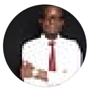
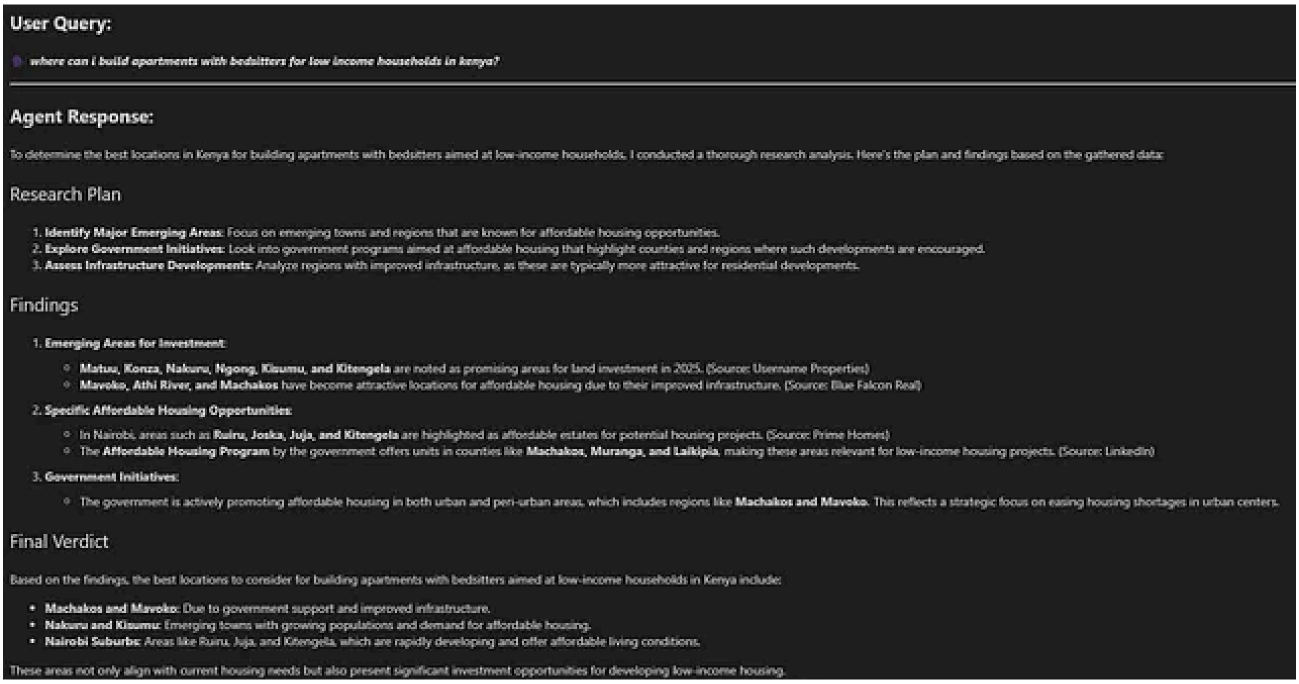
Building an AI Agentic System For the Kenyan Real Estate Market.



John Wafula · Follow

10 min read · 4 hours ago



User Query:

where can i build apartments with bedsitters for low income households in kenya?

Agent Response:

To determine the best locations in Kenya for building apartments with bedsitters aimed at low-income households, I conducted a thorough research analysis. Here's the plan and findings based on the gathered data:

Research Plan

1. **Identify Major Emerging Areas:** Focus on emerging towns and regions that are known for affordable housing opportunities.
2. **Explore Government Initiatives:** Look into government programs aimed at affordable housing that highlight counties and regions where such developments are encouraged.
3. **Assess Infrastructure Developments:** Analyze regions with improved infrastructure, as these are typically more attractive for residential developments.

Findings

1. **Emerging Areas for Investment:**
   - **Matuu, Konza, Nakuru, Ngong, Kisumu, and Kitengela** are noted as promising areas for land investment in 2025. (Source: Username Properties)
   - **Mavoko, Athi River, and Machakos** have become attractive locations for affordable housing due to their improved infrastructure. (Source: Blue Falcon Real)
2. **Specific Affordable Housing Opportunities:**
   - In Nairobi, areas such as **Ruiru, Joska, Juja, and Kitengela** are highlighted as affordable estates for potential housing projects. (Source: Prime Homes)
   - The **Affordable Housing Program** by the government offers units in counties like **Machakos, Muranga, and Laikipia**, making these areas relevant for low-income housing projects. (Source: LinkedIn)
3. **Government Initiatives:**
   - The government is actively promoting affordable housing in both urban and peri-urban areas, which includes regions like **Machakos and Mavoko.** This reflects a strategic focus on easing housing shortages in urban centers.

