



Hotel Finder AI Agent

Περιγραφή Συστήματος

Στόχος της εφαρμογής είναι μέσω chat ο χρήστης να ρωτάει τον πράκτορα που κατασκευάστηκε, μια προσφορά για δωμάτιο ξενοδοχείου, δίνοντάς του τον αριθμό των ατόμων, την πόλη, τη χώρα, την ημερομηνία check-in, την ημερομηνία check-out και το είδος του νομίσματος. Ο πράκτορας θα επιστρέψει στον χρήστη μια προσφορά στο νόμισμα που ζήτησε.

Τεχνολογίες και Αρχιτεκτονική

Για την ανάπτυξη του προγράμματος, χρησιμοποιήθηκαν οι εξής τεχνολογίες:

1. **LangGraph**: χρησιμοποιήθηκε για την κατασκευή του πρακτορικού συστήματος, δημιουργώντας το απαραίτητο γράφημα.
2. **Gemini-2.5-flash-preview-05-20**: το μεγάλο γλωσσικό μοντέλο που χρησιμοποιήθηκε.
3. **Amadeus**: το API για τη συλλογή στοιχείων από πόλεις και ξενοδοχεία.
4. **Exchangerate-API**: το API για τις ισοτιμίες νομισμάτων.
5. **Gradio**: η βιβλιοθήκη η οποία δημιουργεί τη διεπαφή που θα αλληλεπιδράσει μέσω chat ο χρήστης με τον ευφυή πράκτορα.

Τα εργαλεία που χρησιμοποιήθηκαν για την εφαρμογή είναι τα εξής:

Πρώτο εργαλείο: `get_iata_of_city`.

Το συγκεκριμένο εργαλείο παίρνει ως είσοδο την πόλη και τη χώρα που επιθυμεί ο χρήστης και στη συνέχεια προσπαθεί να βρει τις πόλεις που πληρούν αυτές τις προϋποθέσεις. Για κάθε πόλη που αναζητείται προσπαθεί να αντιστοιχήσει έναν κωδικό IATA.

Συγκεκριμένα, μέσω ενός try...except, προσπαθεί να βρει πρώτα την πόλη που ζητήθηκε, κάνοντας κλήση μέσω API στην υπηρεσία Amadeus. Η υπηρεσία αυτή επιστρέφει σε ένα json αρχείο πληροφορίες για την πόλη που έγινε το αίτημα. Στις πόλεις που επιστράφηκαν, προσπαθεί να βρει ποιες από αυτές είναι στη χώρα που ζητήθηκε. Αν δεν βρει καμία πόλη που να βρίσκεται σε αυτή τη χώρα, τότε θα συλλέξει όλες τις πόλεις που βρήκε ανεξαρτήτου χώρας.

Στη συνέχεια, συλλέγει το όνομα και τα στοιχεία της κάθε πόλης (π.χ. κωδικός IATA) που βρήκε και τα εισάγει σε ένα λεξικό. Τέλος, το εργαλείο επιστρέφει αυτό το λεξικό.

Αν κατά την εκτέλεση του συγκεκριμένου τμήματος κώδικα παρουσιαστεί κάποιο exception, τότε το εργαλείο θα επιστρέψει ένα λεξικό με περιεχόμενο {"error": "The city not found (no IATA code)."}.

Δεύτερο εργαλείο: `get_best_offer`.

Το συγκεκριμένο εργαλείο παίρνει ως είσοδο τον κωδικό IATA της πόλης, τον αριθμό των ατόμων, την ημερομηνία check-in και την ημερομηνία check-out. Με βάση αυτές τις παραμέτρους, προσπαθεί να βρει την καλύτερη προσφορά.

Συγκεκριμένα, μέσω ενός try...except, προσπαθεί να βρει τα ξενοδοχεία με βάση τον κωδικό IATA της πόλης, κάνοντας κλήση μέσω API στην υπηρεσία Amadeus. Η υπηρεσία αυτή επιστρέφει σε ένα json αρχείο τις πληροφορίες από όλα τα ξενοδοχεία που βρέθηκαν να έχουν ως IATA τον κωδικό που έλαβε ως όρισμα.

