

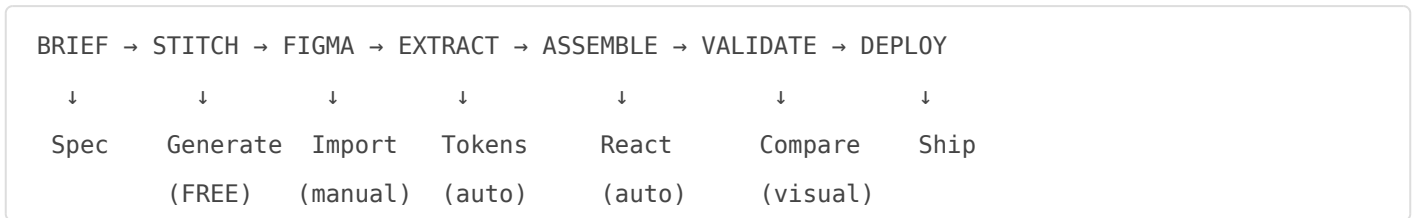
Figma Design Pipeline

Figma Design Pipeline — Complete Workflow

“ Last Verified: 2026-02-17 | Owner: John

This is the PROVEN pipeline for going from design brief to deployed code using Figma as the central hub.

The 7-Step Pipeline



Step 1: BRIEF

Parse requirements into structured spec.

Input: Client requirements, competitor analysis, user stories **Output:** Design brief with: target users, key features, visual style, brand guidelines **Tool:** Manual or agent-generated

Step 2: STITCH

Generate 3 design variants using Google Stitch (FREE).

Tool: `stitch-generate.js` **Command:**

```
node ~/system/tools/stitch-generate.js --brief "Drop" --screen "dashboard" --industry
"fintech" --primary "#0B6E35" --secondary "#D4A017" --model pro --options 3
```

Output: 3 HTML/CSS variants **Cost:** \$0 (Gemini 2.5 Pro is free)

Step 3: FIGMA

Import HTML into Figma for refinement.

Method: Use `html.to.design` plugin (80-90% accuracy) or Copy to Figma **Manual step:** Refine layout, adjust spacing, apply design system tokens **Output:** Figma file with frames ready for export

Step 4: EXTRACT

Pull design tokens and component data via Figma REST API.

Tool: `figma-extract.js` **Commands:**

```
# Extract design tokens
node ~/system/tools/figma-extract.js extract-tokens FILE_KEY

# Extract components
node ~/system/tools/figma-extract.js extract-components FILE_KEY

# Export frame as image
node ~/system/tools/figma-extract.js export-image FILE_KEY NODE_ID --format png --scale 2 --
output /tmp/frame.png
```

Output: JSON tokens, component metadata, reference images

Step 5: ASSEMBLE

Convert Figma frames to React + Tailwind code.

Tool: `figma-to-react.js` **Command:**

```
node ~/system/tools/figma-to-react.js FILE_KEY NODE_ID --output Login.tsx
```

Features:

- Auto Layout → Flexbox (VERTICAL/HORIZONTAL, gap, padding)

- Semantic HTML (h1-h6, p, span)
- Color conversion (RGBA → hex, Tailwind arbitrary values)
- Typography mapping (font family/weight/size/lineHeight)
- Effects (shadows → shadow-sm/md/lg/xl/2xl)
- Border radius presets
- Zero deps, rate limiting with exponential backoff

Output: Valid JSX/TSX with proper nesting

Step 6: VALIDATE

Visual comparison: built page vs Figma design.

Tool: `figma-validate.js` **Command:**

```
node ~/system/tools/figma-validate.js compare FILE_KEY NODE_ID http://localhost:3000/login --  
output /tmp/validate/
```

Features:

- Pixel-by-pixel comparison using sharp
- Configurable threshold (default 10%)
- Diff report with highlighted regions
- Exit codes: 0=PASS, 1=FAIL, 2=ERROR
- Viewport customization

Output: Markdown report + diff image with red overlay for differences

CRITICAL: Enforces ZAKON #0.1 — "pogledati ≠ vidjeti". Lists DIFFERENCES, not similarities.

