

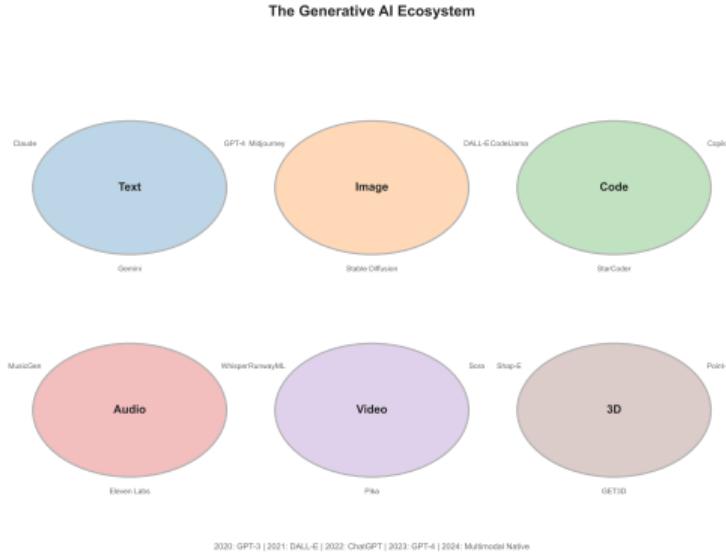
## Week 6: Generative AI for Rapid Prototyping

Design & AI Program

# Agenda

- 1 Foundation: The Prototyping Revolution
- 2 Algorithms: The Magic Behind Generation
- 3 Implementation: Making It Real
- 4 Design: Human-AI Creative Partnership
- 5 Practice: Build Something Amazing

# The Prototyping Revolution



## From Idea to Prototype

### In Minutes, Not Months

- Text generation: 175B parameters
- Image creation: 1024x1024 in seconds
- Code synthesis: Full applications
- Design iteration: 100x faster

Generative AI transforms the innovation timeline

# Traditional vs AI-Enhanced Prototyping

## Traditional Prototyping

- Weeks of manual design
- High resource requirements
- Limited iteration cycles
- Sequential workflows
- Expert dependency

Timeline: 4-12 weeks

Cost: \$10,000-50,000

Iterations: 3-5

## AI-Enhanced Prototyping

- Hours to first prototype
- Minimal resources needed
- Unlimited iterations
- Parallel exploration
- Democratized creation

Timeline: 1-3 days

Cost: \$100-1,000

Iterations: 100+

10x speed improvement, 50x cost reduction, unlimited creativity

# What is Generative AI?

## Core Concept

AI that creates new content from learned patterns

## Key Capabilities:

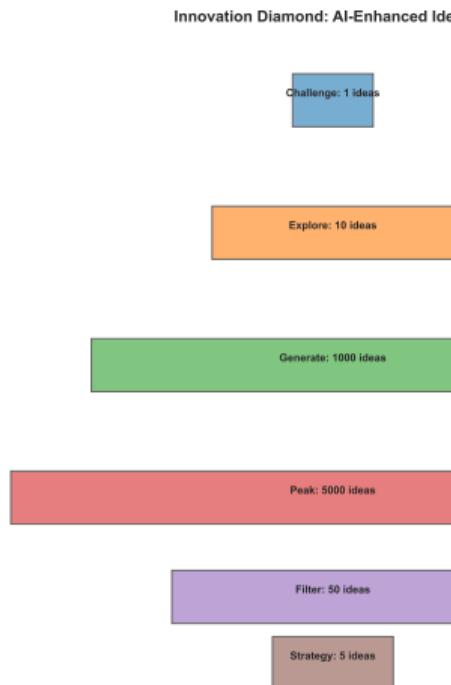
- ① Learn from existing data
- ② Generate novel outputs
- ③ Maintain coherence & quality
- ④ Respond to guidance (prompts)

Each modality opens new prototyping possibilities

## Generation Types

- **Text:** GPT-4, Claude, Gemini
- **Image:** DALL-E 3, Midjourney, Stable Diffusion
- **Code:** Copilot, Codex, CodeLlama
- **Audio:** MusicGen, Whisper
- **Video:** RunwayML, Pika Labs
- **3D:** Point-E, Shap-E

# Generative AI in the Innovation Diamond



## 1. Challenge

Define problem space

## 2. Explore

AI suggests directions

## 3. Generate

AI creates 1000s of variants

## 4. Filter

AI ranks & selects

## 5. Strategy

AI refines final concepts

AI amplifies each stage of innovation

## Real-World Impact: Success Stories

### Airbnb

AI-generated property descriptions  
+12% booking rate  
50% time saved

### GitHub Copilot

Code generation assistant  
55% faster coding  
40% fewer bugs

### Canva

Magic Design feature  
10M designs/day  
3x user engagement

### Notion AI

Content generation  
2M+ users  
70% productivity gain

### Adobe Firefly

Creative generation  
1B+ assets created  
90% time reduction

### Jasper AI

Marketing content  
\$125M revenue  
10x content speed

Companies using GenAI report 30-70% efficiency gains

## Foundation Models

- **GPT-4:** 1.76T parameters
- **Claude 3:** Constitutional AI
- **Gemini Ultra:** Multimodal native
- **LLaMA 2:** Open source, 70B
- **Mistral:** Efficient, 7B-8x7B

