



คู่มือการเชื่อมต่อระบบสร้าง QR ชำระเงิน



(v1.3 – 20220824 By Sorapong Ruanmanee)



Table of Contents

1. ความเป็นมา	3
2. ขั้นตอนการใช้งาน	3
3. ตำแหน่ง URL ที่เรียกใช้.....	4
4. ตำแหน่ง URL ของ WSDL	4
5. ข้อมูลที่ต้องส่งมา	4
6. การกำหนดค่า Ref1 และ Ref2.....	6
6.1 โครงสร้างของ Ref1 และ Ref2	6
6.2 ตัวอย่าง Ref1 และ Ref2 ของการชำระเงินค่าธรรมเนียมการศึกษา.....	7
7. ข้อมูลที่ได้รับกลับไป	8
8. ข้อมูลที่ได้รับกลับเมื่อมีข้อผิดพลาด	9
9. เว็บไซต์สำหรับทดสอบ	9
10. ผลตอบกลับ (Callback)	10
11. การใช้งานโดยภาษา PHP	10
12. การใช้งานโดยภาษา Java	12
13. การใช้งานโดยภาษา JavaScript.....	12
Appendix A - WSDL และ XSD.....	14

1. ความเป็นมา

เพื่อให้ผู้ใช้ที่ต้องการสร้างรูปภาพ QR Code ที่ไว้สำหรับชำระเงินให้กับ สำนักบริการการศึกษา มก. โดยเปิดบริการให้ส่งข้อมูลตั้งต้นเข้ามาเป็น WebServices SOAP และจะได้รับรูปภาพ QR Code กลับไปเป็นรูปแบบ PNG เพื่อนำไปแสดงผลบนแอปพลิเคชันปลายทางได้

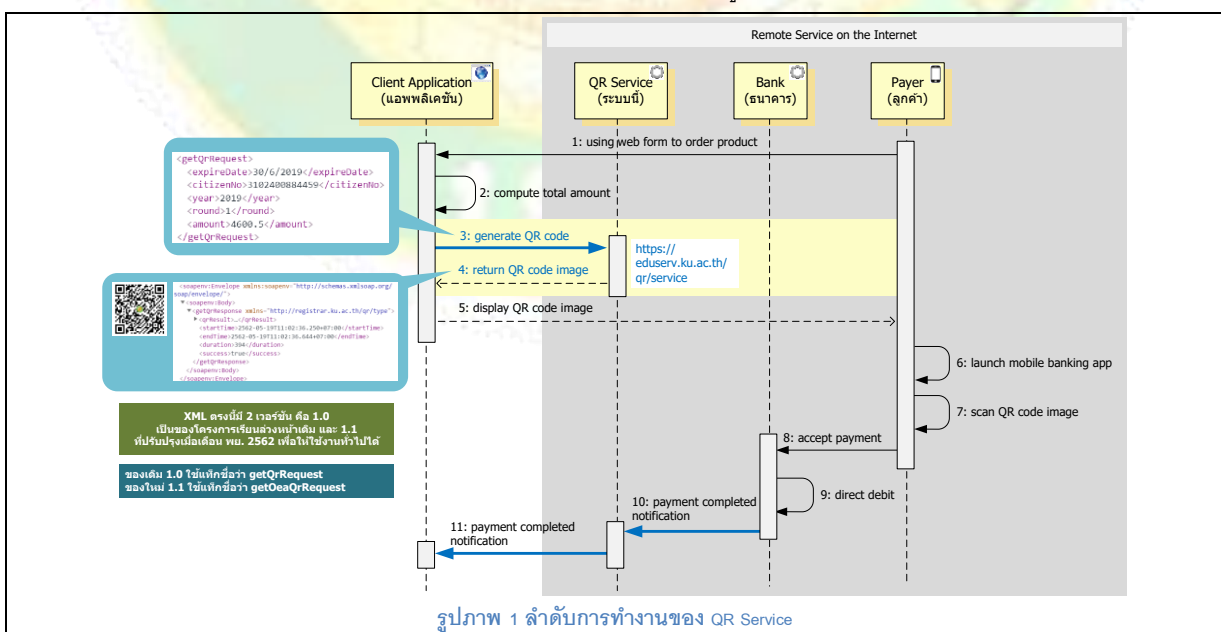
เอกสารฉบับนี้ทำขึ้นก่อน โดยระบบยังไม่สมบูรณ์มีวัตถุประสงค์เพื่อเผยแพร่ WSDL (Web Services Description Language) และ XSD (XML Schema) ที่จะยึดเป็นมาตรฐานในการแลกเปลี่ยนข้อมูล เพื่อให้แอปพลิเคชันปลายทางผู้เรียกใช้สามารถพัฒนาระบบไปพร้อมกันได้

2. ขั้นตอนการใช้งาน

ต่อไปนี้เป็นลำดับขั้นตอนการใช้งาน

1. แอปพลิเคชัน มีหน้าเว็บหรือ User Interface ให้ลูกค้าชำระเงิน
2. แอปพลิเคชัน ตรวจสอบจำนวนเงินกับลูกค้า
3. แอปพลิเคชัน ส่งคำสั่ง (เป็น SOAP) มาที่ระบบ QR เพื่อขอให้สร้าง QR Code
4. ระบบ QR รับคำสั่ง ตรวจสอบความถูกต้องข้อมูล แล้วบันทึกไว้ในฐานข้อมูล แล้วสร้าง QR Code เป็นรูปภาพ และส่งกลับไปในรูปแบบ Base64 ของเนื้อหาภาพประเภท PNG
5. แอปพลิเคชันเอาภาพ QR ไปแสดงให้ผู้ใช้ดู (โดยใช้แท็ก หรือ ImageViewer เป็นต้น)
6. ลูกค้าใช้แอปของธนาคารสแกนภาพ QR เพื่อชำระเงิน และปรากฏรหัส Ref1 และ Ref2 ให้เห็น
7. ลูกค้ากดปุ่มยืนยันชำระเงินที่แอปของธนาคาร
8. ธนาคารรับคำสั่งให้โอนเงินจากบัญชีลูกค้ามาที่บัญชีของ มก.
9. ธนาคารโอนเงินจากบัญชีลูกค้ามาที่บัญชีของ มก.
10. ธนาคารแจ้งมาที่ระบบ QR ว่ามีการโอนเงินแล้ว
11. ระบบ QR แจ้งกลับไป แอปพลิเคชัน ว่ามีการโอนเงินแล้ว

ขั้นตอนข้างต้นสามารถอธิบายได้ด้วยไดอะแกรมต่อไปนี้ (ตามหมายเลขลูกศร)



ในขณะที่เขียนเอกสารฉบับนี้เป็น (พย. 2562) ระบบได้เปิดใช้ไปแล้ว โดยสร้าง QR ให้กับโครงการเรียนล่วงหน้า และได้มีการปรับปรุงให้มี QR ที่เป็นแบบทั่วไปเพื่อให้ใช้กับระบบอื่นได้ด้วย การให้บริการจึงมี Operation แยกออกเป็น 2 ชื่อคือ getQr จะเป็นเวอร์ชัน 1.0 เดิมของโครงการเรียนล่วงหน้า แต่ getOeaQr จะเป็นเวอร์ชัน 1.1 ที่กำลังพัฒนาขึ้นมาใหม่เพื่อให้ใช้งานได้ทั่วไป

3. ตำแหน่ง URL ที่เรียกใช้

ตำแหน่ง URL ที่ให้เรียกใช้งานสำหรับระหว่างทดสอบอยู่ที่

<https://eduserv.ku.ac.th/qrTest/service>

หมายเหตุ – เปิดใช้งานวันที่ 29 พย. 2562

และตำแหน่ง URL ที่ให้เรียกใช้งานของระบบจริงจะอยู่ที่

<https://eduserv.ku.ac.th/qr/service>

4. ตำแหน่ง URL ของ WSDL

สำหรับแอปพลิเคชันที่ใช้วิธีการสร้างโค้ดอัตโนมัติจากไฟล์ WSDL และ XSD เมื่อเปิดระบบแล้ว (วันที่ 29 พย. 2562) ท่านสามารถดาวน์โหลด WSDL ได้จาก

