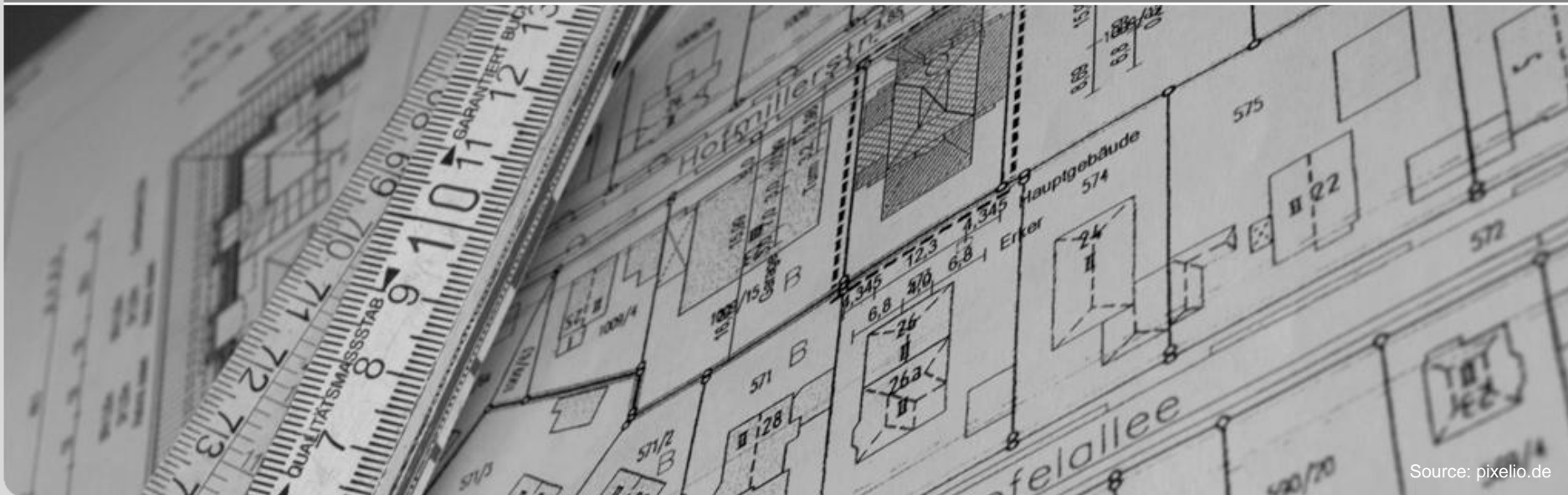


PSE - Entwurfsheft Analysesoftware für soziale Netzwerke

Team 7

SOFTWARE DESIGN AND QUALITY GROUP



Source: pixelio.de

Systemarchitektur

Systemarchitektur



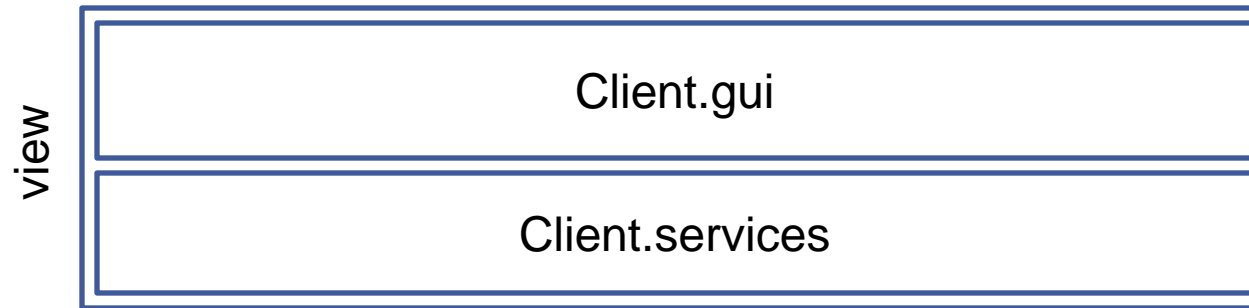
