Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

IT Service Continuity: A Cloud-Based and NLP-Driven Solution Approach

Mohammed Tou [1*], Adil Toumouh [2]
[1] Ph.D candidate, ESI - Higher School of Computer Science, Sidi-bel-Abbes, Algeria
[2] Doctor, Institute of Computer Science, Sidi-bel-Abbes, Algeria
* Corresponding Author: m.tou@esi-sba.dz

ARTICLE INFO

Received: 28 Dec 2024

Revised: 18 Feb 2025

Accepted: 26 Feb 2025

ABSTRACT

Ensuring the availability of IT services within organizations is increasingly important in an interconnected world. This paper explores the concept of service continuity and presents technical approaches and methodologies to address this issue.

The core issue discussed in this paper is the lack of service continuity. The paper defines service continuity in the context of information system services and identifies failures that occur as a result of discontinuity. Relevant research is referenced to extract tools and technological paradigms that facilitate solutions to ensure a minimum level of service continuity.

Given that Internet availability is crucial for continuity, alternative paths, such as the conventional PSTN telephone network, are considered. To complete the solution chain, concepts like voice and speech recognition, AI, NLP, and cloud computing are utilized.

A key element introduced in this research is the "voice request", which acts as an intermediary between the user and the service. A broker ensures that the message is delivered to the appropriate recipient service and facilitates the user's response. These elements are orchestrated by a pipeline that maintains the integrity of the request and response. Speech recognition initiates the solution process, combined with NLP and its statistical approaches and neural networks. Cloud technology further secures the solution in both directions.

While the proposed solution does not completely replace service availability, its aim is to maintain a minimum level of service continuity, preventing a complete shutdown of the organizational information system.

Keywords: *IT, PSTN, AI, NLP, internet, cloud, service availability, service continuity*

INTRODUCTION

The Internet initially served as a platform for displaying static web pages containing informational content. As its popularity grew, experts, researchers, and businesses worked together to transform the Internet into a foundation for building corporate information systems.

In addition to providing information, the Internet now fulfills more complex needs such as transactional operations, storage and calculation services, and application and platform outsourcing. This has given rise to a new paradigm known as Cloud Computing.

Since the 2000s, the Internet has become crucially important, especially within organizations across various industries. It serves as the primary means of ensuring the availability and continuity of services provided by organizations. However, it is important to differentiate between two related but distinct concepts: service availability and service continuity. While they may appear similar, in IT practice, these two notions have different meanings [1-3].

Availability

Service availability refers to the operational and accessible state of a system at all times. It is measured as a percentage, indicating the extent to which a system is accessible within the planned timeframe

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

The percentage availability is calculated based on two factors. The first factor is the Agreed Service Time (AST), which represents the duration during which the service should be available within the specified period. The second factor is the Downtime (DT), which refers to the time when the service is not accessible, and is expressed as a percentage[1].

$$availability = ((AST - DT)/AST) \times 100\%$$

If, for example, AST is 100 hours and downtime is 2 hours, the availability percentage will be ((100-2)/100) * 100% = 98%.

Tolerance levels: The tolerance level varies depending on the business sector of the company. For example, a medical organization is less tolerant of service unavailability compared to a company in a less critical service sector.

In order to ensure service availability in IT, there are certain key performance indicators (KPIs) that are monitored and measured. Commonly used KPIs for assessing IT service continuity include system availability. Essential KPIs include mean time to restore (MTTR), mean time between failures (MTBF), incident rate, restore success rate, recovery time objective (RTO), recovery point objective (RPO), service level objective (SLO), and service level agreement (SLA)[2],[3].

Continuity

Service continuity refers to the strategies and protocols implemented to ensure the long-term availability of a system. In an organization, the effectiveness of service continuity is measured by its ability to restore services after a disruption or significant incident. This capability is enhanced by proactive measures to minimize the impact, as well as a comprehensive framework of procedures and contingency plans[4].

PROBLEMATIC

In developing countries like Algeria, frequent internet outages are a prevalent issue that can greatly disrupt IT services. The precarious state of infrastructure and information technology systems, along with limitations in digital government and governance of information systems, have greatly eroded the government's confidence in the existing computerized systems. These outages can occur inadvertently due to infrastructure vulnerabilities or intentionally, such as during nationwide events like the Algerian "bac", where the government intentionally disrupts the network due to the impact of interpersonal connectivity on system functionality.

