GEOfox

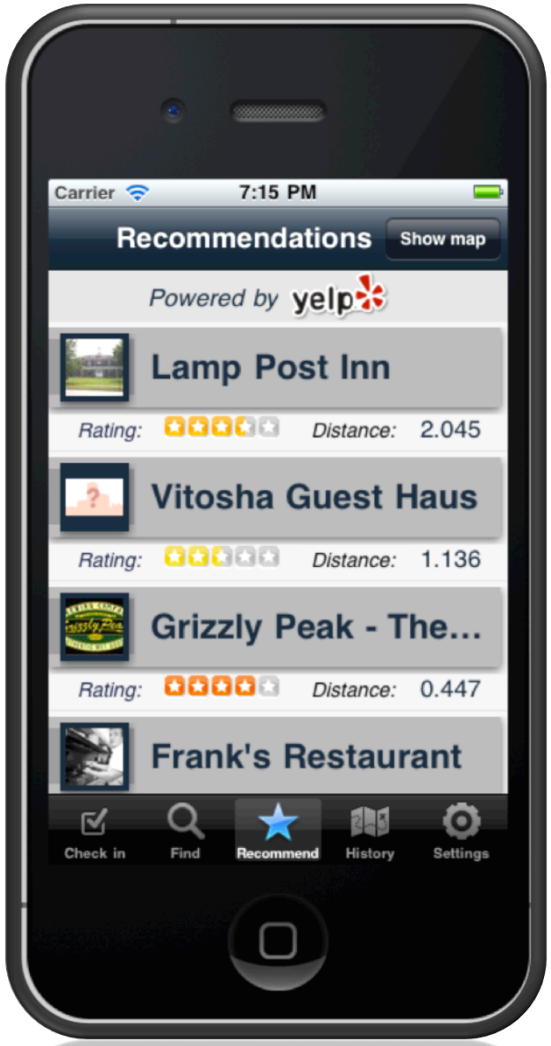
Rusty Dekema
Matt Colf
Mike Brown
Adam Budde
Mike Billau

Problem

- Hard to find new places
- Current check-in applications do very little with gathered data
- Friend based applications struggle to provide fresh content

Solution

- Place recommendations via aggregate user data
- List local places to explore by category
- Extensible framework for additional features and platforms

