

ARSITEKTUR PERANGKAT LUNAK

MIDDLEWARE

Anggota :

Dian Rahmaji (17/413896/TK/46336)

Farand Farhansyah (17/415233/TK/46522)

M. Fitroh Fajariyadi (17/413907/TK/46347)

Definisi

Middleware adalah layanan yang beroperasi di antara lapisan aplikasi dan lapisan komunikasi jaringan. Layanan tersebut muncul pada sisi client dan server. Middleware biasanya digunakan pada perangkat lunak terdistribusi yang dapat menyederhanakan pengembangan perangkat lunak tersebut dengan menerapkan hal-hal berikut:

1. Menyembunyikan detail operasi dari suatu perangkat lunak
2. Menyelesaikan masalah perbedaan *hardware*, sistem operasi, dan protocol yang digunakan
3. Menyediakan antarmuka yang *high - level* dan seragam sehingga perangkat lunak dapat lebih mudah digunakan dan portable
4. Menyediakan layanan yang umum digunakan sehingga dapat mengurangi duplikasi dan meningkatkan kolaborasi antar perangkat lunak

Message Oriented Middleware (MOM) adalah aplikasi yang independen, autonomous, dibangun dengan teknologi bermacam-macam dan berjalan di platform yang berbeda. MOM menempatkan queue diantara sender dan receiver. Manfaat menggunakan MOM adalah:

1. Pesan yang dikirim langsung dihapus
2. Meskipun proses membutuhkan waktu lama, hal ini tidak menghambat sender
3. Kegagalan yang terjadi menjadi tanggung jawab middleware

Penjelasan Project

Project yang dibuat untuk tugas ini adalah JADWALIN, yaitu aplikasi berbasis yang memiliki fitur untuk pengingat tugas kuliah. Project ini adalah aplikasi website yang dibuat dengan bahasa pemrograman Javascript untuk bagian backend (NodeJS) dan frontend (VueJS). Karena aplikasi berbasis website, protokol komunikasi yang digunakan adalah HTTP Protocol dengan browser Mozilla sebagai client dan server

Two-way Messaging

1. Pengguna memasukkan credential (email dan password) melalui form di client.

LOGIN

```
OPTIONS /api/login 200 7.694 ms - 2
credential:
{ email: 'dianrahmaji@gmail.com', password: 'root' }

Generating token . . .

token:
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VySUQiOiI1Y2FhZmQzLnZyYydkNkTi8VUYtcNRRs6fqKtrbTn621UG4GTcPyhGM

POST /api/login 200 221.141 ms - 255
OPTIONS /api/homeworks 200 0.663 ms - 2
GET /api/homeworks 200 108.555 ms - 1525
```

A screenshot of the Chrome DevTools Console. The top toolbar shows icons for Inspector, Console (selected), Debugger, Style Editor, Performance, Memory, and a menu icon. Below the toolbar, the 'Filter output' section is visible with a trash icon and a search filter. The console log shows a message: '[HMR] Waiting for update signal from WDS...' with a source link 'log.js:24'. Below this is a long alphanumeric string: 'eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2V5SUQiOiIiY2RhYWU1M2ZiZWlwMDEzZW40MTA0NTM1LCl1bWVwPbCI6ImRyYW5yWhWtYwppQ6dtYWlsLmNmVbSI6ImhhdCI6MTU1NzgzNTE0NywiOiJxNjU0NTU3ODY0Nz03f0.SzCvdKkNkTi8VUYytCNRrs6foKtrbTn621UG4GtCPvhGM' with a source link 'auth.js:15'.

Transaction

Konsep transaction yang digunakan pada aplikasi ini adalah HTTP transaction. Jenis transaksi yang ada pada aplikasi ini adalah POST dan GET. Transaksi POST digunakan untuk mengirim data dari client dan menyimpannya di database. Transaksi GET digunakan untuk melakukan query database dan kemudian mengirimkan ke client untuk ditampilkan kepada user. Proses transaksi POST yang terjadi adalah:

1. Pengguna memasukkan data yang ingin disimpan ke database melalui form. Dalam kasus ini, data yang disimpan adalah detail tugas kuliah yang ingin dicatat oleh pengguna.

Add a New Homework

Title

determinan 2

Course

linear algebra

Difficulty

medium

Information (Optional)

Due Date

May 21, 2019

ADD

2. Data tersebut diterima oleh server

```
GET /api/homeworks 200 124.843 ms - 1525
OPTIONS /api/homeworks 200 0.547 ms - 2
{ title: 'determinan 2',
  course: 'linear algebra',
  due: '21 May 2019',
  difficulty: 'medium',
  status: 'ongoing',
  info: '' }
POST /api/homeworks 200 3.822 ms - 17
OPTIONS /api/homeworks 200 0.305 ms - 2
```

3. Data disimpan di database (MongoDB Atlas)

```
_id: ObjectId("5cdb2c89ab6fd4568ce863d5")
title: "determinan 2"
course: "linear algebra"
due: 2019-05-20T17:00:00.000+00:00
difficulty: "medium"
status: "ongoing"
info: ""
userID: ObjectId("5cd4aa53fbeb0013ec810453")
__v: 0
```

Proses transaksi GET yang terjadi adalah:

1. Setelah berhasil login, pengguna akan dialihkan ke link <http://localhost:8080/> oleh browser. Ketika hal tersebut terjadi, client akan mengakses API di <http://localhost:3000/api/homework>. Ketika API tersebut diakses, middleware yang handle route tersebut akan melakukan query ke database dan kemudian mengirimkannya ke client.

