RuleML' 14

Using Rules to Develop a Personalized and Social Location Information System for the Semantic Web

Iosif Viktoratos[1], Athanasios Tsadiras[1], Nick Bassiliades[2],

[1]Department of Economics, [2]Department of Informatics,
Aristotle University of Thessaloniki
GR-54124 Thessaloniki, Greece
{viktorat, tsadiras, nbassili}@auth.gr

Contents

- Location Based Social Networking Services (LBSNSs) & Context
- Semantic Technologies
- The System Geosocial SPLIS
  - Design and General idea
  - Geosocial SPLIS's Features
  - Geosocial SPLIS's Architecture
  - Geosocial SPLIS's processes
- Usage Scenarios
- Evaluation
- Conclusions
- Future work

Location Based Social Networking Services (LBSNSs)

- One of the most important sectors of LBS
- Used daily by millions of people
- Provide users with the capability to locate each other and interact with one another depending on their physical distance
- Facebook Places, Foursquare…

LBSNSs & Context

- Should offer information to users, relevant to their situation - context
  - Time, Location, User Preferences, Relationships…
- Context awareness enables proactive personalized information of higher quality
- Researchers and industries focus on contextual knowledge
  - Hardware (e.g. GPS, sensors)
  - Software (e.g. ontologies,rules)

Semantic Technologies

Ontologies & Rules

- Ontologies
  - Represent physical entities and their associations
  - Formal representation standard
    - Reusability
    - Interoperability
    - Flexibility
    - Knowledge sharing
- Rules
  - Extensive reasoning capabilities
  - Autonomy – Proactiveness
  - Intelligence

GeoSocial SPLIS

Geosocial Semantic Personalized Location Information System

- What?
  - A personalized LBSNS which connects user defined preferences (regarding POIs) with those of their nearby friends and POI owners' group targeted offers
- Why?
  - To provide proactive, customized and contextualized information
- How?
  - Combining semantics with LBSNSs

Design and General idea

- Human mobility behavior is not completely random
- Regular users have preferences/daily patterns
  - If it is Saturday noon I would like some restaurants that serve Italian cuisine
- POIs adopt a rule-based policy to deploy their specific marketing strategy
  - A museum offers 15% discount to students on Fridays
- The service collects user's context
- Combines all the above and presents personalized offers on Google Maps

Geosocial SPLIS's Features (1/2)

- Collects data from external sources
  - Google+, Google Places API, POI websites
- Regular users add contextualized rule based preferences via a web editor
- POI owners add group targeted offering policies via a web editor
- Data from editor → RuleML → Jess →Sesame
- Executes and evaluates data and rules on the fly
- Uses Google Maps for visualization

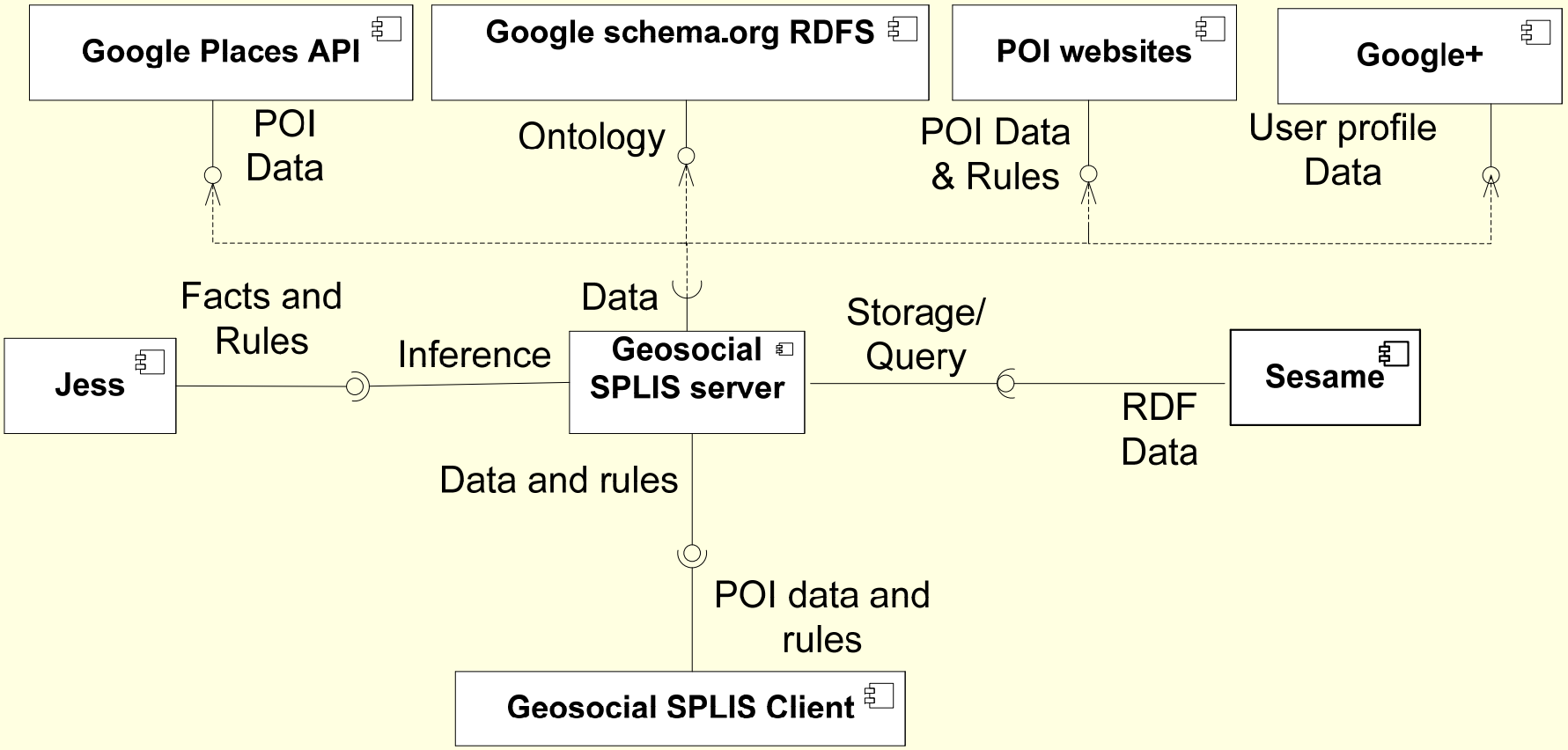
Geosocial SPLIS's Features (2/2)

