

# City Public Transport Mobile Payment Protocol

Wersja 1.0

## 1 Opis protokołu

Protokół przeznaczony jest obsługi mobilnego zakupu biletów w komunikacji miejskiej. Obsługuje on żądania pomiędzy telefonem komórkowym pasażera a serwerem operatora transportu publicznego. Dla zapewnienia bezpieczeństwa protokół wykorzystuje numer IMSI<sup>1</sup> klienta i kod PIN (wymagany do potwierdzania niektórych operacji).

Protokół nie obsługuje warstwy rozliczeń za zakupowane usługi – funkcjonalność ta powinna być zrealizowana za pomocą innych metod, np. przedpłaty na specjalne wirtualne konto klienckie u usługodawcy.

W celu ułatwienia przeglądania komunikatów jest to protokół tekstowy.

### Aktorzy

- Aplikacja na telefonie komórkowym pasażera
- Serwer operatora komunikacji miejskiej

## 2 Możliwości protokołu

### 2.1 Kupowanie biletów

Podstawową funkcją obsługiwaną przez protokół jest możliwość zakupu biletów. Aby zakupić bilet, należy podać jego parametry: strefę, typ zniżki i typ biletu (zob. 2.3). Dzięki zapamiętywaniu klienta i jego ustawień (zob. 2.4) podczas kupowania nie trzeba podawać wszystkich tych parametrów, a jedynie te, które różnią się od danych skojarzonych z kontem klienta.

### 2.2 Ustawianie konta i sprawdzanie jego parametrów

Protokół obsługuje wiadomości zakładania i usuwania konta oraz zmianę kodu PIN. Możliwe jest również pobranie salda konta oraz listy ostatnich operacji na koncie (zakupionych biletów).

### 2.3 Pobieranie dostępnych opcji

Klient może pobrać z serwera dostępne rodzaje biletów. W szczególności:

- strefę (np. miejska, podmiejska, etc.);
- typ zniżki (np. normalny – bez zniżki, studencki, ulgowy, etc.);

---

1 Numer IMSI jest wyspecyfikowaną wartością, której opis można znaleźć np. na Wikipedii: [http://en.wikipedia.org/wiki/International\\_Mobile\\_Subscriber\\_Identity](http://en.wikipedia.org/wiki/International_Mobile_Subscriber_Identity).

- typ biletu (np. jednorazowy, godzinny, tygodniowy, etc.);
- parametr własny ustalany przez twórcę aplikacji;

## **2.4 Zapisywanie opcji klienta**

Protokół obsługuje żądania przesłania do serwera i odczytu z serwera domyślnych wartości opcji wymienionych w paragrafie 2.3. Zapisywanie tych informacji umożliwi m.in. dodawanie na pulpicie telefonu widgetów z możliwością bezpośredniego zakupu ulubionego biletu bez konieczności wybierania dodatkowych opcji z menu programu. Dzięki zapisowi po stronie serwera nie będzie trzeba się przejmować utratą danych na telefonie.

## **2.5 Parametry własne**

Powyższe grupy parametrów można rozszerzać o własne rozwiązania. W specyfikacji ABNF parametry te kończą się wartością -param. Opcja rozszerzania wymaga jednak podania w nagłówku informacji o podwersji protokołu.

### 3 Specyfikacja wiadomości (ABNF)

protocol-message = protocol-request | protocol-response

#### 3.1 Specyfikacja zapytań

protocol-request = request-init request-header request-body

request-init = protocol-version SP "REQUEST" CRLF

request-header = client-id CRLF [ authenticate-number CRLF ]

##### 3.1.1 Treść zapytania

request-body = account-settings | check-account | get-ticket-options | set-client-options | get-client-options | ticket-checkout CRLF

##### 3.1.2 Ustawienia konta klienta

account-settings = register-client | unregister-client | change-pin | settings-param CRLF

register-client = "Register-Client" ":" PIN

unregister-client = "UnRegister-Client" ":" PIN

change-pin = "Change-PIN" ":" PIN

settings-param = custom-param-name ":" custom-param-value

##### 3.1.3 Sprawdzanie konta klienta

check-account = get-account-history | get-account-balance | get-account-param CRLF

get-account-history = "Get-Account-History"

get-account-balance = "Get-Account-Balance"

get-account-param = custom-param-name

##### 3.1.4 Pobieranie listy opcji

###### Opcje biletów

get-ticket-options = get-ticket-zones | get-ticket-reductions | get-ticket-types | get-ticket-param CRLF

get-ticket-zones = "Get-Zones-List"

get-ticket-reductions = "Get-Reductions-List"

get-ticket-types = "Get-Types-List"

get-ticket-param = custom-param-name

###### Opcje klienta

get-client-options = get-client-all-options | get-client-zone | get-client-reduction | get-client-type | get-client-param CRLF

get-client-all-options = "Get-My"

get-client-zone = "Get-My-Zone"

get-client-reduction = "Get-My-Reduction"

get-client-type = "Get-My-Type"

get-client-param = custom-param-name

##### 3.1.5 Wysyłanie opcji

set-client-options = 1\*(( set-client-zone | set-client-reduction | set-client-

```
type | set-client-param) CRLF )
set-client-zone      = "Set-My-Zone" *1(":" zone-code)
zone-code           = token
set-client-reduction = "Set-My-Reduction" *1(":" reduction-code)
reduction-code      = token
set-client-type      = "Set-My-Type" *1(":" type-code)
type-code           = token
set-client-param     = custom-param-name *1(":" custom-param-value)
```

