

Y. Chen, M. Anev,  
R. Khalid, F. Baylac-Jacque, H. Sedighizadeh

# Requirements and Design



# Overview

- Intro
- Requirements
- User stories and Use cases
- Design & Technologies
- Conclusion

# Project Background



- Identify Open source communities
- Evaluate and uncover characteristics of different communities
- Visualize the relationship and types of communities

# Real world Example



# Requirements Gathering Process

- Getting the corresponding answers from customer
  - Users: Guest, User, Administration
  - Meeting via Skype
- Identified Use Case scenarios
- Gathering questions from team

# Goals

**[G1]** Allow users to measure community social and organizational characteristics

**[G2]** Allow users to compute social network that makes characteristics explicit

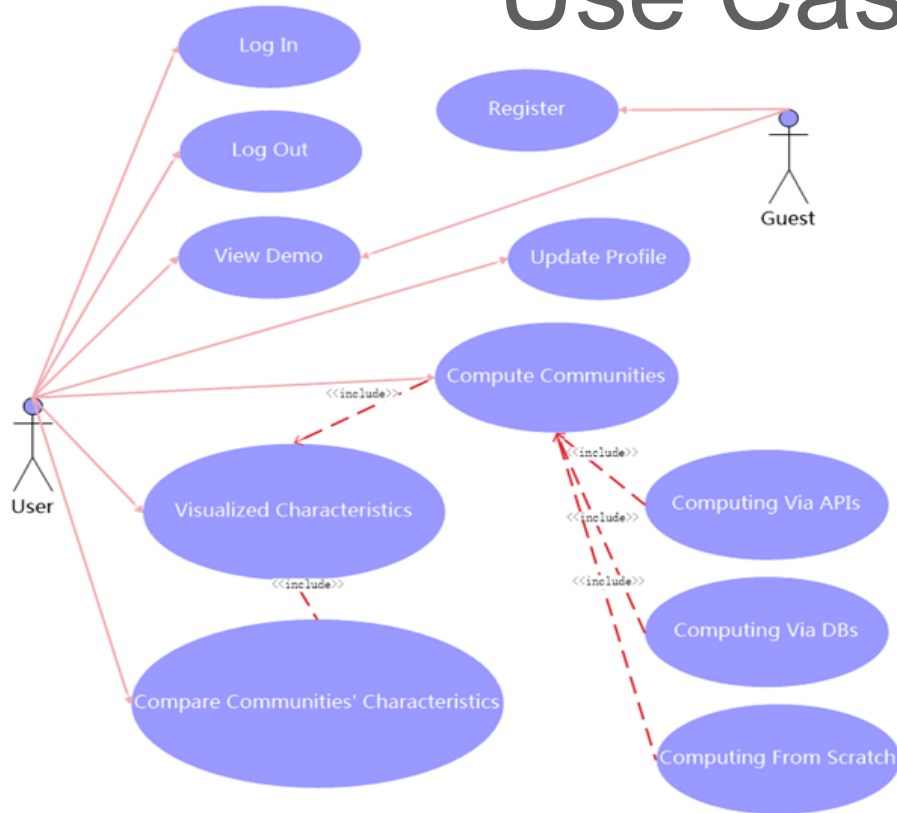
**[G3]** Allow users to study type and characteristics against performance metrics to evaluate fitness

**[G4]** Allow users to understand the characteristics

**[G5]** Allow users to visualize the characteristics

**[G6]** Allow users to focus on one characteristic or inspect multiple communities and their visualization at the same time, in the same screen.

