UPPER

TES_10: To incentivise the use of PT in combination with active modes

Description of the measure and main outcomes expected

TES_10 aims to incentivize the use of Public Transportation (PT) in combination with active modes such as walking, bicycling, and e-scootering through a user-friendly application. The app will track and validate users' trips using smartphone sensors to accurately identify transport modes. Users will accumulate points based on the distance travelled, with extra points awarded for incorporating multiple modes of transport. These points can be redeemed for coupons that provide discounts or gifts from registered businesses. The expected outcomes include increased adoption of public transportation, enhanced integration of active modes, reduced traffic congestion, lower carbon emissions, and improved public health through increased physical activity. Additionally, local businesses may benefit from increased customer engagement.

Preparation of the measure

Survey

For the purpose of this measure, an online survey was carried out. The web questionnaire was disseminated through local websites to ensure broad participation. A total of 301 responses were collected in March 2024. The aim of the survey was to collect information regarding citizens' willingness to use the application developed as well as their preferences on the incentives. The questionnaire was divided into four sections: a) demographic and socioeconomic characteristics; b) existing mobility preferences; c) familiarity with technology; d) willingness to use the app and preferences.

The aim of this document is not to present the results in detail, however it is considered important to include some key results that verify the potential of the application developed:

- Of the participants who mainly use a car for their commute, about 80% are willing to walk more than 15' to reach their final destination.
- Of the participants who mainly use a car for their commute, about 45% are willing to travel by bicycle/e-scooter for more than 15'.
- Of the participants who mainly use other modes of transport (other than car), about 85% are willing to walk more than 15' to reach their final destination.
- Of the participants who mainly use other modes of transport (other than car), about 60% are willing to travel by bicycle/e-scooter for more than 15'.

It may be concluded that citizens of Thessaloniki are in general eager to use active modes, with walking being more popular than bicycle/e-scooter. In addition, people that already use other transport modes than car, are more willing to use active modes.

Additionally, it was identified that the user would ideally like to be rewarded through coupons for super markets and for mobility services. Also, the users would prefer the provision of incentives for their personal "achievements" rather than being incentivized for competing between teams.

System architecture

The TES_10 high-level architecture consists of a mobile app, a backend (Node.js), and a backend for the validation algorithm (Python), as presented in Figure 6.

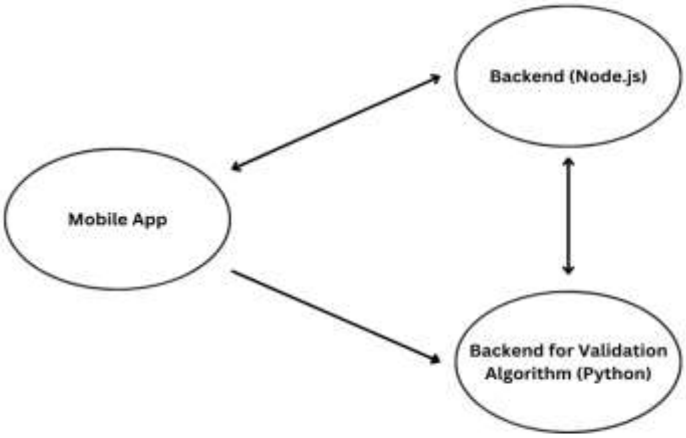

Figure 1: TES_10 high-level architecture

Mobile application: It is an Android[1] application which is used by users or businesses. Each role (user or business) can view a different dashboard based on the functionalities needed for this specific role.

Backend (Node.js[2]): It is the main backend component (written in Node.js) used by the application. For example, it is used for functionalities such as gifts and coupons that are not related to the validation algorithm. It is connected with a database (MySQL[3]) for storing data.

Backend for validation algorithm (Python): It is a backend component that serves the needs of the validation algorithm. Except for the validation algorithm code that is written in Python[4], [5]Flask is used in order to allow communication with the other components. This backend component uses its own database as well (MySQL).