## Specialized Models

- **Code:** StarCoder, CodeT5
- **Science:** Galactica, BioGPT
- **Math:** Minerva, MathGPT

## Creative Models

- **DALL-E 3:** Text-to-image leader
- **Midjourney v6:** Artistic quality
- **Stable Diffusion XL:** Open, customizable
- **Imagen:** Google's photorealism
- **Parti:** 20B parameter vision

## Emerging Frontiers

- **Video:** Sora, Gen-2, Pika
- **3D:** GET3D, Magic3D
- **Audio:** AudioCraft, Jukebox

New models launch weekly - staying current is critical

# Why Now? The Convergence

## Technical Breakthroughs

- Transformer architecture (2017)
- Attention mechanisms
- Self-supervised learning
- Diffusion models
- RLHF (Human feedback)

## Infrastructure Scale

- 10,000+ GPU clusters
- Trillion parameter models
- Petabyte training datasets
- Cloud API access

## Market Drivers

- \$10B+ investment (2023)
- 500+ GenAI startups
- Enterprise adoption: 67%
- Developer tools mature
- Regulatory frameworks emerging

## Cultural Shift

- AI literacy growing
- Creative acceptance
- Workflow integration
- Educational programs

2024: The year generative AI became essential for innovation

# Key Concepts: Building Blocks

## Prompting

The art of instructing AI models

- Context setting
- Role definition
- Output formatting
- Iterative refinement

## Temperature & Sampling

Controlling creativity vs consistency

- Temperature: 0.0 - 2.0
- Top-p: Nucleus sampling
- Top-k: Vocabulary limiting

## Context Windows

Input/output limitations

- GPT-4: 128K tokens
- Claude: 200K tokens
- Gemini: 1M tokens
- Local models: 4-32K

## Fine-tuning vs RAG

Customization approaches

- Fine-tuning: Model adaptation
- RAG: Retrieval augmented
- LoRA: Efficient tuning
- Prompt tuning: Soft prompts

Master these concepts to unlock GenAI potential

# The Promise and The Challenge

## The Promise

- Democratized creativity
- Infinite iterations
- Cross-domain synthesis
- 24/7 availability
- Multilingual/multimodal
- Personalization at scale

## Potential Impact:

30% GDP growth by 2030  
\$15.7 trillion economic value

## The Challenge

- Quality consistency
- Hallucination risks
- Bias amplification
- IP/copyright concerns
- Computational costs
- Skills gap

## Critical Questions:

How do we validate?  
Who owns the output?  
What about ethics?

Success requires balancing innovation with responsibility

## Key Takeaways

- ① GenAI transforms prototyping speed
- ② Multiple modalities available
- ③ Foundation models democratize creation
- ④ Prompting is the new programming
- ⑤ Integration & isolation

## Next: Algorithms

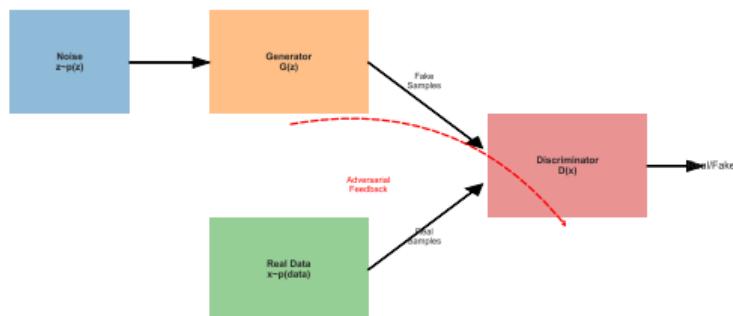
We'll explore:

- How GANs create images
- VAE latent spaces
- Diffusion model magic
- Transformer attention
- Architecture tradeoffs

Ready to dive deeper?

# Generative Adversarial Networks (GANs)

GAN Architecture: The Creative Duel



## The Creative Duel

### Generator (G)

- Creates fake samples
- Learns from noise
- Tries to fool discriminator

### Discriminator (D)

- Detects real vs fake
- Provides feedback
- Forces improvement

Adversarial training creates photorealistic outputs

## The Minimax Game

$$\min_G \max_D V(D, G)$$

$$V(D, G) = \mathbb{E}_{x \sim p_{\text{data}}} [\log D(x)] + \mathbb{E}_{z \sim p_z} [\log(1 - D(G(z)))]$$

## Training Process:

- ① Update D to maximize V
- ② Update G to minimize V
- ③ Repeat until equilibrium