<https://eduserv.ku.ac.th/qrTest/xml/qr.wsdl>

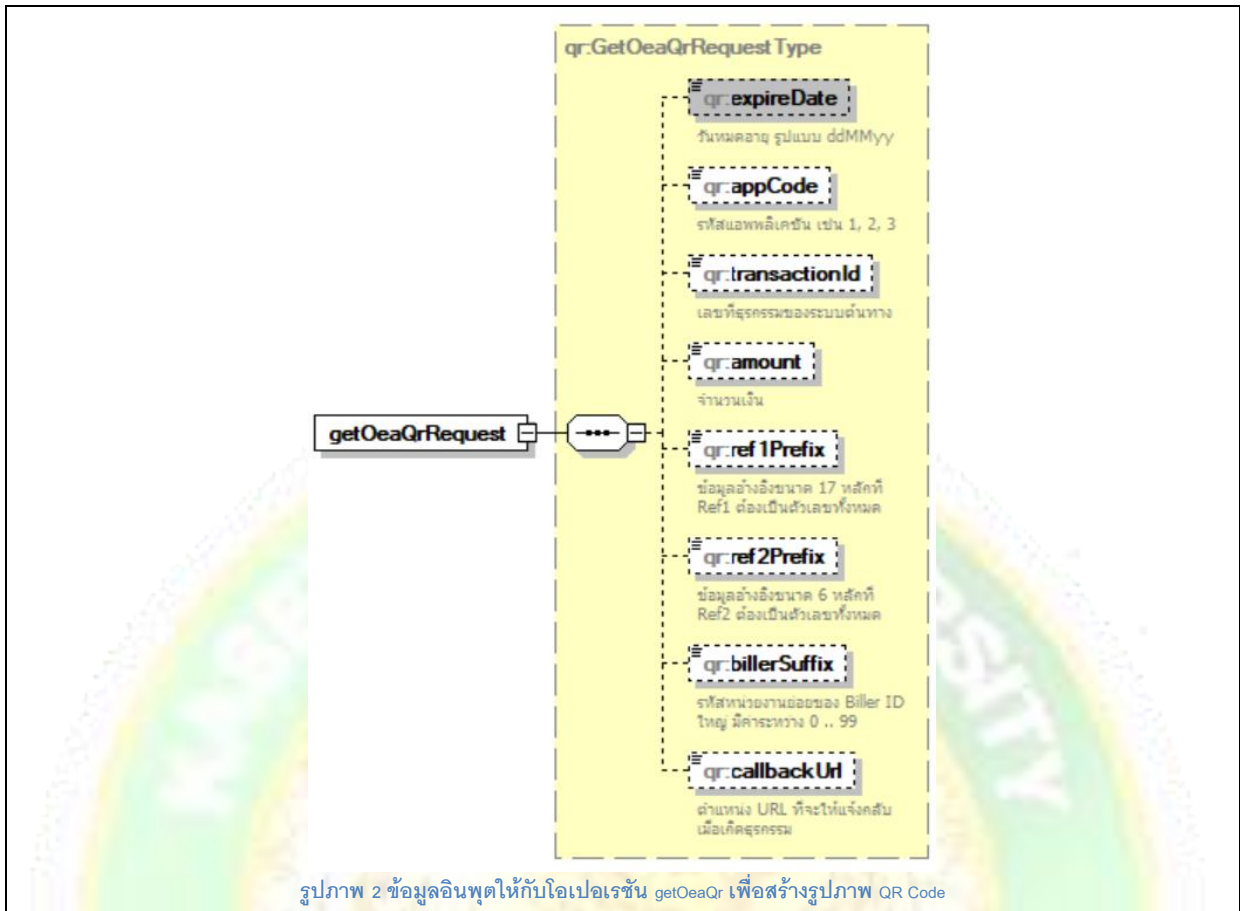
สำหรับระบบทดสอบ และสำหรับระบบจริงให้ดาวน์โหลดได้ที่

<https://eduserv.ku.ac.th/qr/xml/qr.wsdl>

ปัจจุบัน หากไม่สามารถดาวน์โหลดผ่าน URL ได้ ท่านสามารถใช้เนื้อหา WSDL และ XSD ที่ปรากฏใน Appendix A ของเอกสารนี้ไปใช้เพื่อสร้างโค้ดอัตโนมัติได้ก่อน (การสร้างโค้ดอัตโนมัติขึ้นอยู่กับแพลตฟอร์มการพัฒนา ระบบที่ท่านใช้ เช่น Java จะใช้เครื่องมือ wsimport ส่วน .NET จะใช้เครื่องมือ wsdl เป็นต้น)

5. ข้อมูลที่ต้องส่งมา

ผู้ใช้งานที่เรียกใช้ผ่านทาง Web Services จะต้องใช้ Operation ชื่อว่า getOeaQr และบอกรับข้อมูลเป็นโครงสร้างข้อมูลตามภาพต่อไปนี้



ตัวอย่างของ XML ที่ส่งมาเพื่อขอรูปภาพ QR Code (เวอร์ชัน 1.1)

ซอร์สโค้ด 1 ข้อมูล XML ที่ส่งมาที่เว็บเซิร์ฟเวอร์เพื่อขอบริการสร้างภาพ QR

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:type="http://registrar.ku.ac.th/qr/type">
  <soapenv:Header/>
  <soapenv:Body>
    <type:getOeaQrRequest>
      <type:expireDate>301119</type:expireDate>
      <type:appCode>34</type:appCode>
      <type:transactionId>1234</type:transactionId>
      <type:amount>5500.50</type:amount>
      <type:ref1Prefix>00001234567890123</type:ref1Prefix>
      <type:ref2Prefix>000012</type:ref2Prefix>
      <type:billerSuffix>87</type:billerSuffix>
      <type:callbackUri>http://xxx.ku.ac.th/callback/qr</type:callbackUri>
    </type:getOeaQrRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

ต่อไปนี้เป็นคำอธิบายฟิลด์ข้อมูลปรากฏในเนื้อหา XML

ตาราง 1 คำอธิบายของฟิลด์ข้อมูลอินพุต

ลำดับ	ฟิลด์ข้อมูล	ชนิดข้อมูล	คำอธิบาย	ตัวอย่าง
1	expireDate	ตัวอักษรในรูปแบบ ddMMyy	วันที่หมดอายุ ปี คศ.	ถ้าเกินกว่าวันที่ระบุนี้ ระบบต้นทางจะไม่รับชำระ โดย App ของ SCB จะรับรู้และตรวจสอบให้
2	appCode	ตัวอักษรแบบตัวเลข	ประเภทการชำระหรือรหัสของระบบ	ปัจจุบันมีค่าดังนี้ 01 – TCAS 02 – KU9 03 – ค่าปรับ 04 - เรียนล่วงหน้า



				05 - ตลาดนัดหลักสูตร
3	transactionId	ตัวอักษรแบบตัวเลข	เลขที่ธุรกรรม	เลขอะไรก็ได้ ให้ยึดระบบต้นทางเป็นหลัก ระบบ QR จะช่วยเก็บเลขนี้ไว้ให้และส่งกลับไปให้ตอนที่แจ้งกลับเมื่อมีการชำระเงินผ่าน QR ซึ่งจะให้ระบบต้นทางผูกโยงข้อมูลเพื่อเสร็จสิ้นธุรกรรมได้
4	amount	ตัวเลขทศนิยม 2 ตำแหน่ง	จำนวนเงิน	ใช้ ทศนิยม 2 ตำแหน่ง และต้องมีค่ามากกว่า 0 และมีค่าไม่เกิน 99999.99 (8 หลักรวมจุดทศนิยม)
5	ref1Prefix	ตัวอักษรแบบตัวเลข	ข้อมูลเสริมเฉพาะระบบ	เป็นตัวอักษรแบบตัวเลขขนาด 17 หลัก โดยให้แอปพลิเคชันต้นทางเป็นผู้กำหนดเองได้ (กรุณาดูที่ ภาพโครงสร้าง QR เพื่อความเข้าใจ)
6	ref2Prefix	ตัวอักษรแบบตัวเลข	ข้อมูลเสริมเฉพาะระบบ	เป็นตัวอักษรแบบตัวเลขขนาด 6 หลัก โดยให้แอปพลิเคชันต้นทางเป็นผู้กำหนดเองได้ (กรุณาดูที่ ภาพโครงสร้าง QR เพื่อความเข้าใจ)
7	billerSuffix	ตัวอักษรแบบตัวเลข	เลขระบุหน่วยงานย่อย	เป็นตัวเลข 2 หลักท้ายของ Merchant ID ที่ปกติมี 15 หลัก โดย 13 หลักแรกเป็นเลขนิติบุคคลของ มก. และ 2 หลักท้ายเป็นเลขที่หน่วยงานย่อย เช่น ปัจจุบัน สำนักบริการการศึกษา ใช้เลขที่ 87 ในการระบุหน่วยงานย่อย ถ้ามีเพียงหลักเดียว กรุณาเติม 0 ข้างหน้าด้วย เช่น 01 เป็นต้น
8	callbackUrl	ตำแหน่ง URL	ที่จะให้ส่งรีควีสต์ยืนยัน	เมื่อธนาคารแจ้งกลับมาจะส่งรีควีสต์กลับไปให้ระบบต้นขั้วพร้อมกับข้อมูล Ref1 Ref2 ที่ได้จากระบบธนาคาร

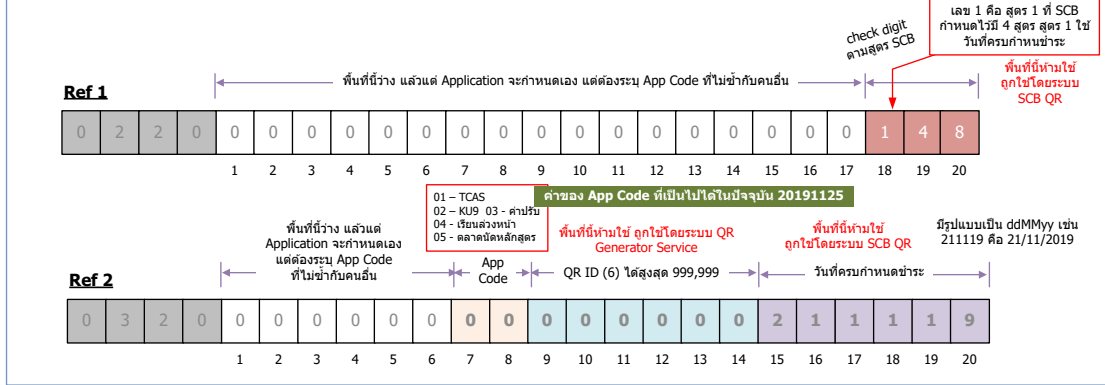
6. การกำหนดค่า Ref1 และ Ref2

ค่าของ Ref1 และ Ref2 เป็นค่าอ้างอิงที่ธนาคารจะส่งกลับมาให้เมื่อมีการชำระเงินเสร็จสิ้น และระบบต้นขั้วข้อมูลจะนำเอา Ref1 และ Ref2 ไปจับคู่เข้ากับข้อมูลตั้งต้นเพื่ออัปเดตกลับเข้าสู่ระบบของตนเอง อย่างไรก็ตามพบว่า Ref1 และ Ref2 มีความเป็นเฉพาะทางแยกกันไปตามบริบทการใช้งานของแต่ละประเภทการจ่ายเงิน ดังนั้น การออกแบบ Ref1 และ Ref2 ให้สามารถใช้งานร่วมกับ SCB และร่วมกับหน่วยงานย่อยใน มก. ได้จึงกำหนดให้ทำตามรูปแบบต่อไปนี้

