TJEDAN OTVORENOG PRISTUPA 2026

OpenCode.HR: The National Platform for Collaborative Development and Sharing of Research Software

PRESENTERS & SESSION ROLES

Dubravko Penezić, MSc • Software Developer, SRCE

Davor Davidović, PhD • Head of the CIR, IRB

Tom Wildman, MSc • Software Engineer, CZI, IRB

Zagreb, 21 October 2026

SESSION STRUCTURE & TIMELINE

How Today's Session is Organized

PART 1 • 15–20 MIN

National Platform Overview

Presenter: Dubravko Penezić (SRCE)

- National open-source platform initiative
- Sovereign hosting on SRCE infrastructure
- Digital sovereignty, GDPR & FAIR open science
- Foundational ecosystem tools overview

PART 2 • 20–25 MIN

Practical Adoption at RBI

Presenter: Davor Davidović & Tom Wildman (RBI)

- Navigating the OpenCode.hr GitLab interface
- RBI internal structure: departments & lab subgroups
- Visibility levels (Public, Internal, Private) & inheritance
- 3-step onboarding: AAI@EduHr, name format, email request
- Best practices for research software & FAIR code

PART 3 • 10–15 MIN

Open Discussion & Q&A

Presenter: All Presenters & Audience

- Open floor for questions from RBI researchers
- Specific project & laboratory use cases
- Data migration from GitHub / personal machines
- Direct assistance and scheduling next steps

PLATFORM NAVIGATION & ECOSYSTEM

Exploring the OpenCode.hr

Central Hub: https://gitlab.opencode.hr • Authentication via AAI@EduHr • Self-hosted Community GitLab

CORE NAVIGATION AREAS

- **Explore (Javni katalog)**
  Discover public repositories, scientific packages, and tools across Croatian academic institutions.
- **Groups (Grupe & Zavodi)**
  Navigate hierarchical institutional spaces: institutes, faculties, departments, and research labs.
- **Projects (Repozitoriji)**
  Fast access to your active repositories, starred projects, personal forks, and shared codebases.
- **Activity & Snippets**
  Track real-time commits across your teams; store and share quick code snippets and configuration recipes.

IN-REPOSITORY TOOLS & INTEGRATIONS

- ✔ **Repository & File Viewer**
  Full Git version history, branches, tags, visual blame, and commit graph.
- ✔ **Issue Tracking & Kanban**
  Organize research milestones, bug reports, and project deliverables with agile boards.
- ✔ **Merge Requests (MR)**
  Peer code review, inline diff comments, threaded discussions, and branch protection.
- ✔ **Integrated Web IDE**
  Edit code, write Markdown documentation, and commit directly from your web browser.
- ✔ **Auxiliary Tools Suite**
  Excalidraw (architecture diagrams), HedgeDoc (collaborative Markdown notes), IT Tools & Community Wiki.

The RBI Hierarchy: Subgroups for divisions/centres

TOP-LEVEL INSTITUTIONAL GROUP: IRB (Public)

- **Core Architectural Rule: ONLY designated RBI administrators are direct members (Davor Davidović, Tom Wildman, Dubravko Penezić).**
- Why? GitLab permissions inherit downwards. Restricting top-group membership ensures that individual researchers cannot automatically see or modify private projects across other departments.

LEVEL 1 SUBGROUPS: DEPARTMENTS & CENTRES (Zavodi i centri)

- **Departmental Umbrella: Dedicated subgroup for each RBI department/centre: CIR, ZRPZ, CZI, ZKM, ZMO, ZFZ, etc.**
- Clean Segregation: Departmental spaces maintain organizational clarity. Each division manages its own code and personnel without cross-department noise.

LEVEL 2 SUBGROUPS: RESEARCH LABS, PROJECTS & INDIVIDUALS

- **Flexible Grouping: Nested directly under the appropriate department (e.g. IRB / CIR / HPC-Lab, IRB / ZRPZ / Project-X, or IRB / Dept / FirstnameLastname).**
- Full Subgroup Autonomy: The applicant is appointed as Owner of his/her subgroup, allowing them to independently create projects, invite lab members, and assign roles.

ACCESS CONTROL & PRIVACY

Visibility Options for Groups, Subgroups & Repositories

PUBLIC

Open to the Entire World

- Accessible to anyone on the internet without any login or account.
- Mandatory for institutional groups (IRB & department levels) for transparent structure.
- Essential for Open Science: publishing code accompanying journal papers, open-source packages, FAIR research principles, and EU grant obligations.
- Included automatically in the national OpenCode.hr public Project Catalog.

