Enhancement of ONTAP Rest API UI

Noor Buchi, Teona Bagashvili, Christian Lussier, Kobe Coleman

Teona Bagashvili

- About Me:
  - Allegheny College Junior, Computer Science Major & Dance and Movement Studies minor

- Why I chose this Project:
  - Get industry experience in software engineering
  - Improve my web development skills.

Noor Buchi

- About Me:
  - Allegheny College Junior, Computer Science Major, Political Science Minor

- Why I chose this Project:
  - Get hands on experience in software engineering using industry standards and collaborate with a team and develop new skills

Kobe Coleman

- About Me:
  - Allegheny College Sophomore, Computer Science Major & Music Theory Minor

- Why I chose this Project:
  - Wanted to go deeper into React applications and how the different components in a visualizer interacted with each other
  - Have hands on experience with open source projects

Christian Lussier

- About me:
  - Allegheny College Senior -- Computer Science Major & Economics Minor

- Why I chose this Project:
  - I wanted to gain industry experience and apply what I learned in the classroom.
  - Wanted to become more skilled in developing with Javascript, CSS, and REACT

Outline

Overall Project Goals and Motivations

- **Main Goal:** To enhance the user experience in NetApp’s REST API visualization
- Motivating issues:
  - The current visualizer, Swagger UI, is not user friendly
  - No way to find a specific endpoint/model/parameter -- lots of scrolling!
  - No automated version tracking
- Project goals:
  - Implement a deep searching feature
  - Automate the version tracking process

Team Organization & Communication

- Slack, Zoom, and Google Meets for constant communication
- Git workflow using Github and it's features (project board, pull requests)
- AGILE development practices -- one weekly sprint meeting with the team from NetApp (Sami and Anuradha)
  - Discussed our progress through each week
  - Created project board stories
  - Identified completed and remaining tasks
  - Solved technical difficulties on setting up a Javascript environment and working with tools such as React

Technology Overview

Swagger™

Supported by SMARTBEAR

HTML

React JS

CSS

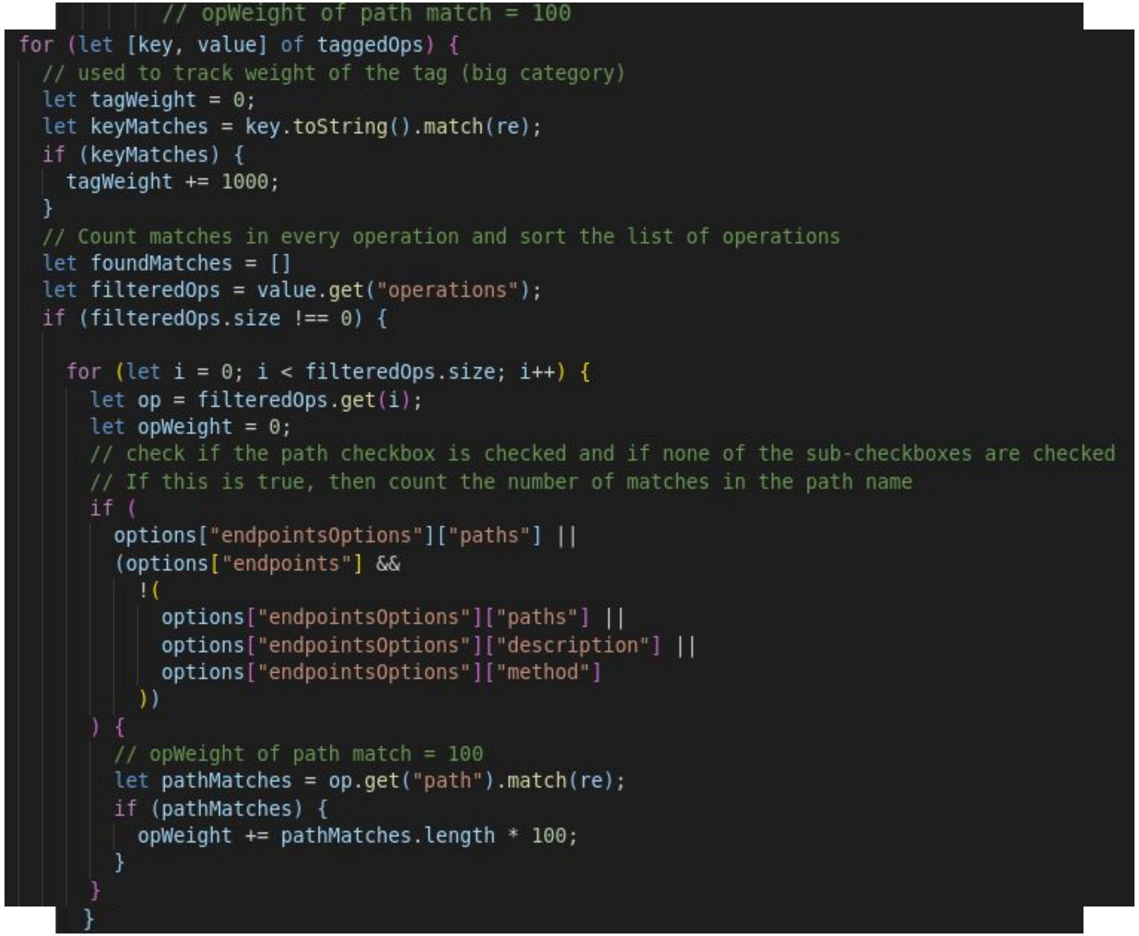
Midterm Status Review

- Enabled basic version of deep-search functionality for the operations based on:
  - Path name
  - Name of the tag
  - Description
- Ranked the search results based on:
  - Number of regular expression matches to the phrase being searched
  - Strength of those matches

