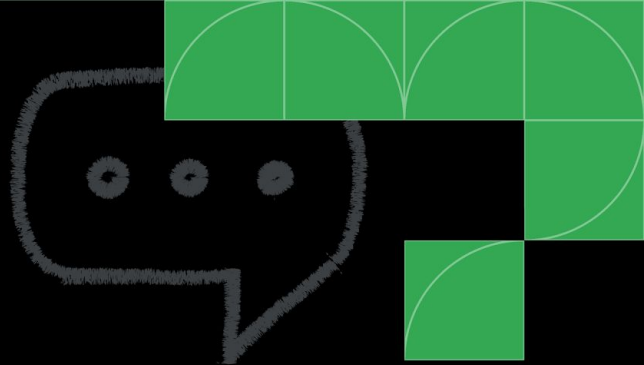


Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.



```
Text('Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.green[200],  
  ),  
)  
)  
)  
  
s.star,  
r: Colors.green[500],  
Text('23'),
```

devfest



**Continuous attention to
technical excellence and
good design enhances agility.**



```
Text('Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.green[200],  
  ),  
)  
)  
)  
  
s.star,  
r: Colors.green[500],  
  
Text('23'),
```

devfest



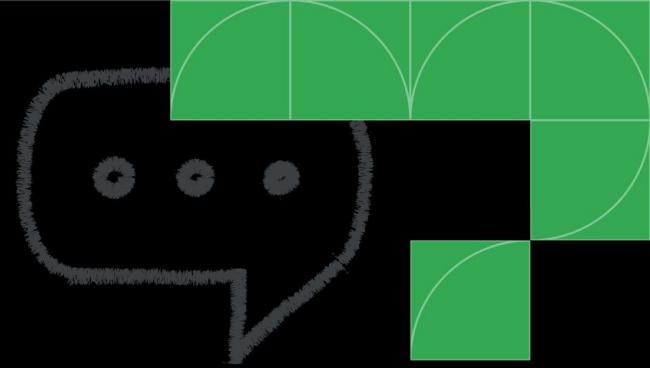
**Simplicity--the art of
maximizing the amount of
work not done--is essential.**



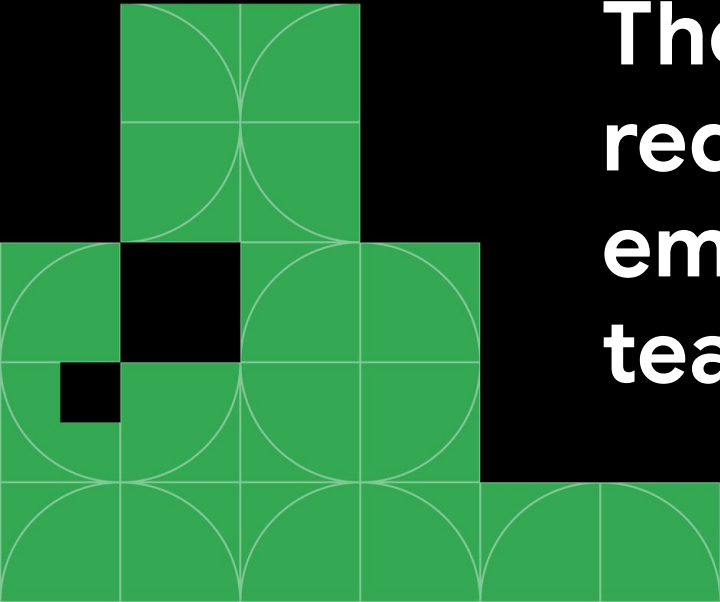
```
Text('Simple Statement or URL',  
style: TextStyle(  
color: Colors.green[200],  
),  
),  
),
```

devfest

```
s.star,  
r: Colors.green[500],  
Text('23'),
```



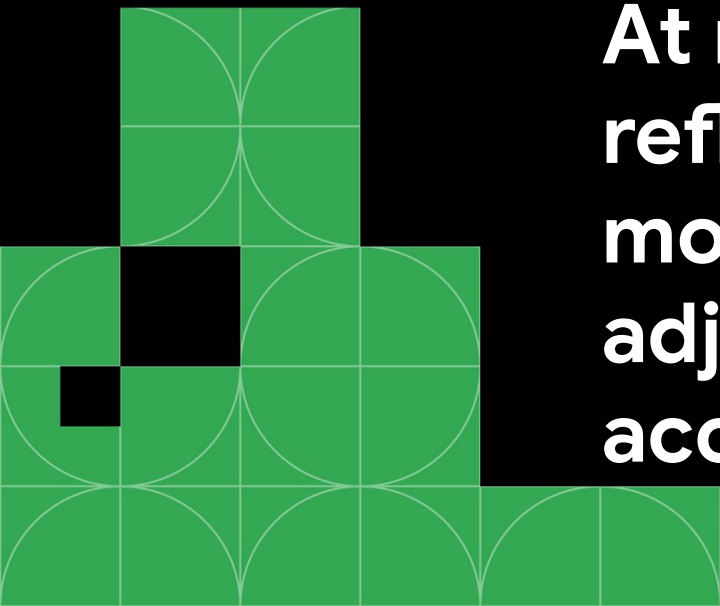
**The best architectures,
requirements, and designs
emerge from self-organizing
teams.**




```
Text('Simple Statement or URL',  
style: TextStyle(  
  color: Colors.green[200],  
),  
),  
),
```

devfest

```
s.star,  
r: Colors.green[500],  
Text('23'),
```



At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.



When? Examples?



Agile

- When we need to easily and quickly adapt to changes
- When we need high quality
- When we want to minimize risk
- When we want greater satisfaction
- When we want predictable delivery dates

Agile

- Building an house
- Building a car
- Organizing a conference
- ...