According to "Jeune Afrique," Internet was suspended for five days, from June 20 to June 25, 2018, during business hours, coinciding with the exam period. Ali Kahlane, President of the Association of Alternative Telecom Operators (AOTA), reported a production loss of 760 million dinars (5.5 million euros) due to an eleven-hour outage. Kahlane noted that the international standard for fault tolerance in real-time internet connections is no more than 5.6 minutes per year. However, recent disruptions by Algerian authorities resulted in a total of 660 minutes of outage [5].

SOLUTIONS APPROACH

Our work aims to ensure"a minimum of IT service continuity" during an Internet outage. In this paper, we will present our research results so far, starting with an overview of the current state of the art and potential solutions.

The first approach involves utilizing Cloud Computing capabilities as a stable technological platform, alongside incorporating new artificial intelligence technologies like NLP and deep learning. This allows us to establish an application infrastructure that seamlessly takes over when services are disrupted. We examined the two service models offered by the cloud, IAAS and PAAS, to extract the necessary technical elements for our solution. We are particularly interested in IAAS's support of PSTN networks and PAAS's tools for implementing NLP.

The second solution is considered secondary because it requires a higher level of tolerance. It draws inspiration from the concept of "disaster recovery" (DR), which involves procedures and plans to help organizations quickly resume operations after a service interruption. Our approach does not fully replicate this expensive and complex plan, but instead takes inspiration and improves certain aspects of it.

CLOUD BASED SOLUTION

In the cloud, Infrastructure as a Service (IaaS) is a fundamental model offering basic tools, including hardware components like machines, storage, networks, routers, and firewalls[6].Minimum of Continuity as a Service (MoCaaS) is our proposed solution to enhance IaaS by ensuring minimum service continuity during Internet interruptions.

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

The issue arises when a local service user attempts to access a feature requiring remote connectivity. Disruptions in the Internet connection, whether due to theft or involuntary causes, can render the required enterprise service unavailable. MoCaaS aims to mitigate this by providing an ecosystem that maintains minimal connectivity between local and remote services during such disruptions.

MoCaaS does not replace existing Disaster Recovery (DR) mechanisms but complements them. This ecosystem offers an alternative method to maintain minimal connectivity. To ensure this connectivity, both local and remote services must adhere to specific contracts.

At the local level:

All organizations should be equipped with standard (analogic) telephone service (PSTN protocol), if not, the employee's cell phone does the matter.

At remote level:

All services requiring high availability should be subscribed in a Directory of Services Requiring Continuity (DSRC). This directory is an integral component of the MoCaaS.

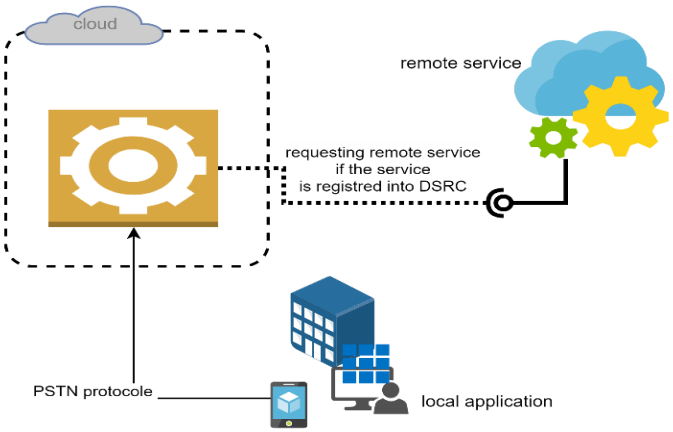

Figure 1. cloud based solution

MOCAAS ECOSYSTEM

As discussed, MoCaaS utilizes Voice Recognition, NLP, and LLM techniques to extend cloud capabilities, ensuring minimum service continuity. This process involves multiple stages in a pipeline where each stage's output becomes the input for the next. Each phase incorporates concepts and techniques from various research domains, including Identity Management, Voice Recognition, NLP, LLM, APIs, and Cloud Computing. The pipeline processes an input data model through transformations, exploration, and extensions, culminating in the final outputs or results.

