

GlowBot GPT Module Code Archive

Beginner Skincare Kit Generator (generateBeginnerKit.js)

```
// 📄 gpt/generateBeginnerKit.js
module.exports = async function generateBeginnerKit(openai) {
  async function gptPrompt(prompt) {
    const response = await openai.chat.completions.create({
      model: "gpt-4",
      messages: [
        { role: "system", content: "You are a skincare educator introducing beginners to affordable, effective routines." },
        { role: "user", content: prompt }
      ],
      temperature: 0.85
    });
    return response.choices[0].message.content.trim();
  }

  const [intro, content, outro] = await Promise.all([
    gptPrompt(`In 1-2 short sentences, introduce a beginner skincare kit. Keep it under 30 words.`),
    gptPrompt(`List exactly 5 essential skincare products that belong in a beginner's kit. Use this format:\n\n1. Gentle Cleanser - Removes dirt and oil without drying.`),
    gptPrompt(`In 1 sentence, encourage beginners to stick with their routine and stay consistent. Keep it under 20 words.`)
  ]);

  return {
    type: "beginnerKit",
    intro,
    content,
    outro,
    watermark: "generated-by-glowbot-v1"
  };
};
```

Demo Script Generator (generateDemoScript.js)

```
// 📄 gpt/generateDemoScript.js
module.exports = async function generateDemoScript(openai, product) {
  async function gptPrompt(prompt) {
    const response = await openai.chat.completions.create({
      model: "gpt-4",
      messages: [
        { role: "system", content: "You are a creative, engaging skincare content writer focused on UGC scripts." },
        { role: "user", content: prompt }
      ],
    });
  }
};
```

```

    temperature: 0.85
  });
  return response.choices[0].message.content.trim();
}


const [intro, content, outro] = await Promise.all([
  gptPrompt(`In 1-2 short sentences, write a punchy intro for a demo video featuring "${product}". Keep it under 30 words.`),
  gptPrompt(`Write a short 100 word demo script for "${product}". It should walk the viewer through how to use it and what it feels like, in a confident and casual tone.`),
  gptPrompt(`In 1 sentence, write a confident outro that encourages viewers to try "${product}". Keep it under 20 words.`)
]);

return {
  type: "demoScript",
  intro,
  content,
  outro,
  watermark: "generated-by-glowbot-v1"
};
};

```

Drugstore Dupe Generator (generateDrugstoreDupe.js)

```

//  gpt/generateDrugstoreDupe.js
module.exports = async function generateDrugstoreDupe(openai, product) {
  async function gptPrompt(prompt) {
    const response = await openai.chat.completions.create({
      model: "gpt-4",
      messages: [
        { role: "system", content: "You are a skincare expert revealing affordable alternatives to luxury products." },
        { role: "user", content: prompt }
      ],
      temperature: 0.85
    });
    return response.choices[0].message.content.trim();
  }

  const [intro, content, outro] = await Promise.all([
    gptPrompt(`In 1-2 short sentences, introduce a great drugstore dupe for a high-end product. Keep it under 30 words.`),
    gptPrompt(`Describe how "${product}" is a great drugstore dupe for a more expensive product in 110 words or less. Compare the key ingredients, performance, and results.`),
    gptPrompt(`In 1 sentence, encourage readers to try "${product}" and save money. Keep it under 20 words.`)
  ]);

  return {

```

```

    type: "drugstoreDupe",
    intro,
    content,
    outro,
    watermark: "generated-by-glowbot-v1"
  };
};

```

Dry Skin Product List Generator (generateDrySkinList.js)

```

// 📁 gpt/generateDrySkinList.js
module.exports = async function generateDrySkinList(openai) {
  async function gptPrompt(prompt) {
    const response = await openai.chat.completions.create({
      model: "gpt-4",
      messages: [
        { role: "system", content: "You are a skincare expert known for recommending top-tier product picks for specific skin concerns." },
        { role: "user", content: prompt }
      ],
      temperature: 0.85
    });
    return response.choices[0].message.content.trim();
  }

  const [intro, content, outro] = await Promise.all([
    gptPrompt(`In 1-2 short sentences, introduce a list of the top skincare products for dry skin. Keep it under 30 words.`),
    gptPrompt(`List the top 5 skincare products for dry skin. For each, include the name and a short benefit in this format:
    1. CeraVe Moisturizing Cream - Deeply hydrating and packed with ceramides.
    Ensure you provide exactly 5 products.`),
    gptPrompt(`In 1 sentence, encourage readers to try a product and maintain consistency. Keep it under 20 words.`)
  ]);

  return {
    type: "drySkinList",
    intro,
    content,
    outro,
    watermark: "generated-by-glowbot-v1"
  };
};

