NCR ATLEOS

Artificial Intelligent Teller Machine Solution

6/11/2024

NCR Atleos Hong Kong

Table of Contents

1 Solution overview

1.1 ATM & ITM Today

ITM is self service. there is no expectation on customer service, availability is almost everything. It's a waste for such well accepted channel if not making it works harder and smarter for the bank.

ITM could be a good branch transformation solution to deliver teller services in a cost efficient way. However, the need of remote teller operation together with small scale deployment become a barrier.

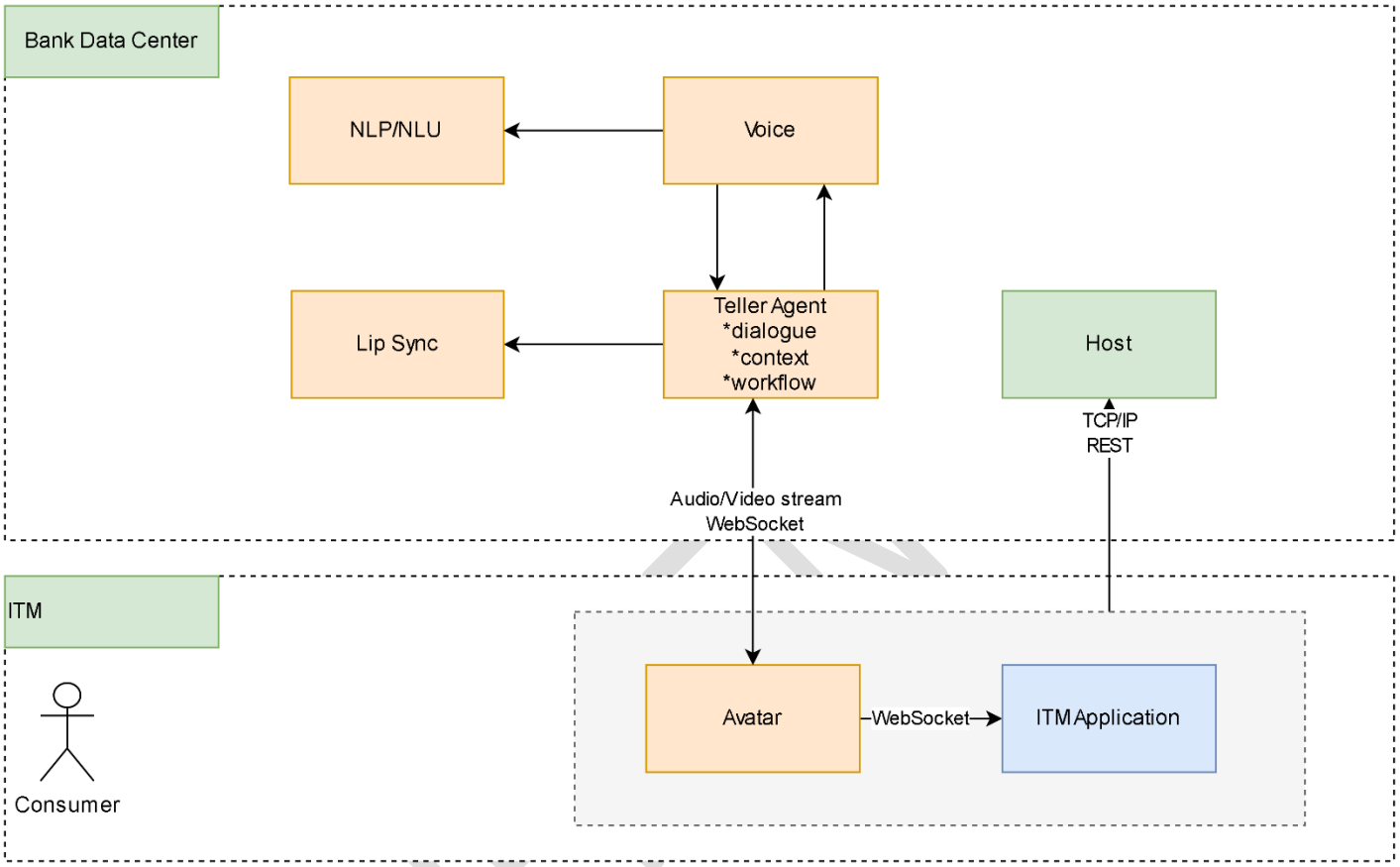
1.2 Next Generation ATM & ITM

We plan to bring AI technique in ATM & ITM solution, enrich ATM & ITM functionality and remove the need of remote teller, and help bank to gain unfair advantage through superior customer experiences and best-in-class operation efficiency.

