

end-to-end
tests



end-to-end
test?

promotion

JETZT ZU WECHSEL
JETZT ZU BLAU.DE

9CENT
PRO MINUTE UND SMS
FREI VON VERTRAGSBINDUNG
INKL. KOSTENSCHUTZ

Startguthaben der SIM-Karte:	5€	10€
Endguthaben bei Handynr. -		25€

Aktion bis 17.10.

Jetzt SIM-Karte für 9,90€ bestellen!

The advertisement features a blue background with yellow leaves. A red oval highlights a table comparing the start and end credit of a SIM card. The start credit is shown as 10€ with a crossed-out 5€ next to it. The end credit is 25€. Below the table, it states the promotion ends on 10/17 and provides a call to action to order the SIM card for 9.90€.

order with
promotion...

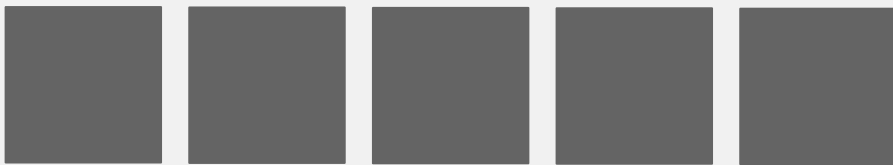
... bill should have
discount

easy?

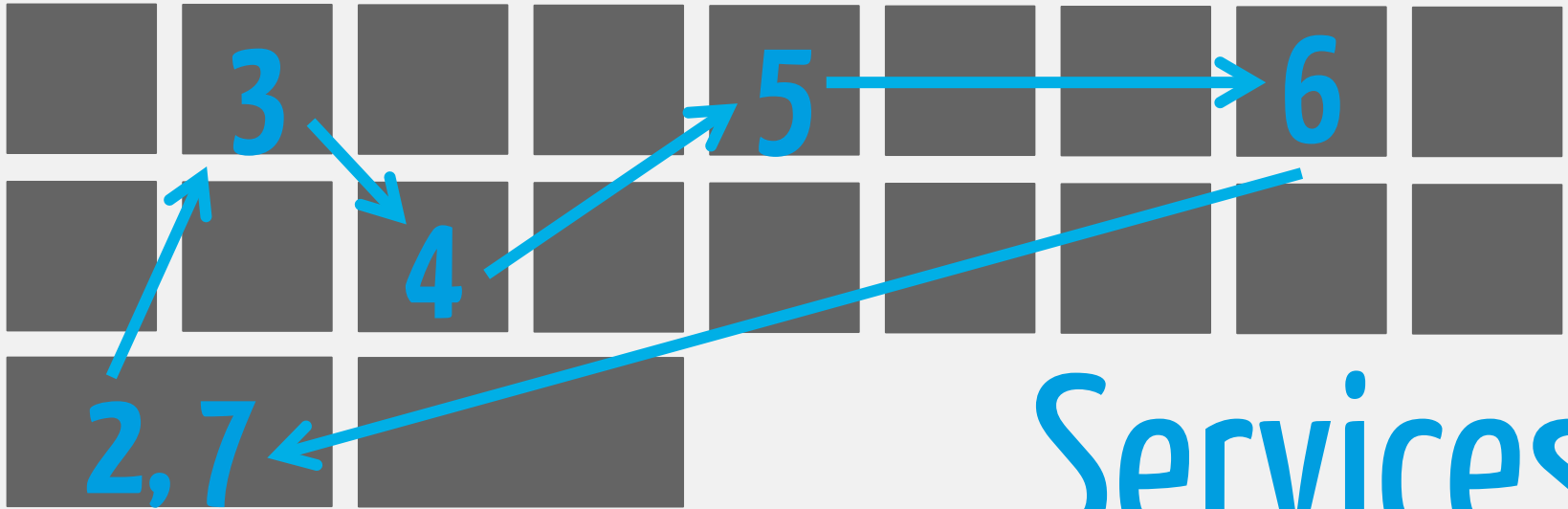
hard?



