



Functional Specification

for

mobeegal v1.0.0

find stuff closer

Version 0.8 Draft

Muthu Ramadoss

created - **Apr 01, 2008** / updated - **May 02, 2008**

1. INTRODUCTION

This specification is simply a starting point for the design of mobeegal v1.0, not a final blueprint. As we start to build the content, we'll discover a lot of things that won't work exactly as planned. We'll invent new features, we'll change things, we'll refine the wording, etc. We'll try to keep the spec up to date as things change. By no means should you consider this spec to be some kind of holy, cast-in-stone law.

mobeegal works on the following premise:

- Tell mobeegal what you are - iStuff.
- Tell mobeegal what you want - uStuff.
- mobeegal gives you the perfect match - mStuff.

mobeegal is the revolutionary search engine which will help mobeegal users "find stuff closer" the mStuff by matching iStuff "what you are" with uStuff "what you want". Simple, yet Relevant!

This is the functional specification for mobeegal 1.0 software. The core functionality for the mobeegal 1.0 system are:

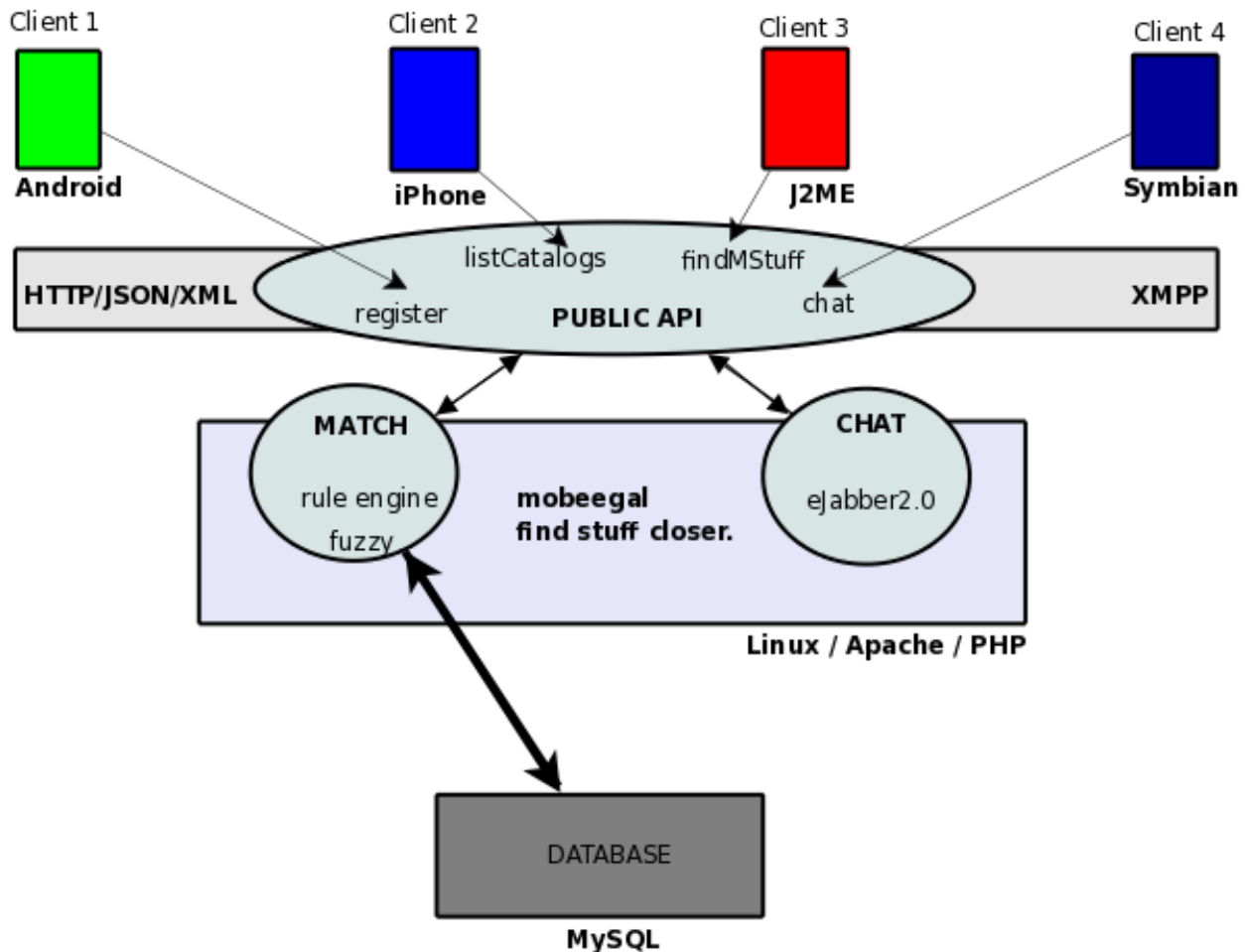
- "find stuff closer" to wherever you are.