GANs excel at high-fidelity image generation

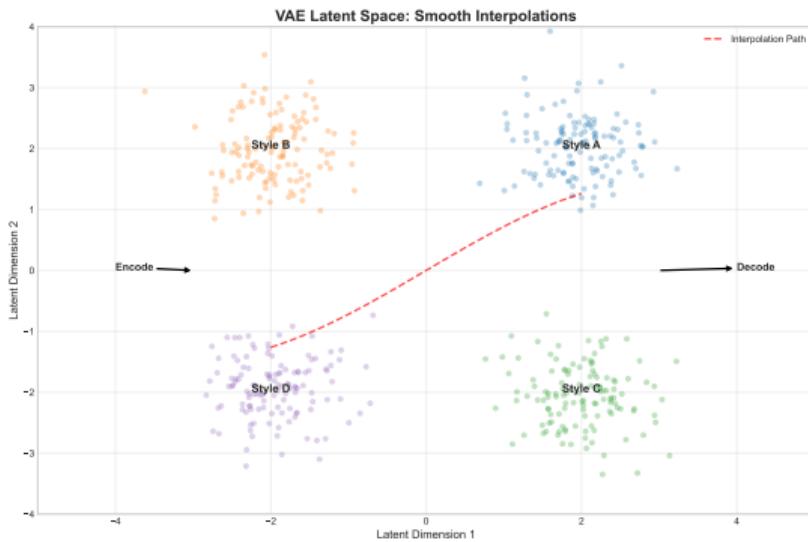
## Common Variants

- **StyleGAN:** High-quality faces
- **CycleGAN:** Image translation
- **BigGAN:** Large-scale generation
- **ProGAN:** Progressive growing
- **WGAN:** Wasserstein distance

## Applications:

- Photorealistic images
- Style transfer
- Super-resolution
- Data augmentation

# Variational Autoencoders (VAEs)



## Latent Space Magic

### Encoder

- Maps input to latent space
- Learns mean and variance
- Creates compressed representation

### Decoder

- Reconstructs from latent
- Generates variations
- Smooth interpolations

VAEs enable controlled generation via latent manipulation

## The ELBO Objective

Maximize Evidence Lower Bound:

$$\mathcal{L} = \mathbb{E}_{q(z|x)}[\log p(x|z)] - KL(q(z|x)||p(z))$$

### Components:

- Reconstruction term
- KL regularization
- Latent prior  $p(z) = \mathcal{N}(0, I)$

## Key Advantages

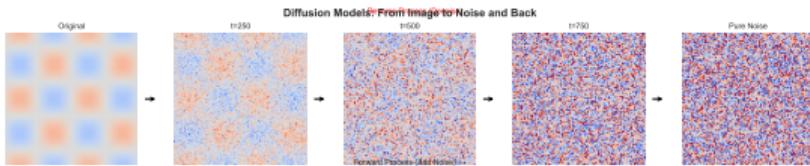
- Principled probabilistic framework
- Smooth latent space
- Interpolation capabilities
- Disentangled representations
- Fast generation

## Limitations:

- Blurry outputs
- Posterior collapse
- Limited complexity

VAEs trade quality for controllability and speed

# Diffusion Models: The New Paradigm



## Noise to Art

### Forward Process

- Add Gaussian noise
- $T$  timesteps
- Destroys information

### Reverse Process

- Learn to denoise
- Predict noise at each step
- Generate from pure noise

Diffusion models achieve state-of-the-art quality

# How Diffusion Models Work

## Training Process

- ① Sample image  $x_0$
- ② Sample timestep  $t$
- ③ Add noise:  $x_t = \sqrt{\alpha_t}x_0 + \sqrt{1 - \alpha_t}\epsilon$
- ④ Predict noise:  $\epsilon_\theta(x_t, t)$
- ⑤ Loss:  $||\epsilon - \epsilon_\theta||^2$

## Key Innovations:

- DDPM: Denoising foundation
- DDIM: Deterministic sampling
- Classifier guidance
- Latent diffusion (Stable Diffusion)

## Why Diffusion Wins

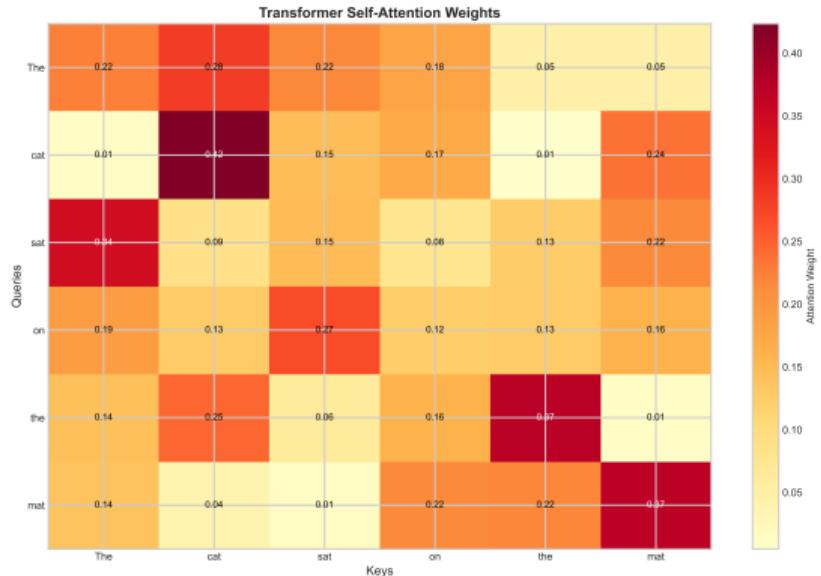
- Superior image quality
- Stable training
- Mode coverage
- Text conditioning natural
- Controllable generation

## Leading Models:

- **DALL-E 3:** OpenAI's best
- **Stable Diffusion:** Open source
- **Imagen:** Google's approach
- **Midjourney:** Artistic focus

Diffusion models dominate current image generation

# Transformers: The Universal Architecture



## Attention is All You Need

### Self-Attention

- Query, Key, Value
- Parallel processing
- Long-range dependencies

### For Generation:

- Autoregressive decoding
- Context understanding
- Multi-modal fusion

Transformers unified NLP and vision

## Text Generation

- **GPT Series:** Decoder-only
- **T5:** Encoder-decoder
- **BERT:** Bidirectional (not generative)
- **XLNet:** Permutation language model

