MicroMaps Requirements

|  |  |
| --- | --- |
| REQ No.  1 | Project  MicroMaps |

Abstract

| **Rev** | **Date** | **Author** | **Status** |
| --- | --- | --- | --- |
| 1 | 26/01/2015 |  | start |
| 2 |  |  |  |
| 3 |  |  |  |
|  |  |  |  |
|  |  |  |  |

Distribution List

|  |  |
| --- | --- |
| **Reviewers** | **FYI** |
| Ji Lucas  Patrick  Heather |  |

Table of Contents

1. Introduction 4

2. Related Documents/Links/People 4

3. Glossary 4

4. Application Design 4

4.1. High Level Application Design 4

4.2. Database 6

4.3. Application Design 6

4.3.1. GeoClickers and KML Flow 6

4.3.2. Application 8

4.4. Client Application 8

4.4.1. Front UI Mockup 8

4.4.2. The required search function 8

4.4.3. The required output 9

4.4.4. Expected Behavior 9

4.5. In Scope 10

4.6. Out of Scope 10

4.7. Assumptions 10

4.8. Constraints 10

4.9. Dependencies 10

4.10. Risks 10

5. Requirements 11

5.1. Requirement Spec. 11

5.2. Non-functional 11

5.2.1. Security 11

5.2.2. Testing 11

Issues/Questions 11

6. Revision History 12

MicroMaps

# Introduction

# Related Documents/Links/People

References in the text throughout this document appear in square brackets (e.g., [1], [JS]).

| **Reference** | **Document/Link/Person/Application** |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# Glossary

| **Term** | **Definition** |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

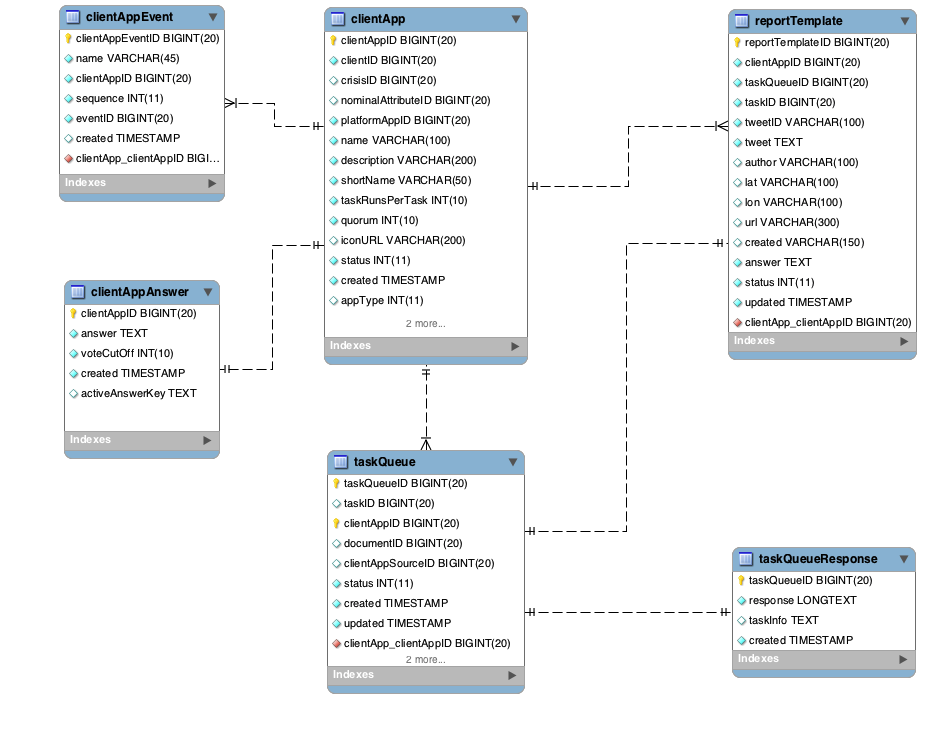
# Application Design

## High Level Application Design



There is an existing process that filters data into geo-map ready data for the map. In this document, we assume that data has been processed and has been geo-tagged properly based on each clicker’s business logic. Therefore, we will discuss about data generation and client map template development.

