NanoSim

Yina Arenas
Mona Sergi
Tamal Saha

Overview

- Focus of the project
- Nanosim as a case study
  - Role playing simulation
  - Different types of participants and services
  - Deliver content according to participants properties
- Agenda:
  - Lifecycle
  - Risk Analysis and Mitigation
  - Requirements and Specifications
  - Architecture
  - Implementation
  - Source code management

Life Cycle

- Spiral/Evolutionary model
  - 2 weeks
    - Risk analysis
    - Requirement analysis
    - Preliminary architecture design
    - Wiki
  - 3 weeks
    - Complement initial architecture adding web service layer
    - Data access layer
    - Initial services
  - 3 weeks
    - Service implementation
    - Functional testing

Risk Analysis

- The first step our spiral model
- The risks:
  - Scope and time management
  - Team background
  - Git/SVN
  - Extensibility/Flexibility
  - GWT
  - Correctness

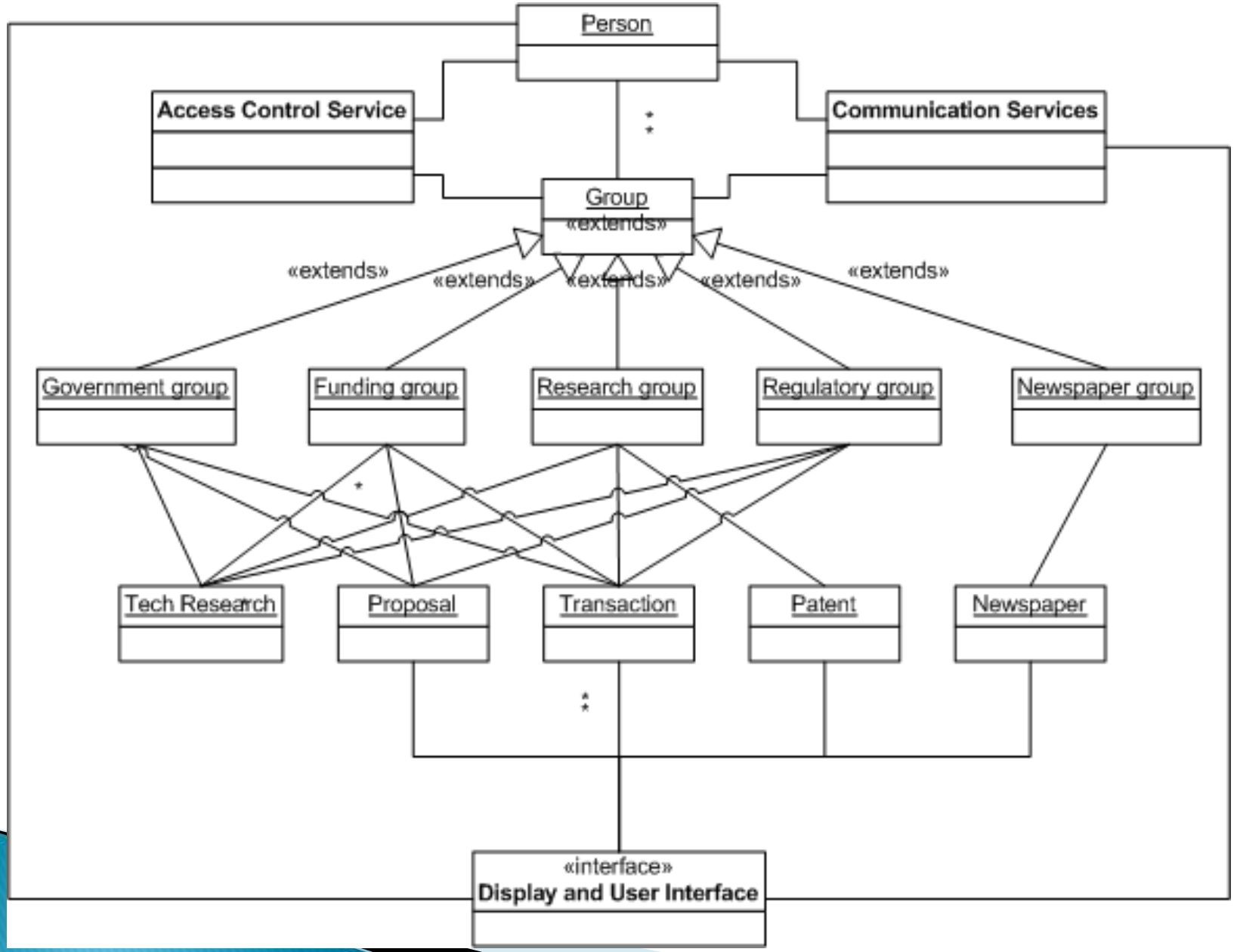
Requirements & Specifications

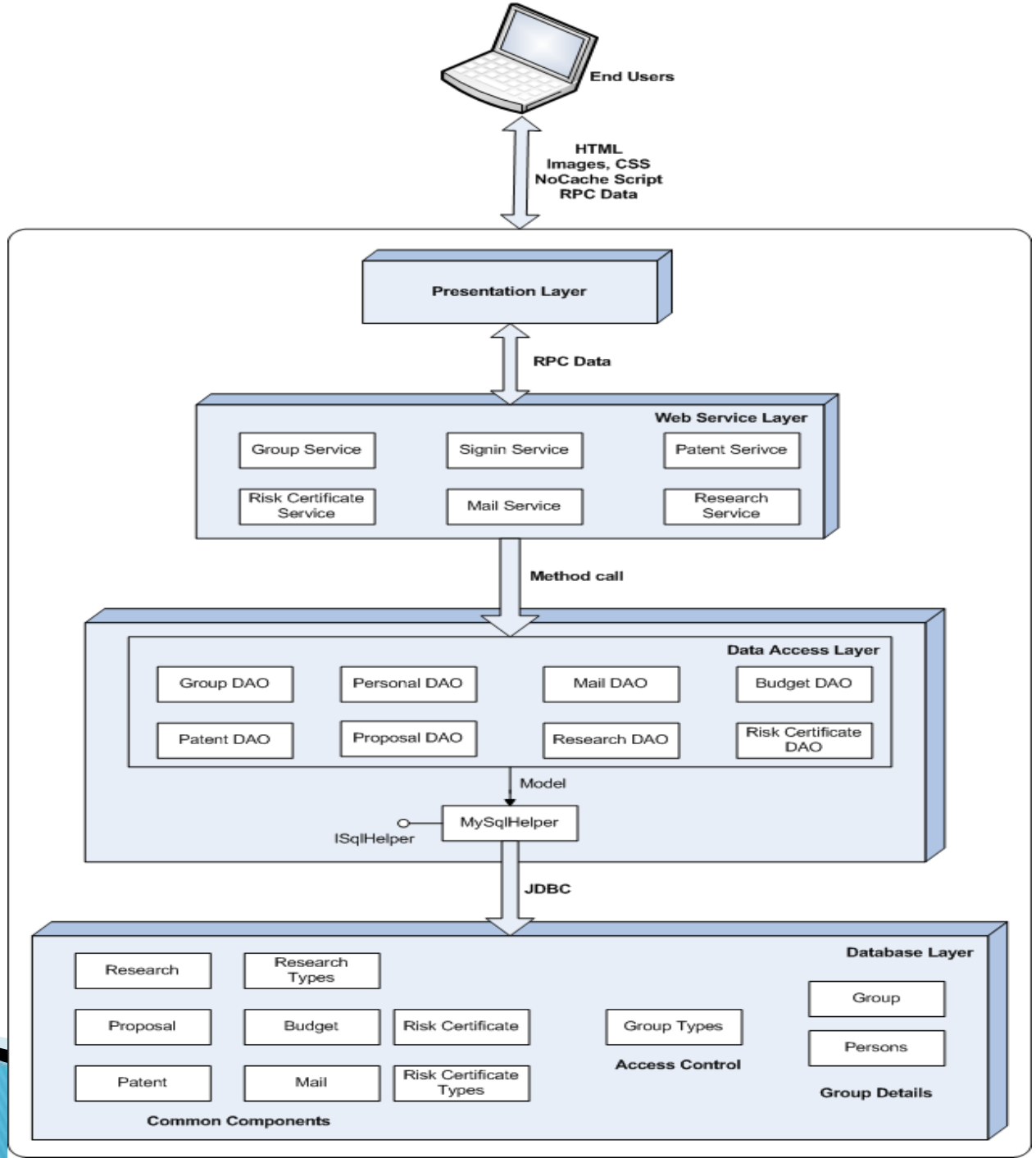
- Requirements
  - For developers
    - Easy extension
  - For instructors
    - Small time setup
  - For the students:
    - User friendly environment
    - Fulfilment of pedagogical goal
  - Access from different locations
  - No extra installation on the end user's system