# Use Case Diagram



## Key Problems:

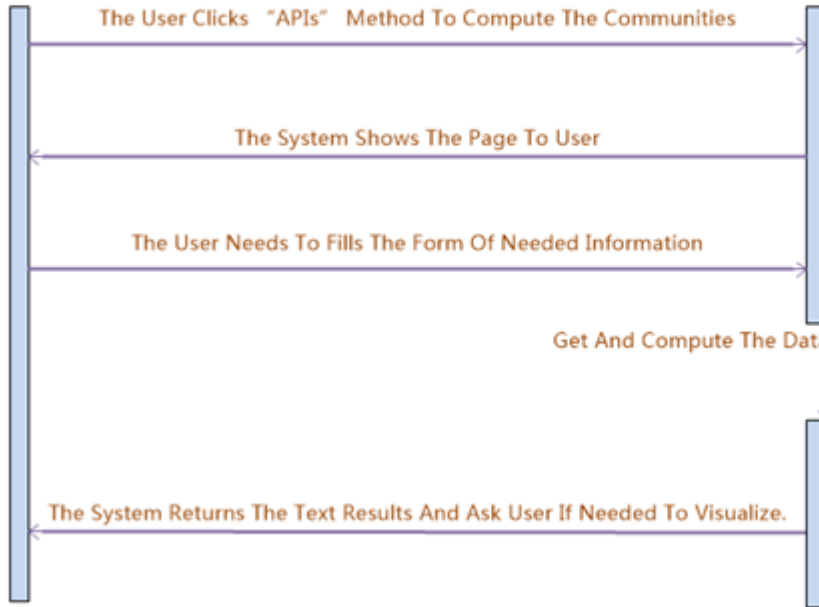
- Compute Communities
  - via APIs
  - via DBs
  - from Scratch
- Visualize Characteristics

# User Story



1. A user is curious about a (open source) community
2. This user finds some visualized demo for famous communities
3. This user registers and uses YOSHI to compute the community
4. This user gets visualized report for the researched community





## A use case

1. User opens the specific page
2. User selects interesting API and community
3. User will get the text result, and if needed, the visualized report.

# First sprint

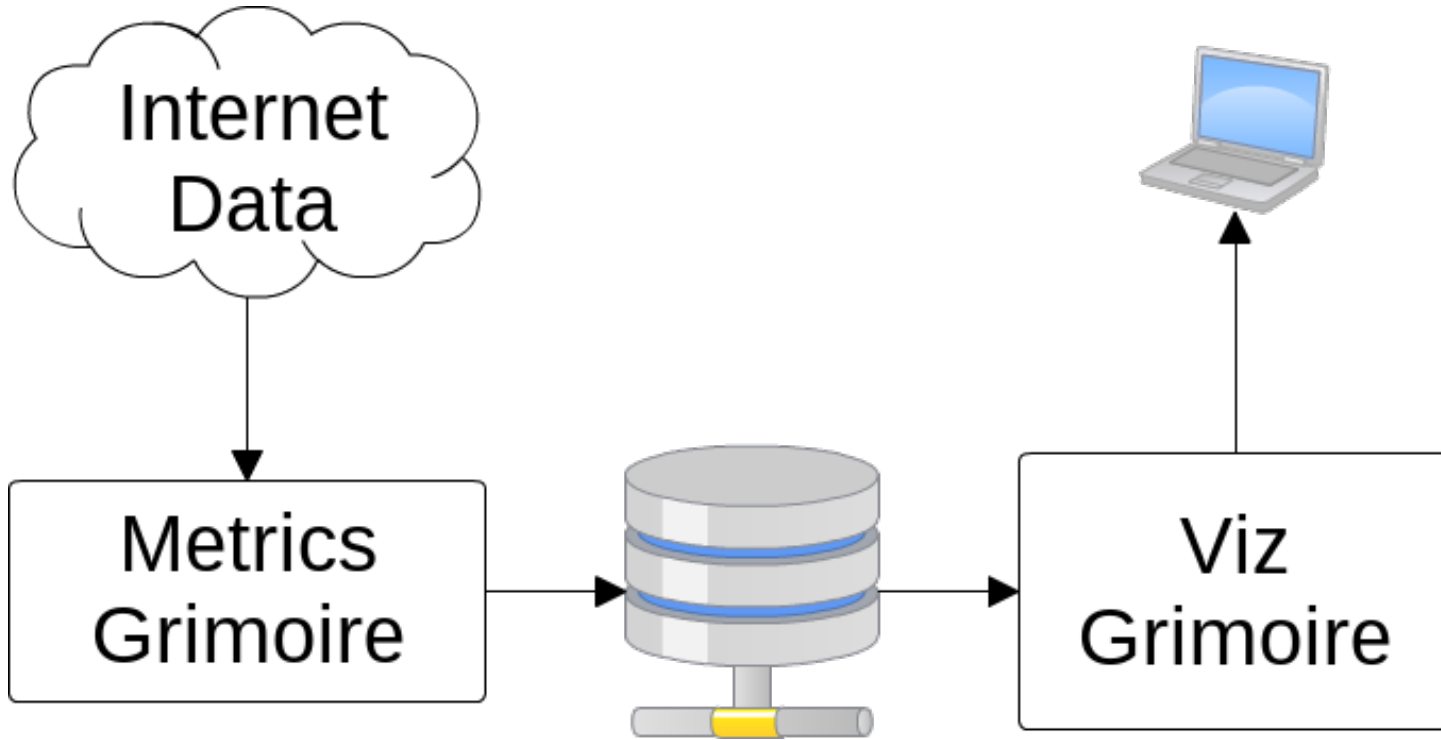
## Milestones:

- Look for existing software (Grimoire).
- Try to deploy existing software.
- Analyse how to extend existing software.