Methodologies

- Scrum
- Kanban
- eXtreme Programming (XP)
- Scaled Agile Framework (SAFe)
- Lean Software Development

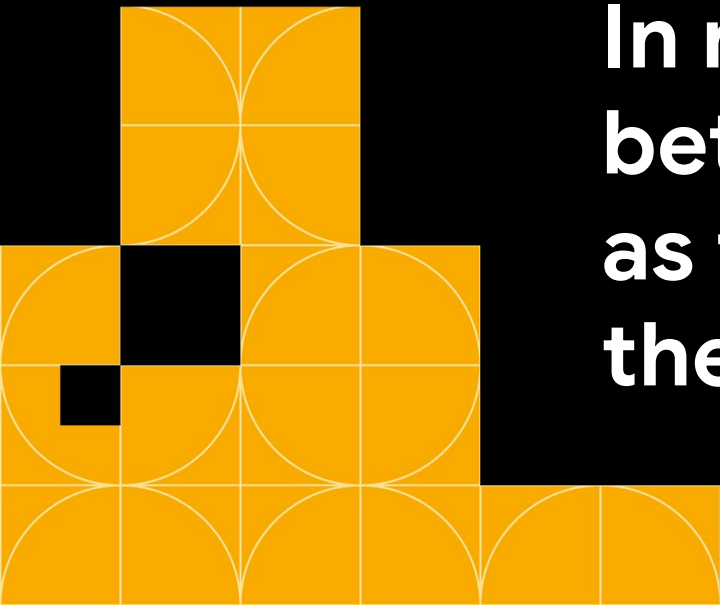
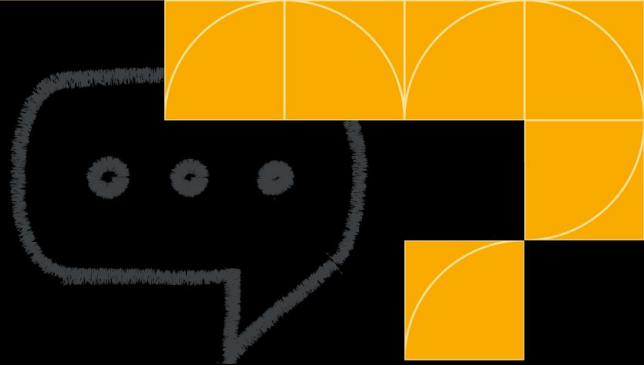
The name
comes
from rugby




```
Text(
  'Simple Statement or URL',
  style: TextStyle(
    color: Colors.yellow[200],
  ),
),
),
```

devfest

```
s.star,
r: Colors.yellow[500],
Text('23'),
```



In rugby, the ball is passed between players of a team as they move as a unit on the field



```
ext(  
  'Section Title',  
  style: TextStyle(  
    color: Colors.yellow[200],  
  ),  
),  
)
```

devfest

```
s.star,  
r: Colors.yellow[500],  
Text('23'),
```



 Google Developer Groups
Pescara

Scrum

-

Roles

Product Owner

- Responsible for maintaining the product backlog
- Must understand user needs and act as a spokesperson with the team and executive stakeholders
- Informs the team about the next essential deliverables to be achieved
- Decides whether the product is ready to be distributed or not, often prioritizing frequent distribution



Pescara

Scrum Master

- Manages the various Scrum events, a kind of project manager and facilitator
- Responsible for facilitating daily stand-up meetings and leading sprint planning, review and retrospective meetings

Scrum Team

- Anyone participating in the sprint
- The members of the team should know how to organize and collaborate, in order to achieve the Scrum goal of continuous improvement

```
ext(  
  'Section Title',  
  style: TextStyle(  
    color: Colors.yellow[200],  
  ),  
),  
),  
),  
s.star,  
r: Colors.yellow[500],  
Text('23'),
```

devfest



Google Developer Groups

Pescara

Scrum • Artifacts

Product backlog

- The main list of activities to be carried out
- Must be evaluated by the project manager or product owner
- The team can choose which product backlog tasks to work on during a Scrum sprint
- Product owners should frequently reorder and update the product backlog based on new information from customers, the market, or the project team

Sprint backlog

- The collection of activities or products that the team has worked on during a Scrum sprint
- The activities are chosen from the product backlog during the sprint planning session and moved into the sprint planning project
- The team may not complete everything in the backlog during every sprint, but it is unlikely to add tasks to the backlog mid-sprint

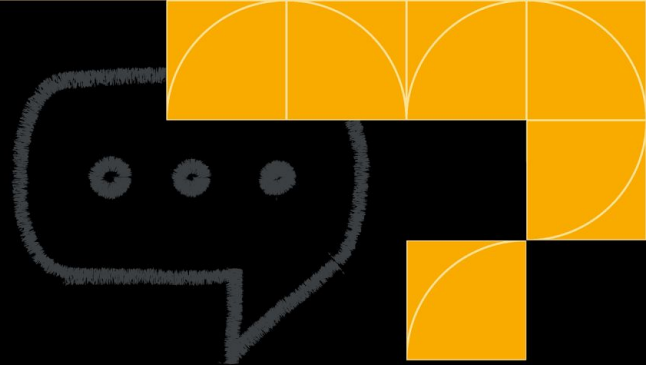
Product increase

- What will be delivered at the end of a sprint
- Can be a new product or a new feature, an improvement or a bug fix, or anything else depending on the nature of the team
- Presented during the sprint review. At that point, it will launch or not based on what Scrum stakeholders think about it and whether it is “Done”

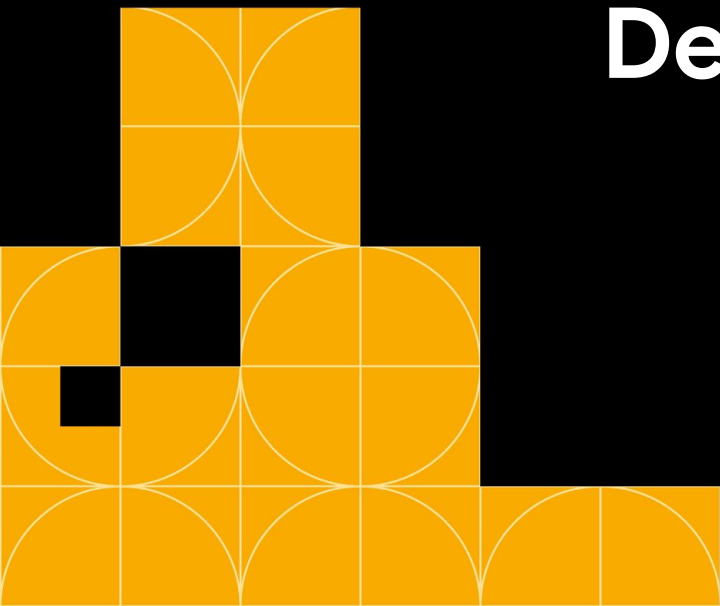
```
Text('Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.yellow[200],  
  ),  
),  
,
```