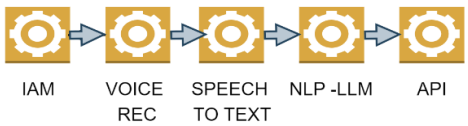

Figure 2. MoCaaS Ecosystem

Architecture

Our research method involves deconstructing the MoCaaS ecosystem into granular modules. Each module represents a phase in the MoCaaS pipeline, receiving a specific input model, processing it, and delivering an output model to the next stage. Given the heterogeneous nature of the data, a significant portion of our research focuses on establishing frameworks for mapping and transforming these datasets.

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Nomenclatures

Table 1. Nomenclature of MoCaaS terms and concepts

Nomenclature	Abbrv.	Description
Client service requestor	SRr or client	the person who comes to the organisation requesting a service
User	User	Final user of the system. Taking request from client and invoke MoCaas
Context	ctx	The business domain as : Gov, Bank, Healthy..
subscriber	sSub	The organisation subscriber to MoCaaS
Endpoint	EP	The targeted

Elements and concepts

As black box description, MoCaaS receives voicerequest (VReq) from requesting users (UReq),processes it and routes it to an external system that exposethe appropriate API subscriber (API-sub). This latter returnsa response to MoCaas, then MoCaas builds a voice responsein order todeliver it to UReq

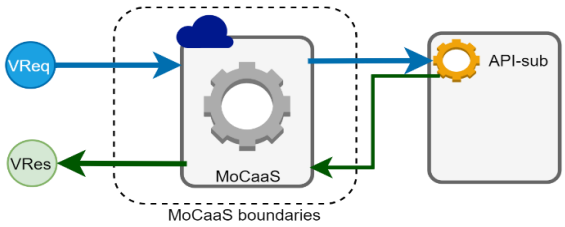

Figure 3. MoCaaS as black box

MoCaaS model

At abstract level, organizations belonging to different contexts (business domain) can subscribe to MoCaaS:

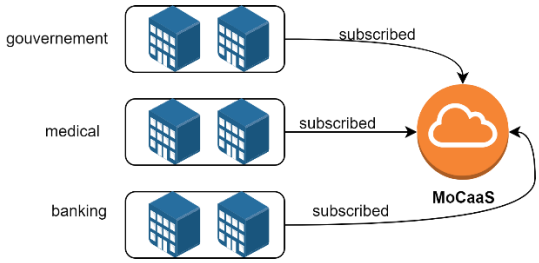

Figure 4. MoCaaS subscribers

The following data model represents the integration of all actors within MoCaaS, incorporating both conceptual elements and key stakeholders.

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

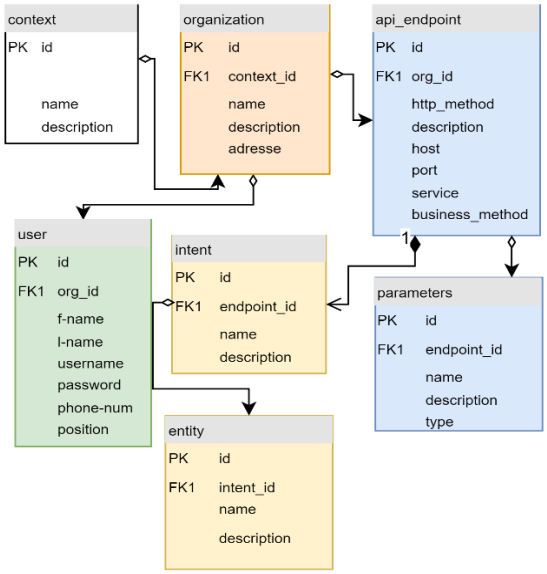

Figure 5. MoCaaS data model

Where:

Context (context) represents a broader category or domain that groups different organizations.

Organization (organization) represents an entity (e.g., a company) that operates within a specific context.

User (user)represents an individual associated with an organization, such as employees or system users.

API Endpoint (api_endpoint), represents an API service provided by an organization, defining its communication parameters.

Parameters (parameters), stores the parameters required for API requests, specifying details such as data types.

Intent (intent), represents an action or intent linked to an API endpoint, often used in AI-based interactions.

Entity (entity) represents key elements related to an intent, useful for recognizing structured data within user queries.

Scope

As explained aboveMoCaaS processing is bidirectional. The scope of this publication will be limited to the processing of the request.

