

VISUALISASI KURIKULUM 2018 DENGAN VIS.JS DAN ELECTRON

JOSHUA DELAVO SETIADI—2017730028

1 Data Skripsi

Pembimbing utama/tunggal: **Pascal Alfadian**

Pembimbing pendamping: -

Kode Topik : **PAN5091**

Topik ini sudah dikerjakan selama : **1 semester**

Pengambilan pertama kali topik ini pada : **Semester 8 - Genap 21/22**

Pengambilan pertama kali topik ini di kuliah : **Skripsi 1**

Tipe Laporan : **B** - Dokumen untuk reviewer pada presentasi dan **review Skripsi 1**

2 Latar Belakang

Electron adalah salah satu kerangka kerja terbaru yang memungkinkan pengembang membuat aplikasi *desktop* asli dengan teknologi web populer : JavaScript, HTML5, dan CSS. Dengan *Electron*, pengembang web dapat menggunakan keterampilan yang mereka miliki untuk membangun aplikasi yang memiliki banyak kemampuan seperti aplikasi *desktop* asli. *Elektron* telah menjadi sangat populer sejak dirilis dan digunakan oleh perusahaan, seperti : *Microsoft*, *Facebook*, *Slack*, dan *Docker*. Aplikasi ini dapat dikemas untuk dapat berjalan langsung di *macOS*, *Windows*, dan *Linux*. Bisa juga didistribusikan melalui *Mac App Store* atau *Microsoft Store*.

Vis.js adalah sebuah *library* visualisasi berbasis *browser* yang bersifat dinamis. *Library* ini dirancang agar mudah digunakan untuk menangani data dinamis dalam jumlah yang besar dan memungkinkan untuk memanipulasi serta berinteraksi dengan data tersebut. *Library* ini terdiri dari komponen - komponen, seperti *DataSet*, *Timeline*, *Network (tree)*, *Graph2d*, dan *Graph3d*. *DataSet* berfungsi untuk mengelola data yang tidak terstruktur. Terdapat fitur *add*, *update*, dan *remove* data di dalamnya. *Timeline* berfungsi untuk menampilkan data dalam bentuk *timeline* yang dapat disesuaikan dengan *item* dan rentangnya. *Network (tree)* berfungsi untuk menampilkan data dalam bentuk jaringan yang dinamis, dapat diatur secara otomatis, dan dapat disesuaikan. *Graph2d* berfungsi untuk menampilkan data dalam bentuk grafik dan diagram batang pada *timeline* yang interaktif sesuai yang diinginkan. *Graph3d* berfungsi untuk menampilkan data dalam bentuk grafik 3d dengan animasi yang interaktif.

GitHub adalah sebuah aplikasi berbasis *website* dengan *Version Control System* (VCS) yang menyediakan layanan untuk menyimpan *repository* dengan gratis. VCS adalah sebuah infrastruktur yang dapat mendukung pengembangan *software* secara kolaboratif. Setiap anggota yang berada di dalam sebuah tim pengembangan *software* dapat menulis kode programnya masing - masing kemudian digabungkan ke server yang sudah memiliki VCS yang digunakan. *Repository* merupakan tempat yang dapat digunakan untuk menyimpan berbagai file berupa *source code*. Aplikasi ini termasuk sangat populer dan banyak digunakan termasuk oleh perusahaan - perusahaan besar, seperti : *Facebook*, *Google*, dan *Twitter*.

Saat mahasiswa akan melakukan *FRS*, seringkali mereka kesulitan untuk melihat kurikulum tahun ajaran yang berlaku. Karena setiap kurikulum memiliki aturan yang berbeda dalam pengambilan matakuliah ataupun matakuliah yang disediakan, maka mahasiswa kadang bingung untuk memilih matakuliah apa yang akan diambil di semester berikutnya.

Maka dari itu, pada skripsi ini akan dibuat sebuah aplikasi visualisasi kurikulum 2018 berbasis *Electron* dengan menggunakan *library Vis.js*. Dengan aplikasi ini diharapkan mahasiswa dapat lebih mudah

untuk melihat kurikulum yang ada sehingga mempermudah mereka untuk memilih matakuliah apa yang akan diambil di semester berikutnya, kemudian dengan penggunaan *framework cross platform Electron* ini diharapkan semua mahasiswa pengguna *macOs*, *Windows*, dan *Linux* dapat mengaksesnya dengan mudah.