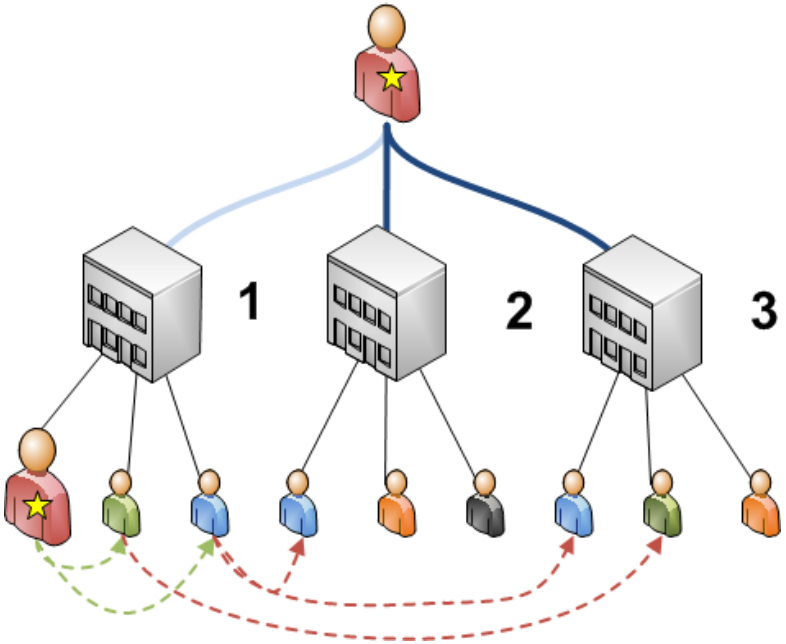


iPhone

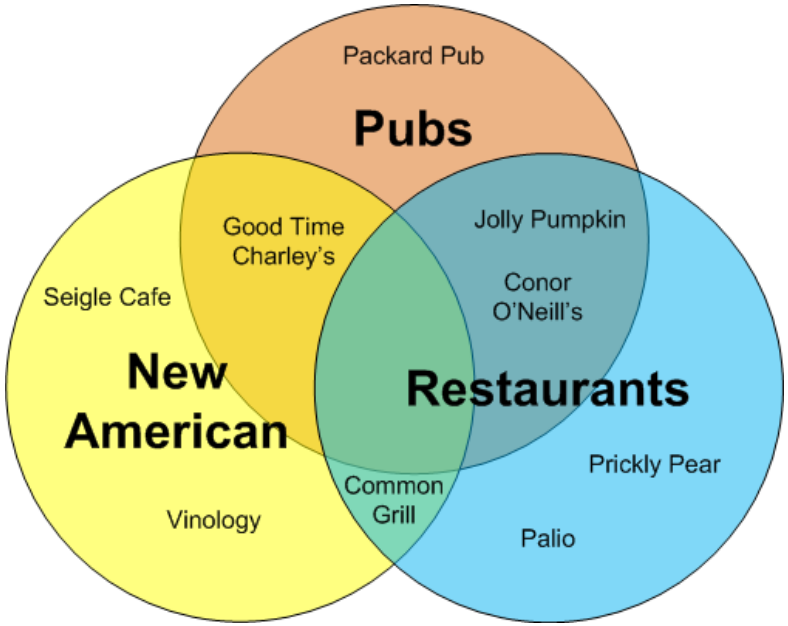
android

Recommendations

User Clustering



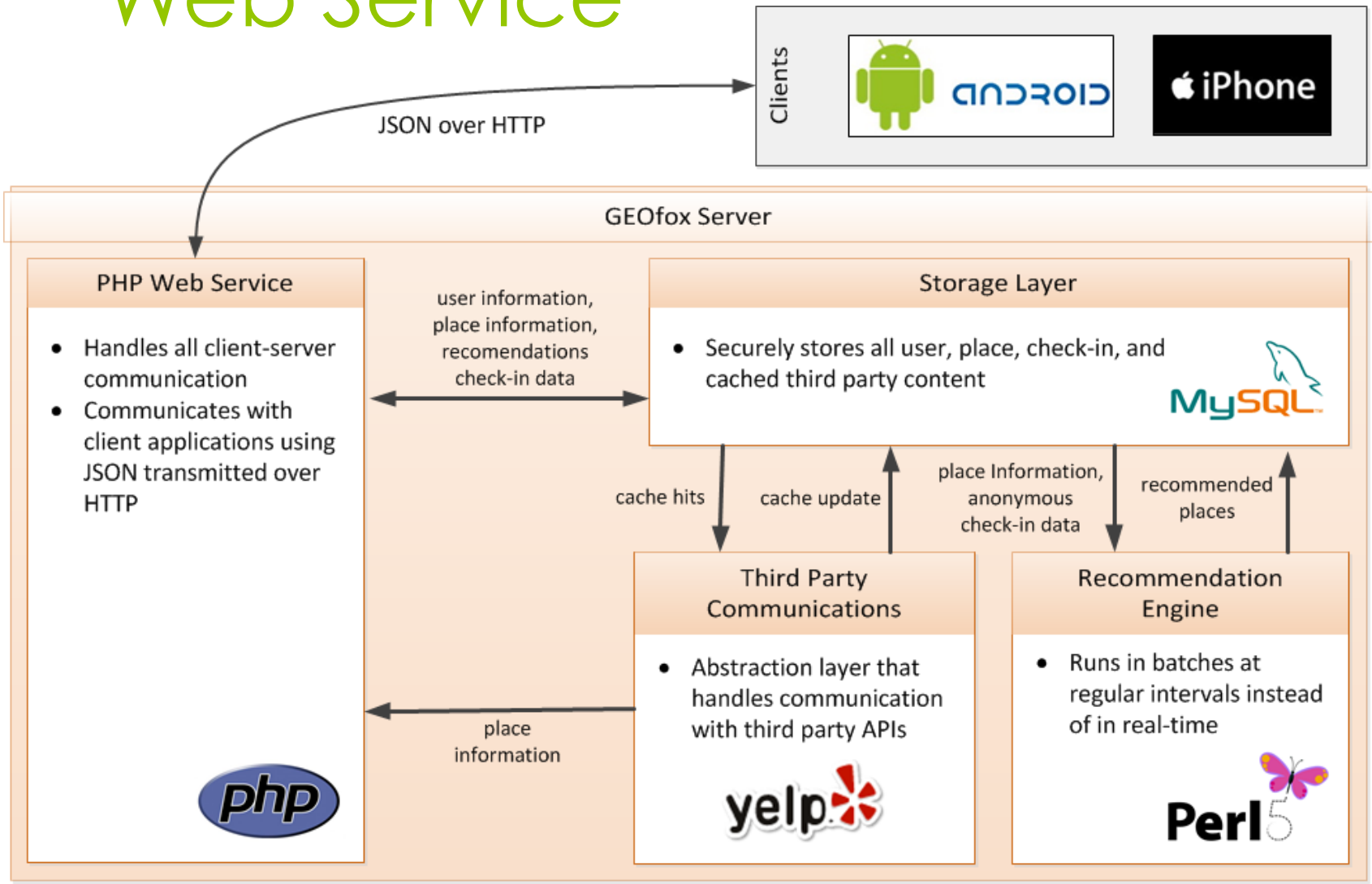
Category Correlation



Web Service

- Centralized application logic
- Lightweight clients
- Retrieves place data from the Yelp API
- Easily extensible private API

Web Service


Clients
android
iPhone
JSON over HTTP
GEOfox Server
PHP Web Service
Handles all client-server communication
Communicates with client applications using JSON transmitted over HTTP
php
user information, place information, recomendations check-in data
Storage Layer
Securely stores all user, place, check-in, and cached third party content
MySQL
cache hits
cache update
place Information, anonymous check-in data
recommended places