Requested data:

```
[ { id: 5cdb0ce8ae940d44b1225be0,
  title: 'middleware',
  course: 'software architecture',
  due: 2019-05-14T17:00:00.000Z,
  difficulty: 'hard',
  status: 'complete',
  info: '',
  userID: 5cdaae53fbeb0013ec810453,
  v: 0 },
  { id: 5cdb0d1aae940d44b1225be1,
  title: 'user centered design',
  course: 'human computer interaction',
  due: 2019-05-29T17:00:00.000Z,
  difficulty: 'hard',
  status: 'ongoing',
  info: '',
  userID: 5cdaae53fbeb0013ec810453,
  v: 0 },
  { id: 5cdb0d3fae940d44b1225be2,
  title: 'faraday law',
  course: 'electromagnetics',
  due: 2019-05-15T17:00:00.000Z,
  difficulty: 'medium',
  status: 'ongoing',
  info: '',
  userID: 5cdaae53fbeb0013ec810453,
  v: 0 },
  { id: 5cdb0d63ae940d44b1225be3,
  title: 'determinant',
  course: 'linear algebra',
  due: 2019-05-16T17:00:00.000Z,
  difficulty: 'medium',
  status: 'ongoing',
  info: '',
  userID: 5cdaae53fbeb0013ec810453,
  v: 0 },
  { id: 5cdb0d99ae940d44b1225be4,
  title: 'COCOMO II',
  course: 'software engineering',
  due: 2019-05-19T17:00:00.000Z,
  difficulty: 'medium',
  status: 'ongoing',
  info: '',
  userID: 5cdaae53fbeb0013ec810453,
  v: 0 },
  { id: 5cdb0ddaee940d44b1225be5,
  title: 'maxwell equation',
  course: 'electromagnetics tutorial',
  due: 2019-05-20T17:00:00.000Z,
  difficulty: 'medium',
  status: 'ongoing',
```

2. Client akan menerima data dan kemudian ditampilkan kepada user.

by homework name by course name by due time by difficulty				
Homework Title Middleware	Course Software Architecture	Due By May 15, 2019	Difficulty Hard	Complete
Homework Title User Centered Design	Course Human Computer Interaction	Due By May 30, 2019	Difficulty Hard	Ongoing
Homework Title Faraday Law	Course Electromagnetics	Due By May 16, 2019	Difficulty Medium	Ongoing
Homework Title Determinant	Course Linear Algebra	Due By May 17, 2019	Difficulty Medium	Ongoing
Homework Title COCOMO II	Course Software Engineering	Due By May 20, 2019	Difficulty Medium	Ongoing
Homework Title Maxwell Equation	Course Electromagnetics Tutorial	Due By May 21, 2019	Difficulty Medium	Ongoing
Homework Title Least Square	Course Linear Algebra	Due By May 21, 2019	Difficulty Medium	Ongoing
				ADD

Source Code (Terlampir)

1. Two-way Messaging (Login Handler)

```
router.post('/login', (req, res, next) => {
  User.findOne({email: req.body.email}).exec().then(user => {
    bcrypt.compare(req.body.password, user.password, (err,
result) => {
      if(err) {
        return res.status(401).json({
          msg: 'Auth failed'
        });
      }
      if (result) {
        console.log(' ');
        console.log('credential: ');
        console.log(req.body);
        console.log(' ');
        const token = jwt.sign({
          userID: user._id,
          email: user.email
        }, 'secret', {expiresIn: '1h'});

        console.log(' ');
        console.log('Generating token . . .');
```

```

        console.log(' ');
        console.log('token: ');
        console.log(token);
        console.log(' ');
        return res.status(200).json({
            msg: 'Auth successful',
            token: token
        });
    }
    res.status(401).json({
        msg: 'Auth failed'
    });
});
}).catch(err => {
    console.log(err);
    res.status(500).json({
        error: err
    });
})
});

```

2. HTTP GET Transaction

```

router.get('/', auth, (req, res, next) => {
    const token = req.headers.authorization.split(' ')[1];
    const decoded = jwt.verify(token, 'secret');
    Homework.find({ userID: db.Types.ObjectId(decoded.userID) },
    async (err, homework) => {
        if(err) res.status(500).send(err);
        console.log(' ');
        console.log('Requested data: ');
        console.log(homework);
        console.log(' ');
        res.status(200).send(await homework);
    });
});

```

3. HTTP POST Transaction

```
router.post('/', auth, async (req, res, next) => {
  const token = req.headers.authorization.split(' ')[1];
  const decoded = jwt.verify(token, 'secret');
  console.log(req.body);
  const homework = new Homework({
    _id: new db.Types.ObjectId,
    title: req.body.title,
    course: req.body.course,
    due: req.body.due,
    difficulty: req.body.difficulty,
    status: req.body.status,
    info: req.body.info,
    userID: db.Types.ObjectId(decoded.userID)
  });
  homework.save().then(res.status(200).json({msg:
'success'})).catch(error => {
    res.status(500).json({error});
    console.log(error);
  });
});
```

Source code diblock saat akan dikirimkan melalui e-mail. Oleh karena itu, saya kirimkan melalui:

- Google Drive :

<https://drive.google.com/drive/folders/1VRr7OO5GS29-iZ-n76NWwXdW2bw1IDWo?usp=sharing>

- GitHub :

<https://github.com/dianrahmaji/middleware.git>