6.1 โครงสร้างของ Ref1 และ Ref2

ต่อไปนี้เป็นภาพโครงสร้างข้อมูล Ref1 และ Ref2 ที่ได้จากการปรับปรุงให้สอดคล้องกับระบบ Check Digit ของ SCB ที่กำหนดว่าจะวางวันที่หมดอายุไว้ที่ Ref2 และจะวาง Check Digit ไว้ที่ 3 หลักสุดท้ายของ Ref1 ดังภาพต่อไปนี้

KU QR ปรับปรุงปี 62 (ใช้ SCB Check Digit) - ให้แต่ละ Application กำหนดเอง



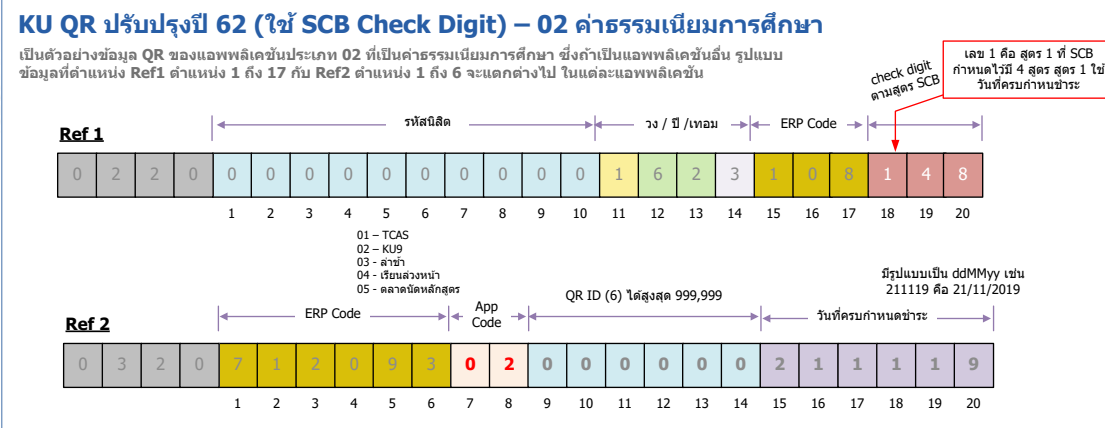
รูปภาพ 3 โครงสร้าง Ref1 และ Ref2 ของ QR ที่ใช้งานร่วมกันระหว่าง สบค. และ สบค. (ปี พ.ศ. 2562)

ในภาพข้างต้น ที่ Ref1 หลักที่ 18 – 20 จะถูกจองใช้โดย Check Digit ของ SCB และที่ Ref2 หลักที่ 15 – 20 จะถูกจองใช้โดยวันที่หมดอายุ ส่วนที่เหลือ มก. สามารถออกแบบเองได้ และจากการประชุมเมื่อวันที่ 25 พย. 2562 ระหว่าง สบค. และ สบค. ได้ออกแบบให้มี App Code เป็นตัวระบุแยกแยะว่าเป็น QR ของระบบอะไร และระบบ QR ได้ของจองพื้นที่หลักที่ 9 – 14 ของ Ref2 ไว้สำหรับเป็น QR ID ที่จะเป็นเลขรันนิ่งแยกกันไปตามแต่ละ App Code ด้วย

ส่วนที่ Ref1 หลักที่ 1 – 17 อนุญาตให้ผู้พัฒนาระบบที่มี App Code เป็นของตนเองสามารถออกแบบฟิลด์ข้อมูลได้เองว่าจะวางค่าอะไรในบริเวณนี้ รวมทั้งที่ส่วนที่ Ref2 หลักที่ 1 – 6 ก็เช่นกัน คือเป็นพื้นที่เปิดที่อนุญาตให้นักพัฒนาระบบออกแบบโครงสร้างข้อมูลเอง

6.2 ตัวอย่าง Ref1 และ Ref2 ของการชำระเงินค่าธรรมเนียมการศึกษา

ตัวอย่างของการออกแบบพื้นที่ Ref1 และ Ref2 เพื่อใช้กับระบบชำระเงินค่าธรรมเนียมการศึกษาเป็นดังภาพต่อไปนี้



รูปภาพ 4 โครงสร้างข้อมูล Ref1 และ Ref2 สำหรับการชำระค่าธรรมเนียมการศึกษา

ในภาพข้างต้นมีข้อสังเกตดังต่อไปนี้

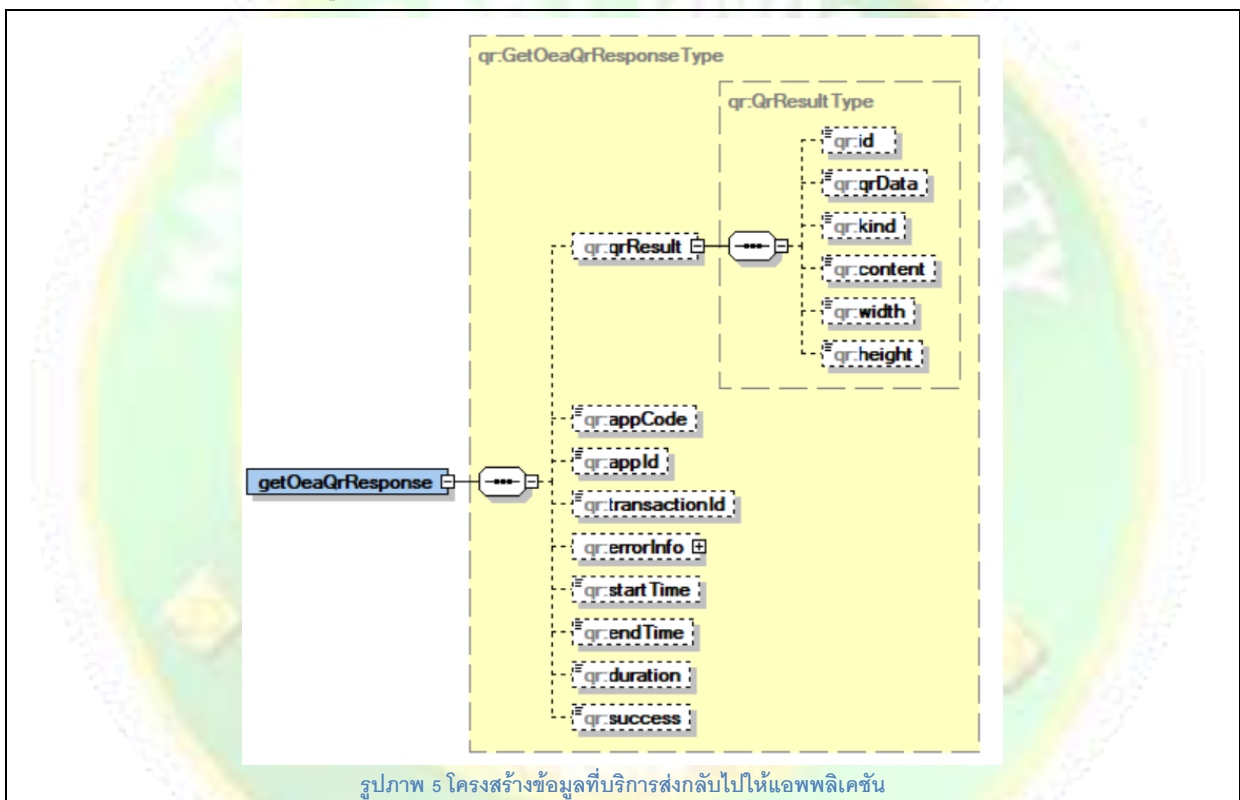
- ที่ Ref1 หลักที่ 18 – 20 เป็น Check Digit ตามสูตรของ SCB ที่ล๊อคไว้แล้วว่าต้องการใช้งาน 3 หลักนี้
- ที่ Ref1 หลักที่ 1 – 17 ทาง สบค. ได้ออกแบบให้วางข้อมูล รหัสบัตร วง ปี เทอม และส่วนหัวของ ERP Code
- ที่ Ref2 หลักที่ 7 – 8 กำหนดให้เป็นพื้นที่ของ App Code และ App Code ของค่าธรรมเนียมการศึกษาคือ 02
- ที่ Ref2 หลักที่ 9 – 14 กำหนดให้เป็นพื้นที่ของ QR ID (ตรงกับ App ID ในข้อมูล XML Schema)

