December 14th, 2021
Dr. Mark Lanthier
Computer Science Apartment
Carleton University

Dear Mark Lanthier,

From September 3, 2021, to December 17, 2021, I had my last CO-OP term as a Web Developer in part of the Development - NRC-P  team at Nokia. My Coop course number is COMP3999A. My supervisor is Ehsan Rezaaifar who is the software design manager in NRC-P.

The following full CO-OP work term report follows the guidelines set from the Co-op Work Term Report Guidelines. The report is my own original work and has not been previously submitted for credit

Sincerely,
Zeye Gu
101036562

Name: Zeye Gu
Student Number: 101036562
zeyegu@cmail.carleton.ca
COOP 3999A

Web Developer - IP SDN Co-op (DB6)
Fourth work term report
Full report
Employer: Nokia
Supervisor: Ehsan Rezaaifar

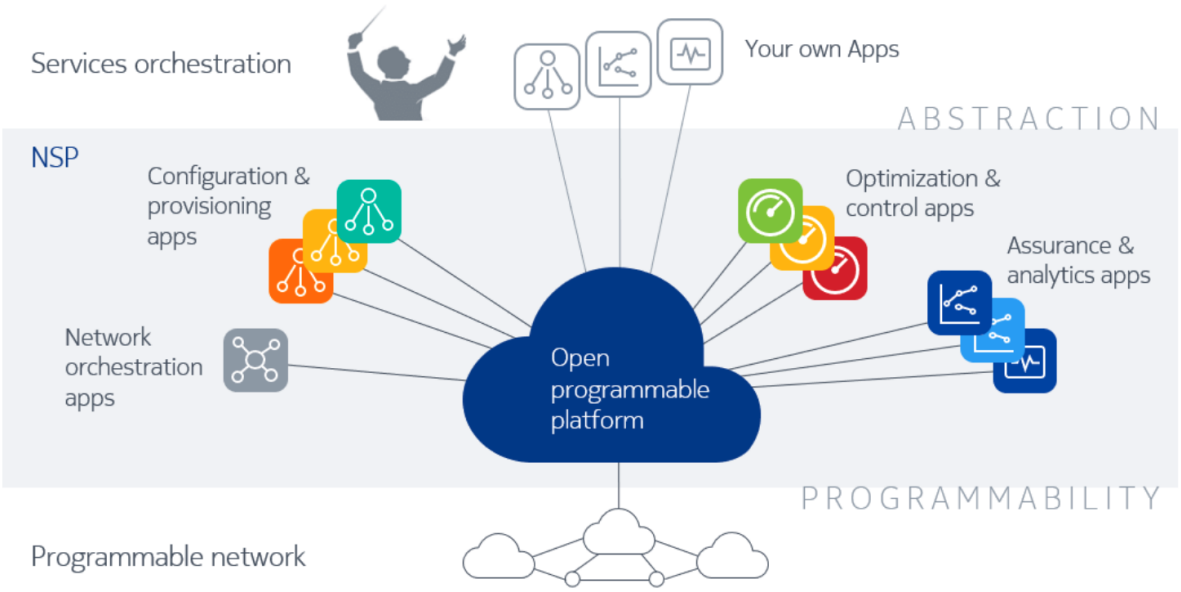


Figure 1. NSP Services Platform

## Executive summary

September 3, 2021, to December 17, 2021,  I am having my last Co-op term as a Web Developer in part of the Development - NRC-P team at Nokia

I have worked at Nokia for one year up to  now. This year, I am working on a Network Services Platform (NSP). My main task is to update and maintain the two apps which are called IP/MPLS Optimization(NRC-P) and IP/MPLS Simulation(NRC-P simulator) respectively. As I got familiar with these two apps, I started to do some bug fixes for the other three projects which are Nokia-React-Components, Nokia-React-SchemaForm, and Nokia-React-Table respectively.

## Acknowledgments

None of my achievements is possible without my supervisor’s trust and colleagues’ support. I would like to send a sincere appreciation to Nazia, Maryam, and Sam who gave me a lot of support during the work. I am also grateful to Ehsan for giving me this opportunity to work in our great team.

## Table of Contents

# 1. Introduction

## 1.1 Organizational context

Nokia is a global leader in the technologies that connect people and things. With state-of-the-art software, hardware and services for any type of network, Nokia is uniquely positioned to help communication service providers, governments, and large enterprises deliver on the promise of 5G, the Cloud and the Internet of Things.  Serving customers throughout in over 100 countries, our research scientists and engineers continue to invent and accelerate new technologies that will increasingly transform the way people and things communicate and connect.

Network Service Platform (NSP) is the app that I am working on. NSP helps people automate their IP and optical networks to simplify their operations, respond quickly to fast-changing demand, get the most from your resources and ensure maximum service performance and reliability.