Final Verdict

Based on the findings, the best locations to consider for building apartments with bedsitters aimed at low-income households in Kenya include:

- **Machakos and Mavoko:** Due to government support and improved infrastructure.
- **Nakuru and Kisumu:** Emerging towns with growing populations and demand for affordable housing.
- **Nairobi Suburbs:** Areas like Ruiru, Juja, and Kitengela, which are rapidly developing and offer affordable living conditions.

These areas not only align with current housing needs but also present significant investment opportunities for developing low-income housing.

Agent Response On the Best Places To build bedsitters

A friend of mine wanted to invest in real estate — buy land and build apartments in Kenya — but he faced a major challenge: he didn't know which location was the best. He had bits and pieces of information, but nothing concrete or data-driven to guide his decision. The real estate market can be overwhelming, with prices fluctuating, regulations varying, and future potential often unclear. Seeing this gap, I decided to build an AI-driven agentic system specifically designed to help investors like him make informed, data-backed decisions in the Kenyan real estate market. This system aims to turn scattered insights into actionable intelligence, making property investment smarter and more strategic. Additionally the system can still house hunt and help do comparata analysis on investment spots.

Requirements

You can install these dependencies in a Python 3.11 environment.

```
.env
OPENAI_API_KEY= "Your Open AI Key"
TAVILY_API_KEY = "Your Tavily Key Here"
```

```
requirements.txt
langgraph==0.2.56
langchain-community==0.3.9
langchain-openai==0.2.11
tavily-python==0.5.0
promptquality==0.69.1
```

Defining the Problem

For this demonstration we will build a Real Estate agent that thinks through and acts on problems within a real estate dataset. We will design a process that takes a query, dissects it into more specific queries, uses Tavily to search the web, and then examines the findings.
We employ the ReAct agent, which collaborates with the Tavily API to think through and solve issues, to examine the outcomes.

We will make a new Jupyter Notebook agent in vscode.Ipynb.
We will import Tavily, a web search tool, and a prebuilt ReAct agent. we will use multiple agents for different tasks, even if in this example we use the same agent for every step. The finest aspect? Later examples will allow us to further personalize it.

```
from langchain_openai import ChatOpenAI
from langchain_community.tools.tavily_search import TavilySearchResults
from langgraph.prebuilt import create_react_agent

system_prompt = '''You are a helpful Real Estate expert named Wakasiaka in year
Then you use tools to get answers to the questions. Finally you use the answers

llm = ChatOpenAI(model = "gpt-4o-mini")
tools = [TavilySearchResults(max_results = 3)]
agent_executor = create_react_agent(llm, tools, state_modifier = system_prompt)
```

State management

Let's now discuss how our agent manages all of its responsibilities. It is comparable to a clever three-part to-do list approach.
We must first find a means of monitoring the agent's plans. We'll make use of a straightforward text string with a list of actions. This functions similarly to a checklist of things the agent must do.
Second, we want to recall its past performance and the outcomes of each challenge. We'll use a list of pairs — or, in programming parlance, tuples — for this. Each pair includes the action performed as well as the outcome of that action. Last but not least, we must save two additional crucial pieces of data: the initial query (the input) and the final response (the response), which is provided after the agent completes its task.
With this configuration, our agent has all it needs to work efficiently.

```
import operator
from pydantic import BaseModel, Field
from typing import Annotated,List, Tuple

class PlanExecute(TypedDict):
    input: str
```

```
    plan: List[str]
    plan_steps: Annotated[List[Tuple], operator.add]
    response: str

class Plan(BaseModel):
    """ plan to follow in the future """

    steps: List[str] = Field(
        description= "diffeerent steps to follow, should be in sorted order"
    )
```

The PlanExecute class, a dictionary type, manages an execution process,including input, plan steps, previous steps, and a response. The Plan class, usingPydantic, defines a structured plan with steps that should be followed in a sorted order.

