MLM-Focused Workflow Map & n8n Build Guide

Step-by-Step Implementation Guide

Built by TheAIDepot – Your 24/7 Recruiting Assistant

Table of Contents

---

1. Introduction

"At {{companyName}}, we don't chase — we attract. This system qualifies leads, explains your offer, handles follow-ups, and even books calls or signs them up — all without lifting a finger."

This comprehensive guide will walk you through implementing a fully automated MLM workflow system that captures leads, qualifies them through AI, segments them based on interest, and follows up automatically - all while being completely white-label ready for your business.

What You'll Build

A complete end-to-end system that: - Captures leads from multiple sources (Google Form, Telegram, WhatsApp) - Qualifies leads through an interactive quiz - Uses AI to personalize responses and detect interest - Segments leads into appropriate tracks (customer vs. recruit) - Delivers automated follow-up via email and SMS - Handles common objections intelligently - Converts leads through Stripe payments or Calendly bookings - Onboards new customers and recruits with tailored content

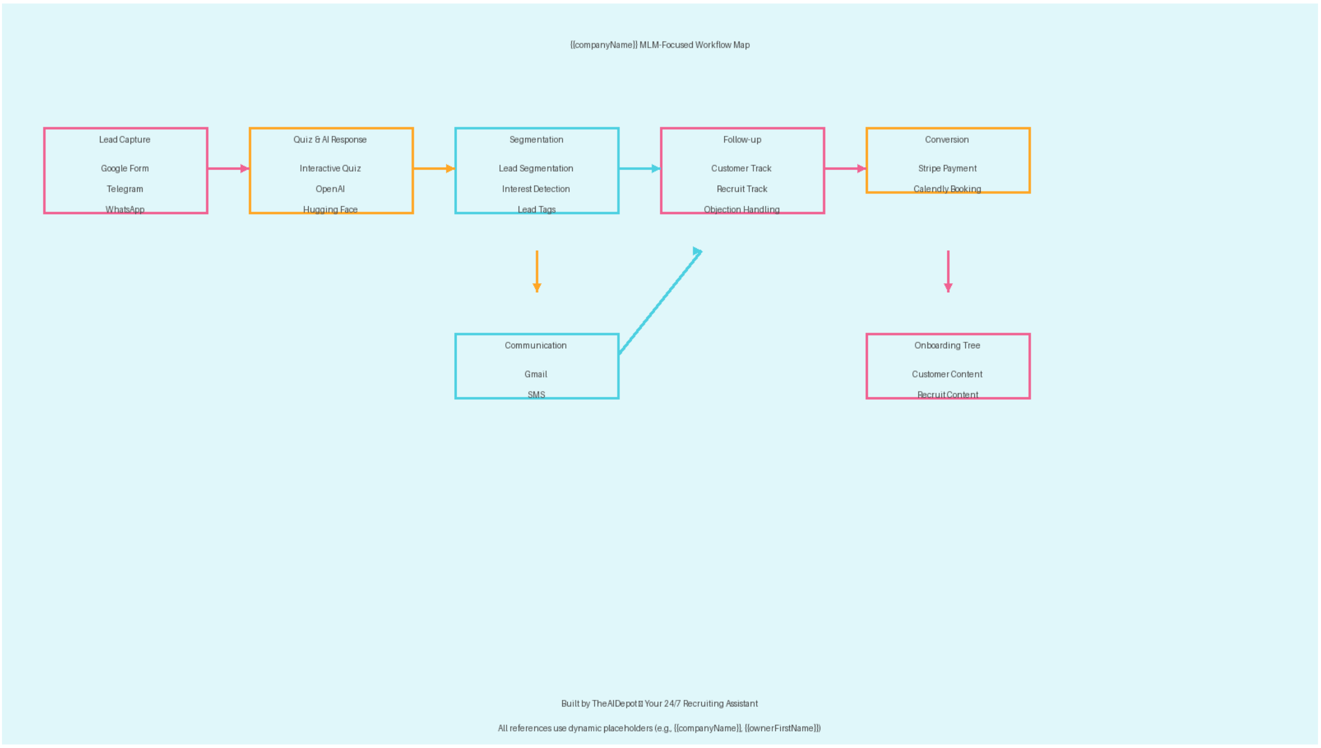
Prerequisites

Before starting, ensure you have: - An n8n account (self-hosted or cloud) - API keys for required services - Basic understanding of workflow automation - Your branding assets and messaging guidelines

Let's begin building your 24/7 recruiting assistant!

