

APLIKASI CHATTING AKAKOM CAREER CENTER (MEMBUAT CHATTING SEPERTI WHATSAPP)

¹⁾Indra Yatini, ²⁾F. Wiwiek Nurwiyati, ³⁾Rini

^{1),2),3)} Staf pengajar jurusan Teknik informatika, STMIK AKAKOM Yogyakarta
Jalan Raya Janti 143 Karang Jambe Yogyakarta 55198
E-Mail : ¹⁾indrayatini@akakom.ac.id, ²⁾femi@akakom.ac.id

ABSTRACT

Communication is one of the most important things humans need in life. In this case chat is used to communicate directly to fellow internet users who are online. With the chat application we can send messages to others who are online, then the intended person will reply to the message, and so on. The current growing technology can be used for various things, especially in solving problems to convey information. One such technology is to use Node JS and Socket.IO technology to implement chat application required for AKAKOM Career Center to convey information about career services as well as to be used for media discussion of AKAKOM Career Center with alumni. The results of this research are implemented into a web application that can be used to convey information by sending and receiving messages within the provided chat page.

Keyword: Node JS, Chatting, Socket.IO, MongoDB.

PENDAHULUAN

Chatting adalah suatu pesan instant ataupun instant messaging di sebuah teknologi jaringan komputer yang memungkinkan pemakainya untuk mengirimkan pesan ke pengguna lain yang tersambung dalam sebuah jaringan komputer ataupun internet.

Chatting tidak hanya populer dikalangan kaum remaja atau anak muda saja tetapi jaman sekarang ini, sudah merambah ke kalangan orang tua sekalipun. Dengan adanya chatting, kita dapat dengan bebas mengobrol mengenai apa saja mulai dari persahabatan, pekerjaan, pelajaran di sekolah, mata kuliah, sampai masalah percintaan dan perjodohan, bahkan sampai dengan hal-hal bersifat pribadi sekali pun.

Di Indonesia sendiri, WhatsApp baru populer sejak 1-2 tahun terakhir berhubung sebelum itu, BlackBerry masih sangat mendominasi di Indonesia dan begitu juga dengan platform chatting mereka, *BlackBerry Messenger* (BBM). Setelah dominasi BlackBerry agak berkurang dalam beberapa tahun terakhir, terutama karena dipandang tidak mampu menyaingi ponsel cerdas berbasis *Android*, masyarakat Indonesia mencari platform chatting lain selain BBM

dan ketika itulah *WhatsApp* mulai populer. Diperkirakan jumlah penggunanya sudah lebih dari 10 juta orang di Indonesia.

Socket.io adalah library JavaScript yang memungkinkan komunikasi real-time antara web browser dan server. Library ini terdiri dari dua bagian, yang berjalan pada web browser dan yang berjalan pada server sebagai NodeJS module.

Node.js adalah platform yang dibangun di atas JavaScript runtime Chrome untuk dengan mudah membangun cepat, aplikasi jaringan scalable. Node.js menggunakan event-driven, non-blocking I/O model yang membuatnya ringan dan efisien, cocok untuk aplikasi real-time data-intensif yang berjalan di perangkat didistribusikan.

Pengembangan aplikasi di ambil adalah AKAKOM Career Centre. Dimana aplikasi chatting ini diharapkan dapat membantu bagian Akakom Career Centre dalam menyampaikan informasi serta dapat digunakan sebagai media diskusi.

Akakom Career Centre adalah sebuah bidang layanan karir STMIK Akakom Yogyakarta yang merupakan bidang yang berada dibawah pembantu ketua III STMIK Akakom Yogyakarta yang bertugas mengelola data alumni dan memberi informasi lowongan pekerjaan kepada alumni.

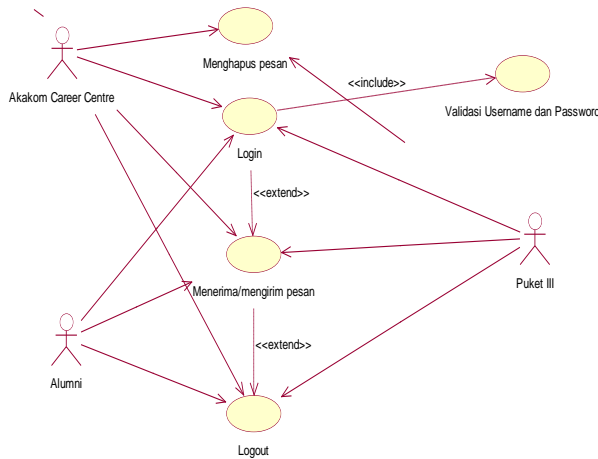
METODE PENELITIAN

Aplikasi Chatting Akakom Career Center(Membuat *Chatting* Seperti *Whatsapp*) Menggunakan Arsitektur Microservice Dengan Node.js, meliputi analisis dan perancangan sistem. Analisis digunakan untuk menganalisa kebutuhan perangkat lunak. Sedangkan perancangan akan digunakan untuk menggambarkan sistem yang akan dibuat.