The mobile app interacts with users, collecting sensor data and providing feedback. This data is sent to the backend (Node.js), which handles communication and data processing tasks. The communication between the mobile app and the Python backend is facilitated by Flask for the "Report Trip" feature. The "Report Trip" feature allows users to start and end their journey on the app, selecting their transportation mode with the trip being validated through smartphone sensor data and GPS. This feature calculates and updates the user's points based on the distance travelled and whether the trip was unimodal or multimodal. The backend (Node.js) interacts with the backend for the validation algorithm (Python), which performs the validation of transportation modes using Machine Learning (ML) models. The results are then sent back to the mobile app, providing users with real-time feedback and updates on their trips and points gained. This architecture ensures efficient data processing and user-friendly interaction.

Validation algorithm

The initial step in implementing the TES_10 application was the development of the Transportation Mode Detection (TMD) algorithm, which serves as its core. The methodology involved extensive research on open data sources for TMD, including desk research to identify suitable datasets and sensors, and a literature review to determine the appropriate methodologies for preprocessing and feature extraction. Key criteria for selecting the appropriate dataset included: 1) the types of sensor measurements necessary to generate the features required for training the algorithm, and 2) the presence of labels/targets in the data that meet the application's requirements. Additionally, it was

[1] Android: https://developer.android.com

[2] Node.js: https://nodejs.org

[3] MySQL: https://www.mysql.com

[4] Python: https://www.python.org

[5] Flask: https://flask.palletsprojects.com

essential to ensure that the dataset's license permitted its use for this application. Concurrently, research on preprocessing raw smartphone sensor data was conducted, focusing on methods critical for accurate TMD.

Dataset

The datasets considered for this project included the Collecty dataset[6], the Occitania Transport Media dataset (OCC-TMD)[7], and the TMD dataset[8]. These datasets provide labelled data for various transport modes, which is crucial for training and evaluating TMD algorithms. Additionally, a new data collection option was explored, where smartphone sensors would gather real-time data from users' journeys, including GPS and accelerometer readings. Finally, we selected the Collecty dataset as it best matched our application requirements. It contains comprehensive sensor data necessary for our analysis and covers the full range of transport modes, including the e-scooter, which is not present in the other two datasets and generally in the most datasets. A significant factor in choosing Collecty was that its license allowed us to use it. However, like the other datasets, it did not include GPS data.

Pre-trained TMD algorithm

In the preprocessing stage, we aimed to reduce the number of classes as much as possible to enhance the performance of the TMD algorithm. We combined e-scooter and bicycle into a single class called 'Micromobility' due to their similar behaviour patterns. This resulted in four classes for prediction:

Active (combining 'Walk' and 'Run')

Micromobility (combining 'Bike' and 'E-scooter')

Public Transportation (combining 'Bus', 'Tramway', and 'Train')

Car

The Collecty dataset was then balanced to ensure that each class was represented equally, followed by stratified sampling to maintain the distribution of classes in both training and testing datasets. A Min-Max scaler was applied to normalize the feature values, ensuring that all features would contribute equally to the model training process.

The preprocessing and feature extraction process in the application follows a structured approach based on best practices for transportation mode detection. After conducting a comprehensive literature review, we adopted a methodology similar to the one proposed by Ashqar, Hut Haifa I., et al. (2019)[9]. This research provided a robust framework for handling and processing sensor data for transport mode detection, which we adapted to our needs. First, raw sensor data from accelerometers and gyroscopes are collected and cleaned to remove any noise or irrelevant information. This data is then interpolated to create a uniform time grid, ensuring consistency of the intervals between data points. First differences and second differences are calculated to record the change in sensor readings from one point in time to another, capturing dynamic aspects of the motion, such as acceleration and deceleration, and information on jerky movements or continuous changes in acceleration. Basic statistical characteristics, such as mean, sum, maximum, minimum, variance, standard deviation and range, are calculated for each time window, initially at 1-second intervals. In addition, advanced features, such as signal variations, energy and spectral entropy, are extracted by transforming the data in the frequency domain using the Welch method. By extracting these features, the raw sensor data is transformed into a structured format that captures the essential characteristics needed for accurately identifying transportation modes. This processed and feature-rich data is then used to train the TMD algorithm, ensuring that it can effectively classify different types of travel based on the user's movements. We

[6] (Erdelić, Erdelić, & Caric, 2023)

[7] (Thibault, Verstaevel, Migeon, & Schettini, 2023)