INTERNAL

Academic Community Only

- Accessible exclusively to logged-in users via AAI@EduHr across all Croatian institutions.
- Invisible to general web crawlers and anonymous public visitors.
- Ideal for cross-institutional collaborations: collaborating with colleagues from FER, FOI, PMF, FERIT, etc., prior to public release.
- Protected environment while sharing across the national academic sphere.

PRIVATE

Strictly Confidential

- Visible only to explicitly invited members of the project or subgroup.
- Completely hidden from other logged-in users and search results.
- Ideal for unpublished work, preliminary research scripts, competitive grant proposals, and work-in-progress algorithms.
- You can switch a Private repo to Public at any time upon paper publication.

INSTITUTIONAL POLICY TAKEAWAY: All parent groups are Public for organizational clarity, BUT within your assigned subgroup, you have complete authority to decide whether each project is Public, Internal, or Private!

SECURITY & GOVERNANCE

How Inheritance Governs Groups, Projects & Permissions

MEMBERSHIP INHERITANCE (TOP-DOWN)

How Group Membership Flows

- **Automatic Cascade:**
  Adding a user to a group automatically grants them access to ALL subgroups and repositories beneath it.
- **Example in Practice:**
  If you add a PhD student as Developer to 'IRB / CIR / Lab-Alpha', they automatically become a Developer across every project inside Lab-Alpha.
- **Granular Project-Level Access:**
  If an external collaborator or student should work on ONLY one specific repository, invite them directly at the project level, not at the group level.
- **Permission Escalation Rule:**
  A user's permissions can be elevated in a specific project (e.g. Developer in group, Maintainer in repo), but cannot be lowered below what they inherit from the parent group.

VISIBILITY RESTRICTION RULES

Parent vs. Child Visibility Rules

✔ **Core GitLab Rule:**
A child project or subgroup CANNOT have a higher visibility level than its parent group (e.g. you cannot create a Public project inside a Private group).

✔ **The RBI Design Advantage:**
Because the top-level 'IRB' group and departmental subgroups are all configured as Public, you have UNRESTRICTED freedom beneath them.

✔ **Full Child Flexibility:**
Under your department, you can freely create Public, Internal, or Private subgroups and repositories without hitting parent constraints.

✔ **Safe Isolation:**
A Private repository inside a Public group remains 100% invisible to unauthorized users — only members with explicit invitations can see it.

ROLES & COLLABORATION

Understanding GitLab Member Permissions

OWNER (Vlasnik)

Subgroup Leader / Lab Head

Assigned to subgroup leaders by RBI administrators. Holds full autonomy over their subgroup space.

Key Permissions:

- Invite & remove members, assign/change user roles
- Create new nested subgroups and repositories
- Manage subgroup settings, transfer or archive projects
- Responsible for managing access to lab code

MAINTAINER (Održavatelj)

Project Technical Lead

Technical lead for a specific repository or system.

Key Permissions:

- Configure repository settings, protected branches
- Configure CI/CD pipelines, runners & environment secrets
- Approve and merge production pull/merge requests
- Add project members with Developer or Reporter rights

DEVELOPER (Razvojni član)

Standard Researcher Role

Default working role for researchers, postdocs, and team members.

Key Permissions:

- Push code directly to unprotected development branches
- Create and review Merge Requests; submit inline code comments
- Open, assign, and track issues and research tasks
- Can create new projects inside assigned subgroup

REPORTER (Preglednik)

Read-Only / Auditor / Guest

Ideal for external reviewers, auditors, or students.

Key Permissions:

- Read-only access: view and clone code, inspect branches
- View and comment on issues and Merge Requests
- Cannot push code or modify repository contents
- Safe for thesis students or paper peer reviewers

Step 1 — Initial Authentication via AAI@EduHr

FIRST LOGIN PROCEDURE

Signing In to gitlab.opencode.hr

Step 1: Open Portal

Navigate to https://gitlab.opencode.hr in your web browser.

Step 2: Select AAI@EduHr

On the login screen, click the official AAI@EduHr authentication button.

Step 3: Enter RBI Credentials

Log in using your standard institutional identity credentials (user@irb.hr + password).

Step 4: Automatic Account Creation

Upon first successful authentication, your GitLab user account is generated automatically! No manual registration or password setup needed.

SECURITY & IDENTITY

Federated AAI Login

✔ No separate passwords: OpenCode.hr does not store separate passwords; all verification is handled by AAI@EduHr infrastructure.

✔ Institutional Verification: Guarantees that only authorized staff and students of Croatian academic institutions can create accounts.

✔ Immediate Access: You can immediately browse public projects and use online tools like HedgeDoc and Excalidraw.