### 3.1.6 Kupowanie biletu

```
ticket-checkout     = buy-ticket | check-ticket CRLF
buy-ticket          = "Buy-Ticket" [ ( SP flash-options | ":" type-code) ]
flash-options       = flash-zone | flash-reduction | flash-type | flash-param
flash-zone          = "Zone" ":" zone-code
flash-reduction     = "Reduction" ":" reduction-code
flash-type          = "Type" ":" type-code
flash-param         = custom-param-name ":" custom-param-value
check-ticket        = "Get-Current-Ticket"
```

## 3.2 Specyfikacja odpowiedzi

```
protocol-response      = response-init response-header response-body
response-init         = protocol-version SP "RESPONSE" response-status CRLF
response-status       = response-status-code SP response-status-phrase
response-status-code  = "88"
                       | "20" | "21" | "22"
                       | "30" | "31" | "32" | "33"
                       | 2DIGIT
response-status-phrase = token
response-header       = operator-id client-id
```

### 3.2.1 Treść odpowiedzi

```
response-body         = 1*( request-question [ ":" status ] CRLF response-answer
  CRLF )
request-question      = token
status                = status-code SP status-phrase
status-code           = "20" | "21" | "22"
                       | "30" | "31" | "32" | "33"
                       | 2DIGIT
status-phrase         = token
response-answer       = 1*( token | separators ) CRLF
```

### 3.2.2 Kody odpowiedzi

```
response-status      = "20" SP "SUCCESS"           ; zapytanie zakończone sukcesem
                       | "21" SP "SAVED"           ; dane zapisane
                       | "22" SP "DOWNLOADED"     ; dane pobrane
                       | "30" SP "ERROR"          ; zapytanie zakończone błędem
                       | "31" SP "NO_CLIENT"      ; błąd związany z klientem
                       | "32" SP "NO_PIN"         ; błąd związany z kodem PIN
                       | "33" SP "TO_LESS_PARAMS" ; za mało parametrów
                       | "88" SP "MULTI"         ; wielokrotna odpowiedź
                       | 2DIGIT SP token
```

### 3.3 Argumenty wspólne

#### Wersja protokołu

protocol-version = "CPTMPP" SP 1\*DIGIT "." 1\*DIGIT [ "." 1\*10ALPHA ]

#### Personal Identification Number

authenticate-number = "PIN" ":" PIN CRLF  
PIN = 4DIGIT

#### Identyfikator klienta i operatora

client-id = "Client" ":" IMSI CRLF  
IMSI = MCC MNC MSIN  
MCC = 2DIGIT  
MNC = 2\*3DIGIT  
MSIN = 1\*DIGIT  
operator-id = "Operator" ":" operator-code CRLF  
operator-code = token

#### Inne

custom-param-name = token  
custom-param-value = token  
token = 1\*<any CHAR except CTLs or separators>  
separators = "(" | ")" | "<" | ">" | "@"  
| "," | ";" | ":" | "\" | "<">  
| "/" | "[" | "]" | "?" | "="  
| "{" | "}" | SP | HT

## 4 Przykłady

- Kupowanie biletu bez rejestracji

```
CPTMPP 1.0 REQUEST
Client:310150123456789
PIN:1234
Buy-Ticket:15min
→ CPTMPP 1.0 RESPONSE 31 NO_CLIENT
Operator:PLKRK
Client:310150123456789
Buy-Ticket
Please register in system.
```

- Rejestracja klienta

```
CPTMPP 1.0 REQUEST
Client:310150123456789
Register-Client:1234
→ CPTMPP 1.0 RESPONSE 20 SUCCESS
Operator:PLKRK
Client:310150123456789
Register-Client
```

- Sprawdzenie dostępnych stref

```
CPTMPP 1.0 REQUEST
Client:310150123456789
PIN:1234
Get-Zones-List
→ CPTMPP 1.0 RESPONSE 22 DOWNLOADED
Operator:PLKRK
Client:310150123456789
Get-Zones-List
1:Miasto;2:Miasto+Aglomeracja
```

- Ustawienie strefy i zniżki

```
CPTMPP 1.0 REQUEST
Client:310150123456789
PIN:1234
Set-My-Zone:1
→ CPTMPP 1.0 RESPONSE 21 SAVED
Operator:PLKRK
Client:310150123456789
Set-My-Zone
```

- Kupowanie biletu z niepełnymi danymi

```
CPTMPP 1.0 REQUEST
Client:310150123456789
PIN:1234
Buy-Ticket:15min
→ CPTMPP 1.0 RESPONSE 33 TO_LESS_PARAMS
Operator:PLKRK
Client:310150123456789
Buy-Ticket
Please set reduction type.
```

- Kupowanie biletu

```
CPTMPP 1.0 REQUEST
Client:310150123456789
PIN:1234
Buy-Ticket Reduction:student
Type:15min
→ CPTMPP 1.0 RESPONSE 20 SUCCESS
Operator:PLKRK
Client:310150123456789
Buy-Ticket
Zone:1;Reduction:student;Type:15min
```