MOCAAS PROCESS FLOW

The primary objective of decomposing MoCaaS is to simplify its structure. MoCaaS's complexity arises from three dimensions: the nature of unstructured data to be processed, the uncontrolled volume of input data, and the multitude of concepts and technologies involved. Consequently, MoCaaS is designed as a pipeline with six stages:

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

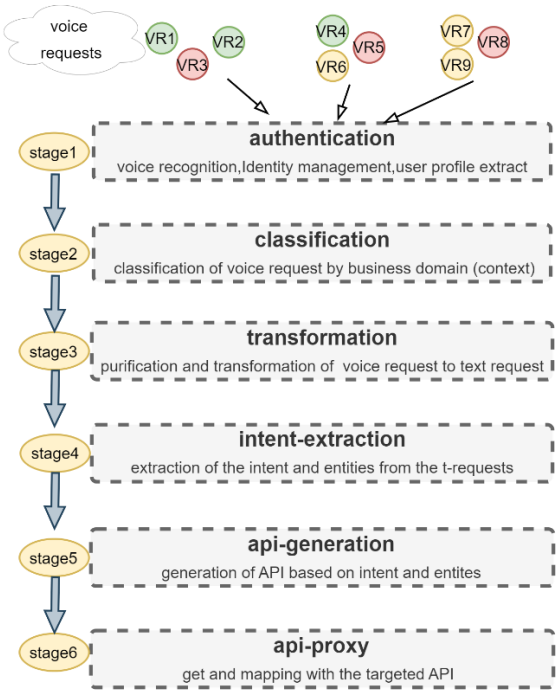

Figure 6. MoCaaS pipeline to process aV-REQ

Stage 1, Authentication

To handle voice requests originating from the client (requestor) through the user, the Authentication/Authorization module uses IAM_DB (Identity and Access Management Database) to retrieve the user's credentials and the client's information or metadata (cMD).The IAM_DB database manages user identities, authentication credentials, roles, permissions, and metadata while implementing a password-less authentication approach. Instead of relying on traditional passwords, it utilizes biometric signatures, specifically voice recognition, to authenticate users securely. This enhances both security and user experience by ensuring seamless and fraud-resistant access control, reducing the risks associated with password breaches and unauthorized access.

There are various password-less authentication methods, including biometric authentication, one-time passwords (OTP), device-based authenticationand others[7].Each approach has its advantages depending on security needs, scalability, and user experience.The detailed study and comparison of different password-less authentication methods are beyond the scope of this work and will be explored in future research. So, initially we suggest using biometric signatures, specifically voice recognition, as a secure and seamless authentication method.

Model 1. User authentication data and metadata

```
{ "_id": "user_uuid",
 "username": "org_user",
 "phonenum": "+21300000000",
 "biometric_signature":
{ "type": "voice",
  "data":  BinData(0,
        "base64_encoded_voiceprint")
 },
 "metadata": {
  "created_at": "2024-02-08T12:00:00Z",
  "last_updated": "2024-02-08T12:30:00Z"
 }
}
```

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Output model

This stage filters and delivers only the VReq for authenticated users plus their metadata.

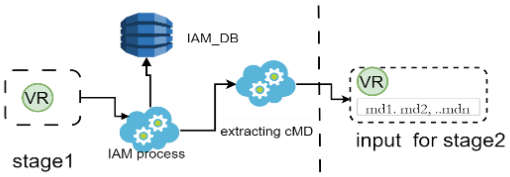

Figure 7. stage1 processing, in/out

Given that, R the set of all incoming **voice requests (VReq)** → $R=\{r_1,r_2,...,r_n\}$,U the set of registered **users**in the IAM_DB → $U=\{u_1,u_2,...,u_m\}$, C be the set of **clients (requestors)** making requests → $C=\{c_1,c_2,...,c_p\}$ and M the set of **metadata attributes** containing relevant information about both users and clients→$M=\{m_1,m_2,...,m_q\}$, then The **authentication function**F_o ensures that the **user** $u_j$submitting the request is authenticated, and the system links the request to the corresponding client ck:

$$F_o:R\rightarrow(U\times C)$$

Where, for an incoming voice request r_i*:*

$$(u_j,c_k)=Fo(r_i),where\ u_j\in U,c_k\in C$$

Once the identities of both **the user** and **the client (requestor)** are verified, the system retrieves **metadata** m_j from IAM_DB and attaches it to the request:

$$(r_i,u_j,c_k,m_j),\ where\ m_j\in M$$

Stage 2, Classification

In this stage, the system extracts the user's **business domain**, referred to as context (**CTX**), and categorizes the voice request (**VReq**) accordingly. This classification ensures that requests are processed within the appropriate business domain, allowing for **context-aware service handling**. The extracted context serves as a crucial parameter for subsequent processing stages, enabling precise intent detection and API mapping.

