

KOMUNIKACJA & ESKALACJA - Agent Zero v1


Mission: Zapewnić efektywną, przejrzystą i szybką komunikację w zespole deweloperskim

Core Principles:

- **Transparency:** Wszystkie decyzje są dokumentowane
- **Speed:** Maksymalnie 4h response time dla urgent issues
- **Clarity:** Jasne kanały dla różnych typów komunikacji

COMMUNICATION MATRIX

Kanały Komunikacji

Typ Komunikacji	Kanał	Frequency	Response Time	Owner
Daily Updates	Notion + Daily Standup	Codziennie 9:00 AM	Real-time	Both
Technical Issues	Slack/Discord 	Ad-hoc	< 4h	Developer z expertise
Architecture Decisions	Notion ADR + Call	As needed	< 24h	Both (consensus)
Code Review	GitHub PR	Per PR	< 24h	Assigned reviewer
Project Planning	Notion + Friday Meeting	Weekly	N/A	Both
Urgent Bugs	Phone/Text + Slack	Immediate	< 1h	Available developer
General Discussion	Slack/Discord General	Ad-hoc	< 8h	Both

Contact Information

Developer A (Backend Focus):

- **Slack/Discord:** @developer-a
- **Phone:** [PERSONAL_NUMBER] (Emergency only)
- **Email:** dev.a@project.com
- **Timezone:** CEST (Poland)
- **Working Hours:** 9:00-17:00 CEST (flexible)

Developer B (Frontend Focus):

- **Slack/Discord:** @developer-b
 - **Phone:** [PERSONAL_NUMBER] (Emergency only)
 - **Email:** dev.b@project.com
 - **Timezone:** CEST (Poland)
 - **Working Hours:** 9:00-17:00 CEST (flexible)
-



ESCALATION PROCEDURES



Level 1: Peer Support (0-4h)

When to Use:


- Stuck na problemie > 2 godzin
- Potrzebujesz input techniczny
- Nie jesteś pewny approach


Process:


1. **Slack message:** "@[teammate] - stuck on [issue], need help"
2. **Include:** Problem description, what you tried, current status
3. **Expected response:** Within 4h (sooner if available)
4. **Resolution:** Pair programming lub detailed explanation



Template Message:

 NEED HELP - [Issue Title]

 What I'm trying to do:
[Brief description]

 What I've tried:
- Attempt 1: Result
- Attempt 2: Result

 Specific question:
[What exactly you need help with]

 Time stuck: [X hours]
 Related code: [GitHub link/file path]

Level 2: Project Blocker (4-24h)




When to Use:

- Issue blocks dalszy progress
- Affects project timeline
- Peer support nie rozwiązało problemu

Process:

1. **Escalate w Slack + Notion**
2. **Schedule immediate call** (within 2h)
3. **Document decision** w ADR jeśli needed
4. **Update project timeline** jeśli needed

Escalation Triggers:

-  Critical bug affecting system stability
-  Architecture decision needed
-  External dependency failure

- 🚨 Performance issue > 50% degradation

🔴 **Level 3: Emergency (0-1h)**

When to Use:

- Production system down
- Data loss risk
- Security breach suspected
- Critical deployment failure

Process:

1. **Immediate phone call**
2. **Slack 🚨 emergency channel**
3. **Both developers join call ASAP**
4. **Document incident** post-resolution



MEETING CADENCE



Daily Standup (Every day 9:00-9:15 AM)

Format: Synchronous call lub async Notion update

Template:

Daily Update - [Date]

Developer A:

****Yesterday**:**

- ✅ Completed: [Task 1]
- 🚧 In Progress: [Task 2]

****Today**:**

- 🎯 Plan: [Task 3]
- 🕒 Est. time: [X hours]

****Blockers**:**

- 🚨 [Issue if any]

Developer B:

[Same format]

Sync Needed:

- [] Topic 1
- [] Topic 2

Rules:

- **Max 15 minutes** jeśli synchronous
- **Update by 9:30 AM** jeśli async
- **Flag blockers immediately**
- **Schedule additional sync** if needed



Weekly Planning (Friday 10:00-12:00)

Agenda:

1. **Sprint Review** (30min)

- What was completed?
- What wasn't and why?
- Velocity calculation

2. **Sprint Planning** (60min)

- Next sprint goal
- Task selection i estimation
- Capacity planning

3. **Process Improvement** (30min)

- What went well?
- What can be improved?