Post-Midterm Work

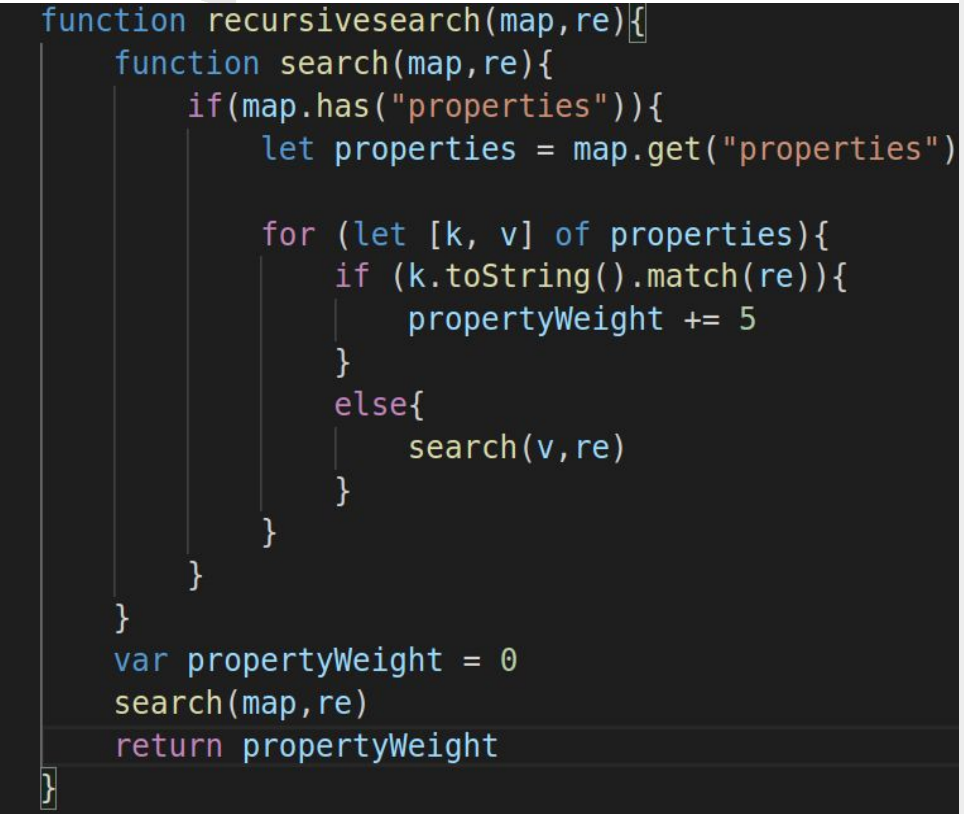
- Looked into our algorithm efficiency and possible improvements
  - Only run new search queries when ‘enter’ is pressed to avoid running search after typing a new character
  - Explored external libraries that could help in increasing efficiency
  - Bug fixes
- Expanded deep-search functionality to include the models
  - Based on model and property names
  - Similar ranking conventions as operations searching
- Implemented automatic version tracking
- Allow for more user flexibility via parameterized searching
  - Achieved using radio buttons and checkboxes

