IOWA STATE UNIVERSITY

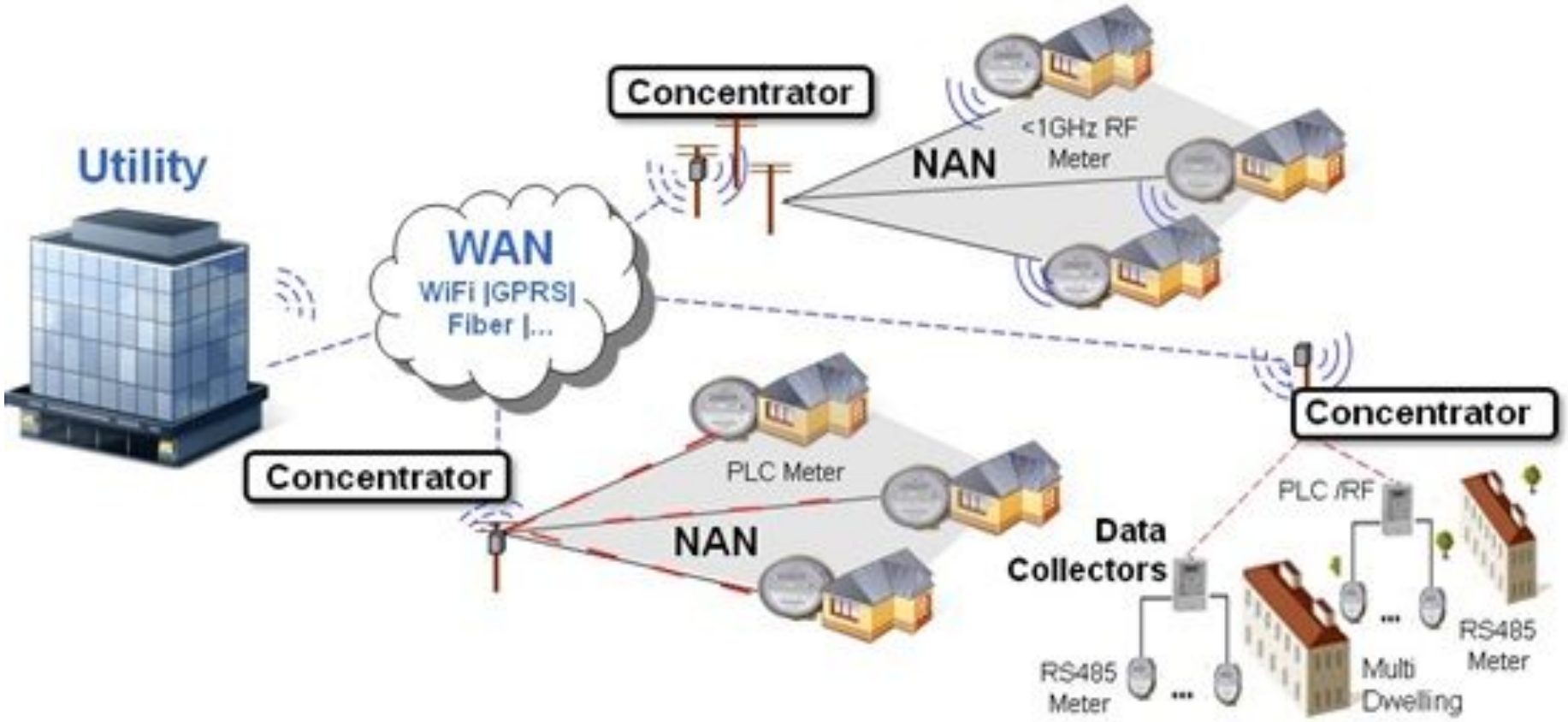
Applying Blockchain to Energy Delivery Systems

Client: Grant Johnson
Adviser: Manimaran Govindarasu
Team: sdmay20-12
Website: sdmay20-12.sd.ece.iastate.edu

Anthony Cosimo - Test Engineer
Jacob Dawson - Project Manager
Keegan Bloedel - API Architect
Katherine Ringgenberg - UI Architect
Steven Rein - Blockchain Architect
Dakota Moore - Cybersecurity Manager