Στη συνέχεια μέσω μιας for επανάληψης, συλλέγονται προσφορές από τα ξενοδοχεία που βρέθηκαν. Η λίστα με τα ID των ξενοδοχείων, διασπάται σε επιμέρους τμήματα των 50 IDs το κάθε ένα. Δηλαδή, για κάθε 50 IDs ξενοδοχείων, γίνεται προσπάθεια για request στην υπηρεσία Amadeus, ώστε να επιστρέψει προσφορές από τα εκάστοτε 50 ξενοδοχεία.

Αν επιστρέψει κάποιο αποτέλεσμα, τότε τα αποτελέσματα εισάγονται στη λίστα με τις προσφορές των ξενοδοχείων και επαναλαμβάνεται η παραπάνω διαδικασία για τα επόμενα 50 ξενοδοχεία.

Αν δεν επιστρέψει κάποιο αποτέλεσμα, τότε ξαναγίνεται προσπάθεια για request. Οι προσπάθειες που γίνονται είναι συνολικά πέντε.

Αν δεν επιστραφεί καμία απάντηση και στην πέμπτη προσπάθεια, τότε οι προσπάθειες μηδενίζονται και το πρόγραμμα επαναλαμβάνει την παραπάνω διαδικασία για τα επόμενα 50 ξενοδοχεία.

Στον ευφυή πράκτορα που υλοποιήσαμε συλλέγουμε προσφορές μόνο από τα 100 πρώτα ξενοδοχεία που μας δόθηκαν από την υπηρεσία Amadeus για την εκάστοτε πόλη για λόγους απόδοσης.

Στη συνέχεια, για κάθε ξενοδοχείο που βρήκε να υπάρχει διαθέσιμη προσφορά και για κάθε προσφορά του κάθε ξενοδοχείου, συλλέγει τη τιμή αυτής της προσφοράς. Αν αυτή

η προσφορά έχει τιμή μικρότερη από την καλύτερη προσφορά που έχει δηλωθεί ως τότε, τότε δηλώνεται καλύτερη προσφορά αυτή που ελέγχθηκε και αποθηκεύεται σε ένα λεξικό. Το λεξικό με την καλύτερη προσφορά περιέχει την εξής πληροφορία:

1. Το όνομα του ξενοδοχείου.
2. Το συνολικό κόστος διαμονής.
3. Το νόμισμα του συνολικού κόστους διαμονής.
4. Την περιγραφή του δωματίου.
5. Τον αριθμό των ατόμων.
6. Την ημερομηνία check-in.
7. Την ημερομηνία check-out.

Τέλος, μετά την ολοκλήρωση της παραπάνω διαδικασίας, επιστρέφεται η καλύτερη προσφορά.

Αν κατά την εκτέλεση του συγκεκριμένου τιμήματος κώδικα παρουσιαστεί κάποιο exception, τότε το εργαλείο θα επιστρέψει ένα λεξικό με περιεχόμενο {"error": "No Offers Found"}.

Τρίτο Εργαλείο: `get_price_in_wanted_currency`.

Το συγκεκριμένο εργαλείο παίρνει ως είσοδο τη τιμή της προσφοράς που βρέθηκε, τον τύπο του νομίσματος στην οποία είναι αυτή η προσφορά και τον τύπο του νομίσματος που του έχει ενδεχομένως ζητηθεί από τον χρήστη. Υπολογίζει την ισοτιμία και στη συνέχεια επιστρέφει την τελική τιμή στο νόμισμα που του έχει ζητηθεί.

Συγκεκριμένα, χρησιμοποιείται το url της υπηρεσίας Exchangerate-API στο οποίο τοποθετείται στο τέλος του συνδέσμου ο τύπος του νομίσματος από την τιμή της προσφοράς. Η υπηρεσία επιστρέφει ένα json αρχείο με τις ισοτιμίες νομισμάτων και έπειτα ελέγχεται αν έχει επιστραφεί κάποια απάντηση. Αν έχει επιστραφεί κάποια απάντηση, τότε ελέγχεται αν υπάρχει η ισοτιμία στο νόμισμα που ζητήθηκε από τον χρήστη. Αν υπάρχει, τότε επιστρέφεται η ισοτιμία { 'price': str(price * x) } που είναι το γινόμενο ανάμεσα στην τιμή της προσφοράς και την ισοτιμία. Σε οποιαδήποτε άλλη περίπτωση, επιστρέφεται το λεξικό {"error": "Could Not Estimate Price"}.

