

# Hackathon Best Practices Checklist

24 practices to run a hackathon that produces real results.

## BEFORE THE HACKATHON

### Planning and Resource Alignment

- 1. Define your why first.**  
Product adoption, talent discovery, community growth, or open innovation — each goal shapes format, audience, and prize structure.
- 2. Choose your format strategically.**  
In-person for depth and collaboration. Virtual for global reach. Choose based on your goal, not convenience.
- 3. Set KPIs before anything is booked.**  
Reach: registrations, impressions. Engagement: submission rate, mentor utilization. Outcome: projects advanced, hires made, API adoption.
- 4. Build your resource stack early.**  
Set up API keys, cloud credits, and sandbox environments before the event opens. Participants should spend day one building, not troubleshooting access.

### Promotion, Registration, and Onboarding

- 5. Choose promotion channels deliberately.**  
GitHub, Discord, and Reddit for technical audiences. LinkedIn and email for professional and enterprise segments. Meet your audience where they already are.
- 6. Segment your promotion by audience type.**  
Write distinct messages for students, mid-career engineers, and senior builders. Match incentives to motivations.
- 7. Design registration as your first quality filter.**  
Capture skills, tools, and domain experience at sign-up for effective team formation. Keep the form to five fields or fewer.
- 8. Start onboarding before the event.**  
Distribute starter kits, API docs, and mentorship schedules at least one week out. Run pre-event workshops and send milestone emails.

## DURING THE HACKATHON

### Virtual Hackathons

- 9. Unify your platform stack.**  
Registration, submissions, project tracking, and communication in one place. Fragmented tools create friction and accelerate drop-off.
- 10. Default to async for global participation.**  
Recorded workshops, written briefs, and async Q&A threads remove time zones as a barrier. Add a dedicated progress channel to keep momentum visible.
- 11. Run live engagement moments on a schedule.**  
Structure them around energy dips: hour 12, hour 18, the final stretch. A hosted AMA, mini-challenge, or emcee check-in resets momentum.

## In-Person Hackathons

- 12. Treat venue layout as strategy.**  
Co-locate similar-challenge teams. Place mentor stations at foot traffic points. Designate quiet zones and separate social spaces.
- 13. Test everything before doors open.**  
Plan for 3x the expected device count on power and WiFi. Stress-test the network. Backup generator on standby. AV tested the day before.
- 14. Design specifically for first-timers.**  
Buddy systems, clear signage, and a physical help desk reduce the anxiety that causes promising participants to disengage before they have started anything.
- 15. Plan food, drink, and rest areas for overnight events.**  
Scheduled meals prevent the energy crash at the 18-hour mark. Designate a rest area. Have an on-site emergency contact protocol.
- 16. Keep energy high throughout.**  
Mid-event social activity, stretch prompts, periodic emcee check-ins. Background music and a live progress board maintain the buzz.

## Demos, Judging, and Awards

- 17. Decide on demo format upfront.**  
Virtual: recorded submission plus live Q&A for finalists. In-person: 5 minutes to demo, 5 to 7 minutes Q&A. Communicate before registration closes.
- 18. Draft judging criteria before you open registration.**  
Publish publicly. Use 4 to 5 weighted categories. Define what each score looks like per category so all judges use the same scale.
- 19. Enforce blind scoring in the first round.**  
Anonymize submissions before judges see team names or institutions. Removes institutional bias faster than any other single intervention.
- 20. Run judge calibration before demo day.**  
Share 2 to 3 sample projects. Have judges score independently, then compare. 15 minutes of calibration prevents inconsistent results.
- 21. Give the awards ceremony real weight.**  
Recognize every team, not just the top three. Acknowledgment is what converts one-time attendees into long-term community members.

### AFTER THE HACKATHON

## Follow-Up and Long-Term Impact

- 22. Act within 48 hours.**  
Publish project repositories, send personalized follow-ups to standout teams, and post a public recap. Waiting a week is waiting too long.
- 23. Measure outcomes at 30, 60, and 90 days.**  
Track teams that continued building, connections made with industry partners, and adoption rate on promoted developer tools. These numbers justify the next program.
- 24. Invest in your alumni community.**  
A well-managed Discord or Slack channel keeps the community warm, builds a ready pool for future activations, and turns past participants into recruiters for the next cohort.

AngelHack has more than 15 years of experience, running 450+ hackathons across 100+ cities. We design programs end-to-end or as modular services: from strategy and promotion to on-the-ground execution and post-event follow-through. [Let us help you build your next program.](#)