IDEA

Innovations Deserving
Exploratory Analysis Programs

Transit IDEA Program

Citopia MaaS/Multimodal Transit

Final Report for
Transit IDEA Project 101

Prepared by:
Tram Vo
Citopia

April 2023

NATIONAL ACADEMIES Sciences Engineering Medicine

TRB TRANSPORTATION RESEARCH BOARD

Innovations Deserving Exploratory Analysis (IDEA) Programs Managed by the Transportation Research Board

This IDEA project was funded by the Transit IDEA Program.

The TRB currently manages the following three IDEA programs:

- The NCHRP IDEA Program, which focuses on advances in the design, construction, and maintenance of highway systems, is funded by American Association of State Highway and Transportation Officials (AASHTO) as part of the National Cooperative Highway Research Program (NCHRP).
- The Rail Safety IDEA Program currently focuses on innovative approaches for improving railroad safety or performance. The program is currently funded by the Federal Railroad Administration (FRA). The program was previously jointly funded by the Federal Motor Carrier Safety Administration (FMCSA) and the FRA.
- The Transit IDEA Program, which supports development and testing of innovative concepts and methods for advancing transit practice, is funded by the Federal Transit Administration (FTA) as part of the Transit Cooperative Research Program (TCRP).

Management of the three IDEA programs is coordinated to promote the development and testing of innovative concepts, methods, and technologies.

For information on the IDEA programs, check the IDEA website (www.trb.org/idea). For questions, contact the IDEA programs office by telephone at (202) 334-3310.

IDEA Programs
Transportation Research Board
500 Fifth Street, NW
Washington, DC 20001

The project that is the subject of this contractor-authored report was a part of the Innovations Deserving Exploratory Analysis (IDEA) Programs, which are managed by the Transportation Research Board (TRB) with the approval of the National Academies of Sciences, Engineering, and Medicine. The members of the oversight committee that monitored the project and reviewed the report were chosen for their special competencies and with regard for appropriate balance. The views expressed in this report are those of the contractor who conducted the investigation documented in this report and do not necessarily reflect those of the Transportation Research Board; the National Academies of Sciences, Engineering, and Medicine; or the sponsors of the IDEA Programs.

The Transportation Research Board; the National Academies of Sciences, Engineering, and Medicine; and the organizations that sponsor the IDEA Programs do not endorse products or manufacturers. Trade or manufacturers' names appear herein solely because they are considered essential to the object of the investigation.

TRANSIT IDEA PROGRAM COMMITTEE

CHAIR
SANTOSH MISHRA
IBI Group

MEMBERS
MELVIN CLARK
HATCH LTK
LAUREN COCHRON SCOVILLE
Proterra Inc.
DREW DARA-ABRAMS
Interline Technologies LLC
SUZIE EDRINGTON
Capital Metropolitan Transit Authority
AL MARTINEZ
Los Angeles County Metropolitan Transportation Authority
SANTOSH MISHRA
IBI Group
LOUIS SANDERS
Ayers Electronic Systems
MANJIT SOOCH
Alameda-Contra Costa Transit District
JASON STARR
Via Transportation
DAVID SPRINGSTEAD
Metropolitan Atlanta Rapid Transportation Authority
DAVID THURSTON *Canadian Pacific Railway*

FTA LIAISON
RIK OPSTELTEN
Federal Transit Administration

TRB LIAISON
STEPHEN ANDRLE
Transportation Research Board

IDEA PROGRAMS STAFF
CHRISTOPHER HEDGES, *Director, Cooperative Research Programs*
GWEN CHISHOLM-SMITH, *Manager, TCRP*
INAM JAWED, *Senior Program Officer*
MIREYA KUSKIE, *Senior Program Assistant*

EXPERT ADVISORY PANEL TRANSIT IDEA PROJECT 101
Suzie Edrington, *Austin Capital Metro*
Julie Fernandez, *Houston Metro*
Sarah Hackett, *Oregon Department of Transportation*
Chris Tomlison, *Deolotte Consulting, LLP*
Santosh Mishra, *IBI Group*
Lex Powers, *SEPTA*
P.J. Wilkins, *EZ Pass*
Bhaskar Krishnamachari, *University of Southern California*

Citopia MaaS/Multimodal

Transit IDEA Program Final Report

IDEA Project T-101

Prepared for

The Transit IDEA Program
Transportation Research Board
National Academies of Sciences, Engineering, and Medicine

by

Tram Vo

Citopia

12 April 2023

Table of Contents

Acknowledgement

Citopia MaaS/Multimodal Transit - Transit IDEA Project T101 was supported by the Transit IDEA Program of the Transportation Research Board (TRB) as part of the National Academies of Sciences, Engineering, and Medicine.

We would also like to thank the following expert advisory panel members:

- Suzie Edrington, *Director, Demand Response,* Austin Capital Metropolitan Transit Authority
- Julie Fernandez, *Lead Financial, Management, & Budget Analyst, Revenue/Fare Policy,* Houston METRO
- Sarah Hackett, *Education Staff Retention Grants Administrator,* Oregon Department of Education formerly with the Oregon Department of Transportation, Public Transportation Division
- Bhaskar Krishnamachari, *Director*, USC Viterbi Center for Cyber-Physical Systems & IoT
- Santosh Mishra, *Assoc Dir./Practice Lead, Mobility Technologies,* IBI Group
- Lex Powers, *Strategic Planning Manager,* SEPTA
- Chris Tomlison, *Managing Director, Technology Operations, Deloitte Consulting, LLP*
- PJ Wilkins, *Executive Director,* EZ Pass

Glossary

Appendix H contains the Detailed g+Glossary of Terms with the definitions of terms used throughout the report.

API	Application Programming Interface
BART	Bay Area Rapid Transit
CPU	Central Processing Unit
DID	Decentralized Identifier
DLT	Distributed Ledger Technology
EKS	Elastic Kubernetes Service
EDV	Encrypted Data Vault
GPS	Global Positioning System
GBFS	General Bikeshare Feed Specification
GDPR	General Data Protection Regulation
GTFS	General Transit Feed Specification
GTFS RT	General Transit Feed Specification Real Time
IoT	Internet of Things
ITN	Integrated Trust Network
MaaS	Mobility-as-a-Service
MOBI	Mobility Open Blockchain Initiative
MTT	MOBI Trusted Trip
MOD	Mobility on Demand
MUNI	San Francisco Municipal Railway
OTP	Open Trip Planner
VC	Verifiable Credential
VP	Verifiable Presentation
SSDT	Self-Sovereign Digital Twin
SDK	Software Development Kit
TNC	Transit Network Companies
TRB	Transit Research Board
RAM	Random Access Memory
RUC	Road Usage Charge
UAT	User Acceptance Testing
UI	User Interface
PII	Personally Identifiable Information
SEPTA	Oregon Department of Education, Southeastern Pennsylvania Transportation Authority
W3C	World Wide Web Consortium
ZK	Zero-Knowledge
ZKP	Zero-Knowledge Proof

Executive Summary

It is estimated that currently 83% of the U.S. population lives in urban cities. This figure is projected to increase to 89% by 2050. Meanwhile, 68% of the world population is expected to live in urban cities by 2050 (*1*). With increasing populations, cities in the U.S. and around the world are looking for solutions to manage existing infrastructure demands as well as congestion and air pollution. In light of this, many governments and enterprises have begun to re-think mobility and explore climate change mitigation strategies. Mobility/transportation demand of riders is changing with the emergence of usage-based business models. Riders want a seamless, efficient, integrated multimodal transit and Mobility-as-a-Service (MaaS) experience. Increasing the utilization of usage-based mobility business models such as public transit, ride hailing, e-bikes, car renting/sharing, Road Usage Charge (RUC), usage-based parking, etc. is a promising approach to tackling mobility demands and low-carbon transportation challenges. Most mobility/transportation services are offered by siloed centralized platforms today. They lack the interoperability needed to connect disparate mobility services due to the high cost of one-off integrations between service providers. In addition, these platforms fail to offer the data privacy and control of personal data that users demand.

Centralized platforms are also strained by the lack of fair competition and cooperation. Often, market players can't offer their services in the same marketplace or platform while being able to cooperate and compete fairly. In a survey study carried out by the BearingPoint Institute, mobility leaders pointed out the lack of privacy and standardization as well as the lack of cooperation among stakeholders as some of the factors that hold back their organizations from integrated multimodal transport (*2*). To create an ecosystem for integrated multi-modal mobility services and data sharing, stakeholders need to collaborate.

As data breach and aggregation scandals often reveal, many major companies and centralized platforms fall short of meeting data privacy claims (*3*). These incidents, along with the emergence of Web3 technologies, have resulted in an increasing demand for higher data privacy standards and transparency on Personally Identifiable Information (PII) handling. This is why data privacy protection for riders and service providers alike must be an essential component in MaaS/multi-modal transit applications.

As a solution, MOBI created Citopia, a federated Web3 marketplace and tokenized ecosystem where mobility stakeholders can securely transact and share data with assured privacy and interoperability. With Citopia MaaS, riders can plan and book multimodal trips with seamless itinerary, ticketing, and payment. Citopia uses the blockchain-based Integrated Trust Network (ITN) as the trust anchor for identity management. Both Citopia and the ITN are member-owned and operated. Citopia MaaS users can select their trip preferences to choose from different mobility providers/transit agencies and travel routes that best suit their needs. Citopia leverages Zero-Knowledge Proofs (ZKP) to allow users and providers to retain full control over who sees their data and how that data is used. This seamless and secure connected mobility experience is enabled mainly by Self-Sovereign Digital Twins (SSDTs), World Wide Web Consortium (W3C) Verifiable Credentials (VCs), and ZKPs. Users and providers can transact using their SSDTs by issuing VCs with embedded Decentralized Identifiers (DIDs) for verification (*4*). SSDTs are digital twins that can autonomously participate in transactions using self-sovereign identity technology (*5)*. Personal or organizational data is always stored in the respective SSDT on the user's or service provider's device or server and cannot be shared without permission. W3C-compliant DIDs allow the SSDTs of riders, transit agencies, vehicles, mobility service providers, and other entities in the ecosystem to communicate and transact with each other in an interoperable ecosystem. The goal of this Transit IDEA project (T101) is to create a product-market fit for real-world implementation and identify areas for future improvement.

At the end of the first stage of the project, the complete and fully functional demo of the Citopia MaaS mobile app and web interface was successfully presented to the expert advisory panel in December 2022 along with ZKP generation. The project team successfully demonstrated the creation of Citopia SSDTs; multimodal trip planning using real-time General Transit Feed Specification (GTFS) and General Bikeshare Feed Specification (GBFS) data involving bus, metro, and bike/scooter-as-as-a-service MaaS options; and trip booking, navigation, and payments on Citopia MaaS.

In the second and final stage of the project, performance testing for Citopia MaaS was carried out to evaluate the system suitability for operating under real-world conditions and document the findings in the final project report. This effort involved determination of the testing approach, scope, KPIs, setup for the system components and execution of the performance testing. The main components to be tested were the onboarding of riders and rider trip execution, which involved the querying, booking/reserving, trip execution, and invoicing of a multimodal trip. The rider onboarding flow involved the creation of DIDs (through ITN) and SSDTs. The trip execution involved different types of VCs that are used to simulate the transactions between service providers and riders for searching, requesting/booking, and paying for different modes of transit.

The project advisory panel is made up of members of the following organizations: Deloitte, Austin Capital Metropolitan Transit Authority, EZ Pass, Houston METRO, IBI Group, Oregon Department of Education, Southeastern Pennsylvania Transportation Authority (SEPTA), and University of Southern California Viterbi Center for Cyber-Physical Systems & IoT.

IDEA PRODUCT

Citopia MaaS offers a secure, easy-to-use federated Web3 marketplace where riders can plan or book multimodal trips from different service providers with seamless itinerary, ticketing, and payments. Riders can select their trip preferences from fastest, cheapest, least number of transfers, and greenest routes; and choose their preferred mode(s) of transit such as bus, bike-as-a-service, and more. Citopia MaaS enables riders to access MaaS services and manage trips all in one place. This provides both public and private service providers with an opportunity to expand their ridership/customer base and increase their net revenue. Service providers can include but are not limited to, transit agencies, ridesharing companies, tolling authorities, ride-hailing companies, usage-based insurance companies, parking operators, and micro-mobility providers.

Citopia MaaS leverages Web3 technologies in a way that redefines data privacy for riders and service providers. Riders have full ownership and control of their PII. To understand how Citopia MaaS maintains data privacy, it is important to highlight how PII can be exposed. Firstly, PII can be exposed if it is stored on a platform/network. Therefore, only the minimum set of necessary PII should be stored and only when necessary. PII is also vulnerable to being exposed if the data controller or processor has not adopted proper pseudonymization or better anonymization techniques. Lastly, PII can be extracted if the correlability between data elements makes the data subject identifiable, meaning that data elements that can be related through statistical or other means to allow for the identification of the data subject. For example, the combination of name and address or unique identifiers tied to name, address, or phone number can be used to obtain personal data. As a data processor, Citopia processes different types of data (explained in more detail in the Investigation section) to enable seamless MaaS and uses W3C-compliant DIDs and VCs, along with ZKPs and other pseudonymization techniques, to ensure that PII is not exposed. Similarly, service providers' organizational data is maintained and is only accessible to them (*6*). All PII or organizational data is kept in rider or service provider SSDTs, which only the owner can access. It should be noted that while Citopia offers SSDTs as a service, it has no access to the information stored in the SSDTs.

In addition to data privacy, DIDs, SSDTs, and VCs enable business automation for service providers, which can reduce the cost of trust (e.g. costs to verify claims, information, identities). This type of multiparty business automation can translate into significant cost savings as providers can work with each other without relying on a centralized authority/platform to verify identities of entities and assets in a value chain and verify claims or assertions made by entities for a transaction to take place. A DID is linked to a Citopia SSDT through the ITN, which is the trusted identity services network that provides the DID registry. It should be noted that DIDs do not contain/store any PII or organizational data. An SSDT is a digital twin whose owner and/or controller has the ability to participate as an autonomous economic agent in trusted Web3 transactions through self-sovereign identity technology. Meanwhile, VCs enable parties in Citopia to present certain information that is required for transactions in a secure and verifiable way. A VC essentially acts like a unique stamp that serves to attest to specific information about an entity. The verifier no longer has to contact the issuer to confirm the credential since the claim is verifiable (*7*). It is also secure and authentic as the data is only seen by the intended recipient (the verifier). This can save businesses (especially administratively-heavy ones, such as insurance carriers) substantial sums of money in data verification costs (*8*). Additionally, VCs can be coupled with ZKPs to enable greater data privacy protection on Citopia. ZKP is a cryptographic method that enables a prover of a statement to prove to a verifier that a statement is true, without providing any other information than whether the statement is true (not false) (*9*). This way the ZKP is made a part of the claim of the VC that the issuer attests to.

Citopia's federated Web3 marketplace is designed for scalability and to provide service providers flexibility in choosing technology vendors. SSDTs and VCs are interoperable technologies that can work with W3C standards-based networks/platforms. In addition to providing SSDT as-a-service, Citopia MaaS will also allow riders and service providers to bring their own compliant SSDTs onto Citopia. By the nature of their architecture, VCs are interoperable across many W3C-compliant systems and can be used in many use cases. This interoperability will enable service providers on Citopia MaaS to communicate and transact with other systems/platforms, facilitating the adoption of Citopia MaaS and increasing the scalability.

Citopia MaaS also enables service providers to leverage analytics of the aggregate of anonymized trip data. Citopia processes this data for use by service providers. Service providers can substantially benefit from aggregated anonymized trip data, as it can be used to improve connectivity over various networks, getting more insights into transportation trends, forecasting, and planning. Service providers will be responsible for providing data retention policies to Citopia regarding gathering aggregated data. Aggregated trip data may contain PII and Citopia prevents this PII from being exposed. PII is extracted in multiple steps: each trip leg (mode of transit) is processed by a different Citopia node, then user DIDs are removed from anonymized trip data and ZKPs are used to ensure data privacy, integrity, and verifiability. This way neither the service provider nor Citopia can identify PII from the anonymized trip data.

Citopia MaaS encourages the generation of new pay-as-you-go / usage-based mobility business models. Through Citopia MaaS' federated Web3 marketplace, service providers can create new business models, services, and products for seamless MaaS while reaping the benefits of business automation and multiparty coordination. This can increase existing revenue and create new revenue streams for service providers depending on the service/product.