Third Party Communications
Abstraction layer that handles communication with third party APIs
yelp
place information
Recommendation Engine
Runs in batches at regular intervals instead of in real-time
Perl 5

Application Demonstration

Changes & Challenges

- Scope Changes
  - Application lacked a clear focus
  - Removed extraneous features
- Challenges
  - Server response times matter
  - Slow & limited Yelp API responses

Competition

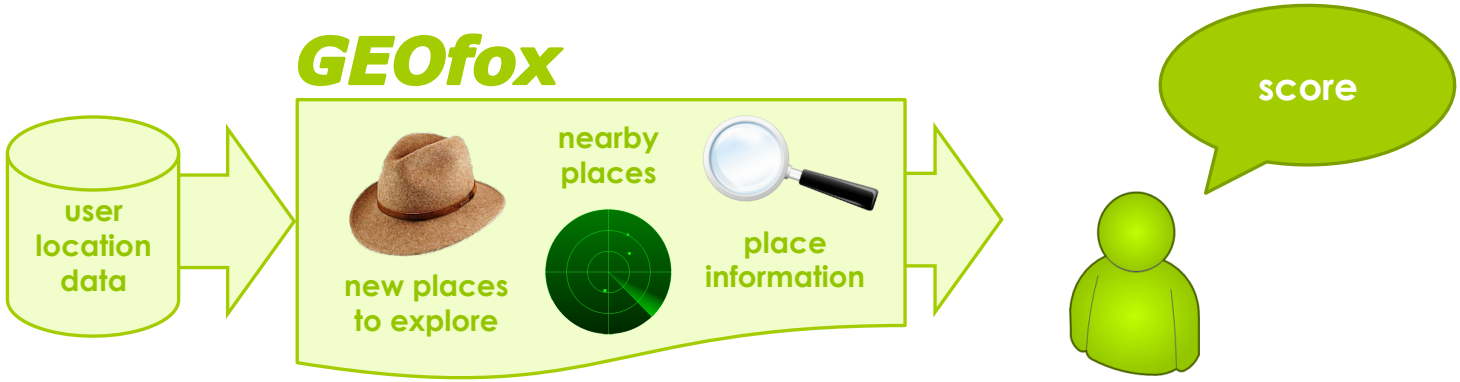
- We think that location based networks are the next "big thing"
- Recent competition
  - Google hotpot
  - Facebook Places
  - Yelp Check-ins

Facebook Places
Who. What. When. And now **where.**

Google hotpot

Secret Sauce

- Providing place recommendations
- Fresh content from aggregate data
- Extensible framework



