

How to Create SKILL using antigravity skill Creator?

1. Test Plan Generator Skill

1. Add the File from here into Anti Gravity

[AI-Tester-Blueprint/chapter_01_foundation_model/antigravity-skill-creator.md at main · PramodDutta/AI-Tester-Blueprint](https://github.com/PramodDutta/AI-Tester-Blueprint/blob/main/Chapter_01_foundation_model/antigravity-skill-creator.md)

2. Write following prompt in Antigravity

@antigravity-skill-creator.md Create a SKILL Where you will be helping me to generate test plan document for me, It standard document, End user upload test plan document, you need to verify it, if it is not proper then you make it proper, Base on uploaded test plan document you have to create new one.

3. Output:

```
---
name: generating-test-plans
description: Generates, verifies, and improves test plan documents. Use when the user needs to create a new test plan, evaluate an existing one, or transform a draft into a professional standard document.
---

# Generating Test Plans

This skill provides a structured approach to creating, verifying, and refining test plan documents to ensure high-quality software testing.

## When to use this skill
- When the user uploads a test plan and wants it verified against industry standards.
- When the user needs to generate a new test plan from scratch or based on requirement documents.
- When an existing test plan needs to be improved, updated, or reformatted.
```

Workflow

To process a test plan, follow this "Verify-Plan-Refine" pattern:

1. ****Verify Phase**** (If a document is uploaded):
 - Review the uploaded document using the [Verification Checklist] ([resources/verification-checklist.md](#)) .
 - Identify missing sections or weak descriptions (e.g., vague test objectives, missing exit criteria).
 - Provide feedback to the user on what is missing or needs improvement.
2. ****Generation/Update Phase****:
 - Use the [Standard Test Plan Template] ([resources/test-plan-template.md](#)) as a base.
 - If generating from an existing document, merge the existing content into the template, filling in gaps and enhancing details.
 - If generating from requirements, map requirements to test objectives and scope.
3. ****Refinement Phase****:
 - Ensure the tone is professional and technical.
 - Verify that all specific user constraints are met.
 - Present the finalized test plan to the user.

Instructions

1. Document Analysis

When a user uploads a document, check for these core components:

- ****Test Items****: What is being tested?
- ****Features to be Tested****: Specific functional/non-functional areas.
- ****Features NOT to be Tested****: Explicit exclusions.
- ****Pass/Fail Criteria****: How do we know it passed?
- ****Suspension/Resume Criteria****: When to stop testing (e.g., blocking bugs).
- ****Deliverables****: What reports will be generated?
- ****Risks and Contingencies****: What could go wrong?

2. Improvement Rules

- ****Specific****: Ensure steps and criteria are not ambiguous.
- ****Actionable****: All test items should lead to a clear test activity.

- ****Format**:** Use Markdown headers and lists for readability.

```
## Resources
- [Standard Test Plan Template] (resources/test-plan-template.md)
- [Verification Checklist] (resources/verification-checklist.md)
```

2. Create a ATS Friendly Resume

1. Add the File from here into Anti Gravity

[AI-Tester-Blueprint/chapter_01_foundation_model/antigravity-skill-creator.md at main · PramodDutta/AI-Tester-Blueprint](#)

2. Write following prompt in Antigravity

@[antigravity-skill-creator.md](#) Create a SKILL where you will be helping me to generate resume base on user input, you will take the user input and make ATS friendly resume, ATS means (applicant tracking system) ensures your application is accurately read, boosting your odds of advancing past the first round,

A clear structure paired with relevant keywords helps avoid rejection due to formatting glitches, it also consider following

Keywords (the BIG one)

ATS compares your resume to the job description.

They look for:

Skills (e.g., Python, SQL, Project Management)

Job titles (Data Analyst, HR Executive)

Tools & software (SAP, Excel, Salesforce)

Certifications (PMP, AWS, CPA)

👉 Tip: Use the exact wording from the job posting.
“Data Analysis” ≠ “Analyzing Data” to an ATS.

② Job Title Match

Your current or recent job title should closely match the target role.

ATS ranks resumes higher when titles align.

Example:

Job asks for “Marketing Specialist”

Resume says “Digital Marketing Executive” → OK

Resume says “Creative Lead” → weaker match

③ Skills Section (Hard Skills > Soft Skills)

ATS heavily prioritizes:

Technical skills

Tools

Programming languages

Industry-specific skills

Soft skills (team player, leadership) matter more to recruiters after ATS approval.

④ Work Experience (Relevance + Recency)

ATS checks:

Years of experience

Relevant roles

Consistency in employment

Recent experience weighs more than older roles

👉 Use reverse chronological order.