## Database



|  |  |  |
| --- | --- | --- |
| TABLE NAME | DESCRIPTION | NOTE |
| clientApp | holds each clicker configuration |  |
| taskQueue | holds all available & completed tasks |  |
| taskQueueResponse | holds completed task responses |  |
| reportTemplate | holds task info with answer that is used to generate geo clicker’s data feed |  |
| clientAppAnswer | Holds answers and answer business logic | Need to re-structure json |
|  |  |  |

\* To generate geojson or kml values for MicroMaps, the system should get the data from taskQueueResponse & reportTemplate.

## Application Design

### GeoClickers and KML Flow





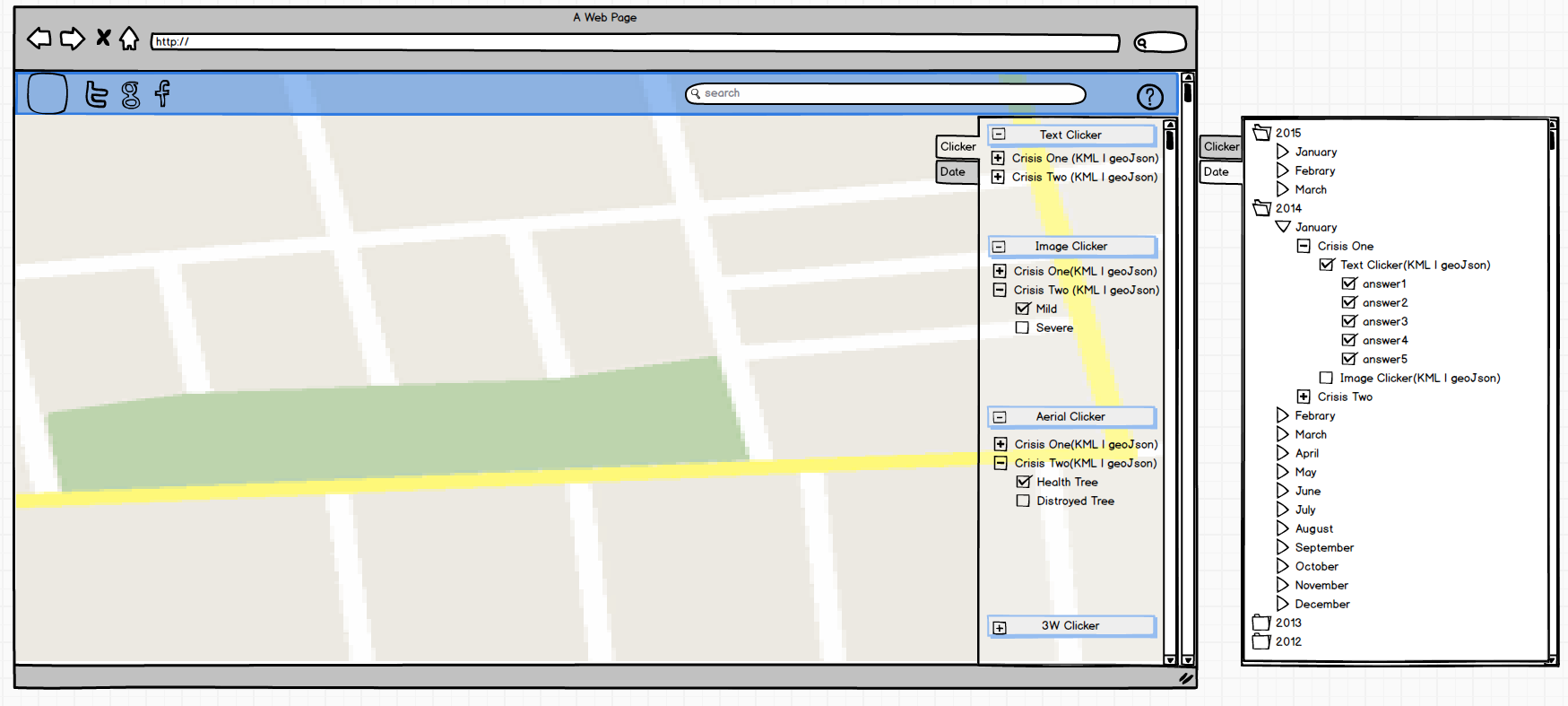
### Application



* Application should be event-driven process.
* Maps template should have channels to listen event
* Maps should contain all history and the current data
* Application should push notification to the front-end
* Application should generate geojson/kml based on each application’s need. In other word, it should be minimal info of the below.
* mm-gis-api will have restful api for data download
* mm-gis-api will have push notification workflow

## Client Application

### Front UI Mockup



### The required search function

* Search by Clicker Type
  + Text Clicker
  + Image Clicker
  + Aerial Clicker
  + Satellite Clicker
  + 3W Clicker
* Search by Date

### The required output

* User should be able to download the labled data based on clicker + crisis only.
* User should be able to download dataset with geojson or kml format.
* When user clickers on a pin, user should be able to output answer
* Text clicker download dataset should contain :
  + Text
  + Answer
  + Geo with lat & lon & address
* Image clicker download dataset should contain:
  + Text
  + Image-URL
  + Answer
  + Geo with lat & lon & address
* Aerial/satellite clicker download dataset should contain:
  + Image-URL
  + Answer
  + Geo with geojson
* 3W clicker download dataset should contain:
  + Link
  + Who
  + Geo with lat & lon & address
  + What

### Expected Behavior

* The system should listen any update from server
* The data update should render on the map properly
* Expected data format example:

[

{