- ที่ Ref2 หลักที่ 15 – 20 เป็นที่วางค่าวันที่หมดอายุ และกำหนดโดย SCB ให้วางที่ตำแหน่งนี้เท่านั้น
- ที่ Ref2 หลักที่ 1 – 6 ทาง สบค. ได้ออกแบบให้วางข้อมูลส่วนท้ายของ ERP Code

จากตัวอย่างข้างต้นจะเห็นว่า QR นี้มีมาตรฐานที่กำหนดโดย 3 ระบบ คือ ระบบ Check Digit ของ SCB ระบบ QR และระบบค่าธรรมเนียมของ สบค. ซึ่งหากท่านเป็นนักพัฒนาระบบใน มก. และต้องการใช้ QR กับระบบของท่านเอง ท่านสามารถตั้ง App Code ที่ไม่ซ้ำกับระบบอื่น และออกแบบพื้นที่บริเวณเปิดใน Ref1 และ Ref2 ได้ตามที่ท่านต้องการ เช่นเดียวกับตัวอย่างในหัวข้อนี้

7. ข้อมูลที่ได้รับกลับไป

ถ้าระบบได้รับคำสั่งที่ถูกต้องจะสร้างภาพ QR ในรูปแบบ PNG แล้วเข้ารหัสเป็น Base64 ส่งกลับมาให้ทาง SOAP Response โดยมีโครงสร้างข้อมูลดังนี้



ตัวอย่างของข้อมูล XML ที่ส่งกลับไปให้แอปพลิเคชัน

ซอร์สโค้ด 2 ข้อมูลรูปภาพ QR ที่ได้รับกลับไป

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <getOeaQrResponse xmlns="http://registrar.ku.ac.th/qr/type">
      <qrResult>
        <id>11</id>
        <kind>image/png</kind>
        <content>data:image/png;base64,
iVBORw0KGgoAAAANSUheUgAAAwAAAMMCAYAAAD+fyB7AACAAE1EQVR42uy9B3SW5bZo/W3dxbbt
vSsqKAoC0ktCQkLovXdfEAtFRRBBOiI91NBC77333nvvTXqVAKEktHnf5Fwde5/xnzP+c8/9/7PV
...
AAAAAEYAAAAAAAAYQAAAAAAAIQBAAAAAAQBgAAAAAAQBgAAAAAAOD95n8A1owymV+yNpAAAAAA
SUVORK5CYII=</content>
        <width>780</width>
        <height>780</height>
      </qrResult>
      <appCode>34</appCode>
      <appid>1</appid>
    </getOeaQrResponse>
  </soapenv:Body>
</soapenv:Envelope>
```



```
<transactionId>1234</transactionId>
<startTime>2562-11-27T12:06:13.850+07:00</startTime>
<endTime>2562-11-27T12:06:14.750+07:00</endTime>
<duration>900</duration>
<success>>true</success>
</getOeaQrResponse>
</soapenv:Body>
</soapenv:Envelope>
```

สังเกตว่าท่านควรตรวจสอบก่อนว่า success = true จึงจะดึงภาพ QR ไปใช้แสดงผล แต่ถ้า success = false แสดงว่ามีข้อผิดพลาดบางอย่างเกิดขึ้น

8. ข้อมูลที่ได้รับกลับเมื่อมีข้อผิดพลาด

หากระบบพบว่ามีข้อผิดพลาดจะส่งกลับมาเป็น XML ที่มีรูปแบบคล้ายดังต่อไปนี้

ซอร์สโค้ด 3 ตัวอย่างเนื้อหา XML ที่แอปพลิเคชันจะได้รับกลับเมื่อมีข้อผิดพลาดเกิดขึ้น

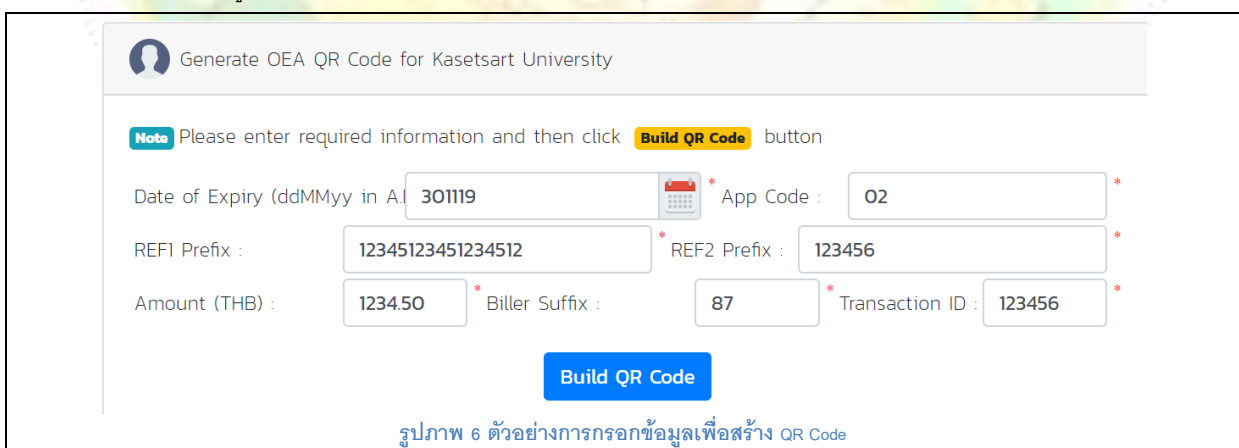
```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <getOeaQrResponse xmlns="http://registrar.ku.ac.th/qr/type">
      <errorInfo>
        <code>501</code>
        <message>Invalid request data</message>
        <stackTrace>Thai citizen No. must not be empty.
Purchase ID must not be empty.</stackTrace>
      </errorInfo>
      <startTime>2562-11-15T08:49:33.111+07:00</startTime>
      <endTime>2562-11-15T08:49:33.128+07:00</endTime>
      <duration>17</duration>
      <success>>false</success>
    </getOeaQrResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

9. หน้าเว็บสำหรับทดสอบ

ท่านสามารถทดสอบการสร้าง QR ได้โดยเข้าไปที่ตำแหน่ง URL

<https://eduserv.ku.ac.th/qrTest/page/OeaQrBuilder.html>

ตัวอย่างการกรอกข้อมูลเป็นดังนี้




The screenshot shows a web form titled "Generate OEA QR Code for Kasetsart University". It includes a note: "Please enter required information and then click Build QR Code button". The form contains several input fields with asterisks indicating they are required:

- Date of Expiry (ddMMyy in A.): 301119
- App Code: 02
- REF1 Prefix: 12345123451234512
- REF2 Prefix: 123456
- Amount (THB): 1234.50
- Billor Suffix: 87
- Transaction ID: 123456

A blue "Build QR Code" button is located at the bottom of the form.

รูปภาพ 6 ตัวอย่างการกรอกข้อมูลเพื่อสร้าง QR Code

เมื่อกดปุ่ม Build QR Code จะได้ภาพ QR ดังนี้

Amount (THB) :	<input type="text" value="1234.50"/>	* Biller Suffix :	<input type="text" value="87"/>	* Transaction ID :	<input type="text" value="123456"/>
<input type="button" value="Build QR Code"/>					
					
รูปภาพ 7 ภาพ QR ที่ได้จากการเรียกใช้งานผ่านเว็บเซอวิส					

ท่านสามารถใช้ App ของธนาคารเช่น SCB Mobile App ทดลองสแกนภาพ QR นี้เพื่อตรวจสอบว่าสามารถอ่านค่าได้อย่างถูกต้องหรือไม่

10. ผลตอบกลับ (Callback)

เมื่อธนาคารประมวลผลการโอนเงินเสร็จจะแจ้งกลับมายังระบบนี้และระบบนี้จะส่งกลับไปให้ระบบผู้ร้องขอ โดยมีรูปแบบข้อมูลคล้ายกับภาพต่อไปนี้

```
{
  "qrId": 14635,
  "appCode": "01",
  "appId": 5,
  "ref1Prefix": "50001234567890121",
  "ref2Prefix": "143333",
  "amount": "1.0",
  "expireDate": "311219",
  "qrCreatedTime": 1574956726000,
  "bankConfirmedTime": 1574956792000,
  "transactionId": "143333",
  "bankNotification": {
    "payeeProxyId": "099400015938287",
    "payeeProxyType": "BILLERID",
    "payeeAccountNumber": "2353002135",
    "payerAccountNumber": "2352030614",
    "payerAccountName": "ทดสอบ ทดสอบ",
    "payerName": "ทดสอบ ทดสอบ",
    "sendingBankCode": "014",
    "receivingBankCode": "014",
    "amount": "1.00",
    "transactionId": "cab14d7757e641e4b15743e62f5c1e6d",
    "transactionDateandTime": "2019-11-28T23:06:42.000+07:00",
    "billPaymentRef1": "50001234567890121164",
    "billPaymentRef2": "14333301000005311219",
    "currencyCode": "764",
    "channelCode": "PMH",
    "transactionType": "Domestic Transfers"
  }
}
```