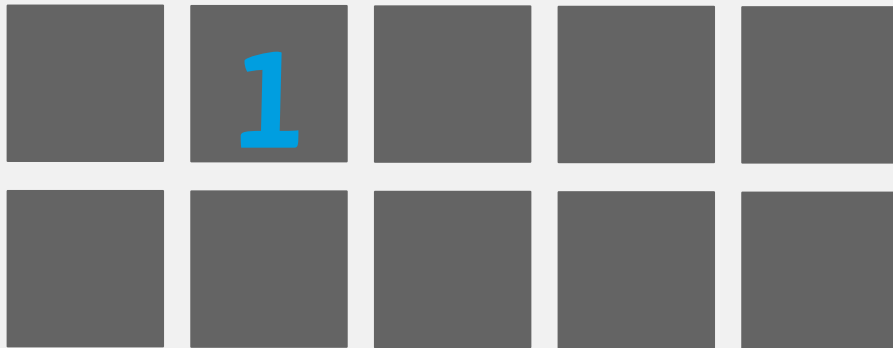
Services



Frontends



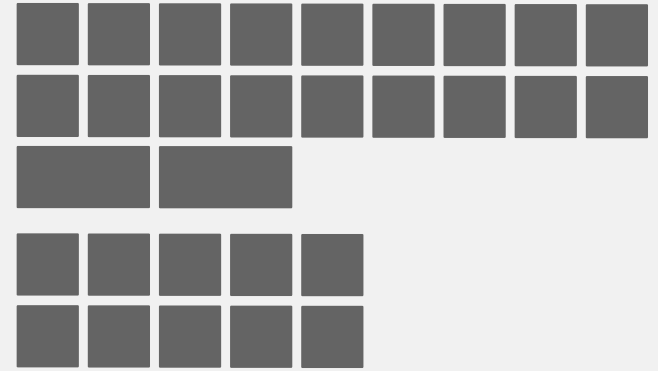
Services



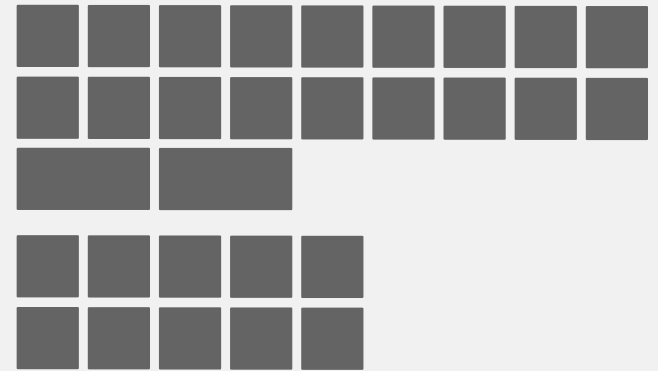
Frontends

tool 1:
deployer

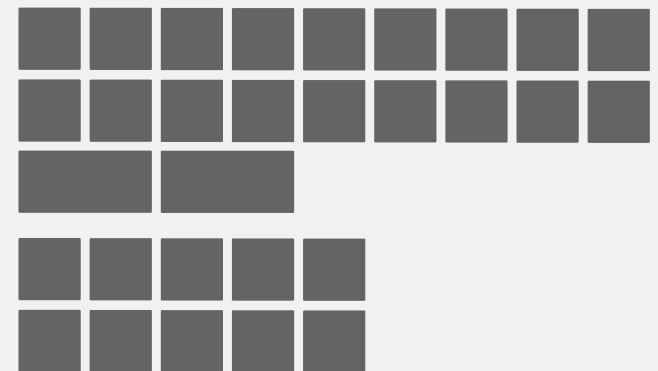
CI test 1



CI test 2



CI test 3



get current

stuff

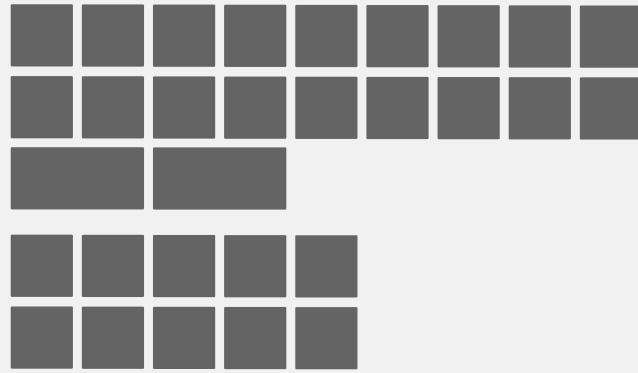
deploy to
container

(passenger, tomcat)