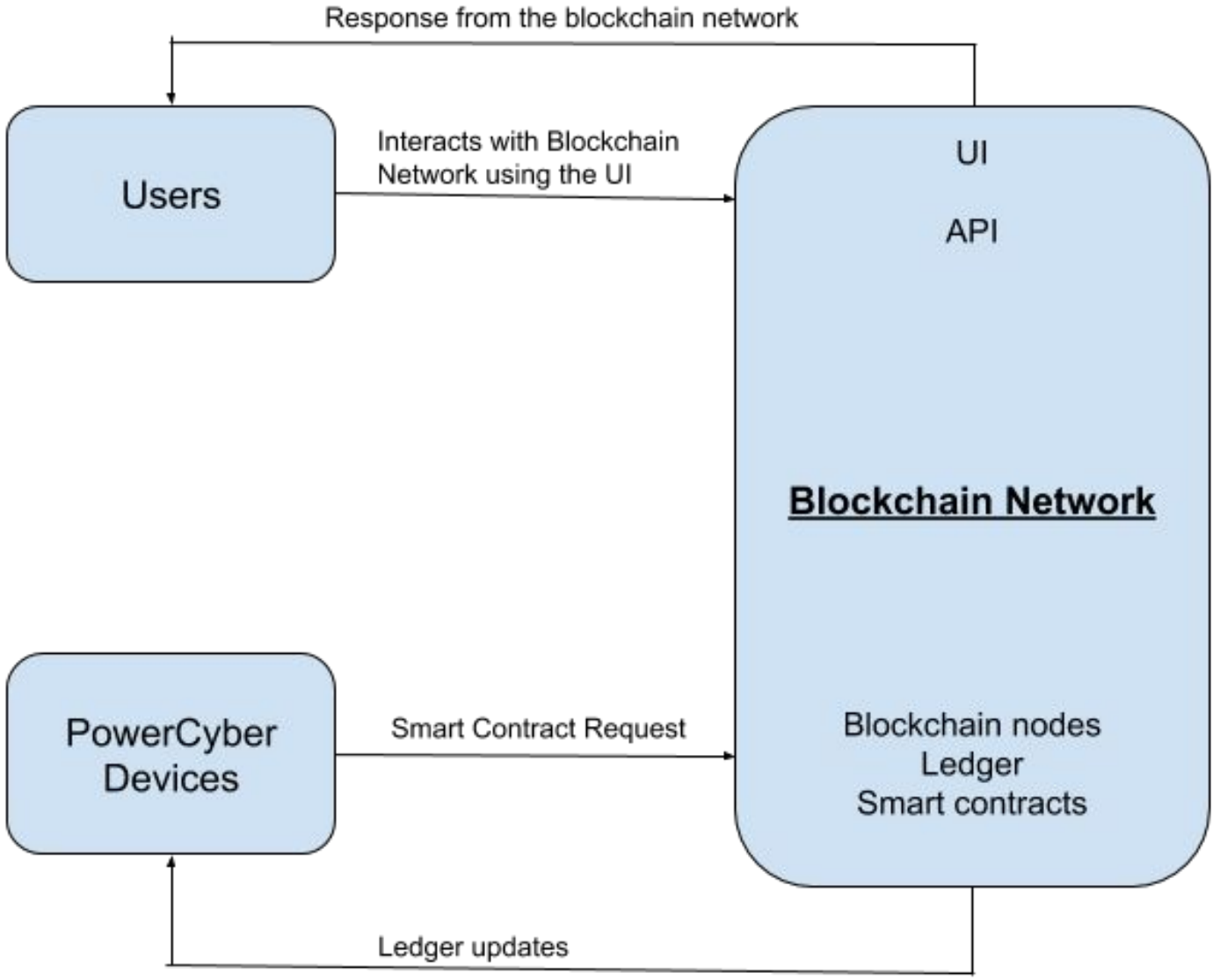
Our Vision

Power grids are getting more complex while still relying on public internet infrastructure for communications, making it more vulnerable to attacks. Implementing Block chain can fix this!



Source: https://e2e.ti.com/blogs_/archives/b/toolsinsider/archive/2014/06/05/the-power-of-information-onthegrid

Conceptual Diagram



Functional Requirements

Blockchain:

- Consist of at least five nodes for transaction consensus
- Consist of multiple nodes to act as orderers

Smart Contract Layer & API:

- API: expose the Smart Contract functions upon authentication
- Smart Contracts: read, update, delete, and query data

User Interface:

- Ability to request/view measurements and/or metrics if the user has the authority to do so

Operational Environment:

- Blockchain network, UI, and API: run on a linux-based virtual machine provided by PowerCyber

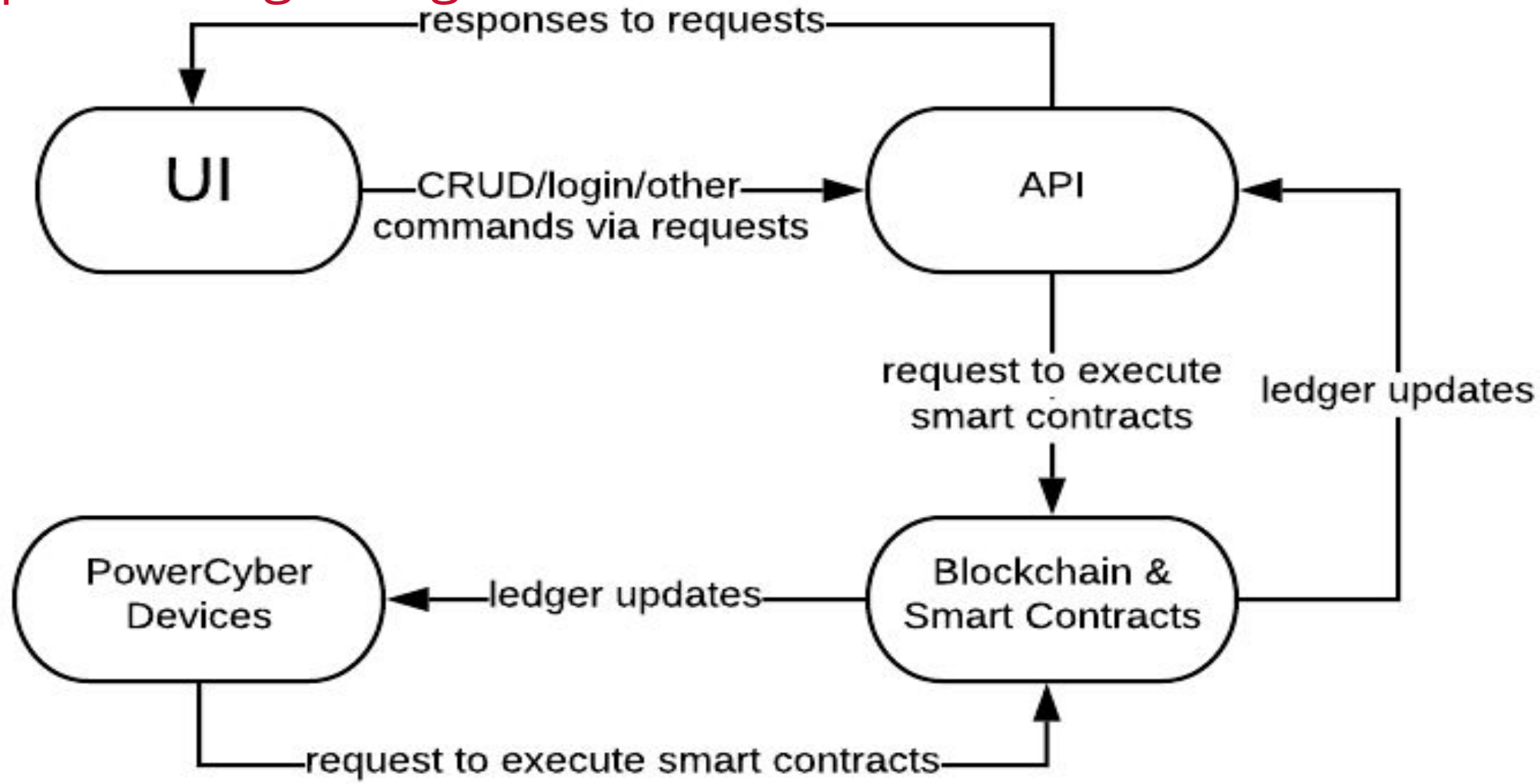
Non-Functional Requirements

- API and Blockchain: have swagger documentation
- Blockchain network nodes: be ran using Docker containers
- Smart Contracts: update blockchain network
- Include a descriptive project wiki
- CI/CD: can run all tests and deploy to the necessary environments

