



ICML 2026

FullStack-Agent: Enhancing Agentic Full-Stack Web Coding via Development-Oriented Testing and Repository Back-Translation

Zimu Lu^{*1} Houxing Ren^{*1} Yunqiao Yang¹ Ke Wang¹ Zhuofan Zong¹ Mingjie Zhan^{†12} Hongsheng Li^{†134}



香港中文大學
The Chinese University of Hong Kong



Problem: Enhancing Agentic Full-Stack Web Coding

From frontend demos to real web paps

Many agents look interactive, but lack:

- backend logic
- persistent storage
- real data flow

Goal: generate production-like full-stack websites



Main Idea

FullStack-Agent = 3-part system (Build, improve, and verify full-stack coding agents)

FullStack-Dev

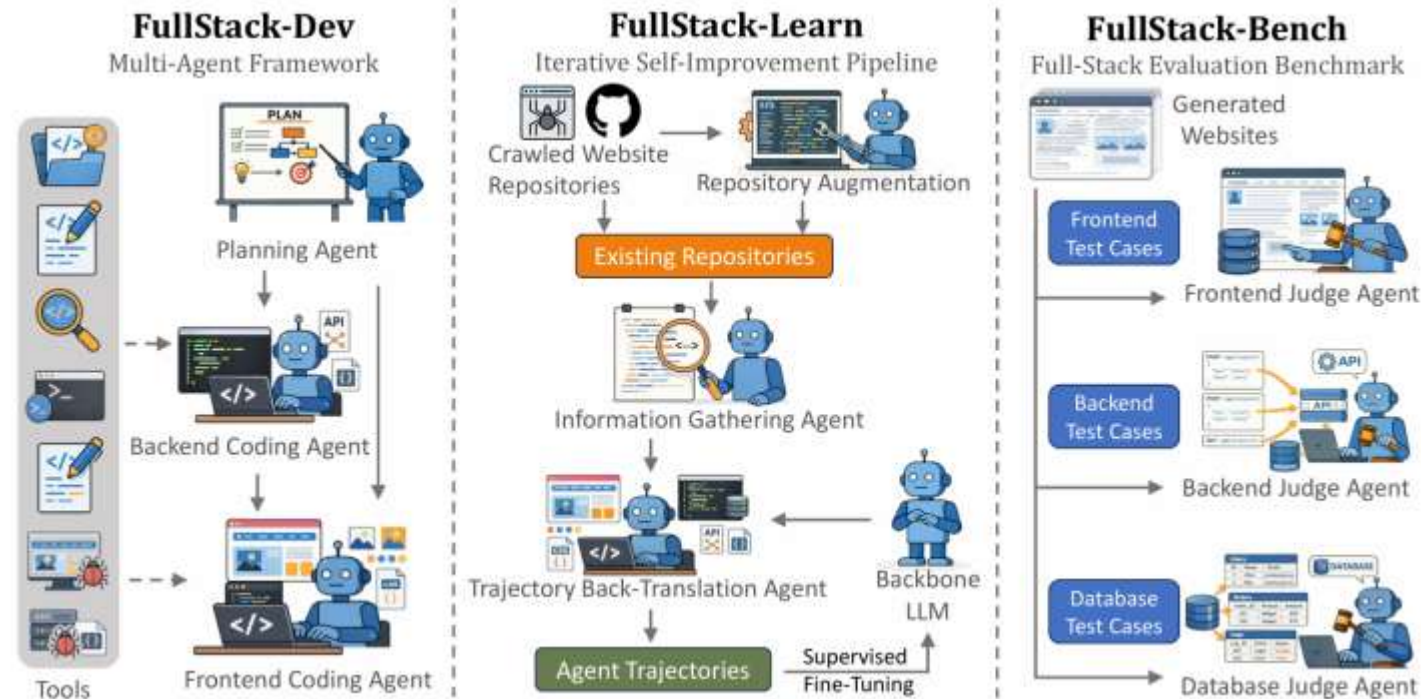
