Project Status Report - 25 September 2023

Table of Contents

1. Executive Summary

This status report provides an in-depth overview of the progress made in the development of our Node.js OPC UA Server and React Client project. We detail the achievements, challenges faced, and the roadmap for future enhancements.

2. Project Overview

Our project involves building a Node.js server powered by `node-opcua` and a React client for interfacing with the server. It focuses on secure data retrieval from an OPC UA server, user

management, and authentication, reinforced with OTP verification.

3. Project Progress

3.1. Backend Development

3.1.1. OPC UA Data Retrieval

- Successfully implemented data retrieval from OPC UA server.
- Overcame challenges related to excessive requests by integrating the subscription model.


```
{
  message: { name: 'L1T1', value: 18.556779861450195, nodeId: 23, unit: '°C' }
}
{
  message: {
    name: 'L0_Q1_F4',
    value: 1.7700326442718506,
    nodeId: 33,
    unit: 'm/s'
  }
}
{
  message: {
    name: 'L0_Q1_COND',
    value: 278.53826904296875,
    nodeId: 34,
    unit: 'µS/cm'
  }
}
{
  message: {
    name: 'L0_Q1_F2',
    value: 52.107601165771484,
    nodeId: 31,
    unit: 'kg/min'
  }
}
{
  message: {
    name: 'L0_Q1_F1',
    value: 52.107601165771484,
    nodeId: 30,
    unit: 'l/min'
  }
}
{
  message: { name: 'L2T1', value: 18.613239288330078, nodeId: 21, unit: '°C' }
}
{
```


3.1.2. User Management and Security

- Implemented user authentication with login and signup functionality.

- Added privilege escalation via API.
- Enhanced security through OTP verification during signup.

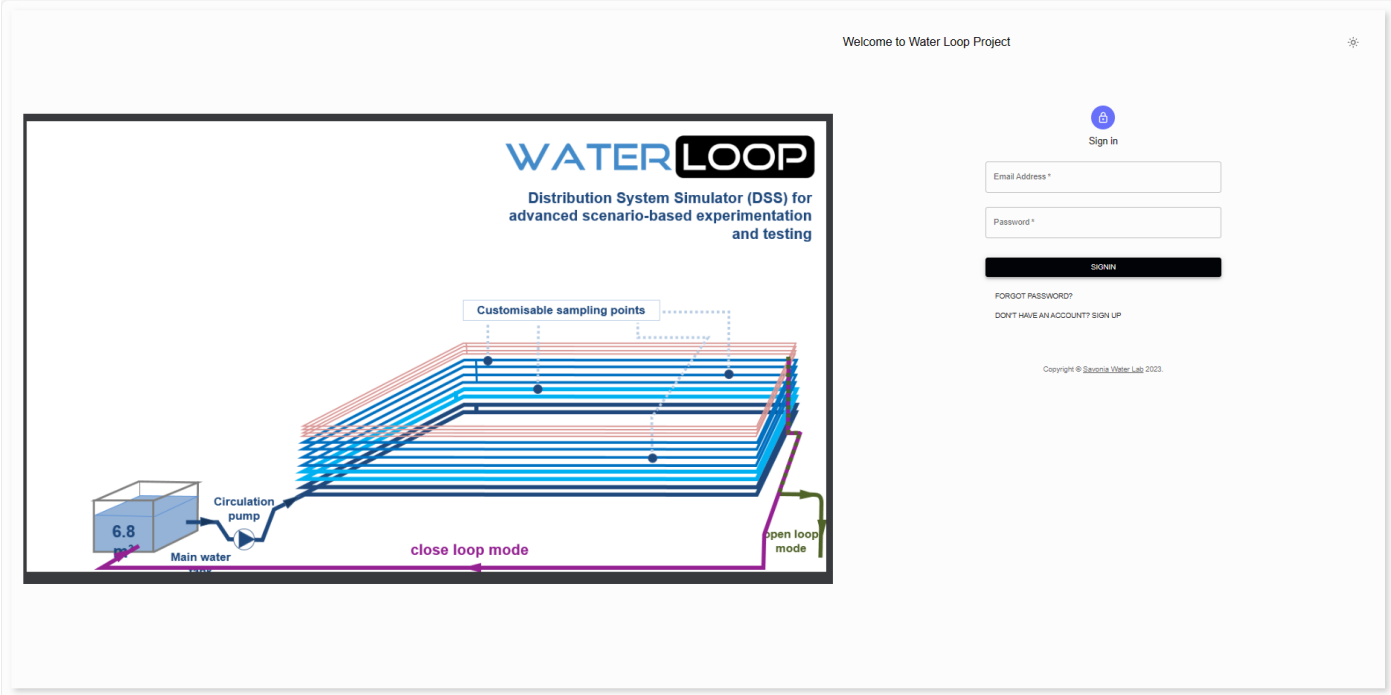
3.1.3. MongoDB Cloud Database Integration

- Integrated MongoDB Cloud Database for efficient data storage and retrieval.
- Designed database schemas to accommodate project requirements.