Citopia MaaS allows sustainable mobility and congestion reduction incentives and other (service provider specific) mobility/transportation rewards to be built in. This is especially useful for cities, policymakers, transit agencies, and various service providers, as it gives them tools to implement incentive programs to effectively promote public transit, strive toward achieving carbon emissions reduction goals, manage demand for mobility services and infrastructure and reduce congestion.

Concept and Innovation

Citopia MaaS is the first application that enables riders to seamlessly plan, book, and pay for trips in a privacy-preserving manner via Web3 technologies. As a federated marketplace, Citopia MaaS provides a more robust and secure medium for entities to transact through its network of nodes and trusted identity management.

The emergence of Web3 technologies presents new avenues for protecting PII data in compliance with privacy-centric regulations appearing in Europe, California, and other regions or countries. As data privacy protection is becoming ever more important through these regulations, the combination of DIDs, VCs and ZKPs present a potential solution to addressing the issue. Citopia MaaS represents a step toward developing systems that emphasize privacy preservation and challenge the siloed approach favored by centralized platforms today by using these Web3 technologies (explained in the previous section). Any PII belonging to users and any organizational data of service providers are maintained in respective SSDTs, that are only accessible by respective owners. It should be noted that while Citopia offers SSDTs as a service, it has no access to the information stored in the SSDTs.

Citopia MaaS has successfully demonstrated mapping of Citopia technical capabilities and W3C Universal Wallet capabilities for SSDTs to enable business automation on Citopia MaaS and interoperability across stakeholders' systems/platforms using W3C DID standards (different service providers and riders can communicate and transact using W3C compliant DIDs). Because Citopia MaaS also focuses on interoperability and seamless travel between siloed services, transit users will significantly benefit from not having to use multiple identities, mobile/app applications, and payment channels. Users will be able to communicate and transact with multiple MaaS providers all from a single marketplace. Citopia MaaS can play a significant role in enabling cities and metropolitan regions to implement seamless transportation and demand management strategies such as tracking real-time data trends.

Citopia MaaS is the first implementation of SSDTs for all ecosystem participants such as riders, providers (e.g. transit agencies, insurers, tolling agencies, RUC administrators), and vehicles. The SSDT acts as a universal translator and an encrypted locked data vault that enables the owner of the SSDT to communicate with virtual agents of other entities in Web3 transactions. Data is stored in SSDTs by using DIDs, VCs and ZKPs to ensure privacy and security. An important differentiator of the SSDT is that the owner has full control of their SSDT and can use it outside of Citopia unlike centralized digital twins, which rely on the issuing platform or organization to operate. This enables service providers on Citopia MaaS to easily access multiple customer segments (riders) on one marketplace without vendor or technology lock-in. Citopia enables the network effect with SSDTs by connecting riders directly to the service providers for usage-based business automation, which facilitates more riders to access seamless transit. In return, this can potentially increase the value of MaaS/transit services.

Service providers can also reduce the reliance on middlemen and potentially cut down on trust costs through usage-based business automation with Citopia MaaS. Seamless transactions and a singular payment gateway eliminate the inefficiencies of using multiple payment methods. In some MaaS services, these cost reductions can be passed on to the consumers in the form of lower transaction costs for the end user.

Blockchain technology is not yet mature enough for many enterprise applications and sensitive data should not be stored on chains. The Citopia MaaS pilot demonstrated a novel enterprise use case of blockchain by using the blockchain-based ITN to register the DIDs of participants in a secure, decentralized, and privacy-preserving manner.

It should be noted that there are other multi-modal transit applications using OTP as an open source software to build trip planning capabilities as OTP provides passenger information and transportation network analysis services via rest APIs. Citopia MaaS uniquely combines these trip planning capabilities that are established with the help of OTP with self-sovereign identity, automation, and data privacy through DIDs, VCs, ZKPs, and SSDTs.

To our knowledge this is the first project to write core libraries in the UI software framework React-Native to enable functionalities of MaaS/Multimodal transit applications. In a larger context, this will enable portability of applications across platforms (such as iOS and Android apps, servers, and standard browser implementations) without having to do adjustments, which is a significant competitive advantage.

INVESTIGATION

Work Accomplished Stage I & Stage II
Table 1 shown below describes the tasks carried out throughout the project and progress at the time of completion. Please refer to Appendix H for the Detailed Glossary of Terms.

TABLE 1 Citopia MaaS/Multimodal Transit Project Progress

TASKS		% COMPLETED
	Stage 1	
Task 1	Organize a Kickoff Meeting with Expert Advisory Panel	100%
Task 2	Design and Develop Citopia MaaS/Multimodal Marketplace	100%
Added	*ITN SDK Packages - Added React Native Support*	100%
Task 3	Integrate Citopia and ITN (formerly mobiNET) Core Services	100%
Task 4	Verify Integration Functionality	100%
Task 5	User Acceptance Testing and Simulated Data Definition	100%
Task 6	Complete Demo	100%
Task 7	Submit a Stage I Report and Project Progress Review	100%
	Stage 2	
Task 8	Test (Performance Testing) and Evaluate the System	100%
Task 9	Prepare and Submit a Final Report	100%

TASK 1 — ORGANIZE A KICKOFF MEETING WITH THE EXPERT ADVISORY PANEL

Citopia MaaS/Multimodal Transit IDEA Project–T101 was launched on January 19, 2022, via the kickoff meeting with the expert advisory panel. The entirety of the project plan, including the proposed timeline, objectives, scope, deliverables, and Key Performance Indicators (KPIs) were discussed and reviewed. The panel and the project team decided to meet monthly to discuss project updates and deliverables.

TASK 2 — DESIGN AND DEVELOP CITOPIA MAAS/MULTIMODAL MARKETPLACE

In the first and second quarters, the project mainly focused on Task 2 - Design and Develop Citopia MaaS/Multimodal Marketplace while other tasks were also carried out in parallel.

The project team began work on Task 2 by defining the user journeys along with the relevant done criteria and assumptions. These are required to define user stories for each user journey each with its specific assumptions and done criteria. The project team held a workshop with volunteers from the expert advisory panel in January to guide the user journey design and identify data needs for the demo. Then the user stories, including the acceptance criteria for the successful execution of system functionalities, were defined. The user stories were categorized into three main sections: Rider and Service Provider Onboarding, Trip Planning and Navigation, and Payments.

In the Trip Planning section, two types of generic flows were defined based on the mode of transit/MaaS option: prepay MaaS and post-pay MaaS. Any MaaS option falls under either the post-pay or the prepay flow. These flows define the acceptance criteria regarding the creation of the necessary VCs to execute a particular segment of a single mode and/or multimodal trips on Citopia MaaS/Multimodal. After user stories were completed, the required schemas to build the demo application with the functionalities defined in the user stories were identified and developed. This was followed by drafting the product backlog to organize and prioritize development work. To design the required modules for service provider onboarding, the project team investigated transit data feeds through the GTFS RT APIs (Application Programming Interface), GBFS APIs, and other open API sources. After the schemas were finalized, the project team started designing the mobile app and the web interface screens for the demo with required functionalities and features based on the user stories and schemas. The mobile app screen designs (see Figure 2 for examples) were broken down into five main categories that were designed in batches and sub-batches: rider onboarding, account management, main screen, trip planning, and navigation. The web interface screens see Figure 1 for examples) were categorized into three main categories and were also broken down further into batches and sub-batches: service provider onboarding, asset onboarding, and account management screens. Starting with the onboarding screens, once a batch of screen designs was completed, the development team started building the user interface (UI) of the demo dApp by coding the frontend. Simultaneously, the team started coding the backend for rider and service provider onboarding including the creation of SSDTs. While rider SSDTs and digital wallets were successfully deployed in the first quarter, service provider SSDTs were successfully deployed and the associated digital wallets were set up in the second quarter.

In the second quarter, mobile app screen designs were completed and presented to the expert advisory panel. Approx. 150 screens were designed for the mobile app. Similarly, all screen designs for the web interface, which include service provider onboarding and account management, were completed. With the release of a testable demo of the mobile app, the team also worked on resolving UI errors and improvements. The UI was continuously reviewed and improved as new sets of screens with backend logic were added to the mobile app. In addition, various functionalities were added and edited throughout the development process.

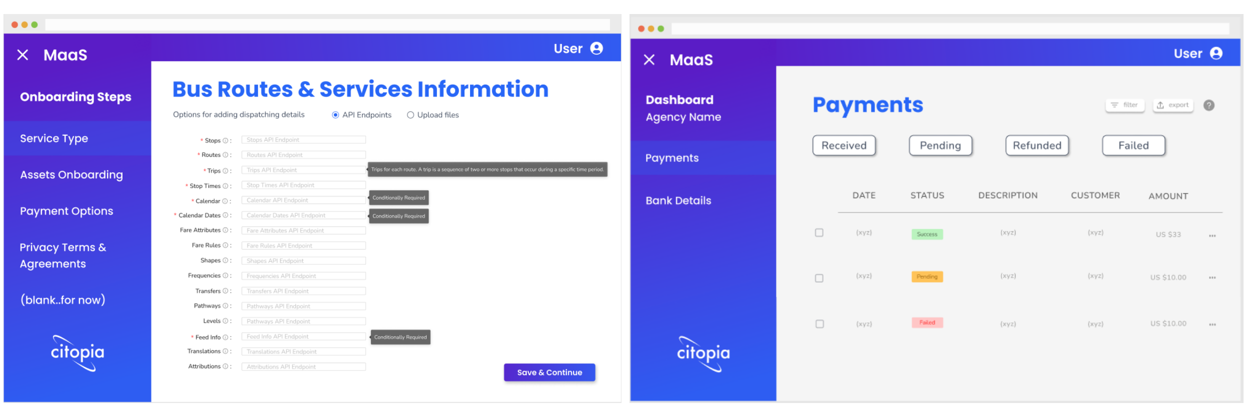


FIGURE 1 Citopia MaaS web interface.

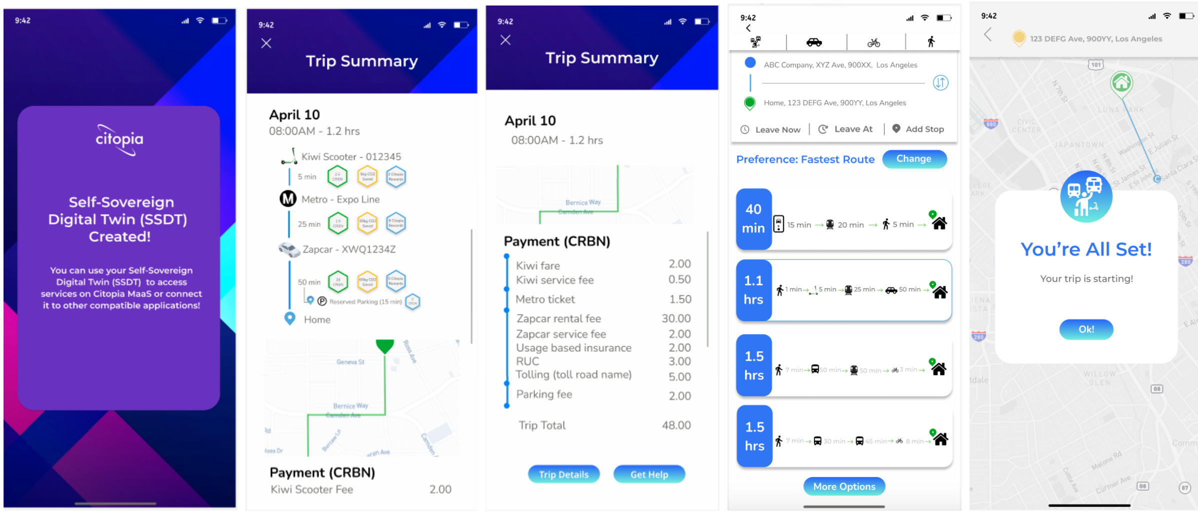


FIGURE 2 Screens from Citopia MaaS mobile app UI.

After building the backend for onboarding, the team worked on developing the backend for the service provider asset onboarding process. This process enables the service providers to onboard their mobility assets to Citopia MaaS through which they offer MaaS to riders such as bus, metro, bike-as-a-service, and more. This way, service providers can upload mobility asset data such as what bikes or ride-hailing options are available in a particular geographic region at a given time or which buses are running on a particular route.

Since service provider asset onboarding involves the integration of service provider APIs, it is one of the core components of the backend capabilities that enables the key mobile app functionalities. Once the initial version of the setup was developed, the asset onboarding process was continuously refined and improved through iterations throughout Stage I. During the second quarter, the setup of the asset onboarding process faced various challenges. This was due to the lack of trip data available prior to trip booking, which precluded the data to be provided in the service provider onboarding step. The project team successfully solved this by creating asset DIDs at runtime using the GTFS-RT feed when a rider was booking a trip. These asset DIDs are managed by the service providers that own the assets. There were multiple iterations of building this data pipeline. Another major task that the development team focused on in the second quarter was addressing the storage and usage issues of the GTFS data from multiple service providers (due to the multiplicity of both real-time and static GTFS Data from different service providers). This is essential to enable trip planning functions in Citopia MaaS including setting up the Open Trip Planner (OTP) server, and loading and processing of GTFS data. The project team utilized a solution that involved designing and building a streaming data pipeline to manage the GTFS data. This would result in querying all service providers' GTFS data and return routes and transit options by deploying the OTP Server Instance when the rider queries routes from the app. Please see the high-level design architecture as shown in the figure below. Both efforts resulted in a delay in the completion of the demo due to unexpected task complexities. After these tasks were successfully completed, the project focused on enabling multimodal route generation and multimodal trip navigation using GTFS and GBFS data, while work on React Native development in Task 3 continued.

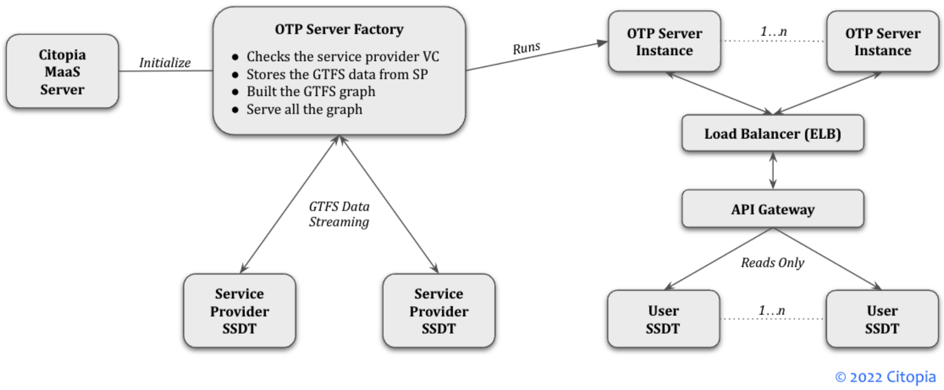


FIGURE 3 High-Level design for the streaming data pipeline.

In May 2022, the project team released the first demo version of the mobile app and web interface and presented it to the expert advisory panel. The initial versions only had basic backend logic. Newer versions of the demo app, which were more complete (however, not fully functional until Task 2 was completed), were presented in each expert advisory panel meeting to discuss demo progress and allow the team to make necessary iterations based on the panel's feedback. One of the significant capabilities demonstrated to the expert advisory panel with the early versions of the demo was the successful creation of Citopia membership VCs during account creation, and agreement VCs and MOBI Trusted Trip (MTT) VCs during multimodal trip navigation. Citopia membership VCs are used along with DIDs to verify that a rider is a valid Citopia MaaS user. Agreement VCs are issued to approve and verify a rider's access to MaaS services that a service provider offers (such as booking, reserving, or direct usage). MTT VCs are issued to verify the beginning and the end of a particular mode of transit.

In the second quarter, the team mostly worked on building the static demo (explained in detail in Task 6). The static demo is intended to show the user journey, which includes three modes of transit, usage-based parking, usage-based insurance, Road Usage Charge (RUC), and tolling. After the backend challenges in the second quarter were overcome, the development team started working on the dynamic demo, which involves route generation with bike-as-a-service, scooter-as-a-service, bus, and metro in the San Francisco region as modes of transit. The dynamic demo is testable in real-time through integration with the GTFS and GBFS APIs from certain service providers. The dynamic demo, along with the list of service providers, is described in detail in Task 6. Since live data cannot be integrated into the static demo for reasons explained in Task 6, it is not sufficient to demonstrate the key multimodal route generation capabilities of Citopia MaaS. The dynamic demo, on the other hand, shows all the route generation capabilities of Citopia MaaS. Thus, the team put the static demo on hold and started working on the dynamic demo.