Requirements & Specifications

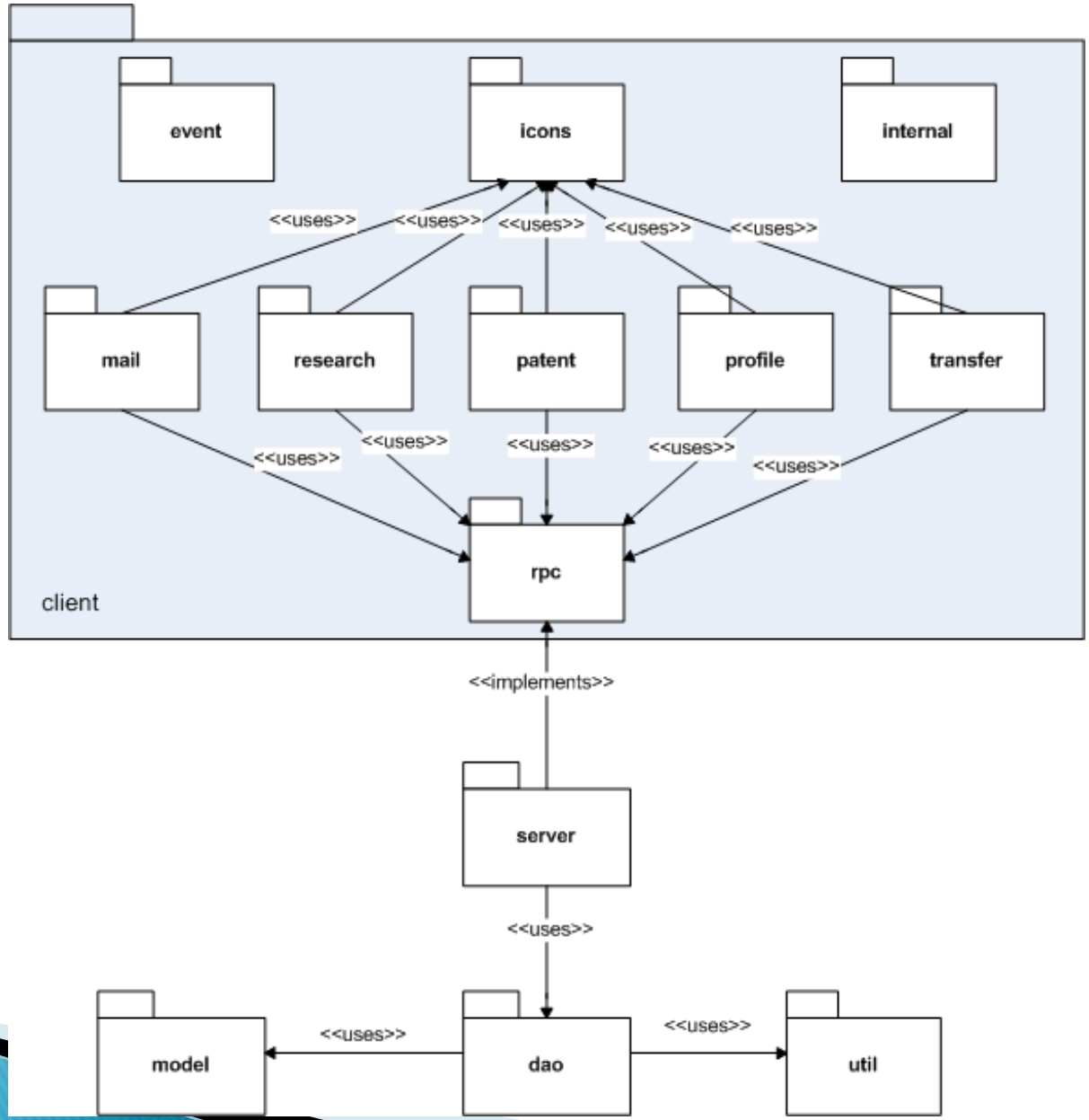
- Specifications:
  - Database independency
  - Access control
  - Services
    - Group / user management
    - Budget transactions
    - Mail
    - Proposal
    - Research
    - Patent

Architecture




End Users
HTML
Images, CSS
NoCache Script
RPC Data
Presentation Layer
RPC Data
Web Service Layer
Group Service
Signin Service
Patent Serivce
Risk Certificate Service
Mail Service
Research Service
Method call
Data Access Layer
Group DAO
Personal DAO
Mail DAO
Budget DAO
Patent DAO
Proposal DAO
Research DAO
Risk Certificate DAO
Model
ISqlHelper
MySqlHelper
JDBC
Database Layer
Research
Research Types
Proposal
Budget
Risk Certificate
Patent
Mail
Risk Certificate Types
Common Components
Group Types
Access Control
Group
Persons
Group Details

Implementation



Source code management

Conclusion

- Developed an understanding of the challenges involved in developing software
- Better literature comprehension by doing an actual project implementation.
- Learned about new technologies
- Got further insight on the interaction strategies and team work skills by using a collaborative development environment.

Any questions?