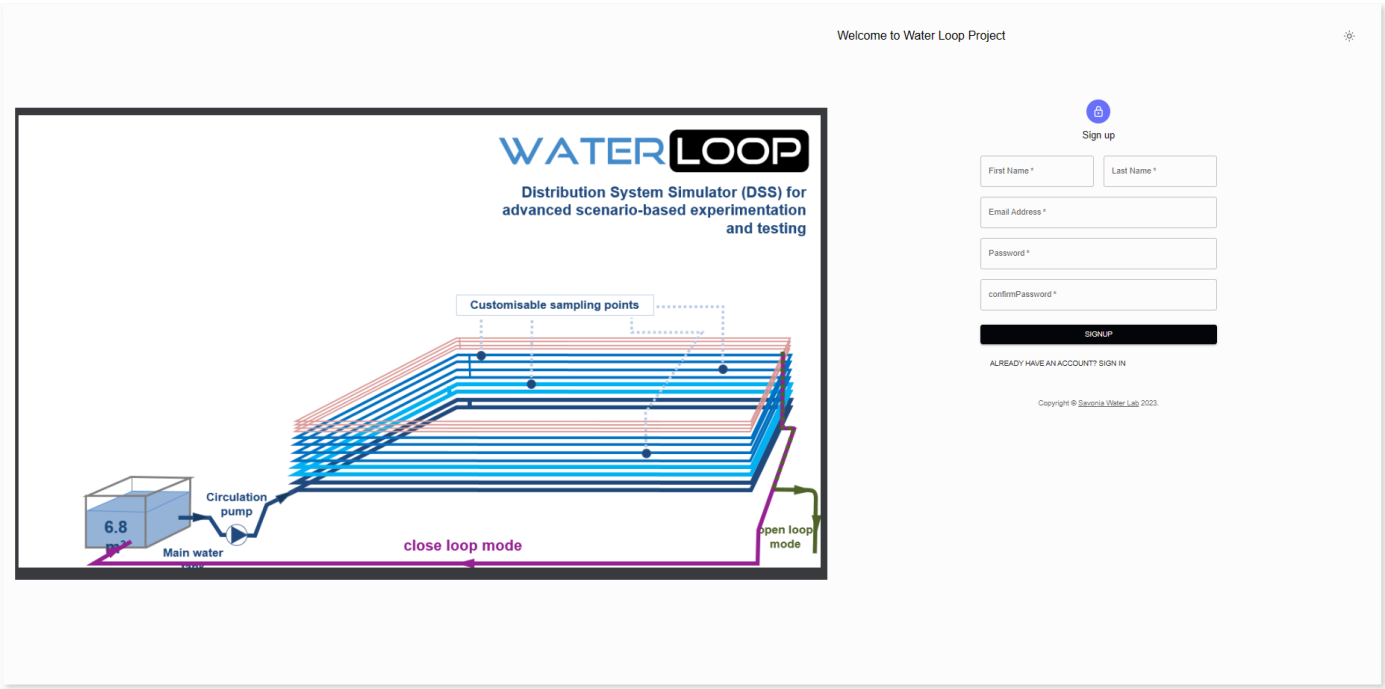
3.2. Frontend Development

3.2.1. User Interface

- Designed an intuitive and user-friendly interface for the React client.
- Login Page