mobeegal will be powered by the mobeegal server, and by the mobeegal UI on the client. mobeegal can be customized with User requested mobeegal catalogs like dating, shopping, jobs, business, restaurants, movies etc., The mobeegal user can remain private and anonymous or public. mobeegal will be available worldwide, for all audience, for all age groups in his/her favorite languages.

1.1 Roadmap

- V1.0.0-Alpha - Apr 30
- V1.0.0-Beta - May 30
- V1.0.0-Final - Jun 30

2.OVERVIEW



mobeegal is a revolutionary search engine to “find stuff closer” developed by IntelliBitz Technologies, Chennai, India.

Here's a brief summary of how mobeegal works:

- mobeegal server will be hosted and up and running on the public Internet. The mobeegal server has the most important mobeegal search engine running in it. The mobeegal server also has a mobeegal chat engine to enable clients to chat with each other.
- mobeegal client can be downloaded and installed on a user mobile. The mobeegal user running mobeegal client, will be supported by a mobeegal service which will connect to server as per the user preferences.

- User activates the mobeegal service in the mobeegal client.
- User can now optionally select and install mobeegal catalogs on his mobile. Additional modules will be delivered and installed on the user mobile on demand.
- User can now select the “catalog” he wants to search. There might be additional “category” available. Once a catalog/category is selected, the user can begin searching.
- User must enter an mStuff Query to begin his search. mStuff Query is a Query mechanism defined by mobeegal and is explained later. mStuff Query can be thought of as a combination of iStuff and the target uStuff for now.
- User can create many mStuff Queries spanning many catalogs.
- mStuff Query will be sent to the mobeegal engine based on user request. The mobeegal engine now cranks into action, and finds all “relevant” matches and returns mStuff results to the mobeegal client.
- mobeegal server maintains a mobeegal user profile for all mobeegal users. The system profile will be consulted in determining the relevancy of matches for the user.
- mobeegal client will let the mobeegal users chat in real time with the matches using the mobeegal chat engine running on XMPP protocol.

3. MAJOR FEATURES

- Find ANY “stuff” that you want, right where you are.
- Privacy and Security are top priority. User remains completely anonymous if he wishes to be private. All communication is secure.
- Localization and Internationalization will be available from version 1.0. User can search in their favorite languages.
- mStuff results can be stored in his mobile if the user wishes to keep them for further conversation. User can vote on mStuff results – like or dislike.
- mobeegal will be free and supported by paid ads from providers.
- User will be offered a premium paid service with no ads, where the provider will be authenticated and certified by mobeegal. The match authenticity and integrity will be guaranteed by mobeegal in premium paid service.

4. DESIGN GOALS

All design and engineering decisions will be taken with the following principle in mind:

Simple is better than complicated.

We will always try to eliminate choices that the user must make. Every bump on the user experience must be relentlessly sanded down. Our customers will say: "It's the easiest thing in the world and It Just Works." No jargon will remain in the user experience, anywhere.