## Scaling Laws:

Performance  $\propto$  (Parameters)<sup>0.5</sup>

GPT-3: 175B → GPT-4: 1.76T

Transformers scale to trillions of parameters

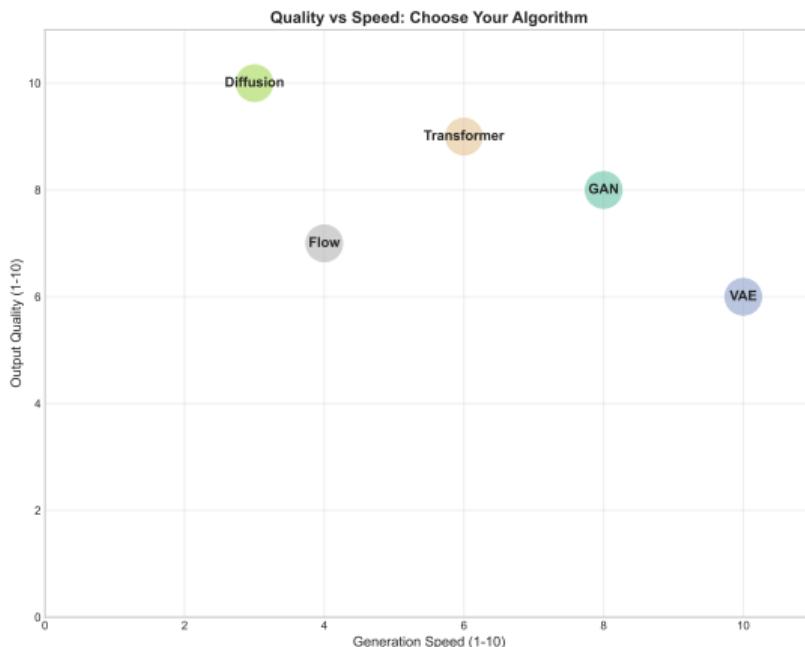
## Vision Transformers

- **ViT:** Image patches as tokens
- **CLIP:** Vision-language alignment
- **DALL-E:** Discrete VAE + GPT
- **Parti:** Autoregressive images

## Multimodal:

- **Flamingo:** Few-shot vision-language
- **BLIP-2:** Efficient bridging
- **Gemini:** Native multimodal

# Algorithm Comparison Matrix



## Choose Your Fighter

### Quality Leaders:

- Diffusion models
- Large transformers

### Speed Champions:

- VAEs
- Distilled models

### Control Masters:

- Conditional GANs
- Guided diffusion

No single best - depends on use case

# Head-to-Head: Algorithm Showdown

Metric	GAN	VAE	Diffusion	Transformer	Flow
Quality	High	Medium	Highest	High	Medium
Speed	Fast	Fastest	Slow	Medium	Slow
Training Stability	Low	High	High	High	Medium
Control	Medium	High	Highest	High	Low
Diversity	Low	Medium	Highest	High	High
Memory	Low	Lowest	High	Highest	Medium
Interpretability	Low	High	Medium	High	Medium

## Recommendations by Use Case:

Real-time: VAE, Small GAN

Quality: Diffusion, Large Transformer

Research: All architectures

Hybrid approaches often combine strengths

## Current Frontiers

- **Consistency Models:** 1-step generation
- **Flow Matching:** Optimal transport
- **Score-Based:** Continuous time
- **Energy Models:** Physics-inspired
- **Neural ODEs:** Continuous depth

## Efficiency Focus:

- Model distillation
- Quantization (INT8, INT4)
- Sparse models
- Flash attention

## Next 2 Years

- 10T parameter models
- Real-time video generation
- Perfect 3D synthesis
- Autonomous agents
- Multimodal natives

## Research Directions:

- Controllable generation
- Compositional reasoning
- Efficient architectures
- Robust evaluation
- Alignment techniques

The field evolves weekly - continuous learning essential

## Key Insights

- ① Diffusion models lead quality
- ② Transformers dominate scale
- ③ VAEs offer speed/control
- ④ GANs excel at specific domains
- ⑤ Hybrids emerging

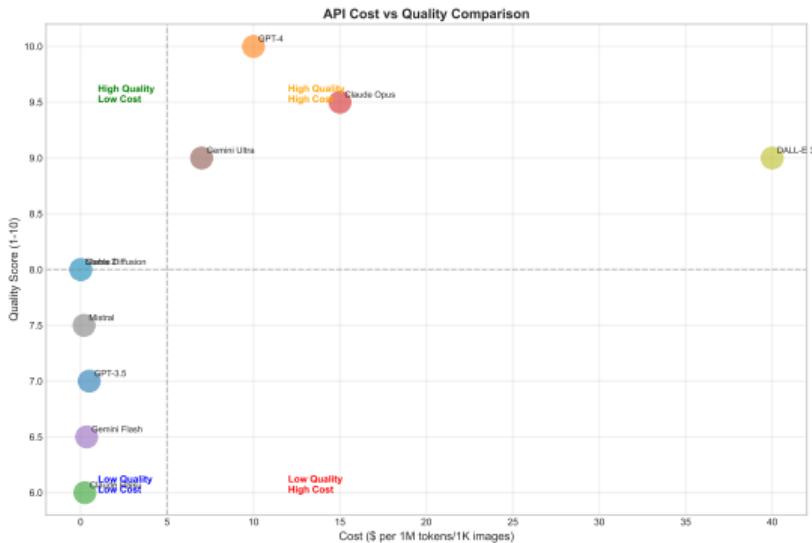
## Next: Implementation

We'll explore:

- API integration
- Prompt engineering
- Fine-tuning strategies
- Production pipelines
- Cost optimization

From Theory to Practice!

