Modelling Use Cases with Pharo

Problem statement and solution proposal
What are models?
Why do we need models?

- Communicate design
- Promote ideas
- Visualize functionality
- Design production process
Why do we need models in software systems?

- Communicate with stakeholders and team members
- Visualize logic and behaviour
- Enable different perspectives on the problem
- Plan implementation
How are models used in software systems?

• Proof of concept phase
• UMLs for:
  ○ Classes
  ○ Activities
  ○ Sequences
  ○ Use cases
• DSLs for:
  ○ Behaviour tests
• Documentation
Do the models fulfil all their purposes?

Kind of, but not completely.
What’s the problem?

- Static
- Doesn’t get communicated enough
- No transparency for the stakeholder between the model and the product after POC phase
Why is it a problem?

- End product deviates from the initial models
- Team members do it their way - “There was a UML for that?”
- Expectations of the product will not be met - “This is not what we have payed for!”
What could be an approach to these problems and why?

Visualisation of all acquired use cases in a single dynamic model:

- Model is **directly** linked to the developed program code
  - Model represents code and vice versa
- Designed to evolve
- Understandable with minimum knowledge about domains
What have I done so far?

- Initial programming experience with Pharo and Bloc
- Data acquired
  - 93’000 diagrams from 24’000 projects from github
Why Pharo, why Bloc?

- Live programming
  - Advanced run-time reflectivity
  - Live customizable objects inspection
- Bloc
  - Bloc is a low-level UI infrastructure & framework for Pharo
  - Boxes and arrows!
What will I do next?

- Data analysis and interpretation
- Implementation cycles of the modelling tool
Questions?

Thanks for listening