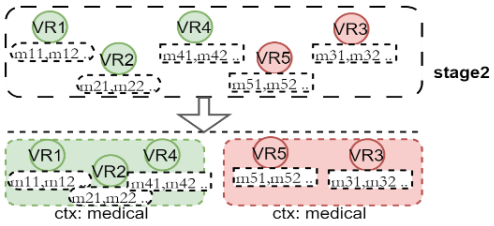

Figure 8. stage2 processing

Given CTX the set of business contexts, exp: $CTX=\{GOV,BANK,HEALTH,...\}$, The classification function F_1 assigns the business context c_k based on the user handling the request: $F_1:U\rightarrow CTX$, then, for an authenticated voice request $r_i$from user u_j:$ck = F_1(u_j),\ where\ c_k \in CTX$.

Output model

This stage delivers as output information: **Voice Request (VReq)**: The original **recorded audio file, Context (CTX)** andMetadata (cMD)retrieved from Stage 1**.**

Stage 3, Transformation

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

In this stage, the voice request (VReq) is transformed into a text request (TReq) through speech-to-text processing, followed by semi-formalization and purification to ensure the extracted text is structured and clean for further processing.

Tools and concepts

Speech-to-Text (STT) processing is a process that includes multiple stages such as Automatic Speech Recognition (ASR), text normalization, punctuation correction, and domain adaptation to produce structured and meaningful text.

Automatic Speech Recognition (ASR) is a core component within the STT pipeline, responsible for the initial transcription of speech into raw text without additional enhancements.[8][9][10].

Transformation process overview

This process includes three steps: (1). Speech-to-Text Conversion (VReq → TReq), (2). Semi-Formalization and (3) Purification.

During the step1, more than converting speech to text, the ASR model used, should supports: different Arab Algerian accents, the domain-specific (context) vocabulary and noise filtring. Regarding the Semi-Formalisation step, it normalizes variations in phrasing and ensures that **different ways of phrasing the same request** are aligned to a common structure, for example (Arabic):

Raw Transcription: **“"** *لدي رغبة في تحويل مبلغ يقدر ب 1000 دينارإلى حساب التوفير الخاص بي*

Semi-Formalized: تحويل مبلغ 1000 دينار إلى حساب التوفير

Then, the Purification step, firstly removes irrelevant and filter words (e.g., "allo", "please", "can you", ...), and unnecessary sentences fragments and secondly it applies some grammar corrections and tokenization, as:

Noisy Transcription (Francais): « *Euh, je voudrais, euh, vous savez, envoyer, euh, mille euros s'il vous plaît à mon épargne* »

Purifies output : « *Virement de 1000DA sur le compte d'épargne*».

Given that T the set of **text requests (TReq)**:$T=\{t_1,t_2,...,t_n\}$

The transformation function F_2 maps a voice request to a cleaned text request:$F_2:R\rightarrow T$

This transformation occurs in three stages:

Speech-to-Text Conversion (ASR Model):

$$t_i'= ASR(r_i),\ t_i'$$: **raw transcribed text**

Semi-Formalization (Normalization N):

$$t_i''=N(t_i'),\ t_i''$$: is the **normalized text**.

Purification (Cleaning Function P):

$$t_i=P(t_i''),\ t_i \text{ is the } \textbf{final purified text request}$$

The full transformation process can be expressed as:

$$t_i=P(N(ASR(r_i)))$$

Output model

The system delivers: Text Request (TReq) – The cleaned, structured version of the voice request, Context (CTX) – The business domain assigned in Stage 2 and Metadata (cMD) – The client and user information carried forward.

Stage 4: Intent Extraction&Entity Recognition

In this stage, Natural Language Processing (NLP) and Large Language Models (LLMs) are used to analyze the request context and extract key components, including the intent and relevant entities. The intent represents the goal or

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

purpose of the user's request, while entities(Named Entity Recognition - NER) are specific details that provide additional information, such as amounts, locations, or account types[11].