Initially, the route generation capabilities in the multimodal trip planning were working as expected for bus and metro with the GTFS APIs that were integrated. However, as more service providers were onboarded and the GBFS data were integrated to add bike-as-a-service and scooter-as-a-service options in the multimodal routes, the team faced various challenges using OTP 2.1. Firstly, GTFS data expired for various service providers used in the San Francisco region. One of the bottlenecks here was that we do not have control over when a service provider releases a new data feed. This was successfully resolved by rebuilding the OTP 2.1 instances with the latest version of GTFS data from service providers. We then discovered that we would have to manually monitor and ensure the data feed is not expired otherwise route generation will not work as expected. Another error stemmed from the processing of SamTrans GTFS data, which resulted in build failures. To fix this, the data source was changed. The new data source for SamTrans worked smoothly. Lastly, integrating GBFS resulted in configuration problems due to inherent issues of GBFS data. While researching to develop a solution to this problem, we saw that other projects that use GBFS data have faced similar issues. The project team successfully tackled this by troubleshooting with different configurations to re-enable the GBFS data. Due to the ongoing work in addressing these challenges with route generation, the completion of the demo was further delayed by another month. Recently, the project team migrated from OTP 2.1 to OTP 2.2 once the newer version was made available. The reason for the migration was that OTP 2.2 had improved processing of GBFS data and its API responses, which made it possible to distinguish between bike-as-a-service and scooter-as-a-service options. In the earlier version, they were both categorized as bikes.

In conjunction with this effort, the project team also worked on building and improving various map functions and the trip booking flow. These include: search functions, map display functionalities; enabling vehicle, bike (not bike-as-a-service), walk routing options, and dynamic leave-now and leave-at parameters as well as enabling riders to edit MaaS preferences before choosing a route. MaaS preferences for the riders in Citopia MaaS consist of fastest, cheapest, greenest, and least number of transfers. Currently, the route generation algorithm only prioritizes the fastest route options in route generation results. This is due to using OTP. (The configuration using OTP currently does not enable the other preferences to be offered to the rider). Generating the algorithm to prioritize the other preference options in route search will require significantly more development efforts and resources. Hence, this is planned for future development after the project (Stage I and Stage II) is complete. After resolving the major errors, the project team continuously reviewed the route generation and trip booking capabilities and made iterations to improve the overall trip planning capabilities of Citopia MaaS. Additionally, the frontend development for the mobile app screens was completed at this point.

After ensuring that route generation worked successfully, the project team focused on trip execution and navigation. The VCs that were enabled in the static demo were also enabled in the dynamic demo trip execution. Meanwhile, the necessary backend for the service provider web interface and the mobile app to enable payments for MaaS/transit options and the generation of invoice VCs were built. Invoice VCs are issued to verify the payment for a mode of transit and for Citopia to maintain an accounting ledger. Then the necessary backend capabilities to enable bulk payments to the service providers was built.

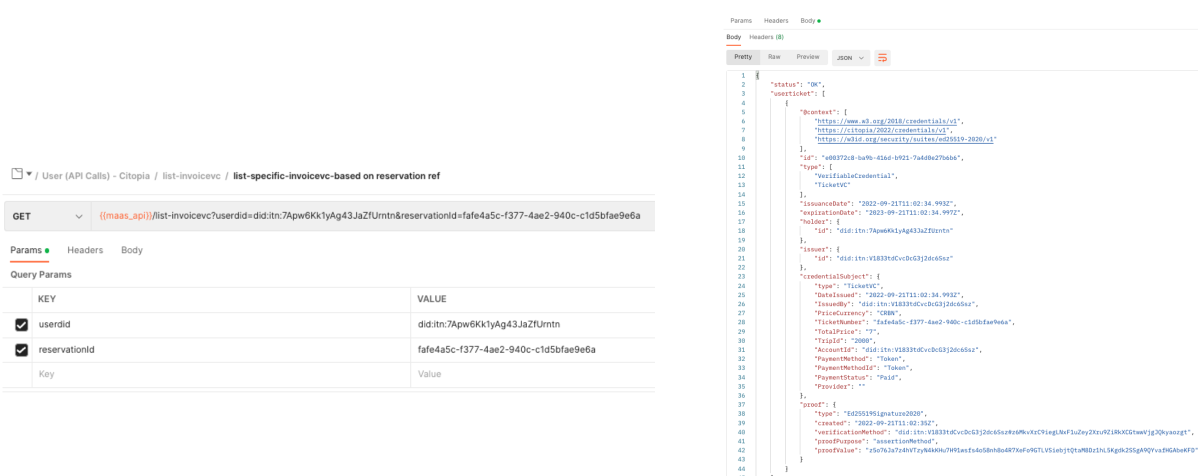


FIGURE 4 An Invoice VC API call request (left) and corresponding API call response (right).

Initially, the navigation SDKs were not written in React Native because the development framework was under development at the time the navigation SDK was created. After deploying the updated navigation SDKs in a React Native implementation, the project encountered navigation issues with Mapbox/Mapquest, which was used for enabling map services on Citopia. This was due to the difficulty of finding stable React Native packages that enable the navigation SDK of Mapbox/Mapquest for easier integration on the iOS and Android codebase. It is important to note that React Native is a relatively new UI software development framework. Several versions of the navigation SDK were tested. Initially, we started integrating with Mapquest Android Navigation SDK and faced issues with the Gradle version as it could not be properly supported. We tried integrating with Mapquest iOS Navigation SDK and found out that it only supports up to iOS 11.0. Mapquest had removed the Navigation SDK support from their official documentation as well. Finally, we tested the Mapbox iOS Navigation SDK and it was chosen as it yielded the best results for smooth integration. The mobile app was being built simultaneously on both iOS and Android up to this point in the development timeline.. However, due to the Mapbox/Mapquest navigation issues, the development on Android paused and the project team decided to build the rest of the demo only on iOS as it would require significantly more time and resources to build the Android navigation SDKs. Thus, the mobile app demo will be presented only on iOS. We aim to continue development on Android for the future phases when the project is complete.

The project team initially started building navigation with turn-by-turn navigation. This involved a time-intensive effort to write the custom wrapper functions over the Mapbox iOS Navigation SDK. It also caused issues while passing the custom routes using the Mapbox iOS Navigation SDK and it was not possible to modify the navigation map UI according to the project's screen designs. As a result, the project team decided to implement point-to-point navigation, which worked successfully.

In parallel with the other sub-tasks, the project team also worked on deploying ZKPs to maintain data privacy in the generation of aggregated trip data. Citopia processes the aggregate of anonymized trip data for the use of service providers. For instance, aggregated anonymized trip data can be used to improve connectivity over various networks. Service providers will be responsible for providing data retention policies to Citopia regarding gathering aggregated data. Trip data may contain Personally Identifiable Information (PII) as it contains multiple legs with various transport modes (each leg consists of network_reference, mobility_mode, lat_start, lon_start, lat_end, lon_end, and start_time). Citopia prevents trip PII from being exposed by extracting it in multiple steps. Each trip leg (mode of transit) of a given trip is processed by a different Citopia node (logical and physical decoupling of trip information). Rider DIDs are removed from anonymized trip data and ZKPs are used to ensure data privacy while preserving data correctness, integrity, and verifiability. This way neither the service provider nor Citopia can identify PII from the anonymized trip data. To use ZKPs at this step, the project team implemented the Zero Knowledge (ZK) Service using cryptographic constructs known as Merkle Trees and Merkle Proofs. The ZK Service generates a Merkle proof for each trip leg as a proof-of-membership in the trip and verifies it. The Merkle tree and the Merkle proof for each trip leg are generated on the rider device side, while the Citopia server side verifies the Merkle Proof. At the end of the trip, each leg of anonymized trip data is processed and stored in the data store (Couchdb), which can be either published or consumed by third parties or service providers for analysis.

The effort to resolve navigation/map challenges caused some delay. The total delay was approximately three months after the initially planned completion time for the demo. The demonstration of the fully functional mobile app for riders and web interface for service providers to the expert advisory panel was postponed to December 21, 2022.

TASK 3 — INTEGRATE CITOPIA MAAS AND THE ITN CORE SERVICES

In conjunction with Task 2, the project team also worked on Integrating Citopia and ITN Core Services (Task 3). The Integrated Trust Network is a protocol-agnostic, blockchain-based trusted identity services network that serves as Citopia's trust layer for identity. Citopia needs to connect to the ITN to enable multiparty business automation via DIDs and SSDTs. Through the distributed network of nodes, blockchain enables the ITN to be trustless, such that entities do not need to know each other to establish a trust relationship to execute transactions. It also enables transactions to be tamper-evident by combining cryptographic digital signatures with the data integrity properties of the utilized cryptographic signing keys because of the distributed nature of the ledger/blockchain. Another advantage of using blockchain is redundancy, meaning that a copy of every DID is distributed among the blockchain nodes; as a result, if one node is down, then the information is not lost as the other nodes have a copy. This adds a layer of decentralization as one single entity does not have control over the network. This also eliminates the need for middlemen, thereby increasing efficiency. ITN core services including identity, assurance, and governance allow for application interoperability and multiparty data sharing so that participants can execute trusted Web3 transactions on Citopia.

The project team started working on Task 3 by designing the onboarding processes for the riders and service providers. Deploying SSDTs is a crucial step in the onboarding process, which requires the ITN Core Services and deployed SSDT instances with its SDK. In the first quarter, the necessary integration for the deployment of SSDTs was completed. After this, DIDs were established using the SSDT's DID management SDK, and built-in DIDComm messaging (a component of an SSDT) was instantiated for communication between Citopia MaaS SSDTs and with the ITN Core Services, since the ITN is a network, and each ITN node has its own SSDT. After this was successfully completed, the project team defined relevant operating and integration parameters and utilized ITN Core Services and the SSDT SDK. After the second quarter, the generation of DIDs for service providers and riders using the ITN was successfully completed. By enabling the service provider and rider SSDTs to interact with DIDs of mobility assets provided by the ITN, these steps ensure that service providers can seamlessly offer multimodal transit/ MaaS services on Citopia.

The development of the mobile app encountered compatibility issues in Task 3 due to Javascript packages utilized in the browser-based version of the SSDT SDK not being compatible with the React Native development framework used. The development team focused on solving the compatibility challenges of the SSDT SDK Javascript libraries with React Native in the second quarter. The cryptographic packages required more complex development efforts to enable compatibility with React Native. The project team then started rebuilding the integration layer between the native code (typically C++) and React Native, which involved rewriting the JavaScript libraries in React Native. This was a substantial undertaking that took approximately 5 months to complete the first iteration. The project team then tested this approach for each utilized library. Once this was successfully completed, the development focused on implementing the trip planning workflows in Task 2. In addition, the coordination of communications regarding trip-specific transitions between riders and service providers was successfully established in Task 3.

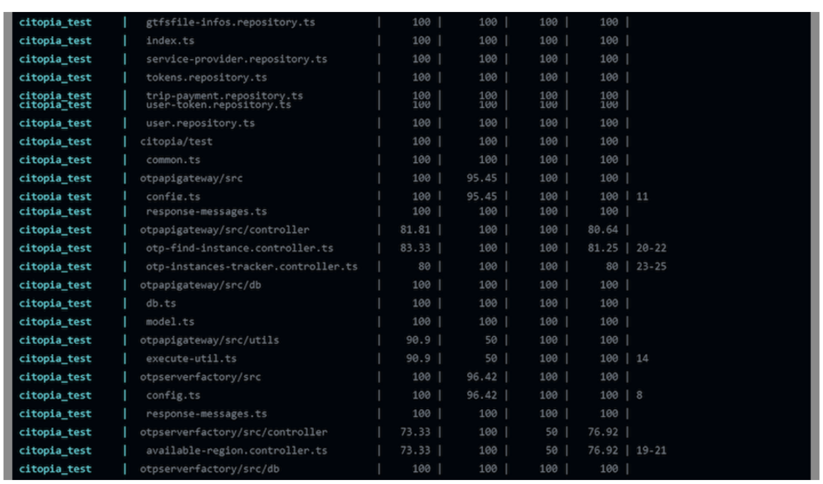
TASK 4 — VERIFYING INTEGRATION FUNCTIONALITY

Concurrent with Task 3 and Task 2, the project team worked on Verifying Integration Functionality (Task 4) to support the ongoing development of those tasks. In the first quarter, the first iteration of configuring the SSDT digital wallet was completed successfully. The digital wallets in the SSDTs enable rider and service provider identities to be stored in the respective SSDTs on the rider's /service provider's device. This way, Citopia or any other party except the owner of the SSDT does not have access to the stored information. The rider always has control over their PII and the service provider always has control over their data. Workflows that helped to demonstrate verifying integration functionality are the successful onboarding of a service provider or rider and verifying the service provider's identity when a rider books a trip. Once the React Native libraries were completed and successfully deployed and the route generation, trip booking, navigation, invoicing, and trip completion workflows were implemented in Task 2, the project team started end-to-end testing. Potential improvements and bugs were identified during testing. The project team incorporated the identified improvements and corrected the errors. After the results of the end-to-end testing yielded the desired results of demonstrating workflows end-to-end, the project team proceeded with the User Acceptance Testing (UAT).

TASK 5 — USER ACCEPTANCE TESTING AND SIMULATED DATA DEFINITION

The project team started work on Simulated Data Definition from Task 5 during the first quarter. By carrying out research and interviews with transit subject matter experts, criteria for demo test cases and the corresponding data needs were defined.

In the second quarter demo, the testing criteria were refined, which guided writing the UAT test cases. The team also started testing the early demo version on iOS (in TestFlight) and Android (in Play Store) in the second quarter. This was a preliminary testing phase to identify UI errors, define potential improvements, implement relevant bug fixes, updates, and add or remove functionalities. The testing continued only on iOS due to the navigation challenges with Mapbox/Mapquest described in Task 2. The app was continuously tested and iterated until the fully functional demo was released with Citopia MaaS version 20 (v20). The test cases were written and once this version was developed, the project team started UAT. The goal of UAT was to test every functionality of every feature of the mobile app while prioritizing the main functionalities such as multimodal trip planning, route generation, and trip execution. More than 250 test cases were generated and thoroughly tested by the project team during UAT. The results of the UAT were reported regularly on the UAT testing spreadsheet, including bugs and errors. Bugs and errors were also reported on a Trello board for the UAT with detailed descriptions. In addition to UAT, the development team also continuously tested the app in the backend. Developers wrote test scripts for several packages, including Citopia Controllers, Service Providers, OTP Server Factory, OTP API Gateway, and Web UI. All reported errors were successfully addressed. The fully functional demo was substantially improved throughout the entire testing period.



citopia_test	gtfsfile-infos.repository.ts	100	100	100	100	
citopia_test	index.ts	100	100	100	100	
citopia_test	service-provider.repository.ts	100	100	100	100	
citopia_test	tokens.repository.ts	100	100	100	100	
citopia_test	trip-payment.repository.ts	100	100	100	100	
citopia_test	user-token.repository.ts	100	100	100	100	
citopia_test	user.repository.ts	100	100	100	100	
citopia_test	citopia/test	100	100	100	100	
citopia_test	common.ts	100	100	100	100	
citopia_test	otpapigateway/src	100	95.45	100	100	
citopia_test	config.ts	100	95.45	100	100	11
citopia_test	response-messages.ts	100	100	100	100	
citopia_test	otpapigateway/src/controller	81.81	100	100	80.64	
citopia_test	otp-find-instance.controller.ts	83.33	100	100	81.25	20-22
citopia_test	otp-instances-tracker.controller.ts	80	100	100	80	23-25
citopia_test	otpapigateway/src/db	100	100	100	100	
citopia_test	db.ts	100	100	100	100	
citopia_test	model.ts	100	100	100	100	
citopia_test	otpapigateway/src/utils	90.9	50	100	100	
citopia_test	execute-util.ts	90.9	50	100	100	14
citopia_test	otpserverfactory/src	100	96.42	100	100	
citopia_test	config.ts	100	96.42	100	100	8
citopia_test	response-messages.ts	100	100	100	100	
citopia_test	otpserverfactory/src/controller	73.33	100	50	76.92	
citopia_test	available-region.controller.ts	73.33	100	50	76.92	19-21
citopia_test	otpserverfactory/src/db	100	100	100	100	

FIGURE 5 Test Coverage Report Console Log from backend testing.

