2531cc1046

Addressing the Cognitive Gap in Software Development: Developer Tools for Keeping Human Understanding Up to Speed with AI Coding Agents

EPSRC theme	Information and Communication Technologies
Principal supervisor name	Federica Sarro
Principal supervisor UCL profile website	https://profiles.ucl.ac.uk/38657-federica-sarro
Host faculty and host department	Engineering Sciences Computer Science
Lead department website	https://www.ucl.ac.uk/engineering/computer-science
Department PGR administration email	cs.researchdegrees@ucl.ac.uk; cs.phdadmissions@ucl.ac.uk
Standard or collaborative project	Collaborative project
Restrictions for candidates applying	Home and International fees candidates may apply
Partner name	NEC Laboratories Europe
Additional partner stipend enhancement (per year)	£2,400

Project summary

Why this research is important
AI coding agents can now write and change large amounts of software in minutes, but something is quietly lost. Writing code by hand forced programmers to build a mental “theory” of their system and to keep reconciling what they intended with what they built. When agents generate the implementation, that understanding is never formed. Developers increasingly deliver code they did not write and do not fully understand, while the intent behind it (goals, constraints, invariants) stays implicit or is lost. This threatens the reliability and maintainability of the software we all depend on.

Who you will be working with
You will be supervised by Professor Federica Sarro at UCL, together with a member of the UCL Interaction Centre, and Dr Roberto Bifulco and Dr Giuseppe Siracusano of NEC Laboratories Europe, bringing industrial research expertise in software systems and AI agents.

What you will be doing
You will investigate how to close this cognitive gap between what agents build and what developers understand and intend. Existing tools (code review, diffs, tests) assume the developer already understands the code and why it was written, an assumption agents break. You will study how developers form, lose and recover intent when working with agents, then design, build and evaluate tools for a new kind of continuous integration that captures, versions and checks

intent as continuously as code. Directions include turning exploration of alternatives into explicit decisions, protecting developer attention, recovering intent from existing code, and detecting drift between intent and implementation.

Who we are looking for
We are looking for candidates with a software engineering background and a working knowledge of generative AI and coding agents. Experience in human–computer interaction is a plus.