3 Rumusan Masalah

Rumusan masalah yang akan dibahas pada skripsi ini adalah :

1. Bagaimana cara memvisualisasikan kurikulum 2018 dalam bentuk *tree*?
2. Bagaimana cara memvisualisasikan kurikulum 2018 dalam bentuk *timeline*?
3. Bagaimana cara membaca kurikulum 2018 FTIS UNPAR dari *github*?

4 Tujuan

Tujuan yang akan dicapai dari penulisan skripsi ini adalah :

1. Memahami cara memvisualisasikan kurikulum 2018 dalam bentuk *tree*.
2. Memahami cara memvisualisasikan kurikulum 2018 dalam bentuk *timeline*.
3. Memahami cara membaca data kurikulum 2018 FTIS UNPAR dari *github*.

5 Detail Perkembangan Pengerjaan Skripsi

Detail bagian pekerjaan skripsi sesuai dengan rencana kerja/laporan perkembangan terakhir :

1. Mempelajari *framework Electron*.

Status : Ada sejak rencana kerja skripsi.

Hasil : Saya menjadi tahu bahwa *Electron* merupakan sebuah *framework* untuk membangun sebuah aplikasi *desktop* dengan menggunakan *JavaScript*, *HTML*, dan *CSS*. *Electron* memungkinkan kita mempertahankan satu basis kode *JavaScript* dan membuat aplikasi lintas *platform* yang berfungsi di *Windows*, *macOS*, dan *Linux*.

2. Mempelajari *library Vis.js*.

Status : Ada sejak rencana kerja skripsi.

Hasil : Saya menjadi tahu bahwa *Vis.js* merupakan sebuah visualisasi *library* berbasis *browser* yang dinamis. *Library* dirancang agar mudah digunakan, untuk menangani sejumlah besar data dinamis, dan memungkinkan untuk manipulasi dan berinteraksi dengan data. *Library* tersebut terdiri dari komponen *DataSet*, *Timeline*, *Network*, *Graph2d* dan *Graph3d*.

• Timeline

Timeline adalah grafik visualisasi interaktif untuk memvisualisasikan data dalam bentuk waktu. Item data dapat berlangsung pada satu tanggal, atau memiliki tanggal mulai dan berakhir. Kita dapat dengan bebas memindahkan dan memperbesar *timeline* dengan *drag* dan *scroll* pada *timeline*.

• Network

Jaringan (*Network*) adalah visualisasi untuk menampilkan jaringan - jaringan yang terdiri dari *node* dan *edge*. Visualisasinya mudah digunakan dan mendukung bentuk kustom, gaya, warna, ukuran, gambar, dan lain - lain. Visualisasi jaringan bekerja dengan lancar di *browser* moderen apapun hingga beberapa ribu *node* dan *edge*.

3. Mencoba membuat aplikasi berbasis *Electron*

Status : Ada sejak rencana kerja skripsi.

Hasil : Saya telah mencoba membuat aplikasi *Electron* dengan mengikuti langkah - langkah *quick start* pada dokumentasi *Electron* seperti berikut :

- Install terlebih dahulu Node.js dan npm (pakai versi terakhir LTS yang tersedia).
- Cek versinya dengan mengetik `node -v` dan `npm -v` pada command prompt.
- Buat folder dan inialisasi paket npm dengan mengetik `mkdir namaFolder` lalu ketik `cd namaFolder` setelah itu ketik `npm init` pada *command prompt*. Maka isi *package.json* akan seperti pada Kode 1.

Kode 1: *Package.json*

```
1 {  
2   "name": "my-electron-app",  
3   "version": "1.0.0",  
4   "description": "Hello World!",  
5   "main": "main.js",  
6   "author": "Jane Doe",  
7   "license": "MIT",  
8 }
```