- Rule conditions
  - LBS context
    - Location (e.g. within 800m)
    - Weather (e.g. sunny, rainy etc.)
    - Time (e.g. between 13:00-17:00)
    - Day (e.g. Monday)
  - Every existing property of a POI
    - E.g. cuisine currently serves
- Rule consequences
  - Add a place in a recommendation list

Geosocial SPLIS's Architecture (1/2)

- Client
  - PC browser-based
    - Html, JavaScript, Css
    - Google Maps
- Server
  - Java Server Pages (JSP)
- RDF data management
  - Sesame
- Rules
  - Reaction RuleML → XSLT → Jess

Geosocial SPLIS's Architecture (2/2)



Geosocial SPLIS's processes(1/10)

- Presentation of Information process
  - Data collection
  - Data retrieval
  - Rule evaluation
  - Presentation of personalized information
- Processes concerning rules
  - Rule insertion through editor
  - Rule modification process
  - "Get a rule process"
- Processes exploiting social ties
  - Common social interaction processes (e.g. friendships)
  - Nearby friends' mode

Geosocial SPLIS's processes(2/10)

Presentation of Information process

- Data collection process
  - Load the schema.org ontology
  - Get user data either from a registration form or from Google+ account
  - Retrieve POIs from Google Places API
  - Store data for POIs into Sesame
- Data retrieval
  - Fetch user's profile data and rules (if any)
  - Calculate contextual property values
  - Fetch POIs' property values and rules (if any)

Geosocial SPLIS's processes(3/10)