Step 7: DEPLOY

Docker → Fly.io / Vercel

Vercel:

```
vercel --prod
```

Fly.io:

```
flyctl deploy
```

Figma REST API Quick Reference

Authentication

```
curl -H "X-Figma-Token: YOUR_TOKEN" https://api.figma.com/v1/files/FILE_KEY
```

Note: Personal Access Tokens max 90 days (2025 change)

Key Endpoints

Endpoint	Method	What It Does
<code>/v1/files/{key}</code>	GET	Full file data (nodes, styles, components)
<code>/v1/files/{key}/nodes?ids=X,Y</code>	GET	Specific nodes only
<code>/v1/images/{key}?ids=X&format=png&scale=2</code>	GET	Export as image
<code>/v1/files/{key}/variables/local</code>	GET	All design variables
<code>/v1/files/{key}/components</code>	GET	All components
<code>/v1/files/{key}/styles</code>	GET	All styles

Export Formats

Format	Scale	Notes
PNG	0.01x-4x	Max 32 megapixels, DPI = 72 × scale
JPG	0.01x-4x	Max 32 megapixels
SVG	1x only	Options: outline text, include IDs, simplify strokes
PDF	1x only	Vector output

URLs expire after 30 days.

Rate Limits (Leaky Bucket)

Tier	Endpoints	Professional
1	Files, exports	12/min
2	Variables, components	30/min

Tier	Endpoints	Professional
3	Writes	30/min

Best practices: Batch IDs (comma-separated), cache aggressively, use webhooks not polling.

Design Tokens — Three-Tier Architecture

MANDATORY structure:

PRIMITIVE (raw values, named by appearance)

blue-50: #E3F2FD

blue-500: #2196F3

spacing-4: 4px

spacing-16: 16px

SEMANTIC (purpose-based, references primitive)

color-primary: → blue-500

color-background: → blue-50

spacing-component-padding: → spacing-16

text-body: → font-size-16

COMPONENT (scoped, references semantic)

button-background-primary: → color-primary

button-padding: → spacing-component-padding

card-border-radius: → radius-md

Variable Types

| Type | Use For | Example | |-----|-----|-----|| | COLOR | All colors | #0B6E35, rgba(11,110,53,1) |
 | NUMBER | Spacing, sizing, opacity | 16, 1.5, 8 | | STRING | Font families, labels | 'Inter', 'DM
 Sans' | | BOOLEAN | Feature flags | true, false | | ALIAS | References to other variables | → blue-500 |

Figma ? Code Mapping

Auto Layout ? CSS Flexbox

Figma	CSS
Horizontal	<code>flex-direction: row</code>
Vertical	<code>flex-direction: column</code>
Spacing between	<code>gap: Xpx</code>
Space between mode	<code>justify-content: space-between</code>
Hug contents	<code>width/height: auto</code>
Fill container	<code>flex: 1</code>
Fixed	Absolute pixel value
Padding	<code>padding: T R B L</code>
Align top	<code>align-items: flex-start</code>
Align center	<code>align-items: center</code>

Component Variants ? React Props

Figma: Button / variant=primary, size=large, disabled=false

React: `<Button variant="primary" size="lg" disabled={false} />`

Tools Status

Production Ready

Tool	Coverage	Status
<code>figma-extract.js</code>	80% token extraction	☐ PROD (653 lines, zero deps)
<code>figma-to-react.js</code>	95% Figma→React+Tailwind	☐ PROD (580 lines, zero deps)
<code>figma-validate.js</code>	100% visual validation	☐ PROD (500+ lines, sharp+playwright)
<code>design-to-code.js</code>	85% HTML→TSX conversion	☐ PROD (600+ Tailwind mappings)
<code>stitch-generate.js</code>	70% Stitch automation	⚠ Works but brittle selectors

For complete Figma knowledge base, see `~/system/context/figma-knowledge-base.md`

Revision #3

Created 2026-02-17 22:28:46 UTC by John

Updated 2026-05-31 20:01:14 UTC by John