Operations Filter Improvements

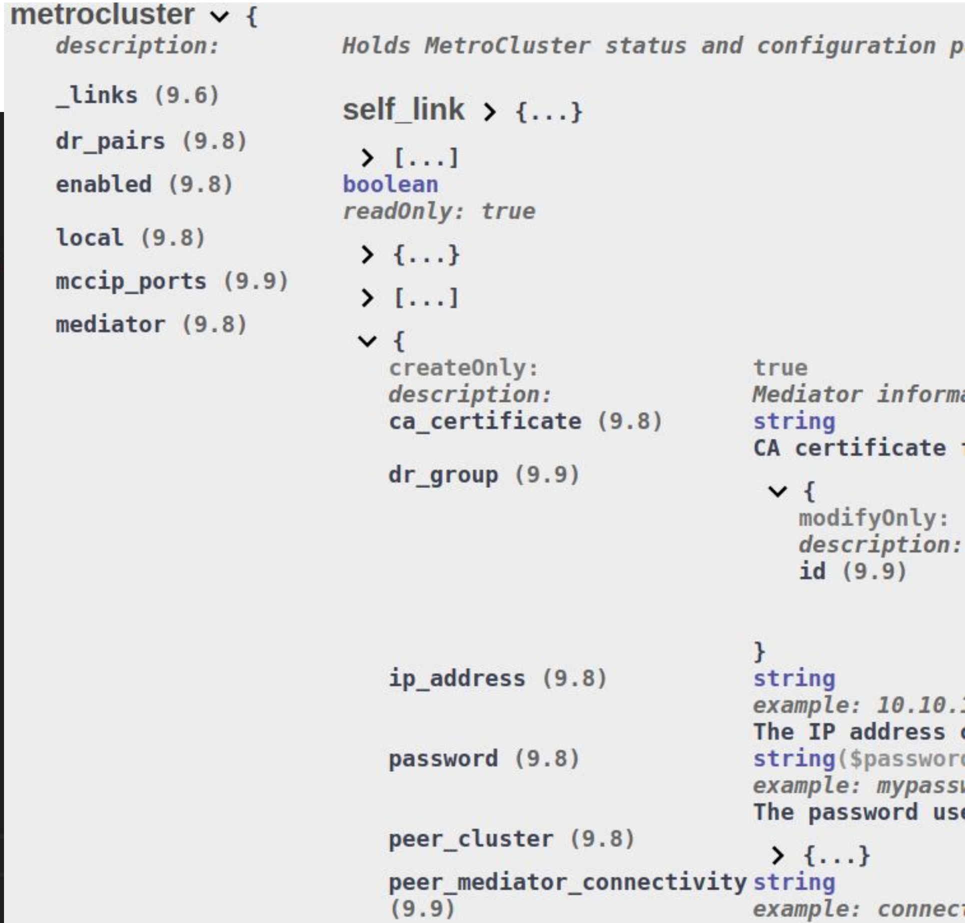


```
        // opWeight of path match = 100
for (let [key, value] of taggedOps) {
  // used to track weight of the tag (big category)
  let tagWeight = 0;
  let keyMatches = key.toString().match(re);
  if (keyMatches) {
    tagWeight += 1000;
  }
  // Count matches in every operation and sort the list of operations
  let foundMatches = []
  let filteredOps = value.get("operations");
  if (filteredOps.size !== 0) {

    for (let i = 0; i < filteredOps.size; i++) {
      let op = filteredOps.get(i);
      let opWeight = 0;
      // check if the path checkbox is checked and if none of the sub-checkboxes are checked
      // If this is true, then count the number of matches in the path name
      if (
        options["endpointsOptions"]["paths"] ||
        (options["endpoints"] &&
          !(
            options["endpointsOptions"]["paths"] ||
            options["endpointsOptions"]["description"] ||
            options["endpointsOptions"]["method"]
          ))
      ) {
        // opWeight of path match = 100
        let pathMatches = op.get("path").match(re);
        if (pathMatches) {
          opWeight += pathMatches.length * 100;
        }
      }
    }
```

Models Filter

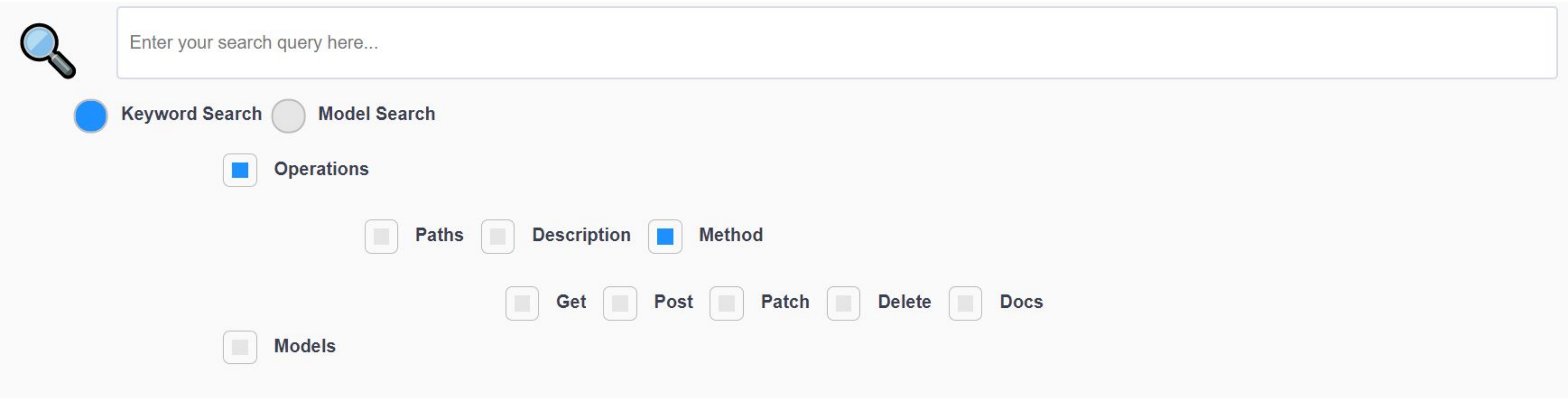


```
function recursivesearch(map,re){
    function search(map,re){
        if(map.has("properties")){
            let properties = map.get("properties")

            for (let [k, v] of properties){
                if (k.toString().match(re)){
                    propertyWeight += 5
                }
                else{
                    search(v,re)
                }
            }
        }
    }
    var propertyWeight = 0
    search(map,re)
    return propertyWeight
}
```



```
metrocluster ⌄ {
   description:        Holds MetroCluster status and configuration p

   _links (9.6)        self_link › {...}
   dr_pairs (9.8)       › [...]
   enabled (9.8)       boolean
                       readOnly: true
   local (9.8)          › {...}
   mccip_ports (9.9)    › [...]
   mediator (9.8)       ⌄ {
                          createOnly:                    true
                          description:                   Mediator informa
                          ca_certificate (9.8)           string
                                                         CA certificate
                          dr_group (9.9)                  ⌄ {
                                                            modifyOnly:
                                                            description:
                                                            id (9.9)

                                                         }
                          ip_address (9.8)               string
                                                         example: 10.10.
                                                         The IP address
                          password (9.8)                 string($passwor
                                                         example: mypass
                                                         The password us
                          peer_cluster (9.8)              › {...}
                          peer_mediator_connectivity     string
                          (9.9)                          example: connec
```

Parameterized Search - UI Enhancements


Enter your search query here...