Constraints

- No budget - constrained to using PowerCyber resources.
- Hyperledger Fabric must be used as the permission-based distributed ledger framework per client's request.
- Hyperledger Fabric related code must be written in JavaScript

Conceptual Design Diagram



Level 0 Diagram

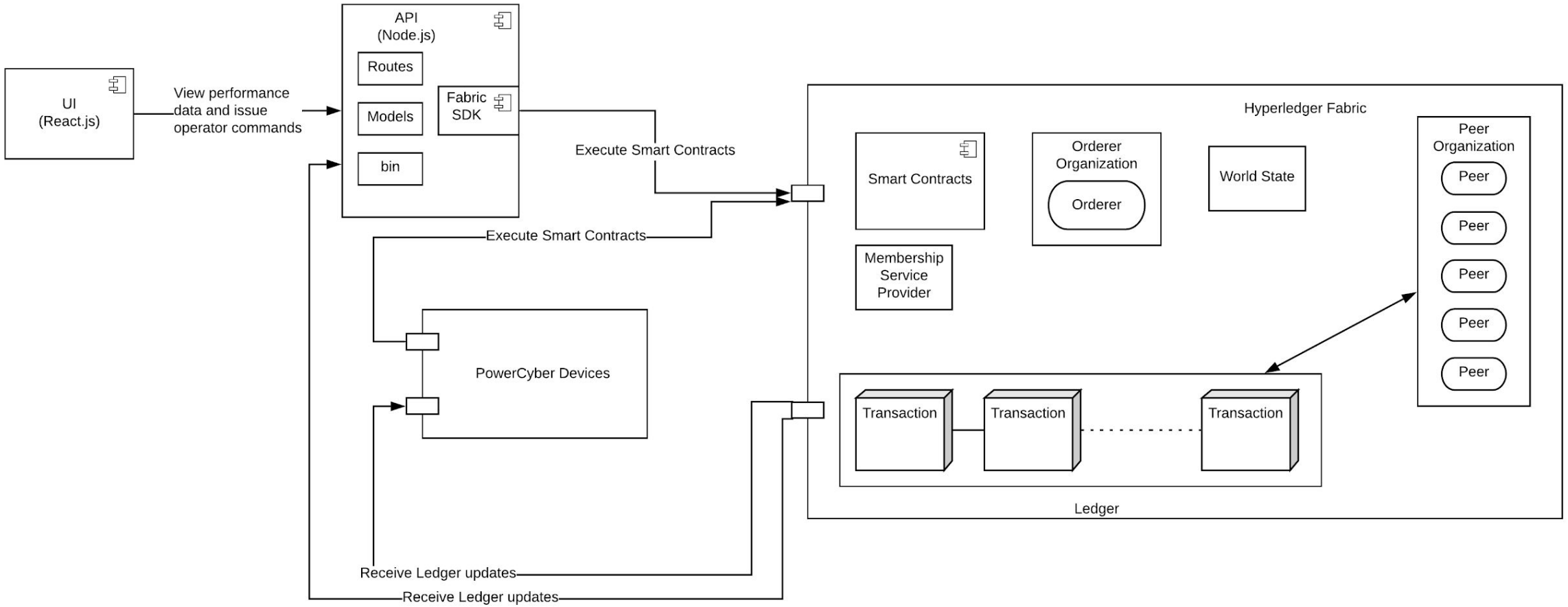
Project Tasks

- Requirements Gathering
- Domain Knowledge
- Blockchain Network
- Smart Contract Layer
- API
- UI
- PowerCyber Device Integration
- System-level Testing

Risk and Mitigation

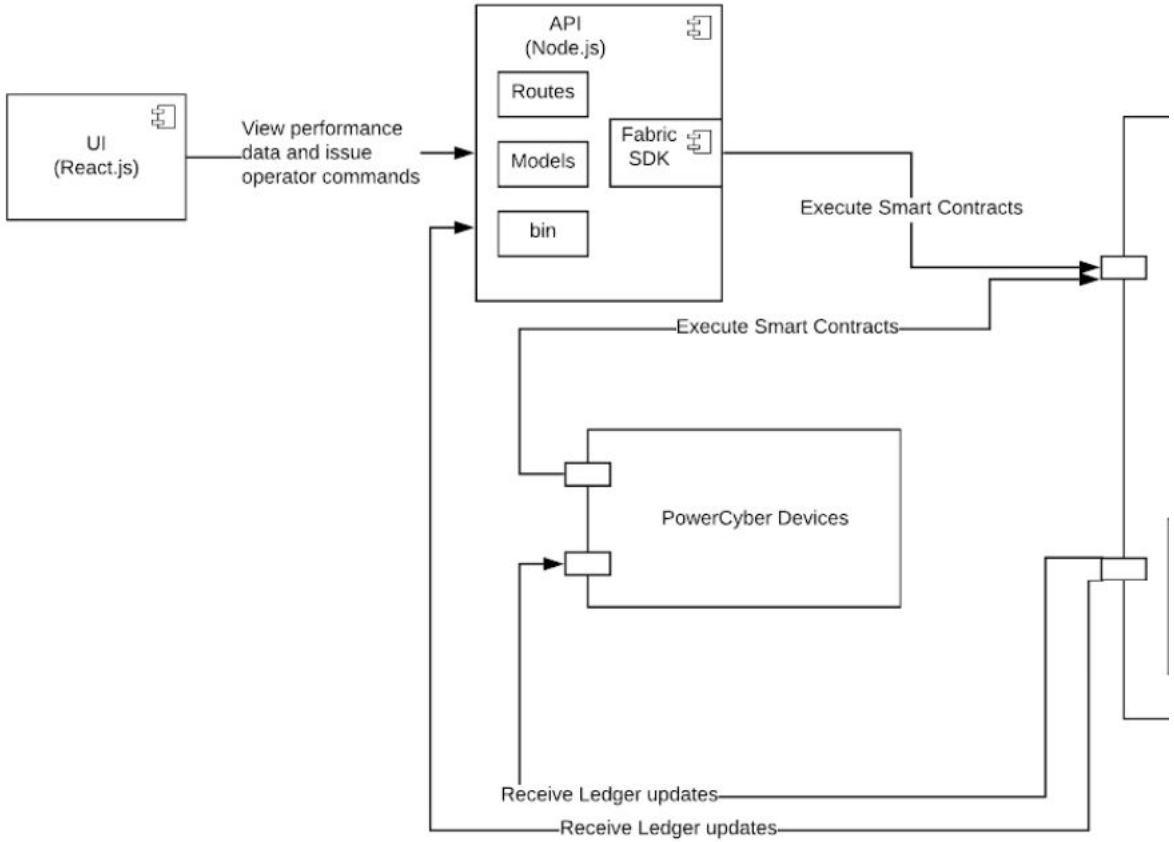
Title	Risk	Mitigation Action
Blockchain could possibly be the wrong solution.	Avoid	Profile Blockchain system to determine fitting use cases.
Team's lack of domain knowledge could lead to easy to detect flaws being introduced into the system.	Mitigate	Gain domain knowledge by visiting PowerCyber, interviews, and consulting with client and advisor. Frequent feedback.
Potential security risks exposed by the UI.	Mitigate	Implement input security and disable file uploads.
A user's login could be compromised.	Mitigate	Create and analyze user action log, two-factor authentication, and HTTPS.
Integration of system components could fail.	Mitigate	Integrate early and integrate often. Maintain Swagger documentation.