5.DISTRIBUTION

The mobeegal distribution contains the following artifacts:

5.1mobeegal Client

Currently mobeegal supports the following clients:

1. Android mobile phones

mobeegal.apk – The mobeegal client application will install as an android software in mobile handsets running android operating system. Other mobiles and other o/s will be targeted for future release.

mobeegal-[catalog].apk – The mobeegal catalogs. This will represent a particular search catalog the user is interested in. ex – dating. There would be many such catalogs available and the users will be able to download and install on demand.

5.2mobeegal Server

The mobeegal server running on public Internet.

<http://38.105.84.198/>

5.2.1mobeegal Match Engine

- The mobeegal match engine will take the mStuff Query and return mStuff results. The search criteria will be applied as filters.
- The mobeegal match engine will consult the mobeegal system profile to make the match results more relevant to the user.
- The mobeegal match engine finally delivers the relevant mStuff results to the user.

5.2.2mobeegal Chat Engine

- The mobeegal chat engine will let the mobeegal users on the mobeegal systems to chat in real time.

5.2.3mobeegal Database

- MySQL will be used as the mobeegal database on the server.
- A database schema named 'mobeegal' will be used for mStuff storage.

6.ABOUT THE ROLLOUT

Our primary goal is to rollout a fully-functional version of mobeegal v1.0 by Jun 30, 2008. That means that the initial version will almost certainly not be optimal in every way and will not contain every feature. Throughout this spec, the notation (v2.0) will be used to indicate features which will not be a part of the initial release but which will be added later.

7.PHYSICAL ARCHITECTURE

The initial rollout will go onto web servers connected with public ip address -- high end, well-balanced infrastructure - each of which can handle the entire mobeegal load by itself if necessary. The system will be highly scalable so we can add additional servers progressively to handle more load. We will move rapidly to add servers if the service is popular.

The servers will be running Apache, PHP. For a database, the servers will use MySQL 5.X which they share with everything else.

The server is a VPS hosted by www.digitaldomainmd.com and the IP address is **38.105.84.198**. The support staff can be reached at support@digitaldomainmd.com

8.SOFTWARE ARCHITECTURE

The mobeegal server is an PHP application with code written in PHP5.x running in Apache. We will use MySQL 5.x as the database.

The mobeegal client will run on Android mobile. The UI will be build on Android using the latest Android SDK and with JDK1.6.0

8.1mobeegal client

- Android SDK version - **android-sdk_m5-rc15_linux-x86**
- Java SDK version - 1.6.0

8.2mobeegal server

- Apache Server version – 2.2
- PHP version – 5.2.6
- mobeegal match engine
- mobeegal chat engine – ejabber 2.0

8.3mobeegal database

- MySQL version – 5.1

8.4mobeegal client - server communication

The mobeegal client communicates on http/json protocol. The chat engine will run xmpp/jabber protocol.

The following HTTP/JSON methods are currently exposed by the server. **All HTTP/JSON method communication will be encrypted/decrypted for maximum security.**

NOTE: All data types are JSON.

1. register
IN – Unique Mobile ID
OUT – Unique mobeegal Client ID
2. listCatalogs
IN – [EMPTY]
OUT – All Catalogs
3. listCategories
IN - Catalog
OUT – All Categories
4. findMStuff
IN – mStuff Query
OUT – mStuff Results

5. findMStuff

IN – mStuff Location Query

OUT – mStuff Results

8.5mobeegal catalogs

The catalogs will be defined upfront by the mobeegal server. Each of the catalog will be a separate module that can be installed with user permission.

Ex:

Dating, Shopping, Rental, Sports, Marriage etc.,

8.5.1Catalog Categories

The catalog might have additional categories and sub categories.

Ex:

Sports, Cricket, Football, Hockey etc.,

8.5.2Catalog Attributes

Each of the leaf category will have one or more attributes. The attributes will be scalar (a point value) or vector (a collection) or a range.