"text": "test text tweet clicker ",

"image": "",

"answer": "Help needed",

"geo": "[125.00571277923882,11.241663227712435 ]",

"taskID": 517397,

"appType": 1

},

{

"text": "image clicker testing",

"image": "/static/namibia/day5\_rgb\_transect\_rosinki\_ixus/slice/IMG\_8218\_2.jpg",

"answer": "severe",

"geo": "[125.00571277923882,11.241663227712435 ]",

"taskID": 517397,

"appType": 2

},

{

"text": "",

"image": "/static/namibia/day5\_rgb\_transect\_rosinki\_ixus/slice/IMG\_8218\_2.jpg",

"answer": "",

"geo": "[[{\"bounds\":{\"\_southWest\":{\"lng\":125.00571277923882,\"lat\":11.241663227712435},\"\_northEast\":{\"lng\":125.00572686083613,\"lat\":11.241673997333708}},\"maxBounds\":{\"\_southWest\":{\"lng\":125.00570573844016,\"lat\":11.241657842901798},\"\_northEast\":{\"lng\":125.00573390163478,\"lat\":11.241679382144344}},\"properties\":{},\"type\":\"Feature\",\"geometry\":{\"type\":\"Polygon\",\"coordinates\":[[[125.00572518445551,11.241673997333708],[125.00572292134164,11.24167251753843],[125.0057204067707,11.241669475737035],[125.00571713782847,11.241669886791287],[125.00571319833398,11.241669229104485],[125.00571277923882,11.241667338254944],[125.00571294687688,11.241663638766672],[125.005715880543,11.241663227712435],[125.00571956858039,11.241664049820937],[125.0057225022465,11.241665529616252],[125.00572543591261,11.241669311315336],[125.00572686083613,11.241673093014368],[125.00572518445551,11.241673997333708]]]}}]]",

"taskID": 517397,

"appType": 5

},

{

"text": "http://edition.cnn.com/2015/03/07/us/st-louis-hospital-copter-crash/index.html",

"image": "",

"answer": [

{

"who": [

"red cross",

"UN"

],

"what": [

"Delivery of tarpaulins - Camp Coordination and Camp Management and/or Emergency Shelter"

]

}

],

"geo": "[125.00571277923882,11.241663227712435 ]",

"taskID": 517397,

"appType": 6

}

]

* appType description :
  + 1 : text clicker
  + 2 : image clicker
  + 5 : aerial/satellite clicker
  + 6 : 3W clicker

## In Scope

* Image, Aerial imagery, Satellite Imagery display only in info. window with its output
* Push notification should be added in the server coding
* Push notification listener should be added in the client app.