- AI Teller Machine offer several benefits for financial institutions and customers. Here are some key advantages:
- Significant Cost Savings: AI ITM reduce labour costs by spreading video tellers across branches.
- Increased Convenience: AI ITM provide a virtual in-person banking experience. Customers can perform various services, including cash withdrawals, deposits, loan payments, and opening accounts, through live video chat with an operator.
- Operational Efficiency: Most transactions handled by traditional tellers can be performed through AI ITM, leading to operational efficiency and revenue growth opportunities.
- Innovative Advantage: AI ITM differentiate financial institutions by offering modernized services and improving customer experiences.
- Market Presence: Deploying AI ITM expands a bank's reach, especially in rural or urban areas where physical branches may be limited.
- Customer Satisfaction: AI ITM provide flexibility and personalized assistance, enhancing overall customer satisfaction.
- Consistency: AI ITM maintain consistent service quality across branches, regardless of location or time.

2 Solution Architecture

2.1 Architecture



AI Agent is responsible for

- Create Avatar running on ITM.
- Create dialogue with customer.
- Extract transaction data from dialogue.
- Manage workflow predefined and context.
- Communicate with AI agent.
- Interact with ITM application, exchange instructions and events.

2.2 AI Teller Responsibilities

- Train AI Teller models with bank transactions flow and data, fully understand all transaction flow.
- Initiate an avatar to make human-like dialogue with a customer.

- Manage workflows and dialogue context.
- Guide customer to complete a transaction or service.
- Extract transaction data from dialogue.
- Send instructions or actions to ITM
    - Complete transaction
    - Enable card reader
    - Enable camera
    - Scan passport
    - ...
- Guide customer to perform operations on ITM
    - Put passport
    - Insert banknotes
    - Insert cheques
    - Take money
    - Tap ID card
    - Take photo
    - Collect fingerprint
    - ...
- Receive and handle actions and events from ITM

2.3 ITM Responsibilities

- Open/Close a chatbot dialog session with AI Teller Server.
- Receive instructions or actions from AI Teller Server, perform operations.
- Report operation execution progress to AI Teller Server.
- Transaction authorization with host or bank end system.
- Record transaction information.
- Settlement.

3 System configuration

ATM & ITM Configuration

CPU	Intel(R) Core(TM) i5-6500T CPU @ 2.50GHz, 2496 Mhz, 4 Core(s), 4 Logical Processor(s)
RAM	8.00 GB
Hard disk	256 GB
Screen resolution	1280 x 960 1024 x 768
OS	Windows 10 Enterprise LTSC
Browser	Edge

AI Teller Server Configuration

CPU	8 CPUs 128GB RAM
GPU	3X Nvidia RTX 4090 24GB
Hard disk	512GB SSD (SATA3 capable)
OS	Ubuntu 22.04

4 Transactions flow & interface

AI Teller can assist customer to complete traditional transactions, internet banking transactions, and all teller transactions.