Future Development

- Change data provider
- Extend recommendation algorithm
- Social network integration
- Spin-off applications

Maintenance Release 1.5

Google places

change data provider

Feature Release 2.0

facebook

twitter

social network integration

Questions?

www.geofoxapp.com
geofoxapp@umich.edu

Rusty Dekema
Recommendations

Adam Budde
iPhone Development

Mike Brown
Android Development

Matt Colf
Infrastructure & Server Development

Mike Billau
Web & Android Development

Supplemental Material

Detailed content that did not fit in the presentation

Video Demonstrations

Android Application
beta release

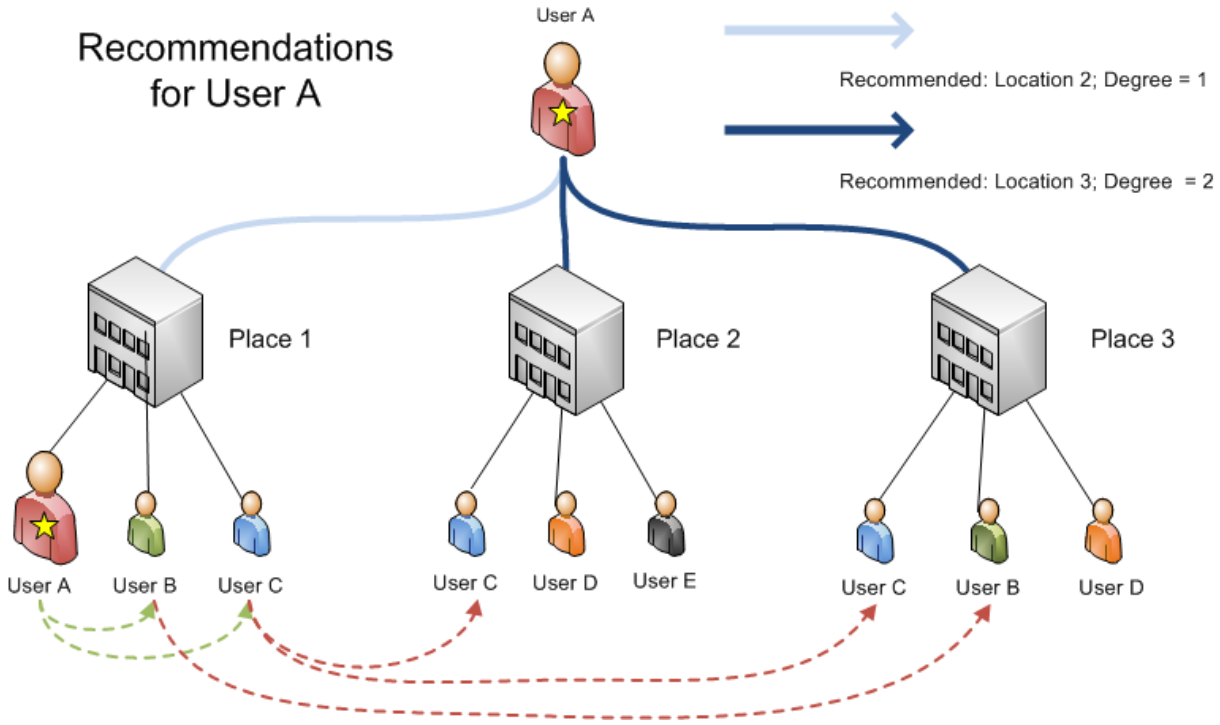
- Youtube: http://www.youtube.com/watch?v=O_0cpKY6yi8
- Download: http://svn.geofoxapp.com/docs/presentations/videos/androiddemofinal.mp4

iPhone Application
final release

- Youtube: http://www.youtube.com/watch?v=JPQH31rZL3M
- Download: http://svn.geofoxapp.com/docs/presentations/videos/GEOfox_iPhone_demo.mp4

Recommendations

User Clustering Details



- Recommendations are found by following similar user trends.
- Users B and C commonly check into Place 1. Since User A does the same, Places 2 and 3 are suggested to User A because those users also check in there.
- Place 3 would be suggested higher because two similar users check in there.

