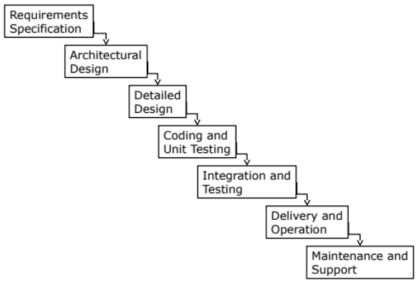
Use-cases and workflow

Use-case diagram

```
Use Case Name:
Withdraw Cash

Participating Actor
Bank Customer (BC)

Goal:
BC withdraws cash from their account

Preconditions:
System prompting to insert bank card
BC has bank card
BC has money in their account

Postconditions:
BC recieves request ammount in cash
BC account reduced by withdrawn ammount
BC has bank card returned
System prompts to insert bank card

Trigger:
BC inserts bank card

Basic Flow:
1. Sys prompts to enter PIN
2. BC enters PIN
3. System prompts for transaction type
4. BC selects type
5. System prompts for ammount
. . .
```

```
User
        Primary - End user, freq usage
        Secondary - Manager of end user, occasional use or by proxy
        Tertiary - Owner of the system, influences funding choices

User considerations
        Background - Motivation to learn, Literacy, Knowledge,
                Computer skills, Attitude to computers
        Kinds of use - infrequent/novice, frequent/expert
        Abilities - Physical, Social, Perception, Motor
        User communication issues - Cannot always express what
                they want, may not know what is and isn't possible,
                stick to set cases and knowledge, fear other factors
        Innovator's dilemma
```

Waterfall Lifecycle Model



Throwaway Prototyping

Process:
- build and test prototype
- gain knowledge for the real product
- "throw away" the prototype
- then "develop" the product for real

Incremental Prototyping

Process:
- triage system into separate "increments"
  - i.e., "must do", "should do", "could do"
- develop and add one increment at a time

Example (accounting system):
- prototype 1 — general ledger
- prototype 2 — accounts receivable/payable
- prototype 3 — payroll

Evolutionary Prototyping

Process:
- feature is refined or "evolved" over time

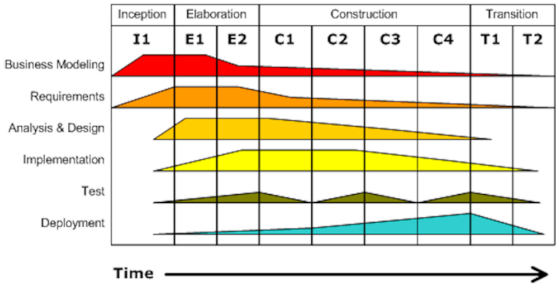
Example (text editor):
- prototype 1 — command key cut/paste
- prototype 2 — undoable cut/paste
- prototype 3 — drag and drop cut/paste

Unified Process

* Iterative
* Incremental
* Customizable
* Phases
  * Inception: Risks and Business Cases and Use Cases
  * Elaboration: use case diagrams and class diagrams
  * Construction Phase: implementation in iterations
  * Transition: Deployment

Iterative Development

Business value is delivered incrementally in time-boxed cross-discipline iterations.



Staged Delivery

Developers:
- deliver the system in a series of working releases or builds

Users:
- use some functionality while the rest continues to be developed

Possible parallelism:
- production and development systems
- staggered development streams

Staggered Builds