# Non-functional requirements

- Measuring the characteristics of the community
- Scaling those characteristics
- Visualizing relationships

# Grimoire



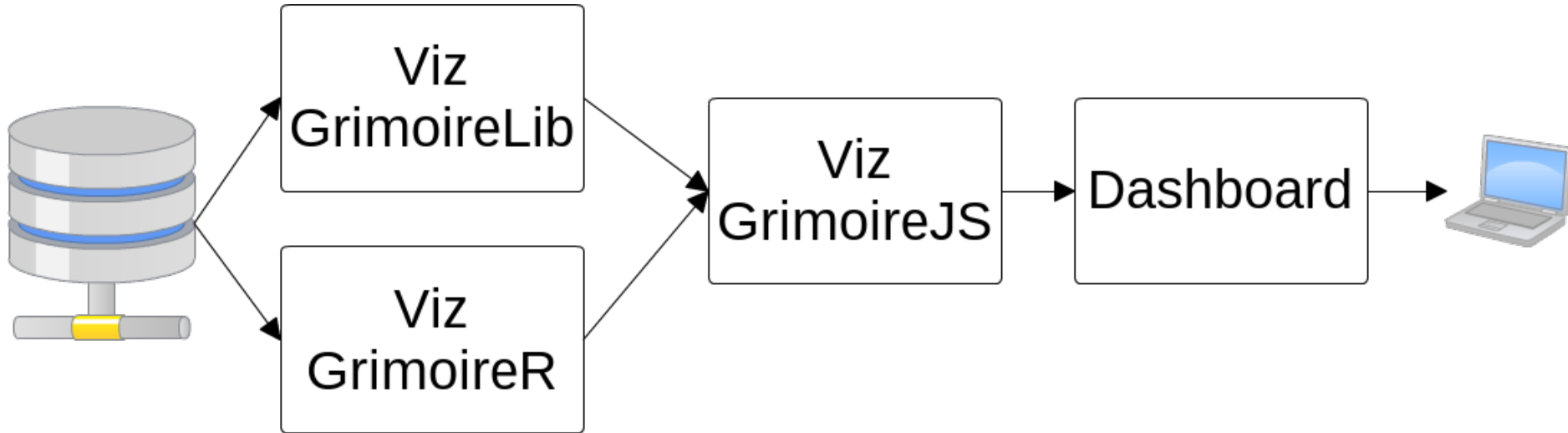
# Metrics Grimoire



Collection of software filling a unique database:

- CVSanaly: code repositories
- Bicho: bug-tracker
- Sibyl: Q&A websites
- IRCAnalysis
- CMetrics: source code analysis

# Viz Grimoire



# Dashboard example



68,428 commits

817 developers

315,252 mail messages

## Code Developers

817

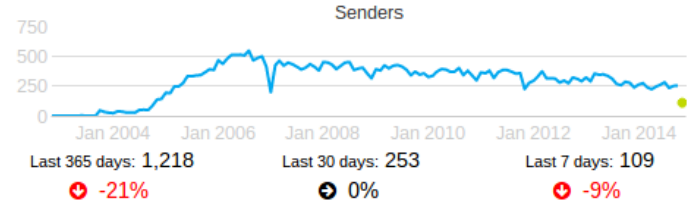
Core	Regular	Casual
35	96	681



## Discussion Participants

13,131

Thread Initiators	First Repliers
11,302	8,159



# Grimoire related risks



- Poor documentation.
- Really complex to deploy.
- Is it extendable?



Open source is good for me. I will fully embrace it.  
Open source is good for me. I will fully embrace it.  
Open source is good for me. I will fully embrace it.  
Open source is good for me. I will fully embrace it.  
Open source is good for me. I will fully embrace it.  
Open source is good for me. I will fully embr



# Questions?