

## Transit Module 21: Mobile Fare Ticketing/Payment

**Ken Leonard:** ITS Standards can make your life easier. Your procurements will go more smoothly and you'll encourage competition, but only if you know how to write them into your specifications and test them. This module is one in a series that covers practical applications for acquiring and testing standards-based ITS systems.

I am Ken Leonard, director of the ITS Joint Program Office for USDOT and I want to welcome you to our newly redesigned ITS standards training program of which this module is a part. We are pleased to be working with our partner, the Institute of Transportation Engineers, to deliver this new approach to training that combines web-based modules with instructor interaction to bring the latest in ITS learning to busy professionals like yourself.

This combined approach allows interested professionals to schedule training at your convenience, without the need to travel. After you complete this training, we hope that you will tell colleagues and customers about the latest ITS standards and encourage them to take advantage of the archived version of the webinars.

ITS Standards training is one of the first offerings of our updated Professional Capacity Training Program. Through the PCB program we prepare professionals to adopt proven and emerging ITS technologies that will make surface transportation safer, smarter and greener which improves livability for us all. You can find information on additional modules and training programs on our website [www.pcb.its.dot.gov](http://www.pcb.its.dot.gov)

Please help us make even more improvements to our training modules through the evaluation process. We look forward to hearing your comments. Thank you again for participating and we hope you find this module helpful.

**Polly Okunieff:** Welcome to Module 21 on Standards related to the Mobile Fare Ticketing and Payment. The prerequisites for this course are Module 10 and Module 12. We'll be reviewing some of those topics in the course of this webinar. I'm Polly Okunieff. Welcome. I'm your instructor for today. I have over 34 years of experiences as a Systems Business and Data Analyst and 26 of those years developing integrated systems for public transit including fare payment, electronic fare payment, and mobile apps and I'm very pleased to start the course.

We have five learning objectives. These include review concepts from the previous modules, Module 10 and 12, and the concepts related to mobile payment. We'll define the ecosystem of the mobile payment apps. We'll also describe elements of the electronic fare payment business model in the context of an ISO standard. We'll review several case studies that are related to mobile fare apps. And we'll discuss the challenges for transit agencies related to mobile technologies.

Getting started, the first learning objective has two parts. Briefly we'll review the electronic fare payment Module 10 and the advanced electronic fare payment Module 12 concepts which, of course, I mentioned before is a prerequisite for this course. And we'll identify the features of mobile payment within the context of electronic fare payment environment and understand the features and existing trends for transit mobile fare payment from a recently published TCRP Synthesis Number 125 by Candace Brakewood.

As described in both transit Modules 10 and 12, the Architecture of the Electronic Fare Payment System, (EFPS), includes the components of fare media and there are different kinds of fare media including the mobile device, the reader, which may be located on the bus on the platform

## Transit Module 21: Mobile Fare Ticketing/Payment

or in a portable or inspector mobile app. A local device to process, that is validate, store and forward the fare media information. A central computer, which may include a processor to manage payment from different sources. Or processing for fare or financial transactions, the product and the customer management tools. Optionally, a payment processor or payment gateway to process credit or debit cards by directing the bankcard transaction to the bankcard network. And finally, also, an optional regional clearinghouse for integration of third party or regional partner to reconcile transactions, perform settlement and keep track of financial liabilities.

As discussed in Module 12, the account-based system stores customer transactions, financial, and profile information in a back-office customer account management system rather than just the minimal amount of information that's stored on the card in card-based systems. The differences between the traditional account-based system and the mobile fare system is that the mobile fare system, the mobile ticketing app, also serves as a point of sales channel and in addition some mobile ticketing apps do not require validation through an electronic reader, and so the reading, the validation services are provided by a driver or the operator.

The mobile ticketing app serves two major purposes. As a point of sales device, it provides the same POS functions as a customer service terminal, a website, or a ticket vending machine; or as a proof of payment where it loads and activates fare products for validation by a reader, the conductor, inspector, or an operator when a proof of payment is required. The types of fare products that may be stored or loaded cut across all the types of fare products that are supported by transit agencies, including period passes or tickets.

This slide on Mobile Payment System Operator was adapted from transit Module 12. The mobile payment system operator or actor includes the following responsibilities: operates the mobile payment system; recruits issuers and enables their integration; offers mobile app in virtual wallets such as Apple Pay, Google Pay, and various bank and virtual cards; is the registrar of a virtual card by cardholders; and performs identity management functionality or front-end cardholder identification and security through tokenization. Examples of these types of mobile wallets include Apple and Google or bank-issued virtual cards. In fact, the mobile app responsibilities may be separated into multiple actors or organizations where each bullet in here of the role and responsibilities may be an independent organization or some of the bullets may be grouped together. For most implementations, the procuring transit agency is shielded from the multitude of actors or organizations participating because most of these mobile app vendors today are implemented as a fare payment as a service, sometimes referred to as FPaaS, more generally called software as a service or SaaS.

The growth and access of smartphones and the use of these devices for payment are increasing worldwide, although have been much slower in the U.S. According to industry sources, more adults own smartphones than ever before, 81 percent as reported by the Pew Research Center. People are increasing their use of mobile payments, specifically according to the payments industry intelligence research from 2016 to 2022, mobile payment transaction volumes in the U.S. is projected to grow 36.6 percent over year. This projection was made prior to COVID-19 pandemic. Since that time the estimates are that the use will increase even more. And based on many transit agency data analysis reported on agency websites, more people are using mobile fare apps, payment apps than ever before.