[8] (Carpineti, Lomonaco, Bedogni, Felice, & Bononi, 2018)

[9] (Ashqar, Almannaa, Elhenawy, & House, 2019)

tested the performance of several ML models to determine the best. The results are presented in Table 12. The final pre-trained algorithm chosen for the application is the Random Forest (RF) model with a 5-second frequency. Frequency refers to the interval at which features are extracted and predictions are made. For instance, in the case of Long Short-Term Memory (LSTM), features are extracted every second, but the predictions are made using 30-second packets. The RF model at 5-second intervals was selected due to its high accuracy of 91%, which matches the highest accuracy observed among all models tested (also achieved by k-Nearest Neighbours at the same frequency). While k-Nearest Neighbours (kNN) also performed well, RF was preferred based on probability-based theory, which considers the robustness and generalization capabilities of RF in handling diverse datasets. Additionally, RF showed consistent high performance across different frequencies, further validating its reliability. Thus, the RF model at a 5-second frequency was determined to be the optimal choice for the TMD algorithm in the application, offering a robust and accurate solution for TMD.

Table 1: Comparison of model's accuracy on TMD task

Model	Frequency	Accuracy
Random Forest (RF)	1s, 5s, 10s	85%, 91%, 91%
k Nearest Neighbours (kNN)	1s, 5s, 10s	80%, 91%, 91%
Support Vector Machine (SVM)	1s, 5s, 10s	82%, 87%, 86%
Decision Tree (DT)	1s, 5s, 10s	70%, 81%, 84%
Long Short-Term Memory (LSTM)	30s	87%

Binary classifier

In order to enhance the accuracy of the TMD model, in particular with regard to the differentiation between PT and car trips, it is recommended to include a verification step. This verification could be done either: a) by matching the route with a database of public transportation routes or b) by developing an additional binary classifier to distinguish between car and PT modes (hierarchical classification or step-by-step classification).

To address the challenge of distinguishing between closely related modes such as public transportation and private car travel, we incorporate a specialized binary classifier. This additional layer enhances the algorithm's accuracy in differentiating these transportation modes, ensuring more precise validations. During the development process, we trained various binary classification algorithms and ultimately selected the Random Forest classifier due to its superior performance and reliability in this context. The RF model provided the highest accuracy and robustness in distinguishing between PT and car trips, making it the optimal choice for our application.

Bus trip validation

The "bus trip validation" step works by comparing the GPS data of a user's journey against predefined bus routes stored in a GeoJSON file. The process involves converting the GPS data into a geographic format and projecting it to match the coordinate system of the bus routes. The algorithm then calculates the distance between each GPS point and the bus routes, checking if a significant portion of the points fall within a specified tolerance distance from the route. By sorting the routes based on proximity to the journey's starting point, the algorithm efficiently determines whether the journey aligns with any bus route. If a sufficient percentage of GPS points match a bus route, the trip is validated as a bus journey. This validation ensures the accuracy and reliability of the reported transportation mode. In Figure 7, a validated bus trip is depicted. The continuous red line represents the user's route as recorded by GPS data. The blue points indicate the bus route data, showing how the user's trip matches the established bus routes, confirming the validation by the "bus trip validation" process.

