

Mobile Logs Server

October 8, 2015

Yeffry Zakizon

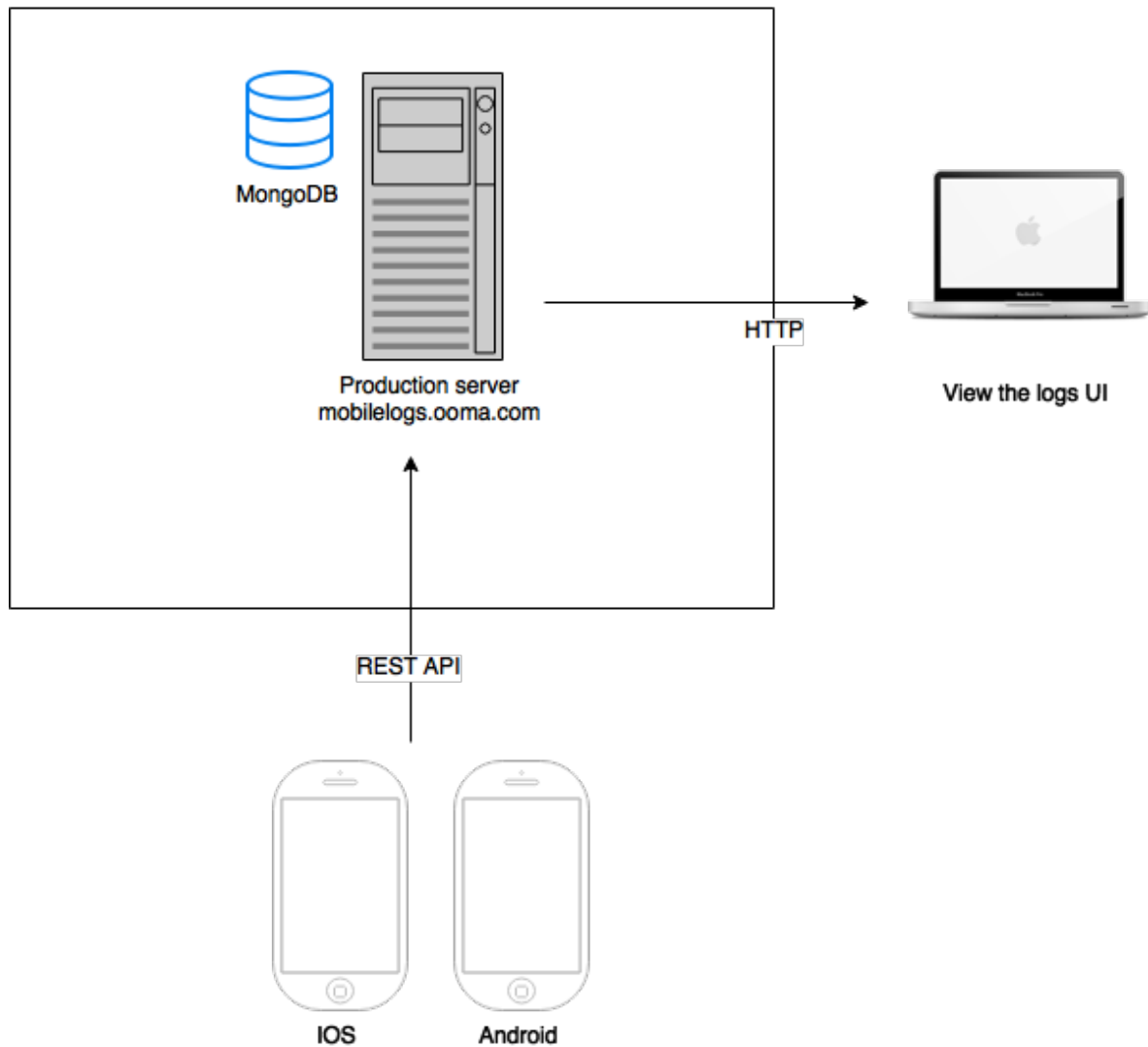
Mobile logs server collects and save Ooma Mobile debug logs and events.

We have one dev server and one production server located at Ooma's HQ.