The planning step is where our agent will begin to tackle a research question. We'll use a special feature called **function calling** to create this plan. Let's break down how it works.

First, we create a template for how our agent should think. We tell it that it's a real estate agent working in October 2024, and its job is to break down big questions into smaller, manageable steps. This template, called planner_prompt (See Fig 3.3), gives our agent clear instructions: create a simple, step-by-step plan where each step leads logically to the next. Ensure that no steps are missing or unnecessary. The final step should give us our answer. The code sets this up by using ChatPromptTemplate, which has two main parts:

• A system message that explains the agent's role and how it should plan

• A placeholder for the messages we'll send it

We then connect this template to ChatOpenAI using gpt-4o-mini with temperature set to 0 for consistent results. We take gpt-4o-mini being low on cost. The "structured output" part means the plan will come out in a specific format we can easily work with. When we test it with a real question like “*where can i build apartments with bedsitters for low income households in kenya*?” the agent will create a detailed plan for researching this investment decision. Each step will help gather the information needed to make an informed recommendation about Bedsitters based on the current electric vehicle market conditions. Think of it like creating a research roadmap. We’re giving our agent the tools and guidelines it needs to break down complex questions into manageable research tasks.

```
from langchain_core.prompts import ChatPromptTemplate
planner_prompt = ChatPromptTemplate.from_messages(
    [
        (
            "system",
            """ You are a real estate agent working in January 2024. For the giv
This plan should involve individual tasks, that if executed correctly will yield
 Do not add any superfluous steps. The result of the final step should be the fi
 Make sure that each step has all the information needed - do not skip steps.
At the end use the info collected to give the final answer to the main question
        ),
        ("placeholder", "{messages}"),
    ]
)
```

```
planner = planner_prompt | ChatOpenAI(
    model = "gpt-4o-mini", temperature= 0,openai_api_key = openai_api_key
).with_structured_output(Plan)

planner.invoke(
    {
        "messages": [
            ("user", "where can i build apartments with bedsitters for low incom
        ]
    }
)
```

Re-planning can be thought of as the agent’s capacity to modify its approach in light of its prior knowledge. This is comparable to how we could modify our study strategy in light of fresh data. Let’s dissect how this operates.

First, we create two types of possible actions the agent can take:

• **Response:** When the agent has enough information to answer the user's question

- **Plan:** When the agent needs to do more research to get a complete answer

The re-planning prompt is like giving our agent a structured way to think about what to donext. It looks at three things:

• The original question (objective)

• The initial plan it made

• What steps have already been completed and what was learned

Using this information, the agent can decide to either:

• Create new steps to gather more needed information

- Give a final answer if it has enough information

The clever part is that the agent won't repeat steps it's already done. It focuses only on what still needs to be investigated. This makes the research process more efficient and prevents redundant work. It's like having a research assistant who can intelligently adjust their approach based on what they've already discovered. This process helps our agent stay focused and efficient, only pursuing new information when needed and knowing when it's time to provide a final answer to the user.

We connect this re-planning ability to gpt-4o with the temperature set to 0. By setting the temperature to 0 , we force the model to generate the same response for the same input. This helps us in making experiments reproducible.

```python
from typing import Union

class Response(BaseModel):
    '''Response To user'''
    response: str

class Act(BaseModel):
    """ Action to perform"""

    action: Union[Response,Plan] = Field(
        description= " Action to perform. If you waant to respond to user , use
        "If you need to further use tools to get the answer, use plan "
    )
replanner_prompt = ChatPromptTemplate.from_template(
    """For the given objective, come up with a simple step by step plan. \
This plan should involve individual tasks, that if executed correctly will yield
The result of the final step should be the final answer. Make sure that each ste
the information needed - do not skip steps.

Your objective was this:
{input}

Your original plan was this:
{plan}

You have currently done the following steps:
{past_steps}

Update your plan accordingly. If no more steps are needed and you can return to
then respond with that. otherwise, fill out the plan. Only add steps the plan th
need to be done. Do not return previously done steps as part of the plan.

"""
)
replanner = replanner_prompt | ChatOpenAI(
    model = "gpt-4o", temperature=0, openai_api_key = openai_api_key
).with_structured_output(Act)
```