# 2.  Work experience

## 2.1 Nature of work

Developing and maintaining the NSP-NRCP app is my main responsibility. This app uses packages that are built by other teams. The packages such as Nokia-React-Components, Nokia-React-SchemaForm, and Nokia-React-Table are the packages that I have worked on. The issues that show on the Graphic User Interface of NSP-NRCP app may need to be fixed in the packages. So when QA sends a PTS to me, I need to determine where the issue truly is and if it is not in the NSP-NRCP itself I need to decide whether to solve the issue in the package and send a merge request to the corresponding team, or I can reproduce the issue on the Showbook and then raise another PTS to the corresponding team. The Showbook is a unified Nokia software portfolio which contains a lot of examples of the React components that the apps are using.

## 2.2 Experience

My Co-op terms at Nokia were full of new learning and challenges. The experience at Nokia expanded my understanding of Javascript, React, Redux, and Git, and it also made me familiar with Jira and Gitlab.

My working period at Nokia is from 9:30 am to 5:30 pm. After I connect to the Nokia VPN, the next thing I do is to choose the highest priority task to start as the first task of my day. Most of the time, the tasks are not sent to me until I was done with the previous one. So the ability to work with multiple tasks is important. There are three or four meetings per week in my team, and I usually list the questions I would like to ask and present my demo during these meetings.

Upgrading a deprecated component to a new component is the easiest type of task. First I need to understand the props for both components. After I correctly replace the props, the GUI will be different in most cases. However changing the css style is not the highest priority, but ensuring the functionality runs successfully is. In the network tab of the Chrome Developer Tools, I need to make sure the payload of the POST call is the same before I upgrade and after I upgrade the component so that the data sent to the server is still the same. Then I need to check local functionalities as well. For example when I enter text into a field which is called ‘Network ID’, the local validation should pop up in red. After I ensure all the functionalities work correctly, to modify the css style is my next step of work.

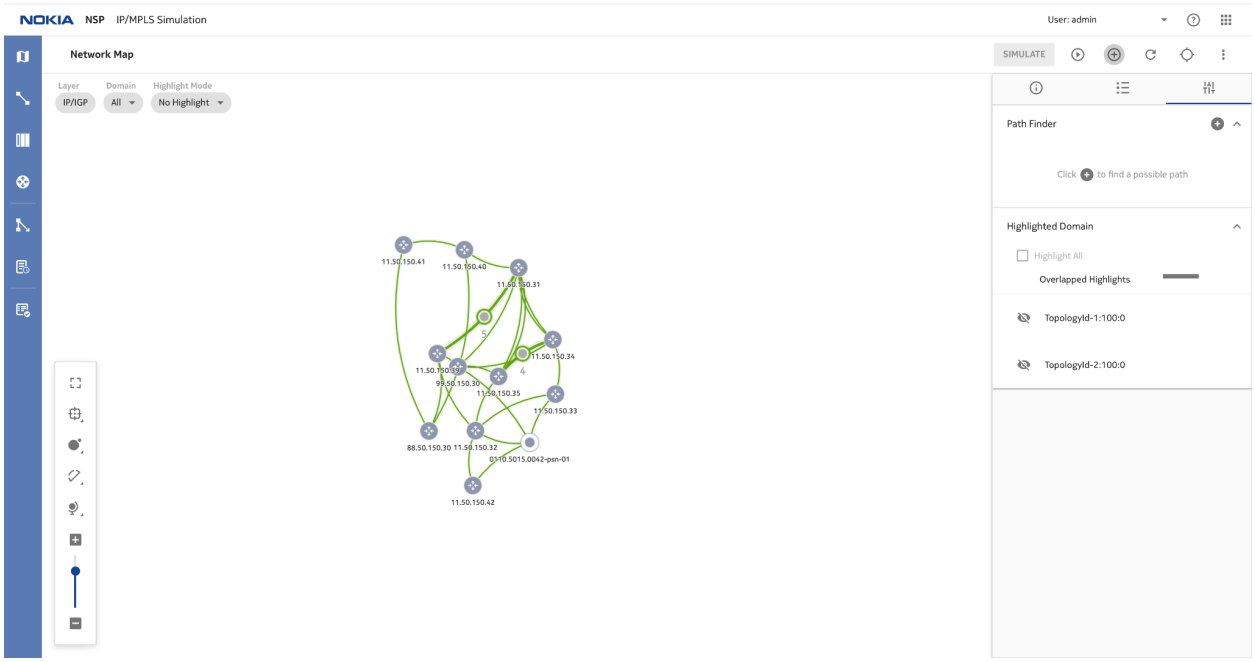


Figure 2. NSP IP/MPLS Simulation - Network Map

When I create the merge request, making screen recordings or screen shots for the upgraded component is a necessary step of documentation so that my colleagues would spend less time reviewing my code. More importantly, in case my colleagues find something wrong with the component which I upgraded later. The screen shots and recordings would be a convenient clue to debug the code.

I have been working on React-Schema-Form for about half a year. It is based on a third party library which is called Fomik. It allows programmers to spend less time coding the forms. However programmers need to spend more time learning how to use the React-Schema-Form. I build a form by passing a JSON object into the Schema Form component. By digging into the React-Schema-Form, I could understand the component faster and deeper. It still took me weeks to get familiar with the package. However, once I understood what each property in the JSON object does, and got familiar with common callbacks such as onChange(), and onSubmit(), the developer process was much faster. I've completed five form upgrades so far.