Ex:

Dating

age, location, height, weight, hobbies etc.,

8.6People Catalog

8.6.1Dating Catalog

iStuff , uStuff Attributes

Age – Scalar – Number – Required

Sex – Scalar – String – Required

Height – Scalar – Units – Optional

Weight – Scalar – Units – Optional

Misc – Scalar – String - Optional
Location – Scalar – Complex – Required
 Location.Country – Scalar – String
 Location.City – Scalar – String
 Location.Area – Scalar – String

8.6.2 Matrimony Catalog

iStuff, uStuff attributes

Religion – Scalar – String - Required
Caste – Scalar – String - Required
Age – Scalar – Number - Required
Sex – Scalar – String - Required
Height – Scalar – Units - Optional
Weight – Scalar – Units - Optional
Color – Scalar – String - Optional
Misc – Scalar – String - Optional
Location – Scalar – Complex – Required
 Location.Country – Scalar – String
 Location.City – Scalar – String
 Location.Area – Scalar – String

8.7 Shopping

8.7.1 Jewelry

iStuff , uStuff Attributes

Item Type – Scalar – String - Required
Weight – Scalar – Units - Required
Misc – Scalar – String - Optional
Location – Scalar – Complex – Required
 Location.Country – Scalar – String
 Location.City – Scalar – String
 Location.Area – Scalar – String

8.7.2Cars

iStuff , uStuff Attributes

Car Type – Scalar – String - Required

Misc – Scalar – String - Optional

Location – Scalar – Complex – Required

Location.Country – Scalar – String

Location.City – Scalar – String

Location.Area – Scalar - String

8.8Fun

8.8.1Restaurants

iStuff , uStuff Attributes

Food Type – Scalar – String - Required

Misc – Scalar – String - Optional

Location – Scalar – Complex – Required

Location.Country – Scalar – String

Location.City – Scalar – String

Location.Area – Scalar - String

8.8.2Movies

iStuff , uStuff Attributes

Movie Type – Scalar – String - Required

Misc – Scalar – String - Optional

Location – Scalar – Complex – Required

Location.Country – Scalar – String

Location.City – Scalar – String

Location.Area – Scalar - String

8.9Home

8.9.1Rental

iStuff , uStuff Attributes

Rental Type – Scalar – String – Required

Misc – Scalar – String – Optional

Location – Scalar – Complex – Required

 Location.Country – Scalar – String

 Location.City – Scalar – String

 Location.Area – Scalar – String

8.10mobeegal match engine

The mobeegal match engine runs on the server. The match engine runs an innovative algorithm to match the mStuff Query with other available mStuff profiles. The engine will also use the mobeegal user profile for any preferences and generate results which are relevant to the user.

8.10.1Rule engine

A rule engine will be used to fire off rules to facilitate quick matching. Drools is a possible rule engine that can be investigated. Since the server will use PHP, a rule engine compatible with PHP needs to be selected.

8.10.2Attributes matching

Scalar, Vector and Range matching will be enabled in the match engine.

Scalar

The user wants the match to be exact.

Ex:

I want 10 tickets to movies.

Vector

The user want a collection of things

Ex:

I want a cricket bat, ball and stumps.

I want shoe, socks, and polish.

Range

This follows the premise that the User is not sure exactly what he wants, so he might be willing to look at a range of options under different parameters.

Ex:

I'm looking for a Job which pays me around 25k.

I want a second hand bike under 20k BUT well maintained. I can pay more than 20k, upto 25k if the bike is in excellent condition.

Fuzzy

The user is really ambiguous.

Ex:

I'm selling a bat which can be used for cricket and also baseball.

8.11mobeegal chat engine

The mobeegal chat engine runs on the server. The chat engine will be based on Jabber server which confirms to the XMPP protocol for chatting.

9.MOBEEGAL USER EXPERIENCE

A mobile user running Android, first needs to install mobeegal application. mobeegal.in server module will be available publicly on the Internet to support the mobeegal client module. mobeegal mobile client will be communicating and polling the mobeegal server to find matches for the required mobeegal module for the mobile user. A mobile user profile is send to the mobeegal server for the perfect match that the mobile user is looking for at this moment.