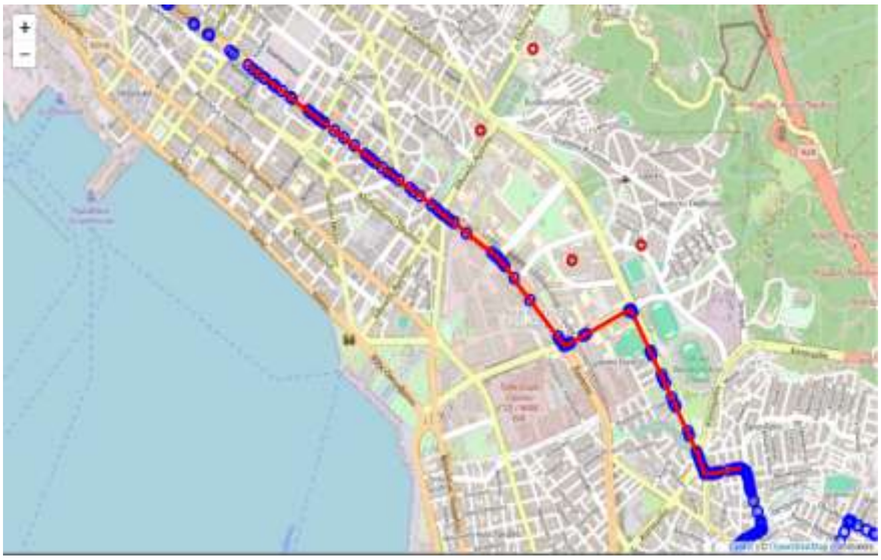

Figure 2: Validated bus trip visualisation

Verification process

The verification process ensures that the user-reported transportation mode matches the mode detected by the application. It involves cross-checking GPS data and sensor readings against predefined patterns for each transportation mode.

Prediction step

For each trip, the pre-trained TMD algorithm generates predictions every 5 seconds. Thus, a 5-minute trip yields 60 predictions. The final mode of transportation is determined by the majority class among these predictions. For example, if the predictions consist of 30 instances of bus, 5 instances of walking, 15 instances of car, and 10 instances of micromobility, the trip is ultimately classified as bus. This approach ensures that the most frequent mode of transport detected during the trip is selected as the final prediction.

Initial verification

For trips that are not car or bus trips such as walking, scooter or bicycle trips, the initial verification process relies solely on the smartphone sensor data and the prediction made by the pre-trained TMD algorithm. These modes do not require the additional verification step used for distinguishing between car and bus.

In case the user-reported mode is public transportation but the final predicted mode is car, an additional verification step is triggered. In this step, the binary classifier performs an extra check specifically on this route to validate the mode.

In the case where the final intended mode of operation matches the mode declared by the user (public transportation), then the additional verification step with the binary classifier is not activated and only the validation based on the bus routes is performed.

Binary classifier check

The binary classifier re-evaluates the sensor data to differentiate between bus and car travel. If the re-evaluation confirms that bus is the correct mode, the result is changed to true.

If the binary classifier still indicates the car as the mode used, the trip is flagged as failed.

Bus route check

For trips in which the bus is indicated as the mode used, an extra check is performed to validate the bus route. This check compares the trip data with standard bus routes.

The algorithm checks the closest bus routes first and returns true if at least one route matches the trip data.

Final verification

Each journey and mode of transport is examined individually. For example, a trip that includes {Walking, Bus, E-scooter, Bus} will have 4 validity checks.

For trips that include a bus, the extra checks discussed above are made.

If the bus route check confirms a match, the trip is verified as valid.

If no bus routes match, the trip is flagged as invalid, and the user is informed about the discrepancy.

Point system

The point system is designed to incentivize the use of public transportation in combination with active modes of transport, such as walking, cycling, and e-scooters. In developing this system, it was important to consider that points should motivate users to engage with the application and be fair for both users and partner companies. Points are awarded only for valid trips in which the transportation mode and distance travelled have been accurately verified. Users gain points based on the distance travelled, with an additional bonus for conducting multimodal trips. Specifically, for every 100 meters travelled, users gain one point for each type of supported trip. The app uses smartphone sensor data to accurately measure the distance covered, ensuring a fair and motivating point system.

Multimodal bonus points

To encourage the integration of multiple modes of transport, users receive bonus points for multimodal trips. A multimodal trip is defined as a single trip that includes two or more different modes of transportation, without taking walking into consideration. A fixed bonus of 20% is provided for multimodal trips. For example, if the user travels 2000 meters with a multimodal trip, the user will gain 20 points (for the distance travelled) plus 4 bonus points (for the multimodal trip). Here is the equation for calculating the total points:

$$P = D/100 * (1 + bonus * M)$$

Where: a) D is the distance travelled (in meters), b) bonus is a constant value, which is defined as 20% for each multimodal trip, c) M is a binary variable (M=1 for multimodal trips and M=0 for non-multimodal trips).

Table 13 summarizes examples and cases of user-reported trips and their verification results by the validation algorithm, along with the distance travelled and points gained. It shows whether each part of the trip was verified, the total distance travelled, the addition of a multimodal bonus and the final points awarded.

