

---

# Don't fear Erlang fear your team

Meike Hecker

Erlang developer @ SumUp

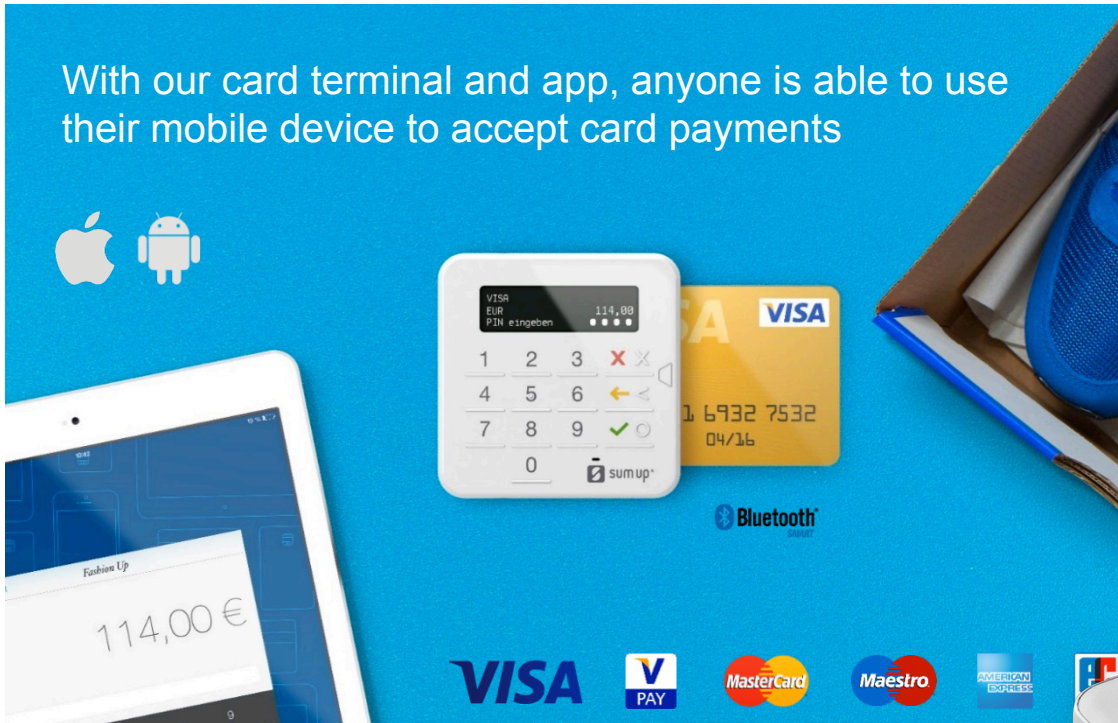
Erlang factory Berlin 2015

# Overview

- SumUp
  - Product
  - Where do we use Erlang
- Why do people fear Erlang
- Why fear your problem
- Why fear your team

# SumUp – the product

With our card terminal and app, anyone is able to use their mobile device to accept card payments



The graphic features a blue background with a white SumUp card terminal in the center. To its left is a white Visa card with the number 6932 7532 04/16. To the right is a brown cardboard shoe box containing a blue sneaker. In the bottom left corner, a white tablet displays the SumUp app interface with a transaction of 114,00 €. The top left corner shows the Apple and Android logos. The bottom of the graphic displays logos for VISA, PAY, MasterCard, Maestro, and AMERICAN EXPRESS. A Bluetooth logo is also present.



# SumUp Where do we use Erlang

- ▮ Payment processing fully powered by erlang
- ▮ Riak for sensitive data
- ▮ Postgres and ruby for the rest

# Why do people fear Erlang?

- Erlang is hard to learn
- Java is hard to learn
- **Any highlevel language is more or less hard to learn**

There is not now, nor has there ever been, nor will there ever be, any programming language in which it is the least bit difficult to write bad code.

# Why do people fear Erlang

- It is difficult to hire Erlang developers
- It is difficult to hire Java developers
- It is difficult to hire good developers
- **Hiring is about finding a dev to solve your problem**

If you want a product with certain characteristics, you must ensure that the team has those characteristics before the product's development.

— Jim McCarthy and Michele McCarthy - Software for your Head

# Why fear your problem

- complexity of problems grows
- Important skill is to understand the problem
- Teamwork is essential in today's software development processes

There are two ways of constructing a software design: One way is to make it so simple that there are obviously no deficiencies and the other way is to make it so complicated that there are no obvious deficiencies.

— C.A.R. Hoare, The 1980 ACM Turing Award Lecture

# Why should you fear your team

- Team needs to share knowledge to find the best solution
- Team needs to work out requirements
- Team needs to agree on design and style

Without requirements or design, programming is the art of adding bugs to an empty text file  
Mickey W. Mantle and Ron Lichty



---

# Thank you Questions?

You can't have great software without a great team, and most software teams behave like dysfunctional families.

— Jim McCarthy