

The Modern Approach



Michael Mazyar

astrails.com

blog.astrails.com

2 approaches to
develop a **product**

predictive development

adaptive development

space travel anyone?

based on true stories

faster, better, cheaper

extreme? agile? scrum?

whatever!

major steps

define a problem

find solution

get feedback

make a research

build a prototype

get feedback

minimal solution

launch

real fun begins

get feedback

yes, again

digest feedback and
release

iterate until perfection

those were major steps

let's get technical

web applications

functional specifications

really short, 1-2 pages

it's changeable

you never end up with what was planned

abstraction

people may understand it differently

functional specifications

is not a real thing

what's "real"?

UI design is real

UI design and usability
are **most important**

legacy flow

- write a spec
- do programming
- create visual design
- implement visual design

legacy flow

- write a spec
- do programming
- create visual design
- implement visual design
- perform usability testing
- update visual design

legacy flow

- write a spec
- do programming
- create visual design
- implement visual design
- perform usability testing
- update visual design

workflows

start with UI design

platforms / frameworks

go open source

innovation, productivity, bang for the buck

- PHP

- something small, not easy to maintain, less skilled R&D

- **Ruby on Rails / Python-Django**

- easy to maintain, rapid development, less code, less pain, more skilled R&D

- Java

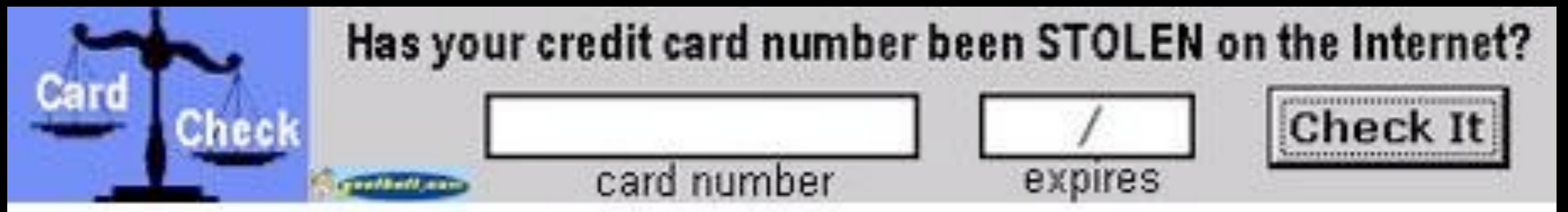
- no, but thanks

build for today

scalability

scalability problems are good
it means you're growing fast
enough

security



Has your credit card number been **STOLEN** on the Internet?

Card **Check**

/

card number expires

Check It

focus on your main
business

everything is online

hosting in the cloud

managed hosting

some numbers

adaptive development
rules

Thanks



Michael Mazyar
astrails.com
blog.astrails.com