TASK 6 — COMPLETE DEMO

Task 6 aims to successfully demonstrate multimodal trip planning, managing ticketing, and payments on Citopia MaaS. Two demos were developed: a dynamic and a static demo.

6.1 Dynamic (Actual) Demo

The completed versions of the fully functional dynamic Citopia MaaS mobile app and web interface demos were presented to the advisory panel on December 21, 2022. The dynamic demo involved the successful demonstration of creating rider Citopia SSDTs, multimodal trip planning using real-time GTFS and GBFS data with the ability to choose from multiple MaaS options, managing upcoming and current trips, trip execution and navigation, seamless payment, viewing trip history, and summary and invoice information on the mobile app. The mobile app also demonstrated the successful generation of agreement, MTT, and invoice VCs. In addition to these, the mobile app will also demonstrate the ability for the rider to upload credentials, access their account information, and manage their data/privacy preferences. The demo app offers multimodal transit options in the northern California/San Francisco region following GTFS and GBFS standards. This region was chosen due to the greater availability of data and ease of access to this data. The transit/MaaS options currently available for testing on the app are bus, metro, bike-as-a-service, and scooter-as-a-service. The providers that were onboarded for the demo include:

- Bus Network: AC Transit and SamTrans via GTFS
- Light Rail: San Francisco Municipal Transportation Agency (MUNI Metro Light Rail) via GTFS
- Heavy Rail: Bay Area Rapid Transit (BART) and Caltrain via GTFS
- Bike-as-a-Service: Bay Wheels via GBFS
- Scooter-as-a-Service: Lime via GBFS

Route generation prioritization for fastest routes and trip execution involving these modes of transit among the above service providers were successfully demonstrated during the demo. Future developments will include more private transport modes such as Uber and Lyft. These were not included in the demo because they will require customizing the trip planning and routing algorithm, which will require more development time and resources. The demo also showed that riders can choose route preferences among the fastest, cheapest, greenest, and least number of transfers; however, the demo app only displays route results based on the first option which is the fastest route. In the future implementation of the app (post-project completion), the development will expand the algorithm to enable prioritizing route options based on the other preferences as explained in Task 2.

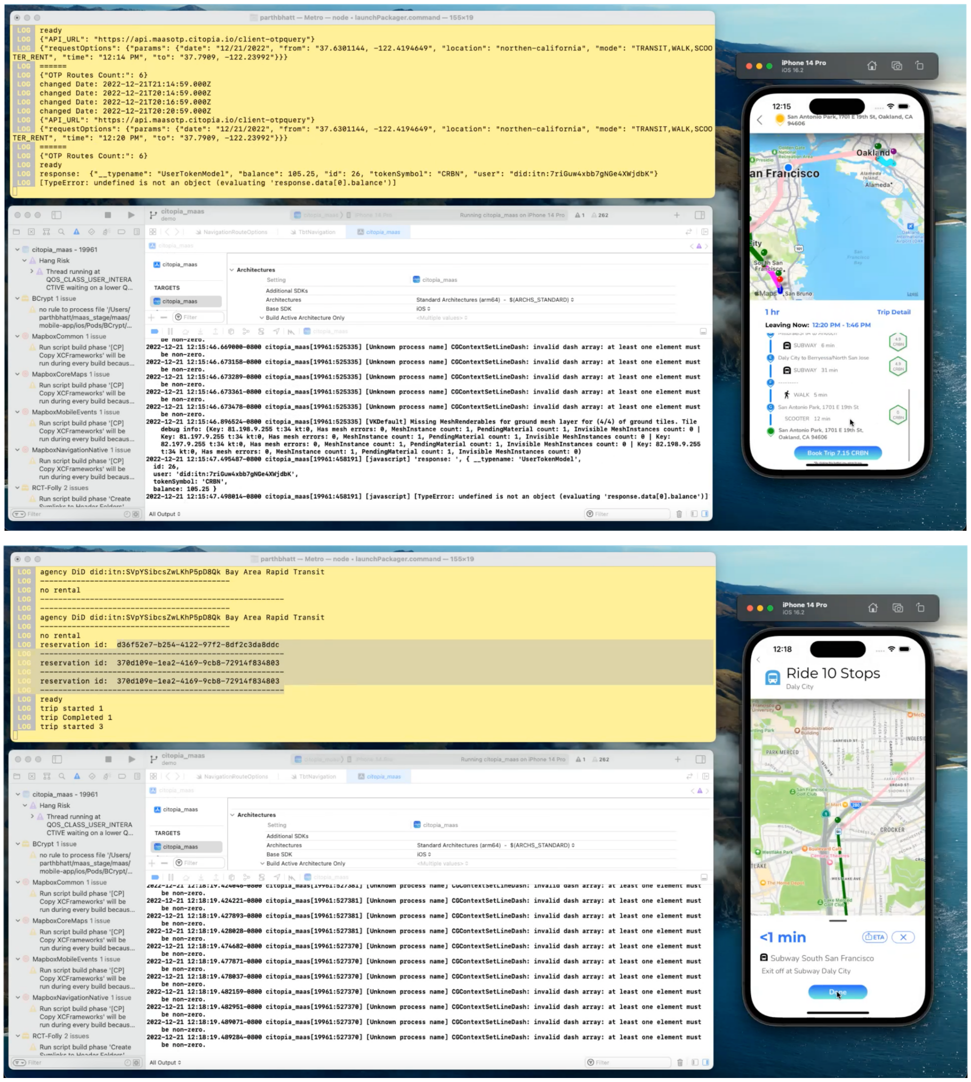


FIGURE 6 Screenshots from the live demo.

For service providers, the demonstration showed the successful creation of service provider Citopia SSDTs, onboarding assets and managing service offerings, and making multimodal trip data available, which involves connecting API service endpoints on the web interface. In addition to these, the web interface demonstrated the ability of the service provider to onboard multiple riders (on behalf of the transit agency/MaaS provider), access their organizational profile, and manage their data/privacy preferences.

6.2 Zero Knowledge Proof Implementation

ZKP generation was successfully presented as a part of the dynamic demo to the expert advisory panel on December 21, 2022. This involved the demonstration of the Merkle tree and proof generation on the rider device and verification of the Merkle proof by the Citopia server. Rider device generates the ZKP proof of payload for trip leg data, which is then forwarded to a Citopia Node. Figure 7 shows the successful verification of the ZKP by the Citopia node (right side). After this, it is forwarded to the data store (left side).

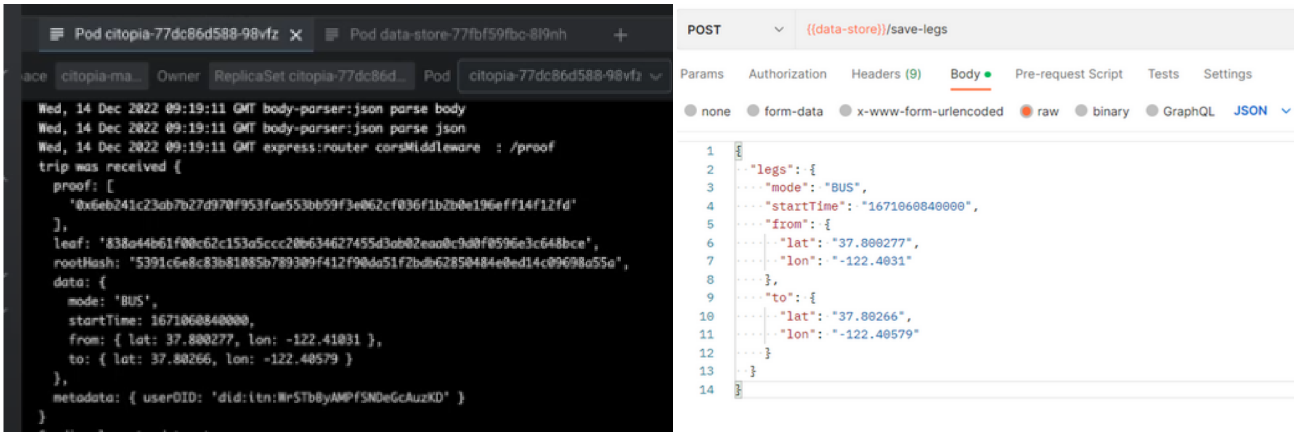


FIGURE 7 ZKP Proof Generation in navigation.

6.3 Static Demo

The static demo was only demonstrated for the mobile app as it uses mock-up data with no real service provider data integration in the backend. This is due to the lack of open-source APIs that are required to enable these MaaS options with real-time data from actual service providers. Additionally, building the functionalities required by modes of transit such as car rental and ride-hailing require significantly more time for standardization and coordination with and among those service providers, along with increased development time and resources. The static demo rider journey aims to show how other types of MaaS (such as car rental, RUC, and usage-based parking) would work in a multimodal trip on Citopia – all of which require additional integrations. The static demo was successfully demonstrated on December 21, 2022. It involved trip planning and execution of a multimodal trip including scooter, metro, and car rental as modes of transit, along with usage-based parking, usage-based insurance, Road Usage Charge (RUC), and tolling.

TASK 7 — SUBMIT A STAGE I REPORT AND PROJECT PROGRESS REVIEW

The project progress in Stage I was reported for each task. The report elaborated on the work done, issues encountered, and solutions adopted to address them. The Stage I report was finalized after the panel reviewed the draft and submitted their feedback to the project team.

TASK 8 — TEST (PERFORMANCE TESTING) AND EVALUATE THE SYSTEM

Within Task 8, the project team worked on the following processes. First, the KPIs for the System were defined. The next step was to define the scope of system components and their endpoints critical for the determined KPIs. A performance testing approach and setup for the in-scope system components will be defined (such as measurement process, required infrastructure, and testing loads.) Then the performance test fixtures (such as test data sets for each system endpoint to be tested, the number of requests for a given testing period, and the number of individual riders) were determined and created. The performance testing infrastructure was set up. Performance testing was done and the testing and infrastructure parameters were adjusted, based on testing results to optimize system performance. Finally, performance testing results were documented to determine system suitability for operating under real-world conditions and report findings.

The main workflows to be tested were rider onboarding and trip execution. Onboarding includes the creation of DIDs (which are anchored on the ITN through Citopia), SSDT, and Citopia membership VCs. The ride execution workflows to be tested included querying/searching, requests per second for individual APIs, and the creation and exchange of VCs for booking (agreement VCs), trip execution via MTT VCs (trip commencement and trip completion), and invoice VCs (and payment to service providers).

8.1 Testing Approach and Specs

The testing approach was to incrementally increase the requests per second for each individual API endpoint or specific workflow (mentioned above). Testing started with 100 requests and adding increments of 100 until 1,000 requests are reached, which would then be followed by increments of 1,000 until 10,000 requests are reached and then increments of 10,000 up to 100,000 requests are reached. The latencies for each request were measured and logarithmic plots of requests per second against latency were generated. The latency value is shown as the average point with the error bars denoting the 1 and 2 sigma interval and min and max for each increment. To understand latency dependency on infrastructure sizing, the server configuration was varied by defining different "small", "medium", "large" and "x-large" server configurations and measuring latencies. The ITN server size was not changed to show if there are internal vs. external bottlenecks (ITN, OTP ~ GTFS/GBFS, etc.). Table 2 below shows the main tasks that were performed in the performance testing.

The project team started testing with the constant arrival tests and proceeded with ramp arrival tests using the hardware and software setup shown in Table 3. A constant arrival test is a type of performance test used to measure the system's ability to handle a steady and constant stream of incoming traffic over a period of time. In this test, a fixed number of virtual users or clients are simulated to generate requests to the system at a constant rate, typically measured in requests per second or transactions per second. The goal of the constant arrival test is to measure the system's ability to handle the incoming traffic and to identify any performance bottlenecks (such as high response times, high error rates, and resource utilization problems) that may cause delays or failures in processing the requests. To perform a constant arrival test, a test tool, Grafana K6, is typically used to generate a constant stream of requests to the system while monitoring the system's response time, error rate, and resource utilization metrics.

The ramp arrival test is a type of performance test used to measure the system's ability to handle increasing traffic over a period of time and system response. In this test, the number of virtual users or clients is gradually increased over time to generate requests to the system at an increasing rate, typically measured in requests per second or transactions per second. The goal of the ramp arrival test is to simulate a realistic traffic pattern and to identify the system's performance limits, bottlenecks (such as slow response times, high error rates, and resource utilization problems), and capacity. To perform a ramp arrival test, a test tool is typically used to gradually increase the number of virtual users or clients over a specified time period, while monitoring the system's response time, error rate, and resource utilization metrics. After both types of tests are completed, the results are analyzed to identify any performance issues and to optimize the system's performance.

TABLE 2 Task Items and Progress

Task	Description
Generating data using faker library for each of the test scenarios	Save output to json format
Grafana K6 tool setup	To run constant arrival tests from generated data
Save the Rider DIDs and associated data from onboarding flow in the redis server	Storing the rider onboarding workflow output in Redis, which later was used as the data source in performance testing for Rider trip workflow
Rider - Searching & Planning API test ~ OTP query	APIs for Rider to search the destinations and see the route options available **Note:** Using Open Trip Planner (OTP) for Route Planning
Setup trip reservation data with DIDs from redis and random reservation data from data-generator	Preparing & storing the data required for trip reservation flow (includes some data attributes from Rider onboarding flow)
Started working on the deployment scripts to run the tests on AWS (Amazon Web Services) Infrastructure	Deployment scripts are required to deploy the performance testing setup on to AWS Infrastructure

TABLE 3 Hardware and Software Setup for Performance Testing

Hardware Setup	Deployment Architecture	Software Deployment
Initially 4 instances of AWS t3.large which consists of 2 virtual Central Processing Units (vCPUs) and 8GB Random Access Memory (RAM) for each instance	The ITN Core Services and the Fabric DLT were deployed into separate AWS 3-node EKS (Elastic Kubernetes Service) clusters with ingress controllers and load balancers	Configuration containerization, and deployment of the multi-node (3 orderer + 2 peers) Fabric DLT (Distributed Ledger Technology) version 2.4 into its designated AWS EKS cluster following well documented deployment scripts
Later on upgraded to 4 instances of AWS t3.2xlarge which consists of 8 virtual Central Processing Units (vCPUs) and 32GB Random Access Memory (RAM) for each instance	The Self-Sovereign Digital Twin representing the riders, service providers (transit agencies), and Citopia MaaS were deployed again into separate AWS 1-node EKS clusters with ingress controllers and load balancers	Configuration, containerization, and deployment of the Core Services with Core Services Self-Sovereign Digital Twin into its designated AWS EKS cluster

It should be noted that the discovery of the SSDTs, and establishment of communication is based on the API endpoints specified in the DID documents created for each actor (riders, service providers, and Citopia MaaS). The communication between the various SSDTs as representatives of the riders, service providers, Citopia MaaS, and the ITN followed the DIDComm messaging standard.

8.2 Performance Testing Results

TABLE 4 Rider Onboarding Flow Test Results

Flow Name	Total Number of Requests	Number of Request Execution Per Second	Average Request Duration
Rider Onboarding Flow	200	0.83	2.62s
Rider Onboarding Flow	800	3.75	24.06s

TABLE 5 Rider Trip Flow Test Results

Flow Name	Total Number of Requests	Number of Request Execution Per Second	Average Request Duration
Roll Up (Total) - End to End Flow	~300k	~1905	147.71s
request-reservation	~100k	833.33	34.20s
trip-commencement	~100k	833.33	46.51
trip-completion	~100k	238.1	1m

It is important to note that testing the rider trip flow (results shown in Table 5), requires OTP query prior to request-reservation. So the average request duration will be higher when an OTP query is included.

TABLE 6 End to End Flow Test Results

Flow Name	Total Number of Requests	Number of Request Execution Per Second	Average Request Duration
End to End Flow	~37590	~277	2m57s

End-to-end testing results shown in Table 6 refer to all trip APIs inclusive of trip querying, trip booking and trip execution (request_reservation + otp_query + trip_commencement + trip_completion). Results shown in Table 5 tested the individual APIs and aggregated the total time (individual API endpoints for trip reservation, trip commencement, and trip completion captured the latency and aggregated those - please see Table 7 in Appendix E). The end to end test by default includes all the API logic in one flow so there was no need to aggregate the total time.