9.1mobeegal service

After installing the mobeegal application, the mobeegal service runs in the background looking for matches for the mobile user. The service also listens for incoming messages from the mobeegal server and uses an appropriate notification service to inform the mobile user. The service can started and stopped anytime by the user.

9.2mobeegal alerts

A mobeegal notifier will be run along with the mobeegal service. This mobeegal notifier will alert the mobile user of all matches found. The mobeegal alert will specify Proximity and the Closeness of the Match. The mobeegal alert will use LBS, Maps and all kinds of locational api's to interact with the mobile user.

9.2.1mobeegal alerts interaction

The mobile user can wish to interact with the match in his own way. The mobile user can select 1 or more matches and communicate with the match in an anonymous way. No private information will be provided to the match by mobeegal. The mobile user can disclose private or contact information at their own discretion. The mobile user can save all the matching profiles in his mobile device locally for future anonymous communication. The mobile user can delete this matching information at anytime.

9.3mobeegal location broadcast

The mobeegal service will be broadcasting the mobile user locations to the mobeegal server when the user is on the move. This location information need not be sent, if the user does not wish to do so. The polling interval can be set by the user. Example: The mobile user can wish to broadcast his location every 5 minutes. The mobile user can also totally turn off this feature.

9.3.1Obtaining Location Information

GPS – If available, GPS will be used. If GPS is not available other options will be tried.

Cell Id – Carrier provided Location information using Cell Id will be used if available.

3rdParty – 3rdParty API's will be queries if available. Ex: "My Location" from Google, Snowball, FluidNexus, etc.,

NOTE: If Automatic location broadcasting is not available, then the User will be asked to enter this information manually.

9.3.2Location locking

The user can lock a location for a particular search. Once location is locked, location broadcasting will be disabled. The user might still want to find matches for the locked location at the specified time interval automatically. In this case the search request will be made with a constant location.

9.4iStuff

The mobile user first needs to fill iStuff (User stuff description). Public information will be collected and kept anonymous. iStuff might be communicated to the match, if the mobile user wishes so. Privacy and Anonymity is kept foremost, and only public information will be used. The iStuff information can be fine tuned at any time. The mobeegal application will also fine tune the profile preferences based on the number of acceptance or rejection rates with matching profiles. The mobile user profile will be stored in his mobile by the mobeegal application. This information will be send to the mobeegal server.

9.5uStuff

The mobeegal user must fill out a uStuff (User wants this stuff). This is the kind of profile that the mobile user requests for matching. Again only public information is asked for such as location, age range etc., and not for any secret or private information. More than one uStuff can be stored, and the mobeegal application will store this in the mobile user's device locally. This information will be send to the mobeegal server.

9.6mStuff

The mobeegal service finds the mStuff (perfect match) for the mStuff search and alerts with a notification. The match results will be delivered by the mobeegal match engine over the agreed mStuff protocol. The match results can be viewed on the map with zoom in/out functionality to get closer to the match.

9.7mStuff Query Criteria

An mStuff search criteria will contain exactly one pair of iStuff and uStuff each. The combination of iStuff + uStuff results in a mStuff Search Criteria.

mStuff Query Criteria =
iStuff + uStuff

9.8mStuff Query Condition

An mStuff query condition will contain exactly one catalog, an optional category and many mStuff Search Criteria.

mStuff Query Condition =
Catalog + [Category] + (1..n) mStuff Query Criteria

All mStuff Query Condition for the current Catalog will be listed in the Catalog detail screen.

9.9mStuff Query

An mStuff search will contain one or more mStuff Query Condition

mStuff Query =

(1..n) mStuff Search Condition

9.10mobeegal User profile

The mobeegal application will maintain a mobeegal profile for the mobile user based on the mobile user's profile and preferences. NOTE: This mobeegal profile is separate from the iStuff/uStuff submitted by the user. This is not a user submitted, but rather mobeegal generated. This mobile user mobeegal profile will be summary of all of the mobile user likes and dislikes which are constantly fine tuned and evolved by the mobeegal application.