Τέταρτο Εργαλείο: `about_you`.

Αυτό το εργαλείο επιστρέφει ένα λεξικό με μήνυμα που περιέχει ως απάντηση τις πληροφορίες σχετικά με τον agent.

Σε κάθε εργαλείο, χρησιμοποιείται μια κλάση η οποία δίνει επιπλέον πληροφορίες στο γλωσσικό μοντέλο ώστε να κατατοπιστεί καλύτερα σχετικά με τη μορφή και τη σημασία των παραμέτρων του εκάστοτε εργαλείου. Συγκεκριμένα:

Πριν από το πρώτο εργαλείο, χρησιμοποιείται η κλάση ToolFields0, η οποία περιέχει τα εξής πεδία:

- **city: str** - μια περιγραφή για την παράμετρο της πόλης και
- **country: str** - μια περιγραφή για την παράμετρο της χώρας.

Πριν από το δεύτερο εργαλείο, χρησιμοποιείται η κλάση ToolFields1, η οποία περιέχει τα εξής πεδία:

- **iataCityCode: str** - μια περιγραφή για τον κωδικό IATA της πόλης,
- **people: int** - μια περιγραφή για τον αριθμό των ατόμων,
- **checkInDate: str** - μια περιγραφή για την ημερομηνία check-in,
- **checkOutDate: str** - μια περιγραφή για την ημερομηνία check-out.

Πριν από το τρίτο εργαλείο, χρησιμοποιείται η κλάση ToolFields2, η οποία περιέχει τα εξής πεδία:

- **wantedCurrency: str** – μια περιγραφή για το νόμισμα που έχει ζητηθεί,
- **currentCurrency: str** – μια περιγραφή για το νόμισμα στο οποίο έχει γίνει η προσφορά,
- **price: float** – μια περιγραφή για την τιμή της προσφοράς.

Επίσης, σε κάθε εργαλείο, υπάρχει και από ένα σχόλιο που περιγράφει συνοπτικά την ιδιότητα του κάθε εργαλείου, το οποίο βοηθάει το γλωσσικό μοντέλο να κατανοήσει τη χρησιμότητα του εργαλείου αυτού.

Μεγάλο Γλωσσικό Μοντέλο (LLM).

Το γλωσσικό μοντέλο που χρησιμοποιείται είναι το Gemini 2.5 Flash (Preview). Το μοντέλο δηλώνεται στη μεταβλητή με όνομα `llm`. Μετά εισάγουμε σε μια λίστα με όνομα `tools` τα εργαλεία που έχουν κατασκευαστεί, και έπειτα δηλώνουμε αυτά τα εργαλεία στο μοντέλο.

LangGraph.

Αρχικά, η κλάση `AgentState` χρησιμοποιείται για να προσδιορίσει την κατάσταση του πράκτορα, έχοντας τις εξής παραμέτρους:

- **messages: Annotated[Sequence[BaseMessage], add_messages]** – που είναι το ιστορικό των μηνυμάτων,
- **number_of_steps: int** – ο αριθμός των βημάτων έχει υλοποιήσει στο γράφημα.

Η συνάρτηση `calling_tools(...)` καλεί τα διαθέσιμα εργαλεία. Συγκεκριμένα, γίνεται προσπέλαση των εργαλείων που το γλωσσικό μοντέλο έχει ζητήσει να κληθούν. Έπειτα, τα εργαλεία που έχει ζητήσει το μοντέλο ενεργοποιούνται, και αποθηκεύεται:

- η απάντηση,
- το όνομα
- και το ID της κλήσης

του καθενός εργαλείου που ζήτησε το γλωσσικό μοντέλο, σε μία λίστα. Επιστρέφεται η λίστα αυτή μέσω ενός `dictionary` με key το “`messages`”.

Έπειτα, μέσω της συνάρτησης `calling_model(...)` καλείται το μοντέλο. Συγκεκριμένα, παίρνει το μήνυμα εισόδου, το μήνυμα αυτό γίνεται `prompt` και δίνεται ως είσοδος στο γλωσσικό μοντέλο που καλείται στη συνέχεια. Επιστρέφει την απάντηση του γλωσσικού μοντέλου.

