

# Pemanfaatan Mean Stack Dalam Digitalisasi Administrasi Tugas Akhir Menggunakan Kombinasi Iteratif dan Scrum Model *by Rita Wijaya*

---

**Submission date:** 28-Apr-2022 09:00AM (UTC+0700)

**Submission ID:** 1822399532

**File name:** evani\_Pemanfaatan\_Mean\_Stack\_dalam\_Administrasi\_Tugas\_Akhir.docx (709.89K)

**Word count:** 3132

**Character count:** 20259

## Pemanfaatan Mean Stack Dalam Digitalisasi Administrasi Tugas Akhir Menggunakan Kombinasi Iteratif dan Scrum Model

Nina Sevani<sup>1</sup>, Rita Wiryasaputra<sup>2</sup>, Jeremy Wijaya<sup>3</sup>, Vini Janti Angellica<sup>4</sup>

<sup>1,2,3,4</sup>Program Studi Informatika, Fakultas Teknik dan Ilmu Komputer,  
niversitas Kristen Krida Wacana,  
<sup>4</sup>Tanjung Duren Raya No. 4, Jakarta

<sup>1,2,3,4</sup>Email: <sup>1</sup>nina.sevani@ukrida.ac.id, <sup>2</sup>Rita.wiryasaputra@ukrida.ac.id,  
<sup>3</sup>jeremy.2017tin005@civitas.ukrida.ac.id, <sup>4</sup>vini.2017tin003@civitas.ukrida.ac.id

<sup>1</sup>(**received:** dd-mm-yy, **revised:** dd-mm-yy, **accepted:** dd-mm-yy (diisi oleh editor))  
(kosong 2 spasi tunggal)

### Abstract

*The image of an organization is reflected in the good administration of an organization that involves leadership, policies, and human relations. In era of digitalization, the administration processing such as writing, duplicating, recording documents transforms more transparent and more easy than before. If the department as a part of the university has manual administration, which trigger many problems as the impact of the complexity of Undergraduate Student's Final Project administration such as human error, the diversity of formating documents, long run communication with complicated coordination among elements in institution. This circumstances influence the decision maker to make the right choice due to limited time, but they can be minimalized by the digitalization administration based on web with combining the Traditional Iterative Model with the Scrum Model. The web development using MEAN (MongoDB, Express, Angular and NodeJS) technology to accommodate the administration software as a SPA (Single-Page Application) which using JavaScript language programming in both client-side and server-side. This cutting-edge technology will gain the adaptive system with a small work team and affordable cost.*

**Keyword:** administration, digitalisation, iterative model, MEAN stack, Scrum

### Abstrak

Citra suatu organisasi tercermin dari baiknya penyusunan administrasi organisasi yang melibatkan kepemimpinan, kebijakan, dan hubungan antar manusia. Kegiatan administrasi mencakup kegiatan pengendalian informasi yaitu tulis menulis/mencatat, menggandakan, menyimpan, dimana kegiatannya bertransformasi menjadi lebih praktis dan transparansi pada era digitalisasi. Program studi sebagai bagian dari perguruan tinggi menghadapi kompleksitas administrasi Tugas Akhir (TA) mahasiswa yang menimbulkan beberapa permasalahan antara lain *human error*, keberagaman formating berkas, komunikasi panjang dan koordinasi bertingkat antar unsur dalam perguruan tinggi jika proses administrasi tersebut dikerjakan secara manual. Permasalahan berdampak pada terganggunya pengambilan keputusan. Akan tetapi hal tersebut dapat diminimalisir dengan pendigitalisasian administrasi TA mahasiswa berbasis web dengan teknologi *MEAN (MongoDB, Express, Angular, dan NodeJS)* stack menggunakan metodologi kombinasi antara *Traditional Iterative* model dengan Scrum model. Pemanfaatan modern teknologi *MEAN STACK* mengakomodir administrasi TA sebagai sebuah SPA (*Single-Page Application*) yang menggunakan bahasa pemrograman JavaScript baik dari *client-side* maupun *server-side*. Upaya ini dilakukan agar perangkat lunak yang dihasilkan dapat lebih adaptif atas perubahan sistem, namun biaya pengembangan terjangkau dan pengendalian dapat dilakukan pada setiap tahap secara transparan dalam sebuah tim kerja kecil.

