

ABank Marketing Crew

Quick Start Guide: Autonomous AI Agents for Marketing

Acalytica

Powered by CrewAI

November 17, 2025

Agenda

- 1 Introduction
- 2 The Use Case
- 3 The AI Agents
- 4 The Workflow
- 5 Running the System
- 6 Expected Output
- 7 Key Concepts
- 8 Customization
- 9 Best Practices
- 10 Next Steps
- 11 Conclusion

What is ABank Marketing Crew?

Definition

An **autonomous multi-agent system** that uses AI agents to execute complete marketing workflows—from market research to content creation—without human intervention.

Traditional Approach:

- Manual market research
- Individual task execution
- Sequential handoffs
- Days or weeks to complete

CrewAI Approach:

- Automated intelligence gathering
- Parallel agent collaboration
- Autonomous workflow
- **Minutes to complete**

Why Autonomous Agents?

Key Benefits

Speed

10-100x faster

Consistency

No human error

Scale

Unlimited parallel

Cost

90% reduction

Key Insight

Agents work **together** autonomously—like a human team, but faster and more consistent.

Example Campaign: Youth Digital Savings Account

Product Brief

Product: Youth Digital Savings Account

Target: 18-25 year olds in South Africa

Budget: R500,000

Timeline: Immediate launch

Required Deliverables:

- 1 Market analysis report
- 2 Competitor research
- 3 Product feature recommendations
- 4 Pricing strategy
- 5 Content marketing plan
- 6 Social media campaign ideas

Traditional: 2-3 weeks

Meet the Marketing Team

Agent 1: Market Intelligence Analyst

- 15 years experience in SA banking
- Analyzes markets, competitors, opportunities
- Provides target audience insights
- Recommends features and pricing

Agent 2: Content Marketing Strategist

- Award-winning campaign creator
- Deep understanding of SA culture
- Develops messaging and channels
- Creates content ideas

Agent 1: Code Structure

```
market_analyst = Agent(  
    role="Market Intelligence Analyst",  
    goal="Analyze SA banking market",  
    backstory="""Expert analyst with 15 years  
experience in South African financial  
services.""",  
    verbose=True  
)
```

Responsibilities

Target audience • Competitive landscape • Features • Pricing

Agent 2: Code Structure

```
content_strategist = Agent(  
    role="Content Marketing Strategist",  
    goal="Develop compelling campaigns",  
    backstory="""Award-winning strategist  
with deep understanding of South African  
culture.""" ,  
    verbose=True  
)
```

Responsibilities

Messaging • Channels • Content creation • Campaign ideas

Sequential Workflow Process

Market Intelligence Analyst

Task 1: Market Analysis

Content Strategist

Task 2: Content Strategy

Output: Complete Campaign

Key Feature: Context Passing

Task 2 receives Task 1's output as context, agents build on each other's work.

Task 1: Market Analysis

```
market_analysis_task = Task(  
    description="""Analyze market opportunity  
    for Youth Digital Savings Account.  
    Consider target audience, competitors,  
    features, and pricing."""  
  
    expected_output="""Market analysis with:  
    - Target audience profile  
    - 3-5 competitor products  
    - Recommended features  
    - Pricing recommendation""",  
  
    agent=market_analyst  
)
```

Task 2: Content Strategy

```
content_strategy_task = Task(
    description="""Develop content strategy
    for launching the product. Include
    messaging, channels, and content ideas.""",

    expected_output="""Content strategy with:
    - 3 key messaging pillars
    - Recommended channels
    - 5 social media post ideas""",

    agent=content_strategist,
    context=[market_analysis_task]    # KEY!
)
```

Context passing ensures coherent results

Creating and Running the Crew

```
# Assemble the crew
crew = Crew(
    agents=[market_analyst, content_strategist],
    tasks=[market_analysis_task,
           content_strategy_task],
    process=Process.sequential,
    verbose=True
)

# Execute
result = crew.kickoff()

# Save result
with open("result.txt", 'w') as f:
    f.write(str(result))
```

That's it!