A. Perancangan Sistem

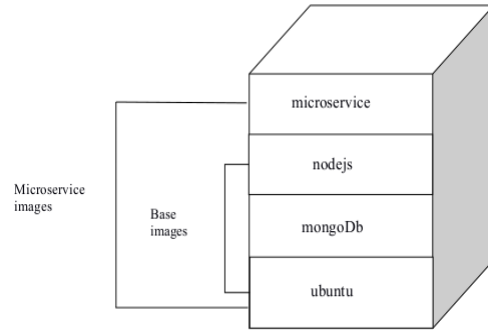
UML (*Unified Modeling Language*) adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasikan, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan software berbasis *OO (Object-Oriented)*. Berikut perancangan sistem yang ada pada aplikasi ini.

Pada kebutuhan perancangan sistem, pemodelan menggunakan *use case*. Diagram *use case* dapat menjadi teknik yang cukup baik untuk menganalisa kebutuhan pengguna sistem karena selain bagus dalam unsur pemahamannya, diagram *use case* juga dapat mendokumentasikan persyaratan sistem dengan baik, yang tampak pada gambar berikut.



Gambar 1. Use Case Diagram Sistem

B. Skema Private Cloud Docker Images



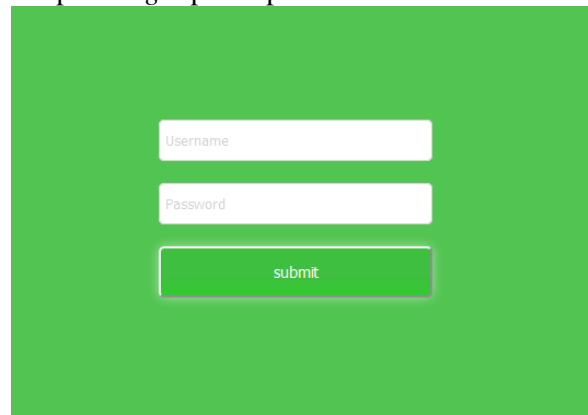
Gambar 2. Microservice Images Docker

Untuk implementasi dalam *private cloud* menggunakan docker, pada seperti terlihat pada gambar *Base Images* yang digunakan adalah ubuntu, mongoDb dan nodejs sedangkan untuk aplikasinya berada paling atas dan semuanya dibungkus dalam satu images yang bernama *microservice images, images* kemudian dijalankan pada komputer.

HASIL

Pembuatan dan uji coba aplikasi dengan menggunakan Arsitektur Microservice atau bisa juga disebut Microservices dengan menggunakan *platform node.js, databases mongoDB* dan menerapkannya kedalam *private cloud* dengan docker. Pembahasan pada sistem ini merupakan sebuah prosedur yang harus dilakukan guna menyelesaikan rancangan sistem.

Proses *login* berguna untuk mengatur setiap *user* yang akan mengelola aplikasi. Login digunakan oleh semua *user* untuk membedakan antara admin dan alumni, dibedakan berdasarkan level. Berikut adalah tampilan *login* pada aplikasi:



Gambar 3. Tampilan Login

```

app.post('/',function(req,res) {
  sess=req.session;
  if (req.body.username!=" && req.body.pass!="){
    username      =req.body.username;
    pass          =req.body.pass;

    User.findOne({username: username, password: pass, online:0}, function(err, user){
      if (err) {
        console.log(err);
        res.redirect('/');
      } else if (!user){
        res.redirect('/');
      } else {
        console.log(user);
        sess.user_id      = user._id;
        sess.nim          = user.nim;
        sess.username     = user.username;
        sess.name         = user.name;
        console.log(sess.name, "Connected");
        sess.level       = user.level
        sess.jurusan     = user.jurusan;
        sess.angkatan    = user.angkatan;
        sess.photo       = user.photo;
        res.redirect('/');
      }    });
    } else {
      res.redirect('/');
    }
  });
};});

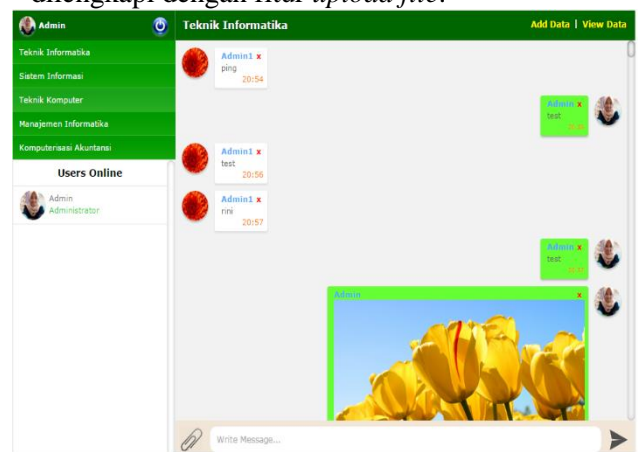
```