Table 2: Examples with reported trips and the corresponding results

Trip reported by the user	Trip verified by the validation algorithm	Distance travelled (km)	Multimodal bonus 20%	Points
[Walking, Bus, Walking]	[True, True, True]	[0.2, 2.4, 0.1]	No	27
[Bus, E Scooter]	[False, True]	[1.31, 1.51]	No	15
[Bicycle, Walking]	[True, True]	[0.82, 0.46]	No	12
[Bus, Escooter]	[True, True]	[1.75, 2.39]	Yes	48
[Walking]	[False]	[8.72]	No	0
[Bus, Walking, Bus]	[True, True, False]	[2.2, 0.4, 3.5]	No	26

Mobile application

The new mobile application supports two different dashboard views: the user dashboard and the business dashboard. So, it is used by two different roles. In Figure 8a, the login page is presented. After a successful login, the user views the dashboard presented in Figure 8b, which includes the following pages: “My trips”, “Report trip”, “Check points - Claim gift” and “My coupons”. The business dashboard consists of the pages “Gifts” and “Scan coupons” (Figure 8c).

Figure 3: a) Login page, b) User dashboard, c) Business dashboard.

<u>User dashboard</u>

On the “My trips” page (Figure 9a), the user can view his/her personal trips, along with the date and time, the status and the points gained for each trip. The green background colour indicates that the trip is completed and the user has already received points, while the orange background colour indicates that the validation of the trip is still pending. On the “Report trip” page (Figure 9b), the user reports a new trip. When the trip starts, the user clicks on the “START TRIP” button. The user reports the transport mode as well. The available transport mode options are “Walking”, “Bus”, “Scooter” and “Bicycle”. When the user clicks on the “END TRIP” button, the user can select “CONTINUE TRIP” if he/she wants to continue the trip. The user can change the transport mode for the next part of the trip. If the user combines two different modes except for walking, it is the case of a multimodal trip. If the user selects “END TRIP”, the trip finishes. After this action, the mobile application sends all the trip information (with GPS and sensor data) to the validation algorithm. The mobile application sends notifications to the user as well. The notification in Figure 9c informs the user that the trip has ended and the corresponding points will be calculated.

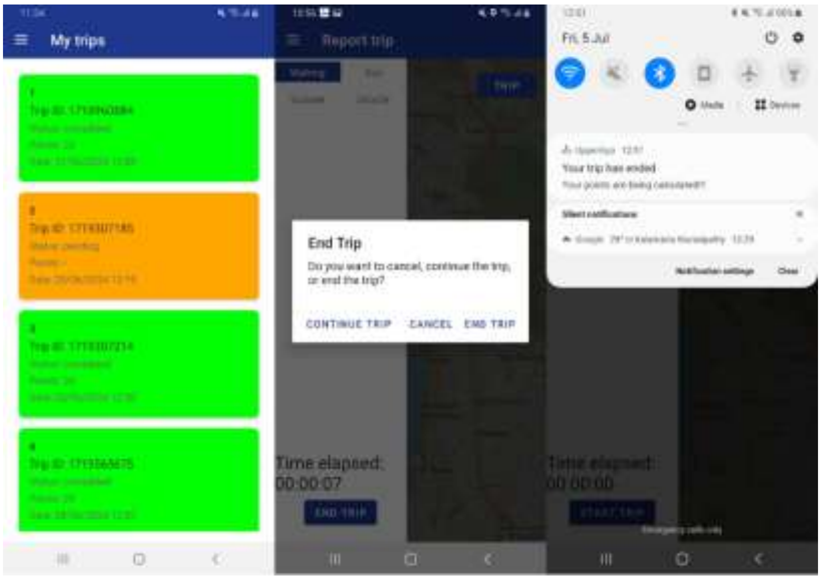

Figure 4: a) My trips, b) Report trip, c) Notification about the end of the trip.

On the “Check points - Claim gift” page (Figure 10a), the user can view his/her current points and the available gifts which are provided by the businesses. The user can sort the available gifts by the points needed for each of them. If the user can afford a gift based on his/her current points and the points needed for it, the “Buy” button is available for this gift. If the user clicks on the “Buy” button, then the user has to confirm the purchase of a coupon for this item (Figure 10b).

