

Quikr QDP Client 1.0

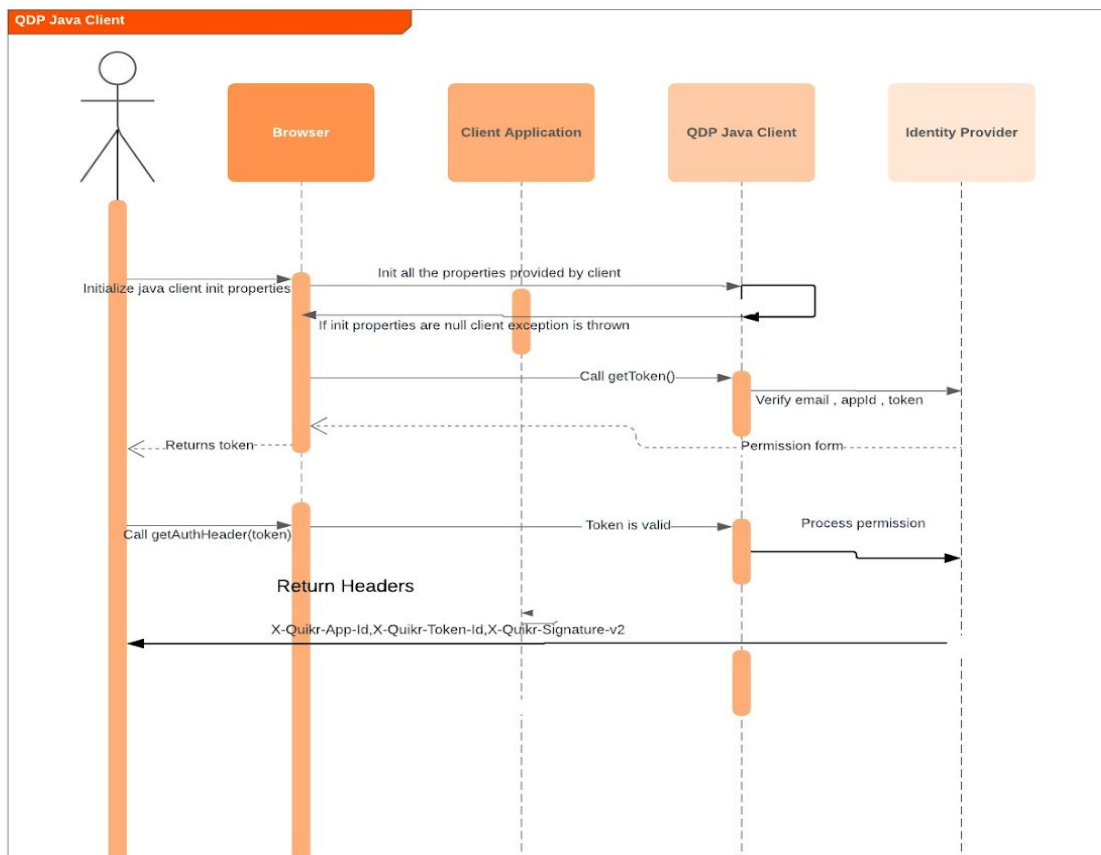
Version	Date	Author	Description
1.0	31-January-2019	Komal Kumra	Java and Php client library for QDP

Sequence Diagrams

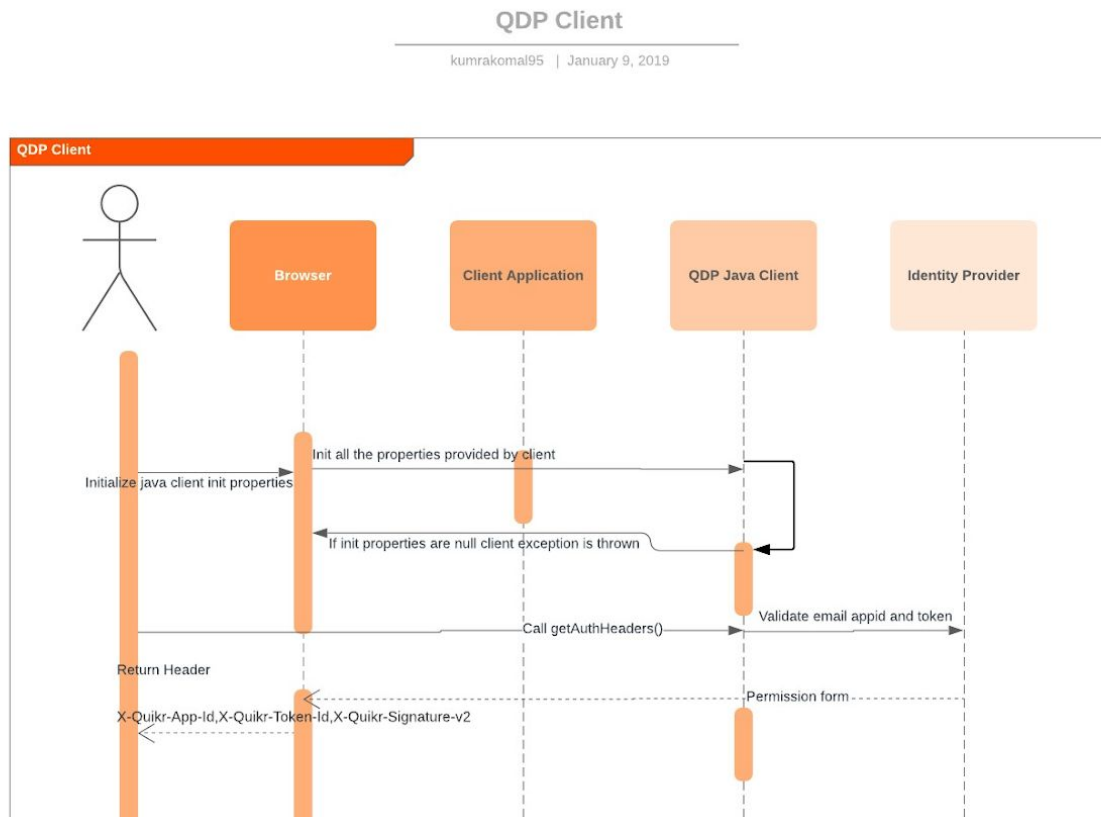
Scenario 1: Generating and passing Token to generate Headers