⑤ Simple Formatting (No Fancy Stuff)

ATS struggles with:

Tables

Columns

Graphics

Icons

Text boxes

Headers/footers

Infographics

Safe formatting:

Single column

Standard headings: Experience, Skills, Education

Bullet points (• or –)

Word (.docx) or clean PDF

6 Education & Certifications

Checked for:

Degree level

Field of study

Certifications mentioned in the JD

Example:

If JD says “Bachelor’s in Computer Science required” — ATS checks that explicitly.

7 Action Verbs + Metrics (for Ranking)

Not just presence — but strength.

Better:

“Increased sales by 32% using CRM automation”

vs

“Responsible for sales activities”

Numbers help your resume rank higher.

⑧ File Name & Language

File name: FirstName_LastName_JobTitle.pdf

Language should match the job posting (English vs local language)

ATS Resume Quick Checklist ✓

- ✓ Match keywords from JD
- ✓ Use standard job titles
- ✓ Simple layout
- ✓ Strong skills section
- ✓ Quantified achievements
- ✓ No graphics or tables

3. Output

```
---
```

`name: generating-ats-resumes`

`description: Generates and optimizes resumes for Applicant Tracking Systems (ATS). Use when the user needs to create a professional resume, align their experience with a job description, or ensure for document compatibility with automated screening tools.`

```
---
```

`# Generating ATS-Friendly Resumes`

This skill specializes in creating resumes that pass through Applicant Tracking Systems (ATS) by focusing on keyword alignment, standard formatting, and quantified achievements.

`## When to use this skill`

- When a user provides a job description (JD) and wants their resume tailored to it.

- When a resume needs to be converted from a "fancy" layout to an ATS-safe format.
- When the user wants to identify missing keywords or skills based on a target role.

Workflow

1. **Input Analysis**:

- Request the **Job Description (JD)** and the user's **Current Resume/Profile**.
- Extract primary keywords (Skills, Tools, Job Titles) from the JD.

2. **Optimization Phase**:

- **Keyword Mapping**: Align the user's skills with the exact terminology used in the JD.
- **Job Title Alignment**: Suggest adjustments to current titles if they are non-standard but functionally equivalent to the target role.
- **Bullet Point Refinement**: Transform generic tasks into "Action Verb + Metric" statements.

3. **Formatting & Generation**:

- Use the [ATS-Safe Template] ([resources/ats-resume-template.md](#)) .
- Ensure a single-column layout with no tables, headers, or complex graphics.
- Provide the final version in Markdown or clear text for easy conversion to .docx/PDF.

4. **Validation**:

- Verify the output against the [ATS Quick Checklist] ([resources/ats-checklist.md](#)) .

Instructions

1. Keyword Extraction Rules

- Focus on **Hard Skills** first (e.g., Python, Salesforce, Project Management).
- Use exact phrasing: if JD says "Data Analysis," do not use "Analyzing Data."
- Include certifications and education explicitly mentioned as "Preferred" or "Required."

```
### 2. The "Action + Result" Formula
Refactor every experience bullet point to follow this structure:
> **[Strong Action Verb]** + **[Task/Project]** + **[Quantifiable Metric/Outcome]**  

*Example: "Managed team" -> "Led a team of 10 developers to deliver 3 high-priority modules, reducing technical debt by 25%."*
```

```
### 3. Formatting Restrictions
- **No Tables**: ATS often breaks text inside tables.
- **No Columns**: Use a flat, single-column top-to-bottom flow.
- **Standard Headings**: Use only "Experience," "Skills," "Education," and "Summary."
```

```
## Resources
- [ATS-Safe Resume Template] (resources/ats-resume-template.md)
- [ATS Quick Checklist] (resources/ats-checklist.md)
```

3. Create a SKILL simple LinkedIn POST Generator

1. Add the File from here into Anti Gravity

[AI-Tester-Blueprint/chapter_01_foundation_model/antigravity-skill-creator.md at main · PramodDutta/AI-Tester-Blueprint](#)

2. Write following prompt in Antigravity

@antigravity-skill-creator.md Create a skill where you will be helping me to create clickbite Linkedin Post for me, User give input topic, base on the input you will generate post concise and shorter with hashtags so my post goes viral with my connection, do not use offensive and irrelevant content, make sure you follow the linkedin guidelines.

3. Output

```
---  
name: creating-viral-linkedin-posts  
description: Generates engaging, concise, and high-impact LinkedIn posts designed for maximum reach and engagement. Use when the user wants to create professional yet "clickworthy" content based on a specific topic.  
---
```

Creating Viral LinkedIn Posts

This skill helps you craft high-engagement LinkedIn posts that are concise, structured for readability, and optimized for the LinkedIn algorithm.

