Startup Idea Analyzer with CrewAI and Agents

Introduction

The fast-paced startup environment and market makes it essential to correctly and efficiently validate a business idea quickly in order to attract users and investors. Fetch.aiAgents technology makes it possible to develop an application being able to evaluate startup ideas autonomously. By usingCrewAI andAgents it is possible to design aStartup Idea Analyzer offering a structured way to assess the feasibility and validity of any business idea you may have. By leveraging agents, which are programmable micro-services designed to communicate with one another, this system can independently evaluate key aspects of your business including market demand, technological needs, and business strategies with the aim of providing a comprehensive evaluation of your startup idea.

This guide aims at walking you through the setting up and running of a simple evaluation system, allowing you to not only assess your ideas viability but also to integrate them with other systems, users and agents, thus unlocking new potential for economic collaboration and growth.

Let's get started!

Supporting documentation

- Creating an agent
- Communicating with other agents
- Register in Almanac
- Almanac Contract
- Utilising the Agentverse Mailroom service
- Protocols
- Agentverse Functions
- Register an Agent Function on the Agentverse

Pre-requisites

- Python
- : Download and install fromPython official website(opens in a new tab)
- .
- Poetry
- : Install by following the instructions onPoetry's official website(opens in a new tab)
- .
- Gemini API key
- : To get the Gemini API key, sign up atGoogle AI Studio(opens in a new tab)
- and generate the API key.
- Mailbox API key
- : You can get the Mailbox API key by following theMailbox
- guide.

Project Structure

The Agent project presented in this guide has the following structure:

.startup-idea-analyzer ├── agent.py ├── crew_ai.py ├── poetry.lock ├── project.json ├── pyproject.toml └── README.md Each file serves a specific purpose in building and managing theStartup Idea Analyzer system:

- agent.py
- : this defines the core agent setup and protocol communication.
- crew_ai.py
- : it defines the market research, technology assistance and business strategy processes.
- poetry.lock
- : this ensures a consistent dependency management.
- project.json
- : it contains the project configuration.
- pyproject.toml
- : it manages Python package dependencies with Poetry.
- README.md
- : it is the documentation needed to explain how to use or contribute to the project.

Poetry Dependencies

pyproject.toml [tool.poetry.dependencies] python = ">=3.10,<3.12" uagents = "0.14.0" requests = "^2.31.0" uagents-ai-engine = "0.4.0" crewai = "0.36.0" python-dotenv = "1.0.1" langchain-google-genai = "1.0.7"

Startup Idea Analyzer

This guide showcases how to use Agents to analyze a startup idea by performingmarket research ,technological assessments , andbusiness planning . It employs two different systems:uagents (for agent-based communication) andcrew_ai (for task-driven collaboration).

Features

- Market Research
- : The Marketing Agent dives deep into understanding the demand for your product, defining the ideal customer, and crafting strategies to reach a wide audience.
- Technological Analysis
- : The Technology Expert Agent assesses essential technologies needed to create your product efficiently and with top quality.
- Business Strategy Development
- : The Business Consultant Agent synthesizes the information into a comprehensive business plan. This plan includes key strategies, detailed milestones, and a timeline to guide you toward profitability and sustainability.
- Provides over 10 crucial insights
- .
- Identifies 5 clear business goals
- .
- Generates a detailed timeline charting your path to success
- .

How It Works

Within the following system, the aim is not to only aid in market analysis and technology assessment but also to provide tailor-made business blueprint for the success of your business idea.

Here below we depict how the system works:

1. Market Research
2. : TheMarketing Agent
3. conducts thorough research to understand market demand, customer demographics, and effective marketing strategies.
4. Technological Analysis
5. : TheTechnology Expert Agent
6. evaluates the necessary technologies, offering insights on the best tools and methods for your product development.
7. Business Strategy Development
8. : TheBusiness Consultant Agent
9. consolidates all gathered information to create a comprehensive business plan, including strategies, milestones, and a timeline.

Agent System

This system defines an Agent calledstartup idea analyzer that interacts with other Agents using aprotocol . The Agent listens for messages containing a startup ideadescription and forwards this input to a market research process, which returns a response based on the input provided.