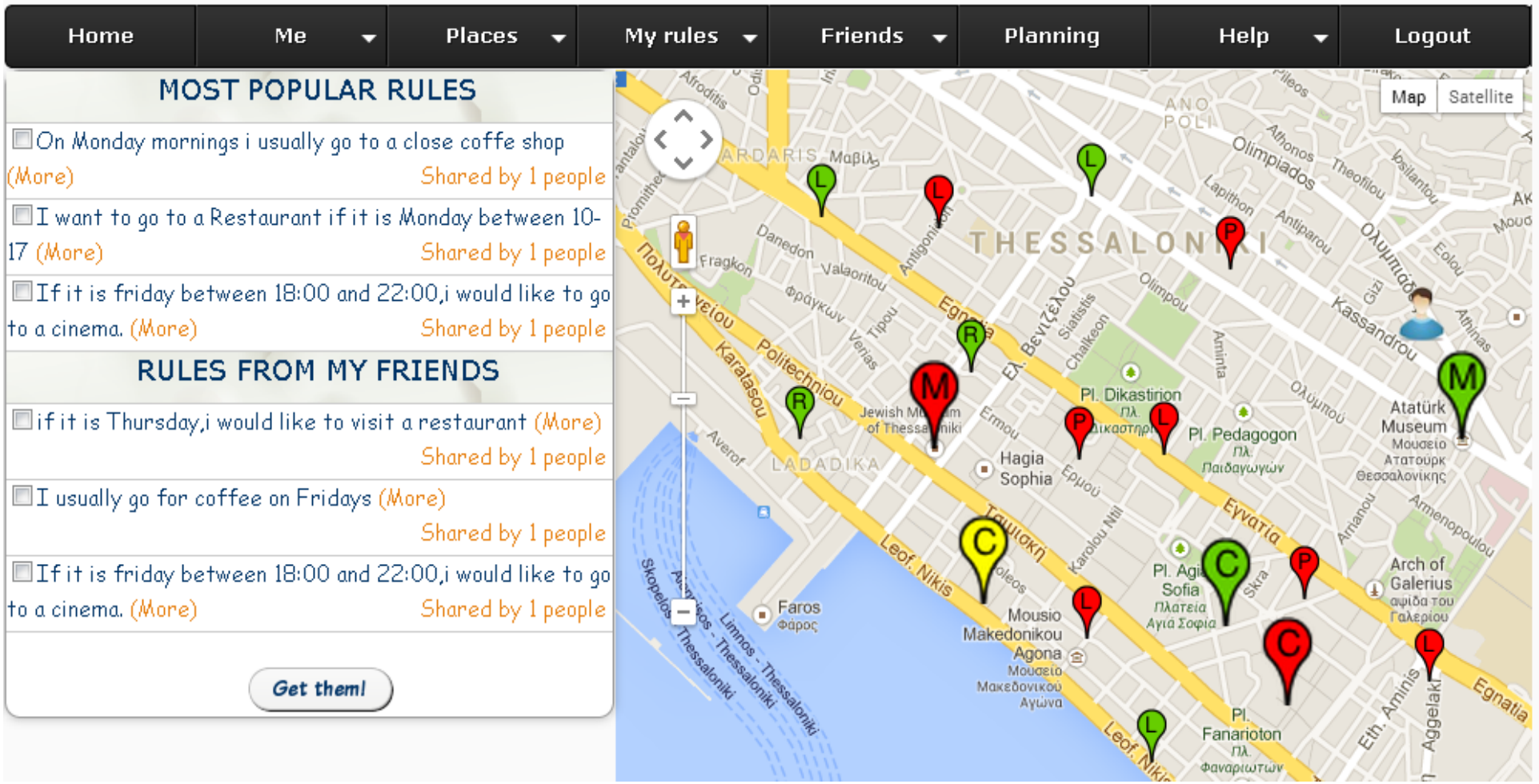
Presentation of Information process

- Rule evaluation
  - Assert data and rules to Jess
  - Evaluate rules using the asserted facts
  - Send evaluated data to server
- Presentation of personalized information
  - Bigger in size marker → recommended POI
  - Green marker → at least one valid offer for the user
  - Yellow marker → no rule is fired for the current user
  - Red marker → no offers at all
  - Red star → POI owner

Geosocial SPLIS's processes(4/10)

Presentation of Information process

- PC browser-based version



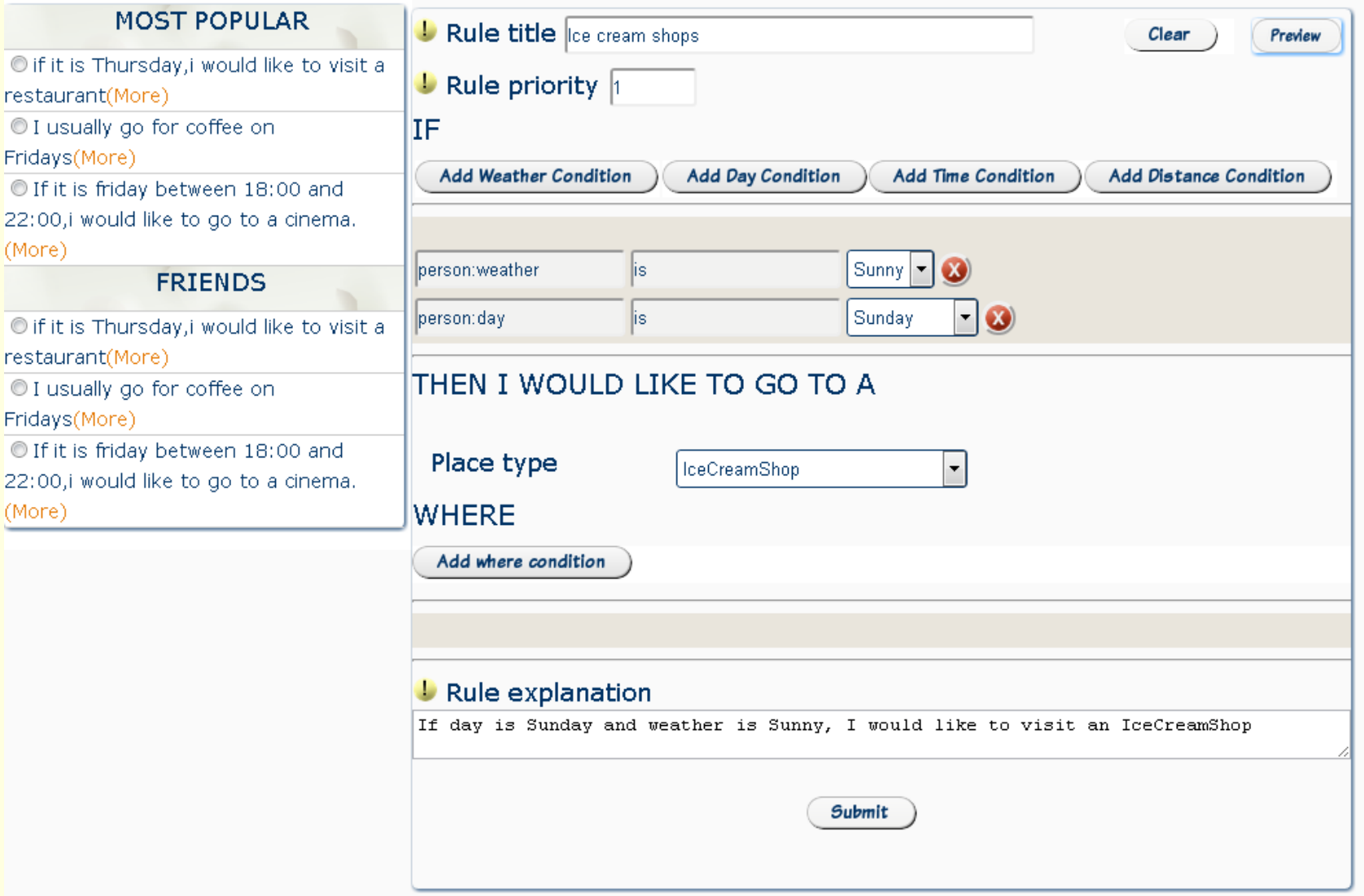
Geosocial SPLIS's processes(5/10)

Rule insertion through editor

- Rule title and the priority fields
- Four "Add …. Condition" buttons
  - Each one for the corresponding contextual condition
  - The condition customization consists of three elements:
    - The property field (weather, day, time, distance)
    - The operator field ("is" and "<",">" for time and distance)
    - The value
    - An "AND" is implied among them
- Select POI category
  - "Add Where Condition" button to add more conditions regarding POI properties
- Add a textual explanation
- Assert the rule "If day is Sunday and weather is Sunny, then I would like an IceCreamShop"