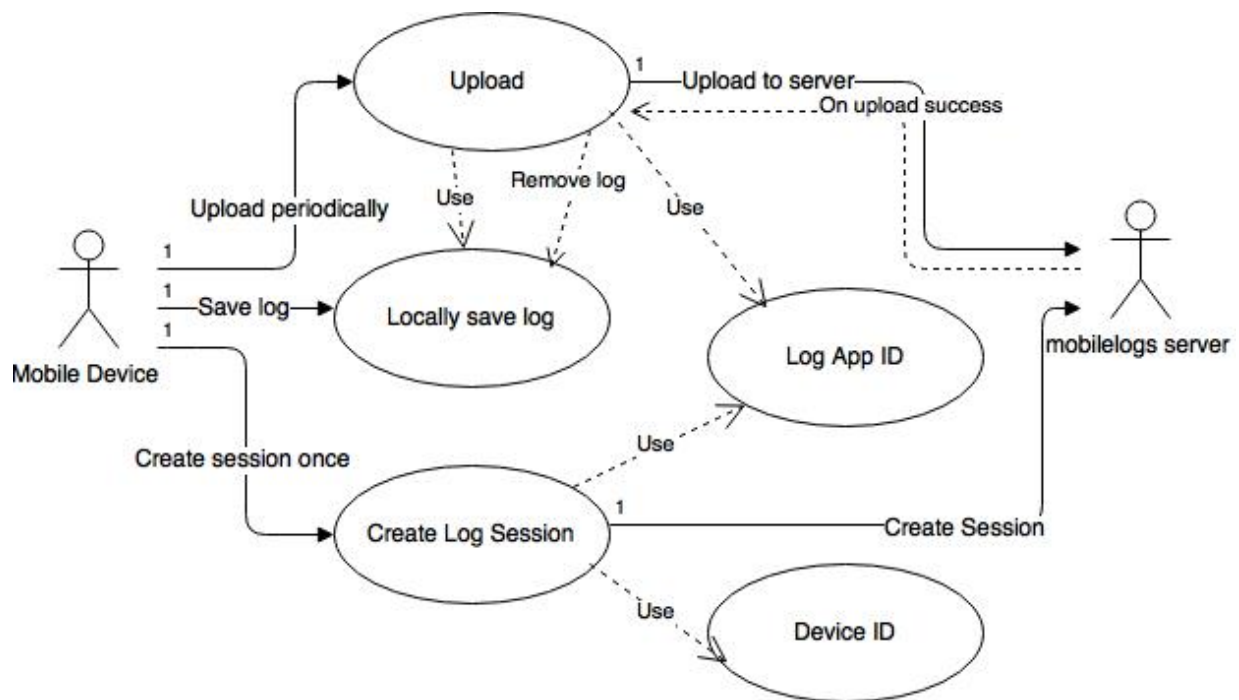
- Dev server: <http-log.corp.ooma.com>
- Production server: <mobilelogs.ooma.com>

To access the website UI you need to signup and use invitation code **!ooma1234@**. This invitation code should be keep remain secret as other people might signup to this service.