With the testing configuration explained in this section, we found that a Citopia Node can handle up to 300 concurrent rider trips per second, which include trip querying, booking and execution (commencement, completion), and invoicing. Please note that this result is valid for only one Citopia Node. If there were multiple nodes it would increase linearly (e.g. if there are two nodes then 600 trips would be processed). We also found that a Citopia node can onboard 200 riders in 2.62 seconds and 800 riders in 24.06 seconds.

8.3 Recommendations based on Performance Testing

The following is a list of recommendations that were derived from the tasks that were carried out and results obtained from the performance testing involving the rider onboarding flow and end-to-end rider trip execution flow. These will benefit the development of a real-world implementation/pilot of Citopia MaaS.

- In order to handle more requests in less time, a K8s cluster (a technical infrastructure configuration/setup) for each service provider with an OTP instance should be set up.
- The Open Trip Planner is an open source multimodal trip planning service, and, therefore, needs additional performance testing.
- It is useful to explore and implement a queuing system to process some of the more time-consuming requests asynchronously such as requests between a Rider and the Open Trip Planner multimodal trip planner service. This will be implemented in a new version of the MaaS Citopia application.
- The Rider onboarding process creates an identity for the rider and anchors the DID onto ITN. The current design of the SSDT's EDV (Encrypted Data Vault), which handles the write operations of an ITN Node on the ITN network, requires some redesign to be able to handle more requests in less time, for example, 100K onboarding requests in 6 to 10 minutes. This is an optimization that will be implemented in the ITN code base.
- Given the desired workloads in a geographic area, the ITN infrastructure configuration will need to be upgraded from "t3.large" to "m5.8xlarge" or higher for the ITN core-services and DID-resolver K8s pods. In addition, performance testing results show that we need a larger number of ITN nodes for processing. Furthermore, we are considering adopting a more scalable DID method such as a DID method based on the DIF (Decentralized Identity Foundation) Sidetree Standard such as did:elem as the primary ITN DID method.

TASK 9 — PREPARE AND SUBMIT A FINAL REPORT

The project drafted the final report including work done throughout the entire project, challenges encountered, solutions adopted to solve them, project findings, and feasibility assessment for real-world implementation of Citopia MaaS. The draft was presented to the expert advisory panel, finalized after their feedback, and submitted to the Transit IDEA Program on April 12, 2023 hence completing the project.

COMMUNICATIONS WITH THE EXPERT ADVISORY PANEL

The project team met with the expert advisory panel every month to discuss project updates, direction, and deliverables focusing on the demo progress until the performance testing began. During the performance testing, the project team sent bi-weekly test reports instead of meetings. A final advisory meeting was held to discuss project results and the final report and conclude the project. The expert advisory panel's feedback regarding project tasks, progress, and deliverables was actively incorporated into the project development.

Plans for Implementation

This project has developed a pilot version to evaluate the feasibility of real-world implementation of Citopia MaaS. The pilot successfully demonstrated seamless MaaS planning, booking, trip execution, and payments. The underlying technology infrastructure and data privacy protection approach (DIDs, VCs, SSDTs, and ZKPs) were also successfully demonstrated.

After the successful user acceptance testing and live demo of the mobile application and web interface, Stage II performance testing was carried out successfully. The results of the performance testing showed that the system is able to handle the expected number(s) of transactions involving rider onboarding and trip booking and trip execution flows (explained in detail in the Investigation section). Rider onboarding test includes rider DID creation (through the ITN) and SSDT and Citopia membership VC creation. Trip booking and execution tests include the creation of VCs that are needed to initiate and complete a multimodal trip (MTT, reservation VCs, and invoice VCs). The results show that the technology is ready for real-world application and can be scaled through more technical resources and adoption by the sector and riders. Given the successful demonstration of technical infrastructure/technology readiness, the most important factor is the latter; adoption. The most important factor for Citopia MaaS to be adopted by service providers is securing sufficient funding from the public and private sectors. In order to get the buy-in, the next step would be to present the project to regional and/or municipal transit agencies and DOTs and conduct outreach for other public and private partners/collaborators.

Once sufficient funding is available to launch a real-world pilot in a particular region/city, the project team will establish a pilot-scale network of MaaS servers, which is not connected to the production network. This step is needed to test the Citopia MaaS network for readiness and implement the necessary configuration and iterations. As the MaaS and transit options and usage trends can be different in each region, they can have their own specific customization of MaaS applications. Each transit region where Citopia MaaS can be used can have a pilot phase. Once a pilot scale of MaaS servers are successfully established in a region (i.e. has demonstrated readiness), then it can be added to the Citopia MaaS production sub-network with transit agencies interacting with one or more nodes in the network. Sub-networks connect to make up the production network, which can be a global network of service providers across different regions in the world. The same process would be followed to launch Citopia MaaS in a new region and connect to a production sub-network. Citopia MaaS will scale up as more sub-networks form with service providers and riders onboarding from different regions.

The demo app developed in this project uses a dummy credit card as the payment method. This is to demonstrate the invoicing flow involving the creation and exchange of invoice VCs, maintaining the accounting ledger and bulk payments to service providers that are in scope. In a real-world scenario, the app would support other payment methods and enable refunds where necessary depending on the specific policies of each service provider.

Once commercially deployed Citopia is planned to generate revenue through Self-Sovereign Digital Twin as-a-service and transaction fees, which include API fees, interactions between SDDTs (anytime VC or VP is exchanged), and between an SSDT and the ITN. It should be noted that ITN core services usage fees for trusted identity services are separate from Citopia MaaS and out of the scope of this project. On the other hand, the cost structure will consist of operating costs including R&D, engineering, cloud storage, marketing, sales, and general and administrative costs.

We were successfully able to integrate public transport services such as buses and some private/public e-mobility services such as scooter-as-a-service onto Citopia MaaS as APIs are available through GTFS and GBFS that are open source. However, we faced obstacles in integrating some private service providers, especially Transportation Network Companies (TNCs) such as ride hailing providers like Uber and Lyft. These could not be included in the demo because they require customizing the trip planning and routing algorithm, which will require significantly more development time and resources. There is currently a lack of regulations and ambiguity around existing regulations for multimodal ridesharing within the context of data privacy and payments. In order to implement this on a large-scale real-world application, there is a need for the creation of data privacy regulations and standardization regarding data sharing, data privacy, and extensions to existing ones. There is also a need for the creation of new API standards to establish interoperability. One of the key enablers is participation and adoption from the private sector in the creation of these standards. Future developments aim to include more TNCs and other private transport modes as there is more clarity around regulations and more data is available in a standard-compliant way.

Although the project focused on demonstrating new technology primitives such as SSDTs, DIDs, and VCs in a MaaS/transit context, we believe the outcomes offer tremendous potential to be used in a vast array of transportation demand and supply chain management strategies. Citopia can unlock new business models for public/private mobility service providers through usage-based business automation and allow them to work and fairly compete with each other on a federated Web3 marketplace without relying on a centralized authority/platform.

We recognize that this project and these opportunities would not have been possible without TRB’s support – both guidance and financial. Citopia and the Advisory Committee hope the Board will see the potential for this technology and help support future efforts for implementing a real world pilot with a city DOT.

Conclusions

The Citopia MaaS project was successfully completed in February 2023 in two stages, demonstrating seamless MaaS planning, booking, trip execution, and payments by leveraging Web3 technologies and ensuring rider and service provider data privacy.

In Stage I, the complete and fully functional dynamic demo of the Citopia MaaS mobile app and web interface was successfully presented to the expert advisory panel along with the static demo and Zero-Knowledge Proof generation. The project team successfully demonstrated the creation of Citopia SSDTs; multimodal trip planning using real-time GTFS and GBFS data involving bus, metro, and bike/scooter-as-as-a-service MaaS options; and trip booking, navigation, and payments on Citopia MaaS.

In Stage II, we conducted performance testing on the feasibility of the project for a real-world pilot project. Results showed that the system is technically viable for real-world implementation because the Citopia node was able to process 300 end-to-end rider trips (including querying, booking, and trip execution, and invoicing) per second. Results also showed that a Citopia node can onboard 200 riders in 2.62 seconds, which goes up to 800 riders in 24.06 seconds. These performance results are for one node, a scaled-up system with more nodes would have incrementally more processing capacity. In a real-world implementation, it is reasonable to assume that the Node can handle a greater number of transactions and API calls if the system is improved following the recommendations in Section 8.3 in the Investigation section. It is important to highlight that in addition to demonstrating technical readiness, Citopia MaaS' data privacy protection meets all necessary regulatory and business requirements.

In the assessment of the project results, we identified the need for a real-world pilot that would address the demands of MaaS riders and service providers. The biggest challenge to real-world deployment is the lack of proper legal and business incentives/ disincentives to encourage collaboration between public and private service providers and other transit providers. Collaboration is key as Citopia MaaS will operate as a network of nodes serving as a federated Web3 marketplace where service providers (who also may or may not be node operators) can offer their services to riders. The adoption of a real-world implementation ultimately depends on collaboration. In order for collaboration to happen with TNCs and ridesharing companies there need to be more standards and legal clarity on data sharing, data privacy, and payments in addition to standard API formats/API standards for integrating MaaS services onto Citopia MaaS. Thus it is important to have monetization and legal incentives for providers to collaborate to encourage the use of MaaS and multi-modal transit applications. Another important factor for Citopia MaaS to be adopted and scaled is buy-in and participation from public and private service providers. Lastly, we have also found that a real-world pilot requires substantially more resources and funding to implement.

We envision that the commercial deployment of Citopia MaaS will redefine seamless multimodal trip planning, ticketing, and payments. It will connect public and private transit/MaaS service providers with riders in a secure, efficient way while maintaining data privacy for all parties in a way that has not been achieved before via state-of-the-art Web3 technologies. Service providers will easily be able to leverage usage-based business automation and multiparty coordination; simultaneously have full control over their data and not have to rely on a centralized platform. This will ultimately enable them to generate new value to create novel usage-based business models or add value to existing business models.

References

(1) "U.S. Cities Factsheet." 2021. Center for Sustainable Systems. 2021. https://css.umich.edu/publications/factsheets/built-environment/us-cities-factsheet.

(2) "Mind the Gap - How to Close the Gap between Theory and Reality in Multi-Modal Mobility." 2017. Www.bearingpoint.com. June 27, 2017. https://www.bearingpoint.com/en-us/insights-events/insights/multi-modal-mobility/.

(3) CNBC. 2022. "Facebook Parent Settles Suit in Cambridge Analytica Scandal." CNBC. August 27, 2022. https://www.cnbc.com/2022/08/27/facebook-parent-settles-suit-in-cambridge-analytica-scandal.html.

(4) "Decentralized Identifiers (DIDs) V1.0." n.d. Www.w3.org. https://www.w3.org/TR/did-core/.

(5) "Self-Sovereign Digital Twins™." MOBI | the New Economy of Movement. n.d. Accessed April 12, 2023. https://dlt.mobi/self-sovereign-digital-twins/.

(6) Intersoft consulting. 2013. "Art. 4 GDPR – Definitions | General Data Protection Regulation (GDPR)." General Data Protection Regulation (GDPR). 2013. https://gdpr-info.eu/art-4-gdpr/.

(7) "Verifiable Credentials Data Model 1.0." 2019. W3.org. November 19, 2019. https://www.w3.org/TR/vc-data-model/.

(8) Hardman, Daniel. 2019. "A Gentle Introduction to Verifiable Credentials - Evernym." Evernym. October 24, 2019. https://www.evernym.com/blog/gentle-introduction-verifiable-credentials/.

(9) CSRC. n.d. "Zero-Knowledge Proof - Glossary | CSRC." Csrc.nist.gov. Accessed April 12, 2023. https://csrc.nist.gov/glossary/term/zero_knowledge_proof.

Appendix A: Research Results

Program Steering Committee: NCHRP IDEA Program Committee
Month and Year: March 2023
Title: Citopia MaaS/Multimodal - Transit IDEA Project T-101
Project Number: T-101
Start Date: January 3, 2022
Completion Date: March 29, 2023
Product Category: MaaS/ Public Transit App and Web Interface
Principal Investigator: Tram Vo
Name, Title: Tram Vo, Founder and CEO of Citopia
E-Mail: tram@mobi.world
Phone: 310.613.1091

TITLE:
Citopia MaaS/Multimodal Transportation App and Web Interface

SUBHEAD:
The pilot successfully demonstrated seamless one-stop Mobility-as-a-Service (MaaS)/Multimodal trip planning, booking, trip execution, and payments using Web3 technologies and data privacy protection.

WHAT WAS THE NEED?
Mobility/transportation demand is changing with usage-based models. Riders want a seamless, efficient, integrated multimodal transit and MaaS experience while having control over their data. Increasing the adoption of usage-based mobility business models such as public transit, ride hailing, e-bikes, car renting/sharing, usage-based parking, etc. is a promising approach to tackling low-carbon transportation and mobility infrastructure challenges. However, most mobility/transportation services are offered by siloed centralized platforms today and lack the interoperability to connect disparate mobility services. These platforms also fail to offer the data privacy and control of personal data that users demand.

WHAT WAS OUR GOAL?
Citopia MaaS aimed to demonstrate seamless and complete multimodal trip planning, booking, trip execution, and payments by leveraging Web3 technologies and ensuring rider and service provider data privacy. The project team developed a pilot version to evaluate the feasibility of a real-world implementation.

WHAT DID WE DO?
The project was carried out in two stages, consisting of nine tasks listed below.

- Task 1: The project was launched. The scope, KPIs, and project plan were defined.
- Task 2: The frontend and the backend of the fully functional Citopia MaaS mobile app (for riders) and web interface (for service providers) was built.
- Added task: Integrated Trust Network (ITN) SDK packages (added React Native Support)
- Task 3: Integration of the ITN to Citopia was successfully executed to enable trusted identity services.
- Task 4: The ongoing work in Task 2 and Task 3 was supported by establishing workflows to verify

integration functionality of the ITN.

- Task 5: User Acceptance Testing was carried out to test every functionality of the mobile application to fix errors and improve functionality.
- Task 6: The complete and fully functional Citopia MaaS mobile application and web interface were demonstrated to the expert advisory panel along with data privacy protection.
- Task 7: The Stage I Report was prepared and submitted to the Transit IDEA Program.
- Task 8: Performance Testing was carried out (with defined KPIs) to evaluate the feasibility of real-world implementation of Citopia MaaS.
- Task 9: The final report was prepared and submitted to the Transit IDEA Program.

WHAT WAS THE OUTCOME?

The Citopia MaaS project was successfully completed in February 2023. The pilot successfully demonstrated seamless MaaS planning, booking, trip execution, and payments. The underlying technology infrastructure and data privacy protection approach involving Decentralized Identifiers (DIDs) , Self-Sovereign Digital Twins (SSDTs), Verifiable Credentials (VCs) and Zero-Knowledge Proofs (ZKPs) were also successfully demonstrated. The feasibility of real-world implementation of Citopia MaaS was evaluated in the performance testing in the second stage of the project. The results of both stages of the project demonstrated technical readiness of Citopia MaaS. The biggest challenge to real-world deployment is the lack of data standardization, policy, and business incentives/disincentives to encourage collaboration between public and private service providers. Buy-in and participation from public and private service providers is essential for Citopia MaaS to scale. Lastly, commercial deployment requires substantially more resources and funding.

WHAT IS THE BENEFIT?

We envision that the commercial deployment of Citopia MaaS will define one-stop multimodal decentralized solution for services discovery, ticketing, and payments. It will connect public and private transit/MaaS service providers with riders in a secure, efficient way while maintaining data privacy for all parties in a way that has not been achieved before via state-of-the-art Web3 technologies. Service providers will easily be able to leverage usage-based business automation and multiparty coordination while retaining full control over their data without needing to rely on centralized platforms. This will ultimately enable them to generate new value to create novel usage-based business models and/or add value to existing business models.