So, let's transition now to the TCRP Report. The TCRP Synthesis Study 148, Business Models for Mobile Fare Apps by Candace Brakewood was conducted in 2019 to explore the types of

## Transit Module 21: Mobile Fare Ticketing/Payment

business models and implementation methods currently in practice in the industry. The synthesis is composed of a literature search, survey of current practice and case examples of detailed descriptions related to the topic. The topics that were surveyed include fare app characteristics and features including those listed here; mobile platforms and devices; system status; transit modes accepting mobile payment; fare products offered; regional integration; accessibility features; and other media. We'll review these features in the next few slides.

Here are some of the survey results from the study. With respect to platforms, all agencies identified the use of Android and IOS or Apple Operating Systems for their mobile apps and status, early in its development, many agencies piloted a mobile app. Today over 77 percent of respondents have deployed permanent fare payment apps and of note a third of them were deployed in 2017. All transit modes use a mobile fare app, but you should note that the number that reflect the percentage of the survey responses should not be used to reflect the industry market share.

Some additional characteristics of fare products offered. The fare products offered on an app are similar or identical to those offered by a ticket vending machine, website, or customer service office. Some agencies wait to issue special or discount fares until the system is matured, but eventually sell all types of products. Regional integration. Most of the respondents have deployed standalone systems, not integrated with their electronic fare payment systems. Those standalone models tend to be the simplest to deploy and most are deployed as software is a service, although with the deployment of several regional account-based systems that use application programming interfaces or APIs, this will change. With respect to accessibility features, the mobile apps are generally using accessible features native to the mobile device including large fonts, high contrast, audio assist, and now even voice activated services like Siri or Google. At the time of the survey, which was mid-May 2019, several agencies indicated that wearables were on their roadmap but had not yet been deployed and this has changed since the publication mostly for merchants who allow open payment using watches, activity bands like Fitbit, but not quite yet for transit wallets or the virtual cars.

The survey also included information on the methods for activation and validation. Activation is defined as the process of making mobile fare payment valid for a given period of time. Close to 30 percent of the respondents indicated that activation of the product needed to be done when the smartphone was online. Many of the others indicated the activation may occur while online or offline. And validation refers to the process or method used to assure that fare products are not fraudulent or expired and have sufficient funds to pay for the fare. The primary validation method, the survey asked about the method used to validate the fare product. In the majority of the cases, that is 87 percent, the major method for the validation was the visual validation or the Flash pass. Some also include NFC or QR codes to validate the payment.

In the screen shot the time, date, type of product, that is a three-hour pass, and expiration, are displayed directly on the screen and as a visual method for the driver to verify payment or as an alternative to a reader detecting the QR code. In 2019, when the study was conducted, there were no agencies in the U.S. using NFC to validate fare products. At that time, NFC was too slow to execute a transaction; now it's not and we'll see that it's changed over the last two years.

The TCRP Survey also included information on additional features integrated into the app or linked to the app. The TCRP Synthesis Survey asked about customer facing features and a third responded that their apps included real time transit information, transit schedules and trip

## Transit Module 21: Mobile Fare Ticketing/Payment

planning applications. Fewer, although some, also indicated service alerts and reporting as well as integration with mobility on demand options such as ride hailing or bike-sharing operators.

The TCRP Survey also identified the roles and responsibilities of the vendor, agency, and others. The responses show how dependent agencies are on the vendors selected to operate payment processes, app posting, and PCI compliance, which is the technology and financial functions. However, marketing outreach and adoption is typically the responsibility of the agency. Some useful definitions for the cited roles are as follows.

Payment processing refers to managing the financial transactions for sales transactions, validation, authentication, and tokenization. Hosting the app refers to the software that implements the back-office functionality including account and user management, fare products, sales and refunds reporting, and other kinds of apps functionality. Customer service for riders refers to technical support for the app, sales support for sales and refunds and other commendation, and complaints that are collected. Marketing the app to riders includes outreach and training materials to travelers and potential customers. This may include advertising and promotion of special services.

PCI Compliance or payment card industry security standard certification is the certification to protect the device data and functions of the system. Credit card companies and banks require annual PCI testing to ensure compliance. For agencies to better manage the deployment and future integration opportunities they have with their transit fare apps, they need to understand the business models and architectures in the context of the roles and responsibilities. And standards can help and that's what we're going to talk about for the most part today.

So, let's review first the standards described in Modules 10 and 12. The next few slides will be an adaptation of slides that were in the original Modules. So, how do standards help? Well, this slide presented in Module 10 shows that the mobile fare app used similar standards as contact list cards. That is the ISO/ IEC 14443, but on a mobile device it's referred to as a virtual or a transit card and the software emulates the same characteristics as a physical card. So, a virtual card is an electronic replica of a physical card and it usually contains a randomly generated credit card number that changes every time a credit card is used for purchase.

Many recently deployed mobile fare apps use NFC to communicate to a reader. Not all mobile devices use NFC. Some use a QR code with QR code readers or a Flash pass. In May of 2020, the NFC Forum approved four specifications that provide faster and more robust data exchange than the older versions called the Tag NFC Data Exchange Format Exchange Protocol Specification or TNEP. As of May 2020, these have not yet been implemented in handsets because it was just approved.