## Out of Scope

* Only kml and geojson will be consumed for the map.
* Video clicker is not consider in the scope
* Image over layer won’t be enabled due to loading issue

## Assumptions

* MicroMaps will consume geojson & kml only to generate Maps.
* The system should be use push notification to render updated data.
* mm-gis-api will handle map related request/process

## Constraints

## Dependencies

* Network connection to get osm, google apis
* Imagery availability

## Risks

* Satellite imagery generation may causes data-loading issue.

# Requirements

The following subsections define software requirements. Each requirement is labeled as follows:

* **ID.** The unique identifier for the requirement.
* **Requirement.** A clear and concise description of the requirement.
* **Source.** A cross-reference to the source of the requirement.
* **Cat**egory**.** A classification for the requirement using the selections from Table 1.

Table - Requirement Categories

| **Category** | **Description** |
| --- | --- |
| **M**andatory | Required minimum functionality. |
| **O**ptional | Nice to have and will be implemented if time permits. |
| **F**uture | Will not be implemented now but should be considered for a future enhancement. |
| **D**ropped | Requirement determined to be completely out of scope **after** baseline. |
| **E**xisting | Functionality already present in the software (for documentation purposes). |
| **Op**erational | Will not be addressed in the software (documented for training purposes). This requirement should have a related item in the issue list in order to identify an owner. |

## Requirement Spec.

| **ID** | **Requirement** | **Source** | **Cat** |
| --- | --- | --- | --- |
| 1 | The system should generate geojson to render data into map | M |  |
|  | The system should generate kml and geojson based on crisis + clicker type only | M |  |
|  | User should be able to download kml or geojson by clicker a crisis + a clicker | M |  |
|  | The system should notify any data update to client | M |  |
|  | Geojson should contain appType,text, image, geo, answer, taskID | M |  |
|  | Text clicker should have max 5 answers. Each answer will have its own color for markers:  Red  Orange  Green  Blue  Purple | M |  |
|  | Image clicker should have max 2 answers. Each answer will have its own color for markers:  Red : Severe  Orange : Milde | M |  |
|  | Aerial Clicker should have its own icon for markers. No color code is required | M |  |
|  | Aerial clicker’s info window should render its geojson on the top of its imagery with drawing info | M |  |
|  | 3W Clicker should have its own icon for markers. No color code is required. | M |  |
|  | 3W clicker’s info window should render text, link, who , what | M |  |
|  | Map should have listener for the update. Push notification should be channeled properly | M |  |
|  | Server should send any update via a channel | M |  |
|  | Map should update when its gets a new data dynamically. | M |  |
|  | Map’s right side search option panel should be updated when its gets a new data dynamically if it is required | M |  |
|  | Search option panel should be displayed in default | M |  |
|  | User should be able to hide Search Option Panel to see a map only | M |  |
|  | User should be able to select 1 or more or all answered data by selection answers in text clicker/image clicker. | M |  |
|  | Map should render proper data based on user’s search | M |  |
|  | Image clicker : user can select mild or severe or both | M |  |
|  | Any server request call should direct to mm-gis-api | M |  |
|  | Any push notification process should resident in mm-gis-api | M |  |
|  |  |  |  |

## Non-functional

### Security

| **ID** | **Requirement** | **Source** | **Cat** |
| --- | --- | --- | --- |
| .1 |  |  |  |

### Testing

| **ID** | **Requirement** | **Source** | **Cat** |
| --- | --- | --- | --- |
| .1 |  |  |  |

# Issues/Questions

| **Issue #** | **Issue/Resolution Description** | **Date/Status** |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Revision History

| **Date** | **Change Description** |
| --- | --- |
|  | Initial draft. |
|  |  |
|  |  |
|  |  |