---

2. System Overview



The workflow consists of seven interconnected components:

1. **Lead Capture**: Multiple entry points to capture leads from various channels
2. **Quiz & AI Response**: Interactive qualification and personalized AI responses

3. **Segmentation**: Interest detection and lead tagging for proper routing
4. **Follow-up**: Separate tracks for customers and recruits with objection handling
5. **Conversion**: Payment processing and appointment booking
6. **Communication**: Automated email and SMS follow-up sequences
7. **Onboarding**: Tailored content delivery based on lead type

Each component is designed to be customizable while maintaining the core workflow logic that drives conversions.

---

3. Setting Up Your Workflow

Trigger Configuration

Google Form Integration

1. Create a new workflow in n8n
2. Add a "Google Forms Trigger" node
3. Authentication: Connect your Google account
4. Form: Select your lead capture form

5. Operation: "Watch for new responses"

6. Create a lead capture form with these essential fields:

7. Name
8. Email
9. Phone
10. Interest Level (1-10)

11. Primary Interest (Product, Business Opportunity, Both)

12. Add a "Set" node to format the incoming data: `javascript { "leadSource": "Google Form", "leadData": { "name": "{{$node["Google Forms Trigger"].json.name}}", "email": "{{$node["Google Forms Trigger"].json.email}}", "phone": "{{$node["Google Forms Trigger"].json.phone}}", "interestLevel": "{{$node["Google Forms Trigger"].json.interestLevel}}", "primaryInterest": "{{$node["Google Forms Trigger"].json.primaryInterest}}" }, "timestamp": "{{$now}}" }`

Telegram Bot Integration

1. Create a Telegram bot using BotFather
2. Add a "Telegram Trigger" node
3. Authentication: Enter your Bot Token
4. Updates: "message"
5. Message Types: "text"
6. Add a "Telegram" node after the trigger
7. Operation: "Send Message"
8. Chat ID: `{{$node["Telegram Trigger"].json.message.chat.id}}`
9. Text: "Hi! Thanks for reaching out to {{companyName}}. I'd like to learn more about what you're looking for. Could you answer a few quick questions?"
10. Add a "Set" node to initialize the conversation: `javascript { "leadSource": "Telegram", "leadData": { "name": "{{$node["Telegram Trigger"].json.message.from.first_name}}", "chatId": "{{$node["Telegram Trigger"].json.message.chat.id}}", "username": "{{$node["Telegram Trigger"].json.message.from.username}}" }, "quizStep": 1, "timestamp": "{{$now}}" }`

WhatsApp Integration

1. Set up WhatsApp Business API access
2. Add a "WhatsApp Business" node
3. Authentication: Connect your WhatsApp Business account
4. Operation: "Send Template Message"
5. To: Use the incoming phone number
6. Template: Create a welcome template with your introduction
7. Add a "Set" node to initialize the conversation: `javascript { "leadSource": "WhatsApp", "leadData": { "phone": "{{$node["WhatsApp Trigger"].json.from}}", }, "quizStep": 1, "timestamp": "{{$now}}" }`

Quiz Flow Setup

1. Create a "Switch" node to route based on quiz step

2. Add cases for each step (1-5)

3. For each step, create:

4. A message node to ask the question
5. A webhook to receive the response

6. A "Set" node to store the response

7. Example quiz questions:

8. Question 1: "What's your biggest challenge right now with your current business or job?"
9. Question 2: "On a scale of 1-10, how interested are you in additional income opportunities?"
10. Question 3: "How much time could you dedicate weekly to a new opportunity?"
11. Question 4: "What would you do with an extra $1,000 per month?"