The NFC Forum is a governing body that develops and promulgates NFC specifications. It is composed of vendors, original equipment manufacturers, and communication operators who develop, manufacture, and use the NFC equipment. The Card Network Specifications also apply to the virtual cards. They're still proprietary and support their own software libraries. Similar to the smart cards, agreements between card network operator and the merchants are still required for all virtual wallets. The card network operating rules are also unique for each network. Many mobile payment app vendors have existing agreements with one or more leading card networks for their payment app similar to (EP), the Electronic Fare Payment Systems.

## Transit Module 21: Mobile Fare Ticketing/Payment

So now, we've come to the end of the first learning objective and we have a question. Who does the primary marketing for an agency's fare app to riders? Is it A, a vendor; B, social media; C, an agency; or D, an app store?

The correct answer is the agency. Respondents of the TCRP Synthesis 148 responded that the agency markets the app to the riders. It's incorrect, vendor is incorrect. Only 7 percent of vendors promote an agency's fare app. It's not social media. Although social media may help promote the app, the primary marketing is performed by the agency. And, it's not the app store, although an app store might help suggest an app, the primary marketing is performed by the agency.

Now we'll move on to learning objective number 2. At the end of this chapter you should be able to distinguish the variety of mobile payment methods and describe the technologies and standards that enable mobile payment.

There are four mobile payment types. The first is a virtual wallet like credit cards and any ISO 14443 product. It typically uses NFC as its payment proof of payment validation. Currently used by transit agencies such as the Ventra Card which may be stored in an Apple Wallet or the OMNY Card, which may be stored in Google Wallets. Payment apps use tokenization process to pay. Also, use NFC payment method validation. It was these apps were initiated in 2012.

Some may use QR codes or visual validation or a virtual bankcard in the app. Examples include the CapMetro App in Austin, Texas or TouchPass Mobile App from the Victor Valley Transit Authority. Although in the next two, although these will not be discussed in this course, there are other payment apps such as peer-to-peer payment methods such as the Venmo card where the payment is between the person from one bank account to another. Then there's also crypto currency apps or wallets such as Coinbase, the Coinbase, app or the Blockchain app. Although not used by U.S. transit agencies as a proof of payment, they are allowed for payment in Argentina and the Japan Railway Group is also investigating whether or not to use Blockchain cryptocurrency.

There are several proof of purchase methods that are important in describing the capabilities of the mobile fare system. Three proof of payment or validation methods that are currently available in the U.S. include the following. So, there is the visual verifiable validation or the V3 sometimes called the Flash pass. The V3 include different colors or animation for the day or period, for example, a week, and it typically also includes the fare product period of activation and duration on the screen. QR codes encapsulate the fare product including validation period and rider type. And NFC is any virtual card that as virtual cards become more available, NFC is being used by transit agencies who deploy a transit wallet or use contactless credit cards. This was identified in transit Module 12 as an open payment method.

Using the model from Module 12, I'm going to walk through the actors and/or actions for the four payment methods in the next few slides. So, here's an example of the V3. Typically, the V3 animation changes the image on the phone screen to help avoid fraud. The change may be a change of color, an image or the movement of the image, or the date that is unique to the day and product type. Using the process flow architecture from the transit Modules 10 and 12, the fare media is the point of purchase-- the proof or purchase and the operator performs the function of the reader to validate the fare product. No electronic record is stored on the passenger boardings. The mobile app may identify where and when the ticket or pass was activated but that is all the information that is recorded. The app also serves as a sales device

## Transit Module 21: Mobile Fare Ticketing/Payment

or a point of sale, looks up account information from the mobile back office and customer account system and information may include the pass transaction. So, they could pull up the pass transactions, purchases and ticket usages on their account on their phone as well.

QR codes are based on the QR codes standard, the ISO/IEC 18004, which was passed in 2015. In addition to the QR code, fare tickets include the fare or pass detail, the ticket type, the date of activation and expiration. The process is similar to the smartcard. The customer scans their mobile phone showing the bar code across the reader. The code is validated against a valid or invalid code list. The information is stored and forwarded to the central computer which may be operated by the transit agency or vendor. Many agencies include a handheld scanner or reader mounted in a bus or gate to validate the bar code. Applications of this technology may be a Flashpass with the operator reviewing the ticket; however, without the scanner there is a high risk of fraud.

The model above shows how the virtual card stored in a virtual wallet works. When activated, the virtual card is read by the NFC reader and the validator device may check that the card number is not on the action list, that is the invalid or blacklist. It then stores and forwards the message to the central computer and the agency may store the information in a regional clearinghouse. The central computer then forwards the credit card information to a payment gateway processor for authentication via the bankcard network and the issuers send authentication messages back through the same path to the validator.

For mobile phones using Google Pay, the wallet function adds an additional step to this model. When the virtual card is activated, the wallet requests a token from Google Pay Cloud to unlock and then encrypt the credit card information in order to generate a message using the 14443 standard. This will be described in more detail in the following slides. The transit wallet and the virtual card typically work in the same way as open payment method, because the virtual transit card uses the bankcard network media, MasterCard or Visa, for example. So, it's treated just like an open payment transaction with electronic fare payment system being the merchant.