multi-agent development framework

FullStack-Learn

self-improving training pipeline

FullStack-Bench

full-stack evaluation benchmark



FullStack-Dev

FullStack-Dev workflow

- **Planning Agent**

- creates frontend and backend plans

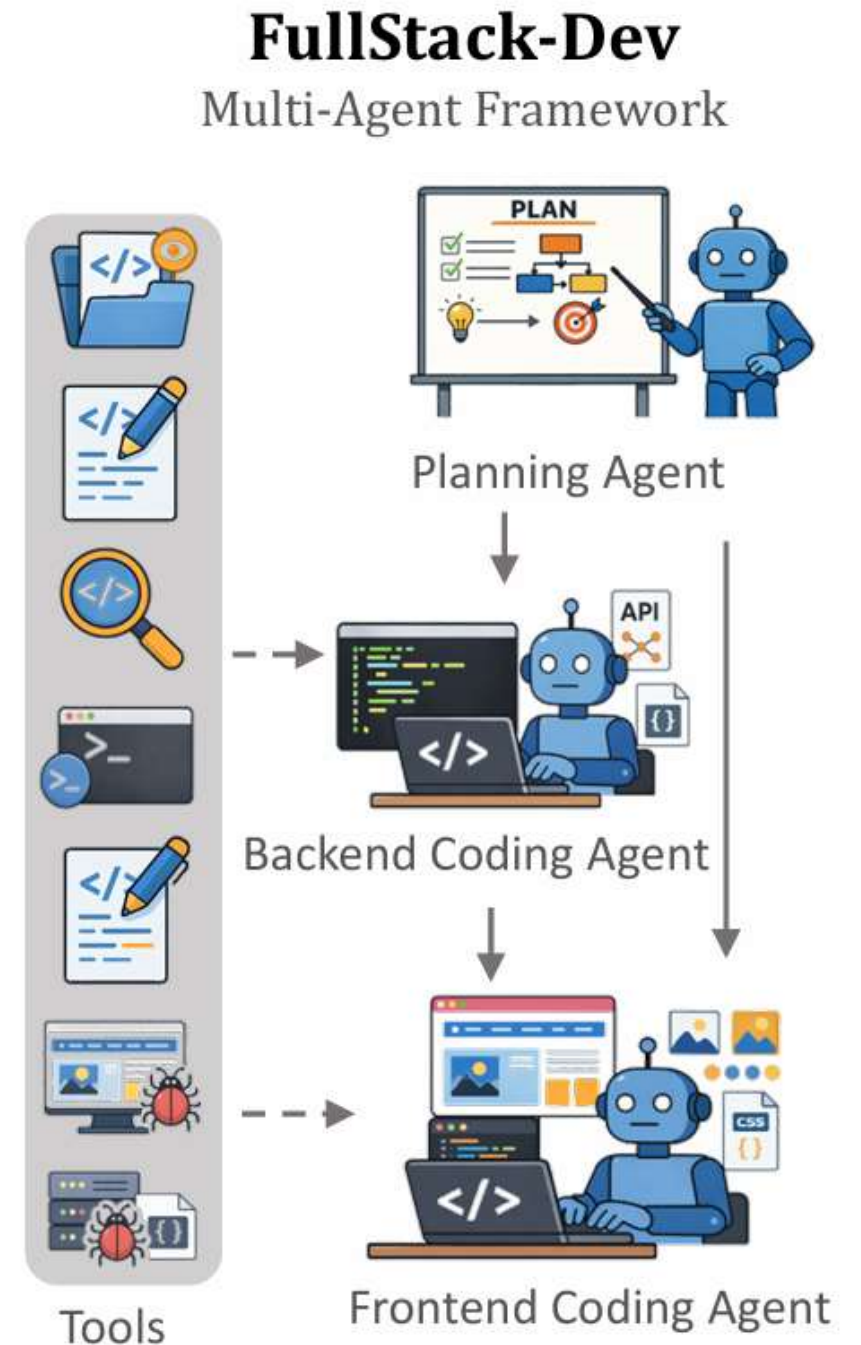
- **Backend Coding Agent**

- builds APIs and database logic first

- **Frontend Coding Agent**

- implements UI on top of real APIs

Design principle: mirror real software roles



Key Technical Strength: Debugging Tools

Specialized debugging tools

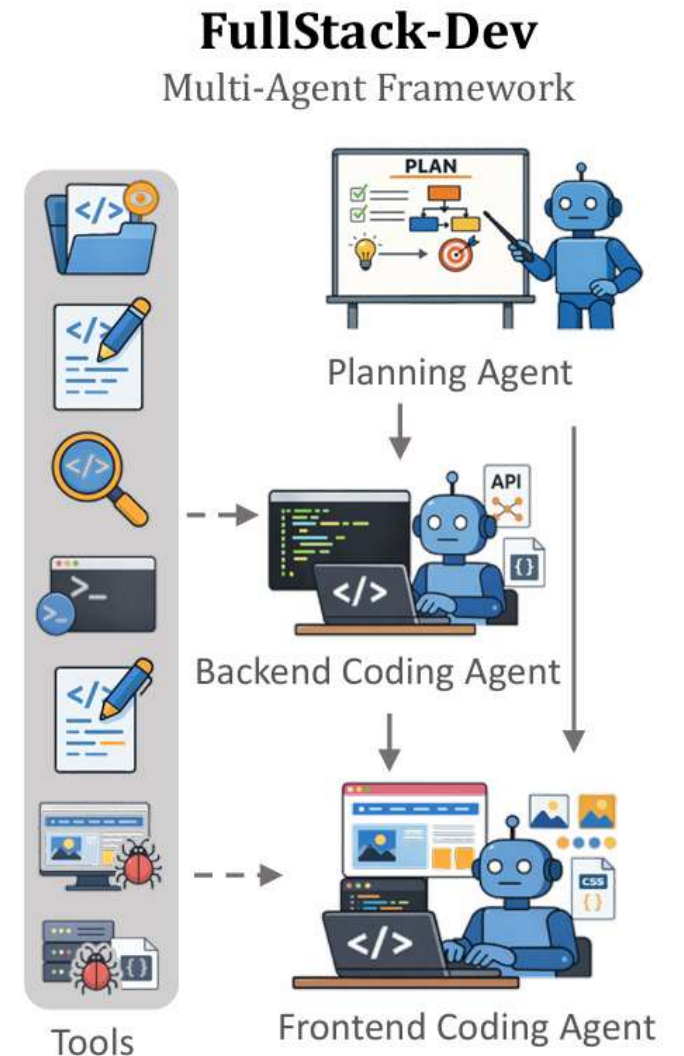
Frontend debugger

- GUI-based testing
- monitors browser + terminal errors
- links failures to user actions