- Action items

Preparation Required:

- ☐ Review completed tasks
- ☐ Update task estimates
- ☐ Identify next sprint priorities
- ☐ Prepare any architectural discussions

Monthly Retrospective (First Friday of month)

Deep Dive Topics:

- Architecture review
- Tool effectiveness
- Process improvements
- Knowledge sharing
- Career development

CONFLICT RESOLUTION PROCESS

Step 1: Direct Discussion

When: Disagreement na technical approach

Process:

1. **Schedule 30min call** within 24h
2. **Each person presents** their approach (10min each)
3. **Discuss pros/cons** together (10min)
4. **Reach consensus** lub escalate

Template for Discussion:

Technical Decision: [Topic]

Option A (Developer X):

****Pros**:**

- Benefit 1
- Benefit 2

****Cons**:**

- Drawback 1
- Drawback 2

Option B (Developer Y):

[Same format]

Decision Criteria:

- Performance impact
- Development time
- Maintenance complexity
- Future scalability

Final Decision: [Agreed approach]

Next Steps: [Action items]

Step 2: Research & Analysis

When: Direct discussion nie przyniosło consensus

Process:

1. **Research time:** 1-2 days max
2. **Each developer** researches their approach
3. **Create comparison document**
4. **Meet again** z data

Step 3: Decision Framework

When: Still no consensus after research

Decision Criteria (w kolejności):

1. **Project timeline impact**
2. **Code maintainability**
3. **Performance implications**
4. **Team expertise alignment**
5. **Future scalability**

Final Authority:

- **Backend decisions:** Developer A has final say
 - **Frontend decisions:** Developer B has final say
 - **Architecture decisions:** External consultation if needed
-

COMMUNICATION TOOLS SETUP

Slack/Discord Channels

#general

- General team discussion
- Daily updates
- Casual conversation

#urgent

- Critical issues only
- Emergency notifications
- @channel mentions allowed

#development

- Technical discussions
- Code review discussions
- Architecture questions

#bugs

- Bug reports i tracking
- Testing issues
- Quality concerns

#deployments

- Deployment notifications
- Release coordination
- Infrastructure updates

Notion Integration

Daily Updates: Linked to task tracking

Meeting Notes: Centralized w project space

Decisions: Documented w ADR system

Issues: Tracked w bug triage process

Video Call Setup

Primary: Google Meet lub Zoom

Backup: Discord voice channel


Screen Sharing: Always enabled

Recording: For important architectural decisions

COMMUNICATION TEMPLATES

Urgent Issue Report

 **URGENT:** [Issue Title]

 **Impact:**

- [Who/what is affected]
- [Severity level 1-4]



Timeline:

- When started: [time]
- When noticed: [time]
- Current duration: [X hours]



Status:

- What's been tried: [list]
- Current state: [description]
- Next steps: [plan]



Help Needed:

- [Specific ask]
- [Urgency level]
- [Availability needed]



Links:

- Code: [GitHub link]
- Logs: [location]
- Related: [other issues]



Status Update Template



Status Update - [Feature/Task Name]



Current Status: [In Progress/Blocked/Complete]



Completed:

- [Specific accomplishments]



In Progress:

- [Current work]
- [Est. completion: X hours/days]



Next Steps:

- [Immediate next actions]



Blockers:

- [Any issues needing help]



Timeline:

- Original estimate: [X]
- Current estimate: [Y]
- Variance: [explanation if needed]

Help Request Template



Help Request: [Topic]



Context:

- [What you're working on]
- [Current situation]



Question:

- [Specific question lub problem]



Urgency:

- [Timeline needed]
- [Impact if delayed]



Background:

- [Relevant information]
- [What you've already tried]



References:

- [Code links, documentation, etc.]



COMMUNICATION METRICS

★ **Response Time Tracking**

Target Response Times:

- **Urgent issues:** < 1 hour
- **Technical questions:** < 4 hours
- **Code reviews:** < 24 hours
- **General messages:** < 8 hours

Weekly Metrics:

- Average response time
- Number of escalations
- Resolution time for blockers
- Communication satisfaction




Weekly Communication Report

Week XX Communication Report

Metrics

- Messages exchanged: X
- Average response time: Xh
- Escalations: X (target: < 2)
- Meetings held: X

Effectiveness

-  What worked well
-  Communication gaps identified
-  Improvements for next week

Action Items

- [] Process improvement 1
- [] Tool adjustment 2

EMERGENCY CONTACT PROTOCOL

When to Use Emergency Contact:

Production Emergencies:

- System completely down
- Data loss occurring
- Security breach suspected
- Critical customer impact

Development Emergencies:

- Corrupted git repository
- Accidental data deletion
- Infrastructure failure
- Blocked for > 8 hours

Emergency Contact Steps:

1. **Try Slack/Discord first** (may be faster)
2. **Phone call if no response** within 30min
3. **Text message** jako backup
4. **Document incident** after resolution

Out of Hours Protocol:

Normal Hours: 9:00-17:00 CEST

Response Expected: Within working hours

After Hours: Emergency only

Weekend: Critical issues only

Communication Protocol Last Updated: 7 października 2025, 13:05 CEST

Next Review: Monthly during retrospective

Protocol Owner: Development Team