I want to step back and discuss different architectures that mobile fare app systems adopt. These are described in the TCRP Synthesis 148 Report. So, the Walled Garden is a closed and proprietary system. It's by the operator of that application. A Deep Link may open another app such as transit fare app opening a ride hailing app like Uber or Lyft, typically using parameters like origin destination. So, when used in a transit application, the application may automatically insert a transit station or stop as the origin or destination. APIs or Application Programming Interfaces allow one app to exchange information with the central computer with account management system of another provider, like a mobile app that enables a customer to book a ride on Uber or vice versa.

This approach is emerging in multimodal transportation booking, ticketing, and payment apps and provides the most promise. Most of the next section will discuss this approach and the specifications and standards that support it. In one recent publication, Tri-Met has identified the APIs as critical to the success of their entire system. And so you can see that this is where the industry is going. The Software Development Kit (SDK) is a set, provides a readymade set of tools, code, and interfaces to a central computer that a third party operates like Google Map allows users to book hotels without leaving their site, but produces a template for each hotel with the same characteristics. So, APIs are a subset of the capabilities of the SDK. But SDK has a few more functions that it provides and also additional services.

## Transit Module 21: Mobile Fare Ticketing/Payment

So, in terms of security, secure storage, secure elements is a tamper-proof chip to encode and decrypt data and secure the personally identifiable information or PII. Most security today is provided using an onboard embedded storage element but in the past external storage was provided through elements such as sim cards or SD cards. Alternatively, some handset and operating systems use host card emulation or HCE, a software emulation process using tokenization to secure the PII. In the case of the HCI, the mobile handset uses cloud storage to store the PII like Amazon and Google Pay. When activated, the HCE processes use tokenization to request a token for the encrypted information. Only token information is stored and transferred by the mobile app. The token is linked to a card on file which is stored in the cloud. In both cases, when the user activates the payment, the NFC controller converts the data or the token from the secure element to the communications medium.

The next slide shows how the interactions between the secure element and NFC controller work. With respect to defining tokenization, tokenization is the process of replacing sensitive data with unique identifiable symbols that retain all the essential information about the data without compromising the security. Encryption, which is an alternative to tokenization, uses a key while tokenization creates a temporary alias for an account number or PII.

So, the secure element works directly with the NFC controller. Below the dotted line in this graphic here is a secure space and it's not accessible, it's not reachable by other applications and functions above the line. So, the space is where the secure element in the HCE is located. When the consumer holds their smartphone or smart wearable over an NFC reader, the NFC controller in the device routes all the data from the reader directly to the secure element. The secure element itself performs the communication with the NFC terminal and no mobile app function is involved in the transaction. And after the transaction is complete, the app can query the secure element directly from the transaction status and notify the user.

To increase NFC performance, that is the speed and security, PII is stored in a secure element that is isolated from other functions and the apps on the smartphone or if this is like a watch or a wearable, that has payment capability. The NFC controller is also located in the secure area and when the virtual card or ticket is activated and it comes near the NFC reader, the reader will read the presented token. For a system that uses HCE and tokenization, the NFC controller will trigger the HCE to exchange messages with the cloud platform using some kind of tokenization method to secure information to exchange with the NFC transceiver each one of those things, the tokenization capability, may be a third party that's doing this or the mobile app.

As described in the previous slides, there are many actors, roles that manage the mobile environment, more than in a card-based system. The ISO community published—the International Standards Organization published a standard called Interoperable Fare Management System or IFMS for short. The standard describes the roles and responsibilities and information flows in electronic fare payment system. The model divides the ecosystem into several domains. That's the external system manager, the IFMS, interoperable fare managers, and identification manager. The IFMS includes the roles depicted in the orange box. The orange box is the interoperable fare manager. The blue box is the external system manager. And the green is the identification manager. These, each one of the smaller boxes in these domains are roles and have responsibilities associated with them. They're described in the supplement in section 1.3. But note, not all the roles are required in an electronic fare payment system, particularly most IFMS roles are assigned to the transit agencies and their vendor.

## Transit Module 21: Mobile Fare Ticketing/Payment

However, in the mobile fare app there are multiple players including that are in the interoperable fare manager box and they include the cellular service providers, handset manufacturers, native SDK providers, app store providers, app developers, mobile fare app operators, cloud providers, tokenization managers, and the transit agencies. Examples are described as follows. So, you have the account provider and the account provider provisions and hosts customer accounts. Media provider, this may be the handset provider or the virtual card provider. The media retailer may be in the Apple or Google Store that provisions the mobile app. The complexity of the roles will become more apparent when we discuss the various business models from the TCRP Synthesis 148 and it's easier to understand the mobile app development characteristics using the role-based architectures. We'll show two examples on the next couple pages.

So, first, in the transit wallet model, the IFMS roles show the interaction using different fare system configurations. For example, an open payment system using a transit wallet is one of them, this one here. Applying this fare payment, the IFMS roles, it would look like this. So, assume we're using the Ventra system in Metro Chicago as an example. So, Ventra is who is the regional fare system for the Chicago Transit Authority base and Metra is the transit wallet provider, that is the payment provider. They hold the electronic purse or payment method even if it's a credit card—even if the credit card that is stored in the Ventra account system, which is converted to the cost of the ticket or a fare, which is—and they are the product owner. And when the passenger boards the transport, which is converted to the cost of a ticket or fare, when the passenger board the transport service operator. The product owner is the transit wallet owner.