# The API Ecosystem



## Choose Your Platform

### Cost Leaders:

- Claude 3 Haiku: \$0.25/1M
- Gemini Flash: \$0.35/1M
- GPT-3.5: \$0.50/1M

### Quality Leaders:

- GPT-4 Turbo: \$10/1M
- Claude 3 Opus: \$15/1M
- Gemini Ultra: \$7/1M

Prices change monthly - always verify current rates

## OpenAI Example

```
from openai import OpenAI
client = OpenAI(api_key="sk-...")
response = client.chat.completions.create(
    model="gpt-4-turbo-preview",
    messages=[
        {"role": "system",
         "content": "You are a designer"},
        {"role": "user",
         "content": "Design a chair"}
    ],
    temperature=0.7
)
```

## Best Practices

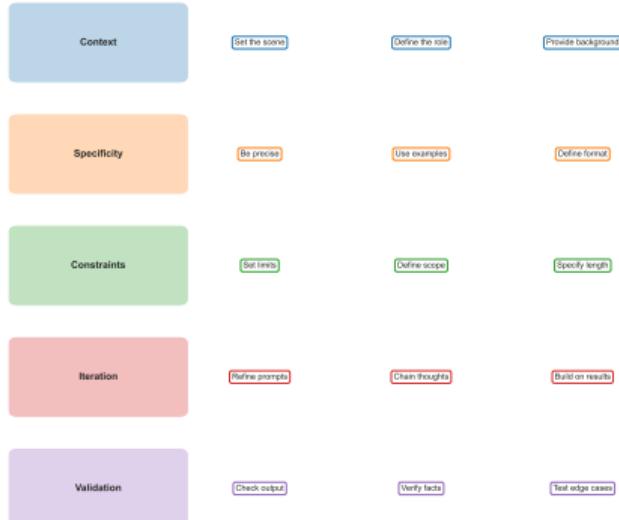
- Use environment variables for keys
- Implement retry logic
- Handle rate limits gracefully
- Log all requests/responses
- Monitor costs in real-time

## Error Handling:

- Timeout: 30s default
- Rate limit: Exponential backoff
- API errors: Fallback models

Robust implementation prevents production failures

## Prompt Engineering: The 5 Pillars



## The New Programming

### Core Techniques:

- Few-shot learning
- Chain-of-thought
- Role playing
- Output formatting
- Negative prompting

### Advanced:

- Constitutional AI
- Tree of thoughts
- ReAct patterns
- Self-consistency

Good prompts 10x output quality

# Prompt Templates That Work

## Product Design Template

Role: Senior product designer  
Task: Design [product] for [audience]  
Context: [market trends, constraints]  
Style: Minimalist, user-centered  
Output:  
1. Problem statement  
2. 3 concept sketches  
3. Key features list  
4. User journey  
Examples: [provide 2-3 examples]

## Code Generation

Language: Python  
Task: Implement [function]  
Requirements:  
- Type hints  
- Docstrings  
- Error handling  
- Unit tests  
Style: PEP 8 compliant

## Content Creation

Audience: [define precisely]  
Tone: [professional/casual/academic]  
Length: [word/character count]  
Structure:  
- Hook (1 sentence)  
- Main points (3-5)  
- Call to action  
Keywords: [SEO terms]  
Avoid: [list restrictions]

## Data Analysis

Data: [describe dataset]  
Goal: [specific objective]  
Methods: [allowed techniques]  
Output format:  
- Summary statistics  
- Key insights  
- Visualizations  
- Recommendations

Templates ensure consistency and completeness

## Cloud APIs

### Pros:

- No infrastructure needed
- Latest models instantly
- Infinite scale
- Pay-per-use

### Cons:

- Ongoing costs
- Internet dependency
- Data privacy concerns
- Vendor lock-in

**Best for:** Prototypes, variable load

Hybrid approach often optimal: prototype cloud, deploy local

## Local Models

### Pros:

- Full control
- No usage costs
- Data stays private
- Customization possible

### Cons:

- Hardware requirements
- Setup complexity
- Model limitations
- Maintenance burden

**Best for:** Production, sensitive data

## When to Fine-tune

- Domain-specific language
- Consistent style needed
- Proprietary knowledge
- Performance optimization
- Cost reduction at scale

## Techniques:

- **Full fine-tuning:** All parameters
- **LoRA:** Low-rank adaptation
- **QLoRA:** Quantized LoRA
- **Prefix tuning:** Soft prompts
- **Adapter layers:** Modular

Fine-tuning can 10x performance for specific tasks

## Process

- ① Collect domain data (1000+ examples)
- ② Clean and format
- ③ Choose base model
- ④ Set hyperparameters
- ⑤ Train (4-24 hours)
- ⑥ Evaluate thoroughly
- ⑦ Deploy with monitoring