Create the graph

```python
from typing import Literal
from langgraph.graph import END

async def execute_step(state: PlanExecute):
    plan = state["plan"]
```

```python
        plan_str = "\n".join(f"{i+1}" for i , step in enumerate(plan))
        task = plan[0]
        task_formatted =f"""For the following plan:
        {plan_str}\n\nYou are tasked with executing step {1}, {task}."""
        agent_response = await agent_executor.ainvoke(
            {"messages": [("user", task_formatted)]}
        )
        return {
            "past_steps": [(task, agent_response["messages"][-1].content)],
        }

    async def plan_step(state: PlanExecute):
        plan = await planner.ainvoke({"messages": [("user",state["input"])]})
        return {"plan":plan.steps}

    async def replan_step(state: PlanExecute):
        output = await replanner.ainvoke(state)
        if isinstance(output.action, Response):
            return {"response": output.action.response}
        else:
            return {"plan": output.action.steps}

    def should_end(state: PlanExecute):
        if "response" in state and state["response"]:
            return END
        else:
            return "agent"
```

The different tasks are managed by the execute_step function. The agent works on the first item from our plan when it has been correctly formatted. It's similar to assigning a research assistant a certain task and receiving their results. The agent records its actions and what it discovered.

It all starts with the plan_step function. It develops the initial research plan in response to a question. This is similar to drafting a first draft of a solution to the issue.

The replan_step function is where the agent decides what to do next. After completing a task, it looks at what it has learned and either:

• Creates new steps if more research is needed

- Provides a final answer if it has enough information

The should_end function, which functions similarly to a checkpoint, comes last. It determines if we have a definitive response prepared. The process is over if we do. If not, it instructs the agent to carry on with their work. All of these features are seen in the code sample shown above.
Using StateGraph, we build a map that directs our agent through its investigation using the various options available to it. This is how it goes:

First, we create the basic structure of the workflow with its three main stops:
• A planning station ("planner")
• A research station ("agent")
• A reviewing station ("replan")

Then, we connect these stations in a logical order:
1. Everything starts at the planning station
2. From planning, the agent moves to doing research
3. After research, it goes to reviewing what it learned
At the reviewing station, the agent makes an important decision:
• Either continue with more research if needed
• Or finish up if it has a complete answer

This establishes a seamless cycle whereby the agent can carry on investigating until it has all the information required to address the first query. It's similar to having a knowledgeable research assistant who understands when to stop looking and when they've gathered enough data. Like every other tool in our system, we finally put this workflow into a user-friendly format. This prepares our research agent to take on actual queries and offer comprehensive, well-thought-out responses.

```
from langgraph.graph import StateGraph, START

workflow = StateGraph(PlanExecute)

workflow.add_node("planner", plan_step)
workflow.add_node("agent", execute_step)
```

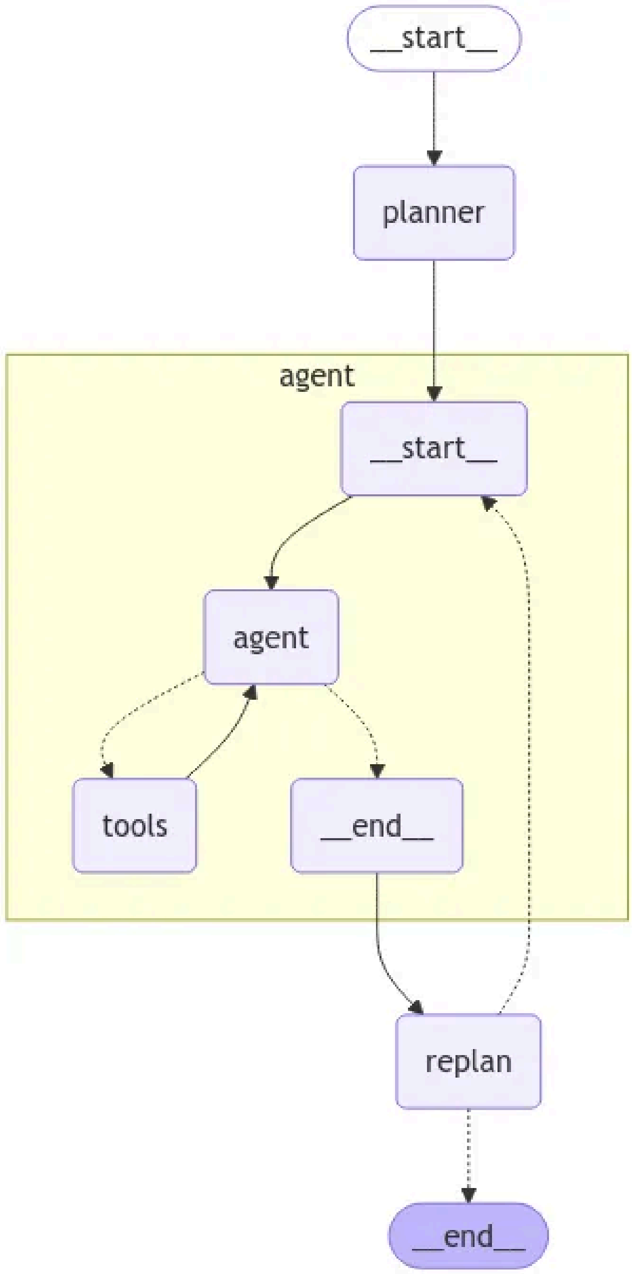
```
workflow.add_node("replan", replan_step)
workflow.add_edge(START, "planner")
workflow.add_edge("planner", "agent")
workflow.add_edge("agent", "replan")
workflow.add_edge(START, "planner")

workflow.add_conditional_edges(
   "replan",
   should_end,
   ["agent", END],
)

app = workflow.compile()
```