Geosocial SPLIS's processes(6/10)

Geosocial SPLIS's Web Editor



RuleML representation

```
<?xml version="1.0" encoding="UTF-8"?>
- <RuleML>
  - <Assert>
    - <Rule style="active">
        <label>drzgjtgt </label>
        <explanation> If day is Sunday and weather is Sunny, I would like to visit an IceCreamShop </explanation>
      - <if>
        - <And>
          - <Atom>
              <Rel>place</Rel>
            - <slot>
                <Ind>type</Ind>
                <Ind> IceCreamShop </Ind>
              </slot>
            - <slot>
                <Ind>uri</Ind>
                <Var>id</Var>
              </slot>
            </Atom>
          - <Atom>
              <Rel>person</Rel>
            - <slot>
                <Ind>day</Ind>
                <Ind>sunday</Ind>
              </slot>
            - <slot>
                <Ind>weather</Ind>
                <Ind>sunny</Ind>
              </slot>
            </Atom>
          </And>
        </if>
      - <then>
        - <Assert>
          - <Atom>
              <Rel>recommendation</Rel>
            - <slot>
                <Ind>id</Ind>
                <Var>id</Var>
              </slot>
            </Atom>
          </Assert>
        </then>
      </Rule>
    </Assert>
</RuleML>
```

Jess representation

```
(defrule kctysfvn (declare (salience 1))
(place( type IceCreamShop) ( uri ?id))
(person ( weather sunny) ( day sunday))
=>
(assert (recommendation( id ?id)))
(store EXPLANATION "If day is Sunday
and weather is Sunny, I would like to visit
an IceCreamShop"))
```

Geosocial SPLIS's processes(8/10)

Rdf data representation

Geosocial SPLIS: policy19d883ef-f735-4521-a8a6 771065b1b2a8	**rdf:type**	**Schema:Policy;**
	schema: policy_description	**IF person:weather is Sunny AND person:day is Sunday THEN I WOULD LIKE TO GO TO A place:type IceCreamShop**
	schema: policy_explanation	**If day is Sunday and weather is Sunny, I would like to visit an IceCreamShop**
	schema: policy_priority	**1**
	schema: policy_link	**platon.econ.auth.gr/.....ruleml**
schema: Person22	**schema:Policy**	**Geosocial SPLIS: policy19d883ef-f735-4521-a8a6 771065b1b2a8**

Geosocial SPLIS's processes(9/10)

Processes concerning rules

- Rule modification process
  - Edit and delete a rule through editor
- "Get a rule" process
  - Get rules from other users
  - Check and get the rule
  - In case of editing a rule, a user is simply "unlinked" from the rule so that not to affect other users

Geosocial SPLIS's processes(10/10)

Processes exploiting social ties

- Common social interaction processes
  - Messages, Friendships' creations, View profiles…
- Nearby friends' mode
  - Collects user's and his/her nearby (logged in) friends' rules and context
  - Evaluates rules and fetches recommended nearby POIs
  - Gets POIs' rules and evaluates them concerning all users' contexts
  - Provides group-based information

Usage Scenarios(1/10)

Normal usage

- A scenario concerning two different users

Name	Age	JobTitle	Time	Day	Weather	Location
John	20	Student	13:45	Saturday	Sunny	Location A

Name	Age	JobTitle	Time	Day	Weather	Location
Mary	21	Student	13:45	Saturday	Sunny	Location B

Usage Scenarios(2/10)

Normal usage

- Possess the following rules

	John's rules	Mary's rules
Rule 1	"If it is Saturday between 13:00 and 16:00, I would like to go for coffee "	"I would like to go for coffee, if weather is Sunny and time is before 18:00 o'clock"
Rule 2	"If it is Wednesday and time is after 18:00, find me cinemas which are closer than 1000 m"	"If it is Friday between 19:00 and 22:00, find me some Restaurants which serve Italian cuisine"
Rule 3	"On Saturday afternoons (12:00-15:00), recommend me a Museum"	_