QDP Client

kumrakomal95 | January 9, 2019



Scenario 2: Accessing stored Token to generate Headers



Steps to integrate:

- From the dashboard create a shared secret for your application. Note the app secret and the app id.
- Download and Add Jar or php folder to your project.
- Initialize host, appId, email Id and secret Key in init function.

Java Developers : Initialize all the properties by using Properties class .

```

Properties p = new Properties();
p.setProperty("host", "");
p.setProperty("appId", "");
p.setProperty("email", "");
p.setProperty("secretKey", "");
QDPServiceImpl.init(p);

```

Php Developers : Initialize all properties in associative array.

```

$properties_array=array("host"=>"" ,"appId"=>"" ,"secretKey"=>"" ,
    "email"=> "");
$quikrServiceObj->init($properties_array);

```

- `getInstance()` function will check if any of the initialized value is null .
Java Developers : `final QDPService qdpService = QDPServiceImpl.getInstance();`
Php Developers : `$qdpServiceImplobj->getInstance()`
- Generating an access token : `getToken()`
Scenario 1:
Generate Auth Headers By Passing Token: `getAuthHeaders(token)` will generate all request headers that will be helpful for accessing the Quikr APIs .
Scenario 2:
Generate Auth Headers By using Stored Token: `getAuthHeaders()` will generate all request headers that will be helpful for accessing the Quikr APIs.

Example Of Using Jar:

Java Developers:

Scenario 1: With Token

```
public static void main(String[] args) throws ConnectTimeoutException,
TokenException, QDPServiceTimedOutException, QDPInvalidResponseException {
    Properties p = new Properties();
    p.setProperty("host", "");
    p.setProperty("appId", "");
    p.setProperty("email", "");
    p.setProperty("secretKey", "");
    QDPServiceImpl.init(p);
    final QDPService qdpService = QDPServiceImpl.getInstance();
    Token result = qdpService.getToken();
    Map map=qdpService.getAuthHeaders(result);
    System.out.println(map);
}
```

Scenario 2:Without Token

```
public static void main(String[] args) throws ConnectTimeoutException,
TokenException, QDPServiceTimedOutException, QDPInvalidResponseException {
    Properties p = new Properties();
    p.setProperty("host", "");
    p.setProperty("appId", "");
    p.setProperty("email", "");
    p.setProperty("secretKey", "");
    QDPServiceImpl.init(p);
    final QDPService qdpService = QDPServiceImpl.getInstance();
    System.out.println(qdpService.getAuthHeaders());
}
```

```
}
```

Php Developers:

Scenario 1: With Token

```
$qdp=new QDPServiceImpl();  
$token=new Token();  
$properties_array = array("host"=>"" ,"apld"=>"" ,"secretKey"=>"" ,  
    "email"=> ""  
);  
$qdp->init($properties_array);
```

```
$qdp->getInstance();  
$token=$qdp->getToken();  
$array = $qdp->getAuthHeaders($token);  
echo "{";  
foreach($array as $x => $x_value) {  
    echo $x . " = " . $x_value;  
    if ($x == 'X-Quikr-Signature-v2'){  
        break;  
    }  
    echo " , ";  
}  
echo "}\n";
```

Scenario 2: Without Token

```
$qdp=new QDPServiceImpl();  
$token=new Token();  
$properties_array = array("host"=>"" ,"apld"=>"" ,"secretKey"=>"" ,  
    "email"=> ""  
);  
$qdp->init($properties_array);
```

```
$qdp->getInstance();  
$array2 =$qdp->getAuthHeaders();  
echo "{";  
foreach($array2 as $x => $x_value) {  
    echo $x . " = " . $x_value;  
    if ($x == 'X-Quikr-Signature-v2'){  
        break;  
    }  
    echo " , ";  
}
```

```
}echo "};
```