AAAI-25 / IAAI-25 / EAAI-25

Hackathon Rules

The Spirit of the Competition

Remember that hackathons are like marathons. Some people go to compete but most people take part to better themselves and have fun. Whatever the reason is you're at a hackathon, make sure you're upholding the hacker spirit by collaborating with other teams, helping beginners, and having fun.

The Rules of the Competition

- The maximum team size is not capped, but we will verify that all members contribute meaningfully to the entry. Names without contributions may be excluded from the team, at organizers' discretion.
- Teams should be made up exclusively of participants who are not organizers, volunteers, judges, sponsors, or in any other privileged position at the event.
- The Hackathon participants should ideally check-in into the Submission platform (GitHub). Checking-In refers to registering your information in the README for your team's repository.
- Teams can of course gain advice and support from organizers, volunteers, sponsors, and others.
- All work on a project should be done during the week of the hackathon.
- Teams can use an idea they had before the event.
- Teams can work on ideas that have already been done. Hacks do not have to be “innovative”. If somebody wants to work on a common idea they should be allowed to do so and should be judged on the quality of their hack. These days it’s hard to find something that’s fully original and teams might not know an idea has been done before anyway.

- Teams can work on an idea that they have worked on before (as long as they do not re-use code).
- Teams can use libraries, frameworks, or open-source code in their projects. Working on a project before the event and open-sourcing it for the sole purpose of using the code during the event is against the spirit of the rules and is not allowed.
- Adding new features to existing projects is allowed. Judges will only consider new functionality introduced or new features added during the hackathon in determining the winners.
- Teams must stop hacking once the time is up. Timestamps for commits before the submission deadline will be checked for this.
- Projects that violate the Code of Conduct are not allowed.

- Teams can be disqualified from the competition at the organizers' discretion. Reasons might include but are not limited to breaking the Competition Rules, breaking the Code of Conduct, or other unsporting behavior.

Additional specific rules for our Virtual Hackathons:

- We require all teams to submit a 2 minute or less demo video.
- Your video must be created the week of the hackathon.
- Your code must be on the team github repository
- We do allow you to submit your project to other hackathons as long as the other hackathon also allows this.
- If you're continuing work on an old project, you MUST specify in your README submission form what was worked on 1) before the hackathon and 2) during the hackathon.

- **Eligibility for Participation Certificate**: The participant/team should have successfully checked in and submitted a project on the README during the Hackathon Timeline.

If at any point during the Hackathon you find a specific rule unclear, or are uncertain whether you are permitted to do something, please contact the organizers and we will help address any concerns.

Eligibility

- Open to researchers, labs, and companies. Sign up as a team or as an individual looking to join a team.

- As per our Code of Conduct, there is no discrimination on the basis of race, religion, national origin, color, sex, gender identity, sexual orientation, social class, economic status, veteran status, disability, or age.

Demos

- After hacking finishes, the organizing team will review all entries for compliance with the rules. All entries found to be in compliance will proceed to the finalist stage.
- The final judging will be based on your demo video, but the organizers will run a reproduction check to ensure that the video matches the submission.
- You are encouraged to present what you have done even if your hack is broken or you weren't able to finish. It's okay if you didn't finish your hack—that happens all the time! Completion is only one part of the judging criteria, so you might still do well.

- Also, demoing is not just about the competition. It's a chance to share with others what you learned and what you tried to build—that's what hacking's all about! —it doesn't matter how good the demo is! In the case that you don't have anything to demo, you can give a presentation about what you tried and what you learned. Hearing what other people learned is interesting and inspiring for other attendees.

Judging Criteria

Teams will be judged on these four criteria. Judges will weigh the criteria equally. During judging, participants should try to describe what they did for each criterion in their project via the demo video. Note that judges are looking to *award* points based on the below criteria, not dock the efforts, of every submitted project.

- **Technology**: How technically impressive was the hack? Was the technical problem the team tackled difficult? Did it use a particularly clever technique or did it use many different components? Did the technology involved make you go "Wow"?
- **Design**: Did the team put thought into the user experience? How well designed is the interface? For a website, this might be about how beautiful the CSS or graphics are. For a hardware project, it might be more about how good the human-computer interaction is (e.g. is it easy to use or does it use a cool interface?).
- **Completion**: Does the hack work? Did the team achieve everything they wanted?

- **Learning**: Did the team stretch themselves? Did they try to learn something new? What kind of projects have they worked on before? If a team which always does virtual reality projects decides to switch up and try doing a mobile app instead, that exploration should be rewarded.

These criteria will guide judges but ultimately judges are free to make decisions based on their experience & technical intuition of which projects are the most impressive and most deserving. Any decision made by the judges would be deemed final.

It's important to note that these judging criteria does not include:

- How good your code is. It doesn't matter if your code is messy, or not well commented, or uses inefficient algorithms. Hacking is about playing around, making mistakes, and learning new things. If your code isn't production ready, we're not going to mark you down.
- How well you pitch. Hacking is about building and learning, not about selling.
- How good the idea is. Again, hackathons aren't about coming up with innovative ideas. It's about building and learning.
- How well the project solves a problem. You can build something totally useless and as long as you're learning and having fun, that's a good hack! Sometimes a pointless project is one of the best hacks!

So don't worry about coming up with the next big idea or building the next Facebook. You'll have plenty of time for that outside the hackathon. just focus on learning, having fun, and making new friends. At the end of the day the skills you learn and the friends you make might lead to the next big thing—but you don't have to do that to win a hackathon.

Remember

The competition is just a part of the hackathon. To make the most out of the event, try something new, teach other people, and make new friends!