Identity, Ethics, and Cooperation in the Age of AI

The Adlerian Software Engineer and the Adlerian Classroom

Marco Vieira

marco.vieira@charlotte.edu

UNIVERSITY OF NORTH CAROLINA CHARLOTTE

THE GAP

A Literature Missing Its Center

What the literature addresses

- **Productivity**: outputs, speed, cost; AI as force multiplier
- **Workflow**: tool adoption, developer experience, pipeline integration
- **Deskilling *vs.* Displacement**: both views are predominantly economic

What the literature leaves unstudied:

Identity: what does it mean to be an engineer when AI writes code?

Motivation: what should engineers strive toward?

Ethics: what do engineers owe the communities they serve?

These dimensions are not peripheral to the AI transition:

They are central to it!

ADLERIAN LENS

Seven Adlerian Principles

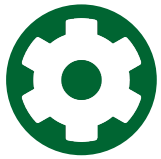
The Individual Psychology of Alfred Adler (1870–1937)

Inferiority & Striving

All humans begin in relative helplessness; this generates a compensatory drive toward mastery and contribution

Goal-Directed Behavior

Behavior is pulled by guiding goals (fictional finals), not pushed by past causes



Gemeinschaftsgefühl

The felt sense of belonging to and responsibility toward one's community — psychological health

Unity and Holism

The person is understood as a unified whole: thoughts, emotions, and behaviors are expressions of a coherent system

Fictional Finalism

Guiding fictions organize a pattern of thought, feeling, and action toward a self-defined ideal of completion

Creative Self

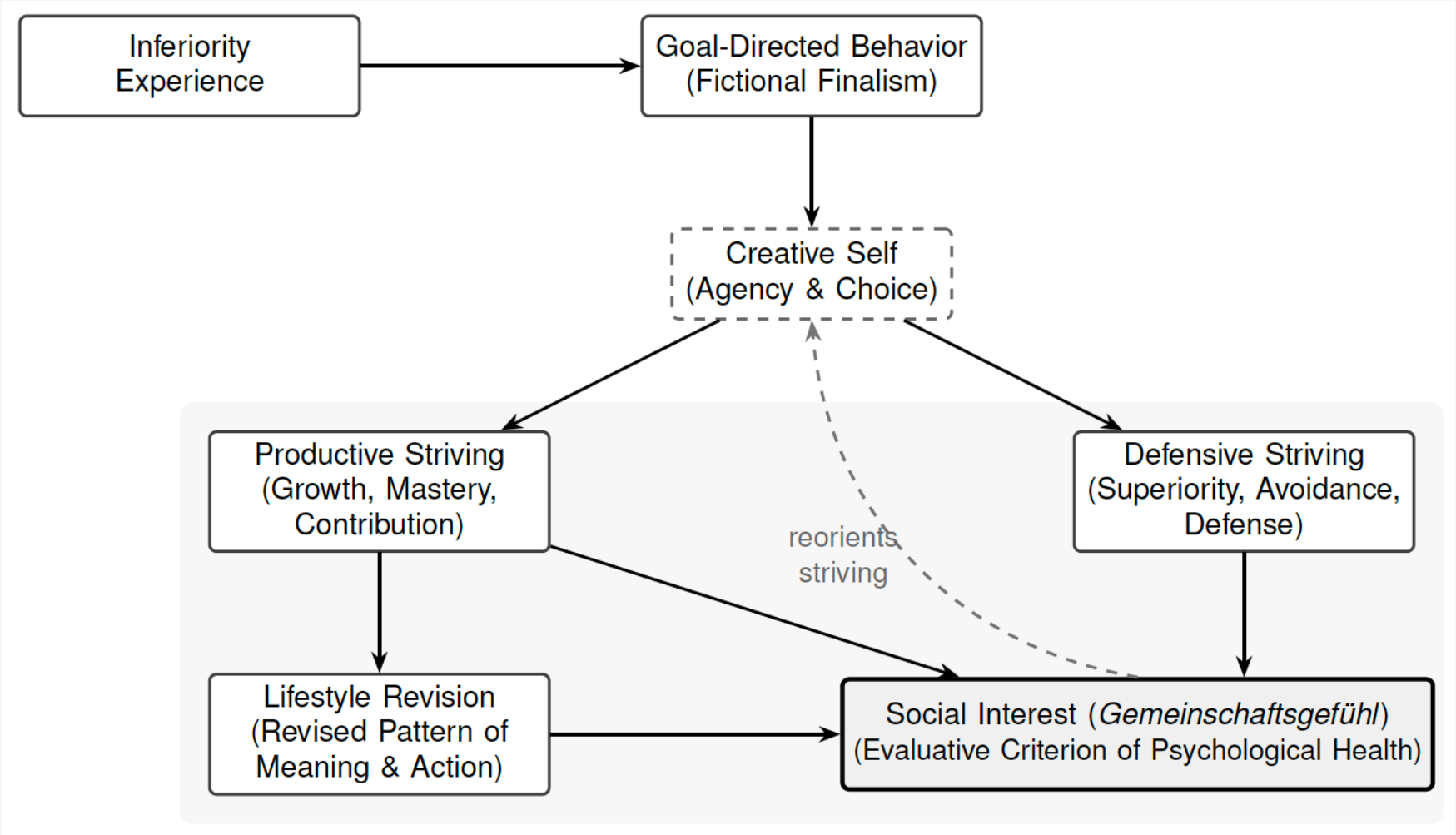
Active agents who interpret circumstances and choose responses — not determined by heredity or environment