When to use this skill

- When you have a topic or idea and want to turn it into a LinkedIn post.
- When you want to increase visibility and engagement with your professional network.
- When you need a quick, safe, and professional post with relevant hashtags.

Workflow

1. **Ideation**:

- Take the user's input topic.
- Brainstorm a "hook" (the first line) that grabs attention within the first 3 lines of the feed.

2. **Drafting**:

- Keep sentences short.
- Use line breaks for white space.
- Ensure the tone is professional, insightful, and bold.

3. **Optimization**:

- Add a "Call to Action" (CTA) to encourage comments.
- Include 3-5 high-relevance hashtags.
- Verify against [LinkedIn Safety Guidelines] ([resources/safety-guidelines.md](#)) .

4. **Review**:

- Present the post to the user.
- Provide 2 variations: (1) Story-driven and (2) List/Insight-driven.

Instructions

1. The "Hook" Logic

The first sentence is everything. It must stop the scroll.

```
- Good Hooks: "I learned this the hard way...", "Most people think X, but the truth is Y.", "3 steps to achieve [Goal] in half the time."
```

2. Formatting Rules

- **No giant paragraphs**: One sentence per line where possible.
 - **Emojis**: Use subtle emojis to emphasize points (e.g., 🚀, ✅,💡).
- Don't overdo it.
- **CTA**: Always end with a question to spark conversation.

3. Content Safety

- **No Offensive Content**: Avoid controversial topics that violate professional standards.
- **No Spam**: Ensure the content provides actual value, not just "clickbait" without substance.
- **Guideline Compliance**: Strictly follow LinkedIn's professional community policies.

Resources

- [LinkedIn Post Templates] ([resources/post-templates.md](#))
- [Safety & Verification Guidelines] ([resources/safety-guidelines.md](#))

4. Create a SKILL for Tweet Post

1. Add the File from here into Anti Gravity

[AI-Tester-Blueprint/chapter_01_foundation_model/antigravity-skill-creator.md at main · PramodDutta/AI-Tester-Blueprint](#)

2. Write following prompt in Antigravity

@antigravity-skill-creator.md

Create a skill where you will be helping me to create clickbite tweet for x.com, User give input topic, base on the input you will generate post concise and shorter with hashtags so my post goes viral with my connection, do not use offensive and irrelevant content, make sure you follow the linkedin guidelines.

3. Output

```
---  
name: creating-viral-tweets
```

```
description: Generates concise, punchy, and high-impact tweets for X.com  
(Twitter) designed for viral reach. Use when the user wants to turn a  
topic into a short, engaging post or a multi-tweet thread.
```

```
---
```

Creating Viral Tweets

This skill specializes in crafting high-engagement content for X.com, focusing on brevity, "punchy" hooks, and the specific writing style that performs well on the platform.

When to use this skill

- When you want to share an insight, news, or opinion in a very concise format.
- When you need to summarize a complex topic into a single tweet or a thread.
- When you want to maximize visibility using X-native formatting and hashtags.

Workflow

1. **Extraction**:

- Identify the "core insight" from the user's topic.
- Determine the intended tone (e.g., Bold, Educational, Humorous, Provocative).

2. **Drafting**:

- Write the "Hook" (the first sentence) to be ultra-short and intriguing.
- Use the remaining space for the "Meat" (the value) and a "Call to Action" (CTA).
 - Ensure the total length is well within 280 characters (aim for <200 for better readability).

3. **Viral Pass**:

- Apply "X Formatting": Use line breaks, lists, and bold declarations.
- Add 1-2 relevant hashtags (X algorithm prefers fewer, more specific hashtags).

4. **Review**:

- Check against [Twitter Safety Guidelines] ([resources/twitter-safety-guidelines.md](#)) .
- Offer the user a single tweet version and a "Thread Starter" version.

Instructions

1. The "X Style" Guide

- **Brevity is King**: If you can say it in 5 words, don't use 10.
- **Strong Openers**: Start with a punchy statement (e.g., "Software engineering is 10% coding, 90% thinking.").
- **White Space**: Don't use blocks of text. Use line breaks to make the tweet scroll-friendly.

2. Thread Logic (Optional)

If the topic is too big for one tweet, output a "Thread" format:

- **Tweet 1**: The Hook + "A thread 
- **Tweets 2-N**: Sequential insights.
- **Final Tweet**: Summary + CTA (Follow/Retweet).

3. Safety & Guidelines

- Follow [Twitter Safety Guidelines] ([resources/twitter-safety-guidelines.md](#)) .
- Avoid misinformation, offensive language, or spammy behavior.
- Ensure content is relevant and adds value to the conversation.

Resources

- [Viral Tweet Templates] ([resources/tweet-templates.md](#))
- [Twitter Safety Guidelines] ([resources/twitter-safety-guidelines.md](#))