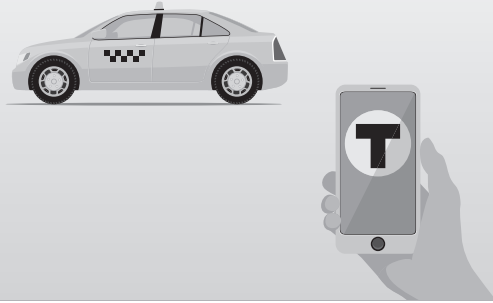


NFC 택시 안심 서비스를 위한 태그 데이터 구조



임예진 모바일센 주식회사

1. 머리말

최근에 발생한 ‘강남역 화장실 여성 살인사건’, ‘수락산 살인사건’, ‘부산 길거리 무차별 폭행 사건’ 등을 떠올려 볼 때, 심야 시간대 또는 우범지대 등에서 여성이나 사회적 약자들에 대한 안전장치는 점점 더 절실해지는 상황이다.

이런 사회적인 안전장치의 일환으로 개발된 NFC 택시 안심 서비스는 ‘비타민 프로젝트’의 7대 중점 분야 중 대국민 안전지원 서비스를 제공하는 재난 안전 SOC(비타민S)의 일환으로 추진되었으며, 택시 좌석에 부착된 NFC 태그에 스마트폰을 대면 앱이 자동으로 구동되어 택시 운행정보를 지인에게 문자로 전송하는 서비스이다.

2013년도에 서울특별시, 인천광역시, 경기도, 울산광역시를 시작으로 폭발적으로 서비스가 확산되었으며, 2014년도에는 전국 택시 기준 60% 이상을 상회하는 시장점유율을 나타내기도 하였다. 이러한 NFC 택시 안심 서비스는 2015년에 들어서면서 카카오가 택시 콜 시장에 진입하면서 카카오 택시 안심서비스 이용의 활성화에 기여하였으며, 더 나아가서 블루투스 기반의 버스 안심 서비스 및 NFC 길

거리 안심 서비스와 같은 여성과 사회적 약자 계층에 대한 안전장치 확대라는 관점에서의 그 의미는 매우 크다고 말할 수 있다.

본 표준은 택시 안심 서비스의 NFC 태그 내 택시 차량 매칭 정보 저장 방식을 통일하여 DB 서버 구축, 유지보수 비용을 절감하고 태그 데이터 구조를 공개하여 NFC 태그의 재활용성을 증대하고자 한다. 택시 차량 정보의 공공 DB화와 일반인 누구나 서비스 개발이 가능하여 다양한 부가서비스 창출이 가능하다.

2. 주요 내용

2.1 표준이 목적 및 내용

본 표준의 목적은 NFC 태그에 저장된 택시 정보(차량번호, 연락처, 운수회사 정보)와 탑승객의 스마트폰 GPS 정보를 매칭하여 탑승객의 스마트폰을 통해 탑승객의 보호자에게 문자 메시지로 전송해줌으로써 택시 이용자들이 안심하고 택시를 이용할 수 있도록 하며, NFC 택시 안심 서비스의

태그 인코딩 규격 및 앱 개발 가이드라인을 공개하여 택시 안심 서비스의 신규 도입과 확산이 용이하다.

또한, 본 서비스의 특징은 오픈 플랫폼 형태로 배포되었기 때문에 NFC 택시 안심 서비스를 제공하고자 하는 사업체는 손쉽게 기 배포된 택시 안심 서비스 태그를 활용하여 서비스를 제공할 수 있다.

본 표준에서는 NFC 택시 안심 서비스 제공을 위한 태그 정보에 대한 정의와 태그와 연동하여 정보를 보여주는 애플리케이션의 개발을 위한 규격을 정의한다. 따라서, 본 표준에서는 제공하는 규격에 맞게 NFC 택시 안심 서비스를 개발한다면, 서비스 제공자는 별도의 태그 인프라 구축 없이도 기 배포된 태그 인프라를 활용하여 서비스 제공이 가능하도록 설계되었다.

본 표준 제정을 통해서 기존에 지자체별로 또는 업체별로 제공 중이었던 택시 안심 서비스에 대한 표준을 제정함으로써 택시 안심 서비스 이용자들에게는 일관성 있는 경험을 제공하며, 대도시 일부 지역에서만 시행되어 오던 택시 안심 서비스의 확대 적용을 예상할 수 있다. 또한 본 표준 제정을 통해서 동일한 규격의 택시 안심 서비스의 확대 발전이 진행되며 그 태그 인프라를 활용한 부가적인 서비스 제공이 가능하다.

2.2 NFC 택시안심 서비스 이용 Process

본 서비스는 이용자의 안드로이드 스마트폰을 택시에 설치된 NFC 태그에 맞대는 것만으로 간편하게 지정된 보호자에게 안심 알림 문자를 발송할 수 있다. 본 표준에서 설명하고 있는 태그 정보를 이용한 서비스에 대한 프로세스와 그 이용 예시를 살펴보면 다음과 같다.

