

Final Project Specification Document

This is the final project rubric that we will use to evaluate final project submissions. The document provided to students, specifying the requirements of the final project can be found above. Graders should download the rubric and, while grading, leave a short comment in each box that corresponds to the score that they think the student should receive for that requirement. There are a few reference points in the rubric to help guide you in determining what score you think the project should receive in each of the requirements. Please remove them and submit your comments where appropriate.

Name of Project Project name here

Author's Information Author's name and email here

Evaluator Information Your name and email here

Scores								
User Interface Requirements	Score	Comments	0	1	2	3	4	5
Run app on a dev server locally for testing/grading (connecting to Rinkeby if required)			App doesn't run		App runs with some tweaking			App runs
Should be able to visit URL and interact with the app			Cannot visit the url and view the running application		Application runs, but does not connect to web3			It works as expected. I can visit the application in the browser and interact with it via metamask
The applications should have the following features: Display the current account Sign transactions using metamask / uPort Reflect updates to to the contract state			None of the features are implemented					All of these features are implemented
Testing	Score	Comments	0	1	2	3	4	5
5 tests (Javascript or Solidity or both) with explanations for each smart contract written (where appropriate)			There are no tests written for the contracts		There are a few tests written for each contract			There are at least 5 tests written for each contract, and they clearly explain why they were implemented.
Tests are properly structured (ie sets up context, executes a call on the function to be tested, and verifies the result is correct)								
Tests provide adequate coverage for the contracts			None of the tests test the contract functions					Tests verify that execution is correct in 5 different contexts (when appropriate)
All tests pass			None of the tests pass		More than 50% of the tests pass			All of the tests pass
Design Pattern Requirements	Score	Comments	0	1	2	3	4	5
Implement a circuit breaker / emergency stop [1]			The project does not implement emergency stop / circuit breaker design pattern.		The project attempts to implement the circuit breaker design pattern			The project successfully implements the circuit breaker design pattern.
What other design patterns have you used or not used?			The submission does not include an explanation of their design pattern decisions					The submission includes an adequate description of their design decisions.
Security Tools / Common Attacks	Score	Comments	0	1	2	3	4	5
Explain what measures they've taken to ensure that their contracts are not susceptible to common attacks			The student does not provide any explanation		The student provides a few explanations			The student expresses adequate knowledge about common attacks and has sufficiently guarded against them.
Library / EthPM	Score	Comments	0	1	2	3	4	5
At least one of the project contracts includes an import from a library or an ethPM package. If none of the project contracts do, then there is a demonstration contract that does.			The student does not import a library or ethpm package.					The student does import a library or ethpm package.
Additional Requirements	Score	Comments	0	1	2	3	4	5
Smart Contract code should be commented according to the specs in the documentation https://solidity.readthedocs.io/en/v0.4.21/layout-of-source .								
Stretch Goals for bonus points	Score	Comments	0	1	2	3		
Project uses IPFS								
Project uses uPort								
Project uses the Ethereum Name Service								
Project uses an Oracle								
Project implements an Upgradable Pattern Registry or Delegation								
Project includes one of their smart contracts implemented in LLL / Vyper								
Testnet Deployment: The addresses provided in deployed_addresses.txt correctly point to deployed contracts on the rinkeby testnet.								

Total Points	Total Possible	General Comments
--------------	----------------	------------------

0

60

[1] consider removing this requirement
-Josh Crites