Self hosted agent.py import os

from ai_engine import UAgentResponse , UAgentResponseType from uagents import Agent , Context , Field , Model , Protocol from uagents . setup import fund_agent_if_low

from crew_ai import MarketResearchProcess

AGENT_MAILBOX_KEY

os . environ . get ("AGENT_MAILBOX_KEY") SEED_PHRASE =

"YOUR_SEED_PHRASE"

agent

Agent (name = "startup idea analyzer" , seed = SEED_PHRASE, mailbox = f " { AGENT_MAILBOX_KEY } @https://agentverse.ai" ,)

protocol

Protocol ("Startup idea Analyzer version" , version = "0.1.2") researcher =

MarketResearchProcess ()

print (agent.address, "agent_address")

fund_agent_if_low (agent.wallet. address ())

class

StartUpIdeaAnalyzer (Model): description :

str

=

Field (description = "describes the field which will be the description of the startup idea and it is provided by the user in context")

@protocol . on_message (model = StartUpIdeaAnalyzer, replies = {UAgentResponse}) async

def

on_message (ctx : Context ,

sender :

str ,

msg : StartUpIdeaAnalyzer): ctx . logger . info (f "Received message from { sender } , message { msg.description } ") result = researcher . run_process (msg.description) await ctx . send (sender, UAgentResponse (message = result, type = UAgentResponseType.FINAL))

agent . include (protocol, publish_manifest = True)

agent . run () TheAgent is defined using a seed phrase for wallet generation and a mailbox for message exchanges. Then, a protocol namedStartup idea Analyzer version (version 0.1.2) defines how the agent processes incoming messages. The agent uses this protocol to respond to requests containing startup idea descriptions. The agent relies on a component calledMarketResearchProcess from an external modulecrew_ai to conduct market research on the provided startup idea. This research process is triggered upon receiving a startup idea description. TheStartUpIdeaAnalyzer class defines a data model for startup idea descriptions; this model expects a fielddescription that holds the string of text describing the startup idea provided by the user.

Whenever the agent receives a message matching theStartUpIdeaAnalyzer model, it logs the incoming message, sends the startup description to theMarketResearchProcess , and then responds to the sender with the result. The agent is set to include thisprotocol , publish a manifest of its functionalities, and run continuously to process incoming requests.

CrewAI System

The following script defines aMarketResearchProcess class that sets up a market research workflow using three specialized agents: aMarket Research Analyst , aTechnology Expert , and aBusiness Development Consultant . The script creates and assigns tasks to these agents, such as analyzing market demand, assessing technological needs, and developing a business plan. It then uses theCrew class to execute these tasks sequentially, passing the results from one agent to the next one, to then return the final outcome.

Self hosted crew_ai.py import os

from crewai import Agent , Crew , Process , Task from dotenv import load_dotenv from langchain_google_genai import ChatGoogleGenerativeAI

load_dotenv ()

class

MarketResearchProcess : def

init (self): api_gemini = os . environ . get ("GEMINI_API_KEY") self . llm =

ChatGoogleGenerativeAI (model = "gemini-pro" , verbose = True , temperature = 0.1 , google_api_key = api_gemini)

self . marketer =

Agent (role = "Market Research Analyst" , goal = "Find out how big is the demand for my products and suggest how to reach the widest possible customer base" , backstory = """You are an expert at understanding the market demand, target audience, and competition. This is crucial for validating whether an idea fulfills a market need and has the potential to attract a wide audience. You are good at coming up with ideas on how to appeal to widest possible audience. """ , verbose = True ,

enable more detailed or extensive output

allow_delegation

True ,

enable collaboration between agent

llm

self.llm,

to load gemini

)

self . technologist =

Agent (role = "Technology Expert" , goal = "Make assessment on how technologically feasible the company is and what type of technologies the company needs to adopt in order to succeed" , backstory = """You are a visionary in the realm of technology, with a deep understanding of both current and emerging technological trends. Your expertise lies not just in knowing the technology but in foreseeing how it can be leveraged to solve real-world problems and drive business innovation. You have a knack for identifying which technological solutions best fit different business models and needs, ensuring that companies stay ahead of the curve. Your insights are crucial in aligning technology with business strategies, ensuring that the technological adoption not only enhances operational efficiency but also provides a competitive edge in the market.""" , verbose = True ,

enable more detailed or extensive output

allow_delegation

True ,

enable collaboration between agent

llm

self.llm,

to load gemini

)