Systemarchitektur



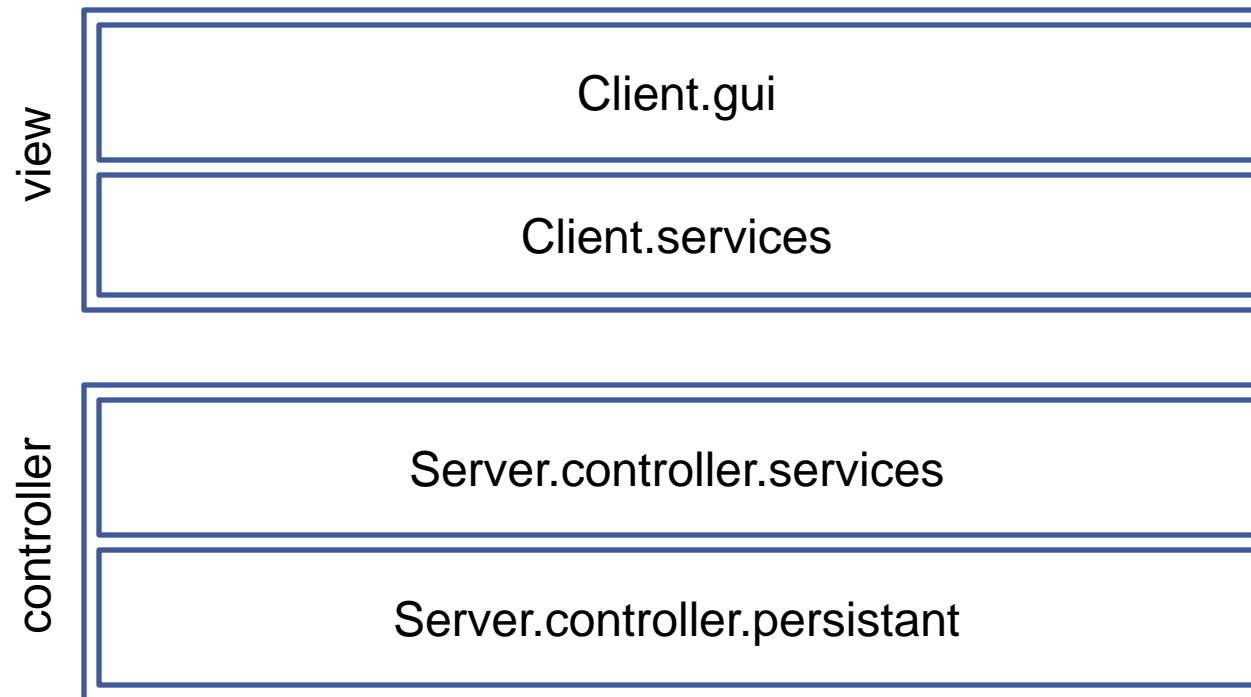
SocialView



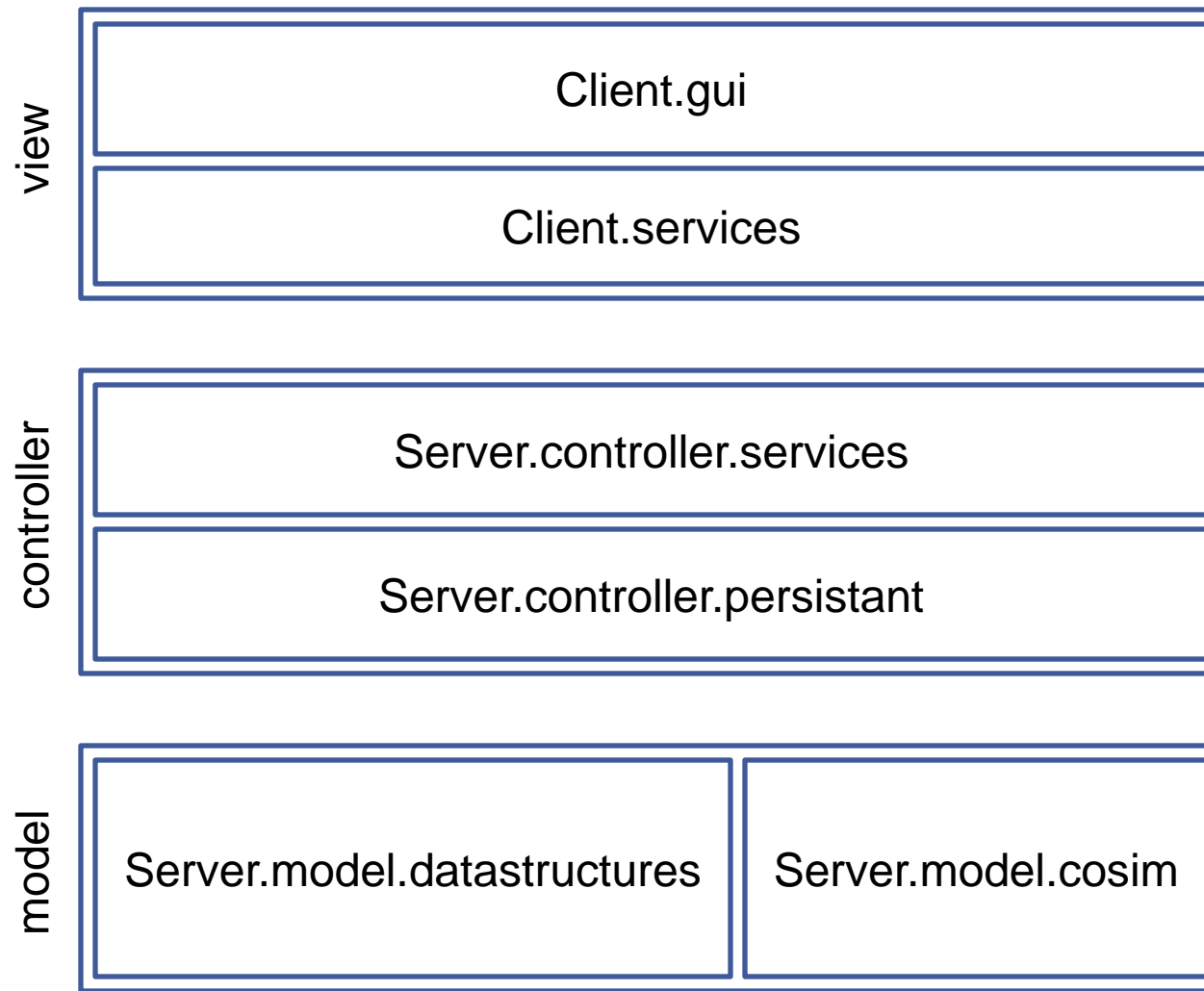
Systemarchitektur



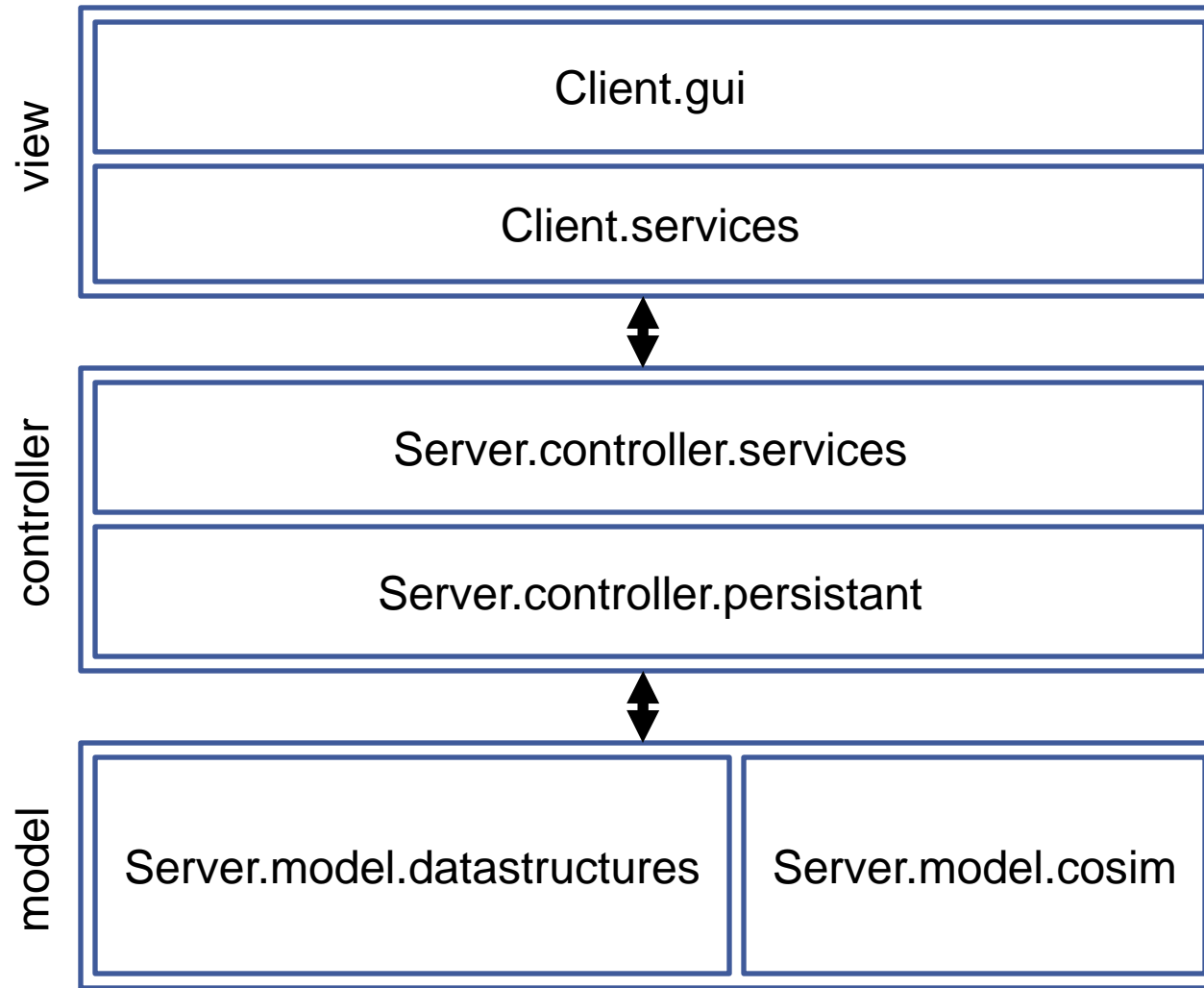
Systemarchitektur



Systemarchitektur



Systemarchitektur



Paketstruktur

client

Client.gui

- Benutzeroberfläche
- Starke Bindung zu GWT
- SocialViewEntryPoint als Einstiegspunkt
- Gestaltung der GUI über Panels

Client.datastructures

- Clientseitige Datenstrukturen
- Zur Kommunikation zwischen Client und Server
- Zur clientinternen Kommunikation

Client.services

- Schnittstellen für Serverkommunikation
- Fassaden
- Zugriff geschieht von Panel-Unterklassen in client.gui
- GWT-Workflow
 - Clientseitige Service-Proxys