So, let's assume that the transport service operator is Pace, but it could be any of the three agencies that accept Ventra. The Ventra system provides top up and fulfillment at over 700 retailers such as Jewel, which is the supermarket, CVS, and other retailers. MasterCard is the bankcard issuer and the Ventra's bank in this case, money network may be the financial institutes that process the card on behalf of Ventra. Note that there are many more actors and interface messages exchanged that are not depicted in this diagram, so this is just a subset of them.

A simpler model, which is the next one, where you have just a bankcard that could be used as both a point of sales as well as a proof of payment, is the virtual bankcard. And so ,for example, in this example, assume I'm using my credit card which may be Visa that's provided by my bank, and that's the payment provider, so that's both the bankcard issuer and the merchant acquire on the left-hand side. I'm the passenger and I want to board the New York City Transit subway, who is the transport service operator. At the gate, I buy a fare from the product owner, who is the New York Metropolitan Transport Authority OMNI Payment System. And the authentication is done by the bankcard issuer and that serves as the proof of payment. So, this is a much simpler approach to it. Fewer of the responsibilities fall on the transport service operator. There's really no ticketing or fare app to rely on. The agency acts as the merchant.

So, let's go to the question. What payment access method is the most proprietary? Is it A, the SDK; B, the API; C, the walled garden; or D, the Deep link? And the answer is the walled garden. And that refers to applications that are closed with the intention of securing and restricting access. It's not the SDK. That's typically a toolkit for incorporating open functions. It's not the APIs, which are open specifications for exchanging information. And it's not the deep link which is a uniform reference link or URL that accesses another application. This selection is also restricted and proprietary but not as restricted as the walled garden.

## Transit Module 21: Mobile Fare Ticketing/Payment

So, we come to learning objective number 3. And it's to in this objective we will classify the business models. At the end of this chapter you should be able to describe the different business models in the TCRP Synthesis 148 and identify how each will be described using the IFMS roles and responsibilities model. And we'll describe the purpose of a use case and how it supports the flow of information between the various roles in the IFMS architecture.

Let's go back to the synthesis and discuss the five business models defined for mobile fare apps. The first description for each model is included in the supplement. So, the first one is the shared app. It's a single platform for multiple agencies. The second one is typically a software service in which agencies or regions can brand a payment service. These don't have hardware and they're based on Flashpass technologies. Similar to the White Label with the validation hardware is similar to the White Label, but hardware such as readers and mobile inspection tools are used to validate the proof of purchase. The open payment app uses a virtual card technology. This includes both transit wallets and virtual bankcards stored in a smartphone's virtual wallet such as Apple Pay or Google Pay. And the last one is the software development kit. These are APIs and other tools that support integration of transit into other apps or integrate other mobility providers such as mobility as a service and transit into mobile fare apps. The next few slides show the business models using the IFMS roles and responsibilities.

So, the first business model, the shared app, is a turnkey system that enables transit agencies to deploy a mobile fare product almost immediately. Although each agency can brand and customize their products, the app is shared by several agencies that reside side by side in the same application. And the example of a shared app is the transit token app. Note that there is limited or no integration with fare systems because these systems are not connected to the hardware, with the fare box or electronic fare payment software.

The architecture colors you see in the boxes on the left-hand side, blue is the shared app vendor, brown is the transit agency, green refers to the app store and orange is the customer or smartphone. The roles are dominated by the vendor including the roles of application retailer by app stores like Apple and Google. The customer downloads the app from the store to their phone and uses it to buy their tickets from the account holder and retailer or the point of sales. Multiple agencies are served by the shared app vendor and their products live side by side on the same platform. The product owner and the service provider roles belong to the transit agency. And like I said, transit token is an example of a shared app mobile fare app.

The White Label app is developed by a mobile fare app vendor that could be branded by an organization. It's also a turnkey product and it's procured as a service, a software as a service model. The user interface product offering account management and reporting services are configurable for an agency to customize using their brand. Unlike the shared app, a white label app tends to be implemented for a single agency or a region and does not share the app with other organizations. Note that the architecture is the same as the previous model but only one transit agency is served by this app. Its vendors include Bytemark, Masabi, Moovel but it's not the total number of vendors who provide these kinds of services.

The White Label with hardware is similar to the white label business model but includes fare validation methods, either QR codes or NFC for validating customer fare products. It's also turnkey and subscription-based. This model has greater potential for integration because it may share a reader with other fare systems and the information is collected onboard the vehicle. Note that the model is similar to the white label app, but there is a collection and forwarding function that is served by the cloud in the graphic. Collection and forwarding is implemented as

## Transit Module 21: Mobile Fare Ticketing/Payment

the tool is read—as a tool to read the validation method. It can be a fixed reader on a bus or a gate or mobile reader or global inspection device and is typically sold by a white label app developer. This is reason for the cloud to be colored blue. Examples, similar to the white label app include Bytemark, Masabi, Moovel, Passport and all those other similar vendors.