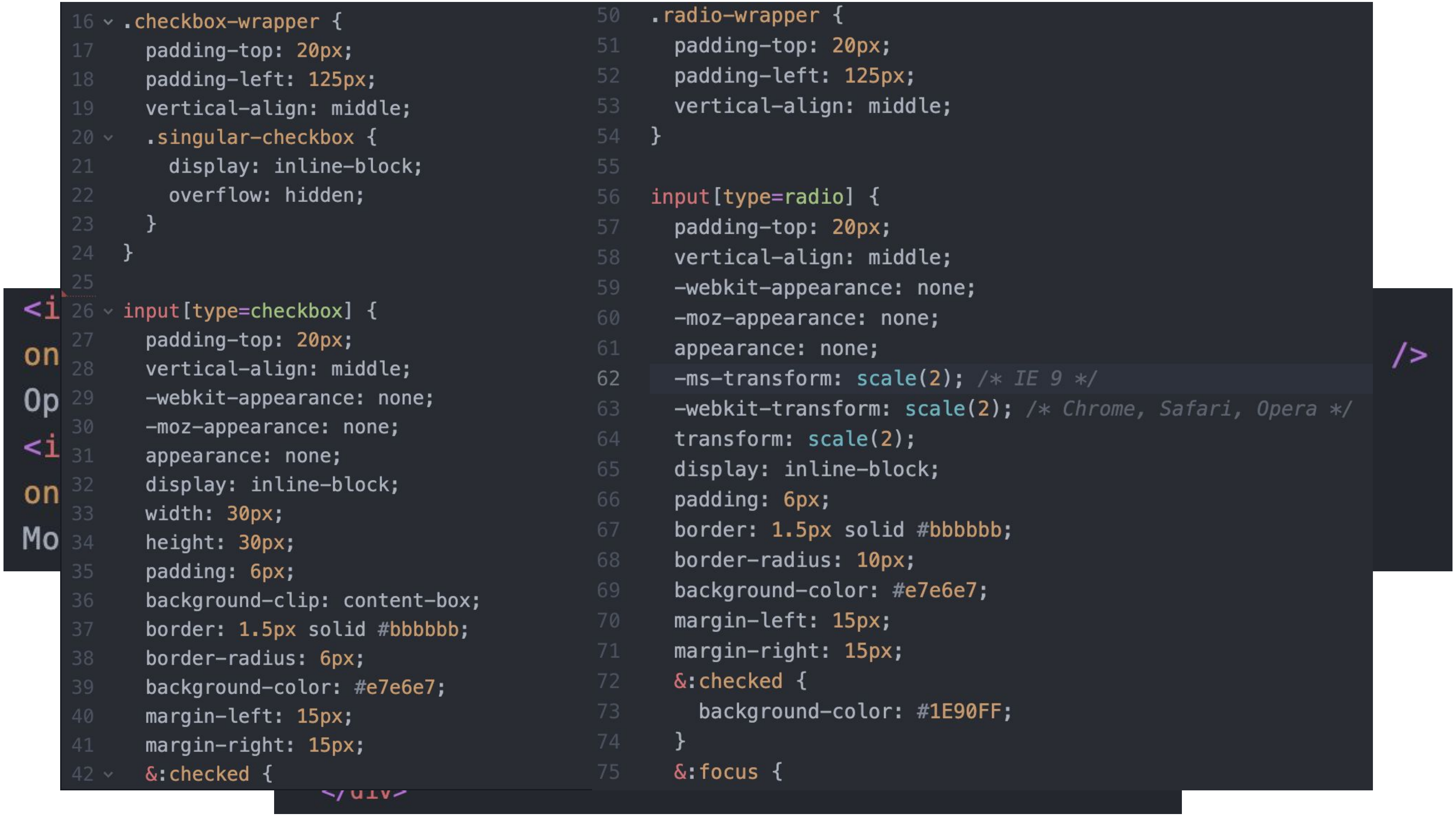
[✓ Keyword Search] Model Search

[✓ Operations]

Paths Description [✓ Method]