Στη συνάρτηση `continue_or_end(...)` ελέγχεται αν στην απάντηση του γλωσσικού μοντέλου υπάρχει `tool call`. Αν υπάρχει, τότε συνεχίζεται η διαδικασία και γίνεται κλήση σε κάποιο εργαλείο. Διαφορετικά, επιστρέφεται το `string` “`end`” που συμβολίζει και το τέλος των μεταβάσεων στο γράφημα.

Ορίζουμε ένα γράφημα.

Προσθέτουμε έναν κόμβο με όνομα `thought_llm` και τη συνάρτηση που καλεί το γλωσσικό μοντέλο.

Προσθέτουμε έναν κόμβο με όνομα `tool_set` και τη συνάρτηση που καλεί τα εργαλεία.

Ο πρώτος κόμβος που θα κληθεί, θα είναι ο κόμβος `thought_llm`.

Στη συνέχεια, ορίζουμε τις μεταβάσεις υπό συνθήκη. Δηλαδή, μετά τον κόμβο `thought_llm`, θα κληθεί η συνάρτηση `continue_or_end(...)`, ώστε να αποφασίσει ο πράκτορας εάν θα συνεχίσει μεταβαίνοντας στον κόμβο που αντιστοιχεί σε κλήση `tool` ή θα σταματήσει μεταβαίνοντας στην κατάσταση τερματισμού (“`END`”). Αυτό ορίζεται από τα κλειδιά “`continue`” και “`end`”.

Προσθέτουμε άλλη μια μετάβαση από τον κόμβο `tool_set` προς το `thought_llm`, ώστε να μπορεί να επιστρέψει το αποτέλεσμα της κλήσης των εργαλείων στο γλωσσικό μοντέλο για επεξεργασία και ανάλυση.

Δημιουργείται το γράφημα.

Gradio.

Χρησιμοποιήθηκε το Gradio για την επικοινωνία του `agent` με τον χρήστη.

Η συνάρτηση `predict(...)` λαμβάνει ως είσοδο το μήνυμα του χρήστη και το ιστορικό της συνομιλίας.