The open payment model like the generic one shown in the previous module or chapter comes in two flavors, the open payment using a virtual bankcard or the transit wallet. We saw the IFMS model for open payments in the previous chapter. The bankcard model also includes the identity provider who manages the wallet tokenization process and the application retailer, which supports the open—the app store. Note that the collection and forwarding function hardware is typically owned by the transit agency in this model. In this model, transit is the merchant, the product owner is the customer. There is no integration with other mobility services. No need to set up customer accounts. No ability to differentiate different fare types, although the transit agency can track and analyze transactions by its customers to collect ridership information.

The open payment using a transit wallet is different from the bankcard in that the payment provider is the transit agency or the electronic fare payment system. The mobile app and retailers may be third parties that can provide a top-up retail and fulfillment capability. This model is just emerging and very few agencies have implemented it. Those that have implement the larger electronic fare payment system with physical and virtual cards, back office card, and customer account management, multiple sales channels, and standardized application programming interfaces to support multimodal and event product integration.

This is a centralized model for the SDK. This can also be implemented as a distributed model, which isn't shown in this presentation. This is enabled through the application owners' SDK that is incorporated in other apps. In the distributed model, the customer is required to sign in for each service provider, that is, the ride hailing, bike share, transit, whoever is integrated in with the system. Examples of a centralized system is the Smart Columbus Pivot app and the common payment system, which is the back office, that's in the case studies that we show later on. And it's the centralization uses APIs, though, and not fully developed SDKs. Distributed system, Masabi, and Uber integration in the RTD and transit app example will show how a distributed system may work.

So, how are the open payment and SDK models implemented? Well, they require an open architecture, which means descriptions of functions and interactions among the actors and publishing the interactions or APIs between the actors. Typically, there are hundreds of APIs that need to be implemented and use cases to describe the flow of control between the mobile apps and the back end of the system.

So, one of the main drawbacks of the turnkey system and the variation of how these complex systems can be built is that an agency cannot pry open a closed system to procure pieces and then build on additional pieces at a later time. The IFMS allows that to happen or it shows where it can happen. It does this by not only by clearly defining the roles and responsibilities of the actors but also by describing the functions and interactions between the actors. The IFMS describes these interactions as a use case, which include interactions between actors irrespective off the business model. So, the actors aren't allocated to a specific organization in the reference model, which is the IFMS. And it covers the functionality needed to implement the ecosystem.

## Transit Module 21: Mobile Fare Ticketing/Payment

The IFMS contains ten categories of use cases that are listed above and over 50 different use cases among them. A description of each category is included in the supplement. In the next two slides, two different business models will be shown using a single use case, and you'll see the different actors and how their responsibilities differ based on those business models.

Now use and inspection of products use case is summarized in the outline that's the first model. And it's the service operator checks and collects the data of a customer medium using the public transport service. The actor that triggers the function in the triggered by row, it's a service operator. The actors that interact with the function include the customer service operator, collection forwarding, and product owner. These are listed on the left-hand side in the flow down. The use case description provides information on actions and flow of information. In this use case, fare products are used and inspected and the relevant actor is matched with the conceptual model displayed on the left of the use case cable. The colored boxes correspond to the boxes in the model description, blue for vendor, brown for transit agency and orange for customer. The first part of the flow occurs between the mobile app and the operator, those highlighted by the label A. It includes the detection and verification of the product. The passenger selects the product to use and applies the appropriate security policies. The operator reader detects and applies the rules associated with the medium and product.

The second part of the flow processes the product and verifies the product meets the product and service rules and collects the information. The inspection process is part of this flow and it is described in the lower box label. Finally, the product status and the transaction is collected and reported to the product owner and that follows that flow down. Applying the same use case to a different business model and that is an open payment model using the virtual card; in this case, the customer will see the usage on their bankcard statement. The flow remains the same, but the roles are assigned to different stakeholders.

In the open payment with a virtual card, the product owner is the customer as opposed to the agency, which is shown in the previous model. The transit agency will see product usage through the reconciliation or through the collection and forwarding logs. The use case shows the flow but the architecture shows the variety of organizations that may be implemented. For example, in multi-agency implementation, where the product owner may be a mobility provider such as a bike share service or another transit agency. The importance of knowing the flow helps to open the system for expansion at a later time.

So, this concludes learning objective number 3. And the question we have is what is a software development kit? A, is it a standalone application that can be installed on a workstation; B, a first aid kit for your computer; C, a set of interfaces that could be used to exchange information between two applications; or D, a software library for building applications, interfaces and user interfaces? And the answer is a software library. An SDK provides interfaces and tools including compiler installation tools and functions to build software typically on a specific platform like a mobile device running a specific operating system. It's not a standalone application. That's already built. It's not a first aid kit for your computer. Those are diagnostic tools. And it's not a set of interfaces. Those are typically just APIs. APIs are a subset, though, of the SDK library.

Moving on to learning objective number 4, we have three case studies here. We're going to be looking at the Denver RTD and Masabi integration with Uber and Transit app, the Smart Columbus Common Payment System and the Dallas Area Rapid Transit GoPass.

## Transit Module 21: Mobile Fare Ticketing/Payment

In the first case, the first case study is the RTD fare app, which includes a Deep Link to the RTD trip planner. The app uses the Masabi Just Ride software as a service platform. It's a white label without hardware. RTD uses the fare app for regular bus FlexRide, SkyRide, and train service. The validation method uses a QR code and a Flashpass. RTD achieved a 40 percent savings based on estimated replacement costs for ticket vending machines as customers shift, that is as customers shift from TVM sales channels to the mobile fare app, that's what their estimation is based on.