Get Post Patch Delete Docs

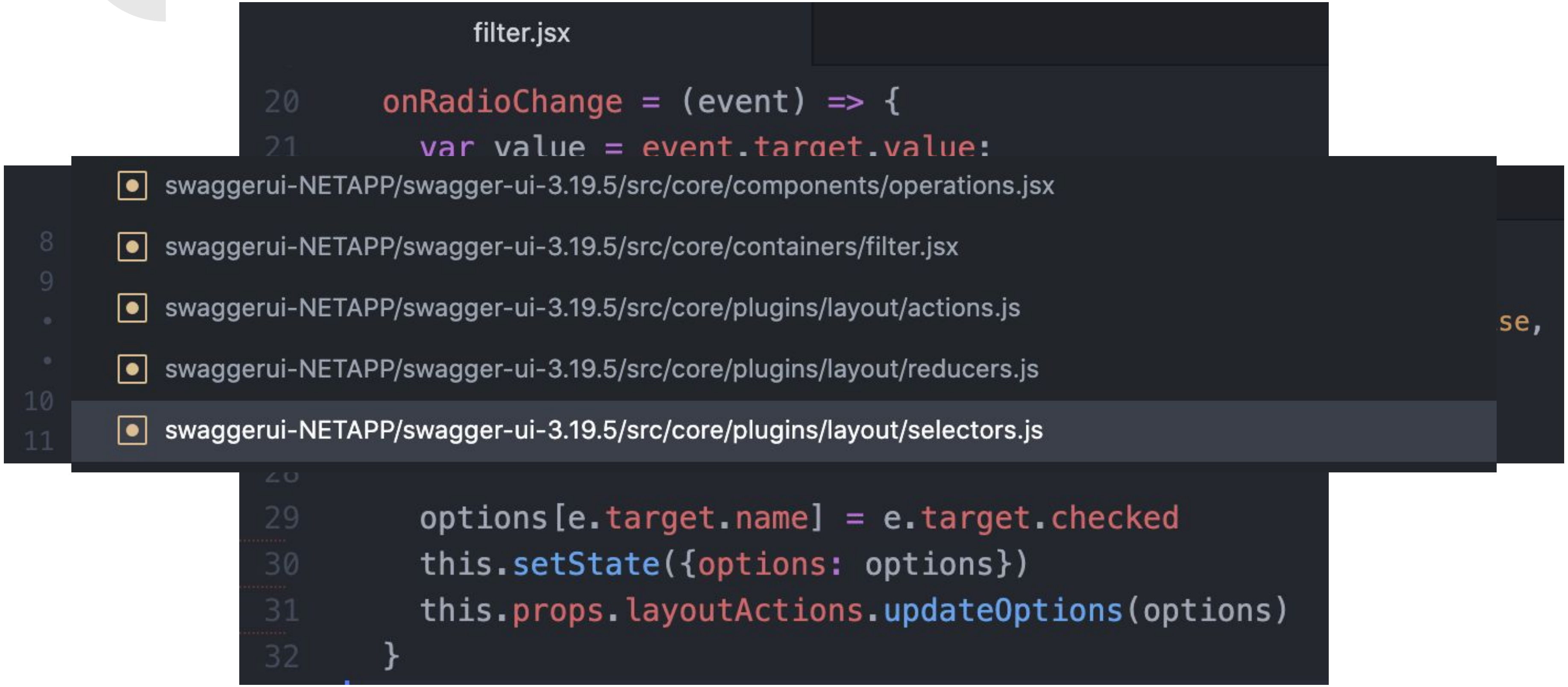
Models



```scss
.checkbox-wrapper {
  padding-top: 20px;
  padding-left: 125px;
  vertical-align: middle;
  .singular-checkbox {
    display: inline-block;
    overflow: hidden;
  }
}

input[type=checkbox] {
  padding-top: 20px;
  vertical-align: middle;
  -webkit-appearance: none;
  -moz-appearance: none;
  appearance: none;
  display: inline-block;
  width: 30px;
  height: 30px;
  padding: 6px;
  background-clip: content-box;
  border: 1.5px solid #bbbbbb;
  border-radius: 6px;
  background-color: #e7e6e7;
  margin-left: 15px;
  margin-right: 15px;
  &:checked {
```

```scss
.radio-wrapper {
  padding-top: 20px;
  padding-left: 125px;
  vertical-align: middle;
}

input[type=radio] {
  padding-top: 20px;
  vertical-align: middle;
  -webkit-appearance: none;
  -moz-appearance: none;
  appearance: none;
  -ms-transform: scale(2); /* IE 9 */
  -webkit-transform: scale(2); /* Chrome, Safari, Opera */
  transform: scale(2);
  display: inline-block;
  padding: 6px;
  border: 1.5px solid #bbbbbb;
  border-radius: 10px;
  background-color: #e7e6e7;
  margin-left: 15px;
  margin-right: 15px;
  &:checked {
    background-color: #1E90FF;
  }
  &:focus {
```

Passing Parameterized Search Values



filter.jsx

```
  onRadioChange = (event) => {
    var value = event.target.value;
```

swaggerui-NETAPP/swagger-ui-3.19.5/src/core/components/operations.jsx

swaggerui-NETAPP/swagger-ui-3.19.5/src/core/containers/filter.jsx

swaggerui-NETAPP/swagger-ui-3.19.5/src/core/plugins/layout/actions.js

swaggerui-NETAPP/swagger-ui-3.19.5/src/core/plugins/layout/reducers.js

swaggerui-NETAPP/swagger-ui-3.19.5/src/core/plugins/layout/selectors.js

```
    options[e.target.name] = e.target.checked
    this.setState({options: options})
    this.props.layoutActions.updateOptions(options)
  }
```