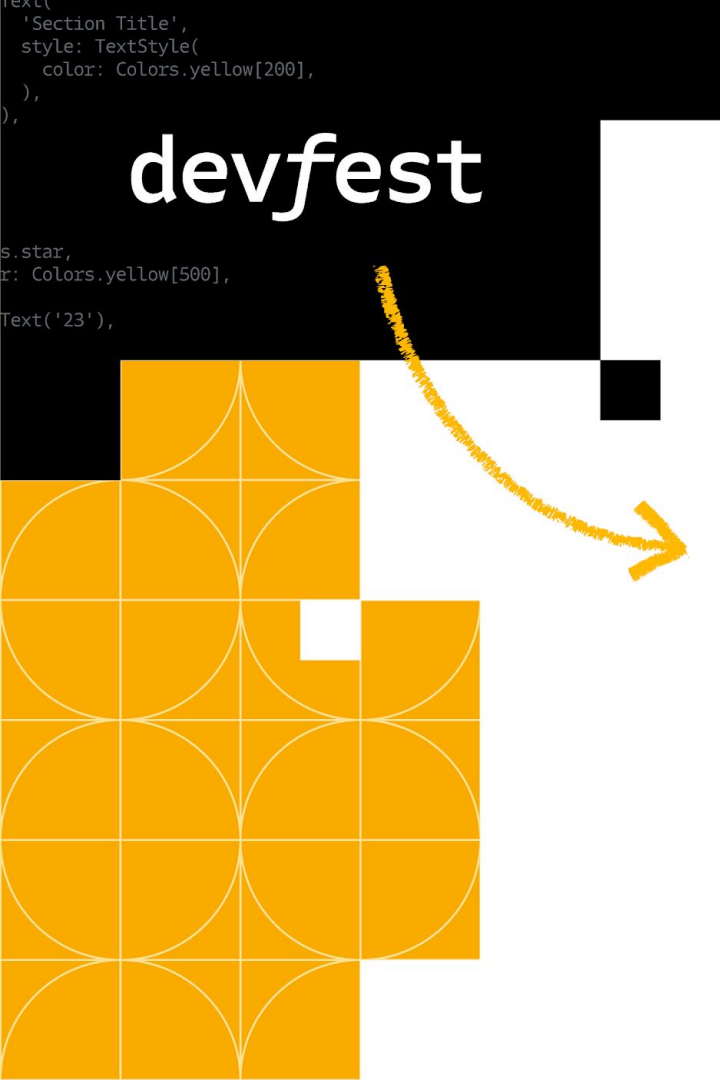
devfest

```
s.star,  
r: Colors.yellow[500],  
Text('23'),
```



Definition of Done





Scrum • Ceremonies

Sprint planning

- Initiates the sprint by laying out the work to be performed for the sprint
- The resulting plan is created by the collaborative work of the entire Scrum team

Developers, Scrum Master, Product Owner

Sprint planning

One hour per week of iteration

Held at the beginning of a sprint

Daily scrum

- The team inspects progress toward the Sprint Goal and adapt the Sprint Backlog as necessary, adjusting the upcoming planned work

Developers, Scrum Master, Product Owner

Daily scrum

Usually, developers answers at these questions:

- What did you do yesterday?
- What will you do today?
- Is anything blocking your process?

Daily scrum

No more than 15 minutes. It must be done standing!

Held once per day, usually at the beginning of the day

Sprint review

- The Product Owner and Stakeholders inspect the outcome of the sprint and determine future adaptations
- The team presents the results of their work to key stakeholders, and progress toward the Product Goal is discussed

Developers, Scrum Master, Product Owner, Stakeholders

Sprint review

45 minutes per week of iteration

Held at the end of a sprint

Sprint retrospective

- The team plans ways to increase quality and effectiveness

Developers, Scrum Master, Product Owner (optional)

Sprint retrospective

45 minutes per week of iteration

Held at the end of a sprint

```
Text('Section Title',  
style: TextStyle(  
color: Colors.yellow[200],  
),  
),  
),  
s.star,  
r: Colors.yellow[500],  
Text('23'),
```

devfest



Scrum

•

**Techniques &
more...**

User Story

- An informal, general explanation of a software feature written from the perspective of the end user
- Articulates how a software feature will provide value to the customer

User Story

As a student, I want to cross the river that flows between Hogswarts and Hogsmeade without getting any risk

Estimations

- T-Shirt sizes
- Fibonacci
- Dog breed

Often done using Planning Poker Decks

Story points

- Units of measure for expressing an estimate of the overall effort required to fully implement a product backlog item or any other piece of work
- The team assigns story points relative to work complexity, the amount of work, and risk or uncertainty
- Are assigned to more effectively break down work into smaller pieces, so they can address uncertainty