11. การใช้งานโดยภาษา PHP

การใช้งานโดยภาษา PHP ให้เขียนโค้ดดังนี้

ซอร์สโค้ด 4 ตัวอย่างไฟล์ test_qr.php ที่ใช้ทดสอบการสร้างภาพ QR

```
<?php
$wsdl = 'https://eduserv.ku.ac.th/qr/xml/qr.wsdl';

$client = new SoapClient($wsdl,
    array(
        'trace' => TRUE,
        'exceptions' => 0,
        'style' => SOAP_DOCUMENT,
        'use' => SOAP_LITERAL
    )
);

$req = array(
    "expireDate"=>"301119",
    "appCode"=>"34",
    "transactionId"=>"1234",
    "amount"=>1.0,
    "ref1Prefix"=>"00001234567890123",
    "ref2Prefix"=>"000012",
    "billerSuffix"=>"87",
    "callbackUrl"=>"https://sriracha.ku.ac.th/payment/notify.php"
);
$res = $client->getOeaQr($req);
// echo var_dump($res);
?>

```

ผลการรันบนเครื่อง Local ที่ติดตั้ง XAMPP และรัน PHP ดังกล่าวข้างต้น



รูปภาพ 8 ผลการทำงานของโปรแกรมภาษา PHP เพื่อสร้าง QR Code

ข้อควรระวัง มีดังต่อไปนี้

- การทดลองด้วย PHP จะต้องเปิด Soap Extension ก่อนโดยแก้ไข php.ini ที่บรรทัด extension=soap ให้เอาคอมเม้นต์ออกแล้ว Restart Apache Server ใหม่
- billerSuffix = 87 เป็นของ สบค. บางเขน ถ้าเป็นที่วิทยาเขตอื่นจะมีเลขรหัสของตนเอง ให้ใช้เลขนั้น
- ref1Prefix มีขนาด 17 หลักเท่านั้น 3 หลักหลังต้องเว้นไว้ให้ SCB ใช้เพื่อเป็น Check Digit
- ref2Prefix มีขนาด 6 หลักเท่านั้น พื้นที่เหลือด้านหลังใช้โดยระบบเองและใช้โดย Expired Date ของ SCB
- ถ้าไม่ระบุตำแหน่ง callbackUrl ก็ได้ ระบบจะไม่ยิง Notification จาก Bank ต่อเนื่องไปให้
- ค่าของ transactionId เป็นค่าอะไรก็ได้เพื่อใช้ผูกโยงกับระบบต้นขั้วของตนเอง
- ค่าของ appCode ควรจะปรึกษากันกับทุกวิทยาเขตเนื่องจากมีรายการ appCode ที่เป็นที่ยู้กันอยู่แล้วว่าใช้ทำอะไรบ้าง

12. การใช้งานโดยภาษา Java

แนวทางการใช้งานโดยภาษา Java ให้ติดตั้ง JDK 1.8 (ไม่ควรสูงกว่านี้เพราะทดสอบที่ 1.8) แล้ว

- ดาวน์โหลดไฟล์ WSDL จาก <https://eduserv.ku.ac.th/qr/xml/qr.wsdl>
- ดาวน์โหลดไฟล์ XSD จาก <https://eduserv.ku.ac.th/qr/xml/qr.xsd> มาวางไว้ที่เดียวกัน
- ใช้คำสั่ง wsimport (ติดมากับ JDK) สร้าง Soap Client เช่นตัวอย่างนี้
- `wsimport -encoding UTF-8 -d c:/temp/qr/classes -s c:/temp/qr/src -keep -p ku.qr.client qr.wsdl`
- เมื่อได้ไฟล์ *.java แล้วให้เรียกใช้ QrService และ QrPort ฟังก์ชัน getOeaQr รูปแบบเดียวกับที่ระบุไว้ใน API ก็จะได้เนื้อหาไฟล์ภาพ QR กลับมา

รายละเอียดซอร์สโค้ดตัวอย่างจะพัฒนาคู่มือในรอบถัดไป

13. การใช้งานโดยภาษา JavaScript

แนวทางการใช้งานโดยภาษา JavaScript ให้ทำดังนี้

- ดาวน์โหลดไฟล์ qr.js จาก <https://eduserv.ku.ac.th/qr/js/qr.js>
- ดาวน์โหลดไฟล์ aes-main.js จาก <https://eduserv.ku.ac.th/qr/js/aes-main.js>

นำไฟล์ 2 ไฟล์นี้มารวมไว้ใน <script src ... > ตามแบบปกติของการเขียนโค้ดภาษา JavaScript หลังจากนั้นจะสามารถเขียนโค้ดเพื่อเรียกใช้ QR ได้ (ผ่านทาง Soap Service) เช่นนี้

ซอร์สโค้ด 5 ตัวอย่างบางส่วนของซอร์สโค้ด JavaScript ที่เรียกใช้ QR API เพื่อดึงภาพ QR

```
Page.prototype.clickBuild = function() {
    console.log('User click Build ...');

    // clear image data
    $('#qrImage').attr('src', '../images/qr.png');

    // get data from form
    var expireDate = $('#input[name=txtExpireDate]').val();
    var ref1Prefix = $('#txtRef1Prefix').val();
    var ref2Prefix = $('#txtRef2Prefix').val();
    var amount = $('#txtAmount').val();
    var transactionId = $('#txtTransactionId').val();
    var appCode = $('#txtAppCode').val();
    var billerSuffix = $('#txtBillerSuffix').val();

    // prepare request data
    var request = new GetOeaQrRequest();
    request.setExpireDate(expireDate);
    request.setAppCode(appCode);
    request.setRef1Prefix(ref1Prefix);
    request.setRef2Prefix(ref2Prefix);
    request.setBillerSuffix(billerSuffix);
    request.setTransactionId(transactionId);
    request.setAmount(amount);

    // request to service
    var xmlDoc = request.marshall();
    XUtils.debugXML(xmlDoc);
    var service = new QrService();
    var options = { noProgressBar : false, secure : false };
    var self = this;
    service.getOeaQr(request, true, function(resp, xml) {
        self.clickOeaQrResponse(resp, xml);
    }, options);
}

Page.prototype.clickOeaQrResponse = function(resp, xml) {
    console.log(xml);
    if(resp.getSuccess() == 'true') {
        // alert('Generate QR code successful.');
```

```
// retrieve image data putting into img tag
var result = resp.getQrResult();

// ดึงรหัสอ้างอิงไปเก็บไว้ในระบบต้นทาง
var qrId = result.getId(); // รหัส ID นี้ไม่ซ้ำกันเลย เป็นรหัสประจำตัว QR
var appId = resp.getAppId(); // รหัส ID นี้ซ้ำกันได้ AppCode ต่างกัน

// แสดงรูปภาพ
var content = result.getContent();
$('#qrImage').attr('src', content);

} else {
var error = resp.getErrorInfo();
var message = error.getMessage();
var stackTrace = error.getStackTrace();
var code = error.getCode();
alert(code + ' : ' + message + '\n' + stackTrace);
}
}
```

ในโค้ดข้างต้นอธิบายได้ดังนี้

- ดึงข้อมูลจากแบบฟอร์ม
- กรอกใส่เข้าไปในออบเจกต์ GetOeaQrRequest
- เรียกใช้เมธอด getOeaQr ของ QrService
- ได้ภาพกลับมาเข้ามาที่ฟังก์ชัน clickOeaQrResponse
- ดึงข้อมูลภาพตอบกลับด้วย getContent
- เอาภาพไปแสดงผลในแท็ก ด้วยการแทนที่เข้าไปในแอททริบิวต์ src

Appendix A - WSDL และ XSD

ถ้าท่านคัดลอกเนื้อหา WSDL จากเอกสารนี้ไปและบันทึกไว้เป็นไฟล์ qr.wSDL เมื่อเปิดดูด้วยเครื่องมือที่เข้าใจเว็บเซอร์วิส (เช่น XML Spy) ท่านควรจะพบภาพโครงสร้างเว็บเซอร์วิสดังนี้



ต่อไปนี้เป็นเนื้อหา WSDL ที่จะใช้งาน

ซอร์สโค้ด ๑ ไฟล์ qr.wSDL (Web Services Description Language)

```
<?xml version="1.0"?>
<definitions xmlns="http://schemas.xmlsoap.org/wsdl/"
xmlns:ns="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:p="http://www.w3.org/2001/XMLSchema"
xmlns:qr="http://registrar.ku.ac.th/qr/type"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://registrar.ku.ac.th/qr/service" name="QrService"
targetNamespace="http://registrar.ku.ac.th/qr/service">
  <types>
    <schema xmlns="http://www.w3.org/2001/XMLSchema">
      <import namespace="http://registrar.ku.ac.th/qr/type"
schemaLocation="qr.xsd"/>
    </schema>
  </types>
  <message name="getOeaQrRequest">
    <part name="getOeaQrRequestPart" element="qr:getOeaQrRequest"/>
  </message>
  <message name="getOeaQrResponse">
    <part name="getOeaQrResponsePart" element="qr:getOeaQrResponse"/>
  </message>
  <message name="getQrRequest">
    <part name="getQrRequestPart" element="qr:getQrRequest"/>
  </message>
  <message name="getQrResponse">
    <part name="getQrResponsePart" element="qr:getQrResponse"/>
  </message>
  <portType name="QrSystem">
    <operation name="getOeaQr">
      <input message="tns:getOeaQrRequest"/>
      <output message="tns:getOeaQrResponse"/>
    </operation>
    <operation name="getQr">
      <input message="tns:getQrRequest"/>
      <output message="tns:getQrResponse"/>
    </operation>
  </portType>
  <binding name="QrBinding" type="tns:QrSystem">
    <soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name="getOeaQr">
      <soap:operation
soapAction="http://registrar.ku.ac.th/qr/service/getOeaQr"/>
      <input>
        <soap:body use="literal"/>
      </input>
      <output>
        <soap:body use="literal"/>
      </output>
    </operation>
    <operation name="getQr">
      <soap:operation soapAction="http://registrar.ku.ac.th/qr/service/getQr"/>
      <input>
```

```
<soap:body use="literal"/>
</input>
<output>
  <soap:body use="literal"/>
</output>
</operation>
</binding>
<service name="QrService">
  <documentation>QrService</documentation>
  <port name="QrPort" binding="tns:QrBinding">
    <soap:address location="https://eduserv.ku.ac.th/qrTest/service"/>
  </port>
</service>
</definitions>
```