Visualize

```
from IPython.display import Image, display
display(Image(app.get_graph(xray = True).draw_mermaid_png()))
```



Testing Our Agent

```
from langchain.schema import AIMessage, HumanMessage
from IPython.display import display, Markdown

test_query = "where can i build apartments with bedsitters for low income househ

# Run the agent
response = agent_executor.invoke({"messages": [{"role": "user", "content": test_
messages = response.get("messages", [])
human_messages = [msg for msg in messages if isinstance(msg, HumanMessage)]
user_query = human_messages[-1].content if human_messages else "No query found."
ai_messages = [msg for msg in messages if isinstance(msg, AIMessage)]
ai_response = ai_messages[-1].content if ai_messages else "No response received.
display(Markdown(f"### **User Query:**\n\n🗣️ ***{user_query}***\n\n---\n\n### **
Response:**\n\n{ai_response}"))
```

The OutPut

User Query:

where can i build apartments with bedsitters for low income households in kenya?

Agent Response:

To determine the best locations in Kenya for building apartments with bedsitters aimed at low-income households, I conducted a thorough research analysis. Here's the plan and findings based on the gathered data:

Research Plan

1. **Identify Major Emerging Areas:** Focus on emerging towns and regions that are known for affordable housing opportunities.
2. **Explore Government Initiatives:** Look into government programs aimed at affordable housing that highlight counties and regions where such developments are encouraged.
3. **Assess Infrastructure Developments:** Analyze regions with improved infrastructure, as these are typically more attractive for residential developments.

Findings

1. **Emerging Areas for Investment:**
   - **Matuu, Konza, Nakuru, Ngong, Kisumu, and Kitengela** are noted as promising areas for land investment in 2025. (Source: Username Properties)
   - **Mavoko, Athi River, and Machakos** have become attractive locations for affordable housing due to their improved infrastructure. (Source: Blue Falcon Real)
2. **Specific Affordable Housing Opportunities:**
   - In Nairobi, areas such as **Ruiru, Joska, Juja, and Kitengela** are highlighted as affordable estates for potential housing projects. (Source: Prime Homes)
   - The **Affordable Housing Program** by the government offers units in counties like **Machakos, Muranga, and Laikipia**, making these areas relevant for low-income housing projects. (Source: LinkedIn)
3. **Government Initiatives:**
   - The government is actively promoting affordable housing in both urban and peri-urban areas, which includes regions like **Machakos and Mavoko**. This reflects a strategic focus on easing housing shortages in urban centers.

Final Verdict

Based on the findings, the best locations to consider for building apartments with bedsitters aimed at low-income households in Kenya include:

- **Machakos and Mavoko:** Due to government support and improved infrastructure.
- **Nakuru and Kisumu:** Emerging towns with growing populations and demand for affordable housing.
- **Nairobi Suburbs:** Areas like Ruiru, Juja, and Kitengela, which are rapidly developing and offer affordable living conditions.