## Costs:

- GPT-3.5: \$0.008/1K tokens
- Llama 2: Free (compute only)
- Custom: \$500-5000 typical

## Knowledge at Scale

Rag Architecture  
[Detailed Diagram]

### Components:

- Document store
- Embedding model
- Vector database
- Retriever
- Generator

### Benefits:

- Up-to-date information
- Reduced hallucination
- Domain expertise
- Citation capability

RAG enables ChatGPT for your data

Production Pipeline  
[Detailed Diagram]

## Scale with Confidence

### Key Components:

- Load balancer
- Request queue
- Model servers
- Cache layer
- Monitoring

### Optimization:

- Batch processing
- Response caching
- Model quantization
- Edge deployment

Production requires 10x more than just the model

## Reduce Token Usage

- Compress prompts (70% reduction)
- Cache common responses
- Use smaller models first
- Implement token limits
- Batch similar requests

## Smart Model Selection:

- GPT-3.5 for drafts
- GPT-4 for refinement
- Specialized models for tasks
- Local models for high volume

## Architecture Optimization

- Edge caching (50% savings)
- Request deduplication
- Async processing
- Progressive enhancement
- Fallback chains

## Monitoring:

- Cost per user
- Token efficiency
- Cache hit rate
- Model performance
- Error rates

Optimization can reduce costs by 80% without quality loss

## Security Concerns

- Prompt injection attacks
- Data leakage risks
- Model theft attempts
- Adversarial inputs
- API key exposure

## Mitigation:

- Input validation
- Output filtering
- Rate limiting
- Encryption everywhere
- Regular audits

## Ethical Guidelines

- Transparency about AI use
- Human oversight required
- Bias monitoring
- Fair use compliance
- User consent

## Best Practices:

- Never process PII
- Implement content filters
- Document AI decisions
- Enable opt-out
- Regular bias testing

Responsible AI is not optional – it's essential

## Key Takeaways

- ① APIs enable rapid prototyping
- ② Prompting is a core skill
- ③ Hybrid deployment optimal
- ④ Cost optimization critical
- ⑤ Security cannot be ignored

## Next: Design Integration

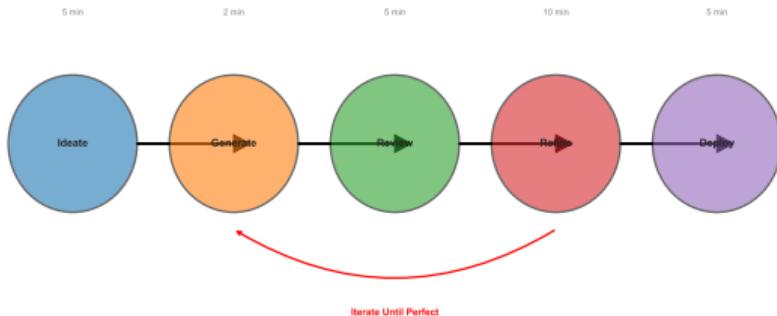
We'll explore:

- Prototyping workflows
- Design iteration with AI
- Human-AI collaboration
- Creative tools
- Case studies

Let's Design with AI!

# The AI-Enhanced Design Workflow

Rapid Prototyping Workflow with AI



**100x Faster Iteration**

**Traditional: 2 weeks**

- Sketch (2 days)
- Mockup (3 days)
- Prototype (5 days)
- Test (3 days)
- Refine (2 days)

**With AI: 2 days**

- Prompt (1 hour)
- Generate (minutes)
- Refine (2 hours)
- Test (4 hours)
- Iterate (1 day)

## Component Generation

- UI components from description
- Consistent styling
- Accessibility built-in
- Responsive by default
- Dark mode variants

## Tools:

- Figma AI
- Framer AI
- Galileo AI
- Uizard

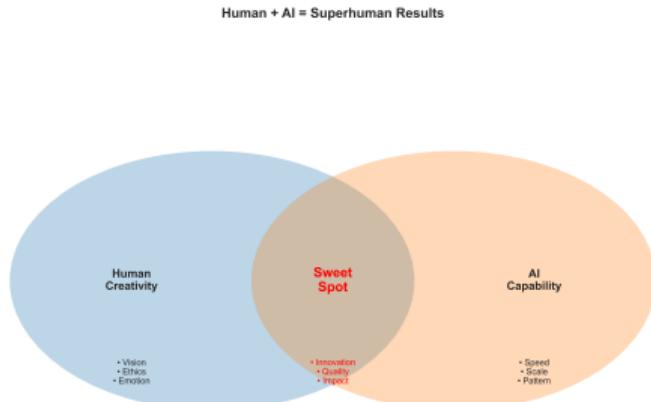
## Brand Consistency

- Style guide enforcement
- Color palette generation
- Typography systems
- Icon families
- Pattern libraries

## Benefits:

- 80% time savings
- Perfect consistency
- Instant variations
- Global updates

AI maintains design system integrity at scale



Human + AI = Superhuman Results

## Better Together

### Human Strengths:

- Vision & strategy
- Emotional intelligence
- Context understanding
- Ethics & values