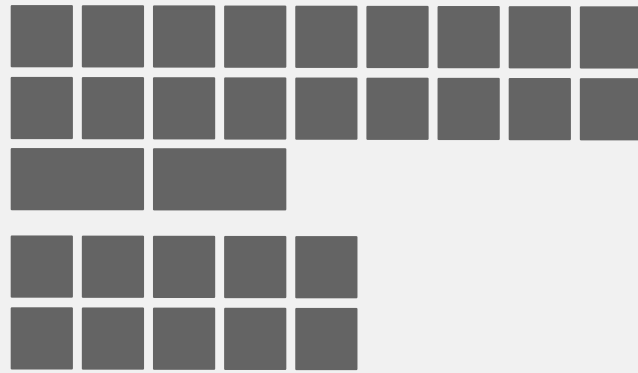
wait until
ready

tool 2:
deploy the
databases

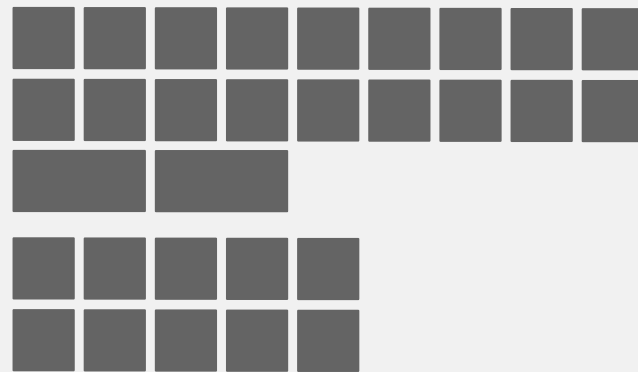
CI test 1



CI test 2



CI test 3



empty db =

fast, small

apply current
schema

almost empty

database

(only master data)

tool 3:

test data

test creates
the data it
needs

```
create_promotion (  
start => "01.01.2011",  
end => "01.02.2011",  
bonus => "5€" )
```

always test
with the
same data

tool 4:
ui tests

RSpec +
Capybara

open_confirm_order

check "termsAndConditionsBox"

click_link "submit"

page.should be_on_path

"orderComplete"

don't test for
content

test for
elements

set browser
size

tool 5:
async jobs

1. place order ✓

2. later: process

3. later: bill

trigger job

processing

process

interesting

order only

tool 6:

time travel

process bill

after 30 days

...but not on
weekends

fixed time!

time of order:

02.01.2011

14:25

processing

02.01.2011

14:30

billing:

03.02.2011

00:00

tool 7:

fail verbose

complex

tests

fail easy

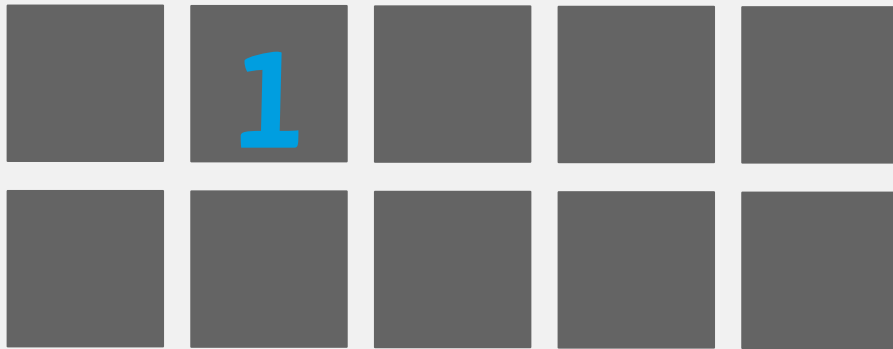
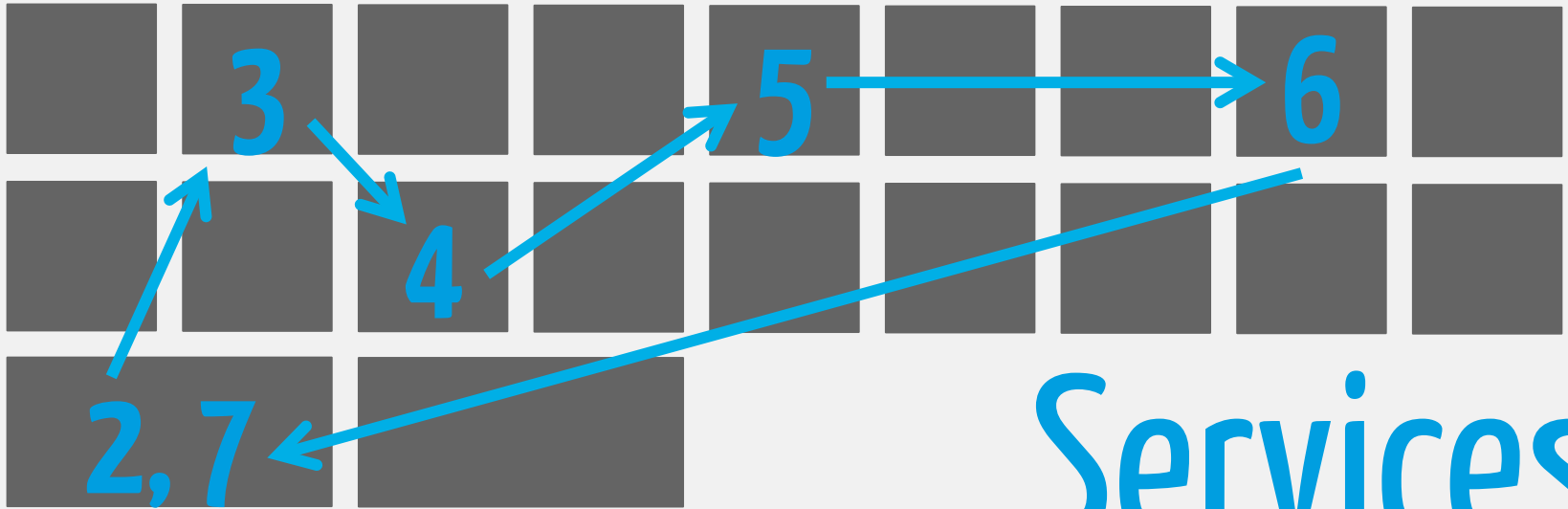
test in
isolation...

