CPR E / SE 492 BIWEEKLY STATUS REPORT 1

January 25 - February 8

Senior Design Team 15

Debugger and Visualizer for a Shared Sense of Time
on Batteryless Sensor Networks

Client/Advisor
Dr. Henry Duwe

Team Members
Adam Ford - Report Manager
Allan Juarez - Scribe
Maksym Nakonechnyy - Design Lead
Anthony Rosenhamer - Facilitator
Quentin Urbanowicz - Test Engineer
Riley Thoma - Project Manager

Biweekly Summary

Over the past two weeks, we were focused on getting everyone established on their subsection of the project. This involved setting up our working environments and becoming familiar with the existing code that we created last semester. We also discussed the interactions between the three major components (frontend, backend, and simulator), and we began writing code to get this communication working. In each subsection, we worked on basic functionality to better understand the libraries that we are using and continued to model the components of our system.

Accomplishments from the Past Two Weeks

- Backend Team (Adam and Allan)
  - We explored communications in ExpressJS using both sockets and Rest API protocols, as well as created the beginnings of a local database. Additionally, we spent time researching if a remote database would be worth having while developing, and concluded it was not worth the overhead of setup time and necessary server space.

- Frontend Team (Maksym and Riley)
    - We did research to find a new graphing library to display the network of nodes. Our current library is React-digraph and our two choices we'll be testing and picking from are React Flow and GoJS. We also met with Vishal to flesh out our requirements and get any new updates since last semester.
- Simulator Team (Anthony and Quentin)
    - We walked through a demo simulation that we built in the fall semester using the SimPy library that we are using for the project. We also experimented with the sockets library that we will be using for simulator to backend communication. In addition, we defined the events that may occur during a simulation, including the data fields that are used for sending information to the backend or a log file. Additionally, work was done to generalize node lifecycle logic to enable arbitrary, user-defined functionality and experiments were performed for introducing pseudo-random variation into simulation parameters for enhanced realism.

Pending Issues

From our discussion with Vishal, we need to clarify with Dr. Duwe if we need to include a special node to communicate with all other nodes (sniffer node) in our simulation. We also need to clarify what time synchronization algorithm to use in the simulation.

We have to ask Dr. Duwe and Vishal about the time precision for a simulation. We need to find out what the resolution is for timing whether it is 1 microsecond, 1 nanosecond, or something else.

Individual Contributions

Name	Individual Contributions	New Hours (last 2 wks)	Total Hours
Adam Ford	Set up local Mongo database and experimented with ExpressJS inserting/altering records Researched if a remote database is feasible or worth the effort for use during development	13	13
Allan Juarez	Looked over the simulators code and design of a websocket and researched how it would connect to ExpressJS. Set up and experimented with get and post api in ExpressJS	12.5	12.5
Maksym Nakonechnyy	Researched and experimented with graph libraries in React that could be used for network visualization. Updated frontend requirements	12	12
Anthony Rosenhamer	Experimented with sockets library for simulator to backend communication Created socket demo to work on Linux and Windows with a mocked version of the backend Create basic events with appropriate fields that can be used to send information to the backend or log	13	13
Quentin Urbanowicz	Defined events based on simulator node lifecycle logic for use with logging and communication with backend Worked on abstracting simulator node lifecycle logic to allow for arbitrary, user-definable node behavior Experimented with methods for introducing variability into node behavior for more realistic simulations	12	12
Riley Thoma	Researched and experimented with graph libraries in React that could be used for network visualization.	12	12

Plans for the Next Two Weeks

- Adam Ford - backend development
  - Set up a script that instantiates the Mongo Database on a system, to ensure same practices by team members
  - Continue working with MongoDB functions in ExpressJs
- Allan Juarez - backend development
  - Try to get the communication working from the backend and simulator and then check the communication between the backend and frontend and make sure that is still working
- Maksym Nakonechnyy - frontend development
  - Have a discussion with Riley to decide which graph library we will use for network visualization.
  - Learn more about the library.
  - Start developing the first panel.
- Anthony Rosenhamer - simulator development
  - Continue to work on message creation and formatting for communication with the backend
  - Incorporate events and logging into the code for the sniffer and nodes
- Quentin Urbanowicz - simulator development
  - Continue work on abstracting node behavior to enable user-defined functionality
  - Refactor node logic to utilize events and logging
  - Begin implementing functionality for pseudo-random behavior variations
- Riley Thoma - frontend development
  - Pick graphing library and start using it to develop the frontend
  - Work on panels and connectivity to graph; once the graph is working

---

Summary of Advisor Meetings

January 29th:

We met with Dr. Duwe and Vishal to discuss our expectations about the workflow this semester.

February 5th:

The frontend team met with Vishal to cover frontend requirements and changes, and the format of the configuration file.