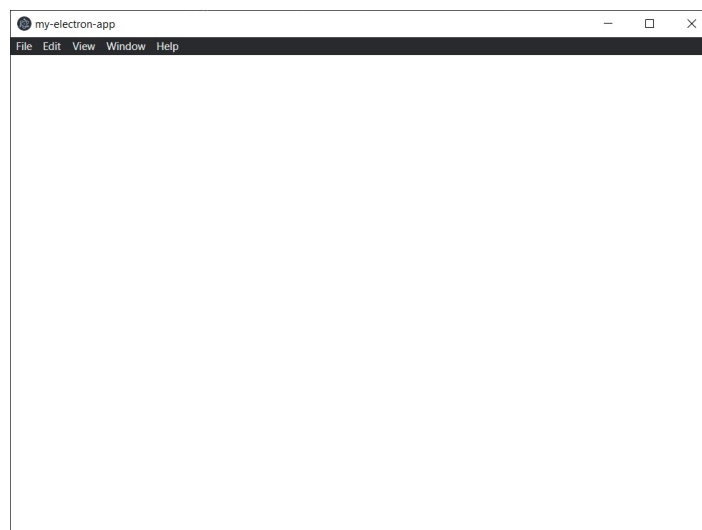
- Install aplikasi *electron* ke dalam folder yang telah dibuat dengan cara mengetik `npm install --save-dev electron` pada command prompt.
- Pada *script file package.json* tambahkan *command start* seperti pada Kode 2.

Kode 2: *Start command*

```
1 {  
2   "scripts": {  
3     "start": "electron ."  
4   }  
5 }
```

- Jalankan aplikasi *electron* dengan mengetik `npm start` pada command prompt.

Untuk hasilnya dapat dilihat pada Gambar 1.



Gambar 1: Hasil contoh aplikasi *Electron*.

4. Mencoba memvisualisasikan data dalam bentuk *timeline* dan *tree* dengan *Vis.js*.

Status : Ada sejak rencana kerja skripsi.

Hasil : Saya telah mencoba membuat *timeline* dan *tree* menggunakan *library Vis.js*. Contoh kode pembuatan *timeline* dapat dilihat pada Kode 3 dan untuk hasilnya dapat dilihat pada gambar 2.

Kode 3: Contoh kode untuk membuat *timeline* menggunakan *vis.js*

```

1 <!DOCTYPE HTML>
2 <html>
3 <head>
4   <title>Timeline | Basic demo</title>
5
6   <style type="text/css">
7     body, html {
8       font-family: sans-serif;
9     }
10  </style>
11
12  <script src="../../dist/vis.js"></script>
13  <link href="../../dist/vis-timeline-graph2d.min.css" rel="stylesheet" type="text/css" />
14 </head>
15 <body>
16 <div id="visualization"></div>
17
18 <script type="text/javascript">
19   // DOM element where the Timeline will be attached
20   var container = document.getElementById('visualization');
21
22   // Create a DataSet (allows two way data-binding)
23   var items = new vis.DataSet([
24     {id: 1, content: 'item 1', start: '2013-04-20'},
25     {id: 2, content: 'item 2', start: '2013-04-14'},
26     {id: 3, content: 'item 3', start: '2013-04-18'},
27     {id: 4, content: 'item 4', start: '2013-04-16', end: '2013-04-19'},
28     {id: 5, content: 'item 5', start: '2013-04-25'},
29     {id: 6, content: 'item 6', start: '2013-04-27'}
30   ]);
31
32   // Configuration for the Timeline
33   var options = {};
34
35   // Create a Timeline
36   var timeline = new vis.Timeline(container, items, options);
37 </script>
38 </body>
39 </html>

```

A basic timeline. You can move and zoom the timeline, and select items.

Gambar 2: Hasil contoh untuk membuat *timeline* menggunakan *vis.js*

Selanjutnya, contoh kode pembuatan *network* dapat dilihat pada Kode 4 dan untuk hasilnya dapat dilihat pada gambar 3.

Kode 4: Contoh kode untuk membuat *network* menggunakan *vis.js*

```

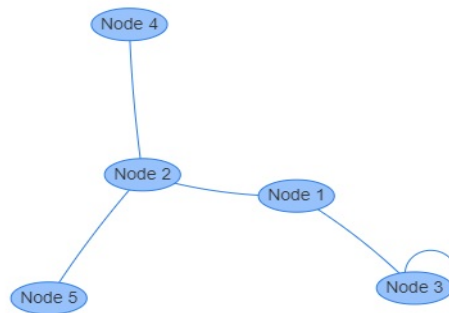
1 <html>
2 <head>
3   <script type="text/javascript" src="https://unpkg.com/vis-network/standalone/umd/vis-network.min.js"></script>
4
5   <style type="text/css">
6     #mynetwork {
7       width: 600px;
8       height: 400px;
9       border: 1px solid lightgray;
10    }
11  </style>
12 </head>
13 <body>
14 <div id="mynetwork"></div>
15
16 <script type="text/javascript">
17   // create an array with nodes
18   var nodes = new vis.DataSet([
19     {id: 1, label: 'Node 1'},
20     {id: 2, label: 'Node 2'},
21     {id: 3, label: 'Node 3'},
22     {id: 4, label: 'Node 4'},
23     {id: 5, label: 'Node 5'}