Prerequisites

Required Setup

- 1 Python 3.10+
- 2 API Key (OpenAI or Anthropic)
- 3 Environment file (.env)
- 4 Dependencies (crewai, python-dotenv)

Installation:

- `pip install crewai python-dotenv`
- Create .env file with API key
- Run: `python quick_start_simple.py`

Running the Quick Start

Four Simple Steps:

- 1 Configure API key in .env file
- 2 Run: `python quick_start_simple.py`
- 3 Watch agents work autonomously
- 4 Review saved results file

Execution Time

Runtime: 5-10 minutes

Output: `test_result_TIMESTAMP.txt`

Sample Market Analysis Output

Target Audience

Age 18-25, digital natives, mobile-first, low initial savings

Competitors

Capitec Live Better • TymeBank GoalSave • Discovery Vitality Money • FNB eWallet

Recommended Features

- Zero monthly fees
- Savings goals with progress tracking
- Round-up features
- Instant transfers

Sample Content Strategy Output

Key Messaging Pillars

- 1 "Your Money, Your Rules" — Empowerment
- 2 "Save While You Spend" — Effortless saving
- 3 "Future-Ready Banking" — Digital innovation

Recommended Channels

TikTok (primary) • Instagram (visual) • Twitter (service) • Campus activations

Sample Posts

TikTok challenge, Instagram stories, Twitter polls, Reels, Duets with influencers

Understanding CrewAI Components

AGENT	TASK	CREW
Who does work	What needs doing	How work flows
Role	Description	Agents list
Goal	Expected output	Tasks list
Backstory	Context	Process type

The CrewAI Formula

Agents + Tasks + Crew = Automation

Process Types

Sequential

- Tasks run in order
- Output passes to next
- Simple and predictable

Use when:

- Tasks depend on each other
- Linear workflow

Hierarchical

- Manager coordinates
- Parallel execution
- Dynamic allocation

Use when:

- Independent tasks
- Need optimization

This example uses Sequential for simplicity

Easy Modifications:

① Change the product

- ▶ Edit task descriptions
- ▶ Update target audience

② Add more agents

- ▶ Compliance officer
- ▶ Financial analyst

③ Add more tasks

- ▶ Risk assessment
- ▶ Budget allocation

Scaling the System

Level 1	Level 2	Level 3
2 agents	5 agents	7+ agents
Basic workflow	Full campaign	Enterprise system
Quick start	With validation	With tools

This example is Level 1—perfect for learning!

Best Practices

Agent Design

- Clear, specific roles
- Detailed backstories
- Measurable goals

Task Design

- Detailed descriptions
- Clear expected outputs
- Proper context linking

Avoid

Vague instructions • Missing context • Overlapping roles • Unrealistic expectations

- ① **Start:** Run `quick_start_simple.py`
 - ▶ Understand basic structure
- ② **Experiment:** Modify the example
 - ▶ Change product/audience
 - ▶ Add a third agent
- ③ **Expand:** Build full system
 - ▶ Add custom tools
 - ▶ Implement hierarchical process
- ④ **Deploy:** Production use
 - ▶ Schedule campaigns
 - ▶ Integrate with marketing stack

Documentation

- CrewAI Docs: docs.crewai.com
- GitHub: github.com/joaomdmoura/crewAI
- Examples: github.com/crewAIInc/crewAI-examples

Support

- Discord: CrewAI Community Server
- Stack Overflow: Tag [crewai](#)

Key Takeaways

- ➊ Autonomous AI agents execute complete workflows
- ➋ CrewAI makes it simple: Agents + Tasks + Crew
- ➌ Sequential process ensures coherent results
- ➍ Context passing enables collaboration
- ➎ Start simple, scale gradually

From weeks to minutes

This is the future of marketing operations

Thank You

Questions?

Get Started Today

`python quick_start_simple.py`

Acalytica

Powered by CrewAI