```

Influencer Caption Generator (generateInfluencerCaption.js)

```

// 📁 gpt/generateInfluencerCaption.js
module.exports = async function generateInfluencerCaption(openai, product, tone = "influencer") {

```

```

    async function gptPrompt(prompt) {
      const response = await openai.chat.completions.create({
        model: "gpt-4",
        messages: [
          { role: "system", content: "You are a skincare influencer who writes captions that convert and engage." },
          { role: "user", content: prompt }
        ],
        temperature: 0.85
      });
      return response.choices[0].message.content.trim();
    }

    const [intro, content, outro] = await Promise.all([
      gptPrompt(`In 1-2 short sentences, introduce an influencer caption for "${product}". Keep it under 30 words.`),
      gptPrompt(`Write a short, punchy influencer-style caption for "${product}" in a ${tone} tone. Include 1-2 emojis and a light call to action.`),
      gptPrompt(`In 1 sentence, write a short outro that encourages followers to try "${product}". Keep it under 20 words.`)
    ]);

    return {
      type: "influencerCaption",
      intro,
      content,
      outro,
      watermark: "generated-by-glowbot-v1"
    };
  };
};

```

Full Content Output Generator (generateOriginal (Full Content Output).js)

```

// 📁 gpt/generateFullOriginal.js
module.exports = async function generateFullOriginal(openai, product, tone = "engaging") {
  async function gptPrompt(prompt) {
    const response = await openai.chat.completions.create({
      model: "gpt-4",
      messages: [
        { role: "system", content: "You are a skincare content expert." },
        { role: "user", content: prompt }
      ],
      temperature: 0.8
    });
    return response.choices[0].message.content.trim();
  }

  const [
    intro,
    productDescription,
    demoScript,
    problemSolution,

```

```

    personalReview,
    socialCaptions,
    hashtagSets,
    trendingHashtags,
    outro
  ] = await Promise.all([
    gptPrompt(`In 1-2 short sentences, write a hook-style intro for "${product}" in a ${tone} tone. Keep it under 30 words.`),
    gptPrompt(`Write a short 100 word and helpful product description for "${product}" in an ${tone} tone.`),
    gptPrompt(`Create a 2-3 sentence skincare demo script for "${product}" that feels casual and confident.`),
    gptPrompt(`Describe a common skincare problem and how "${product}" solves it. Keep it under 4 sentences.`),
    gptPrompt(`Write a 100 word personal-sounding review of "${product}" by someone who used it daily for 2 weeks.`),
    gptPrompt(`Give me 3 social media captions for a post about "${product}". Include 1-2 emojis in each.`),
    gptPrompt(`Give me 3 sets of hashtags for skincare posts. Each set should include 5 hashtags relevant to "${product}".`),
    gptPrompt(`Give me 5 trending skincare hashtags for 2024.`),
    gptPrompt(`In 1 sentence, write a confident outro that encourages people to try "${product}". Keep it under 20 words.`)
  ]);

  return {
    type: "original",
    content: "Full content bundle",
    intro,
    productDescription,
    demoScript,
    problemSolution,
    personalReview,
    socialCaptions,
    hashtagSets,
    trendingHashtags,
    outro,
    watermark: "generated-by-glowbot-v1"
  };
};

```

Personal Review Generator (generatePersonalReview.js)

```

// 📁 gpt/generatePersonalReview.js
module.exports = async function generatePersonalReview(openai, product) {
  async function gptPrompt(prompt) {
    const response = await openai.chat.completions.create({
      model: "gpt-4",
      messages: [
        { role: "system", content: "You are a skincare content creator writing authentic and helpful product reviews." },
        { role: "user", content: prompt }
      ],
    });
  }
}

```

```

        temperature: 0.85
    });
    return response.choices[0].message.content.trim();
}

const [intro, content, outro] = await Promise.all([
    gptPrompt(`In 1-2 short sentences, introduce your review for "${product}". Keep it under 20 words.`),
    gptPrompt(`Write a detailed, authentic-sounding 100 word or less personal review of "${product}". Include how long it was used, what results were seen, and how it felt during use.`),
    gptPrompt(`In 1 sentence, conclude your review and recommend who would benefit from using "${product}". Keep it under 20 words.`)
]);

return {
    type: "personalReview",
    intro,
    content,
    outro,
    watermark: "generated-by-glowbot-v1"
};
};