These areas not only align with current housing needs but also present significant investment opportunities for developing low-income housing.

Let Us try something more quantifiable like an economic househunter looking for a one bedroom apartment that costs Ksh. 15,000

```python
from langchain.schema import AIMessage, HumanMessage
from IPython.display import display, Markdown

test_query = "where can i find a good onebedroom apartment for Ksh. 15000 in Nairobi?"

# Run the agent
response = agent_executor.invoke({"messages": [{"role": "user", "content": test_query}]})
messages = response.get("messages", [])
human_messages = [msg for msg in messages if isinstance(msg, HumanMessage)]
user_query = human_messages[-1].content if human_messages else "No query found."
ai_messages = [msg for msg in messages if isinstance(msg, AIMessage)]
ai_response = ai_messages[-1].content if ai_messages else "No response received."
display(Markdown(f"### **User Query:**\n\n ***{user_query}***\n\n---\n\n### **Agent Response:**\n\n{ai_response}"))
```

✓ 7.7s

User Query:

where can i find a good onebedroom apartment for Ksh. 15000 in Nairobi?

Agent Response:

Here are some options for one-bedroom apartments available for rent in Nairobi for Ksh. 15,000:

1. **PropertyPro** - A 1-bedroom mini flat located near Next Gen Mall in South C, Lang'ata, Nairobi. The rent is Ksh. 15,000. You can find more details here.
2. **Hata** - More than 957 one-bedroom houses and apartments are listed for rent in Nairobi, with options starting from Ksh. 15,000 per month, particularly in South B. Check the listings here.
3. **Private Property Kenya** - A 1-bedroom apartment located in Mountain View Estate, Nairobi, also renting for Ksh. 15,000/month. More details can be found here.

These listings should help you find a suitable one-bedroom apartment within your budget in Nairobi.

User Query:

Is it advisable to buy land in Moyale for quick returns?

Agent Response:

Based on the gathered information, here's a detailed analysis regarding the advisability of buying land in Moyale for quick returns:

1. Market Analysis

- **Current Trends:** The real estate market is showing signs of inventory growth, indicating a potential increase in opportunities for buyers. However, it is essential to note that demand dynamics will play a crucial role in price stability and potential appreciation.
- **Future Predictions:** The housing market is expected to see a rise in demand due to pent-up consumer interest, especially as borrowing rates stabilize. This trend may benefit land investment if the demand in Moyale aligns with national trends.

2. Economic Factors

- **Development Initiatives:** There are ongoing projects aimed at enhancing economic growth in Moyale, particularly around border trade facilitation. Such initiatives can increase land value over time as infrastructure improves and economic activities expand.
- **Socioeconomic Baselines:** The town's development has been somewhat unregulated, which could pose risks but also offers opportunities for growth as planning and development initiatives are implemented.

3. Investment Potential

- **Demand for Land:** While specific data for Moyale was sparse, there is a general trend that agricultural land is gaining interest due to rising demand for local produce. This could indicate a growing market for land in Moyale, especially if agricultural activity increases.
- **Comparative Advantage:** Other regions in Kenya, such as Matuu and Konza, are highlighted as promising investment areas. This suggests that Moyale may need to position itself competitively to attract investors.

4. Risks and Challenges

- **Regulatory Risks:** When purchasing land, it is crucial to conduct thorough research regarding zoning laws and potential communal claims. Incomplete land regularization could pose significant risks to investors.
- **Market Volatility:** The real estate market can be unpredictable, and while there may be potential for quick returns, it also comes with the risk of price fluctuations that could lead to losses.

Final Verdict

Buying land in Moyale for quick returns might be a viable investment under certain conditions. The ongoing economic development initiatives and increasing demand for agricultural land are positive indicators. However, potential investors should be cautious of regulatory risks and market volatility. Conducting thorough research and possibly consulting with local experts or authorities will be essential to making an informed decision. If the infrastructure projects and economic growth continue as planned, there could be significant opportunities for appreciation in land value over time.

Is it Advisable to to buy land in Moyale ?

User Query:

whats the best place to build a night club in Nairobi?