Backend debugger

- API request testing
- returns response + console logs

Enables faster, more precise iteration

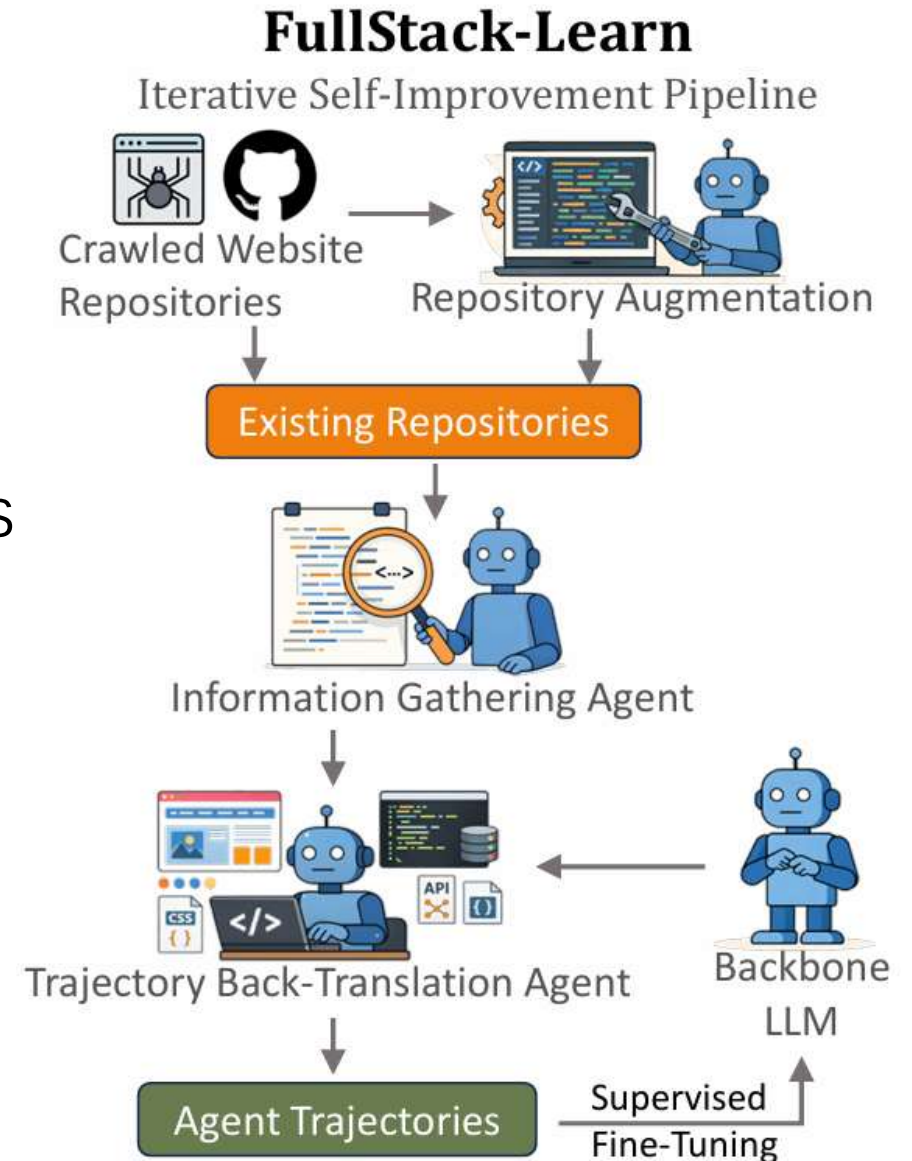


FullStack-Learn

FullStack-Learn: model self-improvement

- Start from real web repositories