...with fixed
data

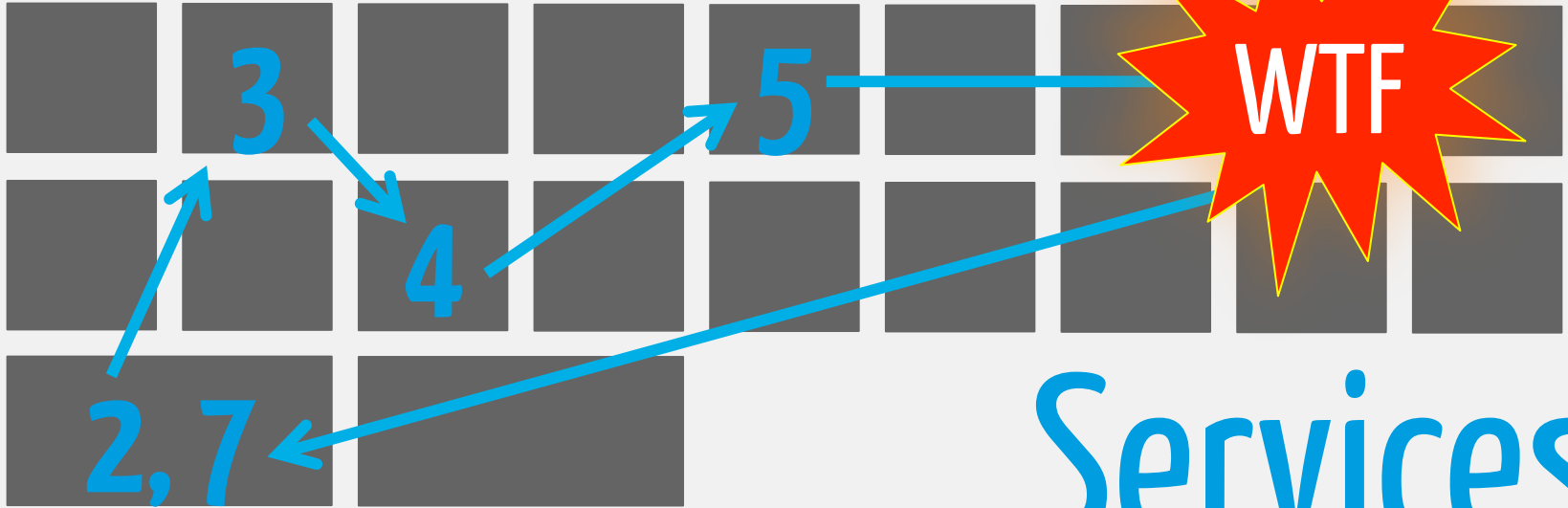
make it easy

to find out

what failed



Frontends



Frontends

show stacktrace
over service
boundaries

show

page source

conclusion

easy?

hard?

depends on
your tools

deployer, test data,
ui tests, async jobs,
time travel,
fail verbose

more stuff:

blaulabs.de