**You will build
Hogsmeade...**

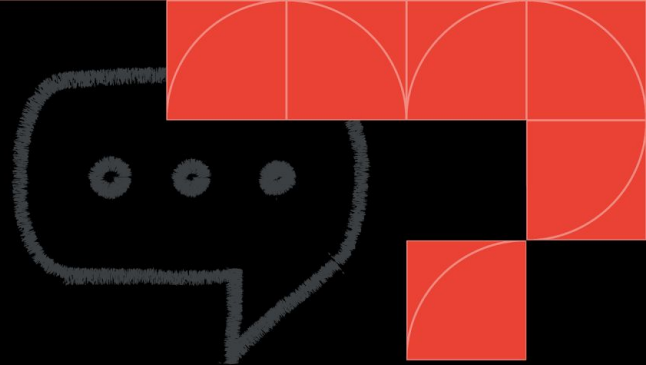


**...using
Legos and
Scrum**

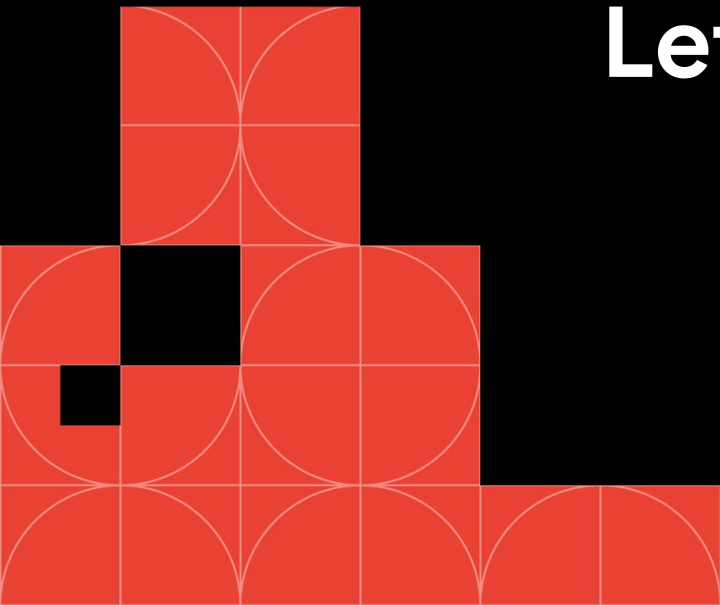
```
Text('Simple Statement or URL',  
style: TextStyle(  
  color: Colors.red[200],  
),  
),  
),
```

devfest

```
s.star,  
r: Colors.red[500],  
Text('23'),
```



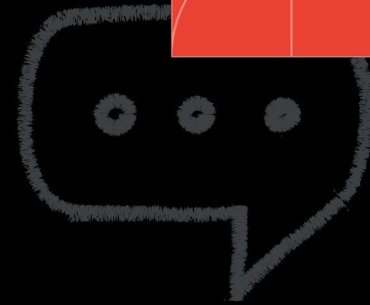
Let's make teams



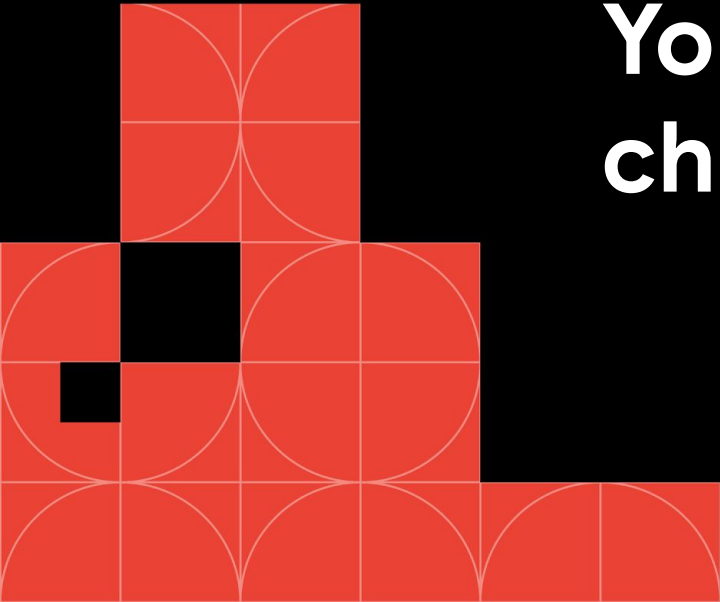
```
ext(  
  'Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.red[200],  
  ),  
),  
)
```

devfest

```
s.star,  
r: Colors.red[500],  
Text('23'),
```

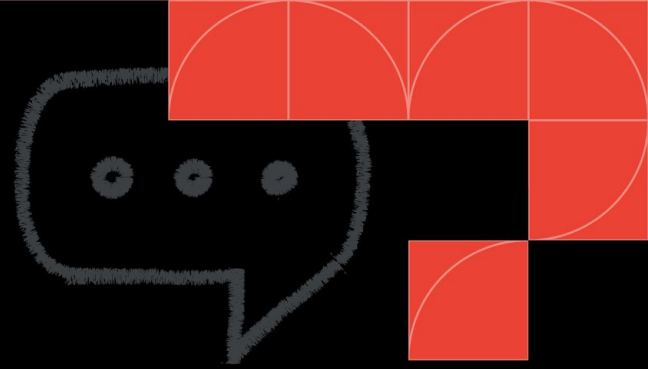


You have 3 minutes to choose a team name

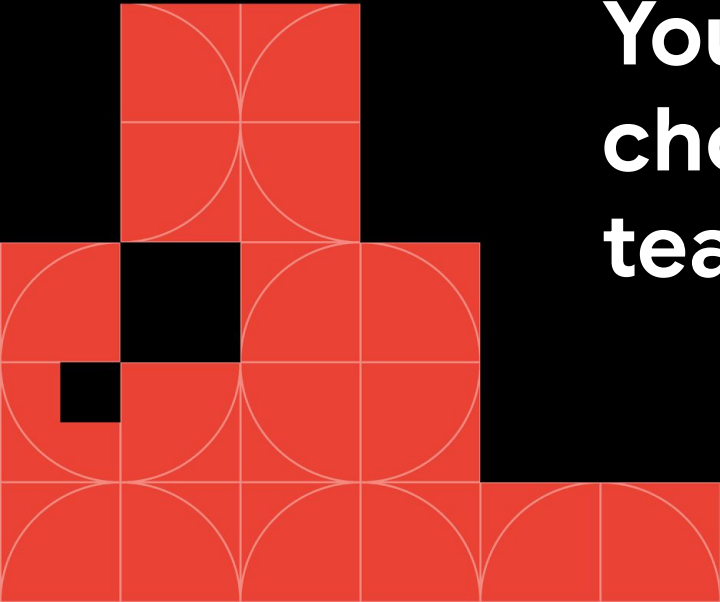


```
Text('Simple Statement or URL',  
      style: TextStyle(  
        color: Colors.red[200],  
      ),  
    ),  
  ),  
),  
s.star,  
r: Colors.red[500],  
Text('23'),
```