- **Back-translate** repos into coding trajectories
- **Augment** repos to scale data:
 - simplify
 - extend
 - parallel apps
- Fine-tune the model iteratively

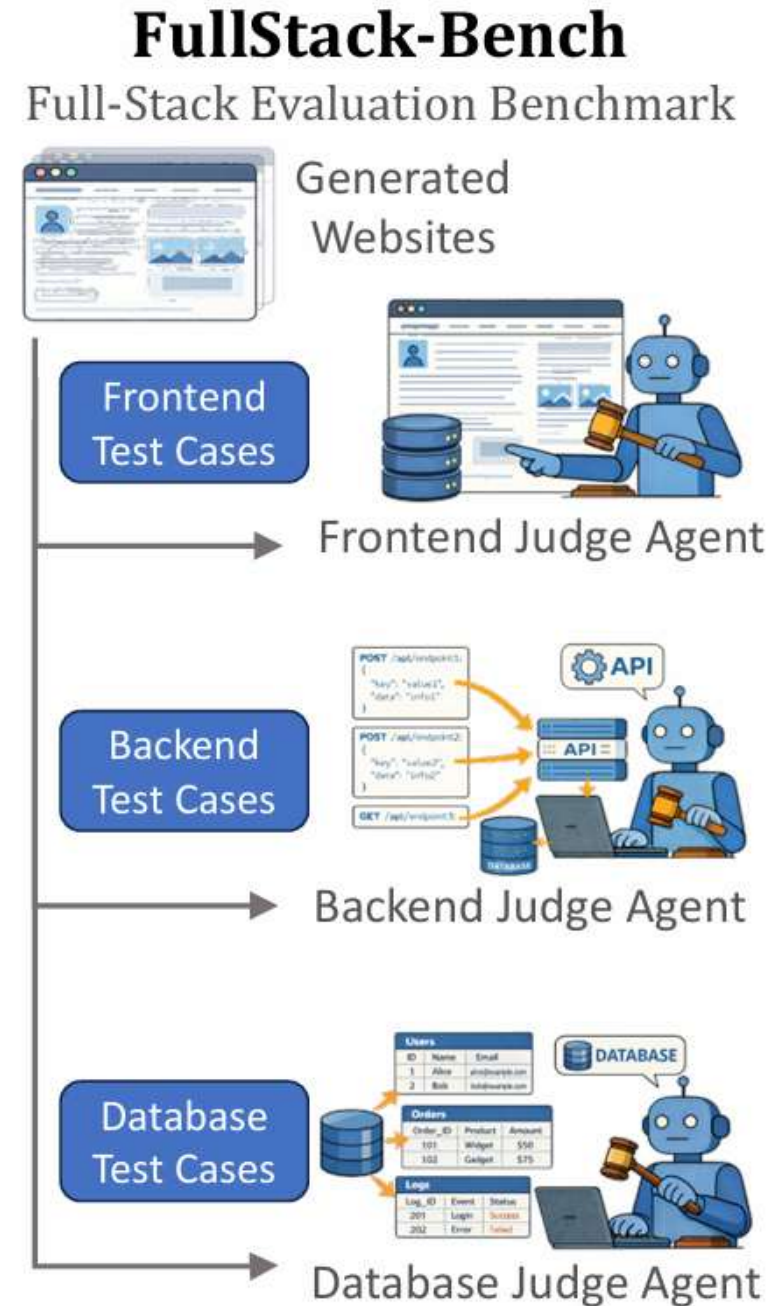
Outcome: stronger full-stack coding ability