System Architecture



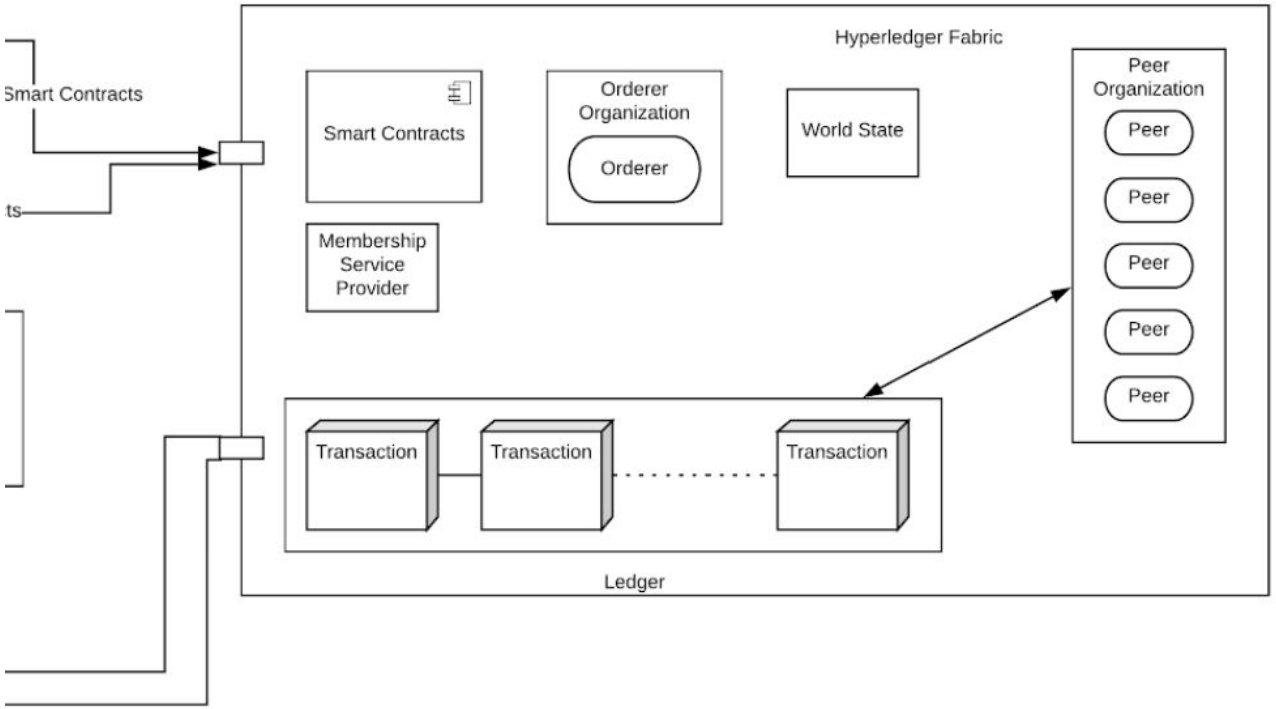
UI & API Architecture

- React.js
- Node.js
- Fabric SDK
- Smart Contracts



Blockchain Architecture

- Hyperledger Fabric
- Docker
- Raft Consensus
- CouchDB



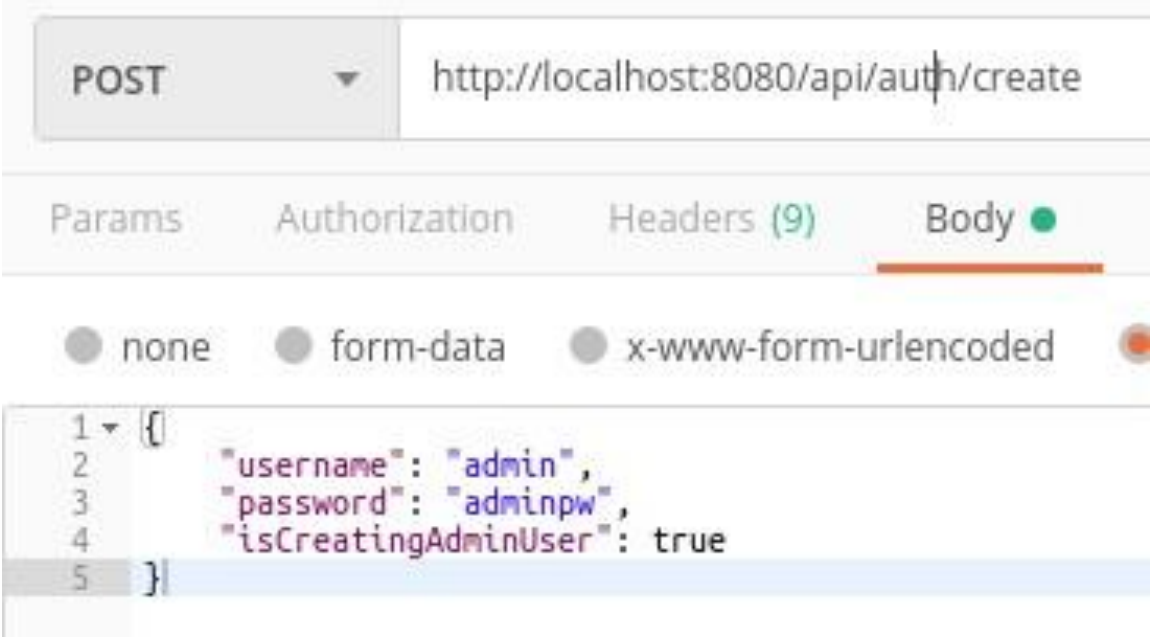
Standards

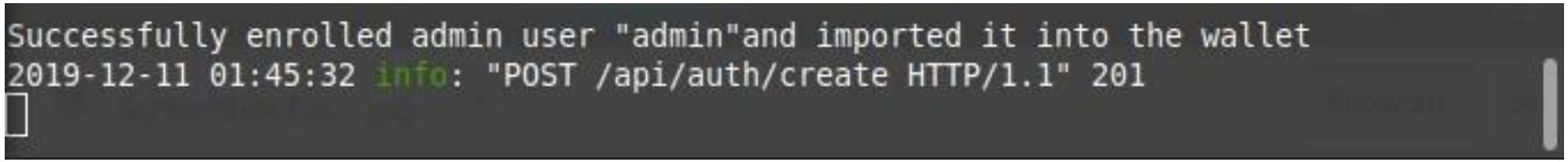
- 1028-2008 - IEEE Standard for Software Reviews and Audits
- 26515-2018 - IEEE International Standard - Systems and software engineering -- Developing Information for Users in an Agile Environment
- NERC CIP-011-2 - Cyber Security - Information Protection

Prototype: Blockchain

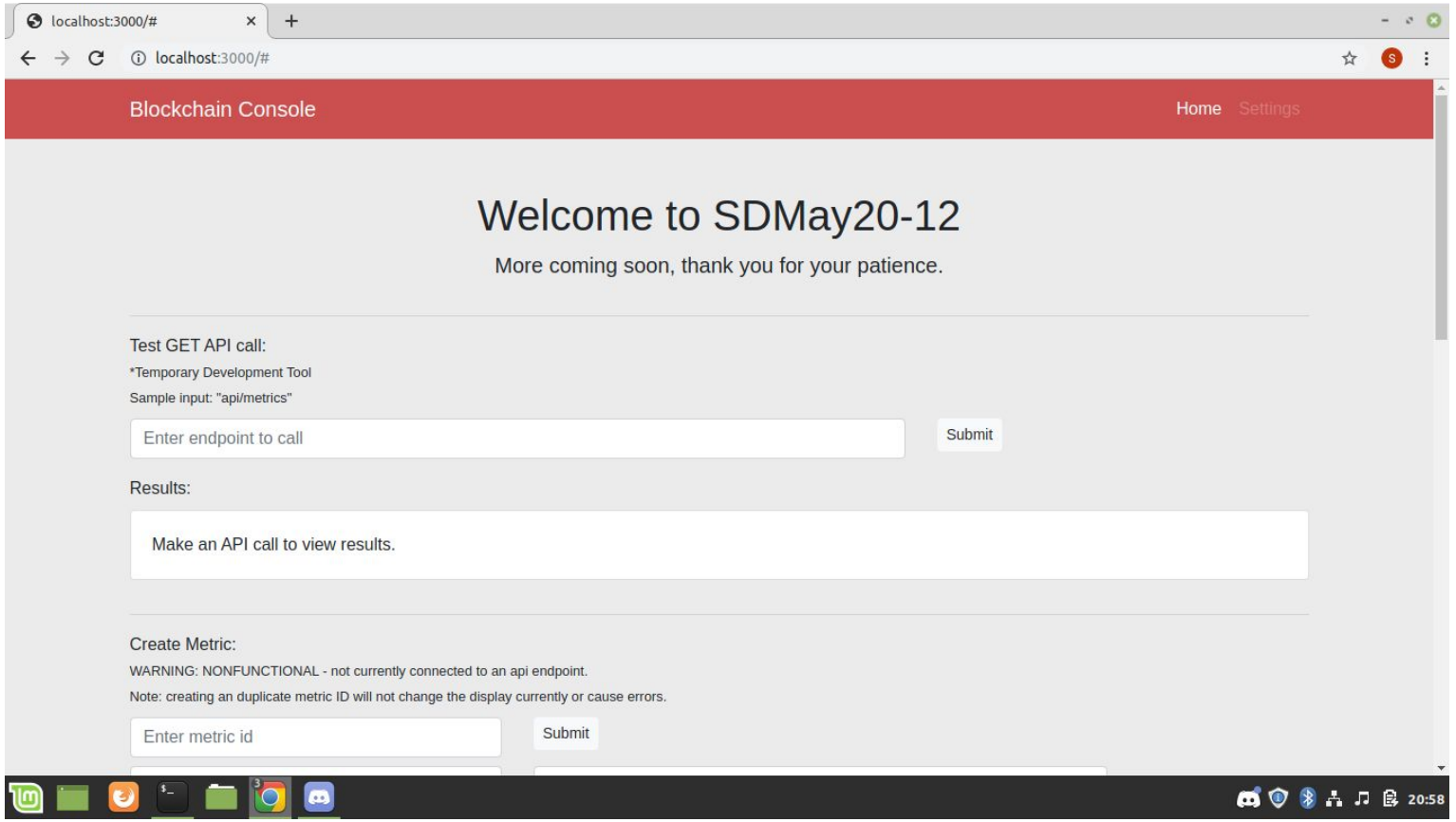
CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
9f13a79f10bb	hyperledger/fabric-tools:latest	"/bin/bash"	12 minutes ago	Up 12 minutes		cli
abd8c6d0b325	hyperledger/fabric-peer:latest	"peer node start"	12 minutes ago	Up 12 minutes	0.0.0.0:11051->11051/tcp	peer4.org1.example.com
9c13320bfb87	hyperledger/fabric-peer:latest	"peer node start"	13 minutes ago	Up 12 minutes	0.0.0.0:8051->8051/tcp	peer1.org1.example.com
7e31c18b2eef	hyperledger/fabric-peer:latest	"peer node start"	13 minutes ago	Up 12 minutes	0.0.0.0:9051->9051/tcp	peer2.org1.example.com
9ad4004a06a6	hyperledger/fabric-peer:latest	"peer node start"	13 minutes ago	Up 12 minutes	0.0.0.0:7051->7051/tcp	peer0.org1.example.com
150ecd85d2f1	hyperledger/fabric-peer:latest	"peer node start"	13 minutes ago	Up 12 minutes	0.0.0.0:10051->10051/tcp	peer3.org1.example.com
bb4deea00130	hyperledger/fabric-orderer:latest	"orderer"	13 minutes ago	Up 13 minutes	0.0.0.0:8050->7050/tcp	orderer1.example.com