LEARN MORE

To view the evaluations: tram@mobi.world

IMAGES

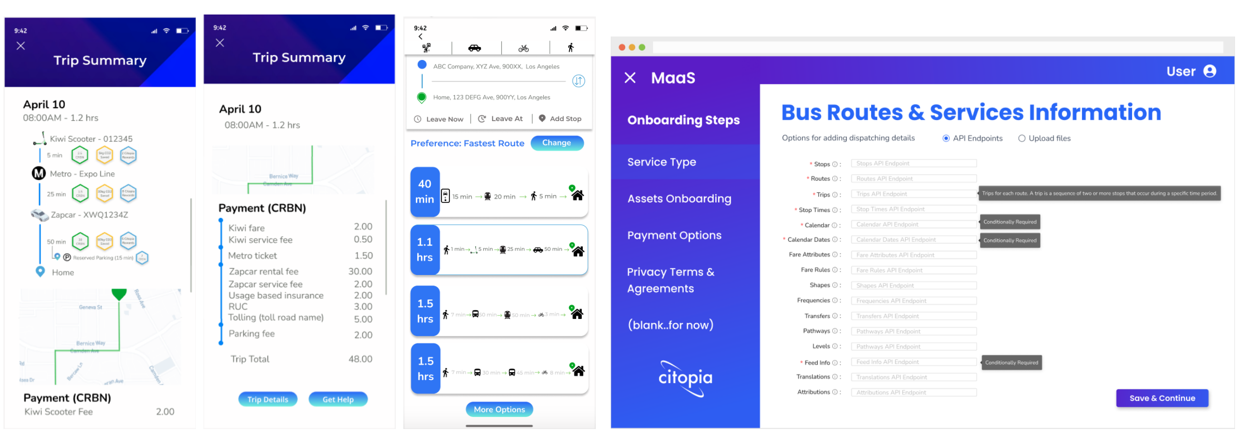


FIGURE 8 Screens from Citopia MaaS mobile app and web interface UI.

Appendix B: Citopia MaaS UI

The following are screenshots of various screens in the mobile app UI.

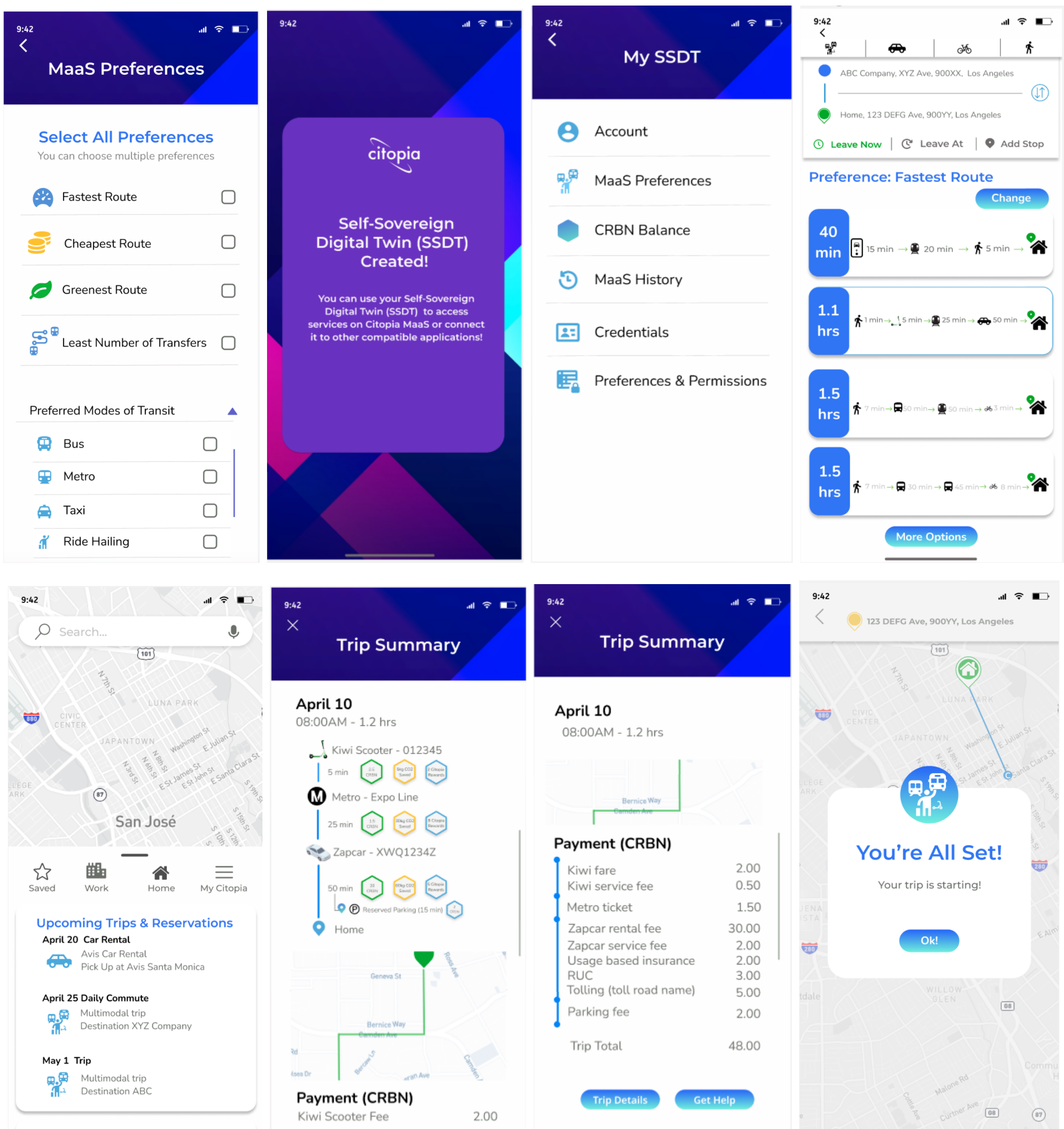


FIGURE 9 UI screenshots.

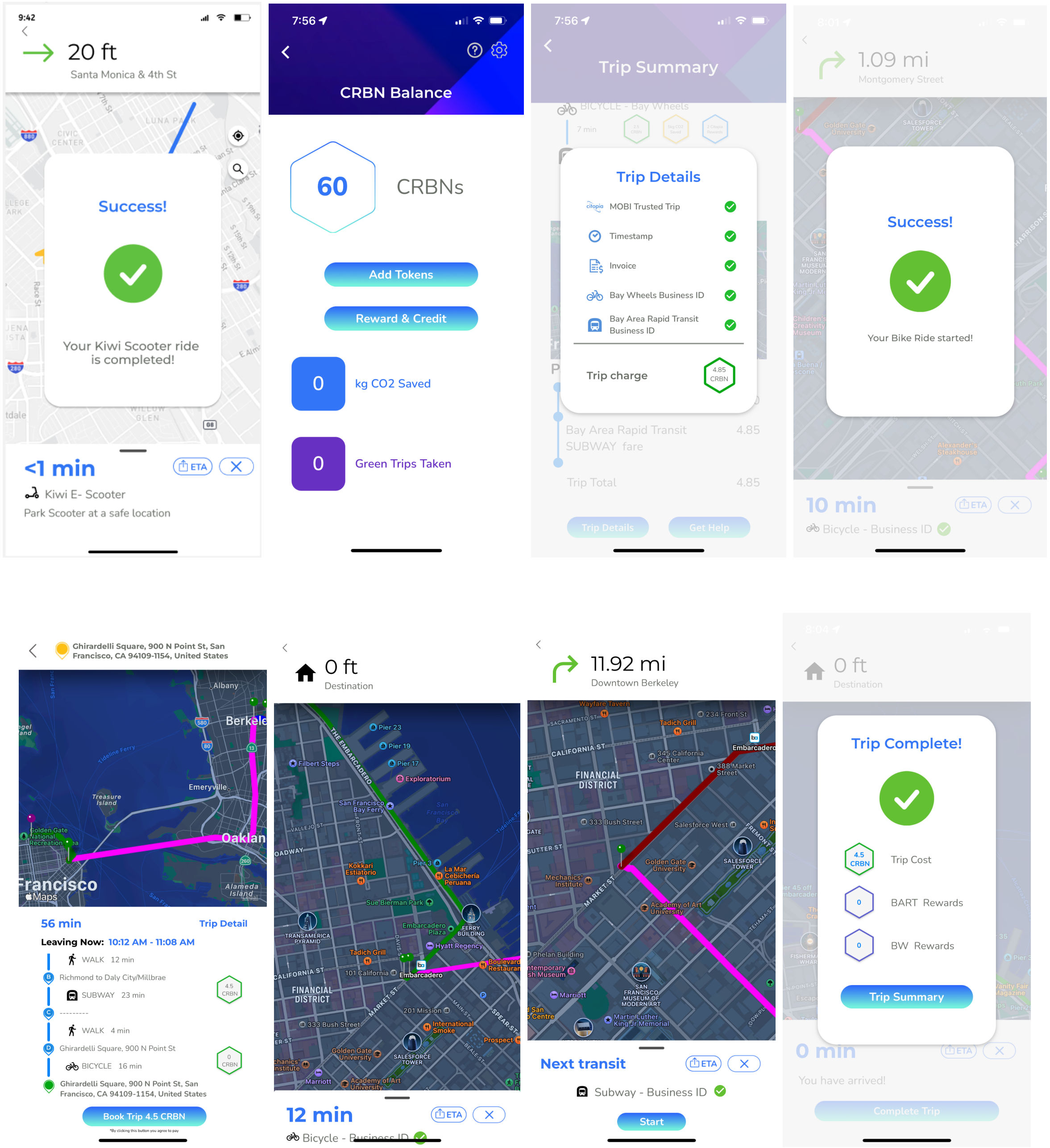


FIGURE 10 UI screenshots.

Appendix C: VC Issuance/Creation

The following are screenshots from VC issuance for Reservation Agreement and Invoice VCs in trip execution flow.

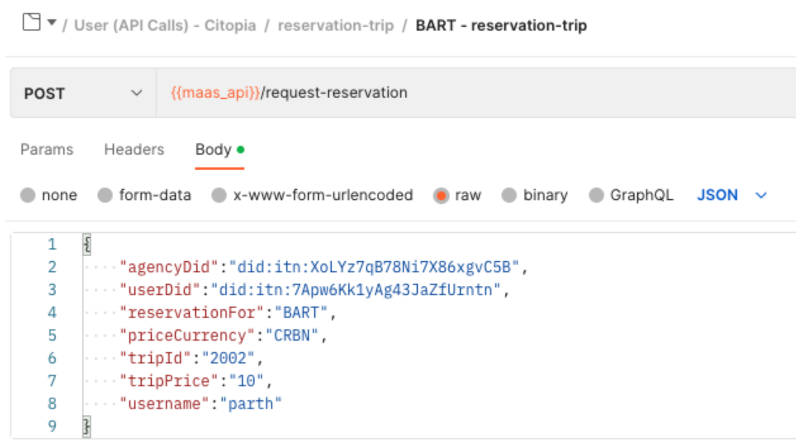


FIGURE 11 Reservation Agreement VC - API call request with body parameters (postpay flow).

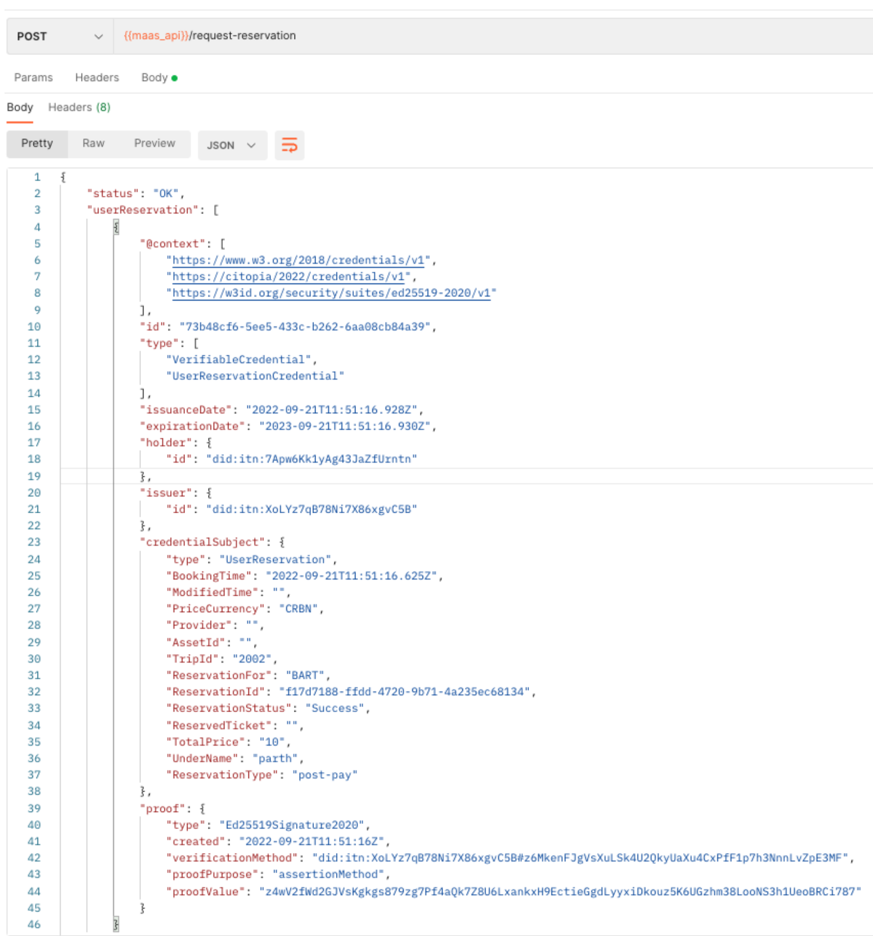


FIGURE 12 Reservation Agreement VC - API call response (postpay flow).

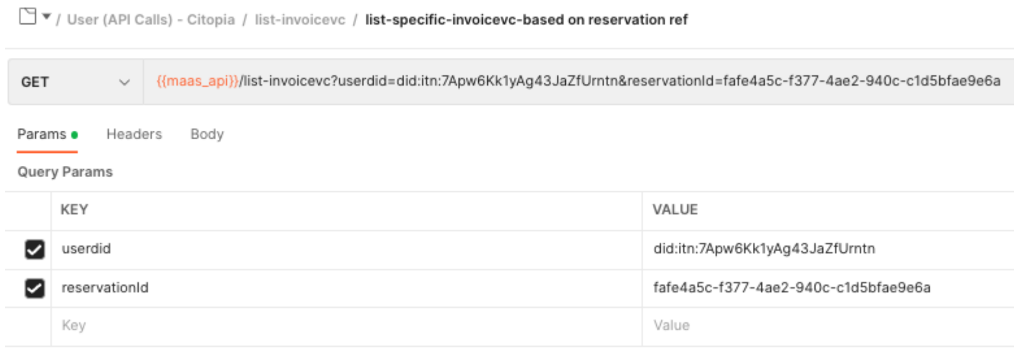


FIGURE 13 An Invoice VC - API call request.