**Kata Kunci:** administrasi, digitalisasi, iterative model, MEAN stack, Scrum

## I. Pendahuluan

Citra suatu organisasi tercermin dari baiknya penyusunan administrasi organisasi yang melibatkan kepemimpinan, kebijakan, dan hubungan antar manusia. Kegiatan administrasi mencakup kegiatan pengendalian informasi yaitu tulis menulis/mencatat, mengandakan, menyimpan, dimana kegiatannya bertransformasi menjadi lebih praktis dan transparansi pada era digitalisasi yang merambah lini kehidupan manusia. Kegiatan administrasi dalam suatu organisasi akan menjadi sumber informasi yang kelak dipergunakan dalam proses pengambilan keputusan, dimana komunikasi koordinasi dan kerjasama akan didukung secara lebih baik. Dalam organisasi perguruan tinggi, untuk menghasilkan lulusan perguruan tinggi yang baik dalam *hardskill* dan *softskill*, maka diperlukan kerjasama antar unsur untuk mengatasi kompleksitas administrasi yang ada dalam berbagai aspek [1]. Salah satunya adalah kompleksitas administrasi Tugas Akhir (TA) mahasiswa juga dihadapi oleh suatu program studi yang merupakan bagian dari perguruan tinggi dan menimbulkan beberapa permasalahan yaitu pertama, *human error* yang meliputi kesalahan dalam pencatatan, kesalahan dalam penyimpanan berkas, kesalahan dalam melakukan unggah berkas. Kedua adalah beragamnya formatting berkas, permasalahan ketiga adalah terjadinya komunikasi panjang dan koordinasi bertingkat antar unsur dalam perguruan tinggi yang berujung pada terganggunya waktu pengambilan keputusan. Solusi permasalahan adalah dengan pendigitalisasian proses administrasi TA mahasiswa, sehingga 4 (empat) manfaat yang dapat dipetik adalah percepatan koordinasi antar unsur yang terlibat dalam proses TA (mahasiswa-dosen-koordinator TA), meminimalisasian *human error* dalam pengarsipan TA, pemusatan pengarsipan dengan lebih terstandar dan penghematan biaya kertas.

Pendigitalisasian berkaitan erat dengan pengembangan perangkat lunak, yang merupakan pekerjaan yang tidak mudah dimana kemungkinan kegagalan sistem dapat terjadi sebagai dampak adanya tugas yang tidak lengkap dalam sistem. Penerapan metodologi dan model yang tepat dalam proses pembuatan perangkat lunak terbukti dapat membantu menjaga kualitas dan waktu pengembangan perangkat lunak [2][3][4]. Metodologi ini juga dapat membantu mengatasi berbagai kendala dalam pengembangan perangkat lunak, seperti *human error*, perbedaan kompetensi dari anggota tim, serta kondisi dari masing-masing anggota tim pembuat program dan juga kondisi dari lingkungan tempat kerja [2][3]. Fase-fase dalam metodologi dan model pembuatan perangkat lunak, antara lain akan membentuk manajemen tim untuk alokasi tugas kepada anggota tim, proses konfigurasi anggota tim, pengontrolan, serta mengatasi konflik yang mungkin terjadi [2][4]. Diperlukan adaptasi dengan pendekatan yang lebih baik untuk menciptakan sistem dengan perangkat lunak yang menjadi bagian dari kehidupan manusia [3]. Pergeseran dalam pengembangan perangkat lunak juga terjadi dewasa ini, yaitu dari metodologi tradisional menjadi metodologi yang *agile*. Salah satu metodologi pengembangan *Agile* yang adaptif adalah *Scrum framework*, dimana pada *framework* tersebut pengguna sistem dilibatkan dalam siklus pengembangan yang bertingkat dan berulang [5].

Tetapi, tidak banyak penelitian terkait dengan adaptasi aktual dan modifikasi sistem yang mengimplementasikan *Scrum framework* sesuai dalam kebutuhan dunia nyata. Meskipun hal tersebut merupakan solusi atas kelemahan metode dalam batasan kontekstual yang spesifik. Lebih dari setengah teknik *Agile* menggabungkan *Scrum* dan kombinasi antara *Scrum* dengan teknik lainnya [6] sedangkan *Traditional Iterative* model merupakan salah satu model tradisional dalam pengembangan perangkat lunak [4].