In this slide, you can see the sales and payment and validation on the app has increased every month since its initiation in late 2018 through December 2019. Mobile sales are rising and at this point, at least by in December of 2019, almost equivalent to the TVM ticket sales.

There are three ways to buy fare products using the mobile payment app or multiple mobile payment apps. Uber app and the transit customer service information app provided by Transit App, that's the one on the right-hand side, they are the three methods. You could see that in the transit app, which is circled, there's a button to select a transit ticket. And in the transit app, the view, the smartphone screen on the right-hand side, you have a branded ticket for RTD over there.

RTD payment app provider brought Uber to the table as a software as a service. The Masabi app API service should be available as a feature they offer. Once deployed, the use of the Uber app to purchase tickets for RTD has steadily risen, as you could see in the graphic. It's gone from so far as of December 2019, over 44,000 tickets were sold with an increase on month by month of over 16 percent.

The next case study is the Smart Columbus, which deployed a Common Payment System or CPS. It's deployed within a larger environment. There are multimodal trip planning application and that was launched in the summer of 2019. It uses Blockchain to validate the chain of transactions. The CPS uses APIs to share information and process payment transactions both as a point of sale and a proof of purchase. Based on account profiles, they support trip reservations for wheelchairs and multiple media including cards, apps, tickets, etc., to pay for tariff and fares as well as discounted fares based on the user profile. Each organization may implement their own branded app provisioned by the APIs from the CPS. Smart Columbus developed the Columbus Ohio Transit Authority (COTA) Pivot app. That's what they call the app to rule them all. That app is used—is still in pilot in beta testing, but you can see it in Apple and Google stores. Should be out within the next year (2021). The CPS APIs integrate with the registering cashbox on the bus, tickets, limo companies, carpooling, micro transit, bike-share services, and car-share services, among others.

Note that with the CPS the financial transaction processing occurs in the back office account-based system. This enables the mobile device to be independent, so the app developer does not need to be a software as its service platform; it just needs an app developer with payment experience. Regional organizations developing account-based systems like the next generation ORCA system or any of the agencies in the L.A. Metro area or HART can extend their mobility marketplace or mobile platform concept using this kind of a model.

The next case study is DART, Dallas Area Rapid Transit. The DART GoPass mobile fare app was one of the first multimodal fare apps developed in the country. Most mobile apps started with transit and events or transit and parking. Most of them integrate features—most of the

## Transit Module 21: Mobile Fare Ticketing/Payment

integrated features are available through an account stored in the back office. The user is asked to register and ordered to use the payment function of the app.

The user account and APIs enable DART to integrate additional series, which they did, as you can see in the timeline here: in 2015, the app provided deep linking to Uber, Lyft, and Zipcar apps. In 2017, the app included the university passes. In 2017, the agency won an FTA grant to integrate microtransit or Go link carpooling and scooter, at that time it was Bird, within the app, and it wasn't deep linking it was actually APIs that they implemented. In 2018, the app added tools to enable fare-capping, cash to mobile, virtual cards, and Apple Pay and Google Pay virtual cards. Last year, 2019, they won the app the Transit Best Mobile Solution for their product, which now includes APIs and services to integrate multimodal mobility services including microtransit, booking and payment, carpooling, shared use mobility conveyances, that is e-scooters and bikes and specialized deep linking to UberPool that identifies origin and destination to book in UberPool service.

The GoPass app serves the Dallas-Fort Worth area three transit agencies: DART, Trinity Transit, and Denton County Transit Authority. It's licensed to the Metropolitan Tulsa Transit Authority and is expected to go live by late summer of 2020. This slide is in your supplement along with additional slides on the GoPass. Note that the digital payments and cash to mobile. This solution addresses the unbanked and underbanked customers which is a significant number of people own smartphones but do not have credit cards or debit cards and so cannot participate in many of the transformational mobility options. The cash to mobile and digital wallet allows GoPass users to load their digital wallets with a value to purchase transit passes, shared micromobility conveyances, transit, microtransit, UberPool, and a growing list of services that are integrated through the API that are contained in the back office. And this is done through retailers in the region in which the app operates.

So, this concludes the case studies. We get to the question and the activities where the question is which of the following is an incorrect statement related to the RTD mobile app? That's the Denver app. The answer choices are A, the RTD Mobile Fare app is implemented as a software as a service; B, RTD manages the interfaces with Uber; C, RTD uses visual verifiable validation for mobile fare products; or D, all fare products offered on RTD mobile fare app are also offered on the Uber app. And the answer to that question is B, RTD, Masabi, not RTD, manages the interfaces settlement and other technical matters associated with integration with Uber. The answer A is incorrect and as well as C and D.

Now to the final learning objective number 5. It presents some of the challenges related to the mobile fare apps. We'll review that in this one. In this section, we discuss four key challenges for implementing mobile fare apps: validation and proof of payment; mobile handset performance; security and personal information; and equity. Each validation method comes with its own challenges. For the V3 method, it does not collect ridership information. The literature shows that many transit agencies collect ridership not only through automated passenger accounting systems, but now through their electronic fare payment systems by tracking the fare transactions. So, in the V3 they don't do that. Additionally, many V3 require online activation and verification for daily changes to the animation.