Paketstruktur

Server.controller

server.controller.services

- Schnittstellenimplementierungen von Interfaces in client.services
- Delegierung an server.controller.persistent
- GWT-Workflow
 - Clientseitige Service-Schnittstellen werden im Server-Package implementiert
 - Namensgebung

server.controller.persistent

- Persistierung
- Anwendungslogik
- Erweiterbarkeit durch Plugins

Paketstruktur

Server.model

server.model.cosim

- Wird durch Datenbankschema von Cosim definiert
- Zugriff auf Netzwerk-Datenbank
- JPA
 - Entity-Klassen
 - Mapping mittels XML-Dateien

server.model.datastructures

- Datenstrukturen ausschließlich für Server
- Graph
- Einzige Abhängigkeit von server.model.cosim

Workflow

Graphanfrage

Spezifikation der Graphenanfrage

NetworkGraph
+getRestrictions ()
+getMeasures ()
+getName ()

- MenuPanel verändert NetworkGraph
 - Anpassen von Restrictions
 - Aktivieren und Deaktivieren von Zentralitätsmaßen

Absenden der Graphenanfrage

NetworkPanel
- networkService
- savedDisplayService

- NetworkPanel beobachtet NetworkGraph
- Bei Veränderung wird NetworkPanel benachrichtigt
- NetworkPanel gibt den NetworkGraph an den Proxy networkService weiter

Versenden der Graphenanfrage



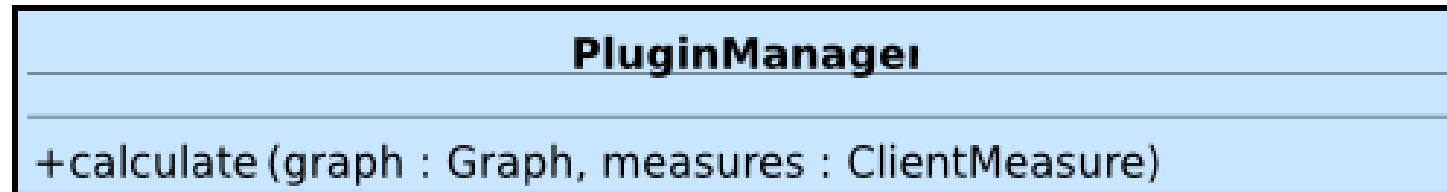
- getGraphDisplay() wird aufgerufen
- NetworkServiceImpl verifiziert Credentials und delegiert Anfrage an Network

Erstellung des Graphen

Network
- name : String
+ getGraph (graphRequest :) : Graph
+ authenticate (cred : Credentials) : NetworkGraph
+ getName () : String

- getGraph() wird aufgerufen
- CachedGraphManager sucht nach Einträgen
- GraphFactory erstellt Graphen
 - Der Supergraph wird geklont
 - Mögliche Restrictions werden angewandt

Delegation an Plugins



- Der PluginManager verwaltet die Plugins
- In caculate() wird die Berechnung der Zentralitätsmaße durch die Plugins angestoßen

Berechnung der Zentralitätsmaße

Plugin

```
+ calculate (graph : Graph, restrictions : Restriction) : Graph  
+ getMeasure () : ClientMeasure
```

- Abstrakte Oberklasse für Plugins
- Jedes Plugin berechnet ein Zentralitätsmaß
- Zu jedem Knoten bzw. Kante wird ein neues Gewicht hinzugefügt
- Restrictions werden angewandt

Berechnung der Graphdarstellung

GraphDisplay
- data : Object
+getData () : Object

- Ein GraphDisplay wird mittels des modifizierten Graphen und NetworkGraph erstellt
 - NetworkGraph enthält Darstellungsarten (MeasureStyle)
 - Graph enthält Zentralitätsmaßberechnungen
 - Speichert xdot
- Das GraphDisplay wird zurück an den Client geschickt

Aktualisierung der Anzeige

- Neues GraphDisplay wird in GraphDisplayModel abgelegt
- GraphPanel beobachtet GraphDisplayModel und wird benachrichtigt
- GraphPanel aktualisiert Anzeige