

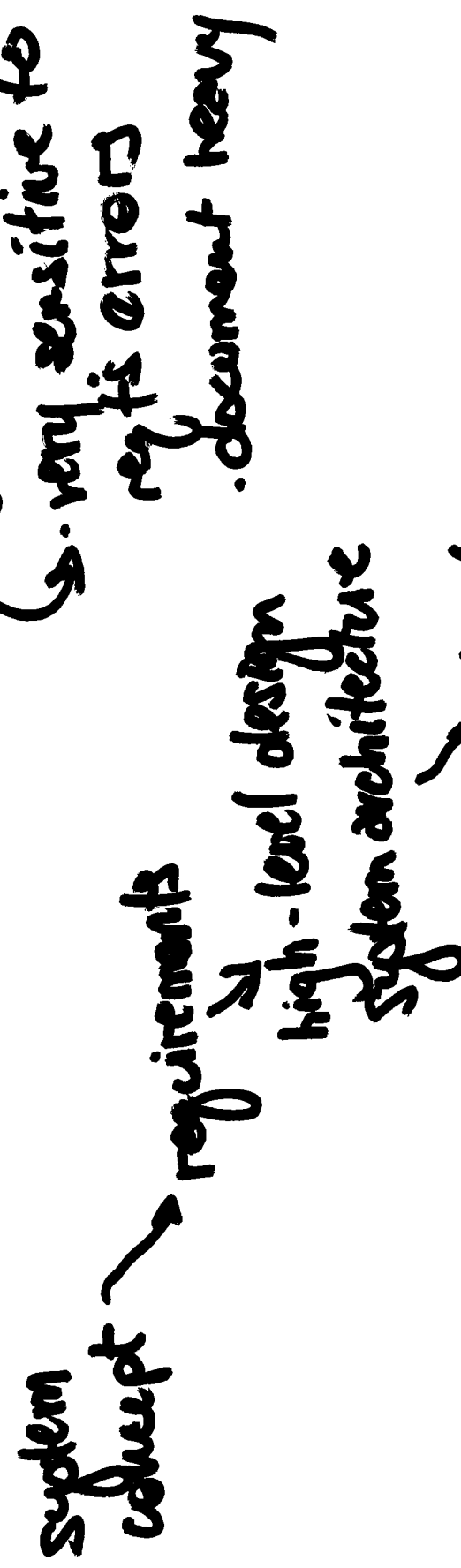
Code and Fix

- informal
- develop shared understanding, then start coding
- low overhead

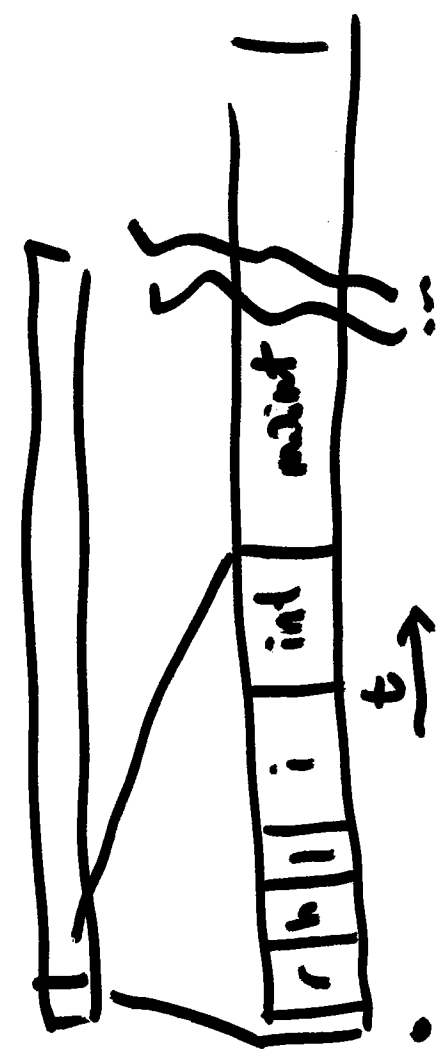
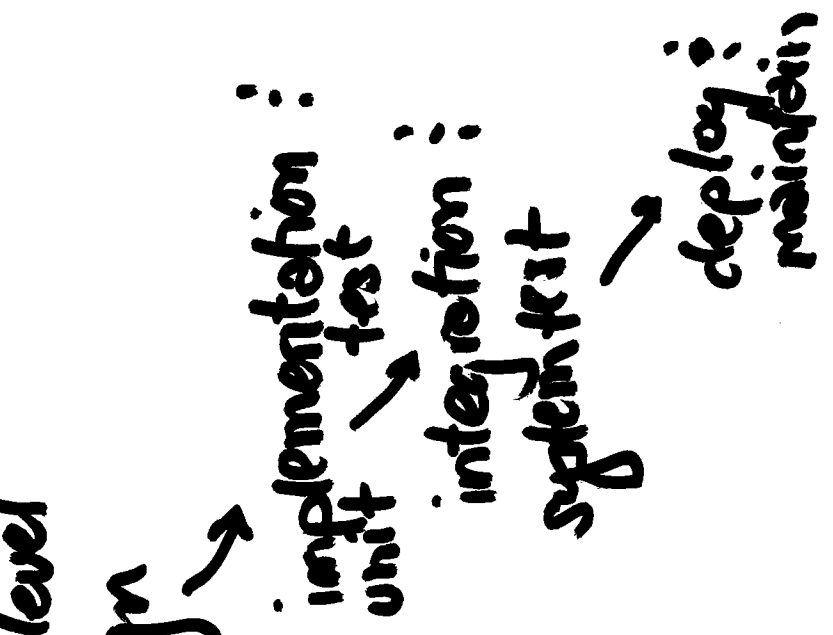
Con:

- hard to predict time to completion
- can be at "80% - 90% complete" for a long time
- hard to ensure s/w meets quality goals

Waterfall



- ↪ slow
- ↪ very sensitive to req's errors
- ↪ document heavy



Evolutionary Prototyping

- develop a prototype, then iterate on it until customer: developer agree it's

"good enough"

- non-executing < Paper prototype
- executing < mock-up app. : refine

Paper prototypes

- fast to create
- not as great an investment in final product - easier to change
- can do walk-throughs with users - early UI feedback
- easy to reason about

Jim's Web App Welcome

User:

Pass:

Login

Invalid Password

Invalid

Invalid Password