Gambar 4 Script Tampilan Login

Pada proses *login*, sistem akan menangkap data yang telah dikirimkan dari halaman *login*, data yang diambil adalah *username* dan *password*, sistem akan memeriksa data *username* dan *password* yang ada di *database* apakah data tersebut sesuai atau tidak. Jika data sesuai maka sistem akan membuat *session* dan mengarahkan kehalaman *chatting*. Jika tidak maka sistem akan mengarahkan ke halaman *login*.

Halaman *chat* admin, terdapat menu perjurusan yang terdiri atas Teknik Informatika, Sistem Informasi, Teknik Komputer, Manajemen Informatika, dan Komputerisasi Akuntansi. Selain itu terdapat daftar *user online* dan terdapat ruang *chat* untuk diskusi. Di bagian atas terdapat menu *add data* dan *view data*. Dan di bagian bawah

terdapat ruang untuk mengirim pesan yang dilengkapi dengan fitur *upload file*.



Gambar 5. Tampilan Halaman Admin

```

if (sess.level=='Administrator'){
  User.findById(sess.user_id, function(err, user){
    if (err) console.log(err);
    user.online=1;
    user.save(function(err){
      if (err) { console.log(err);
      } else if (!user) {
      } else {

```

```

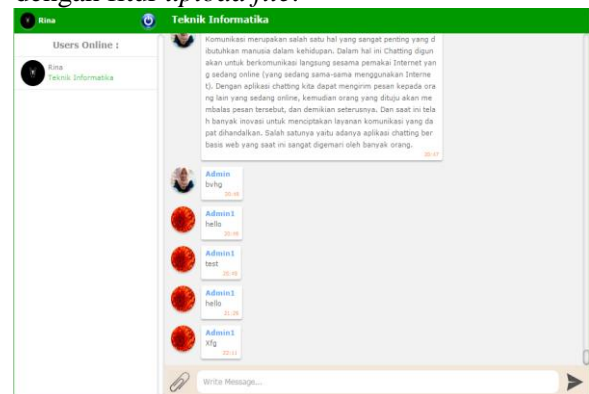
console.log('User online '+user.name+' successfully updated!');
    User.find({online:1}).lean().select({"nim": 4, "jurusan": 3, "photo": 2, "name": 1, "_id":
0}).exec(function (err, users){
if (err) console.log(err);
users_online_all = users;
Message.find({jurusan:          'Teknik          Informatika'}).sort(['message_created_at',
'ascending']).populate('id_user').lean().exec(function (err, messages_teknik_informatika){
Message.find({jurusan:          'Sistem          Informasi'}).sort(['message_created_at',
'ascending']).populate('id_user').lean().exec(function (err, messages_sistem_informasi){
    Message.find({jurusan:          'Teknik
Komputer'}).sort(['message_created_at',          'ascending']).populate('id_user').lean().exec(function
(err, messages_teknik_komputer){
    Message.find({jurusan:          'Manajemen
Informatika'}).sort(['message_created_at',          'ascending']).populate('id_user').lean().exec(function
(err, messages_manajemen_informatika){
    Message.find({jurusan:          'Komputerisasi
Akuntansi'}).sort(['message_created_at',          'ascending']).populate('id_user').lean().exec(function
(err, messages_komputerisasi_akuntansi){
if (err) console.log(err);
var results1 = messages_teknik_informatika.map(function(data_msg){
    return{          _id          :data_msg._id,
message_created_at          :          data_msg.message_created_at,          message_updated_at:
data_msg.message_updated_at,unique_code:          data_msg.unique_code,          message_txt          :
data_msg.message_txt, file : data_msg.file, jurusan : data_msg.jurusan, time1 : data_msg.time1,
time2 : data_msg.time2, nim : data_msg.id_user.nim, name :data_msg.id_user.name, photo :
data_msg.id_user.photo};
    });
    
```

Gambar 6 Script Halaman Admin

Pada proses halaman admin, data yang diambil adalah data *user online* dan data *chat* pesan berdasarkan jurusan yang terdiri atas Teknik Informatika, Sistem Informasi, Teknik Komputer, Manajemen Informatika, dan Komputerisasi Akuntansi. Setiap jurusan diambil data untuk kolom tertentu. Setelah data tersebut diambil, data tersebut dikirimkan atau dirender ke halaman admin.html.