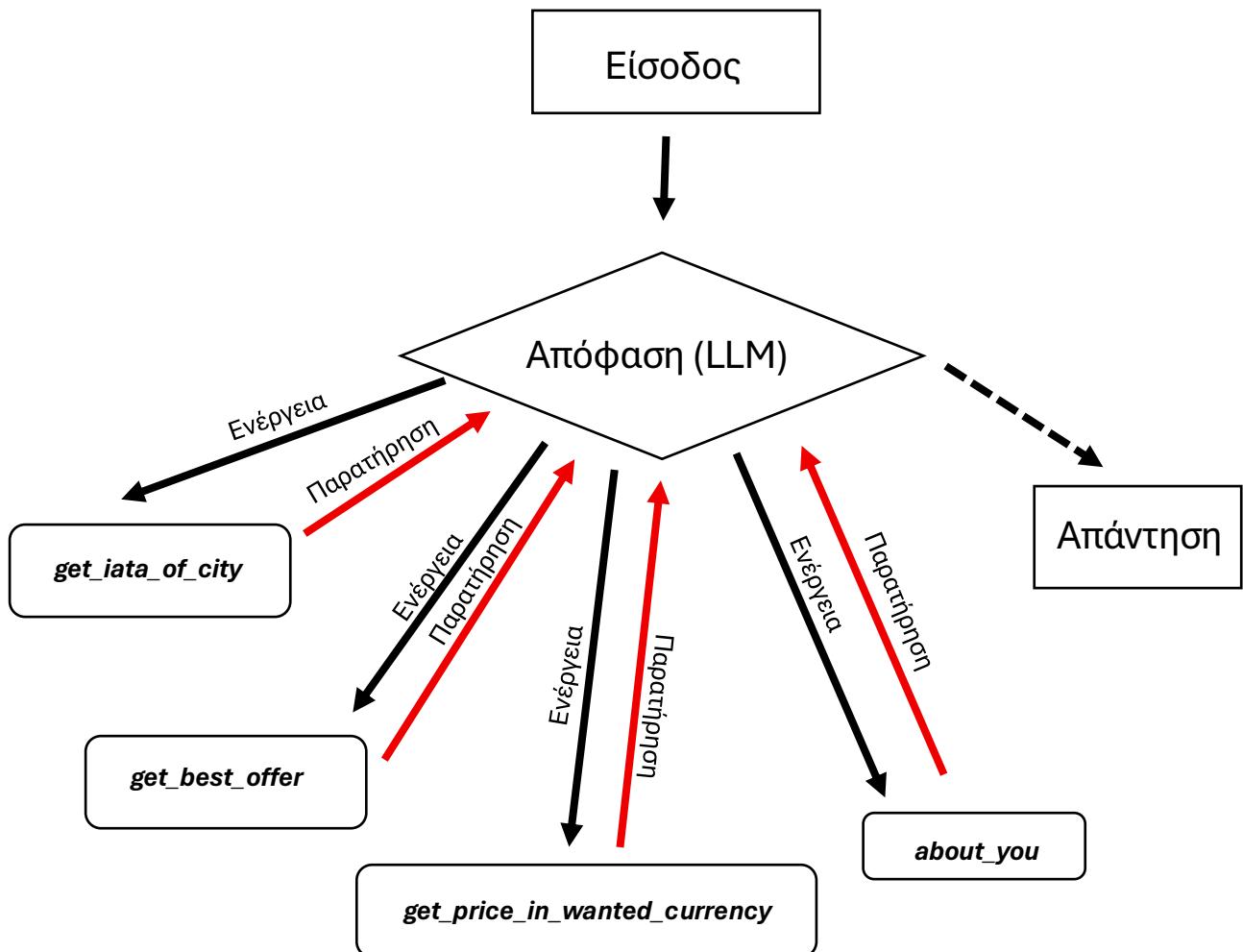
Δίνονται οδηγίες σχετικά με το πως πρέπει να συμπεριφερθεί ο agent (system prompt). Στη συνέχεια, αποθηκεύονται σε μια λίστα όλα τα μηνύματα που έχει δώσει ο χρήστης καθώς και όλες οι απαντήσεις του agent που υπάρχουν στη συνομιλία. Τέλος, αποθηκεύεται το τελευταίο μήνυμα που έστειλε ο χρήστης.

Μετά η λίστα αυτή αποθηκεύεται σε ένα λεξικό με key το “messages”.

Έπειτα, διατηρείται η απάντηση του εκάστοτε ερωτήματος που δόθηκε από τον agent στο τελευταίο state και επιστρέφεται το περιεχόμενο αυτής της απάντησης.

Τέλος, δηλώνεται ο τύπος της γραφικής διεπαφής, ο οποίος είναι Chat Interface.

Διάγραμμα Ροής



Το διάγραμμα ροής διαθέτει τους εξής κόμβους:

- **Είσοδος:** η είσοδος στο σύστημα που είναι και το μήνυμα/αίτημα από τον χρήστη.
- **Απόφαση (LLM):** Το σημείο στο οποίο ο agent αποφασίζει εάν θα καλέσει ένα από τα διαθέσιμα εργαλεία ή αν θα επιστρέψει την τελική απάντηση.
- **get_iata_of_city, get_best_offer, get_price_in_wanted_currency, about_you:** Τα διαθέσιμα εργαλεία.
- **Απάντηση:** Τελική απάντηση στον χρήστη (τερματισμός λειτουργίας).

Επίσης, το διάγραμμα ροής διαθέτει τις εξής συνδέσεις:

- **Ενέργεια:** είναι η απόφαση να εκτελεστεί ένα συγκεκριμένο εργαλείο από το γλωσσικό μοντέλο με βάση την είσοδο που του δόθηκε.
- **Παρατήρηση:** είναι η απάντηση του εργαλείου που στέλνεται στο γλωσσικό μοντέλο για περαιτέρω επεξεργασία.

Προβλήματα που αντιμετωπίστηκαν

Κατά τη δημιουργία του agent, υπήρξαν ζητήματα που χρειάστηκε να αντιμετωπιστούν.