c33748502a05	hyperledger/fabric-ca:latest	"sh -c 'fabric-ca-se…"	13 minutes ago	Up 13 minutes	0.0.0.0:7054->7054/tcp	ca.org1.example.com
bfdb2b01db36	hyperledger/fabric-couchdb	"tini -- /docker-ent…"	13 minutes ago	Up 12 minutes	4369/tcp, 9100/tcp, 0.0.0.0:9984->5984/tcp	couchdb4
89f0955fbf8c	hyperledger/fabric-orderer:latest	"orderer"	13 minutes ago	Up 12 minutes	0.0.0.0:7050->7050/tcp	orderer.example.com
8809d19a23c0	hyperledger/fabric-orderer:latest	"orderer"	13 minutes ago	Up 12 minutes	0.0.0.0:11050->7050/tcp	orderer4.example.com
af7dc61c84a4	hyperledger/fabric-couchdb	"tini -- /docker-ent…"	13 minutes ago	Up 13 minutes	4369/tcp, 9100/tcp, 0.0.0.0:6984->5984/tcp	couchdb1
8bd4a88129b7	hyperledger/fabric-orderer:latest	"orderer"	13 minutes ago	Up 12 minutes	0.0.0.0:9050->7050/tcp	orderer2.example.com
4dcde6c7070f	hyperledger/fabric-orderer:latest	"orderer"	13 minutes ago	Up 13 minutes	0.0.0.0:12050->7050/tcp	orderer5.example.com
e0c4cef46359	hyperledger/fabric-couchdb	"tini -- /docker-ent…"	13 minutes ago	Up 13 minutes	4369/tcp, 9100/tcp, 0.0.0.0:5984->5984/tcp	couchdb0
1f5b166149da	hyperledger/fabric-couchdb	"tini -- /docker-ent…"	13 minutes ago	Up 13 minutes	4369/tcp, 9100/tcp, 0.0.0.0:8984->5984/tcp	couchdb3
3d1ec4d2764c	hyperledger/fabric-orderer:latest	"orderer"	13 minutes ago	Up 13 minutes	0.0.0.0:10050->7050/tcp	orderer3.example.com
fac80ea5170c	hyperledger/fabric-couchdb	"tini -- /docker-ent…"	13 minutes ago	Up 13 minutes	4369/tcp, 9100/tcp, 0.0.0.0:7984->5984/tcp	couchdb2