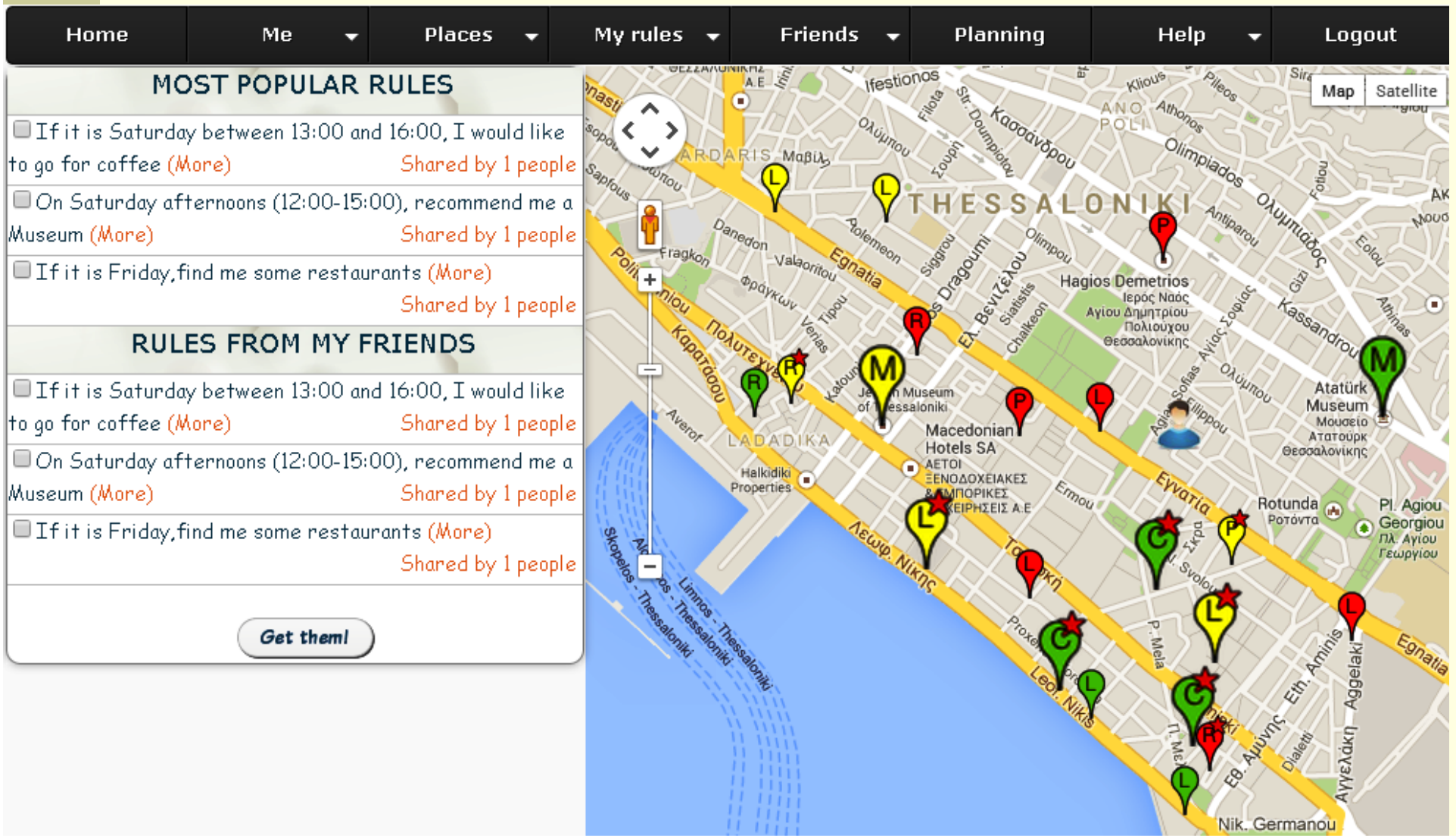
Usage Scenarios(3/10)

Normal usage-John

- Consider John
- John's rules 1 and 3 are fired because it is Saturday and time is 13:45
- Coffee shops and museums are represented with a bigger marker
- Green markers indicate that there is a valid offer for him
- Can get rules from his friends or from other users

Usage Scenarios(4/10)

Normal usage-John



Usage Scenarios(5/10)

Normal usage-John

- POI “Friends Cafe” is represented with a bigger marker because it is a Coffee shop (rule 1)
- John has a valid offer as a student (POI owner’s rule)

Usage Scenarios (6/10)

Friends' mode-John

- Mary is his only nearby friend which is logged in at this time
  - Gets his and Mary's context and rules
  - Evaluates rules and fetches nearby recommened POIs
  - John's rule 1 and 3 are fired and as a result museums and coffee shops are recommended
  - Mary's rule 2 is fired, which recommends coffee shops.
  - Gets POIs' rules and evaluates them based on John and Mary's contexts
  - Presents information

Usage Scenarios (7/10)

Friends' mode-John

- Presentation of information
  - Red marker → POI that does not have any offer at all
  - Yellow marker → POI that has at least one offer, but it is not valid for none of them
  - Half yellow-half green → POI has a valid offer for at least one of the friends or the user
  - Green → POI has an offer for all of them
  - A bigger marker → POI is recommended by a user rule and at least one of his/her friends' rules

Usage Scenarios (8/10)

Friends' mode-John

1) displays your position

2) displays your friend's position

3) indicates that a place does not contain any offers

4) indicates that a place contains offers, but they do not stand for you, or for your nearby friends

5) indicates that a place contains offers, that they stand for at least one of you,but not for everyone

6) indicates that a place contains offers, that they stand not only for you,but also for all your nearby friends

7) A bigger marker indicates that this POI interests you and at least one of your friends

Rules fired now Close

Mary:"I would like to go for coffee, if weather is Sunny and time is before 18:00 o'clock"

My rule:"If it is Saturday between 13:00 and 16:00, I would like to go for coffee "