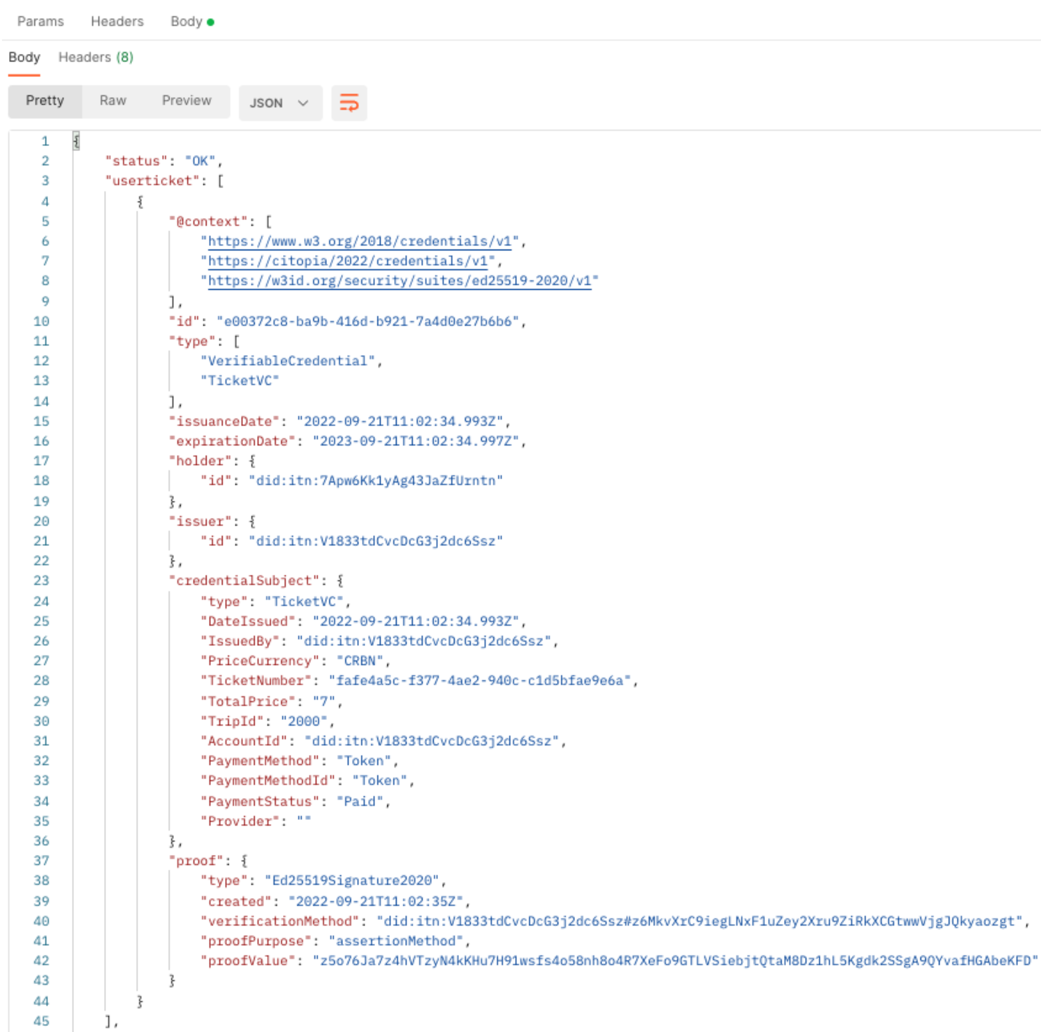


FIGURE 14 An Invoice VC - API call response.

Appendix D: Test Coverage Report for Task 5

The following is the Test Coverage Report Console Log from the backend testing in Task 5.

```
citopia_test  |  db.ts                              |   100 |   100 |   100 |   100 |
citopia_test  | otpserverfactory/src/utils          |    75 |    50 |   100 | 72.72 |
citopia_test  |  execute-utils.ts                   |    75 |    50 |   100 | 72.72 | 15-19
citopia_test  | serviceProvider/src                 |    50 | 70.58 |    20 | 46.66 |
citopia_test  |  config.ts                          |   100 | 81.81 |   100 |   100 | 4,25-26,34-37,44
citopia_test  |  response-messages.ts               |   100 |   100 |   100 |   100 |
citopia_test  |  utils.ts                           |  47.5 |     0 |    20 | 42.85 | 11-22,52-65,69-71
citopia_test  | serviceProvider/src/app             |   100 |   100 |   100 |   100 |
citopia_test  |  app-data-source.ts                 |   100 |   100 |   100 |   100 |
citopia_test  | serviceProvider/src/controller      | 76.24 | 86.36 |  90.9 | 74.84 |
citopia_test  |  bank-details.controller.ts         | 83.33 |     0 |   100 | 81.57 | 35,44-46,57-59
citopia_test  |  gtfs-info.controllert.ts           | 73.17 |   100 |   100 | 69.44 | 16-17,29-31,45-47,59-61
citopia_test  |  gtfs-on-board-status.controller.ts | 84.21 |   100 |   100 | 81.25 | 22-24
citopia_test  |  real-time-data.controller.ts       | 69.64 |   100 | 66.66 | 67.34 | 21-24,39-41,65-74,83,91-93
citopia_test  |  trip-payment-status.controller.ts  | 78.26 | 66.66 |   100 |    85 | 24-26
citopia_test  | serviceProvider/src/entity          |   100 |   100 |   100 |   100 |
citopia_test  |  bank-details.entity.ts             |   100 |   100 |   100 |   100 |
citopia_test  |  gtfs-info.entity.ts                |   100 |   100 |   100 |   100 |
citopia_test  |  index.ts                           |   100 |   100 |   100 |   100 |
citopia_test  |  gtfs-info.respository.ts           |   100 |   100 |   100 |   100 |
citopia_test  |  index.ts                           |   100 |   100 |   100 |   100 |
citopia_test  |  realtime-data.repository.ts        |   100 |   100 |   100 |   100 |
citopia_test  |  service-provider.repository.ts     |   100 |   100 |   100 |   100 |
citopia_test  |  trip-commencement.repository.ts    |   100 |   100 |   100 |   100 |
citopia_test  |  trip-completion.repository.ts      |   100 |   100 |   100 |   100 |
citopia_test  |  trip-payment-status.repository.ts  |   100 |   100 |   100 |   100 |
citopia_test  |  user-reservation.repository.ts     |   100 |   100 |   100 |   100 |
citopia_test  | serviceProvider/test                |   100 |   100 |   100 |   100 |
citopia_test  |  realtime-data.repository.ts        |   100 |   100 |   100 |   100 |
citopia_test  |  service-provider.repository.ts     |   100 |   100 |   100 |   100 |
citopia_test  |  trip-commencement.repository.ts    |   100 |   100 |   100 |   100 |
citopia_test  |  trip-completion.repository.ts      |   100 |   100 |   100 |   100 |
citopia_test  |  trip-payment-status.repository.ts  |   100 |   100 |   100 |   100 |
citopia_test  |  user-reservation.repository.ts     |   100 |   100 |   100 |   100 |
citopia_test  | serviceProvider/test                |   100 |   100 |   100 |   100 |
citopia_test  |  common.ts                          |   100 |   100 |   100 |   100 |
```

```
citopia_test  |  gtfsfile-infos.repository.ts          |   100 |   100 |   100 |   100 |
citopia_test  |  index.ts                              |   100 |   100 |   100 |   100 |
citopia_test  |  service-provider.repository.ts        |   100 |   100 |   100 |   100 |
citopia_test  |  tokens.repository.ts                  |   100 |   100 |   100 |   100 |
citopia_test  |  trip-payment.repository.ts            |   100 |   100 |   100 |   100 |
citopia_test  |  user-token.repository.ts              |   100 |   100 |   100 |   100 |
citopia_test  |  user.repository.ts                    |   100 |   100 |   100 |   100 |
citopia_test  | citopia/test                           |   100 |   100 |   100 |   100 |
citopia_test  |  common.ts                             |   100 |   100 |   100 |   100 |
citopia_test  | otpapigateway/src                      |   100 | 95.45 |   100 |   100 |
citopia_test  |  config.ts                             |   100 | 95.45 |   100 |   100 | 11
citopia_test  |  response-messages.ts                  |   100 |   100 |   100 |   100 |
citopia_test  | otpapigateway/src/controller           | 81.81 |   100 |   100 | 80.64 |
citopia_test  |  otp-find-instance.controller.ts       | 83.33 |   100 |   100 | 81.25 | 20-22
citopia_test  |  otp-instances-tracker.controller.ts   |    80 |   100 |   100 |    80 | 23-25
citopia_test  | otpapigateway/src/db                   |   100 |   100 |   100 |   100 |
citopia_test  |  db.ts                                 |   100 |   100 |   100 |   100 |
citopia_test  |  model.ts                              |   100 |   100 |   100 |   100 |
citopia_test  | otpapigateway/src/utils                |  90.9 |    50 |   100 |   100 |
citopia_test  |  execute-util.ts                       |  90.9 |    50 |   100 |   100 | 14
citopia_test  | otpserverfactory/src                   |   100 | 96.42 |   100 |   100 |
citopia_test  |  config.ts                             |   100 | 96.42 |   100 |   100 | 8
citopia_test  |  response-messages.ts                  |   100 |   100 |   100 |   100 |
citopia_test  | otpserverfactory/src/controller        | 73.33 |   100 |    50 | 76.92 |
citopia_test  |  available-region.controller.ts        | 73.33 |   100 |    50 | 76.92 | 19-21
citopia_test  | otpserverfactory/src/db                |   100 |   100 |   100 |   100 |
```

FIGURE 15 Test Coverage Report console log from backend testing.

```
citopia_test     | -------------------------------------|---------|----------|---------|---------|---------------------------
citopia_test     |   trip-payment-status.repository.ts   |     100 |      100 |     100 |     100 |
citopia_test     |   user-reservation.repository.ts      |     100 |      100 |     100 |     100 |
citopia_test     |  serviceProvider/test                 |     100 |      100 |     100 |     100 |
citopia_test     |   common.ts                           |     100 |      100 |     100 |     100 |
citopia_test     | -------------------------------------|---------|----------|---------|---------|---------------------------
citopia_test     |
citopia_test     | Test Suites: 14 passed, 14 total
citopia_test     | Tests:       42 passed, 42 total
citopia_test     | Snapshots:   0 total
citopia_test     |
citopia_test     | Test Suites: 14 passed, 14 total
citopia_test     | Tests:       42 passed, 42 total
citopia_test     | Snapshots:   0 total
citopia_test     | Time:        29.392 s
citopia_test     | Ran all test suites in 4 projects.
citopia_test     | Done in 31.61s.
citopia_test exited with code 0
```

```
citopia_test     | PASS serviceProvider packages/serviceProvider/test/controllers/trip-payment.controller.test.ts
citopia_test     | -------------------------------------|---------|----------|---------|---------|---------------------------
citopia_test     | File                                 | % Stmts | % Branch | % Funcs | % Lines | Uncovered Line #s
citopia_test     | -------------------------------------|---------|----------|---------|---------|---------------------------
citopia_test     | All files                            |   84.09 |    78.86 |   77.04 |   83.87 |
citopia_test     |  citopia/src                         |   71.13 |    52.38 |   52.94 |   67.56 |
citopia_test     |   mailer-helper.ts                   |   63.63 |        0 |     100 |   66.66 | 24-26
citopia_test     |   response-messages.ts               |     100 |      100 |     100 |     100 |
citopia_test     |   token.ts                           |   95.45 |      100 |   77.77 |   96.77 | 62
citopia_test     |   utils.ts                           |   46.34 |        0 |   14.28 |   39.39 | 13-24,46-48,52-72,76-77
citopia_test     |  citopia/src/app                     |     100 |     77.5 |     100 |     100 |
citopia_test     |   app-data-source.ts                 |     100 |      100 |     100 |     100 |
citopia_test     |   config.ts                          |     100 |     77.5 |     100 |     100 | 3-4,14-15,26-32
citopia_test     |  citopia/src/controller              |   93.51 |      100 |     100 |    92.5 |
citopia_test     |   creditcard-verification.controller.ts |  100 |      100 |     100 |     100 |
citopia_test     |   gtfsfile-info.controller.ts        |     100 |      100 |     100 |     100 |
citopia_test     |   service-provider.controller.ts     |   91.89 |      100 |     100 |      90 | 54-56
citopia_test     |   token-functions.controller.ts      |      88 |      100 |     100 |   86.36 | 31-33,65-67
citopia_test     |   update-serviceprovider.controller.ts |   100 |      100 |     100 |     100 |
citopia_test     |   user.controller.ts                 | [illegible] |  100 |     100 | [illegible] | 45-47
citopia_test     |  citopia/src/entity                  |     100 |      100 |     100 |     100 |
citopia_test     |   gtfsfile-infos.entity.ts           |     100 |      100 |     100 |     100 |
citopia_test     |   index.ts                           |     100 |      100 |     100 |     100 |
citopia_test     |   service-provider.entity.ts         |     100 |      100 |     100 |     100 |
citopia_test     |   tokens.entity.ts                   |     100 |      100 |     100 |     100 |
citopia_test     |   trip-payment-info.entity.ts        |     100 |      100 |     100 |     100 |
citopia_test     |   user-token-balance.entity.ts       |     100 |      100 |     100 |     100 |
citopia_test     |   user.entity.ts                     |     100 |      100 |     100 |     100 |
citopia_test     |  citopia/src/repository              |     100 |      100 |     100 |     100 |
```

FIGURE 16 Test Coverage Report console log from backend testing.

Appendix E: Performance Testing Results and K8 Configuration

TABLE 7 Detailed Performance Testing Results for Rider Trip Execution Flow

Flow Name	Total Number of Flows	Total Number of Flows - Passed	Total Number of Flows - Failed	Time to Complete flow executions	Number of flow executions per second	Average Request Duration	Min Request Duration	Med Request Duration	Max Request Duration	P(90) Request Duration	P(95) Request Duration	Comments
request-reservation	36001	36000	1	4m	150	793.91ms	505.48ms	733.39ms	2.26s	1.04s	1.24s	Request Duration timings are with expected_response: true - parameter
request-reservation	118734	118684	50	11m19s	833.33	30.6s	608.32ms	28.65s	1m59s	55.19s	1m5s	Request Duration timings are with expected_response: true - parameter
request-reservation	115568	115444	124	10m54s	833.33	31.25s	729.45ms	29.29s	1m59s	49.61s	56.61s	Request Duration timings are with expected_response: true - parameter
request-reservation	95430	95186	244	9m15s	833.33	40.76s	442.25ms	38.56s	1m59s	1m13s	1m24s	Request Duration timings are with expected_response: true - parameter
Average	**109910.67**	**109771.33**	**139.33**	**10m29s**	**833.33**	**34.20s**	**593.34ms**	**32.17s**	**1m59s**	**59.27s**	**1m8s**	Average of above three iterations -- highlighted with color in column A
trip-commencement	105397	105395	2	6m01s	833.33	2.21s	300.53ms	492.71ms	1m35s	4.99s	14.75s	Request Duration timings are with expected_response: true - parameter
trip-commencement	93559	93362	197	10m	833.33	45.78s	708.43ms	44.72s	3m22s	1m9s	1m24s	Request Duration timings are with expected_response: true - parameter
trip-commencement	92336	92031	305	10m	833.33	46.87s	740.85ms	45.76s	2m44s	1m8s	1m18s	Request Duration timings are with expected_response: true - parameter
trip-commencement	103243	102826	417	11m	833.33	46.9s	834.25ms	46.08s	3m2s	1m	1m6s	Request Duration timings are with expected_response: true - parameter
Average	**96379.33**	**96073**	**306.33**	**10.34m**	**833.33**	**46.51s**	**761.17ms**	**45.52s**	**3m26s**	**1m5s**	**1m16s**	Average of above three iterations -- highlighted with color in column A
trip-completion	167939	167857	82	4m07s	833.33	6.7s	298.01ms	331.4ms	2m29s	466.34ms	1m10s	Request Duration timings are with expected_response: true - parameter
trip-completion	99554	99281	273	14m	238.1	1m5s	627.7ms	1m6s	3m54s	1m33s	1m41s	Request Duration timings are with expected_response: true - parameter
trip-completion	93974	93252	722	14m	238.1	1m9s	1.44s	1m8s	5m3s	1m31s	1m40s	Request Duration timings are with expected_response: true - parameter
Average	**96764**	**96266.5**	**497.5**	**14m**	**238.1**	**1m7s**	**1.03s**	**1m7s**	**4m28s**	**1m32s**	**1m40.5s**	Average of above two iterations -- highlighted with color in column A

TABLE 8 Detailed Performance Testing Results for End-to-End Testing

Flow Name	Total Number of Flows	Total Number of Flows - Passed	Total Number of Flows - Failed	Time to Complete flow executions	Number of flow executions per second	Average Request Duration	Min Request Duration	Med Request Duration	Max Request Duration	P(90) Request Duration	P(95) Request Duration	Comments
End-to-End	33976	33777	199	23m59s	83.33	4m7s	7s	4m3s	9m58s	6m18s	7m8s	Request Duration timings are with expected_response: true - parameter
End-to-End	36910	27025	9885	21m	250	3m16s	4.5s	3m19s	7m37s	4m44s	5m9s	Request Duration timings are with expected_response: true - parameter
End-to-End	41885	29545	12340	21m	500	1m28s	1.82s	1m32s	7m28s	2m44s	3m31s	Request Duration timings are with expected_response: true - parameter
Average	**37590.33**	**30115.67**	**7474.67**	**21m59s**	**277.77**	**2m57s**	**4.1s**	**2m58s**	**8m21s**	**4m35s**	**5m16s**	Average of above three iterations -- highlighted with color in column A

K8s Configuration for Performance Testing:

ITN
- Region: us-west-1
- 1x core-services pod on t3.large
- 1x did-resolver pod on t3.large

Citopia-MaaS
- Region: us-west-2
- 8x citopia pod on **m5.4xlarge** instance
- 1x citopia postgres pod on **t3.2xlarge** instance
- 8x service-provider pod on **m5.4xlarge** instance
- 1x service-provider pod postgres pod on **t3.xlarge** instance
- 6x nginx pod on **t3.xlarge** instance

Appendix F: Citopia Self-Sovereign Digital Twin™ (SSDT™)

All Web3 transactions on Citopia MaaS take place through SSDTs™. A digital twin is a virtual representation that serves as the real-time digital counterpart of a physical object or process. With the help of IoT, machine learning, and AI, digital twins are used across many industries to model and simulate processes and systems as well as to help decision-making. Although the concept has been more widely used in manufacturing, in theory; anything can have a digital twin, even a human or a vehicle. For instance, the digital twin of a vehicle can be a digital replica of a vehicle's physical attributes and contain life cycle information from production to end of life. Citopia presents a novel application of digital twins, which we define as the Self-Sovereign Digital Twin™. Essentially, an SSDT™ is a digital twin of an entity or system that can autonomously participate in trusted Web3 transactions by issuing World Wide Web Consortium (W3C) compliant Verifiable Credentials (VCs) and Verifiable Presentations (VPs). This is enabled by linking the SSDTs™ to W3C compliant Decentralized Identifiers (DIDs), which are anchored in a federated network (Integrated Trust Network, or ITN). Figure 17 shows the high-level architecture of a Citopia SSDT.

