

Test Cases: Poster Judging Plus

Test 1: Uploading Poster	
Input	<ul style="list-style-type: none">• title• category• emails• image
Test case	<ul style="list-style-type: none">• input proper title• input proper category• input proper emails• validate emails in dB• input proper description• input image as png• validate image

Test 2: Judge a Poster	
Input	<ul style="list-style-type: none">• input the 5 categories:
Testcase	<ul style="list-style-type: none">• make sure each number is valid from one to twenty• make sure all fields have input.• upload to the database.• validate

Test 3: View Feedback	
Input	<ul style="list-style-type: none">• login to the dB
Testcase	<ul style="list-style-type: none">• check if feedback exists in dB• retrieve and display

Risk and Mitigation

RISK #1

Our team may be unable to properly link Flask to the SQLite database in a way that would satisfy the requirements for this project by the time of our presentation.

We can mitigate this risk by having a backup system using PHP instead of Flask that we can create in a timely manner if Flask doesn't end up working out for this project.

RISK #2

Other obligations or other classes may hinder team members from being able to fully commit as much time to the project as is needed to complete it by the deadline.

We can mitigate this risk by removing the feature of having the system automatically determine a winner. This feature can be implemented later given more time, but a winner can still be calculated manually even without this feature so it isn't crucial to the functionality of the application.

RISK #3

A judge may create an account, get assigned posters to judge, but then never actually judge any contestants that they're supposed to.

We can mitigate this risk by creating a deadline that a judge needs to judge at least one poster by. If they don't their account will be deleted and the posters will be reassigned.