4.1 Common interface

4.1.1 Interface of ITM to AI Teller

Open session	{
	"action": "open-session",
	"parameters": {
	"language": "en",
	"cardNumber": "1234567890123456"
	}
	}
		{
		"event": "session-opened"
		}
		
		{
		"event": "session-open-failed",
		"parameters": {
		"reason": "service is not available",
		}
		}

Close session	{
	"action": "close-session",
	"parameters": {
	"reason": "cancelled",
	}
	}
		{
		"event": "session-closed"
		}

End transaction	{
	"action": "end-transaction",
	}	

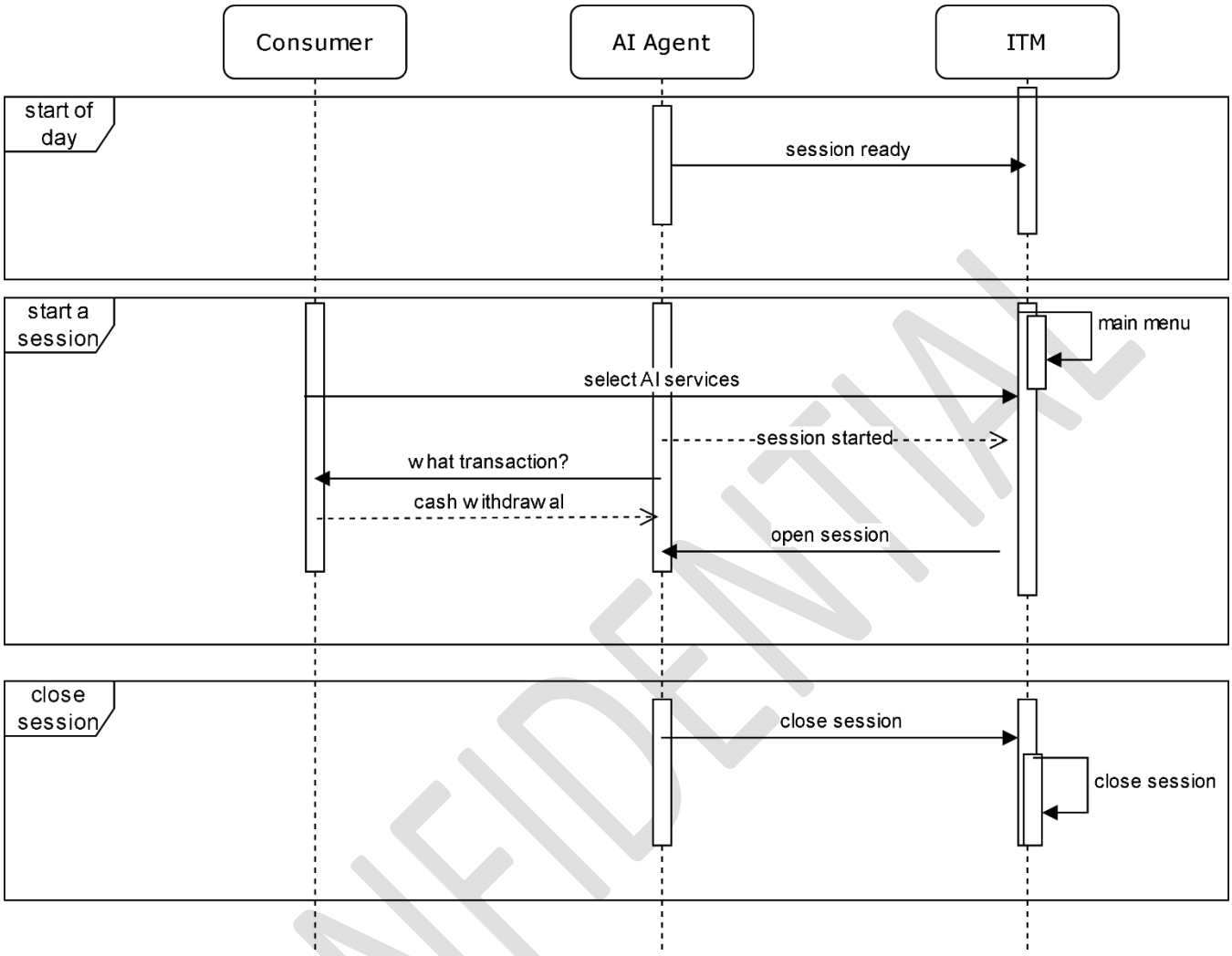
Event	{
	"action": "event",
	"parameters": {
	"actionCode": "continue",
	"vg": "please insert money",
	}
	}	