9.11mobeegal chat

The mobeegal chat will let the mobeegal users chat with their matches in real time. The mobeegal client will communicate with the mobeegal server over XMPP protocol.

10.GETTING STARTED

10.1Admin

10.1.1Catalogs creation

The admin will create catalogs for the mobeegal system. The catalogs will be stored in the database. The catalogs can have additional sub catalogs as child nodes. The catalog with attributes will form the leaf of the tree, and will not have additional child catalogs anymore.

10.1.2Attributes to Catalogs addition

The admin will add attributes to the search catalogs. The number and type of attributes will be determined by the particular search catalog.

10.2User

10.2.1iStuff

iStuff is a description of what you are, given the context of the catalog and the selected category.

10.2.2uStuff

uStuff is a description of what you want, given the context of the catalog and the selected category.

10.2.3mStuff - Find stuff closer

When the user hits search, mStuff Query is send to the mobeegal server for matching. The mobeegal server returns the matched mStuff results to the user.

10.2.4Relevate results mStuff (favorite, hate)

mStuff is the matched results for a particular search criteria. The search criteria is applicable for a search category in a catalog.

The mStuff can be rated and voted up and down. Voting up means the user likes it, and voting down means he dislikes it. This rating will be send to the server, and would be of impact in future search results.

10.2.5Chat

Pick a match from the mStuff and start chatting. The chatting is anonymous and private. Only the public information as allowed by the user will be visible to the match. The match can be blocked or banned at any time.

11.MOBEEGAL MATCH ENGINE

The mobeegal match engine runs on the mobeegal server. The match engine given an mStuff Query will find and output mStuff match results. The iStuff, uStuff data and the catalog data along with the filled in categories/attributes will be the input data. The mStuff match attributes along with relevancy results will be the output data.

12.MOBEEGAL CHAT ENGINE

The mobeegal chat engine runs on the mobeegal server. The chat engine enables the mobeegal users to chat with their matches. The chatting is completely anonymous and only public information from user iStuff is shown to his match.

13.FAQ

1. How many service runs in mobeegal client?
Exactly one.
2. How many catalogs are available for search?

There'll be many catalogs and its not limited by any number. People, Shopping, Fun are some of the existing catalogs.

3. mStuff - can you explain?
mStuff is the match result. The mobeegal server given the mStuff Query, will find relevant mStuff results and return to the client. mStuff is a single match. So the server will return most probably a collection of mStuff which we call mStuff results.
 4. uStuff - what is this?
uStuff is what the user wants. This is the criteria that the user is searching for in his match.
 5. iStuff - again, care to explain?
iStuff is what the user is. This is the criteria that the user uses to describes his stuff.
 6. Will there be User registration?
No. mobeegal will be completely stateless and anonymous as far as the User is concerned.
 7. Is mobeegal Secure?
Yes. All requests are encrypted for maximum User security.
 8. Is mobeegal Private?
Yes. Private information will never ever be send to the mobeegal server. Only the public information is shared with his matches.
 9. How can I search?
The User fills iStuff and uStuff for whatever catalog that he wants. Now the search can begin. The mStuff Query is automatically formed from the iStuff and the uStuff information that the user provides across catalogs.
 10. Do I need to activate the Service to search?
Yes. The service must be activated for search. The service will send the mStuff Query to the server only after its activated.
 11. Does the Service send the mStuff Query with every request to the Server?
No. The Service will be smart enough to send it only once. If the mStuff Query has not changed, any future service request might just be a ping to the server to retrieve only the mStuff results.
 12. Do I need to fill both iStuff and uStuff for searching?
Yes. Filling both iStuff and uStuff will be called mStuff Search criteria. Just like you need a search query for searching, mobeegal requires mStuff Query which will include your mStuff criteria to bring you the relevant mStuff results.
-

13. Can I have multiple mStuff query criteria filled up?

Yes. Any number of mStuff query criteria can be filled for as many catalogs as the User likes. All mStuff criteria will be bundled into exactly one mStuff Query. See mStuff Query, mStuff Search Condition and mStuff Search criteria above.