Dalam halaman *chat* alumni, disamping kiri terdapat daftar *user online*, disamping kanan terdapat ruang *chat* untuk diskusi. Dan di bagian bawah terdapat ruang

untuk mengirim pesan yang dilengkapi dengan fitur *upload file*.



Gambar 7 Halaman Alumni

```

console.log(err); else if (!user) {} else {console.log('User online '+user.name+'successfully
updated!');
user.find({$and: [{online:1}, {$or: [ {jurusan: 'Teknik Informatika'}, {jurusan: 'Administrator'},
{jurusan: 'PUKET III'},] } ] } ) .lean().select({"nim":4, "jurusan":3, "photo":2, "name":1, "_id":0
}).exec(function (err, users_ti) {
user.find({$and: [{online:1}, {$or: [ {jurusan: 'Sistem Informasi'}, {jurusan: 'Administrator'},
{jurusan: 'PUKET III'},] } ] } ) .lean().select({"nim":4, "jurusan":3, "photo":2, "name":1, "_id":0
}).exec(function (err, users_si) {
    
```

```
User.find({$and: [{online:1}, {$or: [ {jurusan: 'Teknik Komputer'}, {jurusan: 'Administrator'}, {jurusan: 'PUKET III'},] } ] } ) .lean().select({"nim":4, "jurusan":3, "photo":2, "name":1, "_id":0}).exec(function (err, users_tk) {
User.find({$and: [{online:1}, {$or: [ {jurusan: 'Manajemen Informatika'}, {jurusan: 'Administrator'}, {jurusan: 'PUKET III'},] } ] } ) .lean().select({"nim":4, "jurusan":3, "photo":2, "name":1, "_id":0 }).exec(function (err, users_mi) {
```

Gambar 8 Script Halaman Alumni

Pada proses halaman alumni, data yang diambil adalah *data user online* dan data *chat* pesan berdasarkan jurusan yang terdiri atas Teknik Informatika, Sistem Informasi, Teknik Komputer, Manajemen Informatika, dan Komputerisasi Akuntansi. Setiap jurusan diambil data untuk kolom tertentu. Setelah data tersebut diambil, data tersebut dikirimkan atau dirender ke halaman *user.html*.

PEMBAHASAN

Aplikasi ini dibangun menggunakan arsitektur *microservice* dengan menggunakan *node.js* dalam pembangunanya, untuk *databases* menggunakan *databases* *noSQL* dengan menggunakan *mongoDb*. Sedangkan untuk penerapan dalam *cloud computing* dengan menggunakan *docker* sebagai wadah dalam aplikasi. Pada aplikasi ini menggunakan *seneca.js* dalam membangun aplikasi dengan *Microservices*.

```
heriipurnama@halloween:~/Documents/node/ha
ai-skripsi node index.js

2016-05-08T08:51:52.538Z
rtl9r5vwu8o6/1462697512481/162/- INFO
hello
Seneca/0.6.2/rtl9r5vwu8o6/1462697512481/16
2/-

2016-05-08T08:51:52.539Z
vsk7hb3i56jz/1462697512416/163/- INFO
hello
Seneca/0.6.2/vsk7hb3i56jz/1462697512416/16
3/-

2016-05-08T08:51:53.479Z
rtl9r5vwu8o6/1462697512481/162/- INFO
listen 10201

2016-05-08T08:51:53.596Z
rtl9r5vwu8o6/1462697512481/162/-
```

Gambar 9. Server Microservice

Pada Gambar 9 server yang digunakan untuk melihat setiap *service* yang berjalan pada console. Dari hasil uji coba aplikasi dengan menggunakan *microservice* ada beberapa keunggulan dalam menggunakan *microservice* diantaranya adalah :

1. Arsitektur memungkinkan setiap layanan untuk dikembangkan secara *independent* oleh tim yang difokuskan pada layanan tersebut.
2. Memungkinkan layanan untuk digunakan secara mandiri.
3. Setiap layanan dapat dipantau, diperbaharui secara *independent* untuk mencocokkan tautan.
4. Setiap layanan dapat dikembangkan dan diperbaharui secara mandiri.