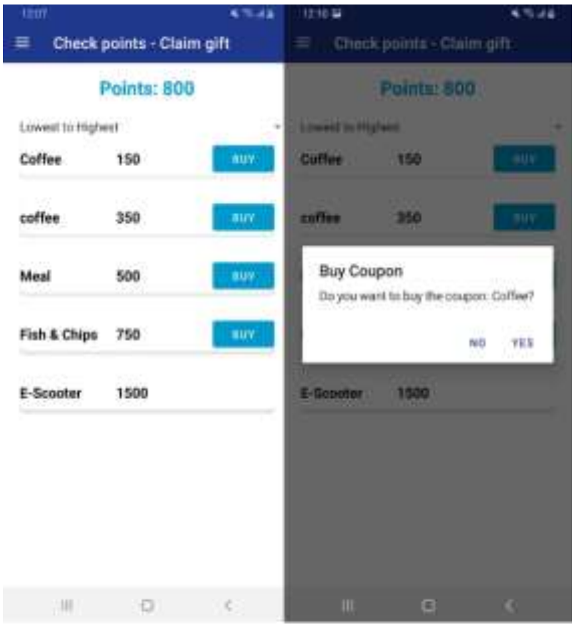

Figure 5: a) Available gifts list, b) Buy coupon confirmation

On the “My coupons” page (Figure 11a), the user can view a list with all his/her available coupons. When the user wants to use a coupon for a business product or service, the user presents a QR code coupon to the business (Figure 11b). After using a coupon, the specific coupon no longer appears on “My coupons”.

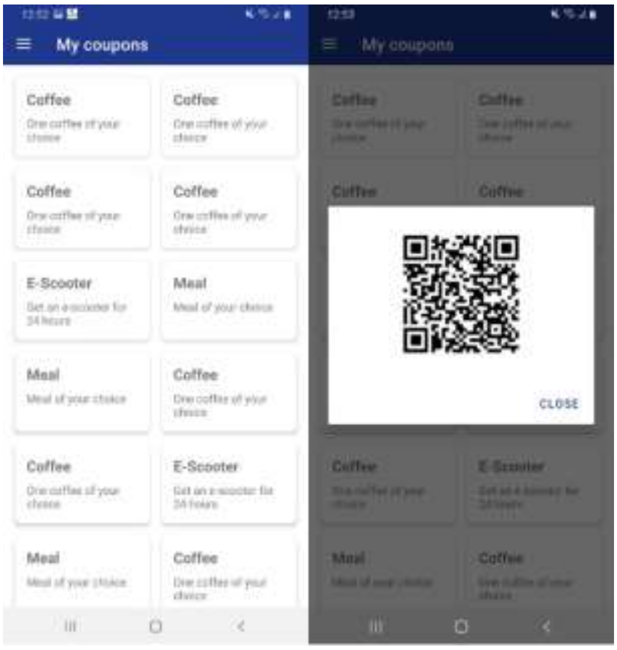

Figure 6: a) My coupons list, b) Coupon QR code display.

<u>Business dashboard</u>

On the “Gifts” page, the business can view and manage the products or services that it offers as gifts. In Figure 12a, the business views the gifts that it offers. For each gift the business can view the title, the description, the points needed and if it is available for the users. After clicking on the “+” button, the business can add a new gift by filling in a form (Figure 12b). The information needed is gift title, gift description, points needed and if it is available. The business can update a gift or delete it (Figure 12c). On the “Scan coupons” page, the business can scan the coupon that the user presents in order to receive a gift. In Figure 13, a successful coupon scan is presented.