### AI Strengths:

- Speed & scale
- Variation generation
- Pattern recognition
- Consistency

# Case Study: Spotify AI DJ

## The Challenge

Create personalized radio experience for 500M users

## AI Solution:

- Voice synthesis (natural DJ)
- Music recommendation
- Context awareness
- Real-time adaptation

## Results:

- 2x engagement
- 30% longer sessions
- 4.8 star rating

12 weeks from concept to 500M users

## Design Process

- ① User research (2 weeks)
- ② AI voice prototypes (1 week)
- ③ Music curation models (2 weeks)
- ④ Interface design (1 week)
- ⑤ A/B testing (2 weeks)
- ⑥ Global rollout (4 weeks)

## Key Learning:

AI enables hyper-personalization impossible with human DJs

# Creative AI Tools Landscape

Tool	Type	Strength	Cost/mo	Best For
Midjourney	Image	Art quality	\$10-120	Concepts
DALL-E 3	Image	Accuracy	\$20	Products
Figma AI	Design	Integration	\$15	UI/UX
RunwayML	Video	Effects	\$15-95	Motion
GitHub Copilot	Code	Context	\$10	Development
Claude	Text	Reasoning	\$20	Content
Eleven Labs	Voice	Quality	\$5-330	Audio

**Selection Criteria:** Quality × Speed × Control ÷ Cost = Value

Average designer uses 3-5 AI tools daily

## Traditional A/B Testing

- 2 variants manually created
- Weeks to get significance
- Designer bottleneck
- Limited exploration
- High cost per test

**Result:** Slow learning

AI enables testing at previously impossible scales

## AI-Enhanced Testing

- 100+ variants generated
- Multi-armed bandit optimization
- Automatic generation
- Exhaustive exploration
- Minimal marginal cost

**Result:** 10x faster insights

# Complete Workflow: Landing Page Creation

## Hour 1: Research & Brief

- ① Competitor analysis (15 min)
- ② User research synthesis (15 min)
- ③ Value prop definition (15 min)
- ④ Success metrics (15 min)

## Hour 2: Generation

- ① Headlines: 50 variants
- ② Hero images: 20 options
- ③ Copy blocks: 30 versions
- ④ CTA buttons: 10 styles

4 hours from concept to live A/B test

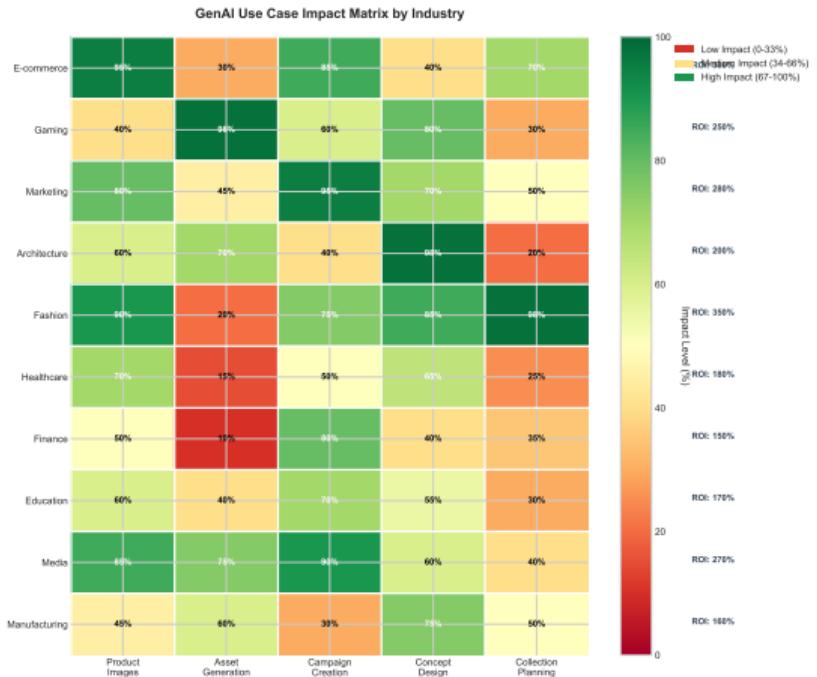
## Hour 3: Assembly

- ① Best combinations (30 min)
- ② Layout optimization (15 min)
- ③ Responsive design (15 min)

## Hour 4: Testing

- ① Deploy 5 variants
- ② Set up analytics
- ③ Launch traffic split
- ④ Monitor performance

# Industry Applications of GenAI Design



GenAI design adoption growing 200% year-over-year

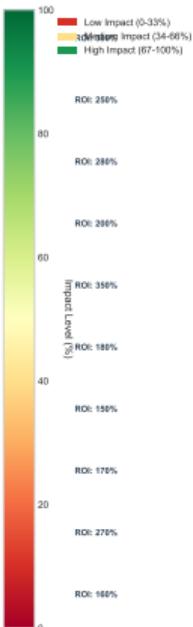
## Every Industry

### Top Adopters:

- E-commerce: Product images
- Gaming: Asset generation
- Marketing: Campaign creation
- Architecture: Concept design
- Fashion: Collection planning

### ROI Leaders:

- 300% productivity gain
- 80% cost reduction
- 10x iteration speed



## System Components

- **Token Library:** Auto-generated variables
- **Component Factory:** Variant creation
- **Pattern Engine:** Layout templates
- **Brand Guardian:** Consistency checks
- **A11y Assistant:** Accessibility fixes