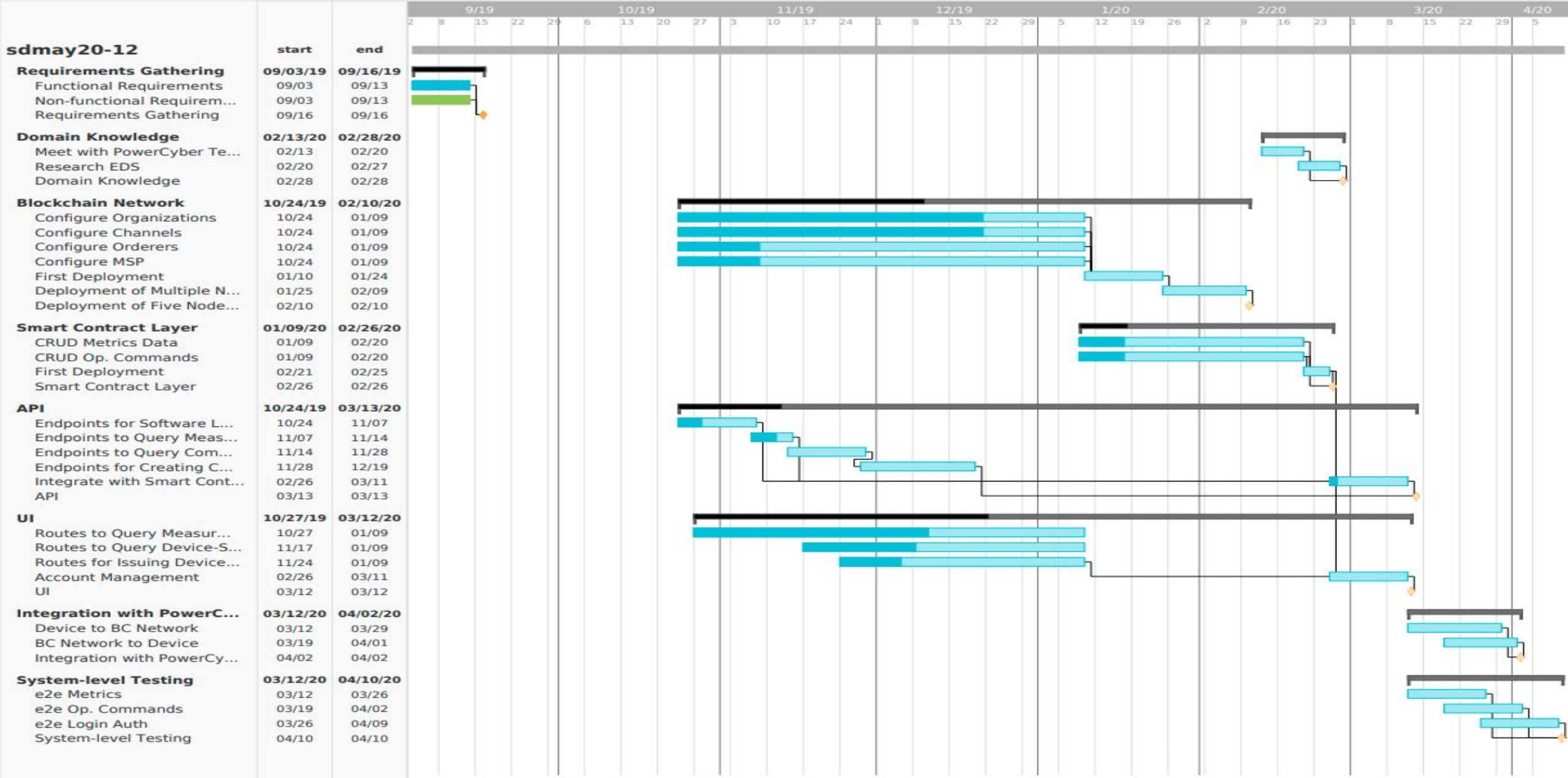
Prototype: API





Prototype: UI




sdmay20-12 start end
Requirements Gathering 09/03/19 09/16/19
Functional Requirements 09/03 09/13
Non-functional Requirem... 09/03 09/13
Requirements Gathering 09/16 09/16
Domain Knowledge 02/13/20 02/28/20
Meet with PowerCyber Te... 02/13 02/20
Research EDS 02/20 02/27
Domain Knowledge 02/28 02/28
Blockchain Network 10/24/19 02/10/20
Configure Organizations 10/24 01/09
Configure Channels 10/24 01/09
Configure Orderers 10/24 01/09
Configure MSP 10/24 01/09
First Deployment 01/10 01/24
Deployment of Multiple N... 01/25 02/09
Deployment of Five Node... 02/10 02/10
Smart Contract Layer 01/09/20 02/26/20
CRUD Metrics Data 01/09 02/20
CRUD Op. Commands 01/09 02/20
First Deployment 02/21 02/25
Smart Contract Layer 02/26 02/26
API 10/24/19 03/13/20
Endpoints for Software L... 10/24 11/07
Endpoints to Query Meas... 11/07 11/14
Endpoints to Query Com... 11/14 11/28
Endpoints for Creating C... 11/28 12/19
Integrate with Smart Cont... 02/26 03/11
API 03/13 03/13
UI 10/27/19 03/12/20
Routes to Query Measur... 10/27 01/09
Routes to Query Device-S... 11/17 01/09
Routes for Issuing Device... 11/24 01/09
Account Management 02/26 03/11
UI 03/12 03/12
Integration with PowerC... 03/12/20 04/02/20
Device to BC Network 03/12 03/29
BC Network to Device 03/19 04/01
Integration with PowerCy... 04/02 04/02
System-level Testing 03/12/20 04/10/20
e2e Metrics 03/12 03/26
e2e Op. Commands 03/19 04/02
e2e Login Auth 03/26 04/09
System-level Testing 04/10 04/10
9/19
10/19
11/19
12/19
1/20
2/20
3/20
4/20