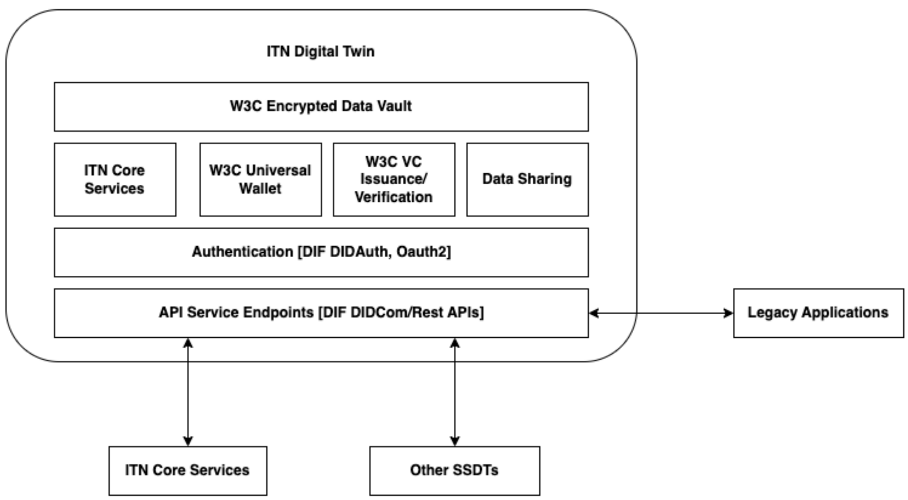


FIGURE 17 Citopia SSDT architecture diagram.

In Citopia, any entity — whether it be an individual, an organization, an object such as a vehicle etc. — can have an SSDT™ as a necessary gateway to participate in the federated Web3 marketplace. The SSDT™ acts as a universal translator and an encrypted locked data vault that enables the owner of the SSDT™ to communicate with virtual agents of other entities in Web3 transactions. This way it securely stores a combination of static and real-time data to log an entity's journey throughout its lifetime. Data is stored in SSDTs™ by using DIDs, VCs, and other advanced cryptographic methods. This gives users control over who sees their data and how that data is used. Zero-Knowledge Proof cryptography is used for data privacy compliance. An important differentiator of Citopia W3C standards-based SSDT™ is that the owner has full control of their SSDT™ and can use it outside of Citopia, unlike centralized digital twins, which rely on the issuing platform or organization to operate. This enables service providers on Citopia MaaS to easily access multiple customer segments (riders) on one marketplace without vendor or technology lock-in. Citopia enables the network effect with SSDTs™ by connecting riders directly to the service providers for usage-based business automation, which facilitates more riders to access seamless transit.

Appendix G: Use of Blockchain Via ITN in Citopia MaaS

Citopia MaaS uses the ITN as the trust layer for verification of identity and/or transaction. ITN is a protocol-agnostic, blockchain-based trusted identity core-services network. It is a permissioned network of nodes operated by ITN operators. A permissioned network is a distributed ledger which is not publicly accessible; it can only be accessed by trusted authorized users.

Citopia connects to the ITN core services to enable multiparty business automation via World Wide Web Consortium (W3C) Decentralized Identifiers (DIDs). Through the distributed network of nodes, blockchain enables the ITN to be trustless, such that entities do not need to know each other to establish a trust relationship to execute transactions. Although blockchain technology is not yet mature, cost-effective, and fast enough for many enterprise applications, Citopia MaaS pilot presents a novel enterprise use case of blockchain by using the blockchain-based ITN to register the DIDs of participants in a secure, decentralized, and privacy-preserving manner. It should be noted that Citopia itself is not a blockchain-based network, only the ITN uses blockchain. The sole use of blockchain through the ITN in Citopia MaaS is as a ledger for DID registry (for generating and verifying DIDs) and verifying transactions. Blockchain is essentially used to anchor DIDs and does not store any data relevant to Citopia MaaS.

Blockchain also enables transactions to be tamper-evident by combining cryptographic digital signatures with the data integrity properties of the utilized cryptographic keys. Another advantage of using blockchain is redundancy, meaning that a copy of every DID is distributed among the blockchain nodes; as a result, if one node is down, then the information is not lost as the other nodes have a copy. The distributed nature of the blockchain adds a layer of decentralization as one single entity does not have control over the network. This also eliminates the need for middlemen or centralized platforms, thereby increasing efficiency. ITN core services including identity, assurance, and governance allow for application interoperability and multiparty data sharing so that participants can execute trusted Web3 transactions on Citopia.

FIGURE 18 Technology Stack overview.

Appendix H: Detailed Glossary of Terms

Application Programming Interface (API): An application programming interface, or API, enables companies to open up their applications' data and functionality to external third-party developers, business partners, and internal departments within their companies. This allows services and products to communicate with each other and leverage each other's data and functionality through a documented interface. (Source: https://www.ibm.com/topics/api)

AWS EKS: Amazon EKS is a managed Kubernetes service to run Kubernetes in the AWS cloud and on-premises data centers. In the cloud, Amazon EKS automatically manages the availability and scalability of the Kubernetes control plane nodes responsible for scheduling containers, managing application availability, storing cluster data, and other key tasks. With Amazon EKS, you can take advantage of all the performance, scale, reliability, and availability of AWS infrastructure, as well as integrations with AWS networking and security services. On-premises, EKS provides a consistent, fully-supported Kubernetes solution with integrated tooling and simple deployment to AWS Outposts, virtual machines, or bare metal servers. (Source: https://aws.amazon.com/eks/)

Citopia Controllers: Citopia Controller is responsible for controlling the way that a user interacts with an application. A controller contains the flow control logic. A controller determines what response to send back to a user based on the user interaction with the application (either a request from the web browser or mobile app).

Couchdb: CouchDB is a database that completely embraces the web. Store your data with JSON documents. Access your documents with the web browser, via HTTP. Query, combine, and transform documents with JavaScript. CouchDB works well with modern web and mobile apps. You can distribute data efficiently using CouchDB's incremental replication. CouchDB supports master-master setups with automatic conflict detection. (Source: https://docs.couchdb.org/en/stable/intro/index.html)

Data Processor (as in the General Data Protection Regulation): A data processor is a natural or legal person, public authority, agency, or other body which processes personal data on behalf of the controller. (Art. 4 no. 8 GDPR)

Decentralized Identifier (DID): W3C Decentralized Identifier (DID) represents a globally unique identifier that can be resolved to a DID Document, or de-referenced on a specific distributed ledger network, much like a URL on the Internet. (Source: https://dlt.mobi/glossary/)

did:elem: Element is an implementation of the Sidetree Protocol that uses the Ethereum blockchain as the ledger layer and IPFS as the content-addressable storage layer. (Source: https://github.com/transmute-industries/sidetree.js)

DIF Sidetree Standard: Sidetree is a protocol for creating scalable Decentralized Identifier networks that can run atop any existing decentralized anchoring system (e.g. Bitcoin, Ethereum, distributed ledgers, witness-based approaches) and be as open, public, and permissionless as the underlying anchoring systems they utilize. The protocol allows users to create globally unique, user-controlled identifiers and manage their associated PKI metadata, all without the need for centralized authorities or trusted third parties. The syntax of the identifier and accompanying data model used by the protocol is conformant with the W3C Decentralized

Identifiers specification. Implementations of the protocol can be codified as their own distinct DID Methods and registered in the W3C DID Method Registry. (Source: https://identity.foundation/sidetree/spec/)

Digital Twin: A Digital Twin is a digital representation of a real-world entity or system. The implementation of a digital twin is an encapsulated software object or model that mirrors a unique physical object, process, organization, person, or other abstraction. (Source: https://dlt.mobi/glossary/)

Digital Wallet: A software application that runs, for example, on a rider's mobile phone, a laptop, server, or on a vehicle's hardware. A digital wallet contains private keys that can be used to encrypt messages, prove control over a DID, permissions or property on a distributed ledger, information about (private) peer-to-peer connections with and references of other entities in the digital identity ecosystem. (Source: https://dlt.mobi/glossary/)

Encrypted Data Vault (EDV): An implementation of the W3C/DIF Encrypted Data Vault for document storage currently utilizing CouchDB with Leader-Leader replication (Source: ITN Technical Requirement and Architecture Paper ITN0002/TS/2021 Version 1.0)

General Bikeshare Feed Specification (GBFS): GBFS is a real-time data specification that describes the current status of a mobility system. GBFS data is used by trip planning and Mobility as a Service (MaaS) applications, to provide information travelers need to discover and use shared mobility. Public GBFS APIs enable the integration of shared mobility services with public transportation, allowing users to make first-mile, last-mile connections. (Source: https://gbfs.mobilitydata.org/)

General Transit Feed Specification (GTFS): The General Transit Feed Specification (GTFS) is a data specification that allows public transit agencies to publish their transit data in a format that can be consumed by a wide variety of software applications. GTFS is split into a schedule component that contains schedule, fare, and geographic transit information and a real-time component that contains arrival predictions, vehicle positions and service advisories. (Source: https://gtfs.org/)

Grafana k6: Grafana k6 is an open-source load testing tool that makes performance testing easy and productive for engineering teams. k6 is free, developer-centric, and extensible. Using k6, you can test the reliability and performance of your systems and catch performance regressions and problems earlier. k6 will help you to build resilient and performant applications that scale. (Source: k6.io)

JavaScript: JavaScript, often abbreviated as JS, is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS. As of 2022, 98% of websites use JavaScript on the client side for webpage behavior, often incorporating third-party libraries. (Source: https://en.wikipedia.org/wiki/JavaScript)

K8 Cluster: Kubernetes is a portable, extensible, open source platform for managing containerized workloads and services that facilitates both declarative configuration and automation. It has a large, rapidly growing ecosystem. Kubernetes services, support, and tools are widely available. (Source: https://kubernetes.io/docs/concepts/overview/)

Mapbox: Mapbox is an American provider of custom online maps for websites and applications. Mapbox is the creator of, or a significant contributor to, some open source mapping libraries and applications, including the Mapbox GL-JS JavaScript library. (Source: https://en.wikipedia.org/wiki/Mapbox)

Mapquest: MapQuest is an American free online web mapping service. (Source: https://en.wikipedia.org/wiki/MapQuest)

Merkle Proof: Merkle proofs are established by hashing a hash's corresponding hash together and climbing up the tree until you obtain the root hash which is or can be publicly known. (Source: https://en.wikipedia.org/wiki/Merkle_tree, https://computersciencewiki.org/index.php/Merkle_proof)

Merkle Tree: In cryptography and computer science, a hash tree or Merkle tree is a tree in which every "leaf" (node) is labeled with the cryptographic hash of a data block, and every node that is not a leaf (called a branch, inner node, or inode) is labeled with the cryptographic hash of the labels of its child nodes. A hash tree allows efficient and secure verification of the contents of a large data structure. (Source: https://en.wikipedia.org/wiki/Merkle_tree)

MOBI Trusted Trip Verifiable Credential (MTT VC): An abstract credential consisting of a subject's information related to location and other attributes structured using W3C Verifiable Credential Data Model 1.0 standards. (Source: https://dlt.mobi/glossary/)

Open Trip Planner (OTP): OTP is a family of open source software projects that provide passenger information and transportation network analysis services. The core server-side Java component finds itineraries combining transit, pedestrian, bicycle, and car segments through networks built from widely available, open standard OpenStreetMap and GTFS data. (Source: https://www.opentripplanner.org/)

Personally Identifiable Information (PII) – PII is any information: (1) that identifies or can be used to identify, contact, or locate the person to whom such information pertains, (2) from which identification or contact information of an individual can be derived, or (3) that is or might be directly or indirectly linked to a natural person. (Source: ISO/IEC 29100:-1)

React Native: React Native is a popular JavaScript-based mobile app framework that allows you to build natively-rendered mobile apps for iOS and Android. The framework lets developers create an application for various platforms by using the same codebase. (Source: https://www.netguru.com/glossary/react-native)

Road Usage Charge (RUC): Road usage charging (RUC), sometimes referred to as vehicle miles traveled (VMT) fees or mileage-based user fees (MBUF), is a policy whereby motorists pay for use of the roadway network based on distance traveled. (Source: https://www.ibtta.org/road-usage-charge-ruc)

Self-Sovereign Digital Twin (SSDT): A Self-Sovereign Digital Twin™ (SSDT) is a digital twin which is anchored in a decentralized trust network using W3C's DIDs) Standard. By using the SSDT and W3C's VC Standard, the controller of the SSDT can participate as an autonomous economic agent in trusted transactions through issuing VCs and Verifiable Presentations (VPs). Citopia SSDT is a universal translator and encrypted lock data vault that no individual or organization has access to other than the owner/controller. The SSDT stores a combination of static and real-time data to log an entity's journey throughout its lifetime. (Source: https://dlt.mobi/glossary/)

User Acceptance Testing (UAT): User Acceptance Testing (UAT), sometimes called beta testing or end-user testing, is a phase of software development in which the software is tested in the "real world" by the intended audience or business representative. This type of testing is not intended to be menu-driven, but rather to be performed by business users to verify that the application will meet the needs of the end-user, with scenarios and data representative of actual usage in the field. (Source: https://uit.stanford.edu/pmo/UAT)

User Interface (UI): The user interface (UI) is the point of human-computer interaction and communication in a device. This can include display screens, keyboards, a mouse and the appearance of a desktop. It is also the way through which a user interacts with an application or a website. (Source: https://www.techtarget.com/searchapparchitecture/definition/user-interface-UI)

Verifiable Credential (VC): The W3C Verifiable Credentials Standard defines Verifiable Credentials as "a part of our daily lives; driver's licenses are used to assert that we are capable of operating a motor vehicle, university degrees can be used to assert our level of education, and government-issued passports enable us to travel between countries. This specification provides a mechanism to express these sorts of credentials on the Web in a way that is cryptographically secure, privacy-respecting, and machine-verifiable. (Source: https://dlt.mobi/glossary/)

Verifiable Presentation (VP): In many interactions, it is necessary for an entity (the prover) to convince the counterparty (the verifier) of the validity of statements regarding some of their attributes, so-called claims. A verifiable presentation consists of the following sequential steps: proof request, proof creation, proof transmission, and proof verification. (Source: https://dlt.mobi/glossary/)

Web3: Web3 is an idea for a new iteration of the World Wide Web which incorporates concepts such as decentralization, blockchain technologies, and token-based economics. Some technologists and journalists have contrasted it with Web 2.0, wherein they say data and content are centralized in a small group of companies sometimes referred to as "Big Tech". The term "Web3" was coined in 2014 by Ethereum co-founder Gavin Wood, and the idea gained interest in 2021 from cryptocurrency enthusiasts, large technology companies, and venture capital firms. (Source: https://en.wikipedia.org/wiki/Web3)

Zero-Knowledge Proof (ZKP): In cryptography, a zero-knowledge proof (ZKP) is a method by which one entity (the prover, assumed to be computationally powerful) can prove to another party (the verifier, assumed to be computationally restricted) that a particular statement is true without revealing any further information. (Source: https://dlt.mobi/glossary/)