devfest



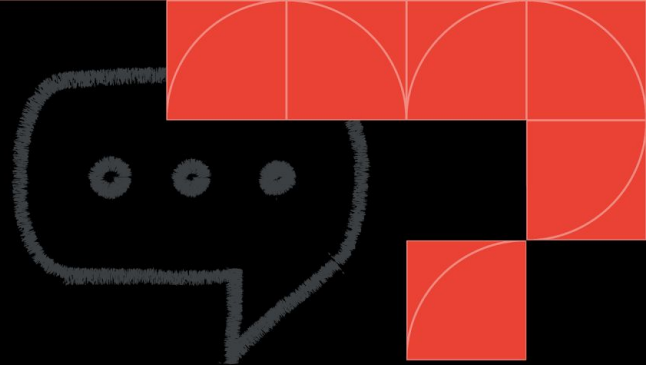
**You have 3 minutes to
choose a PO among the
team members**



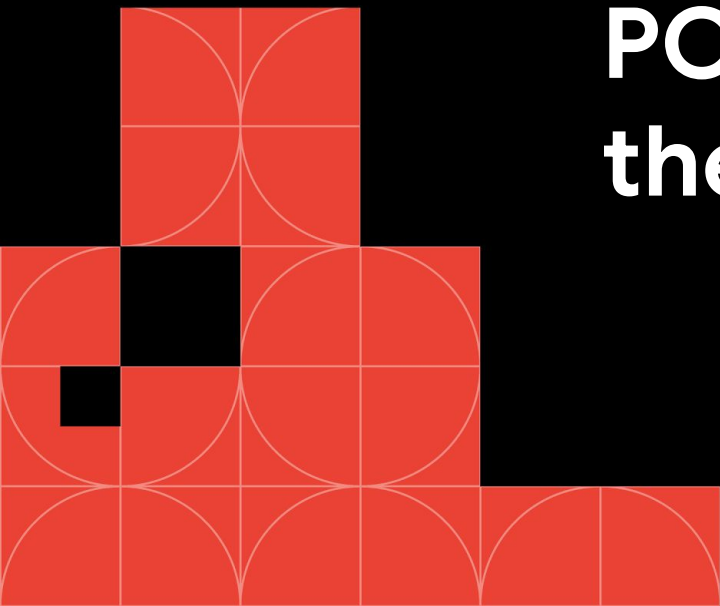
```
Text(
  'Simple Statement or URL',
  style: TextStyle(
    color: Colors.red[200],
  ),
),
),
```

devfest

```
s.star,
r: Colors.red[500],
Text('23'),
```

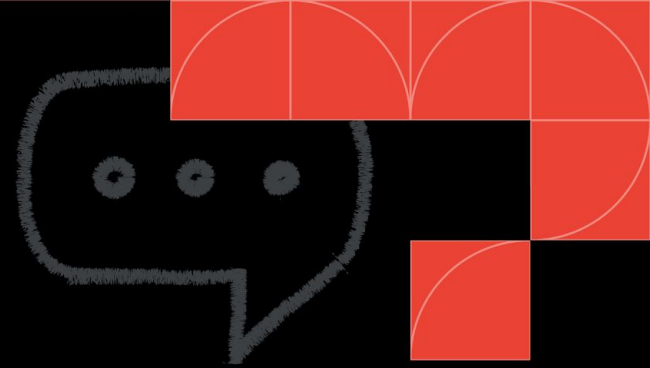


POs can't participate in the building

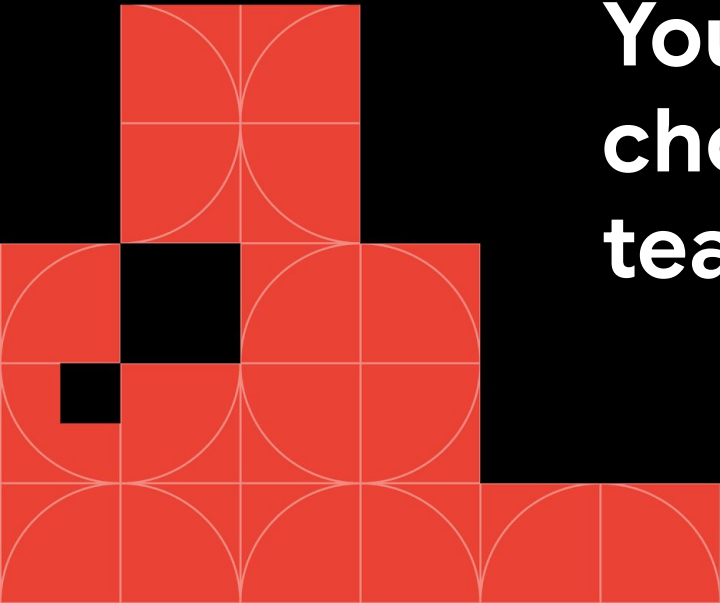



```
Text('Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.red[200],  
  ),  
),  
),  
s.star,  
r: Colors.red[500],  
Text('23'),
```

devfest



**You have 3 minutes to
choose a SM among the
team members**



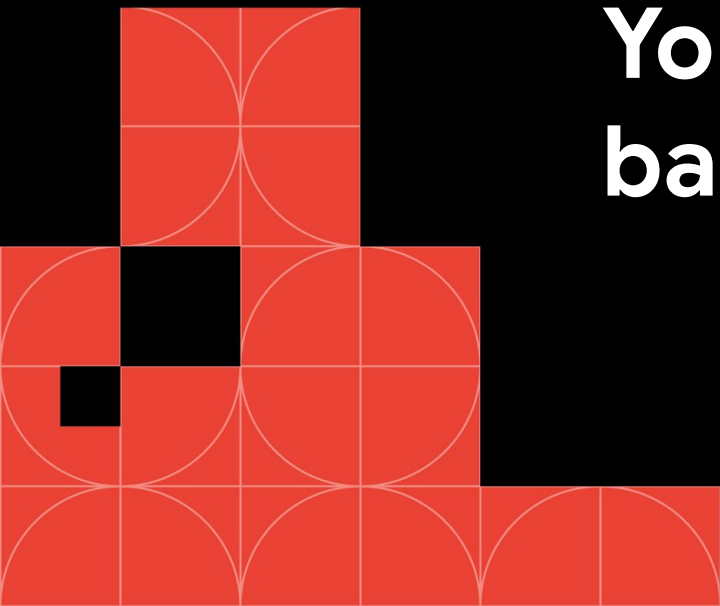
```
Text(
  'Simple Statement or URL',
  style: TextStyle(
    color: Colors.red[200],
  ),
),
),
```