14. Can I search more than 1 catalog?

Yes, More than one catalog will be available for searching. All your mStuff criteria is send as a single mStuff Query which will span multiple catalogs. As a user you just pick any number of catalog, enter your istuff and ustuff and you are done.

15. Can I filter the mStuff results?

Yes. You can filter the mStuff by catalogs. There will be a search button and a filter drop down available in your mStuff views.

16. Can I search within the mStuff results?

Yes. A search box will be provided to search within the mStuff results.

17. Is there more than one way of looking at my mStuff results?

The user can switch between text mode and map mode for viewing the mStuff results. The search and filter widget will be available for both modes of viewing.

18. How do I activate the Service?

The User can hit the menu, go to settings. Activate/De-Activate Service is available here.

19. Where do I store my Preferences?

The User can hit the menu, go to settings. Preferences button is available here. This would take the User to a separate screen to store the User preferences.

20. Help?

Available in the Settings menu. Click Settings menu, and Help is one of the buttons available.

21. mobeegal rocks!

Glad you liked it.

14.MOBEEGAL v1.0.0-ALPHA FUNCTIONALITIES

- User enters a mobeegal search catalog, and can select any of the mobeegal search category available. Dating will be the available catalog for alpha with no sub-categories. Shopping, Jobs, Rental, Restaurants, Movies are other catalogs available but are inactive. These additional catalogs will be made available for beta.
- User can start/stop the mobeegal service. User can set the location broadcast to "auto" or "manual" mode. Anonymous Registration of the User will be done when the Service is first activated. Once Registered the User registration will not happen again.
- User can create and submit iStuff. User can create and submit uStuff.
- User can submit a mStuff Query.
- User gets notifications about mStuff results (matches) found in the background task bar without interfering his normal activities.
- User can view mStuff results (matches) in his map. Can zoom in/ zoom out. A text view will also be provided where the results show up in a list.
- User can add one or more of the match as favorites. This is voting up.
- User can add one or more match as ignore. This is voting down.
- User can chat with one or more of his match at any time. Of course establishing the chat will also depend on the willingness and availability of the match.

15.MOBEEGAL v1.0.0-BETA FUNCTIONALITIES

- Localization – All resources must be localized. Tamil, Hindi to be supported.
- Privacy – Users complete privacy ensured. The user can never be identified.
- Security – All communication completely encrypted.
- Media files to be supported like profile picture, video, audio etc.,
- Keyboard requirement will be optional. Touch mode to be supported fully.
- Text mode will be fully functional and can be used for dumb phones.
- Many Catalogs will be supported – Shopping, Entertainment, Sports, News, NightLife, Culture, Movies, Dining, Restaurants etc.,
- Many devices to be supported on top of smart phones. Must work even in dumbed down devices like older models and be able to support SMS communication.
- mobeegal match engine will support rule engine.
- mobeegal match engine to support fuzzy logic.
- mobeegal server (match + chat) tested for performance.
- Auto expiration of mStuff Query.

- Users offered customized themes, for different catalogs.
- mobeegal UI to be completely enhanced, with overlays, transparency, touch mode with sleek professional design.

16.MOBEEGAL v1.0.0_FINAL FUNCTIONALITIES

- Authority of the matches to be provided based on user rating system.
- Comparisons between matches will be available.
- Recommendations of other matches in other catalogs will be provided.
- Premium paid service will be available.

17.SUMMARY

mobeegal is an innovative mobile search engine application built for mobile users all around the world. Finding your match be it for dating, shopping, jobs, marriage etc., in your location is only one click away from your mobile. Happy mobeegal!