Process overview

The text request (TReq), produced from Stage3 (Transformation), is analyzed with respect to the business context (CTX) assigned in Stage 2 (Classification).This ensures that intent extraction is domain-aware, reducing ambiguity. In this stage we process as follows:

Intent Detection (Classification Task)[12]

A pre-trained LLM is used to classify the intent. In our case, we take into consideration Arabic, French, and the Algerian Arabic dialect (Darja). For this reason, the choice of the LLM model must take multilingual and dialectal aspects into account to ensure robust language understanding. Preliminary investigations say that the suitable models for this task are: mBERT (Multilingual BERT): Supports Arabic, French, and multiple other languages, AraBERT: Specialized for Arabic, including dialectal Arabic, making it useful for Darja andXLM-R (XLM-RoBERTa): A strong multilingual model capable of handling dialectal variations.

Example intents in a banking domain:

"check_account_balance"

"transfer_money"

"open_new_account"

Entity Recognition (Named Entity Recognition - NER)

During the intent detection process, The model identifies key entities relevant to the extracted intent, Example entities:"amount": "1000 EUR", "account_type": "savings" and "recipient": "John Doe"[13].

Given:

I be the set of possible **intents**: $I=\{i_1,i_2,\ldots,i_k\}$ and E be the set of **recognized entities** $E=\{e_1,e_2,\ldots,e_m\}$ then the **intent extraction function** F_3 maps a **text request** t_i to an **intent** i_j and a set of entities E_i:

$$F_3: T\times C \rightarrow (I\ ,\ E)$$

Then, for a given **text request** t_i in context c_j:

$$(i_j,E_i) = F3(t_i,c_j),\ where\ i_j \in I,\ E_i \subseteq E$$

Output model

This stage will deliver: **1.** Extracted Intent (i_j): The identified purpose of the request. **2.** Recognized Entities (E_i) – The structured data extracted from the request. 3. Context (CTX): The business domain assigned in Stage.4. Metadata (cMD): The client and user information carried forward.

Stage 5, APIGenerator:

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

This stage is responsible for mapping the extracted intent and entities from Stage 4 to a REST API request that can be executed. Since the intent represents the purpose of the request, it must be mapped to an API method that corresponds to the desired action.

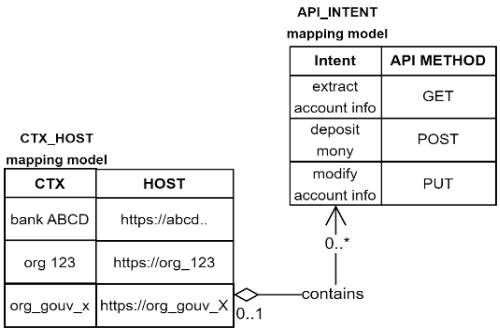

Figure 9. API Generator models

We use REST technology because of its scalability, stateless nature, compatibility with multi-platform systems, and JSON-based request structure.

The stage 4 produces tuples as:

$$(tr\_id,\{m_1,m_2,\ldots,m_n\},ctx,intent,\{e_1,e_2,\ldots,e_k\})$$

Where,

tr_id= Unique transaction ID.

$\{m_1,m_2,\ldots,m_n\}$= Metadata (client and user details).

ctx = Business context (e.g., "BANK").

intent = Extracted intent (e.g., "transfer_money").

$\{e_1,e_2,\ldots,e_k\}$ = Extracted entities.

The mapping process, using the databaseAPI_MAP_DB, proceeds with following assignations: API verb ←intent, API Host ← context, API parameters ← the entities extracted with the intent.The API_MAP_DB contains two main models of mapping;the first one is for CTX to Host (CTX2HOST) mapping and the second to map Intent to API (INT2API).

Then, using the tuple above and API_MAP_DB mapping rules, this stage produces an API request as:

Step 1: Intent-to-API INT2API

Model 2. Intent mapping

Intent tuple (stage 4)	**Intent to API mapping**

{ "tr_id": "123456",
"context": "BANK",
"intent": "transfer_money", "entities": {
"amount":"1000DA",
"account_type":"savings", "recipient": "John Doe"
}
}
	{
	"intent":"transfer_money",
	"api_method":"POST",
	"service_name":"transactions", "business_method":"transfer", "required_params":["amount", "account_type", "recipient"]
	}

Step 2: Context to host (CTX2HOST)

Model 3. context to host mapping

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article


```
{
"context": "BANK",
"host_url": "https://api.bank.com"
}
```


Step 3: Entity-to-Parameter Mapping

Model 4. entity to parameter mapping

Entities extracted:{ "amount": "1000 DA", "account_type": "savings" **Mapped API parameters:** ?amount=1000DA&account_type=savings

Step 4: API Request Construction

The final "template" API request is then constructed as:

<API_METHOD>:HTTPS://<HOST_URL>/<BUSINESS_METHOD>/[PARAMS].