ในไฟล์ข้างต้นจะอ้างอิงโครงสร้างข้อมูลจากไฟล์ qr.xsd ซึ่งมีเนื้อหาดังนี้

ขอร์สโค้ด 7 ไฟล์ qr.xsd (XML Schema สำหรับเป็นโครงสร้างข้อมูลให้กับไฟล์ qr.wsdl)

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:qr="http://registrar.ku.ac.th/qr/type"
  targetNamespace="http://registrar.ku.ac.th/qr/type" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:complexType name="ConfirmType">
    <xs:annotation>
      <xs:documentation>ข้อมูลเก็บทุกครั้งที่ได้รับเมสเสจจากธนาคาร</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="id" type="xs:integer" minOccurs="0">
        <xs:annotation>
          <xs:documentation>รหัสข้อมูล</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="requestHost" type="xs:string" minOccurs="0"/>
      <xs:element name="requestAddress" type="xs:string" minOccurs="0"/>
      <xs:element name="requestPort" type="xs:integer" minOccurs="0"/>
      <xs:element name="localAddress" type="xs:string" minOccurs="0"/>
      <xs:element name="localPort" type="xs:integer" minOccurs="0"/>
      <xs:element name="requestPath" type="xs:string" minOccurs="0"/>
      <xs:element name="requestTime" type="xs:dateTime" minOccurs="0"/>
      <xs:element name="requestAgent" type="xs:string" minOccurs="0"/>
      <xs:element name="requestReferer" type="xs:string" minOccurs="0"/>
      <xs:element name="requestHeader" type="xs:string" minOccurs="0"/>
      <xs:element name="requestContentType" type="xs:string" minOccurs="0"/>
      <xs:element name="requestContentLength" type="xs:integer" minOccurs="0"/>
      <xs:element name="requestProtocol" type="xs:string" minOccurs="0"/>
      <xs:element name="requestMethod" type="xs:string" minOccurs="0"/>
      <xs:element name="requestScheme" type="xs:string" minOccurs="0"/>
      <xs:element name="requestUser" type="xs:string" minOccurs="0"/>
      <xs:element name="requestLocale" type="xs:string" minOccurs="0"/>
      <xs:element name="requestCookie" type="xs:string" minOccurs="0"/>
      <xs:element name="requestBody" type="xs:string" minOccurs="0"/>
      <xs:element name="payeeProxyId" type="xs:string" minOccurs="0"/>
      <xs:element name="payeeProxyType" type="xs:string" minOccurs="0"/>
      <xs:element name="payeeAccountNumber" type="xs:string" minOccurs="0"/>
      <xs:element name="payeeName" type="xs:string" minOccurs="0"/>
      <xs:element name="payerProxyId" type="xs:string" minOccurs="0"/>
      <xs:element name="payerProxyType" type="xs:string" minOccurs="0"/>
      <xs:element name="payerAccountNumber" type="xs:string" minOccurs="0"/>
      <xs:element name="payerName" type="xs:string" minOccurs="0"/>
      <xs:element name="sendingBankCode" type="xs:string" minOccurs="0"/>
      <xs:element name="receivingBankCode" type="xs:string" minOccurs="0"/>
      <xs:element name="amount" type="xs:string" minOccurs="0"/>
      <xs:element name="transactionId" type="xs:string" minOccurs="0"/>
      <xs:element name="transactionDateandTime" type="xs:string" minOccurs="0"/>
      <xs:element name="billPaymentRef1" type="xs:string" minOccurs="0"/>
      <xs:element name="billPaymentRef2" type="xs:string" minOccurs="0"/>
      <xs:element name="billPaymentRef3" type="xs:string" minOccurs="0"/>
      <xs:element name="currencyCode" type="xs:string" minOccurs="0"/>
      <xs:element name="beginTime" type="xs:dateTime" minOccurs="0"/>
```



```
<xs:element name="endTime" type="xs:dateTime" minOccurs="0"/>
<xs:element name="duration" type="xs:integer" minOccurs="0"/>
<xs:element name="notifyStatus" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Y = ได้แจ้งกลับลูกค้าแล้ว, N = ยังไม่แจ้งกลับ</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="status" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>สถานะ Y = ใช้งาน, N = ไม่ใช้งาน</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="createTime" type="xs:dateTime" minOccurs="0">
  <xs:annotation>
    <xs:documentation>วันที่สร้าง</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="createBy" type="xs:integer" minOccurs="0">
  <xs:annotation>
    <xs:documentation>รหัสผู้ใช้ที่เป็นผู้สร้าง</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="updateTime" type="xs:dateTime" minOccurs="0">
  <xs:annotation>
    <xs:documentation>วันที่แก้ไข</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="updateBy" type="xs:integer" minOccurs="0">
  <xs:annotation>
    <xs:documentation>รหัสผู้ใช้ที่เป็นผู้แก้ไข</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="payerAccountName" type="xs:string" minOccurs="0"/>
<xs:element name="channelCode" type="xs:string" minOccurs="0"/>
<xs:element name="transactionType" type="xs:string" minOccurs="0"/>
<xs:element name="notifyMax" type="xs:integer" minOccurs="0"/>
<xs:element name="notifyRetry" type="xs:integer" minOccurs="0"/>
<xs:element name="notifyTime" type="xs:dateTime" minOccurs="0"/>
<xs:element name="notifyNext" type="xs:dateTime" minOccurs="0"/>
<xs:element name="editState" type="xs:string" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ConfirmListType">
  <xs:annotation>
    <xs:documentation>รายการข้อมูลเก็บทุกครั้งที่ได้รับเมสเสจจากธนาคาร</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="confirm" type="qr:ConfirmType" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ConfirmItemType">
  <xs:annotation>
    <xs:documentation>ข้อมูลอย่างย่อของเก็บทุกครั้งที่ได้รับเมสเสจจากธนาคาร</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="id" type="xs:integer" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ConfirmItemsType">
  <xs:annotation>
    <xs:documentation>รายการข้อมูลอย่างย่อของเก็บทุกครั้งที่ได้รับเมสเสจจากธนาคาร</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="confirmItem" type="qr:ConfirmItemType" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="confirm" type="qr:ConfirmType">
  <xs:annotation>
```




```
<xs:documentation>ตัวอย่างข้อมูลเก็บทุกครั้งที่ได้รับมสเสจจากธนาคาร</xs:documentation>
</xs:annotation>
</xs:element>
<xs:complexType name="CounterType">
  <xs:annotation>
    <xs:documentation>ข้อมูล</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="name" type="xs:string" minOccurs="0"/>
    <xs:element name="value" type="xs:integer" minOccurs="0"/>
    <xs:element name="editState" type="xs:string" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="CounterListType">
  <xs:annotation>
    <xs:documentation>รายการข้อมูล</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="counter" type="qr:CounterType" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="CounterItemType">
  <xs:annotation>
    <xs:documentation>ข้อมูลอย่างย่อของ</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="name" type="xs:string" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="CounterItemsType">
  <xs:annotation>
    <xs:documentation>รายการข้อมูลอย่างย่อของ</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="counterItem" type="qr:CounterItemType" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="counter" type="qr:CounterType"/>
<xs:complexType name="ErrorType">
  <xs:annotation>
    <xs:documentation>ข้อมูลเก็บข้อผิดพลาดทั้งหมดในระบบ</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="id" type="xs:integer" minOccurs="0">
      <xs:annotation>
        <xs:documentation>รหัสข้อมูล</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="exception" type="xs:string" minOccurs="0"/>
    <xs:element name="message" type="xs:string" minOccurs="0"/>
    <xs:element name="stackTrace" type="xs:string" minOccurs="0"/>
    <xs:element name="freeMemory" type="xs:decimal" minOccurs="0"/>
    <xs:element name="freeDiskSpace" type="xs:decimal" minOccurs="0"/>
    <xs:element name="status" type="xs:string" minOccurs="0">
      <xs:annotation>
        <xs:documentation>สถานะ Y = ใช้งาน, N = ไม่ใช้งาน</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="createTime" type="xs:dateTime" minOccurs="0">
      <xs:annotation>
        <xs:documentation>วันที่เวลาที่สร้าง</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="createBy" type="xs:integer" minOccurs="0">
      <xs:annotation>
        <xs:documentation>รหัสผู้ใช้ที่ผู้สร้าง</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```

```
<xs:element name="updateTime" type="xs:dateTime" minOccurs="0">
  <xs:annotation>
    <xs:documentation>วันที่เวลาที่แก้ไข</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="updateBy" type="xs:integer" minOccurs="0">
  <xs:annotation>
    <xs:documentation>รหัสผู้ใช้ที่เป็นผู้แก้ไข</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="editState" type="xs:string" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ErrorListType">
  <xs:annotation>
    <xs:documentation>รายการข้อมูลเก็บข้อผิดพลาดทั้งหมดในระบบ</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="error" type="qr:ErrorType" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ErrorItemType">
  <xs:annotation>
    <xs:documentation>ข้อมูลอย่างย่อของเก็บข้อผิดพลาดทั้งหมดในระบบ</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="id" type="xs:integer" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ErrorItemsType">
  <xs:annotation>
    <xs:documentation>รายการข้อมูลอย่างย่อของเก็บข้อผิดพลาดทั้งหมดในระบบ</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="errorItem" type="qr:ErrorItemType" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="error" type="qr:ErrorType">
  <xs:annotation>
    <xs:documentation>ตัวอย่างข้อมูลเก็บข้อผิดพลาดทั้งหมดในระบบ</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:complexType name="NotifyType">
  <xs:annotation>
    <xs:documentation>ข้อมูลเก็บทุกครั้ง ที่ Notify กลับไปที่ Consumer</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="id" type="xs:integer" minOccurs="0">
      <xs:annotation>
        <xs:documentation>รหัสข้อมูล</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="requestHost" type="xs:string" minOccurs="0"/>
    <xs:element name="requestAddress" type="xs:string" minOccurs="0"/>
    <xs:element name="requestPort" type="xs:integer" minOccurs="0"/>
    <xs:element name="localAddress" type="xs:string" minOccurs="0"/>
    <xs:element name="localPort" type="xs:integer" minOccurs="0"/>
    <xs:element name="requestPath" type="xs:string" minOccurs="0"/>
    <xs:element name="requestTime" type="xs:dateTime" minOccurs="0"/>
    <xs:element name="requestAgent" type="xs:string" minOccurs="0"/>
    <xs:element name="requestReferer" type="xs:string" minOccurs="0"/>
    <xs:element name="responseStatusLine" type="xs:string" minOccurs="0"/>
    <xs:element name="responseBody" type="xs:string" minOccurs="0"/>
    <xs:element name="requestHeader" type="xs:string" minOccurs="0"/>
    <xs:element name="requestBody" type="xs:string" minOccurs="0"/>
    <xs:element name="qrId" type="xs:string" minOccurs="0">
      <xs:annotation>
        <xs:documentation>วันที่หมดอายุ</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```