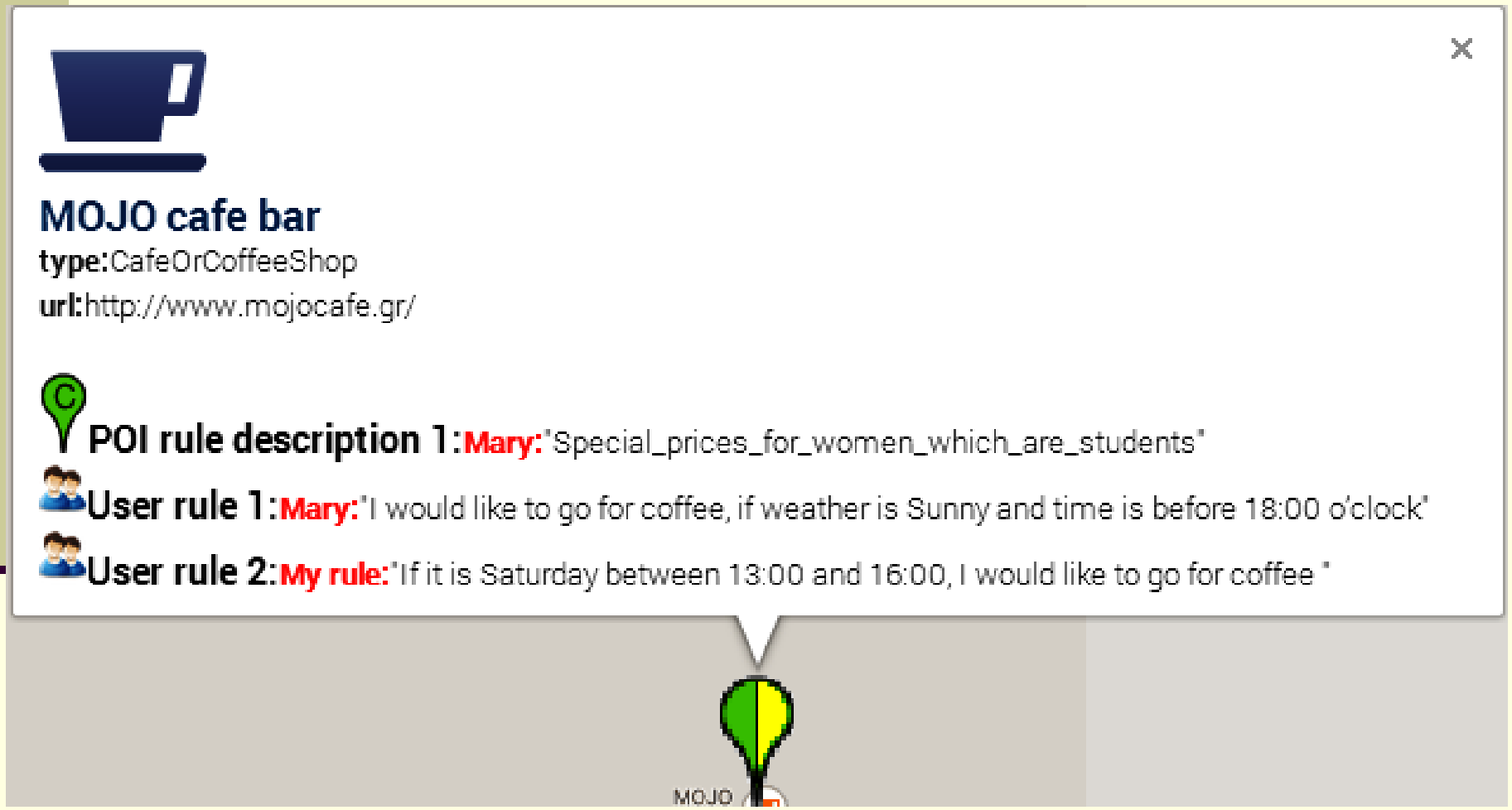
My rule:"On Saturday afternoons (12:00-15:00), recommend me a Museum"

Usage Scenarios (9/10)

Friends' mode-John

Usage Scenarios (10/10)

Friends' mode-John



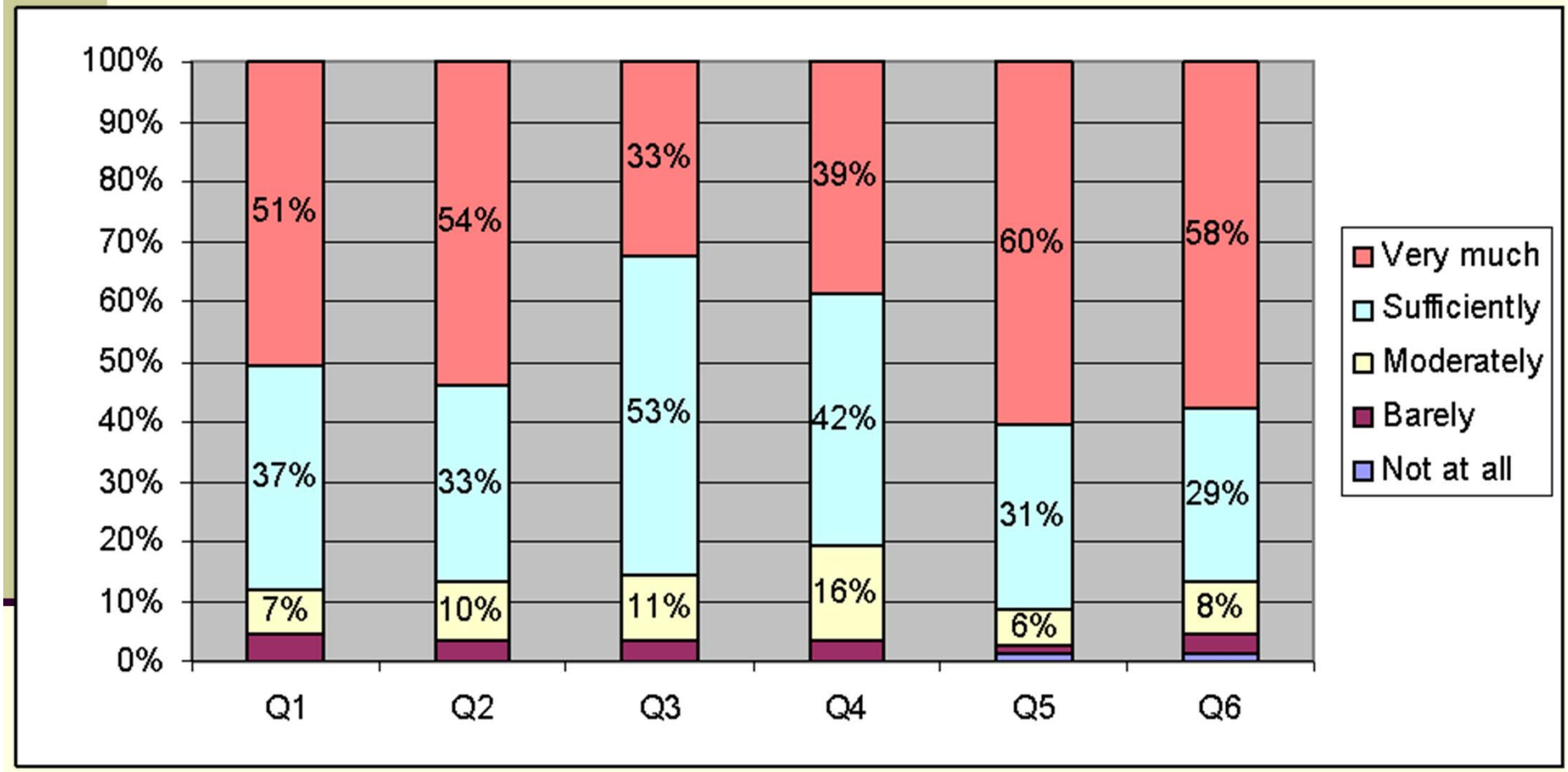
Evaluation(1/6)