Although issues were always discovered in our app first they might actually come from the packages that I imported. After a series of bug tracing, if that was the case, I needed to either raise a PTS to the corresponding team or I could solve the problem by myself. First I need to ensure that the problem was fixed in the Showbook. Then I needed to do gradle build and npm pack the package to get a tgz file of it so that I can test it by using npm install the tgz file in our app to make sure the component that I fixed also works correctly in our app. After that I could submit a merge request to another team.

During last two weeks, I was working on a component which was called MultiSelectInput. It was a drop-down menu, and there were multiple checkboxes as the options of the menu. A hint text '0 Item(s) selected ' would change according to the number of the options that were chosen by the user. There were two main issues for this component, the first one was that the hint text is not changed at all even when users select the checkboxes, and another one was that the data payload provided a list of the labels that were selected, rather than the indices or name fields, so data would need to be remapped. After my investigations, The variable selectedItems was only modified in the constructor, so I reworked the onChange() function of it and used names as values with higher priority than using labels. After my fix, the user could select the checkboxes and the hint text was shown correctly on the GUI.

### 2.3 Challenges and solutions

Sometimes my coding skills cannot solve every issue that I have in my work. There was a pre-push process which was called GIT HOOKS that I needed to follow when I pushed the code to a repo from another team. Once this pre-push process was updated. It would generate a new format of changelogs file. Although I followed exactly the description of the GIT HOOKS, the changelogs file was still in an old format. Unfortunately, I was the first one who pushed the code after the pre-push process got updated. The solution to solve this problem needs time,

patience, and communications. I asked different people several times. I finally found a colleague who had a similar issue. He suggested me to delete the whole npm module, and to npm install again. The solution finally worked.

Since there were situations that were similar to what I mentioned above. The ability to work on multiple tasks was essential. I needed to continue with other tasks when I was waiting for responses from the other colleagues.

I used the git stash command when I wanted to clean and save the current state of a branch without actually pushing it to the remote branch. In my current work term I already get familiar with the git commands since I am using it every day at work.

## 3. Reflections on Work Experience

### 3.1 Contributions

I have contributed to five projects. They are IP/MPLS Optimization, IP/MPLS Simulation, Nokia-React-Components, Nokia-React-SchemaForm, and Nokia-React-Table respectively. I have fixed and raised many PTSs. I also collaborate and communicate with colleagues from other teams. The following figure is a summary of issues, merge requests, pushes, and comments. I only push the code when I complete the PTSs. I started to learn React-Schema-Form in July so there were no activities in that month.

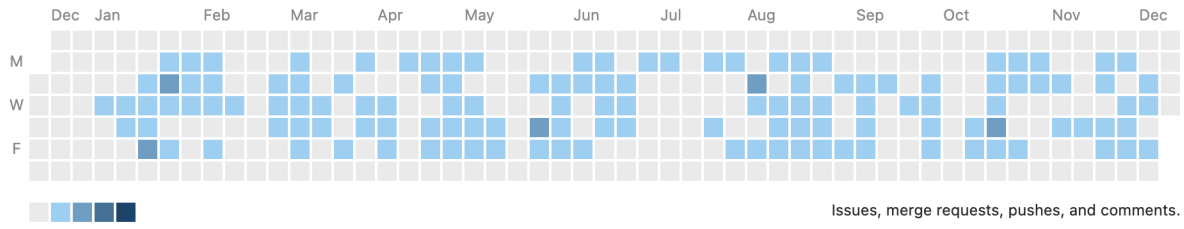


Figure 3. Activity of issues, merge request, pushes, and comments

### 3.2 Relation to academic studies

Programming as a Web Developer at Nokia is different from Programming in academic studies. When I was taking COMP 2406 Fundamentals of Web Applications or even when I was working on my own projects, the packages that I imported did not have any issue in most cases. Even if there were issues, I could always find workarounds or solutions on Google. However I need to spend plenty of time reading the code from the other teams, and I have already sent a

couple of merge requests to them. In addition, I need to debug the code by setting the break points in the sources tab of the Chrome Developer Tools to trace the code line by line at Nokia. While when I was working on an academic project, it was enough to debug the code by printing out the variables that I wanted. In my opinion, the algorithms part of school projects are more difficult, but the overall amount of the code is much less than the Nokia projects. Working at Nokia was a valuable experience.

### 3.3 Career development

I have been working at Nokia as a web developer for a whole year. During the three work terms at Nokia, I have developed my debugging skills, developing skills, communication skills, and the ability to work multiple tasks at the same time. Academic studies only provided basic programming skills. There are more knowledge and competence to learn as a developer.

Front-end engineer will be one of my career plans in the future. I also decide to build a web app as my Honours project in the next semester.

## 4. Summary

This is my last Co-op term. I have practiced and improved my Javascript, React, and debug skills. What's more, the communication with colleagues in daily work has made me self-driven and self-motivated, which are key capabilities of working in a large company. I truly believe the experience of working at Nokia is significant and beneficial in my career.

## 5. References

Network Services Platform Automate, manage and control IP and optical networks https://www.nokia.com/networks/solutions/network-services-platform/ [Dec 14, 2021]