Where,form the intent we extract the API verb and the BUSINESS_METHOD and from the Entities we extract the PARAMS.

Example:

GET :https://api.bank.com/transactions/transfer?amount= 1000DA&account_type=savings.

Output Model

This stage generates a structured API request model, which will be passedto the next stage.

The output model of this stage follows the structure (example):

Model 5. API Request template


```
{
"API_METHOD": "POST",
"HOST_URL": "https://api.bank.com",
"SERVICE_NAME": "transferFunds",
 "BUSINESS_METHOD": "transferMoney",
 "PARAMS":
 {
"amount": "500",
"sourceAccount": "12345",
"destinationAccount": "67890"
 }
}
```


Stage 6, API Proxy:

The proxy broker should validate whether the invoked remote host/service exists in the DSRC. If it exists, it also filters thetargeted remote service API to invoke using the API_METHOD and the BUSINESS_METHOD supplied by stage 5. After that, it maps parameters produced by stage5 with real parameters of the real API CALL.

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

The API Proxy acts as the final intermediary that validates, and routes API requests generated in Stage 5 to the intended remote service. This stage ensures the integrity and validity of the API invocation by:

1. Verifying the existence of the targeted remote service.
2. Mapping the API request's parameters with the actual requirements of the remote service.

Key Responsibilities

1. **Validation:**
   - Ensure the targeted remote host or service is listed in the **Directory of Services Requiring Continuity (DSRC)**.
   - This step minimizes the risk of invoking non-existent or invalid services during disruptions.
2. **Mapping and Transformation:**
   - Adapt the API call (constructed in Stage 5) to meet the specific requirements of the target service.
   - Parameters and methods are reconciled with the real API's expectations, ensuring compatibility and correctness.
3. **Invocation:**
   - Once validated and mapped, the API Proxy triggers the actual API call to the external service.
   - It processes the response and relays it back through the MoCaaS pipeline to the user or system.

Input and Output Models

Model 6. API proxy input | output models

Generated by stage 5	Equivalent artefacts from targeted remote service
Elements mapped directly	
API_METHOD	Get, Post, Put, Delete
HOST_URL	Remote HOST_URL
SERVICE_NAME	Remote SERVICE_NAME
Elements need some understanding	
BUSINESS_METHOD	To be resolved from the intent
PARAMS	To be extracted from entities

Example of input/ouput (json object)

Model 7: Input model from stage 5

```
{
"API_METHOD": "POST",
"HOST_URL": "https://api.bank.com",
"SERVICE_NAME": "transferFunds",
 "BUSINESS_METHOD": "transferMoney",
 "PARAMS":
 {
"amount": "500",
"sourceAccount": "12345",
"destinationAccount": "67890"
 }
}
```

Model 8. Output model by stage 6

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

```
{
"API_ENDPOINT": "https://api.bank.com/v1/transactions",
"BODY":
{
"amount": "500",
"from": "12345",
"to": "67890"
},
"HEADERS":
{
"Authorization": "Bearer <token>",
"Content-Type": "application/json"
}
}
```

FUTURE RESEARCH

As part of our ongoing work, this paper serves as the foundation for a broader research project aimed at improving IT service continuity. Each stage of the MoCaaS pipeline that we introduced will be explored in greater detail in subsequent publications. Specifically, we plan to dedicate individual papers to each of the following stages:

Voice Request Processing: The first stage, which involves the authentication and classification of voice requests, will be examined in depth. This research will focus on improving the accuracy of identity management and voice-based request processing.

Transformation and Purification: Future work will delve into how voice requests are transformed into text, focusing on natural language processing (NLP) techniques that enhance the quality and efficiency of data transformation.

Intent and Entity Extraction: We will thoroughly investigate methods for intent extraction, with a particular focus on how large language models (LLMs) can be optimized for this process, ensuring that the system accurately understands user requests.

API Mapping and Invocation: In this stage, we will explore how APIs are mapped and invoked based on extracted intent, proposing enhancements to the mapping process that increase efficiency and adaptability to various service contexts.