devfest

```
s.star,
r: Colors.red[500],
Text('23'),
```



You will be given a backlog

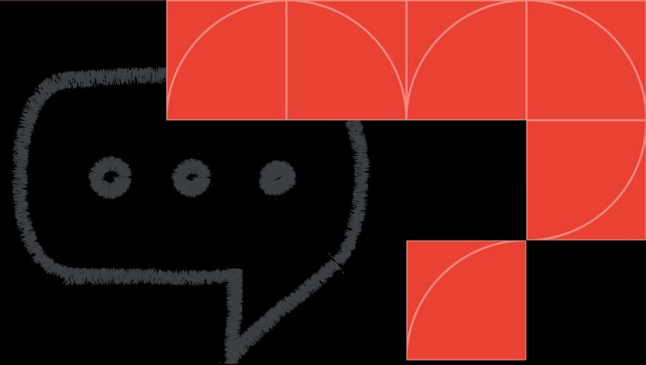


```
text(
  'Simple Statement or URL',
  style: TextStyle(
    color: Colors.red[200],
  ),
),
),
```

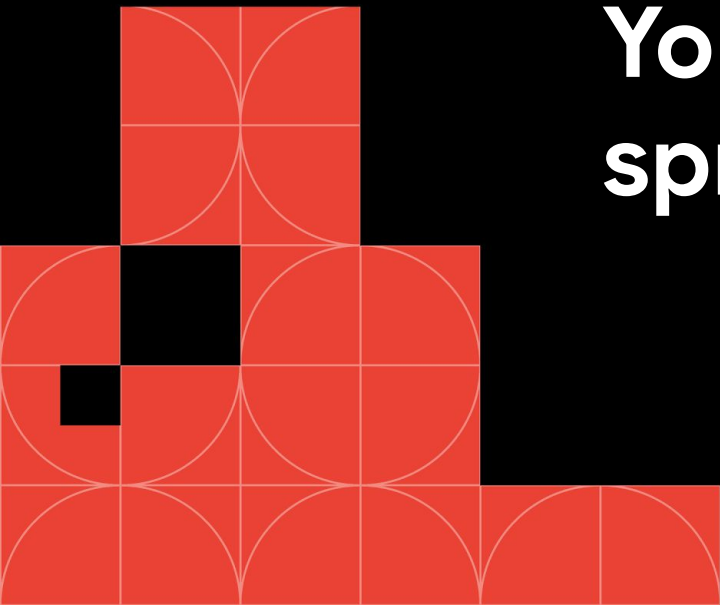
devfest

```
s.star,
r: Colors.red[500],
```

```
Text('23'),
```

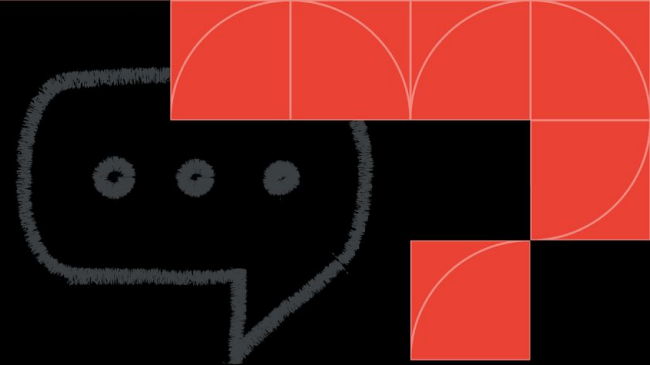


You will perform 4 sprints

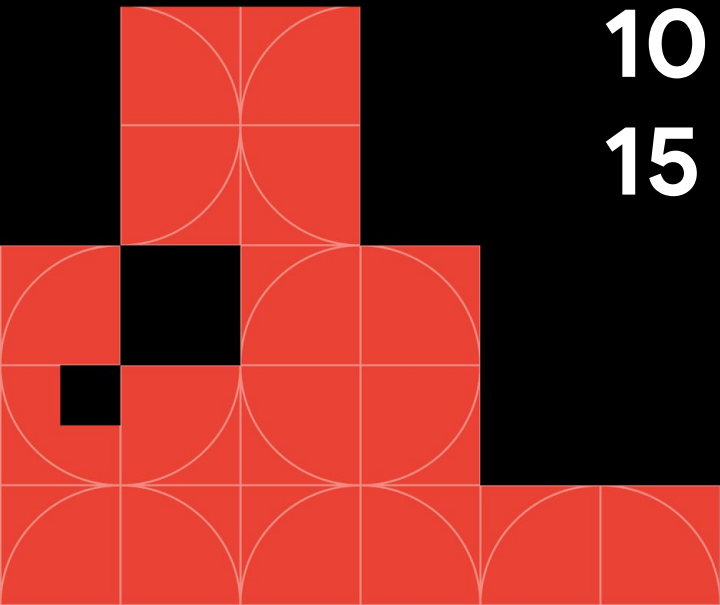


```
Text('Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.red[200],  
  ),  
),  
),  
s.star,  
r: Colors.red[500],  
Text('23'),
```