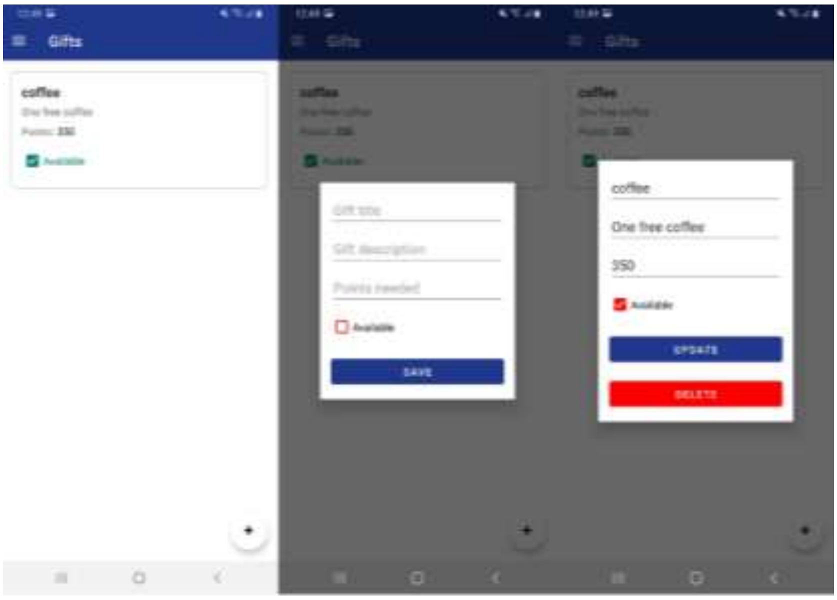

Figure 7: a) Gift list, b) Add gift, c) Update/Delete gift.

Figure 8: Coupon successfully scanned

Challenges & Mitigations

During the development of the application, several challenges were encountered and various solutions were implemented to address them. A major challenge was the absence of GPS data from the datasets and the difficulty in finding datasets containing data on e-scooters. The absence of GPS data was critical, as the literature highlights its importance in TMD algorithms. To improve the performance of the validation algorithm, we reduced the number of transport classes by combining e-scooter and bicycle into a single class called "Micromobility". Since these two modes have a different impact profile (especially for public health) and should therefore be subject to different incentives, this classification will have to be revisited in future applications, when more datasets will be available to distinguish between the two. In addition, we refined settings such as the criteria for the majority class in the verification process, the prediction frequency, and the percentage threshold for validating bus routes to optimize accuracy. Distinguishing between car and bus routes was particularly difficult, so we applied a binary classifier to improve the accuracy in these cases. In addition, we also implemented the bus route validation step to enhance the route verification process.

Walking trips, which are part of almost every journey, posed another challenge. Although considered in the calculations even in small distance trips, they were excluded from bonus points to encourage the use of different modes of transport. Points have to serve as an incentive for users while remaining fair to both users and partner companies. This required developing a precise equation for point calculation and determining appropriate bonus points.

Creating a user-friendly mobile application was paramount. We focused on designing an intuitive interface and ensuring efficient communication between the backend and the mobile application, speeding up the code to produce quick results. To enhance the efficiency of the application, asynchronous processes for tasks such as data validation and transport mode prediction, which can be time-consuming, were implemented to make the application fast and user-friendly. To minimize user input and enhance the user experience, the app was designed to require minimal interaction from users, only at the beginning and the end of their journeys. For instance, users would not need to specify the exact bus route used. These strategies collectively ensured the development of an effective and user-friendly application.

Moreover, the app provides users with detailed feedback, including the status of their trips and the number of points gained. This real-time or near real-time feedback helps users stay informed and motivated. A comprehensive dashboard allows users to track their progress, view their total points, and see the history of their trips. This transparency not only enhances user engagement, but also builds trust in the system.

Next steps towards implementation

As next steps, we plan to further enhance the application through several key initiatives:

Testing: We plan to conduct thorough testing to ensure the system's accuracy and reliability, identifying and addressing any potential issues before the final deployment.

Enhancing the validation algorithm: Our goal is to enhance the validation algorithm based on the results of our testing. This will involve refining the feature set and validation criteria to improve accuracy and reliability in distinguishing between different transportation modes.

Incorporation of GPS data: We aim to enhance the validation algorithm by integrating GPS data into the verification process. This will include creating speed thresholds for trip category categorization, which will contribute to identify and verify the used modes of transportation more accurately.

Integration of the new mobile application: We will integrate the new mobile application user's pages and functionalities into an existing mobile application.

Except of the above initiatives that aim to optimize the digital service and enhance the user experience, as a next step efforts for engaging both users and businesses will be implemented. Up to now, one business has already expressed its interest to participate, by providing free hours of bike-sharing usage.