When I develop software (for a project, assignment, etc.) I...

...use a "trial and error" approach

88%

14 votes

...follow a well defined process

0%

0 votes

...find it challenging to choose where to start from

31%

5 votes

...add new code and keep existing code as it is, to avoid breaking it

44%

7 votes

...add new code and change existing code when needed,
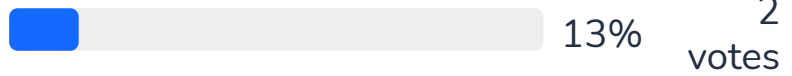
44%

7 votes
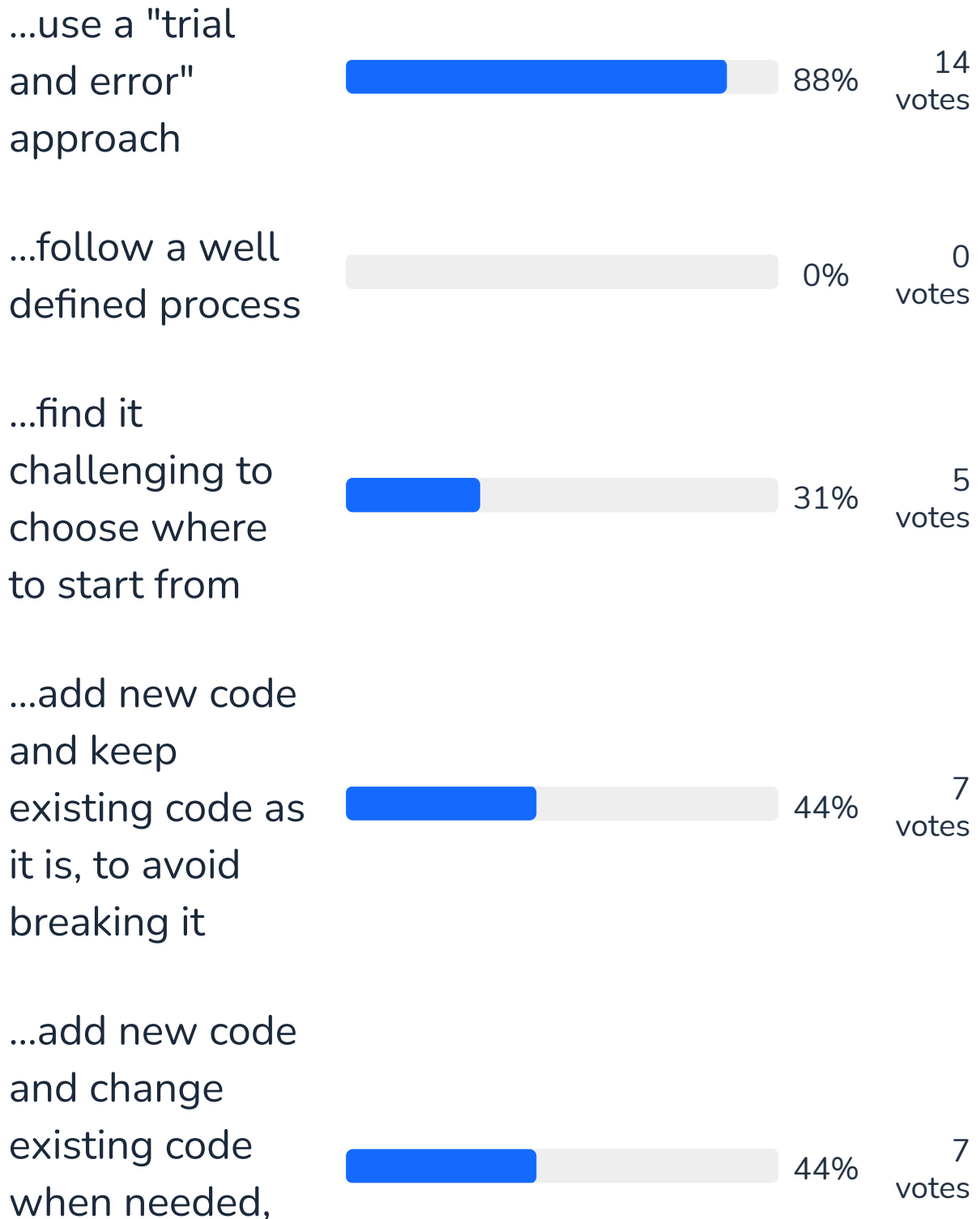
without fear

...manually check that everything is still working as expected

75% | 12 votes
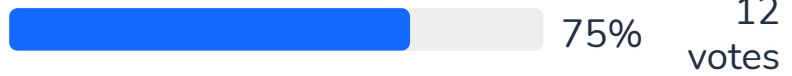
...automate as much tasks as possible

13% | 2 votes

What qualities do you associate to clean code and ugly code?

Clean code

Only essential commands

Understandable, well commented (but not unnecessarily)

Elegant, simple

Easy to understand, short and well commented

Easy to understand

understandable

Not too nested

Easy to read

Commented

Easy to debug and to maintain

Understandable
Short
Explainable

Well documented

Readability

Easy to understand

Easy to read & understand

Simple

Clear

Readable

Few lines

Understandable

Short

Ugly code

Too many checks

Spaghetti code

Bad variables name

Hard to understand

With Duplicated code

No modularity

Variables are obscure

Difficult to understand

Full of cycles and if statements

Complicated

Bad to maintain

Difficult to debug

Long

spaghetti like

Difficult to read and maintain

Long difficult flows of code

Hard to understand

Why does code grow ugly?

Worrying only and only about making it work

No good first implementation

Incremental programming without changing old code

Lack of organization

Bad design

Lack of coding experience

One thought behind can be: It works we do not need to change it

inexperienced developer

Inability to separate small tasks