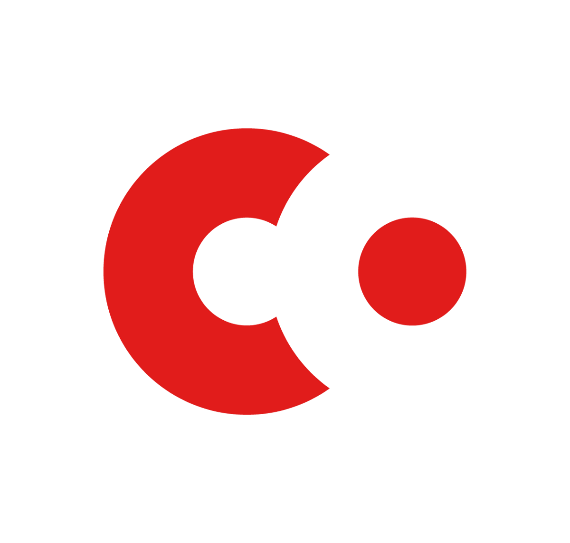
Innovation Challenge 2019:

# **Blockchain Balloting**



## **Background**

As an emerging technology, blockchain solutions are beginning to be used by more established corporations. ConocoPhillips is currently exploring blockchain technologies for a variety of use cases. An example of a use case Oil & Gas companies might be interested in is Joint Operating Balloting. When multiple companies own a stake in a single piece of land, one company (the operator) will drill a well and ballot the other owners (non-operators) to see if they would like to help cover the cost of the well in exchange for an increased share of any profits the well produces. You will be building a simplified version of this process using the Corda blockchain framework. You may begin with Corda’s provided example app and modify it to allow for basic balloting and response capabilities between a couple companies.

## **Deliverables**

* Corda network running locally with flows that address the use case above and a web GUI that allows users to interact with these flows.

## **Milestones**

* Base Case:
  + Get Corda example app running on a local network on a single machine with the ability to create and view IOUs using the web frontend.
* Develop/Enhance:
  + Modify the example app by adding a flow that allows creation of a basic ballot with a state recorded on the blockchain (Company A can ballot Company B). This will include adding a new flow, state, and contract to the app for Ballots.
  + Modify the example app by adding a flow that allows a company to respond to a ballot with a simple yes/no. This will include adding a new flow, state, and contract to the app for Responses.
  + Modify the Spring Server API and web frontend to allow creation of and response to ballots using a GUI.

## **Bonus**

* Modify ballot flow to allow balloting multiple companies in a single ballot.
* Modify ballot response validation to include a due date.
* Deploy Corda network to multiple machines.
* Anything else above and beyond these requirements.

## **Rules**

* You may use online resources, but you may not post on any online forum asking for assistance.
* If you use open source software or code snippets, you must give proper credit to the author.

## Documentation

A GitHub repo has been set up so that you may collaborate on your code if you wish. <https://github.com/jarrettalexander-cop/innovate2019>

Here are some key files in the Corda example app that you will want to modify as part of the challenge, highlighted in the screenshot below (note that kotlin paths, for example workflows-kotlin, can also be used):

* MainController.kt – this is where you will modify the Spring Server API by adding calls to your ballot and response flows so that the web frontend may create and respond to ballots.
* Angular-module.js – this is the javascript for the web frontend. Modify this to call your new Spring Server API methods to create and respond to ballots.
* Index.html – this is the html for the web frontend. Modify this to display and interact with ballots and responses in tandem with your javascript.
* IOUContract – either modify IOUContract or add a BallotContract with a verify function for verifying new ballots. You will also have to add a ResponseContract later.
* IOUSchema – either modify IOUSchema or add a BallotSchema hold the structure of a ballot, such as who is creating the ballot and who is responding. You will need to add a ResponseSchema later.
* IOUState – similar to IOUSchema, either modify this or add a BallotState to hold the structure of a ballot that will be stored on the blockchain. You will need to add a ResponseState later.
* ExampleFlow – either modify ExampleFlow or add a BallotFlow to handle the process flow for how a ballot is added to the blockchain. You will need to add a ResponseFlow later.