12. Question 5: "Are you more interested in using our products or building a business?"

13. Final "Set" node to compile all responses: `javascript { "leadData": { "name": "{{$node["Set1"].json.leadData.name}}", "email": "{{$node["Set1"].json.leadData.email}}", "phone": "{{$node["Set1"].json.leadData.phone}}", "question1": "{{$node["Set2"].json.response}}", "question2": "{{$node["Set3"].json.response}}", "question3": "{{$node["Set4"].json.response}}", "question4": "{{$node["Set5"].json.response}}", "question5": "{{$node["Set6"].json.response}}" }, "quizComplete": true, "leadSource": "{{$node["Set1"].json.leadSource}}", "timestamp": "{{$now}}" }`

AI Integration

OpenAI Integration

1. Add an "HTTP Request" node
2. Method: POST
3. URL: https://api.openai.com/v1/chat/completions
4. Authentication: "Header Auth"
5. Header Parameters:
    - Name: Authorization
    - Value: Bearer {{$node["Credentials"].json.openAIKey}}

6. Request Body: `json { "model": "gpt-4", "messages": [{ "role": "system", "content": "You are an assistant for {{companyName}}, an MLM business. Your job is to analyze lead responses and provide personalized feedback." }, { "role": "user", "content": "Analyze this lead data and provide a personalized response: {{$node['Quiz Complete'].json.leadData}}" }], "temperature": 0.7 }`

7. Add a "Set" node to store the AI response

8. Values to Set:
    - aiResponse: {{$node["HTTP Request"].json.choices[0].message.content}}

Hugging Face Integration

1. Add an "HTTP Request" node
2. Method: POST
3. URL: https://api-inference.huggingface.co/models/lixiaoxi45/WebThinker-R1-32B
4. Authentication: "Header Auth"
5. Header Parameters:
    - Name: Authorization
    - Value: Bearer {{$node["Credentials"].json.huggingFaceKey}}

6. Request Body: `json { "inputs": "Analyze this lead data from {{companyName}} and determine qualification level and next steps: {{$node['Quiz Complete'].json.leadData}}" }`

7. Add a "Set" node to store the Hugging Face response

8. Values to Set:

    - hfResponse: {{$node["HTTP Request"].json[0].generated_text}}

9. Add a "Switch" node to select which AI to use based on configuration

10. Case 1: OpenAI (default)
11. Case 2: Hugging Face

Segmentation Logic

1. Add a "Code" node for interest detection
2. Language: JavaScript

3. Code: ```javascript // Input data from quiz and AI analysis const leadData = $input.item.json.leadData; const aiResponse = $input.item.json.aiResponse;

   // Interest detection logic let interestLevel = 0; let interestAreas = []; let leadType = "unknown";

   // Parse interest level from quiz data if (leadData.question2) { interestLevel = parseInt(leadData.question2); }

   // Detect interest areas from responses if (leadData.question5.toLowerCase().includes("product")) { interestAreas.push("product"); leadType = "customer"; }

   if (leadData.question5.toLowerCase().includes("business")) { interestAreas.push("business"); leadType = "recruit"; }

   // Use AI to enhance detection if (aiResponse.toLowerCase().includes("business opportunity")) { if (!interestAreas.includes("business")) { interestAreas.push("business"); } if (leadType === "unknown") { leadType = "recruit"; } }

   if (aiResponse.toLowerCase().includes("product")) { if (!interestAreas.includes("product")) { interestAreas.push("product"); } if (leadType === "unknown") { leadType = "customer"; } }

   // Determine lead quality let leadQuality = "cold"; if (interestLevel >= 7) { leadQuality = "hot"; } else if (interestLevel >= 4) { leadQuality = "warm"; }

   // Return segmented data return { json: { leadType, interestLevel, interestAreas, leadQuality, originalData: leadData, aiInsights: aiResponse } }; ```

4. Add a "Switch" node for routing based on segmentation

5. Add cases for:

   - Hot Recruit
   - Warm Recruit
   - Cold Recruit
   - Hot Customer
   - Warm Customer
   - Cold Customer

6. Create a MongoDB node to store lead data with tags

7. Operation: "Insert"

8. Collection: "leads"
9. Fields to Send: All segmentation data

Follow-up Automation

Gmail Integration

1. Add a "Gmail" node
2. Authentication: Connect your Gmail account
3. Operation: "Send Email"
4. To Email: {{$node["Segmentation"].json.originalData.email}}
5. Subject: Customize based on segment

6. Body: Use a template with personalization

7. Create email templates for each segment:

Hot Recruit Template: ``` Subject: {{lead.firstName}}, Your {{companyName}} Business Opportunity Details

Hi {{lead.firstName}},

Thanks for your interest in the {{companyName}} business opportunity! Based on your responses, I think you'd be an excellent fit for our team.

{{aiResponse}}

I'd love to schedule a call to discuss how you can get started. Use this link to book a time: {{calendlyLink}}

Looking forward to speaking with you!