Lifestyle

A stable pattern of perception and action developed early in life; revisable, but only through deliberate effort

MOTIVATIONAL MODEL

How Inferiority Drives Professional Development



Four Adlerian Responses to AI Capability

AI generates an inferiority experience. The Adlerian question is: what do you do with it?

Dismissal

"AI is overrated"
Depreciating the tool to restore a sense of superiority – neurotic in the Adlerian sense

Learned Helplessness

The belief that human skill is now obsolete, leading to burnout and disengagement – this is resignation, not striving

Overcorrection

Outsourcing of judgment to the machine – an abandonment of striving: deskilling made permanent

Productive Striving:

- Use inferiority as a signal for growth, not a verdict on worth
- Strive toward higher-order mastery: problem framing, architectural judgment, ethical discernment, and critical evaluation of AI output

The Same Principles, Two Contexts

Adler's framework speaks equally to the practicing engineer and to the educator: the psychological dynamics are the same

The Engineer

- **Inferiority** → productive striving, not dismissal or overcorrection
- **Gemeinschaftsgefühl** → build for community, not personal prestige
- **Lifestyle revision** → toward higher-order mastery, not just new skills

The Educator

- **Inferiority** → structured learning, not avoidance or AI delegation
- **Social interest** → cooperative practice, not solitary AI-assisted work
- **Assessment** → reveal direction of striving, not just measure output

In both contexts, the central question is the same:

- Is what I am building – or teaching – driven by genuine contribution, or by compensation for insecurity?
- The graduate course is where both converge – engineers and students confronting the same Adlerian dynamics in practice

EVIDENCE FROM PRACTICE

AI-Driven Trustworthy Software Development

UNC Charlotte, Spring 2026: 8 graduate students, 3 teams, AI-generated > 90% of code. The Adlerian framework was not by design, but its predictions held.

The Honesty Defect

- Code whose names, comments, and structure claimed to implement three features
- The actual implementation performed no meaningful computation
- Code looked complete and internally consistent: it simply did not do what it said it did

What Students Reported

- 7 of 8: trust in AI-generated code became more nuanced, not simply lower
- 7 of 8: rated “this course changed my perspective on validating software” a 5 out of 5

The Student Insight

- “Before you write any code, write down what you’re assuming the AI will get right — then treat every assumption as a hypothesis that needs a test.”
- This is lifestyle revision in the precise Adlerian sense — not skill acquisition, but a restructuring of professional identity.

FOR DISCUSSION

A Provocation for WG 10.4

We have frameworks for what dependable systems must do. We lack frameworks for what dependable engineers must **become.**

If the engineer cannot evaluate AI output critically, govern it responsibly, or ask “should we build this, and for whom?” — the technical framework is insufficient.

How do we form engineers
who govern AI rather than merely use it — and who take responsibility for what they build?

Does psychological safety
belong in dependability curricula and team design as a precondition for productive striving?

What does Gemeinschaftsgefühl
look like in a safety-critical engineering team — and how do we cultivate it?

The profession that finds credible answers will not merely survive the AI transition — it will be improved by it!

Thank you!

Marco Vieira

marco.vieira@charlotte.edu

https://marcovieira.me

The Adlerian Software Engineer

The Adlerian Classroom

- AI triggers inferiority: the response is a choice, not a fate
- Higher-order mastery • Gemeinschaftsgefühl • Cooperation over competition
- The question is not “how can I protect what I have?” but “how can I contribute more fully?”