Pada aplikasi ini dalam menggunakan *Private Cloud* *docker*, *docker* harus terkoneksi ke internet dalam menjalankan *docker images*. *Images Docker* bisa di *publish* dengan layanan dari *docker* dengan *docker hub*.

Sistem ini dibangun dengan menggunakan teknologi *Node JS*, *Socket.IO* dengan menggunakan database *MongoDB*.

Langkah pengambilan data dari *MongoDB* adalah data diambil dari database *MongoDB* dengan perintah *query* menggunakan bahasa pemrograman *Java Script*. Kemudian data dimasukkan ke dalam array yang akan ditampilkan pada saat sistem dijalankan. Untuk menampilkan data pesan, sistem akan merender halaman admin dan *user* dengan membawa data pesan tersebut.

Langkah memasukan data di lakukan oleh admin, berdasarkan data wisuda yang di dapat dari bagian akademik. User akan mendapatkan unername dan password apada saat pendataan yang di lakukan di bagian alumni sesuai dengan jurusannya.

User bisa langsung masuk ke dalam system dengan menggunakan username dan password yang di berikan bagian alumni dan user bisa langsung chatting dengan admin dan juga dengan sesama alumni.

Admin atau bagian alumni bisa memberikan informasi berupa teks dan gambar

kepada semua alumni yang telah terdaftar pada system.

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan serangkaian proses mulai dari perancangan hingga implementasi pada aplikasi ini, didapatkan beberapa kesimpulan, antara lain :

1. Aplikasi ini dapat digunakan untuk forum diskusi antara AKAKOM Career Centre dengan Alumni.
2. Aplikasi ini juga dapat membantu untuk mendapatkan informasi dan melakukan sebuah diskusi tentang lowongan pekerjaan.
3. Node JS dan Socket.IO dapat digunakan untuk membangun aplikasi *chatting* secara *realtime*.

B. Saran

Berdasarkan hasil evaluasi terhadap perangkat lunak yang dikembangkan agar pada pengembangan selanjutnya didapatkan hasil yang lebih baik lagi. Berikut adalah saran-saran yang dipandang perlu untuk dipertimbangkan untuk penelitian selanjutnya :

1. Aplikasi ini dapat di kembangkan menjadi versi *mobile*.
2. Penambahan penanda bahwa pesan tersebut telah terkirim dan telah terbaca oleh *user* yang lain.
3. Penambahan fitur perekam suara, pengambilan kamera, dan audio.

DAFTAR PUSTAKA

Agung Julisman, 2014, Bikin Aplikasi Android Dengan Angular Mobile mongoDB Studi Kasus Membuat Chat ala

BBM dan Whatsapp, Lokomedia, Bandung.

Eko Sularsono dkk, 2014, Implementasi Algoritma Rijndael 128 pada Aplikasi Chatting Berbasis HTML 5 WebSocket, http://labti.ukdw.ac.id/ojs/index.php/informatika/article/viewFile/326/pdf_1, diakses 8 September 2016

Gelens, J., Bourget, A., & Anderson, J., 2014, *gevent-socketio Documentation*.(Release0.3.1),<https://media.readthedocs.org/pdf/gevent-socketio/latest/gevent-socketio.pdf>, diakses 9 September 2016

Huston Tom, 2015, *What are microservice?*, <http://smartbear.com/all-resources/articles/what-is-microservices-architecture/>, diakses pada 27 September 2016 pukul 14:00 WIB.

Kurniawan Agus, 2014 *Node.js Succintly.[Online]*.Aerial Center Parkway Suite 200 Morrisville, NC 27560 USA: Syncfusion Inc. Tersedia di <https://www.syncfusion.com/resources/techportal/ebooks>, diakses pada tanggal 27 September 2016 pukul 14:00 WIB.

Putra Adi Candra, 2015, *Pengantar databases NoSQL dan MongoDB*, <http://www.candra.web.id/2014/02/27/pengantar-database-nosql-dan-mongodb/>, diakses pada tanggal 27 September 2016 pukul 14:36 WIB.

Roni Setiawan & Edhy Sutanta, 2009, *Membangun Aplikasi Chating Berbasis Multiuser*,<http://p3m.amikom.ac.id/p3m/dasi/2010/DASIMaret2009/3%20-%20STMIK%20AMIKOM%20YOGYAKARTA%20-%20MEMBANGUN%20APLIKASI%20CHATING%20BERBASIS%20MULTIUSER.pdf>, diakses 20 September 2016