{{ownerFirstName}} {{companyName}} ```

Hot Customer Template: ``` Subject: {{lead.firstName}}, Your {{companyName}} Product Recommendations

Hi {{lead.firstName}},

Thank you for your interest in {{companyName}} products! Based on what you shared, I think these would be perfect for you:

{{aiResponse}}

Ready to try them? Use this link to place your order: {{stripePaymentLink}}

If you have any questions, just reply to this email.

{{ownerFirstName}} {{companyName}} ```

SMS Integration

1. Add a "Twilio" node
2. Authentication: Connect your Twilio account
3. Operation: "Send SMS"
4. From: Your Twilio number
5. To: {{$node["Segmentation"].json.originalData.phone}}

6. Message: Customize based on segment

7. Create SMS templates for each segment:

Hot Recruit SMS:
`Hi {{lead.firstName}}! Thanks for your interest in the {{companyName}} business opportunity. I've emailed you details. Book a call here: {{shortCalendlyLink}} - {{ownerFirstName}}`

Hot Customer SMS: `Hi {{lead.firstName}}! Your {{companyName}} product recommendations are in your email. Order here: {{shortPaymentLink}} - {{ownerFirstName}}`

Conversion Pathways

Stripe Payment Integration

1. Add a "Stripe" node
2. Authentication: Connect your Stripe account
3. Operation: "Create Payment Link"
4. Amount: (Set based on product package)
5. Currency: USD (or your local currency)

6. Description: "{{companyName}} Product Package"

7. Add a "Set" node to store the payment link

8. Values to Set:

    - stripePaymentLink: {{$node["Stripe"].json.url}}
    - shortPaymentLink: (Use a URL shortener service)

9. Add a webhook to receive payment confirmation

10. Method: POST

11. Path: /stripe-webhook

12. Authentication: None (use Stripe signing secret for validation)

13. Add a "Code" node to validate the webhook

14. Language: JavaScript
15. Code: (Stripe webhook validation code)

Calendly Booking Integration

1. Add an "HTTP Request" node
2. Method: POST
3. URL: https://api.calendly.com/scheduling_links
4. Authentication: "Header Auth"
5. Header Parameters:
    - Name: Authorization
    - Value: Bearer {{$node["Credentials"].json.calendlyToken}}

6. Request Body: `json { "max_event_count": 1, "owner": "{{calendlyUserURI}}", "event_type": "{{calendlyEventTypeURI}}" }`

7. Add a "Set" node to store the booking link

8. Values to Set:

    - calendlyLink: {{$node["HTTP Request"].json.resource.booking_url}}
    - shortCalendlyLink: (Use a URL shortener service)

9. Add a webhook to receive booking confirmation

10. Method: POST
11. Path: /calendly-webhook
12. Authentication: None (use Calendly signing secret for validation)

Onboarding Tree

1. Create a "Switch" node based on conversion type
2. Case 1: Product Purchase (Customer)

3. Case 2: Business Signup (Recruit)

4. For Customer Onboarding:

5. Add a "Wait" node to delay until after expected delivery

6. Add a "Gmail" node for product usage instructions
7. Add a "Wait" node for 7 days
8. Add a "Gmail" node for follow-up and testimonial request
9. For Recruit Onboarding:
10. Add a "Wait" node to delay until after scheduled call
11. Add a "Gmail" node with getting started materials
12. Add a "Wait" node for 3 days
13. Add a "Gmail" node with training resources
14. Add a "Wait" node for 7 days
15. Add a "Gmail" node with check-in and support offer

---

4. White-Label Configuration

1. Create a "Set" node at the beginning of your workflow to store all customizable variables:

```
 // Company Information
companyName: "{{companyName}}",
companyLogo: "{{companyLogoURL}}",
companyWebsite: "{{companyWebsite}}",
companyColors: {
  primary: "{{companyPrimaryColor}}",
  secondary: "{{companySecondaryColor}}"
},

// Owner Information
ownerFirstName: "{{ownerFirstName}}",
ownerLastName: "{{ownerLastName}}",
ownerEmail: "{{ownerEmail}}",
ownerPhone: "{{ownerPhone}}",

// Product Information
productLine: "{{productLine}}",
productCategories: {{productCategories}},
compensationPlanName: "{{compensationPlanName}}",

// API Keys (store securely)
openAIKey: "{{openAIKey}}",
huggingFaceKey: "{{huggingFaceKey}}",
calendlyToken: "{{calendlyToken}}",
calendlyUserURI: "{{calendlyUserURI}}",
calendlyEventTypeURI: "{{calendlyEventTypeURI}}"
```