4.1.2 Interface of AI Teller to ITM

Close session	{
	"action": "close-session",
	"parameters": {
	"reason": "completed",
	}
	}
		{
		"event": "session-closed"
		}

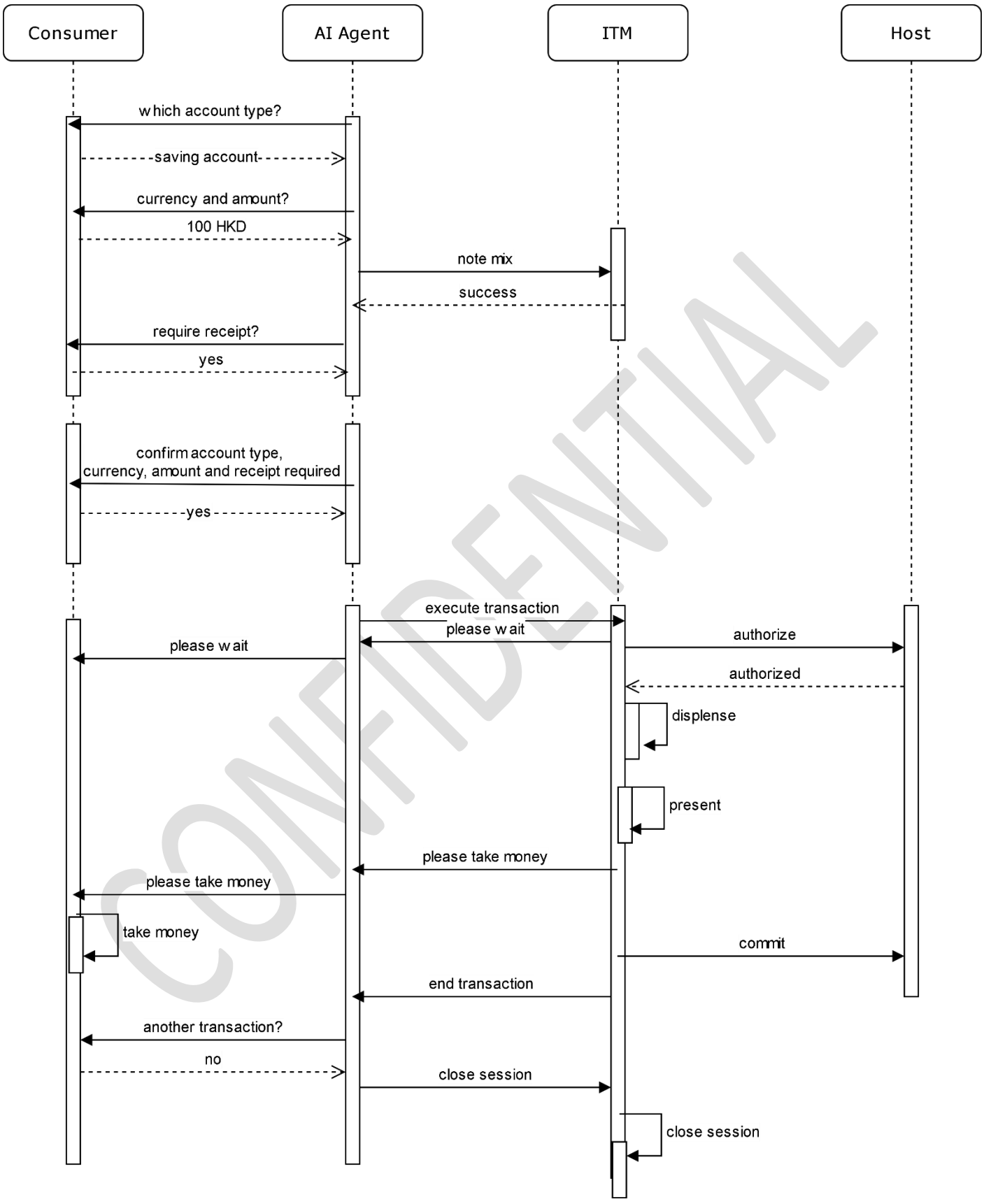
AI Teller ready	{
	"event": "ai-teller-ready"
	}	