Alternative Approaches to Search

- Looked into the javascript library Fuse.js to make our search algorithm more efficient
  - Fuse.js provides API for fuzzy search of nested objects and lists
  - Built-in scoring and weight system
  - Keys to be searched can be picked easily
- Collected runtime data about the different implementations to understand which one is most efficient.
  - Searched for different words multiple times
  - Measured time taken to get the search results

Found Results

Original Implementation

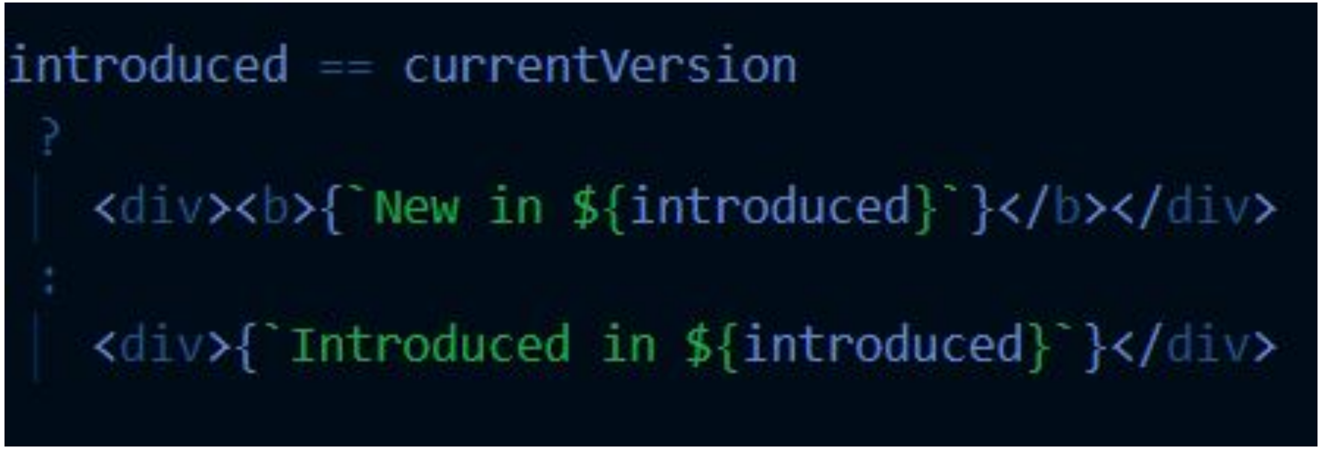
	Times in millisecond					
Keyword	**try 1**	**try 2**	**try 3**	**try 4**	**try 5**	**AVG**
Cloud	65.480	36.220	30.560	24.185	24.480	36.1850
uuid	50.82	20.495	25.185	19.36	18.065	26.7850
cluster	56.585	30.57	21.800	21.325	20.98	30.2520
storage	75.115	19.21	22.645	22.275	23.055	32.4600
parameter	79.565	28.835	27.77	20.93	21.165	35.6530
in	58.545	21.85	22.62	28.235	26.015	31.4530
the	50.835	30.175	17.365	15.79	18.32	26.4970

33x increase in runtime!

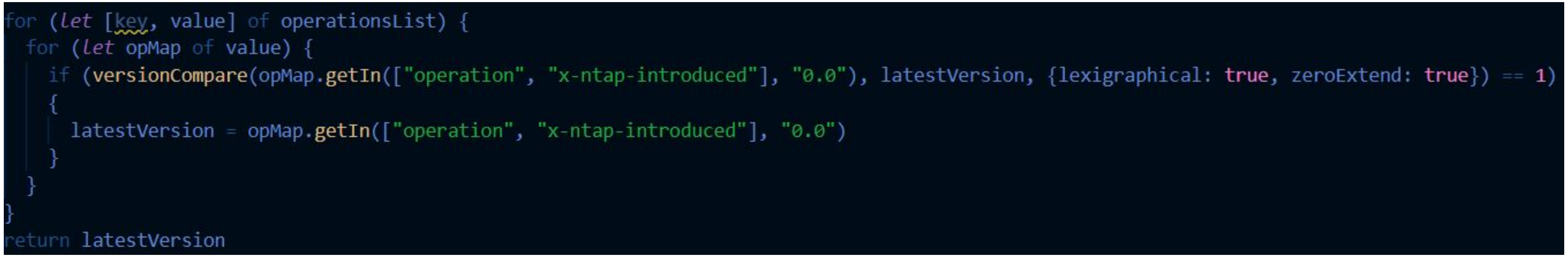
Using Fuse.js

	Times in millisecond					
Keyword	**try 1**	**try 2**	**try 3**	**try 4**	**try 5**	**AVG**
Cloud	1163.530	921.945	812.240	899.510	895.385	938.5220
uuid	1131.95	745.9	789.085	736.325	718.53	824.3580
cluster	1963.705	1196.76	1130.525	1149.785	1193.6	1326.8750
storage	1710.125	1214.39	1244.03	1221.505	1174.23	1312.8560
parameter	1830.895	1347.32	1437.825	1329.27	1325.04	1454.0700
in	875.33	547.62	529.015	485.76	451.67	577.8790
the	925.605	609.145	598.04	585.455	640.515	671.7520