Testing Hardware & Software

API

- Use Jest to send automated HTTP requests to API

Smart Contract

- Use Jest to assure ledger updates occur when executed

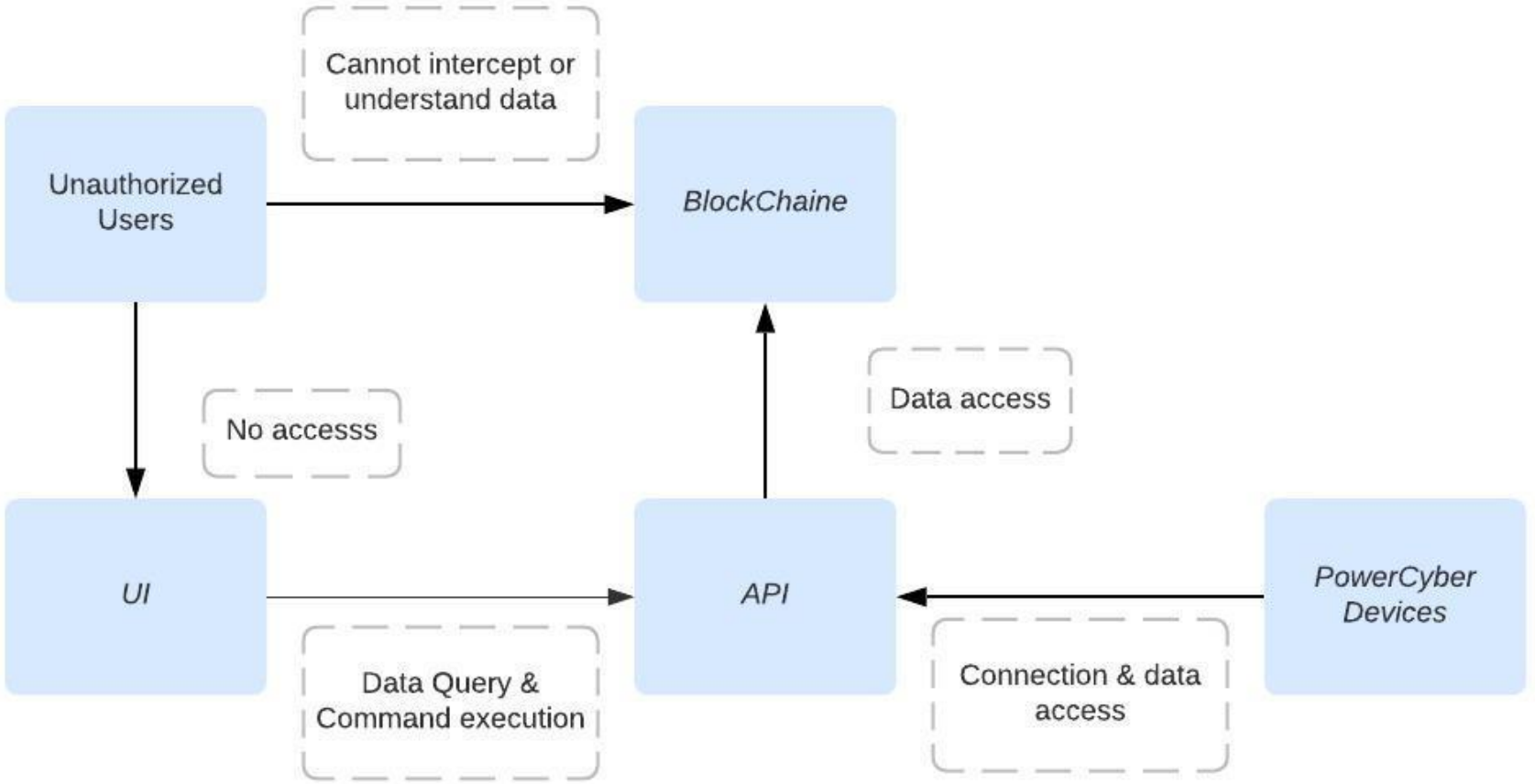
Blockchain Network

- Use Hyperledger Fabric CLI to send basic queries to nodes to test connect to network

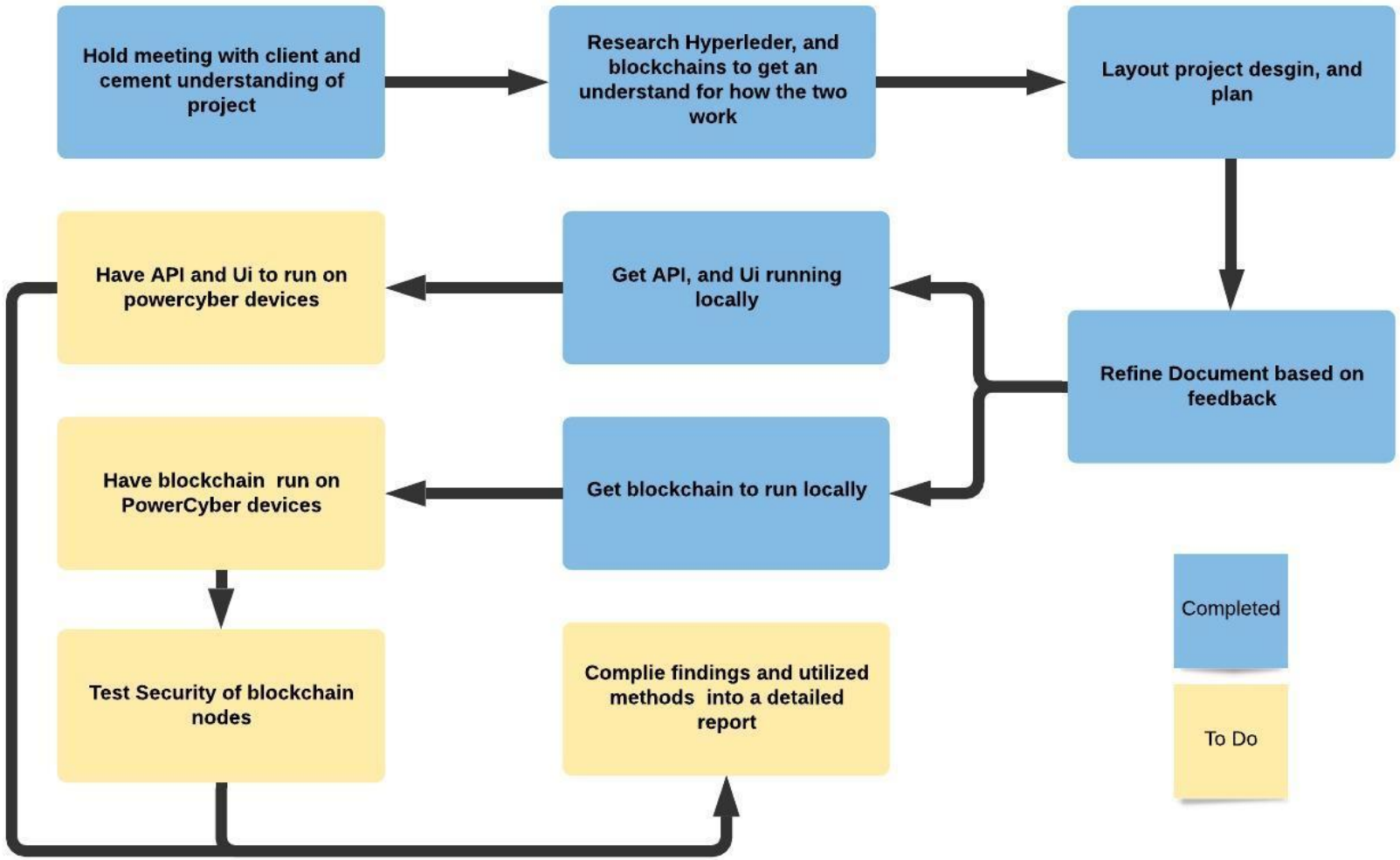
User Interface

- Jest will allow us use the DOM to test React components

End to End Testing



Overview


Hold meeting with client and cement understanding of project
Research Hyperleder, and blockchains to get an understand for how the two work
Layout project desgin, and plan
Have API and Ui to run on powercyber devices
Get API, and Ui running locally
Refine Document based on feedback
Have blockchain run on PowerCyber devices
Get blockchain to run locally
Test Security of blockchain nodes
Complie findings and utilized methods into a detailed report
Completed
To Do

Appendices

Technologies

CouchDB:	http://docs.couchdb.org/en/stable/
Express:	https://expressjs.com/en/4x/api.html
HyperLedger Fabric:	https://hyperledger-fabric.readthedocs.io/en/release-1.4/
HyperLedger Convector:	https://docs.covalentx.com/article/71-getting-started
Moesif Orign & CORS Changer:	https://chrome.google.com/webstore/detail/moesif-orign-cors-changer/digfbfaphojjndkpccljibejjbppifbc
PowerCyber Labs:	http://powercybersec.ece.iastate.edu/powercyber/welcome.php
Raft:	https://raft.github.io/
React.JS:	https://reactjs.org/docs/getting-started.html