## Automation Level:

- 95% component generation
- 100% documentation
- 80% QA coverage

## Real Examples

- Airbnb:** AI generates localized UI  
**Uber:** Dynamic design tokens  
**Netflix:** Personalized layouts  
**Amazon:** Adaptive components

## Benefits:

- Zero design debt
- Instant scaling
- Perfect consistency
- Automatic versioning
- Self-documenting

Design systems become living, evolving organisms with AI

# Case Study: Duolingo's AI Characters

## The Challenge

Create engaging characters for 40+ languages with cultural sensitivity

## AI Solution:

- Character generation pipeline
- Cultural validation models
- Voice synthesis (11 Labs)
- Animation automation
- Personality engines

## Results:

- 400+ characters created
- 87% engagement increase
- 4.9 star rating
- \$500M valuation boost

AI reduced character development from 6 months to 4 weeks

## Technical Stack

- ➊ **Ideation:** GPT-4 for personalities
- ➋ **Visuals:** Stable Diffusion XL
- ➌ **Voice:** Custom TTS models
- ➍ **Animation:** Motion diffusion
- ➎ **QA:** Automated testing

## Timeline:

- Week 1: Concept & testing
- Week 2: Production pipeline
- Week 3: Localization
- Week 4: Global launch

# The Future: Autonomous Design Agents

## 2024-2025

- AI assistants (current)
- Prompt-based generation
- Human review required
- Tool integration

## 2025-2026

- Semi-autonomous agents
- Goal-based creation
- Minimal supervision
- Cross-tool orchestration

## 2026-2027

- Fully autonomous design
- Business metric optimization
- Self-improving systems
- Human creative directors only

### Implications:

- Designer role evolution
- 1000x productivity
- Real-time personalization
- Infinite variations

The future designer is a conductor, not a performer

## Key Insights

- ① AI accelerates ideation
- ② Human creativity essential
- ③ Tools complement skills
- ④ Iteration is free
- ⑤ Quality requires curation

## Next: Practice

Hands-on exercises:

- Build a product concept
- Generate variations
- Optimize prompts
- Compare tools
- Deploy solution

Time to Create!

## Build in 30 Minutes

### Your Mission:

- ① Choose a problem domain
- ② Generate product concept
- ③ Create visual mockups
- ④ Write product description
- ⑤ Generate marketing copy
- ⑥ Present to class

### Available Tools:

- ChatGPT/Claude (concept)
- DALL-E/Midjourney (visuals)
- Figma AI (mockups)
- Copy.ai (marketing)
- Canva (presentation)

### Deliverables:

- 3 concept images
- 1-page description
- 5-slide pitch deck

Real products have been launched from exercises like this

# Step-by-Step: Prompt Engineering

## Step 1: Ideation Prompt

Generate 10 innovative product ideas that solve [problem] for [audience].

For each idea provide:

- Name
- One-line description
- Key differentiator
- Target market size

## Step 2: Refinement

Take idea #3 and expand:

- Problem it solves (specific)
- Solution approach
- 5 key features
- Revenue model
- Competitive advantage

## Step 3: Visual Generation

Create a product mockup:  
"Minimalist app interface for [product], clean design, modern UI, mobile screen, professional, Figma style"

## Step 4: Copy Creation

Write landing page copy:

- Hero headline (7 words)
- Subheadline (15 words)
- 3 benefit statements
- Call to action

Target emotion: excitement

Good prompts = Good outputs

## Mistakes to Avoid

- Vague prompts → Vague outputs
- No iteration → Mediocre results
- Ignoring constraints → Unusable
- Over-relying on AI → Generic
- No human review → Errors

## Quality Checklist:

- Is it original?
- Does it solve a real problem?
- Is it feasible?
- Would you use it?

## Pro Tips

- Start with 10 variations
- Combine best elements
- Add specific constraints
- Use reference examples
- Iterate at least 3 times

## Time Allocation:

- Ideation: 5 min
- Generation: 10 min
- Refinement: 10 min
- Polish: 5 min

The difference between good and great is iteration

## Ethical Guidelines

- Always disclose AI use
- Verify all facts
- Check for bias
- Respect IP rights
- Maintain human oversight

## Legal Considerations:

- Copyright unclear
- Terms of service vary
- Attribution required
- Commercial use restrictions

## Best Practices

- Credit: "Created with AI assistance"
- Document your process
- Keep human in the loop
- Test with real users
- Monitor for harmful content

## Remember:

AI is a tool, not a replacement for human judgment

With great power comes great responsibility

## Free Resources

- Hugging Face (models)
- Google Colab (compute)
- Papers with Code
- Fast.ai courses
- YouTube tutorials

## Communities:

- r/LocalLLaMA
- Discord: AI servers
- GitHub: Awesome lists
- Twitter: AI researchers

The field moves fast - continuous learning is essential

## Stay Current

- The Batch (newsletter)
- AI News (aggregator)
- ArXiv (papers)
- Product Hunt (tools)
- LinkedIn Learning

## Next Week:

Explainable AI - Understanding what the model learned

## What We Learned

- GenAI transforms prototyping
- Multiple algorithms available
- APIs enable quick starts
- Prompting is crucial
- Human+AI & either alone

## Your Homework

- Complete product concept
- Try 3 different AI tools
- Generate 20 variations
- Document your process
- Share with class

Go Create Something Amazing!