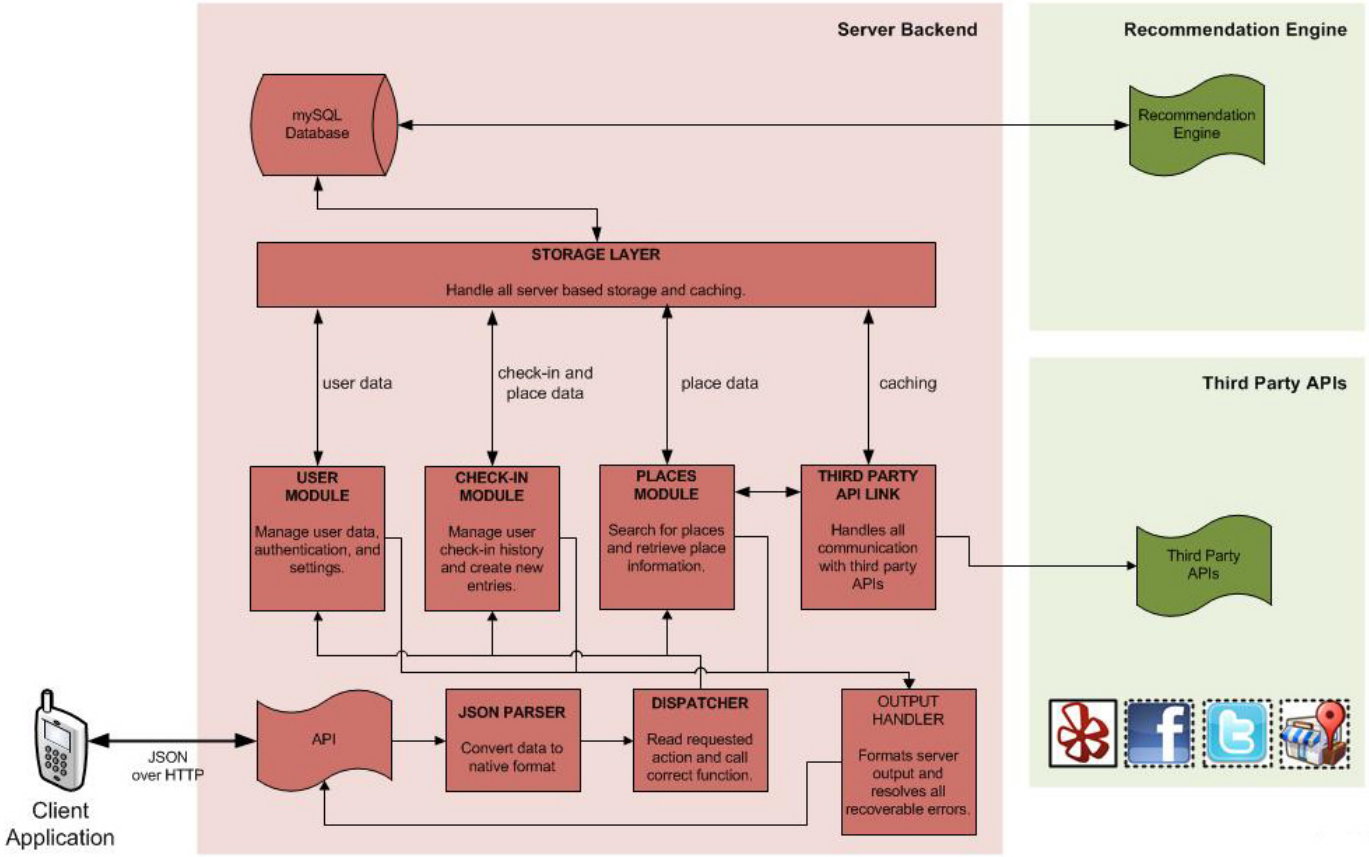
Recommendations

Category Correlation Details

Place A							
Category Matches						Recommendations	
Category 1	R 1.5	Category 2	R 2.0	Category 3	R 5.5	Similar Places	User Ranking
✓		✓		✓		B	9.0
		✓		✓		C	7.5
				✓		D	5.5
✓		✓				E	3.5
		✓				F	2.0
✓						G	1.5
						H	0.0