devfest



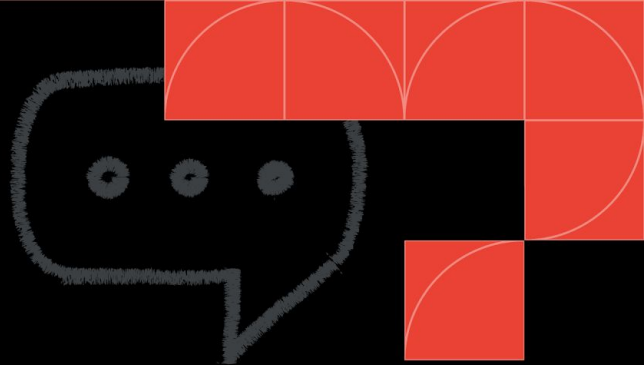
10 mins Planning
15 mins Building



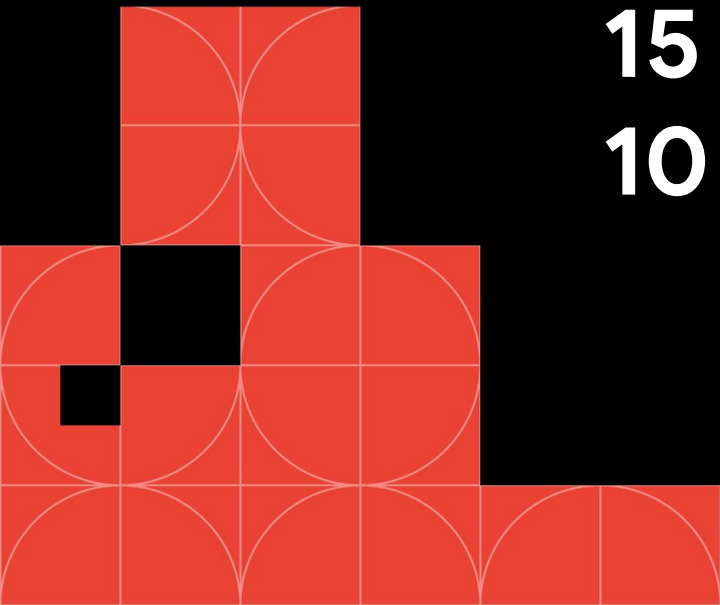
```
Text('Simple Statement or URL',  
style: TextStyle(  
  color: Colors.red[200],  
),  
),  
),
```

devfest

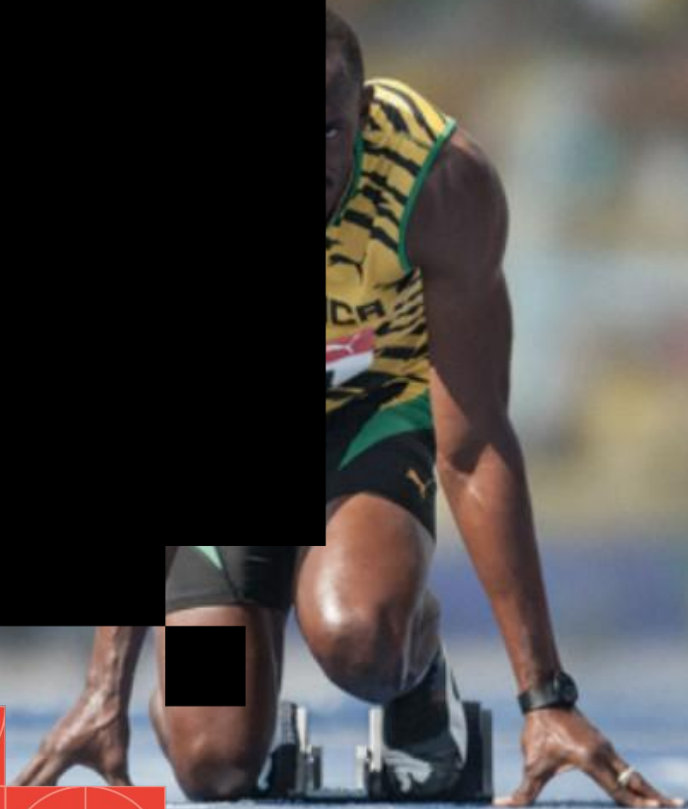
```
s.star,  
r: Colors.red[500],  
Text('23'),
```



15 mins Review
10 mins Retrospective



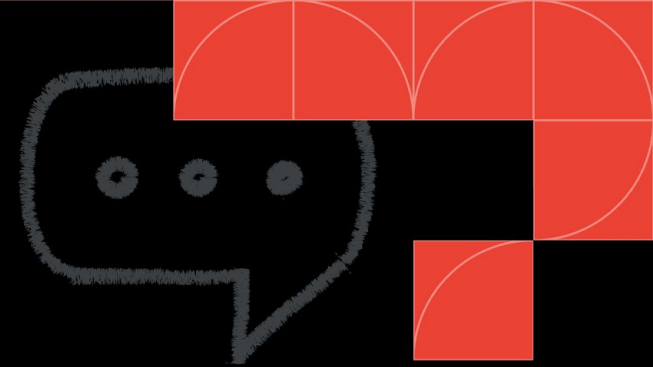
Ready?



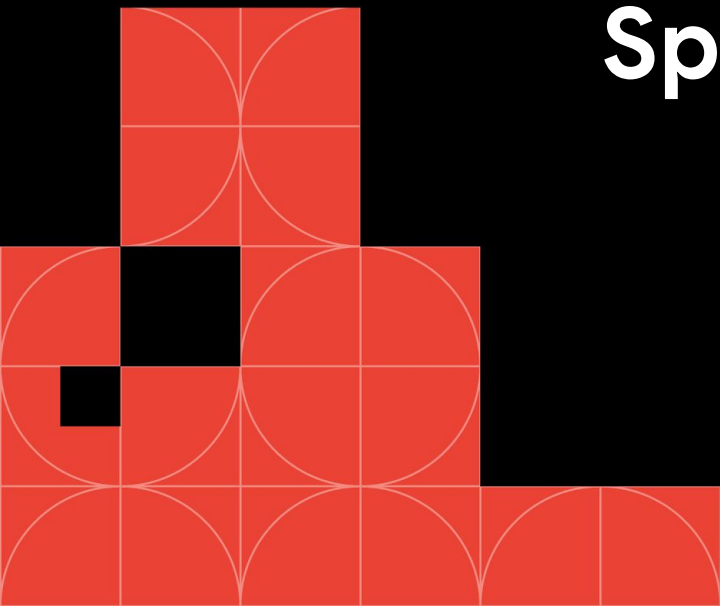
```
ext(  
  'Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.red[200],  
  ),  
),  
),
```