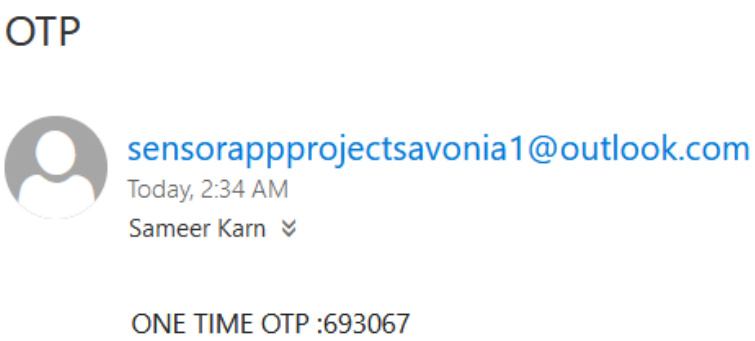


- Sign Up Page

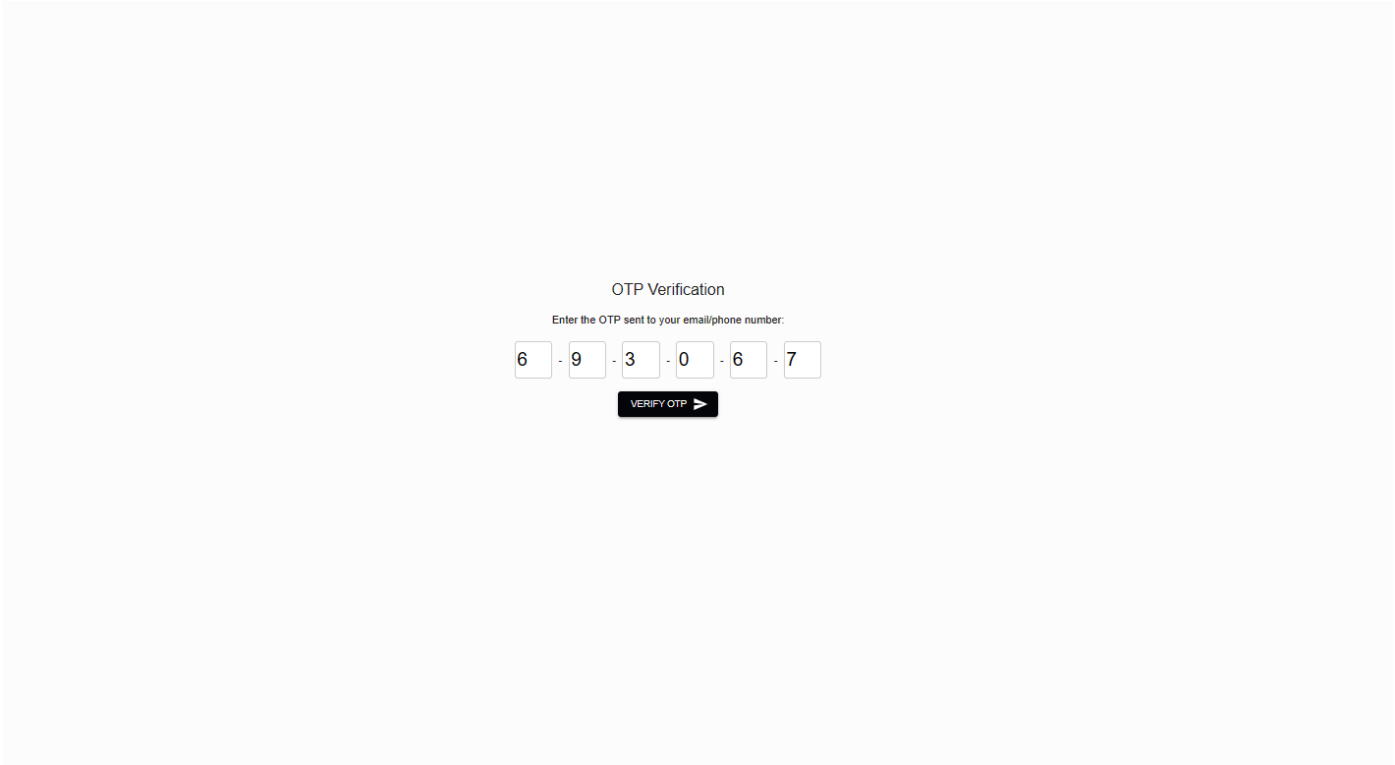


3.2.2. User Authentication

- Users can securely log in or sign up using their email.
- OTP verification enhances account security.
- Email OTP ss



- OTP page



3.2.3. Dashboard

- Created a dashboard that currently displays dummy templates of widgets for user interaction.
- Dashboard page

4. Key Achievements

- Successful integration of `node-opcua` for efficient data retrieval.
- Robust user management system with secure OTP-based verification.
- Intuitive React client interface for user interaction.

- MongoDB Cloud Database integration for scalable and reliable data storage.

5. Challenges and Mitigations

- **Challenge**: Excessive OPC UA requests.
  - **Mitigation**: Implemented the subscription model to optimize data retrieval.

6. Future Enhancements

We have ambitious plans for the project's future, including:

- Real-time data visualization.
- Advanced security features.
- Customizable user dashboards.

7. Conclusion

We are committed to delivering a secure and user-friendly solution for interfacing with OPC UA servers. Your feedback and contributions are invaluable as we continue to improve and expand this project.

8. Acknowledgments

We would like to express our gratitude to all team members and contributors who have dedicated their time and expertise to this project. Your efforts are greatly appreciated.

[Rajeev.Kanth@savonia.fi]
[Patryk.Wojtowicz@savonia.fi]

Thank you for your continued support as we work towards creating a cutting-edge Node.js OPC UA Server and React Client. For further inquiries or feedback, please contact us at [sameer.karn@edu.savonia.fi].

Best regards,

[Sameer Karn]
[Software Developer Intern]
[25/09/2023]