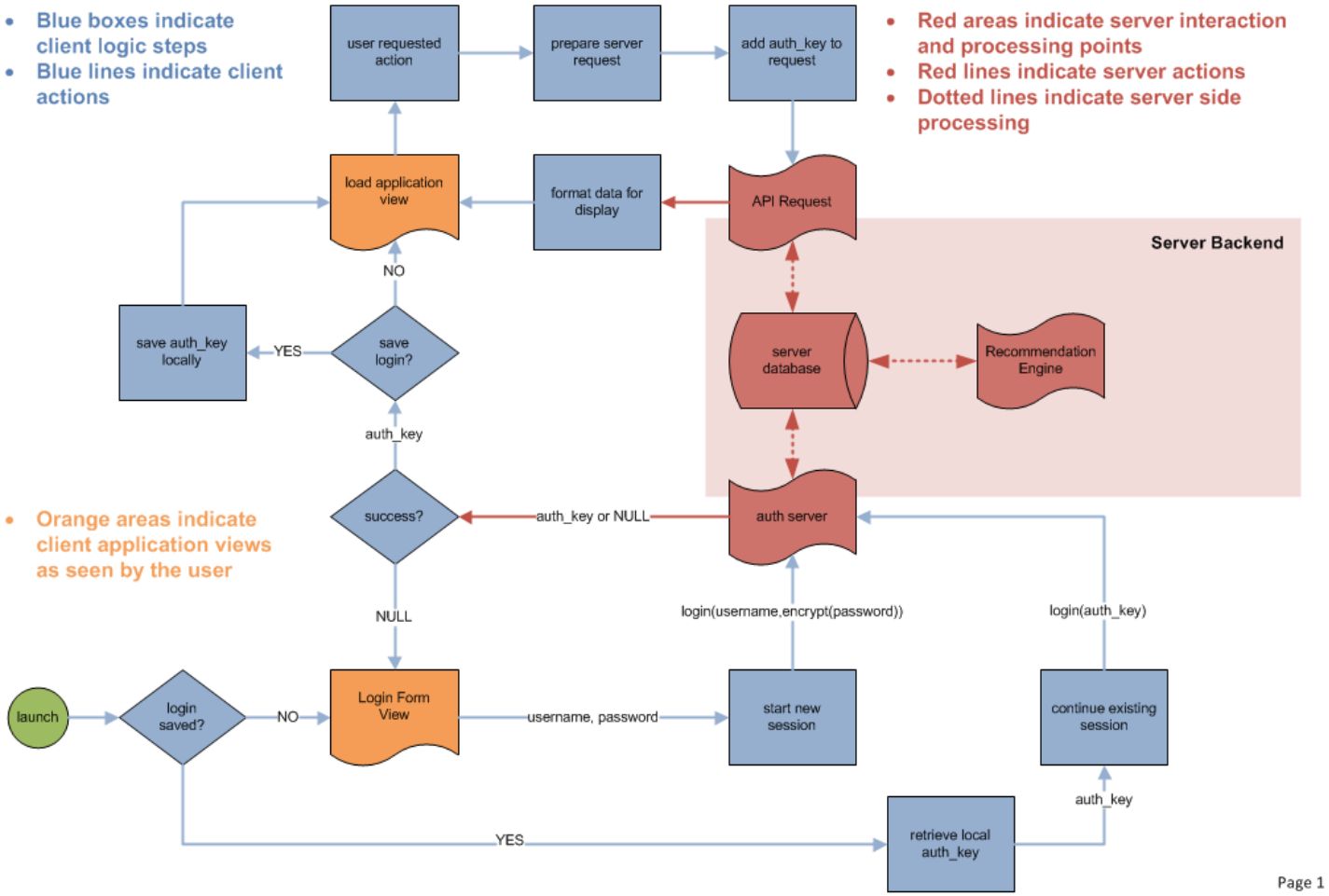
- Each place is assigned up to 3 categories (bar, restaurant, pub, etc.)
- Category R values are based on how many times the user has checked into places that have that category (how well the user likes that category)
- Recommendations are found by finding places with similar categories and then sorting/ filtering by summing the matching category R values for that user

Server Architecture



- This diagram shows the code breakdown of the server architecture.
- Modules are loaded dynamically to reduce the memory footprint.

Data Flow Model



- Shows how data flows between the clients (blue) and the server (red).