```

Product Comparison Generator (generateProductComparison.js)

```

const { enrichPromptWithData } = require('../utils/dataEnricher');

module.exports = async function generateProductComparison(openai, product, scraperData) {
    async function gptPrompt(prompt, contextData = '') {
        const response = await openai.chat.completions.create({
            model: "gpt-4",
            messages: [
                {
                    role: "system",
                    content: "You are a skincare expert who explains product comparisons using verified data and community insights."
                },
                {
                    role: "user",
                    content: contextData + '\n\n' + prompt
                }
            ],
            temperature: 0.8
        });
        return response.choices[0].message.content.trim();
    }

    const enrichedData = await enrichPromptWithData(product, scraperData);

    const [intro, content, outro] = await Promise.all([

```

```

    gptPrompt(
      `In 1-2 short sentences, introduce a product comparison for "${product}".
      Keep it under 20 words.`,
      enrichedData.contextualPrompt
    ),
    gptPrompt(
      `Compare "${product}" to its most discussed alternative based on the
      provided data. Focus on verified prices, ingredients, and community feedback.`,
      enrichedData.contextualPrompt
    ),
    gptPrompt(
      `Give a data-backed recommendation about "${product}" based on the price
      point and community sentiment.`,
      enrichedData.contextualPrompt
    )
  ]);

  return {
    type: "productComparison",
    intro,
    content,
    outro,
    metadata: {
      price: enrichedData.productDetails.price,
      rating: enrichedData.productDetails.rating,
      communityInsights: enrichedData.communityInsights.length
    },
    watermark: "generated-by-glowbot-v1"
  };
};

```

Pros and Cons Generator (generateProsAndCons.js)

```

// 📁 gpt/generateProsAndCons.js
module.exports = async function generateProsAndCons(openai, product) {
  async function gptPrompt(prompt) {
    const response = await openai.chat.completions.create({
      model: "gpt-4",
      messages: [
        { role: "system", content: "You are a balanced skincare reviewer who
        helps people make informed decisions." },
        { role: "user", content: prompt }
      ],
      temperature: 0.8
    });
    return response.choices[0].message.content.trim();
  }

  const [intro, content, outro] = await Promise.all([
    gptPrompt(`In 1-2 short sentences, introduce a pros and cons breakdown for
    "${product}". Keep it under 30 words.`),
    gptPrompt(`List 3 pros and 3 cons of "${product}". Be concise with 100
    words or less, but insightful, and maintain a balanced tone.`),
  ]);

```

```

    gptPrompt(`In 1 sentence, wrap up the pros and cons and recommend who
should try "${product}". Keep it under 20 words.`)
  ]);

  return {
    type: "prosAndCons",
    intro,
    content,
    outro,
    watermark: "generated-by-glowbot-v1"
  };
};

```

Routine Example Generator (generateRoutineExample.js)

```

// 📁 gpt/generateRoutineExample.js
module.exports = async function generateRoutineExample(openai, product) {
  async function gptPrompt(prompt) {
    const response = await openai.chat.completions.create({
      model: "gpt-4",
      messages: [
        { role: "system", content: "You are a skincare educator building sample
daily routines with product integrations." },
        { role: "user", content: prompt }
      ],
      temperature: 0.85
    });
    return response.choices[0].message.content.trim();
  }

  const [intro, content, outro] = await Promise.all([
    gptPrompt(`In 1-2 short sentences, introduce a simple skincare routine with
"${product}". Keep it under 30 words.`),
    gptPrompt(`Create a basic 4-step skincare routine in under 125 words (AM or
PM) that includes "${product}". For each step, include the step name and a
short note, e.g.,\n\n1. Cleanser - Gently removes dirt and oil.`),
    gptPrompt(`In 1 sentence, encourage consistency in the routine and remind
viewers of long-term benefits. Keep it under 20 words.`)
  ]);

  return {
    type: "routineExample",
    intro,
    content,
    outro,
    watermark: "generated-by-glowbot-v1"
  };
};

```


Surprise Me Generator (generateSurpriseMe.js)

```
// 📄 gpt/generateSurpriseMe.js
module.exports = async function generateSurpriseMe(openai, product) {
  async function gptPrompt(prompt) {
    const response = await openai.chat.completions.create({
      model: "gpt-4",
      messages: [
        { role: "system", content: "You are a creative skincare copywriter known for surprising and delighting the audience." },
        { role: "user", content: prompt }
      ],
      temperature: 0.95
    });
    return response.choices[0].message.content.trim();
  }

  const [intro, content, outro] = await Promise.all([
    gptPrompt(`Write a playful, mysterious 30 words intro that teases a surprise piece of content about "${product}".`),
    gptPrompt(`Generate a fun and unexpected 100 word or less piece of skincare content about "${product}". It could be a weird tip, a story, or a bold claim – surprise the reader in a positive way.`),
    gptPrompt(`End with a quirky or clever 20 word outro that makes the viewer smile or want to share the post.`)
  ]);

  return {
    type: "surpriseMe",
    intro,
    content,
    outro,
    watermark: "generated-by-glowbot-v1"
  };
};
```