4.2 Session



4.3 Cash Withdrawal

4.3.1 Flow



Step	Role	Operation
1	AI Teller	Make chatbot dialog and collect transaction type: Cash Withdrawal
2	AI Teller	Make chatbot dialog and collect account type: Saving
3	AI Teller	Make chatbot dialog and collect currency type: HKD
4	AI Teller	Make chatbot dialog and collect amount: 100
5	AI Teller	Send instructions to ITM to perform note mix
6	ITM	Return note mix result
7	AI Teller	Make chatbot dialog and collect receipt requested: True
8	AI Teller	Send instructions to ITM to perform Cash Withdrawal transaction
9	ITM	Authorization with host, wait host response
10	ITM	If transaction authorized, dispense money and send “please take money” event to AI Teller
11	AI Teller	Speak “please take money”
12	ITM	Print receipt and send “please take receipt” event to AI Teller
13	AI Teller	Speak “please take receipt”
14	ITM	Send instruction “end-transaction” to AI Teller
15	AI Teller	Make chatbot dialog whether need other transaction: No
16	AI Teller	Send instructions to ITM to close session
17	ITM	Close chatbot dialog session and move to idle status

4.3.2 Data required

Data	Provided by	type	example
amount	consumer	integer	100
currency	consumer	string	HKD
account type	consumer	string	credit
receptRequested	consumer	boolean	True/false

4.3.3 Interface of AI Teller to ITM

Note Mix	{
	"action": "note-mix",
	"parameters": {
	"currency": "HKD",
	"amount": 100
	}
		{
		"event": "note-mix",
		"parameters": {
		"success": true | false,
		"actionCode": "re-enter-data",

	}	"language": "en", "vg": " Sorry, amount not in multiple of 100", } }

Cash Withdrawal	{
	"action": "cash-withdrawal",
	"parameters": {
	"currency": "HKD",
	"amount": 100,
	"accountType": "saving",
	"receiptRequested": true,
	
	}
		{
		"event": "transaction-started"
		}

4.3.4 Interface of ITM to AI Teller

Event	{
	"action": "event",
	"parameters": {
	"actionCode": "41",
	"vg": "please take money",
	}
	}
	Vg
	• Please take money
	• Please take receipt
	• Please take card
	• Sorry, the transaction can not be completed	

4.4 Open Account

4.4.1 Flow

Step	Role	Operation
1	AI Teller	Make chatbot dialog and collect transaction type: Open Accont
2	AI Teller	Make chatbot dialog and collect personal information, such as: Name, Address, Mobile Number, Email, Birth Date, ID Number or Passport Number, etc..
3	AI Teller	Make chatbot dialog and ask customer to put his/her ID or passport for scanning image
4	ITM	Scan ID or passport and collect image
5	AI Teller	Make chatbot dialog and ask customer to take a photo
6	ITM	Take photo and collect photo image

7	ITM	Authorization with Host
8	ITM	Send event “open account request is submitted” to AI Teller
9	ITM	Print receipt and send “please take receipt” event to AI Teller
10	AI Teller	Speak “please take receipt”
11	ITM	Send instruction “end-transaction” to AI Teller
12	AI Teller	Make chatbot dialog whether need other transaction: No
13	AI Teller	Send instructions to ITM to close session
14	ITM	Close chatbot dialog session and move to idle status

4.4.2 Data Required

Data	Provided by	type	example
name	consumer	string	
Birth date	consumer	string	
mobile number	consumer	string	
email	consumer	string	
Id or passport	consumer	string	
Id or passport image	read image on ITM	string	
photo	capture image on ITM	string	

4.4.3 Interface AI Teller to ITM

Take photo	{
	"action": "take-photo",
	"parameters": {}
	}
		{
		"event": "take-photo",
		"parameters": {
		"success": true | false,
		}
		}

Take photo	{
	"action": "scan-id",
	"parameters": {}
	}
		{
		"event": "scan-id",
		"parameters": {
		"success": true | false,
		}
		}

Open account	{
	"action": "open-account",
	"parameters": {
	"name": "Ivan Chan",
		{
		"event": "open-account",
		"parameters": {