Some riders who have limited battery life may not be able to download the daily changes if their battery is low or they might not be able to activate their ticket before they go on to a bus. QR code, when used with only a FlashPass, the system does not collect information. Otherwise the system requires hardware which in and of itself is a drawback because that requires

## Transit Module 21: Mobile Fare Ticketing/Payment

maintenance and operation costs. NFC, though this validation method requires the most management, the systems are widespread and used by many other industries. The current drawback is speed to perform the roundtrip transaction including encryption and decryption and tokenization processes. The physical differences between handsets will impact the performance but compared to the current physical card read-write performance, which is about 300 milliseconds, the virtual cards, the bankcards, or transit wallets are slower and over twice the transaction speed.

Let's talk about mobile handset performance. As mentioned in the previous slide, handset performances differ. This happens because of the position length and quality of the antenna as well as the power levels and the interference of other nearby communication devices. The proper position to hold the phone relative to the NFC reader may differ by handset. The signal strength of the handset will impact the speed and distance before NFC signal is detected. You can see sometimes that the smartwatch will implement the transaction faster and further away from the reader than a mobile device may, depending on the power, the number of applications that are active and so on. The NFC standards, particularly the new Tag NFC Data Exchange format protocol, the TNEP, supports more uniformity in the performance of the handsets and should improve this challenge in the near future.

Security and protection of personally identifiable information is a continual issue with any system that holds personal information. PII includes all the information included in the diagram above including social security number, address, phone number, age and more. There are general PII and personalized data associated with the personal media and products and product usage. For example, mobile devices can track user location, location of transactions, payment methods used for purchasing products, and more.

The first step for protecting PII is to know what the PII is collecting. Systems don't only use the current security architectures, that is the secure socket layers, encryption at rest or in transit, hence the reason why the new version of the IFMS standard elevated the importance of the identity provider and the tokenization processes. There are emerging standards such as the GDPR and the CCPA that govern privacy of PII requiring functions such as release consent to share protection that will eventually become standard practices in all deployments if they haven't already.

Finally, there are challenges in deploying mobile apps that are emphasized relative to comprehensive electronic fare payment systems. There are many ways to overcome these obstacles but at a minimum, not all fare apps include provisions to do so. Similar to those identified in Module 12, equity challenges include unbanked. Most mobility providers require an app to include a credit or a debit card. Sometimes gift cards can also be used but those typically charge customers for each use or to top up.

Software as a service products that are not integrated with an agency or regional electronic fare payment system do not typically have third-party retailers where customers can top off their accounts or purchase tickets with cash. Limited English language. Many of the apps come in English or a limited number of other languages. Call center staff with translators or translation tools enable limited English language speakers to contact the app provider. Access to smartphone or cellular communications, there are many issues associated with this challenge. First, as mentioned throughout the module, despite the growth of smartphones, there are still approximately 20 percent of the population without. Additionally, many of the mobile apps

## Transit Module 21: Mobile Fare Ticketing/Payment

require that the application be online to validate a ticket. So, for low-income groups, the cost of cellphone plans are burdensome.

Finally, many populations like older adults and people with disabilities have difficulty using some of the functions on the phone app. Voice commands help these functions. At the Pinellas Suncoast Transit Authority, PSTA, these types of challenges are addressed in the Direct Connect program as you can see in the graphic on the right-hand side. First, the Direct Connect program provides call in options. They also have cash options and outreach to riders to help train them in the use of their mobile app. The PSTA Late Shift program, which you can see, is integrated into the Uber app, was set up to support low-income riders requesting Uber when PSTA was not in service or when service was infrequent. They received a discount of 50 percent of the cost. So, they served as a merchant for the Transit Agency.

So, that's the end of the Module 5. And the final question we have is what media type presents a challenge to collect ridership information? Is it NFC, A; Flash pass, B; QR code, C; or open payment, D? And the answer is Flash pass. A Flash pass typically is not validated by a card reader which stores and records time and location. NFC, it's not NFC which validates—is validated by a card reader which records the time and location. It's not the QR code, which is validated by a card reader, which records time and location, although if it's used as a Flash pass then it doesn't. And it's not open payment, which also validates—is validated by a card reader which records the time and location.

So, in conclusion, I want to thank you for participating. To summarize the course, in this course we covered the mobile fare payment standards used by many of the same standards—that use many of the same standards as card and account-based systems. New payment technologies and standards are emerging that support mobile fare payment. Although fare payment business models differ, open APIs and SDKs enable integration with multiple agencies, mobility services, and providers. Mobile payment methods may be used for fare and ticketing, are becoming more prevalent even as new challenges arise. And in deploying mobile fare apps, the important lesson is to adopt technologies that meet your business goals and data needs.

I want to thank you for listening and participating. We have other courses that were mentioned in this module, Module 10, Electronic Fare Payment System. Module 12, Electronic Fare Payment Advanced Payment Systems, which includes open payment acceptance. This is the third course in this curriculum.

And I want to again thank you for completing this module. Please send us your feedback. At the end of this module, there is a link that you could press to provide us with your thoughts and comments about the value of this training. Thank you very much.