Agent Response:

To determine the best place to build a nightclub in Nairobi, I created a plan to gather and analyze data on popular nightlife areas, current trends, and local preferences. Here's the plan and the findings based on the research conducted:

Research Plan

1. **Identify Popular Nightlife Districts:** Look for areas in Nairobi known for their vibrant nightlife.
2. **Examine Current Venues:** Analyze existing successful nightclubs and their locations.
3. **Consider Accessibility and Safety:** Evaluate the accessibility of these areas and safety for patrons.
4. **Check for Future Trends:** Investigate any upcoming areas or developments that could influence nightlife.

Findings

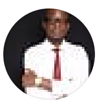
1. **Popular Nightlife Areas:**
   - **Westlands:** This area is highlighted as a vibrant nightlife hub with a wide range of entertainment options. Clubs like K1 Klub House and various bars are well-established here.
   - **Kilimani:** Known for its trendy bars and a youthful crowd, this neighborhood is rapidly gaining popularity for nightlife.
   - **CBD (Central Business District):** Although traditionally quieter at night, it has been seeing an increase in bars and clubs catering to a younger demographic.
2. **Successful Existing Clubs:**
   - **Mulberry Project at The Alchemist:** A popular venue that combines a bar and a lively atmosphere.
   - **K1 Klub House:** Known for its dance clubs and diverse entertainment options, attracting both locals and tourists.
   - **The Tunnel on Mombasa Road:** Another popular spot with a reputation for nightlife.
3. **Accessibility and Safety:**
   - **Westlands** and **Kilimani** both have good transport connectivity, making them accessible for patrons. They are also generally considered safe, especially with the presence of security personnel in entertainment venues.
4. **Future Trends:**
   - Areas like **Kilimani** and parts of **Lang'ata** are being developed, which may become hotspots for nightlife as urbanization continues.

Final Verdict

Based on the research, **Westlands** emerges as the best location to build a nightclub in Nairobi due to its established nightlife scene, accessibility, and safety. Additionally, **Kilimani** is also a promising area due to its growing popularity among younger crowds and increasing number of entertainment venues. Investing in either of these neighborhoods would likely yield favorable results for a new nightclub venture.

whats the best place to build a night club

AI | Agentic Ai | Agentic Rag | Llm | Llm Applications



Written by John Wafula

88 Followers · 9 Following

Follow

Data Scientist | Adanian Labs

No responses yet

What are your thoughts?

Respond

More from John Wafula

John Wafula

Customer Reviews Analysis on AWS EMR with Apache Spark and...

Introduction

Feb 7 5

John Wafula

How i Built a Personal AI-Powered GPT Investment Banker with Equi...

In the dynamic world of finance, having timely and accurate financial analysis is...

Aug 6, 2023 28 3

John Wafula

John Wafula

The Evolutionary Odyssey of Artificial Intelligence: A Journey…

Artificial intelligence could advance in ways that surpass our wildest imaginations. And it…

Mar 25, 2024

The Singularity and the Future of Work: AI Will Not Replace the…

The Singularity is a term used to describe the hypothetical future point at which artificial…

Feb 12, 2023 58

See all from John Wafula

Recommended from Medium

Vipra Singh

AI Agents: Frameworks (Part-3)

Discover AI agents, their design, and real-world applications.

6d ago 167 2

In Level Up Coding by Fareed Khan

Creating an AI Agent That Uses a Computer Like People Do

Sees Your Desktop, Performs Tasks

3d ago 416 9

Lists

Generative AI Recommended Reading

52 stories · 1668 saves

What is ChatGPT?

9 stories · 515 saves

The New Chatbots: ChatGPT, Bard, and Beyond
12 stories · 560 saves

Natural Language Processing
1958 stories · 1603 saves

Julio Pessan

The Future with AI Agents and Vertical AI

Imagine a world where traditional applications become obsolete. Instead of accessing...

5d ago 190 14

Martin Jurado Pedroza

Building My First ChatBot with AI: Python, Ollama, and DeepSeek

Accessing cutting-edge technology without going over budget is a problem for...

Feb 5 10 1

In Data Science Collective by Ari Joury, PhD

Stop Copy-Pasting. Turn PDFs into Data in Seconds

Automate PDF extraction and get structured data instantly with Python's best tools

6d ago 1.1K 22

In Data And Beyond by TONI RAMCHANDANI

Building a Multi-Agent RAG Pipeline with Crew AI

In today's era of intelligent systems, the ability to combine diverse retrieval tools with robus...

Feb 14 71

See more recommendations