- 83 undergraduate students of Economics (18-22 years old, both genders)
- Three parts of questions about:
  - Processes concerning rules and the personalization of information
  - Social processes
  - The system in general

Evaluation(2/6)

- Q1. How easy was to add a rule?
- Q2. How easy was to modify a rule?
- Q3. Are you satisfied with the provided interface?
- Q4. How easy was to find and get a rule from another user?
- Q5. How easy was to understand why a place was recommended?
- Q6. How easy was to find a place that resulted by your rules and had an offer for you?

Evaluation(3/6)

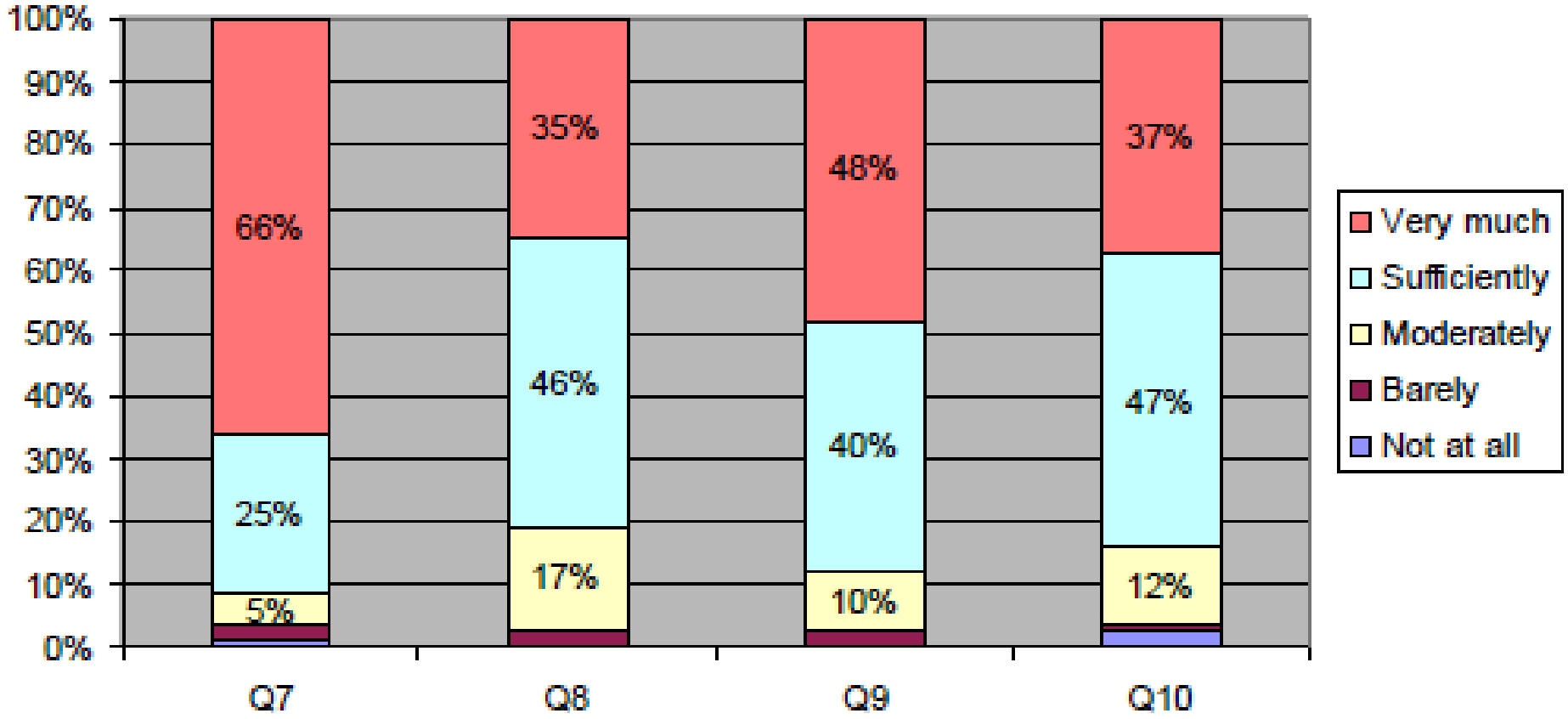


- For every question, over 80% of the answers were “sufficiently satisfied” or “very much satisfied”

Evaluation(4/6)

- Q7. How easy was to send a friend request?
- Q8. How easy was to understand which of your friends recommend a place and why?
- Q9. How easy was to find common places for you and your friends?
- Q10. How easy was to find places that resulted by your friends' rules and had an offer for you?

