



Rico System V5 - Prompts Übersicht

Ziel

Einheitliche Struktur für Master- und Einstiegs-Prompts im Rico-System mit automatischer PDF-Erstellung und Auto-Load-Routine für alle KIs.

Ablagestruktur

Master-Prompts

- `Rico_MasterPrompt_Arbeitsbuch_V5.md` - Zentrale Masterdatei (deutsche Version)
- `Rico_MasterPrompt_Arbeitsbuch_V5.pdf` - Automatisch generierte PDF-Version

KI-spezifische Einstiegsprompts

- `Claude_Einstiegsprompt.md` - Strategischer Analytiker, Texter, Review-Spezialist
- `Perplexity_Einstiegsprompt.md` - Research-Agent für Fakten, Quellen, Daten
- `Cursor_Einstiegsprompt.md` - Technischer Builder für Code, Dateien, Automatisierung

Konfiguration

- `prompt_registry.json` - Zuordnung von Rollen → Dateien
 - `load_prompts.py` - Auto-Load-Routine für automatisches Laden beim Start
-

Auto-Load-Routine

Verwendung

```
# Alle Prompts laden
python load_prompts.py

# Spezifischen Prompt laden
python -c "from load_prompts import get_prompt_for_ki; print(get_prompt_
```

Output

```
 Rico System V5 - Auto-Load-Routine gestartet
=====
📄 Version: V5
📅 Letzte Aktualisierung: 2024-09-24
🎨 Theme: #009688

✅ Master-Prompt geladen: Rico_MasterPrompt_Arbeitsbuch_V5.md
✅ Claude-Einstiegsprompt geladen
✅ Perplexity-Einstiegsprompt geladen
✅ Cursor-Einstiegsprompt geladen

=====
🇩🇪 Zusammenfassung: 4/4 Prompts erfolgreich geladen
🎉 Alle Prompts erfolgreich geladen - Rico System V5 bereit!
```

Ablageregeln

✅ Erlaubt

- **Alle Prompts** in `docs/` ablegen
- **Automatische PDF-Erstellung** in `docs/pdf/`
- **Einheitliches Layout** (Smaragdgrün + Gold, A4, Helvetica)
- **Versionierung** über `prompt_registry.json`

✗ Verboten

- Prompts außerhalb von `docs/` ablegen
 - Manuelle PDF-Erstellung (nur über Pipeline)
 - Abweichende Layouts oder Themes
 - Direkte Änderungen an PDFs (nur Markdown-Quellen bearbeiten)
-

🔧 Update-Regeln

Master-Prompt aktualisieren

1. **NUR** `Rico_MasterPrompt_Arbeitsbuch_V5.md` bearbeiten
2. **NICHT** andere Versionen verwenden
3. Nach Änderung: `npm run pdf:all` ausführen
4. Auto-Load-Routine testen: `python load_prompts.py`

KI-Prompts aktualisieren

1. Entsprechende `.md`-Datei in `docs/` bearbeiten
 2. PDF-Pipeline ausführen: `npm run pdf:all`
 3. Registry prüfen: `docs/prompt_registry.json`
 4. Auto-Load testen
-

🎨 Design-Standards

PDF-Layout

- **Format:** A4, drucktauglich
- **Primärfarbe:** Smaragdgrün (#009688) für Überschriften
- **Akzentfarbe:** Gold (#D4AF37) für Highlights
- **Schriftart:** Helvetica (Sans Serif)
- **Ränder:** Optimiert für Druck

Markdown-Struktur

- **Überschriften:** Klare Hierarchie (H1-H6)
 - **Listen:** Einheitliche Formatierung
 - **Code:** Monospace mit Hintergrund
 - **Links:** Smaragdgrün, unterstrichen bei Hover
-



Registry-Struktur

```
{
  "master": "docs/Rico_MasterPrompt_Arbeitsbuch_V5.md",
  "claude": "docs/Claude_Einstiegsprompt.md",
  "perplexity": "docs/Perplexity_Einstiegsprompt.md",
  "cursor": "docs/Cursor_Einstiegsprompt.md",
  "metadata": {
    "version": "V5",
    "last_updated": "2024-09-24",
    "description": "Rico System V5 - Prompt Registry",
    "pdf_output": "docs/pdf/",
    "theme": {
      "primary_color": "#009688",
      "accent_color": "#D4AF37",
      "font_family": "Helvetica"
    }
  }
}
```



Wichtige Hinweise



Master-Prompt

- **NUR** `Rico_MasterPrompt_Arbeitsbuch_V5.md` bearbeiten
- **NICHT** ältere Versionen verwenden
- **IMMER** nach Änderungen PDF-Pipeline ausführen

⚠️ PDF-Erstellung

- **Automatisch** über `npm run pdf:all`
- **NICHT** manuell PDFs bearbeiten
- **IMMER** Markdown-Quellen als Basis verwenden

⚠️ Auto-Load

- **Beim Start** immer `python load_prompts.py` ausführen
 - **Registry** muss aktuell sein
 - **Alle Dateien** müssen in `docs/` vorhanden sein
-

🎉 Rico System V5 - Prompts erfolgreich strukturiert!

Alle Prompts sind jetzt:

- ☒ Einheitlich in `docs/` abgelegt
- ☒ Automatisch als PDFs in `docs/pdf/` verfügbar
- ☒ Über Auto-Load-Routine beim Start verfügbar
- ☒ Mit einheitlichem Design (Smaragdgrün + Gold)
- ☒ Vollständig dokumentiert und versioniert