devfest

```
s.star,  
r: Colors.red[500],  
Text('23'),
```

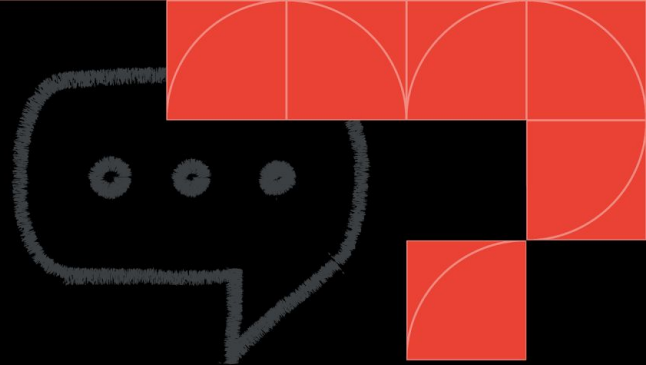


Sprint 1

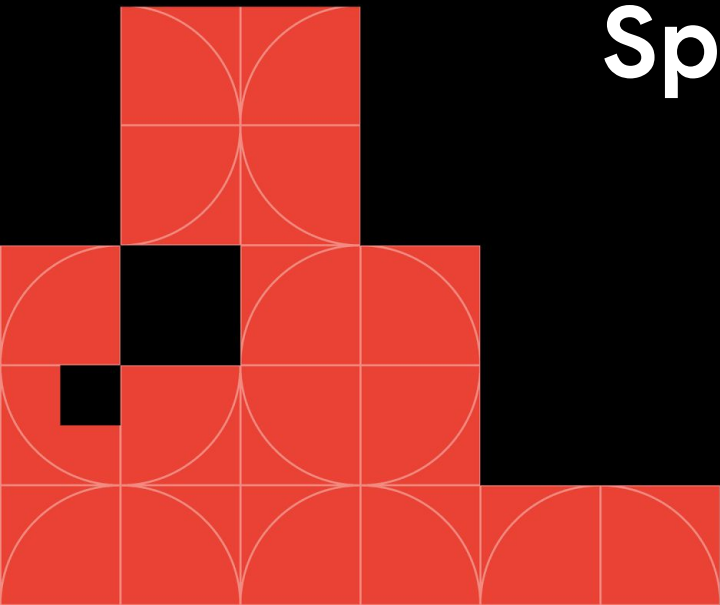



```
ext(  
  'Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.red[200],  
  ),  
),  
),  
s.star,  
r: Colors.red[500],  
Text('23'),
```

devfest



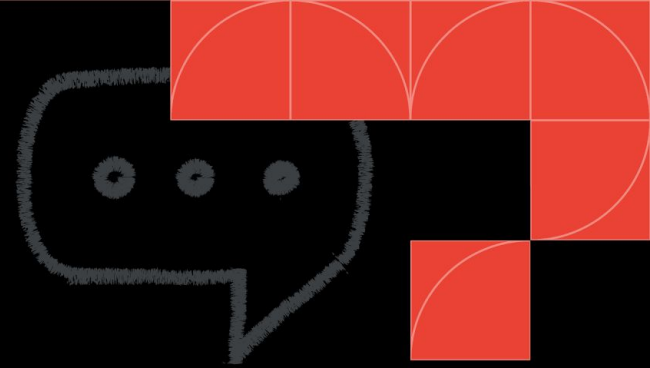
Sprint 2



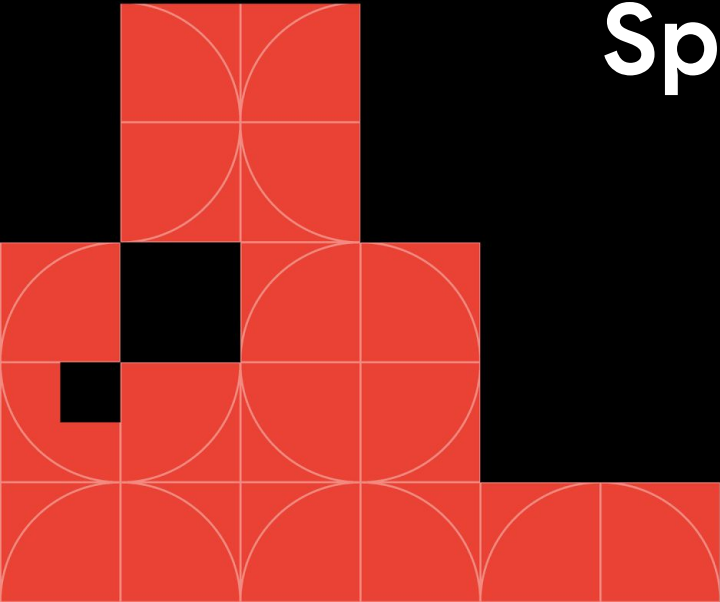
```
ext(  
  'Simple Statement or URL',  
  style: TextStyle(  
    color: Colors.red[200],  
  ),  
),  
),
```

devfest

```
s.star,  
r: Colors.red[500],  
Text('23'),
```



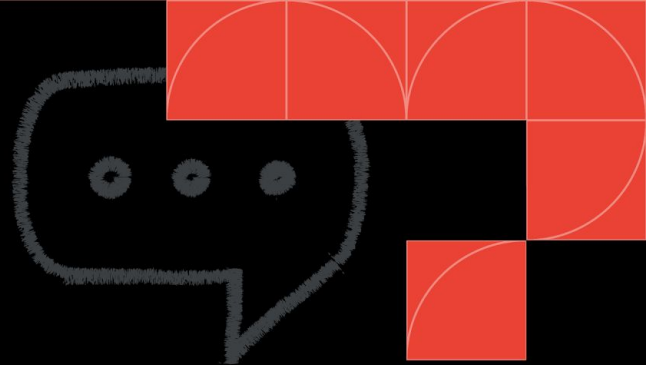
Sprint 3



```
text(
  'Simple Statement or URL',
  style: TextStyle(
    color: Colors.red[200],
  ),
),
),
```

devfest

```
s.star,
r: Colors.red[500],
Text('23'),
```



Sprint 4