```

```

24     });
25
26     // create an array with edges
27     var edges = new vis.DataSet([
28         {from: 1, to: 3},
29         {from: 1, to: 2},
30         {from: 2, to: 4},
31         {from: 2, to: 5}
32     ]);
33
34     // create a network
35     var container = document.getElementById('mynetwork');
36
37     // provide the data in the vis format
38     var data = {
39         nodes: nodes,
40         edges: edges
41     };
42     var options = {};
43
44     // initialize your network!
45     var network = new vis.Network(container, data, options);
46 </script>
47 </body>
48 </html>

```



Gambar 3: Hasil contoh untuk membuat *network* menggunakan *vis.js*

5. Mempelajari data kurikulum 2018 di *github* beserta cara pengambilan datanya.

Status : Ada sejak rencana kerja skripsi.

Hasil : Data kurikulum 2018 FTIS di github berbentuk array untuk setiap matakuliahnya. Setiap struktur matakuliah memiliki properti sebagai berikut :

- kode (String)
- nama (String)
- prasyarat
 - tempuh (String[])
 - lulus (String[])
 - bersamaan (String[])
 - berlakuAngkatan (Number|null)
- sks (Number)
- wajib (Boolean)
- semester (Number)

Untuk lebih jelasnya dapat dilihat pada Kode 5.

Kode 5: prasyarat.json

```

1 {
2   "kode": "AIF181100",
3   "nama": "Dasar_Pemrograman",
4   "prasyarat": {
5     "tempuh": [],
6     "lulus": [
7       "AIF181101"
8     ],
9     "bersamaan": [],
10    "berlakuAngkatan" : 2018
11  },
12  "sks": 4,
13  "wajib": true,
14  "semester": 2
15 }

```

Pada prasyarat lulus, diberikan sebuah string berupa kode mata kuliah yang menjadi prasyarat lulus untuk mata kuliah tersebut. Definisi prasyarat berdasarkan macammnya :

- Tempuh
Mahasiswa diharuskan sudah pernah menempuh mata kuliah yang disebutkan.
- Lulus
Mahasiswa diharuskan untuk lulus mata kuliah yang disebutkan.
- Bersamaan
Mata kuliah tersebut harus di ambil secara bersamaan dengan mata kuliah yang disebutkan (butuh informasi lagi).
- Berlaku Angkatan
Property ini mulai berlaku karena pergantian kurikulum. Prasyarat ini memiliki maksud bahwa matakuliah ini mulai berlaku semenjak angkatan x.

Untuk cara pengambilan datanya, dapat melakukan *looping* data untuk setiap *array* matakuliahnya.

6. Membuat dokumen skripsi.

Status : Ada sejak rencana kerja skripsi.

Hasil : Saya telah membuat dokumen skripsi untuk bab pendahuluan, landasan teori, dan analisis.

6 Pencapaian Rencana Kerja

Langkah-langkah kerja yang berhasil diselesaikan dalam Skripsi 1 ini adalah sebagai berikut:

1. Sudah membuat rencana kerja skripsi.
2. Sudah membuat dokumen skripsi bab pendahuluan.
3. Sudah membuat dokumen skripsi bab landasan teori.
4. Sudah membuat dokumen skripsi bab analisis.
5. Sudah mempelajari cara membuat aplikasi berbasis *Electron*.
6. Sudah mempelajari cara memvisualisasikan data dalam bentuk *tree* dan *timeline* dengan *Vis.js*.
7. Sudah mempelajari data kurikulum 2018 di github beserta cara pengambilan datanya.

7 Kendala yang Dihadapi

Kendala - kendala yang dihadapi selama mengerjakan skripsi :

- Minimnya referensi mengenai *Electron* dan *Vis.js* (*timeline* dan *network*).
- Banyaknya tugas yang diberikan oleh mata kuliah lain, sehingga kurang waktu untuk *explore* lebih jauh.

Bandung, 12/06/2021

Joshua Delavo Setiadi

Menyetujui,

Nama: Pascal Alfadian
Pembimbing Tunggal