Step 2 — Setting the Standard Name Format

⚠ **MANDATORY PLATFORM CONVENTION: Full Name Must Include ', IRB'**

Required Format: Firstname Lastname, IRB (e.g., Ana Horvat, IRB or Ivan Horvat, IRB)

Official Guideline Reference: https://wiki.opencode.hr/opencode/gitlab/teme/ime_korisnika

HOW TO CHANGE YOUR NAME (4 CLICKS)

1. **Click User Avatar:**
   In the top-right corner of the GitLab navbar, click your avatar icon.
2. **Select 'Edit profile':**
   From the dropdown menu, select 'Edit profile' to access your account settings.
3. **Update 'Full name':**
   In the Profile section, locate the 'Full name' field and append: , IRB
4. **Save Changes:**
   Scroll to the bottom of the page and click 'Update profile settings'.

WHY IS THIS STEP ESSENTIAL?

Administrative Verification

- Searchability: Platform administrators search the central database by the ', IRB' suffix to identify our institute's employees among thousands of national academic users.
- Institutional Integrity: Ensures that permissions to RBI departmental subgroups are only granted to verified institute members.
- Quick Approval: Having your name formatted properly enables administrators to approve your subgroup access immediately upon receiving your email.

Step 3 — Subgroup Creation & Access

OFFICIAL REQUEST EMAIL: opencode@irb.hr

After completing Steps 1 & 2, send an email to opencode@irb.hr with the following required details:

OPTION A • LAB / GROUP / PROJECT

Subgroup for Research Team

- For research groups, laboratories, or multi-person research grants.
- State your Full Name, GitLab Username & Department / Centre (e.g. CIR, ZRPZ, CZI...).
- Provide desired subgroup name/slug (concise, up to ~20 characters or acronym, e.g. 'HPC-Lab' or 'BioInfo').
- You receive the Owner role and manage your own members and repositories.

OPTION B • INDIVIDUAL RESEARCHER

Personal Research Subgroup

- For individual researchers hosting personal scientific scripts, packages, and code.
- State your Full Name, GitLab Username & Department.
- Your subgroup will be created under your Full Name beneath your department (e.g. 'IRB / CIR / FirstnameLastname').
- You receive the Owner role with complete control over all your repositories.

OPTION C • EXISTING GROUP MEMBER

Joining an Existing Subgroup

- If your lab or project already has an established subgroup on OpenCode.hr.
- No email to administrators is required!
- Simply notify the Owner (head) of your lab's subgroup.
- The lab Owner can invite you directly through the GitLab interface (Manage → Members → Invite members).

BEST PRACTICES

Making Your Code Reproducible, Citable & FAIR

DOCUMENTATION & CITATION

README & Software Citation

• Clear README.md: State the purpose, prerequisites, installation steps, and a quick-start run example.

• Citation File: Include a CITATION.cff or BibTeX snippet so researchers who use your software can cite it properly.

• DOI Archiving: Connect repositories with data archives (Dabar / Zenodo) for permanent citable DOIs.

LICENSING & OPEN SCIENCE

Open Source Licenses

• Choose a License: Add an explicit LICENSE file (MIT, Apache 2.0, GPLv3, EUPL). Without a license, code is default copyrighted and legally unusable by others.

• National Policy: Fully aligns with the Croatian Open Science Plan and EU Horizon Europe open research mandates.

• Intellectual Property: Balance open science with patent/commercial disclosures where applicable.

REPRODUCIBILITY & CI/CD

Environments & Automation

• Environment Definitions: Include environment.yml, requirements.txt, or a Dockerfile to guarantee reproducible runs.

• Data Separation: Keep code in Git; store large experimental datasets (>500MB) in data repositories (e.g. Dabar).

• GitLab CI/CD: Automate testing, linting, and container builds whenever new code is committed.

CONTACTS & DISCUSSION

Summary, Support Channels & Q&A

RBI SUPPORT & CONTACTS

Central Support Email:

opencode@irb.hr

- **Tom Wildman — Platform Administrator for RBI (CZI)**
  Email: tom.wildman@irb.hr
- **Davor Davidović — RBI Representative at OpenCode.hr (CIR)**
  Email: davor.davidovic@irb.hr

PLATFORM RESOURCES & LINKS

- **GitLab Instance: https://gitlab.opencode.hr**
  Primary repository hosting, issues, CI/CD
- **OpenCode Portal: https://opencode.hr**
  General information, rules, and national news
- **Documentation Wiki: https://wiki.opencode.hr**
  User guides, naming rules & technical docs
- **Community Forum: https://forum.opencode.hr**
  Community discussions, Q&A and support

QUESTIONS & DISCUSSION (10–15 MINUTES)

The floor is open! Let's discuss your specific projects, repository migration from GitHub, or any technical questions.