These future studies will refine the MoCaaS system, advancing each component to ensure a robust solution for maintaining IT service continuity in increasingly challenging environments.

CONCLUSION

This paper has presented an initial solution to address the problem of IT service continuity during Internet outages, particularly in developing countries like Algeria. We introduced the "Minimum of Continuity as a Service" (MoCaaS) system, which leverages Cloud Computing, NLP, and voice recognition technologies to ensure minimal service availability even in adverse conditions. The system integrates various technologies through a pipeline that processes user requests, extracts intent, and interacts with external APIs to maintain critical service operations.

This work represents the first step in our research. In future papers, we plan to delve deeper into each stage of the pipeline, exploring the detailed implementation and optimization of key components such as the voice recognition module, intent extraction, and API mapping. By advancing each of these stages, we aim to further enhance the reliability and resilience of the proposed solution, ensuring even greater service continuity in the face of challenges.

Journal of Information Systems Engineering and Management
2025, 10(56s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

CONFLICT OF INTEREST

No potential conflict of interest was reported by the authors.

REFRENCES

[1] Lubis, M., & Annisyah, R. C. (2020). ITSM Analysis using ITIL V3 in Service Operation in PT. Inovasi Tjaraka Buana. IOP Conference Series.
[2] C. Luca, High Availability, Fault Tolerance, and Disaster Recovery Strategies, ResearchGate,2025: https://www.researchgate.net/profile/Charlie-Luca/publication/388527642_High_Availability_Fault_Tolerance_and_Disaster_Recovery_Strategies/links/679c06a9207c0c20fa6add14/High-Availability-Fault-Tolerance-and-Disaster-Recovery-Strategies.pdf.
[3] Terfas, H. (2019). The analysis of cloud computing service level agreement (SLA) to support cloud service consumers with the SLA creation process.
[4].Bajgorić, N., Turulja, L., Ibrahimović, S., &Alagić, A. (2020). Enhancing business continuity and IT capability: System administration and server operating platforms. Taylor & Francis.
[5] Jeune Afrique, "Internet suspension in Algeria: Economic impact and telecom disruptions," June 2018: https://www.jeuneafrique.com.
[6] Bhardwaj, S., Jain, L., & Jain, S. (2010). Cloud computing: A study of infrastructure as a service (IaaS). International Journal of Engineering and Information Technology, 2(1), 60-63. https://doi.org/10.1007/s10796-011-9302-6
[7] V. Parmar, H. A. Sanghvi, R. H. Patel, et al., "A Comprehensive Study on Passwordless Authentication," *IEEE International Conference on Computing and Data Science*, 2022.
[8] Yolchuyeva, S. (2021). Novel NLP Methods for Improved Text-To-Speech Synthesis. ResearchGate.
[9] Mirishkar, S. G. (2023). Towards Building an Automatic Speech Recognition System in the Indian Context Using Deep Learning. IIIT Hyderabad.
[10] K. Kirchhoff, S. W. Li, K. Livescu, L. Maaløe, et al., "Self-Supervised Speech Representation Learning: A Review," *IEEE Journal of Selected Topics in Signal Processing*, 2022.
[11] J. B. Ilagan, J. R. Ilagan, and P. Y. Zulueta, "Optimizing Conversational Commerce Involving Multilingual Consumers Through Large Language Models' Natural Language Understanding Abilities," in Proceedings of the International Conference on Human-Computer Interaction, 2024.
[12] M. Asif, T. A. Khan, and W. C. Song, "Evaluating Large Language Models for Optimized Intent Translation and Contradiction Detection Using KNN in IBN," IEEE Access, 2025. https://ieeexplore.ieee.org/abstract/document/10855447/.
[13] T. Deußer, C. Zhao, and L. Sparrenberg, "A Comparative Study of Large Language Models for Named Entity Recognition in the Legal Domain," in Proceedings of the IEEE International Conference on Big Data, 2024.
[14] M. Magnini, G. Aguzzi, and S. Montagna, "Open-source Small Language Models for Personal Medical Assistant Chatbots," Intelligence-Based Medicine, 2025. https://ora.uniurb.it/handle/11576/2749051.
[15] W. Zhou, S. Zhang, Y. Gu, M. Chen, and H. Poon, "UniversalNER: Targeted Distillation from Large Language Models for Open Named Entity Recognition