Κάποια από τα κυριότερα προβλήματα που αντιμετωπίστηκαν, ήταν τα εξής:

- **Επιλογή του γλωσσικού μοντέλου.** Η επιλογή ενός γλωσσικού μοντέλου ήταν απαιτητική και κρίσιμη για τη δημιουργία του agent. Συγκεκριμένα, αρκετά γλωσσικά μοντέλα είτε δεν παρείχαν πολλά διαθέσιμα tokens για είσοδο, είτε τα όρια για υποβολή αιτημάτων σε ένα συγκεκριμένο χρονικό πλαίσιο (όπως ώρα ή ημέρα) ήταν περιορισμένα. Επομένως, το γλωσσικό μοντέλο που επιλέχθηκε, έπρεπε να μπορεί να ανταπεξέρχεται στις απαιτήσεις, χωρίς να διαθέτει πολύ περιορισμένα όρια.
- **Prompt engineering.** Η δήλωση του system prompt, οι κλάσεις ToolFields0, ToolFields1, ToolFields2, καθώς και τα σχόλια σε κάθε εργαλείο που δίνουν οδηγίες στο γλωσσικό μοντέλο, είναι κρίσιμα ώστε το μοντέλο να καταλαβαίνει όσο το δυνατόν καλύτερα αυτό που του ζητείται καθώς και ποια εργαλεία πρέπει να επιλέξει κάθε φορά. Για παράδειγμα, κατά τις δοκιμές μας, ο agent μπορεί να εμφανίζε στον χρήστη παραμέτρους εργαλείων οι οποίες έπρεπε να συμπληρωθούν αυτόματα από τον agent και να μην ζητούνται από τον χρήστη. Οπότε, το περιεχόμενο των παραπάνω έπρεπε να είναι στοχευμένο, χωρίς ασάφειες.
- **Εύρεση κατάλληλων APIs.** Η εύρεση των APIs ήταν ιδιαίτερα σημαντική και απαιτητική, αφού έπρεπε ο agent να συλλέγει πληροφορίες σχετικά με τις προσφορές ξενοδοχείων και για τις ισοτιμίες των νομισμάτων. Επίσης ήταν ιδιαίτερα κρίσιμο να αποφασιστεί το πώς θα συνδυαστούν, ώστε να εκπληρώνουν τον σκοπό για τον οποίο δημιουργήθηκε ο συγκεκριμένος agent.

Αξιολόγηση/Δοκιμή με διαφορετικά User Inputs

Στην πρώτη συνομιλία, ο χρήστης ζητάει μια προσφορά για δίκλινο δωμάτιο στη Βαλένθια της Ισπανίας, από τις 5 Αυγούστου 2025 για τρεις ημέρες.

I want to find for me a room in Valencia, Spain.

Ok, I can help you with that. When would you like to check in and check out? And how many people will be staying?

Εικόνα πρώτης συνομιλίας - Μέρος 1

I want a room for two persons on August 5 2025 for three days.

I have found a room for you at the ibis Valencia Bonaire!

Here are the details:

- Check-in Date: 2025-08-05
- Check-out Date: 2025-08-08
- Number of people: 2
- Room Description: ADVANCE SAVER-Room only, Room for 1 or 2 persons
- Total Accommodation Amount: 132.75 EUR

Εικόνα πρώτης συνομιλίας - Μέρος 2

Give the price in the UK currency.

The price of the offer in the UK currency is 111.86 GBP.

Εικόνα πρώτης συνομιλίας – Μέρος 3

Στη δεύτερη συνομιλία, ο χρήστης προσπαθεί να ζητήσει από τον agent να του πει κάποιο ανέκδοτο, το οποίο απορρίπτεται, διότι ο σκοπός του agent είναι να βρίσκει και να δίνει προσφορές για ξενοδοχεία, εμφανίζοντας την τιμή της προσφοράς σε οποιοδήποτε νόμισμα ζητήσει ο χρήστης.

Τι μπορείς να κάνεις;

Μπορώ να αναζητήσω τις καλύτερες προσφορές ξενοδοχείων για την πόλη που θέλετε, βάσει των ταξιδιωτικών σας κριτηρίων. Επίσης, μπορώ να σας δώσω την προσφορά στο νόμισμα που προτιμάτε.

Μπορείς να μου πεις ανέκδοτο;

Λυπάμαι, αλλά δεν μπορώ να σου πω ανέκδοτα. Είμαι εδώ για να σε βοηθήσω να βρεις ξενοδοχεία και πληροφορίες σχετικά με αυτά.