```
</xs:annotation>
</xs:element>
<xs:element name="confirmId" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>เลขประชาชน</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="beginTime" type="xs:dateTime" minOccurs="0"/>
<xs:element name="endTime" type="xs:dateTime" minOccurs="0"/>
<xs:element name="duration" type="xs:integer" minOccurs="0"/>
<xs:element name="isSuccess" type="xs:boolean" minOccurs="0"/>
<xs:element name="errorMessage" type="xs:string" minOccurs="0"/>
<xs:element name="errorStackTrace" type="xs:string" minOccurs="0"/>
<xs:element name="status" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>สถานะ Y = ใช้งาน, N = ไม่ใช้งาน</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="createTime" type="xs:dateTime" minOccurs="0">
  <xs:annotation>
    <xs:documentation>วันเวลาที่สร้าง</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="createBy" type="xs:integer" minOccurs="0">
  <xs:annotation>
    <xs:documentation>รหัสผู้ใช้ที่ผู้สร้าง</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="updateTime" type="xs:dateTime" minOccurs="0">
  <xs:annotation>
    <xs:documentation>วันเวลาที่แก้ไข</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="updateBy" type="xs:integer" minOccurs="0">
  <xs:annotation>
    <xs:documentation>รหัสผู้ใช้ที่ผู้แก้ไข</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="editState" type="xs:string" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="NotifyListType">
  <xs:annotation>
    <xs:documentation>รายการข้อมูลเก็บทุกครั้งที่ Notify กลับไปที่ Consumer</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="notify" type="qr:NotifyType" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="NotifyItemType">
  <xs:annotation>
    <xs:documentation>ข้อมูลอย่างย่อของเก็บทุกครั้งที่ Notify กลับไปที่
Consumer</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="id" type="xs:integer" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="NotifyItemsType">
  <xs:annotation>
    <xs:documentation>รายการข้อมูลอย่างย่อของเก็บทุกครั้งที่ Notify กลับไปที่
Consumer</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="notifyItem" type="qr:NotifyItemType" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="notify" type="qr:NotifyType">
```



```
<xs:annotation>
  <xs:documentation>ตัวอย่างข้อมูลเก็บทุกครั้งที่ Notify กลับไปที่ Consumer</xs:documentation>
</xs:annotation>
</xs:element>
<xs:complexType name="QrType">
  <xs:annotation>
    <xs:documentation>ข้อมูลเก็บทุกครั้งที่มาสร้าง QR Image</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="id" type="xs:integer" minOccurs="0">
      <xs:annotation>
        <xs:documentation>รหัสข้อมูล</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="requestHost" type="xs:string" minOccurs="0"/>
    <xs:element name="requestAddress" type="xs:string" minOccurs="0"/>
    <xs:element name="requestPort" type="xs:integer" minOccurs="0"/>
    <xs:element name="localAddress" type="xs:string" minOccurs="0"/>
    <xs:element name="localPort" type="xs:integer" minOccurs="0"/>
    <xs:element name="requestPath" type="xs:string" minOccurs="0"/>
    <xs:element name="requestTime" type="xs:dateTime" minOccurs="0"/>
    <xs:element name="requestAgent" type="xs:string" minOccurs="0"/>
    <xs:element name="requestReferer" type="xs:string" minOccurs="0"/>
    <xs:element name="requestHeader" type="xs:string" minOccurs="0"/>
    <xs:element name="requestContentType" type="xs:string" minOccurs="0"/>
    <xs:element name="requestContentLength" type="xs:integer" minOccurs="0"/>
    <xs:element name="requestProtocol" type="xs:string" minOccurs="0"/>
    <xs:element name="requestMethod" type="xs:string" minOccurs="0"/>
    <xs:element name="requestScheme" type="xs:string" minOccurs="0"/>
    <xs:element name="requestUser" type="xs:string" minOccurs="0"/>
    <xs:element name="requestLocale" type="xs:string" minOccurs="0"/>
    <xs:element name="requestCookie" type="xs:string" minOccurs="0"/>
    <xs:element name="requestBody" type="xs:string" minOccurs="0"/>
    <xs:element name="expireDate" type="xs:string" minOccurs="0">
      <xs:annotation>
        <xs:documentation>วันที่หมดอายุ</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="citizenNo" type="xs:string" minOccurs="0">
      <xs:annotation>
        <xs:documentation>เลขประชาชน</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="year" type="xs:string" minOccurs="0">
      <xs:annotation>
        <xs:documentation>ปี</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="round" type="xs:string" minOccurs="0">
      <xs:annotation>
        <xs:documentation>รอบที่</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="amount" type="xs:string" minOccurs="0">
      <xs:annotation>
        <xs:documentation>จำนวนเงิน</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="purchaseId" type="xs:string" minOccurs="0">
      <xs:annotation>
        <xs:documentation>เลขที่สั่งซื้อ</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="ref1" type="xs:string" minOccurs="0">
      <xs:annotation>
        <xs:documentation>อ้างอิง 1</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="ref2" type="xs:string" minOccurs="0">
      <xs:annotation>
```



```
<xs:documentation>อ้างอิง 2</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="qrData" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>ข้อมูล QR ทั้งหมดที่สร้างขึ้น</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="callbackUrl" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>URL ที่ชี้แจงกลับ</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="beginTime" type="xs:dateTime" minOccurs="0"/>
<xs:element name="endTime" type="xs:dateTime" minOccurs="0"/>
<xs:element name="duration" type="xs:integer" minOccurs="0"/>
<xs:element name="status" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>สถานะ Y = ใช้งาน, N = ไม่ใช้งาน</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="createTime" type="xs:dateTime" minOccurs="0">
  <xs:annotation>
    <xs:documentation>วันเวลาที่สร้าง</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="createBy" type="xs:integer" minOccurs="0">
  <xs:annotation>
    <xs:documentation>รหัสผู้ใช้ที่ผู้สร้าง</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="updateTime" type="xs:dateTime" minOccurs="0">
  <xs:annotation>
    <xs:documentation>วันเวลาที่แก้ไข</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="updateBy" type="xs:integer" minOccurs="0">
  <xs:annotation>
    <xs:documentation>รหัสผู้ใช้ที่ผู้แก้ไข</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="apiVersion" type="xs:string" minOccurs="0"/>
<xs:element name="billerSuffix" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>รหัสก่อนหลัง 2 ตัวของ Biller ID เป็นหน่วยงานย่อย เช่น 87 (เวอร์ชัน
1.1)</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="ref1Prefix" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>17 หลักแรกของ ref1 (เวอร์ชัน 1.1)</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="ref2Prefix" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>6 หลักแรกของ ref2 (เวอร์ชัน 1.1)</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="transactionId" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>รหัสอ้างอิงต้นขั้ว (เวอร์ชัน 1.1)</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="appCode" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>รหัส App Code เช่น 1 - TCAS, 2 - KU9 (เวอร์ชัน
1.1)</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="appId" type="xs:integer" minOccurs="0">
```