1. 탑승 단계

설치된 태그에 휴대폰을 갖다 대면 '안심알림발송' 화면이 나타난다.
(※시스템 환경설정 → 추가설정 → NFC 기능을 '켜짐'으로 설정 필요)

2. 발송 단계

화면에 나타난 승차한 택시의 정보 및 현재 위치 등을 확인한 후 '승차알림발송' 버튼을 눌러 발송화면으로 이동한다. 별도 입력의 번거로움 없이 이용자의 주소록에서 보호자를 선택할 수 있다. '발송' 버튼을 눌러 안심 알림을 발송하고, 발송된 알림은 '문자메시지함'과 '나의 택시아용기록'에서 확인할 수 있다.

3. 하차 단계

택시에 설치된 태그에 휴대폰을 태그해 주면 '안심알림발송' 화면이 나타난다. 하차 알림 발송을 누르면 탑승 시와 동일한 절차를 통하여 하차 알림을 지정된 보호자에게 발송할 수 있다.

2.3 NFC 택시안심 서비스 안심 태그 정보

택시 내 설치된 NFC 태그에는 멀티 레코딩 방식을 적용하여 스마트폰을 대었을 때 기존의 일방적인 앱이나 페이지를 호출하는 방식에서 벗어나 스마트폰을 대었을 때 URI가 호출하는 Choose List가 나타나고 이용자가 사용할 앱을 선택하여 이용할 수 있다. Taxi://URI를 호출할 수 있는 애플리케이션을 선택하고 태그의 택시정보 확인, 보호자 선택 후 안심 메시지 전송이 가능하다.

택시 안심 서비스 태그에는 [그림 1]과 같이 택시 번호, 운수회사 전화번호, 운수회사명, 애플리케이션 다운로드 URL이 저장된다.

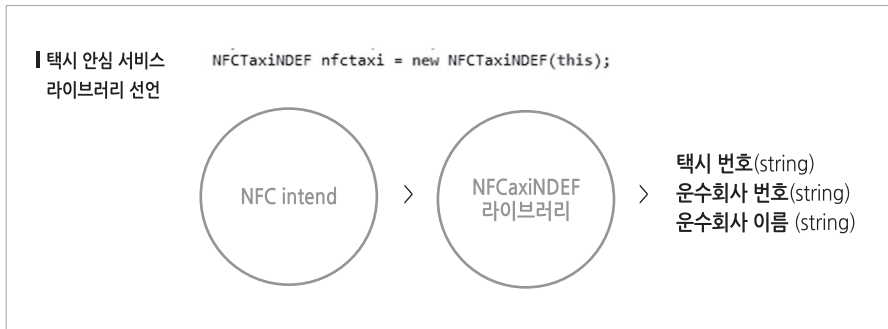
라이브러리는 [그림 2]와 같이 선언된다.

2.4 라이브러리 사용방법

[그림 3]~[그림 5]는 NFC 택시 안심 서비스 개발을 위한 공개된 라이브러리 사용방법이며, 그림과 같이 개발하면 쉽게 개인 또는 단체만의 NFC 택시 안심 서비스를 개발하여 서비스할 수 있다.

태그 Type	NXP Ultra Light C 튜닝 태그
태그 저장 정보	<div> <div>택시 번호</div> <div>운수회사 전화번호</div> <div>운수회사 명</div> </div> <div> <div>택시번호</div> <div>운수회사 전화번호</div> <div>운수회사 명</div> <div>애플리케이션 다운로드 URL</div> </div> <div> <div>경기34도0000 / 2600000</div> <div>02-111-2222 / 021112222</div> <div>00운수</div> </div>
태그 상세 정보	레코드 설명 [1]Type : URL(taxi://cn=차 번호?pn=운수회사 전화번호?tp=운수회사 명) - taxi://=URL scheme 으로 태그를 읽었을 때 OS에 호출하는 부분 - ?=URL에서 택시정보를 나누기 위한 부분 [2]Type : URL(market://details?id=package명) - taxi://를 받을 수 있는 애플리케이션이 없을 시 2번째 레코드를 실행 각 지자체를 애플리케이션 다운로드 페이지로 이동

[그림 1] 태그정보



[그림 2] NFC 택시 안심 서비스 라이브러리 선언

```

<uses-sdk android:minSdkVersion="10" />

<uses-permission android:name="android.permission.NFC" />

NFC 최소 Android Version = 9 권장 Android Version = 10
NFC 이용 Permission      = "android.permission.NFC"

<activity
    android:name="com.example.nfclibrarytest.MainActivity"
    android:label="@string/app_name" >
    <intent-filter> .....
        <action android:name="android.nfc.action.NDEF_DISCOVERED" />

        <category android:name="android.intent.category.DEFAULT" />

        <data android:scheme="taxi" />
    </intent-filter> .....
</activity>
  