TikTok Breakdown Generator (generateTikTokBreakdown.js)

```
// 📄 gpt/generateTikTokBreakdown.js
module.exports = async function generateTikTokBreakdown(openai, product) {
  async function gptPrompt(prompt) {
    const response = await openai.chat.completions.create({
      model: "gpt-4",
      messages: [
        { role: "system", content: "You are a skincare creator breaking down trending TikTok content with insights and attitude." },
        { role: "user", content: prompt }
      ],
      temperature: 0.9
    });
    return response.choices[0].message.content.trim();
  }

  const [intro, content, outro] = await Promise.all([
    gptPrompt(`In 1-2 short sentences, introduce why "${product}" is trending on TikTok. Keep it under 30 words.`),
    gptPrompt(`Break down why "${product}" is blowing up on TikTok in 125 words or less. Mention key creators, viral claims, and results people are seeing.`),
    gptPrompt(`In 1 sentence, encourage viewers to try "${product}" and share their thoughts. Keep it under 20 words.`)
  ]);

  return {
    type: "tiktokBreakdown",
    intro,
    content,
    outro,
    watermark: "generated-by-glowbot-v1"
  };
};
```

Top 5 Under \$25 Generator (generateTop5Under25.js)

```
// 📁 gpt/generateTop5Under25.js
module.exports = async function generateTop5Under25(openai) {
  async function gptPrompt(prompt) {
    const response = await openai.chat.completions.create({
      model: "gpt-4",
      messages: [
        { role: "system", content: "You are a savvy skincare content creator helping people find effective products on a budget." },
        { role: "user", content: prompt }
      ],
      temperature: 0.85
    });
    return response.choices[0].message.content.trim();
  }

  const [intro, content, outro] = await Promise.all([
    gptPrompt(`In 1-2 short sentences, introduce a list of skincare products under $25. Keep it under 30 words.`),
    gptPrompt(`List 5 great skincare products that cost less than $25. For each, include the product name and a short benefit using this format:\n\n1. The Ordinary Niacinamide 10% - Helps reduce oil and visibly smooth skin.`),
    gptPrompt(`In 1 sentence, motivate readers to buy affordable products. Keep it under 20 words.`)
  ]);

  return {
    type: "top5Under25",
    intro,
    content,
    outro,
    watermark: "generated-by-glowbot-v1"
  };
};
```

Trending Products List Generator (generateTrendingProducts.js)

```
// 📁 gpt/generateTrendingProducts.js
module.exports = async function generateTrendingProducts(openai) {
  const prompt = `Give me a list of 5 skincare products that are currently
trending on TikTok or Amazon.
Only return the product names, numbered 1 through 5. No extra text.`;

  const response = await openai.chat.completions.create({
    model: "gpt-4",
    messages: [
      { role: "system", content: "You are a skincare market analyst tracking
trending beauty products." },
      { role: "user", content: prompt }
    ],
    temperature: 0.7
  });

  const text = response.choices[0].message.content.trim();
  const products = text
    .split('\n')
    .map(line => line.replace(/^\d+\.\s*/, '').trim())
    .filter(Boolean);

  return products;
};
```

Why I Switched Generator (generateWhyISwitched.js)

```
// 📄 gpt/generateWhyISwitched.js
module.exports = async function generateWhyISwitched(openai, product) {
  async function gptPrompt(prompt) {
    const response = await openai.chat.completions.create({
      model: "gpt-4",
      messages: [
        { role: "system", content: "You are a skincare storyteller writing persuasive switch-style content." },
        { role: "user", content: prompt }
      ],
      temperature: 0.85
    });
    return response.choices[0].message.content.trim();
  }

  const [intro, content, outro] = await Promise.all([
    gptPrompt(`In 1-2 short sentences, introduce a story about switching to "${product}". Keep it under 30 words.`),
    gptPrompt(`Write a clear 100 word explanation of why the person switched to "${product}". Describe the differences they noticed, the improvements it made, and how it felt.`),
    gptPrompt(`In 1 sentence, recommend others try "${product}" and explain why. Keep it under 20 words.`)
  ]);

  return {
    type: "whyISwitched",
    intro,
    content,
    outro,
    watermark: "generated-by-glowbot-v1"
  };
};
```