Version Tracking

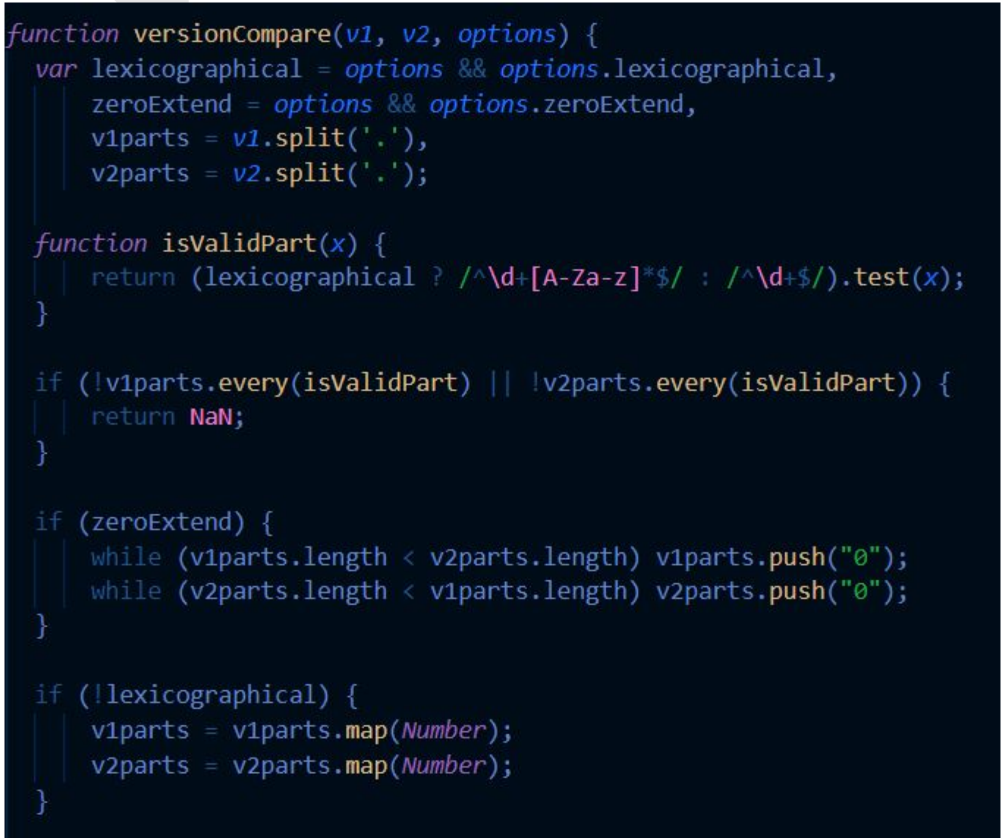


```
introduced == currentVersion
 ?
   <div><b>{`New in ${introduced}`}</b></div>
 :
   <div>{`Introduced in ${introduced}`}</div>
```

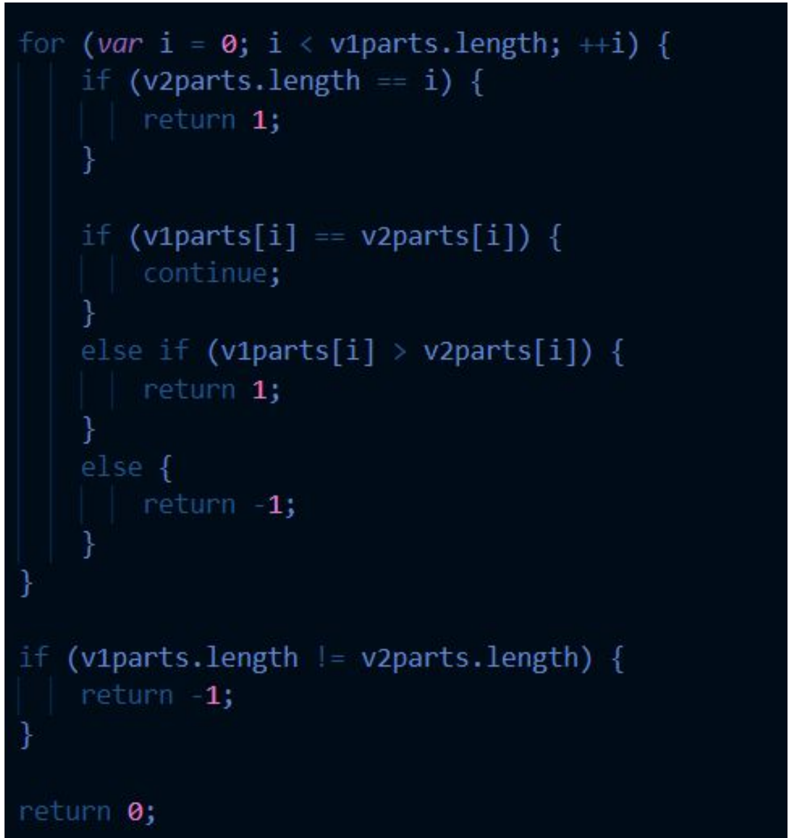


```
for (let [key, value] of operationsList) {
  for (let opMap of value) {
    if (versionCompare(opMap.getIn(["operation", "x-ntap-introduced"], "0.0"), latestVersion, {lexigraphical: true, zeroExtend: true}) == 1)
    {
      latestVersion = opMap.getIn(["operation", "x-ntap-introduced"], "0.0")
    }
  }
}
return latestVersion
```