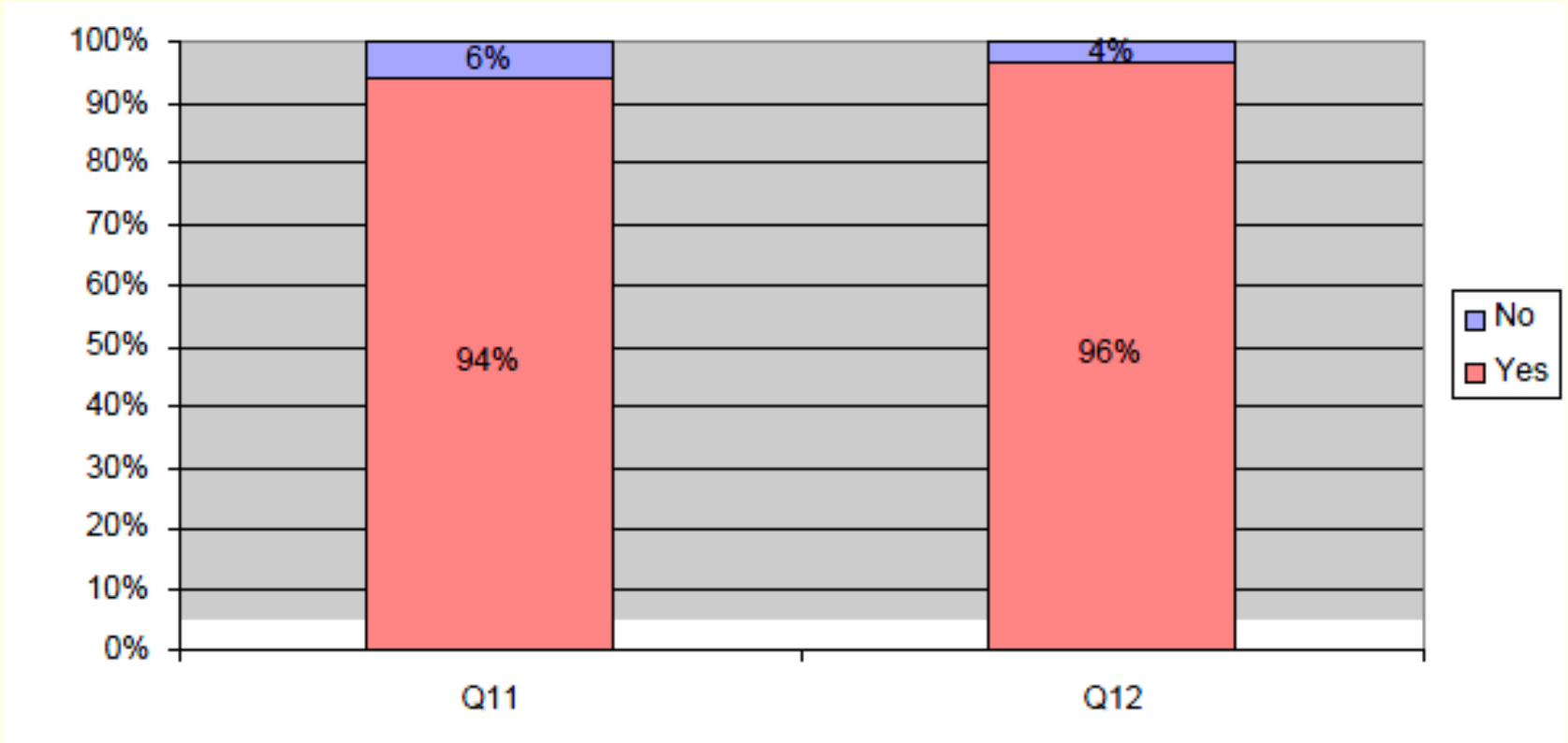
Evaluation(5/6)



- For every question, over 80% of the answers were “sufficiently satisfied” or “very much satisfied”

Evaluation(6/6)

- Q11.Will you continue using the system?
- Q12.Would you recommend the system to your friends?



Conclusions

- Geosocial SPLIS models human daily preferences and connects them with POI owners group based offering policies
- Advantages
  - POI owners have highly targeted marketing audience
  - Regular users enjoy proactive contextualized information
  - Adding rules dynamically leads to more customized and personalized user experience
  - The more rules were added to the system, the more interesting/intelligent it becomes
  - Exploit people collaboration and social intelligence to create large knowledge bases

Future work

- More extensive user evaluation
- Expand the editor to add a wider range of preferences (movies, music etc.)
- Rules (semi-)automatically induced by mining users' logs, likes or reviews
- Connect other social media sources (facebook, twitter etc.)

Thanks for your time!

Geosocial SPLIS is available at: http://tinyurl.com/GeoSPLIS

Mobile version is available at: http://tinyurl.com/GeoSoSPLIS