Εικόνα συνομιλίας με τον agent. Ο χρήστης ζητάει από τον agent να του πει ένα ανέκδοτο αλλά το αίτημά του απορρίπτεται.

Ακολουθεί μια ολοκληρωμένη συνομιλία με τον agent:

What you can do for me ?

I can help you find the best hotel offers in a city for specific dates and number of people. I can also convert the price of an offer to your desired currency. What can I help you with today?

I am looking for a triple room in Rome for my next month vacation. Could you help me?

I can definitely help you with that! To find the best offer for a triple room in Rome, I need to know your exact check-in and check-out dates for your vacation next month. Could you please provide them?

Yes of course, the check-in is on 21/07/2025 and the check-out is on 26/07/2025.

I found a great offer for you in Rome!

Hotel Name: Best Western Globus Hotel

Total Accommodation Amount: 1362.50 EUR

Room Description: ADV PURCHASE FULL PAYNO CANCEL*NO CHANGES - 2 QUEEN BEDS, COMFORT ROOM STANDARD

Check-in Date: 2025-07-21

Check-out Date: 2025-07-26

People: 3

Would you like to convert the price to a different currency?

Θα μπορούσες να βρεις και μια ακόμα προσφορά για εμένα;

Δυστυχώς, μπορώ να βρω μόνο την καλύτερη προσφορά που είναι διαθέσιμη για τις ημερομηνίες και τον αριθμό ατόμων που μου δώσατε. Δεν μπορώ να βρω επιπλέον προσφορές αυτή τη στιγμή.

Ωραία θέλω να με βοηθήσεις γιατί ψάχνω ένα δίκλινο δωμάτιο στη Θεσσαλονίκη την επόμενη εβδομάδα.

Για να σας βοηθήσω να βρω ένα δίκλινο δωμάτιο στη Θεσσαλονίκη την επόμενη εβδομάδα, θα χρειαστώ τις ακριβείς ημερομηνίες check-in και check-out. Μπορείτε να μου τις πείτε;

To check-in είναι στις 18/06/2025 και το check-out στις 25/06/2025.

Βρήκα μια εξαιρετική προσφορά για εσάς στη Θεσσαλονίκη!

'Όνομα Ξενοδοχείου: DAIOS LUXURY LIVING

Συνολικό Ποσό Διαμονής: 1715.00 EUR

Περιγραφή Δωματίου: Καλύτερη διαθέσιμη τιμή, δωμάτιο Comfort με θέα στο αίθριο, διπλό κρεβάτι, wi-fi

Ημερομηνία Check-in: 18-06-2025

Ημερομηνία Check-out: 25-06-2025

Άτομα: 2

Θα θέλατε να μετατρέψετε την τιμή σε διαφορετικό νόμισμα;

Θα ήθελα να μου βρεις μια προσφορά για ξενοδοχείο στη Νέα Υόρκη για τις ίδιες ημερομηνίες.

Βρήκα μια προσφορά για εσάς στη Νέα Υόρκη!

'Όνομα Ξενοδοχείου: FOUR POINTS PLAINVIEW

Συνολικό Ποσό Διαμονής: 1748.66 USD

Περιγραφή Δωματίου: Long Term Stay rate Traditional King, 1 King, Sofa bed, Minifridge, Microwave, 350sqft/32sqm, Wireless

Ημερομηνία Check-in: 2025-06-18

Ημερομηνία Check-out: 2025-06-25

Άτομα: 2

Θα θέλατε να μετατρέψετε την τιμή σε διαφορετικό νόμισμα;

Nαι θα ήθελα να μετατρέψεις τη τιμή σε €.

Η τιμή των 1748.66 USD μετατράπηκε σε 1529.55 EUR.

Τι μπορείς να κάνεις για εμένα ;

Μπορώ να σας βοηθήσω να βρείτε τις καλύτερες προσφορές ξενοδοχείων σε μια πόλη για συγκεκριμένες ημερομηνίες και αριθμό ατόμων.
Επίσης, μπορώ να μετατρέψω την τιμή μιας προσφοράς στο νόμισμα που επιθυμείτε. Πώς μπορώ να σας εξυπηρετήσω σήμερα;