```
<xs:annotation>
  <xs:documentation>รหัส QR ภายใต้อ App (เวอร์ชัน 1.1)</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="editState" type="xs:string" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="QrListType">
  <xs:annotation>
    <xs:documentation>รายการข้อมูลเก็บทุกครั้งที่มาสร้าง QR Image</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="qr" type="qr:QrType" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="QrItemType">
  <xs:annotation>
    <xs:documentation>ข้อมูลอย่างย่อของเก็บทุกครั้งที่มาสร้าง QR Image</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="id" type="xs:integer" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="QrItemsType">
  <xs:annotation>
    <xs:documentation>รายการข้อมูลอย่างย่อของเก็บทุกครั้งที่มาสร้าง QR Image</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="qrItem" type="qr:QrItemType" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="qr" type="qr:QrType">
  <xs:annotation>
    <xs:documentation>ตัวอย่างข้อมูลเก็บทุกครั้งที่มาสร้าง QR Image</xs:documentation>
  </xs:annotation>
</xs:element>
<!-- Below content is copy from
[d:/workshop/payment/qr/WebContent/xml/qr inc.xsd] -->
<!-- =====>
<!--สร้าง OEA QR Code (เวอร์ชันใหม่ 20191128 ปรับปรุงให้สามารถใช้ SCB Check Digit ได้)-->
<!-- =====>
<xs:complexType name="GetOeaQrRequestType">
  <xs:sequence>
    <xs:element name="expireDate" type="xs:string" minOccurs="0">
      <xs:annotation>
        <xs:documentation>วันหมดอายุ รูปแบบ yyMMdd</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="appCode" type="xs:string" minOccurs="0">
      <xs:annotation>
        <xs:documentation>รหัสแอปพลิเคชัน เช่น 1, 2, 3</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="transactionId" type="xs:string" minOccurs="0">
      <xs:annotation>
        <xs:documentation>เลขที่ธุรกรรมของระบบต้นทาง</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="amount" type="xs:double" minOccurs="0">
      <xs:annotation>
        <xs:documentation>จำนวนเงิน</xs:documentation>
      </xs:annotation>
    </xs:element>
    <xs:element name="ref1Prefix" type="xs:string" minOccurs="0">
      <xs:annotation>
        <xs:documentation>ข้อมูลอ้างอิงขนาด 17 หลักที่ Ref1 ต้องเป็นตัวเลขทั้งหมด</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```



```
</xs:element>
  <xs:element name="ref2Prefix" type="xs:string" minOccurs="0">
    <xs:annotation>
      <xs:documentation>ข้อมูลอ้างอิงขนาด 6 หลักที่ Ref2 ต้องเป็นตัวเลขทั้งหมด
</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="billerSuffix" type="xs:string" minOccurs="0">
    <xs:annotation>
      <xs:documentation>รหัสหน่วยงานย่อยของ Biller ID ใหญ่ มีค่าระหว่าง 0 ..
99</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="callbackUrl" type="xs:string" minOccurs="0">
    <xs:annotation>
      <xs:documentation>ตำแหน่ง URL ที่จะให้แจ้งกลับเมื่อเกิดธุรกรรม</xs:documentation>
    </xs:annotation>
  </xs:element>
</xs:sequence>
</xs:complexType>
<xs:element name="getOeaQrRequest" type="qr:GetOeaQrRequestType"/>
<xs:complexType name="GetOeaQrResponseType">
  <xs:sequence>
    <xs:element name="qrResult" type="qr:QrResultType" minOccurs="0"/>
    <xs:element name="appCode" type="xs:string" minOccurs="0"/>
    <xs:element name="appId" type="xs:integer" minOccurs="0"/>
    <xs:element name="transactionId" type="xs:string" minOccurs="0"/>
    <xs:element name="errorInfo" type="qr:ErrorInfoType" minOccurs="0"/>
    <xs:element name="startTime" type="xs:dateTime" minOccurs="0"/>
    <xs:element name="endTime" type="xs:dateTime" minOccurs="0"/>
    <xs:element name="duration" type="xs:integer" minOccurs="0"/>
    <xs:element name="success" type="xs:boolean" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="getOeaQrResponse" type="qr:GetOeaQrResponseType"/>
<!--=====-->
<!--สร้าง QR Code (เวอร์ชันที่ใช้กับโครงการเรียนล่วงหน้า)-->
<!--=====-->
<xs:complexType name="GetQrRequestType">
  <xs:sequence>
    <xs:element name="expireDate" type="xs:string" minOccurs="0"/>
    <xs:element name="citizenNo" type="xs:string" minOccurs="0"/>
    <xs:element name="year" type="xs:integer" minOccurs="0"/>
    <xs:element name="round" type="xs:integer" minOccurs="0"/>
    <xs:element name="amount" type="xs:double" minOccurs="0"/>
    <xs:element name="ref" type="xs:string" minOccurs="0"/>
    <xs:element name="callbackUrl" type="xs:string" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="getQrRequest" type="qr:GetQrRequestType"/>
<xs:complexType name="QrResultType">
  <xs:sequence>
    <xs:element name="id" type="xs:integer" minOccurs="0"/>
    <xs:element name="qrData" type="xs:string" minOccurs="0"/>
    <xs:element name="kind" type="xs:string" minOccurs="0"/>
    <xs:element name="content" type="xs:string" minOccurs="0"/>
    <xs:element name="width" type="xs:integer" minOccurs="0"/>
    <xs:element name="height" type="xs:integer" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ErrorInfoType">
  <xs:sequence>
    <xs:element name="code" type="xs:string" minOccurs="0"/>
    <xs:element name="message" type="xs:string" minOccurs="0"/>
    <xs:element name="stackTrace" type="xs:string" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="GetQrResponseType">
  <xs:sequence>
    <xs:element name="qrResult" type="qr:QrResultType" minOccurs="0"/>
```



```
<xs:element name="errorInfo" type="qr:ErrorInfoType" minOccurs="0"/>
<xs:element name="ref" type="xs:string" minOccurs="0"/>
<xs:element name="startTime" type="xs:dateTime" minOccurs="0"/>
<xs:element name="endTime" type="xs:dateTime" minOccurs="0"/>
<xs:element name="duration" type="xs:integer" minOccurs="0"/>
<xs:element name="success" type="xs:boolean" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
<xs:element name="getQrResponse" type="qr:GetQrResponseType"/>
<!--=====-->
<!--ใช้งานทั่วไป-->
<!--=====-->
<xs:complexType name="PageNavigatorType">
  <xs:sequence>
    <xs:element name="rowCount" type="xs:integer" minOccurs="0"/>
    <xs:element name="rowPerPage" type="xs:integer" minOccurs="0"/>
    <xs:element name="pageCount" type="xs:integer" minOccurs="0"/>
    <xs:element name="pageIndex" type="xs:integer" minOccurs="0"/>
    <xs:element name="startOffset" type="xs:integer" minOccurs="0"/>
    <xs:element name="endOffset" type="xs:integer" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ItemType">
  <xs:sequence>
    <xs:element name="id" type="xs:integer" minOccurs="0"/>
    <xs:element name="code" type="xs:string" minOccurs="0"/>
    <xs:element name="name" type="xs:string" minOccurs="0"/>
    <xs:element name="value" type="xs:string" minOccurs="0"/>
    <xs:element name="selected" type="xs:boolean" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ItemListType">
  <xs:sequence>
    <xs:element name="item" type="qr:ItemType" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="CommonResponseType">
  <xs:sequence>
    <xs:element name="success" type="xs:boolean" minOccurs="0"/>
    <xs:element name="resultId" type="xs:integer" minOccurs="0"/>
    <xs:element name="resultType" type="xs:string" minOccurs="0"/>
    <xs:element name="resultMessage" type="xs:string" minOccurs="0"/>
    <xs:element name="resultScript" type="xs:string" minOccurs="0"/>
    <xs:element name="errorCode" type="xs:string" minOccurs="0"/>
    <xs:element name="errorMessage" type="xs:string" minOccurs="0"/>
    <xs:element name="errorStackTrace" type="xs:string" minOccurs="0"/>
    <xs:element name="errorScript" type="xs:string" minOccurs="0"/>
    <xs:element name="confirmMessage" type="xs:string" minOccurs="0"/>
    <xs:element name="beginTime" type="xs:dateTime" minOccurs="0"/>
    <xs:element name="endTime" type="xs:dateTime" minOccurs="0"/>
    <xs:element name="duration" type="xs:integer" minOccurs="0"/>
    <xs:element name="userId" type="xs:string" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="commonResponse" type="qr:CommonResponseType"/>
<!--=====-->
<!--ใช้สำหรับส่ง Id เป็น list เพื่อไปลบ หรือ ไปค้นหาข้อมูล -->
<!--=====-->
<xs:complexType name="IdListType">
  <xs:sequence>
    <xs:element name="id" type="xs:integer" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
</xs:schema>
```




ถ้าท่านจะใช้ไฟล์ qr.wsdl และ qr.xsd เพื่อสร้างโค้ดอัตโนมัติ ท่านจำเป็นต้องวางทั้ง 2 ไฟล์นี้ไว้ที่โฟลเดอร์เดียวกัน และชี้ตำแหน่งมาที่ไฟล์ qr.wsdl และให้เครื่องมือของท่านสร้างโค้ดอัตโนมัติตามภาษาของแพลตฟอร์มนั้นๆขึ้นก็จะทำให้สามารถนำไปพัฒนาแอปพลิเคชันเพื่อเรียกใช้งานระบบนี้ได้