	"id": "1234123", "email": "ivan.chan@outlook.com", "mobile": "8989668986", "birth": "10-09-1999 ", "passport": "896968689", } }	"success": true | false, } }

4.5 Time Deposit

4.5.1 Flow

Step	Role	Operation
1	AI Teller	Make chatbot dialog and collect transaction type: Time Deposit
2	AI Teller	Make chatbot dialog and collect account type: Saving
3	AI Teller	Make chatbot dialog and collect currency type: HKD
4	AI Teller	Send instructions to ITM to retrieve terms of time deposit
5	ITM	Retrieve terms of time deposit from bank backend and return to AI Teller
6	AI Teller	Make chatbot dialog and collect term of time deposit
7	AI Teller	Send instructions to ITM to retrieve account balance amounts
8	ITM	Retrieve account balance from bank backend and return to AI Teller
9	AI Teller	Make chatbot dialog and collect time deposit amount: 5000
10	AI Teller	Make chatbot dialog and collect receipt requested: True
11	AI Teller	Send instructions to ITM to perform Time Deposit transaction
12	ITM	Authorization with Host
13	ITM	Send event “transaction is authorized” to AI Teller
14	ITM	Print receipt and send “please take receipt” event to AI Teller
15	AI Teller	Speak “please take receipt”
16	ITM	Send instruction “end-transaction” to AI Teller
17	AI Teller	Make chatbot dialog whether need other transaction: No
18	AI Teller	Send instructions to ITM to close session
19	ITM	Close chatbot dialog session and move to idle status

4.5.2 Data required

Data	Provided by	type	example
amount	consumer	integer	100
currency	consumer	string	HKD
account type	consumer	string	credit
term	consumer	integer	3
receptRequested	consumer	boolean	True/false

4.5.3 Interface of AI Teller to ITM

Retrieve terms of time deposit	{
	"action": "retrieve-time-deposit-term",
	}
		{
		"event": " time-deposit-terms",
		"parameters": {
		"success": true,
		"terms": [
		{
		"term": "3 months",
		"interest": 0.035
		},
		{
		"term": "6 months",
		"interest": 0.038
		},
		{
		"term": "1 year",
		"interest": 0.040
		},
		{
		"term": "3 years",
		"interest": 0.045
		}
		]
		}
		}

Retrieve balance amounts	{
	"action": "retrieve-balance-amounts",
	"parameters": {
	"accountType": "Saving",
	}
	}
		{
		"event": "balances",
		"parameters": {
		"success": true,
		"balances": [
		{
		"currency": "HKD",
		"amount": 10000
		},
		{
		"currency": "USD",
		"amount": 20000
		},
		{

		"currency": "CNY", "amount": 5000 }] } }

Time deposit	{
	"action": "time-deposit",
	"parameters": {
	"currency": "HKD",
	"amount": 5000,
	"term": 3,
	"accountType": "saving"
	}
	}
		{
		"event": "transaction-started"
		}

5 Action Code

home	Clear previous transaction data Go back home page to ask customer what service
close-session	Speak “service is cancelled” Send “session-closed” event to ITM
re-enter-data	Speak vg text, waiting customer to speak for previous data required
continue	Continue according to workflow

6 Communication Protocol

Recommend to use **WebSocket** protocol for communication between ITM and AI Teller.

WebSocket is a communications protocol that enables bidirectional communication over a single TCP connection. Here are the key points:

- Simultaneous Two-Way Communication: WebSocket allows real-time, full-duplex communication between a client (usually a web browser) and a server. Unlike traditional HTTP, which is request-response based, WebSocket enables both parties to send messages independently.
- Standardization: The WebSocket protocol was standardized by the IETF as RFC 6455 in 2011. It provides a consistent way for web applications to establish and maintain connections.
- HTTP Compatibility: WebSockets work over HTTP ports (usually 80 and 443), making them compatible with existing infrastructure, including HTTP proxies and intermediaries.
- Event-Driven: With WebSockets, you can send messages to a server and receive event-driven responses without polling the server for updates.