self . business_consultant =

Agent (role = "Business Development Consultant" , goal = "Evaluate and advise on the business model, scalability, and potential revenue streams to ensure long-term sustainability and profitability" , backstory = """You are a seasoned professional with expertise in shaping business strategies. Your insight is essential for turning innovative ideas into viable business models. You have a keen understanding of various industries and are adept at identifying and developing potential revenue streams. Your experience in scalability ensures that a business can grow without compromising its values or operational efficiency. Your advice is not just about immediate gains but about building a resilient and adaptable business that can thrive in a changing market.""" , verbose = True ,

enable more detailed or extensive output

allow_delegation

True ,

enable collaboration between agent

llm

self.llm,

to load gemini

)

def

create_tasks (self ,

input_data): self . task1 =

Task (description = f """Analyze what the market demand for { input_data } . Write a detailed report with description of what the ideal customer might look like, and how to reach the widest possible audience. The report has to be concise with at least 10 bullet points and it has to address the most important areas when it comes to marketing this type of business. """ , agent = self.marketer, expected_output = "A detailed market research report with at least 10 bullet points on the ideal customer and marketing strategy." ,)

self . task2 =

Task (description = f """Analyze { input_data } . Write a detailed report with description of which technologies the business needs to use in order to make High Quality T shirts. The report has to be concise with at least 10 bullet points and it has to address the most important areas when it comes to manufacturing this type of business. """ , agent = self.technologist, expected_output = "A detailed technological report with at least 10 bullet points on the necessary technologies for manufacturing." ,)

self . task3 =

Task (description = f """Analyze and summarize marketing and technological report and write a detailed business plan with description of { input_data } . The business plan has to be concise with at least 10 bullet points, 5 goals and it has to contain a time schedule for which goal should be achieved and when. """ , agent = self.business_consultant, expected_output = "A detailed business plan with at least 10 bullet points, 5 goals, and a time schedule." ,)

def

run_process (self ,

input_data): self . create_tasks (input_data)

crew

Crew (agents = [self.marketer, self.technologist, self.business_consultant], tasks = [self.task1, self.task2, self.task3], verbose = 2 , process = Process.sequential,

Sequential process will have tasks executed one after the

other and the outcome of the previous one is passed as extra content into this next.

)

result

crew . kickoff (inputs = { "input" : input_data}) return result The script initializes by loading a Large Language Model (LLM) calledChatGoogleGenerativeAI , which uses the Gemini API for generating detailed text-based outputs. This LLM is shared among the agents to enhance their ability to generate accurate and detailed reports.

Each agent has a unique role and goal : theMarket Research Analyst focuses on understanding market demand, target audience, and marketing strategies. TheTechnology Expert assesses the technological feasibility of the company and suggests necessary technologies for success. TheBusiness Development Consultant evaluates the business model, scalability, and potential revenue streams to ensure long-term sustainability.

Thecreate_tasks method generates specific tasks for each agent. These tasks involve analyzing input data (e.g., a business idea), producing reports in their areas of expertise, and then passing these reports from one agent to the next one. For instance, the Market Research Analyst creates a marketing strategy, which the Technology Expert uses to assess technological needs, and finally, the Business Development Consultant uses both reports to develop a business plan.

Therun_process method coordinates the execution of these tasks in a sequential manner. ACrew is formed, comprising the three agents and their tasks. The process is set to execute each task one after the other, passing the output of one task as input for the next. The final result is a comprehensive report that includes market research, technological assessments, and a business plan.

How to Run This Example

Update the required environment variables

You need to provide the following API Keys to correctly run the example provided within this guide:

.env.example AGENT_MAILBOX_KEY = "YOUR_MAILBOX_KEY" GEMINI_API_KEY = "YOUR_GEMINI_API_KEY"

Run the example

- Navigate to the root folder of the example.
- Update the.env
- file.
- Install dependencies by running this command:poetry install
- .
- Execute the script by running the following command:python agent.py
- .

Expected Output

To run the example, make sure your scripts are running on your end and then head over toDeltaV and provide a description for your startup idea in the dedicated bar. Choose theAI Engine personality and click onStart button. You should get something similar to the following:

Last updated on October 17, 2024

Was this page helpful?

You can also leave detailed feedbackon Github

AI Engine Javascript SDK Creating an Agent with Crewai

On This Page