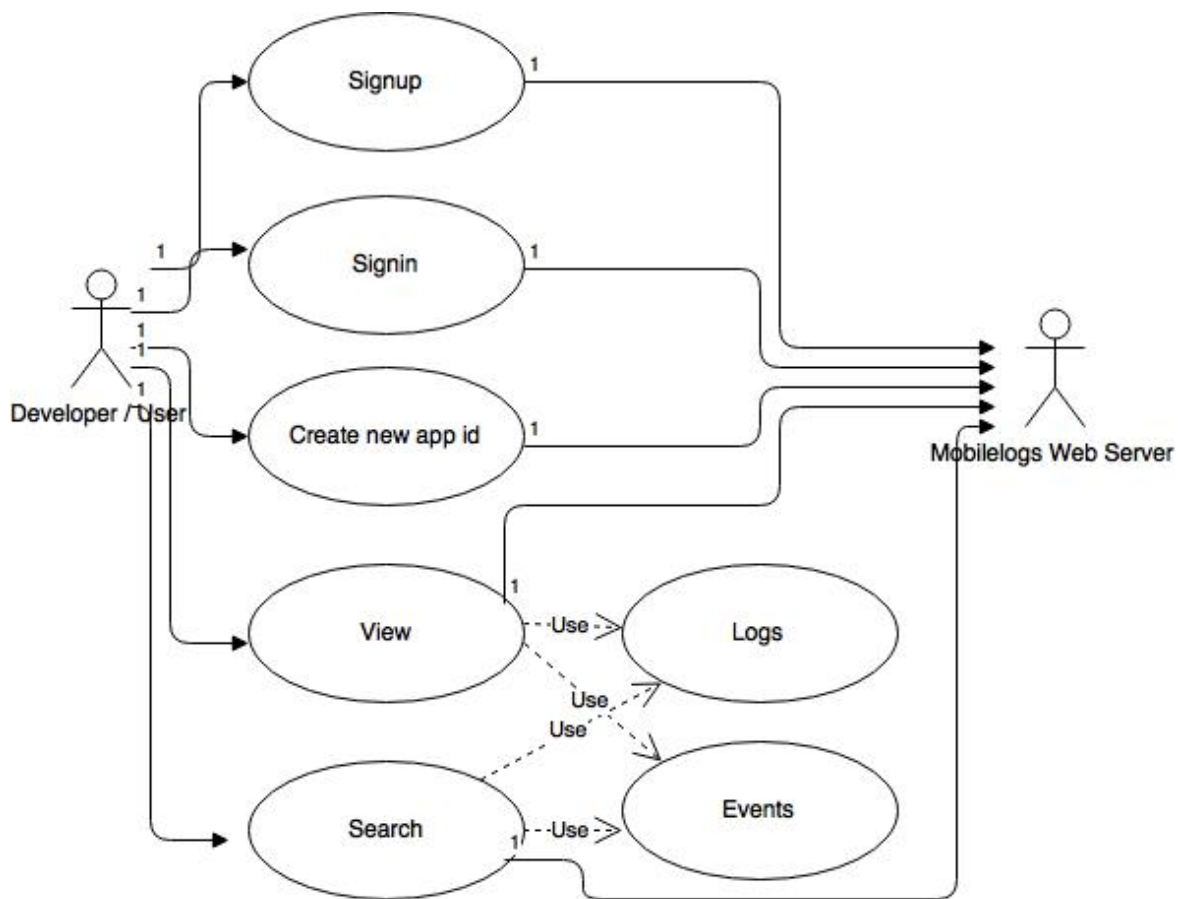
Ooma Premises



Mobile Device Use Case Diagram



Users Device Use Case Diagram



Installation

The server side code can be downloaded from [perforce //internal/software/mobile/ServerSide/...](https://github.com/perforce/internal/software/mobile/ServerSide/)

Software Needed

- Nginx ~> v1.9.2
- nodejs & npm ~> v0.10.33
- mongodb ~>3.0

Nginx

Nginx act as reverse proxy server and SSL termination. SSL certificate must be used. An example of nginx.conf is available in doc directory.

MongoDB

A database user must be created so the nodejs app can connect to MongoDB. Currently there are 2 databases that is required: - applog db. This is the main database - express db. Use to store http sessions.

```
use applog;
db.createUser( { user: "applog", pwd: "ooma123", roles: [ {role: "readWrite", db:
"applog"}] });

use express;
db.createUser( { user: "express", pwd: "ooma123", roles: [ {role: "userAdmin", db:
"express"}] });
```

Nodejs App

Node modules binary is built for linux Scientific Linux 6.4 with kernel 2.6.34. This is OS that Ooma's supported. If you use different OS, you need to be delete all files in directory node_modules and rebuild.

```
rm -rf node_modules/*
npm install
```

To start the nodejs app run

```
Nodejs index.js
```

or

```
npm start
```

API

Create Session (/api/v1/session)

Description:

Create temporary session, output key is used to validate request input body.

input:

- **apikey (String)** : apikey
- **devid (String)** : Device id
- **created_date (Date)**
- **os (String)** : For example: IOS, ANDROID, WP, BLACKBERRY
- **devtype (String)** : For Example [iPhone 5s, iPhone 5c, iPhone 6, Nexus 6, galaxy 6]
- **devManufacturer (String)** : For example //[Apple, Samsung, HTC]
- **osVersion (String)**: Device operating system version for example: 8.3, 5.2

output:

- sessionid (String) : Used when upload log or event

method:

- POST

Save log (/api/v1/log)**Description:**

Upload log to mobilelog server

input:

- **apikey (String)** : apikey
- **sessionid (String)** : Session id from create session
- **devid (String)** : Device id
- **data (log objects)** : Array of log objects

log objects:

```
[{
  level: int,
  log: String,
  processid: String,
  tag: String,
  phone_number: String,
  phone_ext: String,
  local_ip: String, //Local ip address of device
  client_date: date //Local date of device
}];
```

methods:

- POST

Save event (/api/v1/event)**Description:**

Upload events to mobilelog server

input:

- **apikey (String)** : apikey
- **sessionid (String)** : Session id from create session
- **devid (String)** : Device id
- data (event objects) : Array of of log objects

event objects:

```
[{
  event_name: String,
  event_data: Event data object,
  processid: String,
  tag: String,
  phone_number: String,
  phone_ext: String,
  local_ip: String, //Local ip address of device
  client_date: date //Local date of device
}];
```

event data object:

```
{
  action: String,
  label: String,
  value: String
}
```

methods:

- POST