Version Tracking Comparison Function

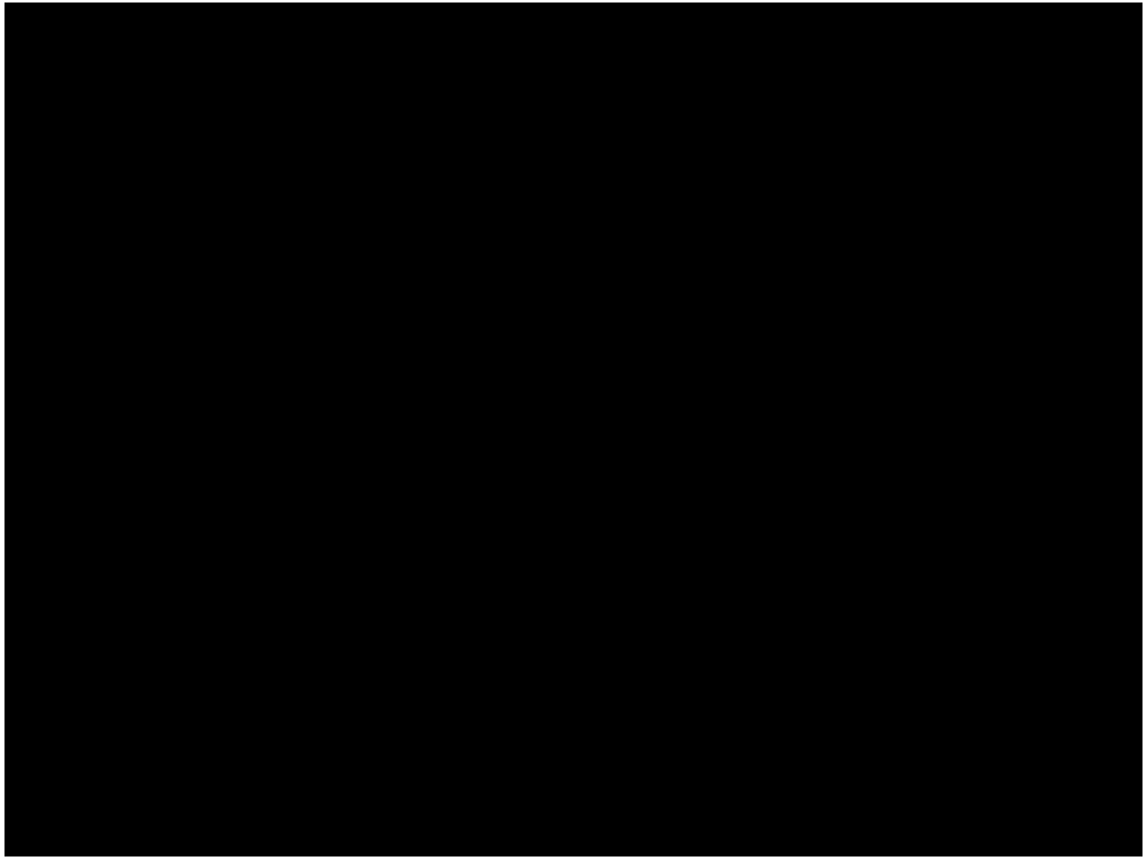


```javascript
function versionCompare(v1, v2, options) {
  var lexicographical = options && options.lexicographical,
      zeroExtend = options && options.zeroExtend,
      v1parts = v1.split('.'),
      v2parts = v2.split('.');

  function isValidPart(x) {
      return (lexicographical ? /^\d+[A-Za-z]*$/ : /^\d+$/).test(x);
  }

  if (!v1parts.every(isValidPart) || !v2parts.every(isValidPart)) {
      return NaN;
  }

  if (zeroExtend) {
      while (v1parts.length < v2parts.length) v1parts.push("0");
      while (v2parts.length < v1parts.length) v2parts.push("0");
  }

  if (!lexicographical) {
      v1parts = v1parts.map(Number);
      v2parts = v2parts.map(Number);
  }
```



```javascript
for (var i = 0; i < v1parts.length; ++i) {
    if (v2parts.length == i) {
        return 1;
    }

    if (v1parts[i] == v2parts[i]) {
        continue;
    }
    else if (v1parts[i] > v2parts[i]) {
        return 1;
    }
    else {
        return -1;
    }
}

if (v1parts.length != v2parts.length) {
    return -1;
}

return 0;
```

Demo



Challenges

- Learning new technologies
- Ensuring our algorithm was efficient
- Finding and fixing identified bugs in the code
- Enhancing our project's parameterized checkbox-based search using React JS *state* variables
- Operations deleting upon expansion when the Models radio button was selected

Thoughts on Future Work

- Creation of a new layout using multiple columns like other visualization tools
- Auto-expansion of endpoints that match a search
- Highlight matching substrings in search results
- Automated testing

Special Thanks!

Special thanks go to the NetApp team that has helped and guided us throughout this project. Their time and effort is always appreciated!

Anuradha Kulkarni

Sami Benbourenane

Brian Kinkade

Thank you! Any questions?