

Judging criteria – technical entry

Modules > Pitch and Competition > Competition > Technical entry

You are now aware of what is expected from you in terms of *entry requirements* for the technical entry, and may want to know how you will be judged and based on what criteria.

We are accepting focused technical entries for the first time in 2014 given that we have now integrated more coding materials into our course framework. Based on feedback from previous years we also want to specifically recognise the achievement of teams who have created a functioning minimum viable product by manipulating code themselves.

How technical competition entries will be judged

Entries will be judged in these four areas:

- a) Functionality
- b) Focus
- c) Suitability to problem
- d) Technical complexity

For the technical entry shortlisting *a) Functionality* and *b) Focus* are the most important areas. Entries that are very strong in both these areas are likely to be shortlisted as long as they don't get very low scores in the other two areas. However, in order to make it to finalist stage you certainly will require a good and balanced entry in all of the four areas.

a) Functionality

Good competition entries will show us that the core feature of the app works really well, i.e. is functional and does what it is meant to do. For example, if you have created an app that uses the Google map API, we would expect that this has been integrated in a way that works and that there are no fatal, show-stopping bugs. Or another example: Imagine that Twitter had created their first app version, and you could draft but not send the tweet. If the core feature is not working the whole product becomes pointless.

b) Focus

The next aspect beyond basic operational functionality is whether the app has implemented the core functionality needed for this app in the simplest, most focussed and intuitive way. Are all the implemented features really essential or are they just distractions for the user and confuse them? Do you have a good experience as a user?

c) Suitability to problem

Has your team used the right approach to coding, libraries, plug-ins or is the problem to be solved approached effectively? Has your team found a solution to make something that is technically difficult easy? For example, integrating SMS into an app is typically very tricky, but using the Twilio API in a smart way is much easier. Have you been street-smart and clever about technical implementation?

Technical complexity

This area looks at how ambitious your team has been in terms of the degree of technical ability that was required to implement the solution. How elegant is the solution? Did you tackle a complex technical problem or implement an advanced level of coding?

Suitability to problem and *technical complexity* will be looked at in a balanced way.

So now that you know the areas you will be judged on, how will the reviewers for short-listing and the final judges pick the best teams? Below you can see the guidance we are giving them via our technical scorecard.

Exercise:

Score your own competition entry or ask other teams in your school to judge your entry, so you can see how you might fare in your own eyes and those of your peers.

Technical entry score card

Score	1	2	3	4
Judgement in words	Poor	Fair	Good	Great
a. FUNCTIONALITY: Does the app work functionally? No fatal, show-stopping bugs				
b. FOCUS: Is core functionality implemented in the simplest, most focussed way? No distractions, confusing unnecessary elements				
c. SUITABILITY TO PROBLEM: Has the team used the right approach to coding? Balanced approach between creating from scratch, remixing or adding libraries. Technically difficult challenge made easy				
d. TECHNICAL COMPLEXITY: How elegant is the solution?				
OVERALL SCORE: Has the team created a full minimum viable product?				