FullStack-Bench

FullStack-Bench

- **101** user instructions
- **647** frontend tests
- **604** backend tests
- **389** database tests



Main Results

Best FullStack-Dev results

- Frontend: **69.6%**
- Backend: **89.7%**
- Database: **79.2%**

Improvement over prior SOTA

- **+25.2** frontend
- **+15.7** backend
- **+4.6** database

Framework	Model	FE Acc.	FE Acc. w/ Valid DB	BE Acc.	BE Acc. w/ Valid DB	DB Acc.	Appearance Score
WebGen-Agent	Qwen3-Coder-480B-A35B-Inst.	<u>61.4</u>	<u>56.0</u>	<u>67.2</u>	<u>39.6</u>	<u>62.0</u>	<u>3.63</u>
TDDev	Qwen3-Coder-480B-A35B-Inst.	23.3	21.8	41.2	10.8	28.5	1.96
OpenHands	Qwen3-Coder-480B-A35B-Inst.	45.1	40.4	64.9	35.8	72.5	2.66
Bolt.diy	Qwen3-Coder-480B-A35B-Inst.	22.8	20.3	35.9	12.7	13.6	1.96
Qwen-Code	Qwen3-Coder-480B-A35B-Inst.	39.9	36.2	59.3	25.8	68.4	2.40
FullStack-Dev	Qwen3-Coder-480B-A35B-Inst.	67.8	64.7	83.4	77.8	77.9	3.72
WebGen-Agent	Qwen3-Coder-30B-A3B-Inst.	<u>39.5</u>	<u>35.8</u>	25.0	16.6	14.4	<u>2.49</u>
TDDev	Qwen3-Coder-30B-A3B-Inst.	13.7	12.5	13.4	8.8	6.9	1.21
OpenHands	Qwen3-Coder-30B-A3B-Inst.	21.5	19.4	<u>26.7</u>	<u>16.7</u>	<u>22.6</u>	1.62
Bolt.diy	Qwen3-Coder-30B-A3B-Inst.	5.9	5.0	5.6	3.6	13.1	0.46
Qwen-Code	Qwen3-Coder-30B-A3B-Inst.	7.0	6.0	20.9	9.9	18.3	0.75
FullStack-Dev	Qwen3-Coder-30B-A3B-Inst.	41.4	37.2	45.5	38.7	50.9	2.97
WebGen-Agent	DeepSeek-V3	<u>48.8</u>	<u>44.4</u>	64.2	59.6	71.7	3.33
TDDev	DeepSeek-V3	41.3	39.3	61.3	57.0	72.5	<u>2.28</u>
OpenHands	DeepSeek-V3	21.8	19.3	<u>76.5</u>	<u>74.0</u>	<u>74.6</u>	1.69
Bolt.diy	DeepSeek-V3	25.4	23.0	62.6	32.1	65.3	2.11
Qwen-Code	DeepSeek-V3	22.1	20.1	65.9	64.6	70.7	1.58
FullStack-Dev	DeepSeek-V3	73.6	69.6	91.1	89.7	79.2	4.02

FullStack-Learn training results

FullStack-Bench results of two rounds of training on Qwen3-Coder-30B-A3B-Instruct using FullStack-Learn.

Test Name	Data	FE Acc.	FE Acc. w/ Valid DB	BE Acc.	BE Acc. w/ Valid DB	DB Acc.	Appearance Score
Qwen3-Coder-30B-A3B-Inst.	–	41.4	37.2	45.5	38.7	50.9	2.97
FullStack-Learn-LM-round1	Crawled 2K	49.3	42.3 ^{+5.1}	55.6	45.4 ^{+6.7}	51.2 ^{+0.3}	3.32 ^{+0.35}
FullStack-Learn-LM-round2	Crawled 2K + Agemented 8K	52.5	46.9 ^{+9.7}	57.6	48.2 ^{+9.5}	53.7 ^{+2.8}	3.40 ^{+0.43}

Thank You

Code: <https://github.com/mnluzimu/FullStack-Agent>

Data & Models: <https://huggingface.co/collections/luzimu/fullstack-agent>