```

필수등록 Intent Filter

[그림 3] 라이브러리 사용을 위한 기본 설정 방법

```

@Override
protected void onResume() {
    // TODO Auto-generated method stub
    super.onResume();
    NfcAdapter nfc = NfcAdapter.getDefaultAdapter(this);
    Intent itnt = new Intent(getApplicationContext(), getClass())
        .addFlags(Intent.FLAG_ACTIVITY_SINGLE_TOP | Intent.FLAG_ACTIVITY_CLEAR_TOP);
    PendingIntent pIntent = PendingIntent.getActivity(getApplicationContext(), 0, itnt, 0);
    IntentFilter filter = new IntentFilter(NfcAdapter.ACTION_TAG_DISCOVERED);
    IntentFilter[] tagFilters = new IntentFilter[] { filter };
    nfc.enableForegroundDispatch(this, pIntent, tagFilters, null);
}
@Override
public void onPause() {
    super.onPause();
    NfcAdapter nfc = NfcAdapter.getDefaultAdapter(this);
    nfc.disableForegroundDispatch(this);
}

```

해당 Activity에서 NFC를 읽겠다는 선언

해당 Activity에서 NFC를 읽겠다는 선언 해제

[그림 4] NFC 기본 설정 방법

```

/**
 * ArrayList<String> 방식 Return
 *
 * @author image5956
 *
 * @return ArrayList<String>
 * @throws Exception
 */

System.out.println("-----Array-----");
ArrayList<String> ndefArray = nfc taxi.getNdefArrayList(intent);
for (int index = 0; index < ndefArray.size(); index++) {
    System.out.println(ndefArray.get(index));
}

```

Array	ex /
1. 택시 번호(String)	서울가12345
2. 운수회사 번호(String)	023663666
3. 운수회사 이름(String)	안심운수

```

/**
 * HashMap<String, String> 방식 Return Key = { CarNumber, CompanyNumber, Company }
 *
 * @author image5956
 *
 * @return HashMap<String, String>
 * @throws Exception
 */

System.out.println("-----Hash-----");
HashMap<String, String> ndefHash = nfc taxi.getNdefHashMap(intent);
System.out.println(ndefHash.get("CarNumber"));
System.out.println(ndefHash.get("CompanyNumber"));
System.out.println(ndefHash.get("Company"));

```

Hash	ex /
CarNumber: 택시 번호	서울가12345
CompanyNumber: 운수회사 번호	023663666
Company: 운수회사 이름	안심운수

(a) Array List 및 Hash Map

```

/**
 * String 방식 Return
 *
 * @author image5956
 *
 * @return String
 * @throws Exception
 */

System.out.println("-----String-----");
String ndefString = nfc taxi.getNdefString(intent);
System.out.println(ndefString);

```

NDEF String	ex /
NDEF String	taxi://com:서울가12345?com=023663666?com=안심운수

```

/**
 * String[] 방식 Return [0] = CarNumber / [1] = CompanyNumber / [2] = Company
 *
 * @author image5956
 *
 * @return String[]
 * @throws Exception
 */

System.out.println("-----StringArray-----");
String[] ndefStringArray = nfc taxi.getNdefStringArray(intent);
for (int index = 0; index < ndefStringArray.length; index++) {
    System.out.println(ndefStringArray[index]);
}

```

NDEF String Array	ex /
1. 택시 번호	[0] : 서울가12345
2. 운수회사 번호	[1] : 023663666
3. 운수회사 이름	[2] : 안심운수

(b) String 및 String Array


[그림 5] Return 방식

3. 맺음말

앞서 설명한 본 표준은 현재 다양하게 서비스 중인 택시 안심 서비스 중 하나이다.

다만, 본 표준 방식은 기존의 방식과는 크게 다른 두 가지 장점을 보유하고 있다. 첫째, 오픈 API 형태로 그 기술을 공개하는 방식을 채택함으로써 다른 서비스들과는 다르게 접근하였다. 실제로 공개된 본 표준 방식으로 자체 NFC 택시 안심 서비스를 구현해서 서비스 중인 업체들도 있다. 둘째, 개인정보 유출에 대한 이슈 및 서비스 유지 비용에 대한 부분

이다. 본 표준 방식은 별도의 서버가 없이 구현이 가능한 방식이기 때문에 최근에도 가장 큰 이슈인 개인정보 유출 문제에서 자유로우며, 서비스 제공자에게 가장 부담스러운 유지보수 비용을 획기적으로 줄여준다는 점이다.

따라서, 본 표준은 기술적 표준이 가지는 가치뿐만 아니라 서비스 기획 및 개발에 대한 새로운 접근 방법을 제시했다고 판단되며, 향후 새로운 기술 기반의 대 국민서비스 개발 시에는 이와 같은 방식의 접근이 매우 효율적일 것으로 판단된다. 



내부 통신망의 라우터나 방화벽에서 외부로 나가는 데이터의 속성을 검사하고 흐름을 제어하는 보안 방법.

조직 내부에 존재하는 공격의 위협이 밖으로 전파되지 않도록 한다.

예를 들어, 내부 컴퓨터가 분산 서비스 거부 공격(DDoS attack)으로 좀비 PC가 된 경우 외부 컴퓨터를 감염시키지 못하게 차단하여 감염 피해를 최소화시킬 수 있다. 반대로 내부 통신망에 유입되는 외부 데이터를 검사하여 유해하다고 판단되면 차단하는 입구 필터링(ingress filtering) 방법이 있다.