1. Reference these variables throughout your workflow using expressions like `{{$node["Credentials"].json.companyName}}`

2. Create a configuration spreadsheet for easy updates:

Variable	Description	Example Value
companyName	Your company name	Vibrant Health Products
companyLogoURL	URL to your logo	https://example.com/logo.png
companyWebsite	Your website URL	https://example.com
companyPrimaryColor	Primary brand color (hex)	#4DD0E1
companySecondaryColor	Secondary brand color (hex)	#F06292
ownerFirstName	Your first name	Jane
ownerLastName	Your last name	Smith
ownerEmail	Your email address	jane@example.com
ownerPhone	Your phone number	+1234567890
productLine	Main product category	Nutritional Supplements
compensationPlanName	Name of your comp plan	Freedom Plan

---

5. Testing and Deployment

1. Test each segment of the workflow individually:
2. Trigger test: Submit test leads through each entry point
3. Quiz flow test: Verify all questions and response storage
4. AI response test: Check personalization and accuracy
5. Segmentation logic test: Verify correct routing
6. Email/SMS delivery test: Check formatting and delivery

7. Payment/Booking link test: Verify functionality

8. Perform end-to-end testing with sample leads:

9. Create test personas for different segments
10. Run complete workflow for each persona
11. Verify all automations trigger correctly

12. Check for any errors or unexpected behavior

13. Deploy your workflow:

14. Set execution to "Production"
15. Configure error handling
16. Set up monitoring alerts

17. Document all webhook URLs and API endpoints

18. Create a monitoring dashboard:

19. Track lead volume by source
20. Monitor conversion rates
21. Track email/SMS delivery rates
22. Monitor AI response quality

---

6. Maintenance and Optimization

1. Regular maintenance tasks:
2. Check API key validity monthly
3. Update email templates seasonally
4. Refresh AI prompts based on performance

5. Review and update segmentation logic quarterly

6. Optimization strategies:

7. A/B test email subject lines and content
8. Experiment with different quiz questions
9. Analyze conversion rates by segment

10. Refine AI prompts based on lead feedback

11. Scaling considerations:

12. Monitor execution credits
13. Consider upgrading to dedicated hosting for high volume

14. Implement batching for large lead volumes
15. Add additional segmentation for more targeted follow-up

---

7. Appendices

Starter AI Prompts

See the complete set of starter AI prompts in the accompanying file: `starter_ai_prompts.md`

These prompts include: - Lead qualification system prompts - Follow-up messaging prompts - Objection handling templates - Common objection responses

Lead Tag System

See the complete lead tagging system in the accompanying file: `lead_tag_spreadsheet.md`

The tagging system includes: - Lead information tags - Interest tags - Funnel stage tags - Communication tags - Objection tags - Conversion tags - Custom tag templates

Troubleshooting Guide

Common issues and solutions:

1. **Trigger not firing**
2. Check webhook URL is correctly configured
3. Verify API credentials are valid

4. Test trigger manually with sample data

5. **AI responses not generating**

6. Verify API key is valid
7. Check request format matches API requirements

8. Test with simplified prompt

9. **Email/SMS not sending**

10. Verify authentication credentials
11. Check for rate limiting

12. Ensure recipient information is correctly formatted

13. **Segmentation errors**

14. Review code logic for edge cases
15. Ensure all required fields are present

16. Add error handling for unexpected inputs

17. **Webhook validation failures**

18. Verify signing secrets match
19. Check webhook URL is accessible
20. Review payload format requirements

Conclusion

This MLM-Focused Workflow Map and n8n Build Guide provides everything you need to create a fully automated lead capture, qualification, and conversion system. By leveraging AI for personalization and segmentation, you'll convert more leads with less manual effort.

Remember that the key to success is continuous refinement. Monitor your conversion rates at each step and optimize your messaging, segmentation logic, and follow-up sequences accordingly.

"At {{companyName}}, we don't chase — we attract. This system qualifies leads, explains your offer, handles follow-ups, and even books calls or signs them up — all without lifting a finger."

Built by TheAIDepot – Your 24/7 Recruiting Assistant