Sistem administrasi digital TA berbasis web pada penelitian ini, menyederhanakan koordinasi antar pihak yang terkait dalam proses TA dan pengarsipan terpusat dengan penggabungan metodologi *Traditional Iterative* model dengan *Scrum* model. Upaya penggabungan metodologi dengan maksud agar perangkat lunak yang dihasilkan dapat lebih adaptif dalam durasi waktu yang lebih cepat, namun dengan biaya yang terjangkau dan pengendalian dapat dilakukan pada setiap tahap secara transparansi dalam sebuah tim kerja kecil dan tanggap terhadap adanya perubahan dalam bentuk apapun. Durasi kecepatan proses pengembangan perangkat lunak didukung oleh teknologi Google yang mengembangkan mesin V8 JavaScript yang mampu mengkompilasi dan mengeksekusi sumber kode JavaScript, menangani pengalokasian memori dan keakuratan manajemen memori yang cukup penting dalam performansi tinggi V8. *MEAN* (*MongoDB*, *Express*, *Angular*, dan *NodeJS*) *stack* merupakan gaya pengembangan web populer yang menggunakan mesin V8 secara efektif dengan memanfaatkan kelebihan ekosistem paket Node yang merupakan ekosistem terbesar dalam kumpulan *open source* [7]. Pemanfaatan modern teknologi pengembangan web *MEAN stack* mengakomodir pendigitalisasian administrasi TA sebagai sebuah SPA (*Single-Page Application*) *fullstack* yang dimana *Mongo* sebagai manajemen basis data, *Express* pada sisi *backend* sebagai *Hypertext Transfer Protocol* (*HTTP*) server *framework* fleksibel yang melayani aplikasi *Angular* di sisi *frontend*, dan *Node* yang menggunakan bahasa pemrograman JavaScript baik dari sisi *client-side* maupun *server-side*.

## II. Metodologi Penelitian

Penelitian mengadopsi lima fase metodologi pengembangan perangkat lunak yang terdiri dari [4]:

### a. *Planning*

Tahap *planning* merupakan tahap awal dalam pengembangan perangkat lunak yang meliputi penentuan masalah yang akan diatasi, termasuk pembentukan tim dengan sumber daya manusia terbatas dan penentuan kebutuhan durasi waktu pengembangan serta pengumpulan kebutuhan data primer dan data sekunder yang digunakan pada tahap analisa. Bentuk data primer diperoleh melalui teknik observasi, wawancara dan diskusi dengan koordinator TA yang bertugas di program studi, dengan mahasiswa yang mengambil TA, dan dengan dosen pembimbing TA maupun dosen penguji TA. Bentuk data sekunder yang dikumpulkan berupa Standard Operasional Prosedur (SOP) TA yang berlaku, panduan TA yang digunakan, formulir penilaian dan contoh bentuk hasil TA yang dikumpulkan oleh mahasiswa.

### b. *Analysis*

Bagian yang dilakukan dari tahap *Analysis* adalah *brainstorming* antara anggota tim untuk membahas lebih dalam tentang masalah yang ditentukan pada tahap *planning*, termasuk strategi penyelesaian masalahnya. Tim pengembang juga berkoordinasi dengan pihak program studi guna menganalisa kebutuhan perangkat lunak berbasis web yang akan dibangun, merumuskan kebutuhan berdasar proses sistem TA yang berjalan, kondisi lingkungan dan kebutuhan pengguna yang akan mengakses perangkat lunak tersebut baik selaku mahasiswa peserta TA, dosen pembimbing, dosen penguji serta koordinator TA.

### c. *Design*

Tahap *Design* berisi abstraksi dari analisa yang telah dilakukan dan meliputi pembuatan diagram yang menjelaskan tentang solusi yang diusulkan dari tahap analisa. Diagram yang akan dibuat antara lain *Use Case diagram* dan *Activity diagram*. Schema terkait basis data non relational. Dalam tahap design juga akan memuat rancangan basis data dan penggambaran tampilan *frontend* perangkat lunak berbasis web yang akan dibuat.

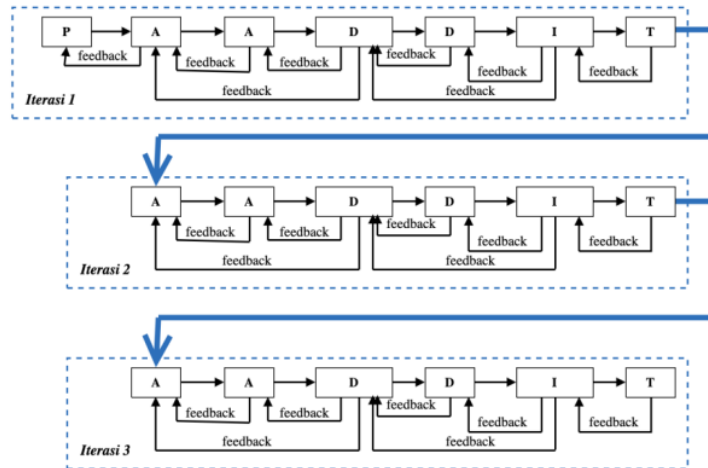
### d. *Implementation*

*Implementation* dalam penelitian menggunakan pengembangan web berteknologi *MEAN stack* untuk menghasilkan website yang deklaratif. *MEAN stack* model mendayagunakan beberapa JavaScript *framework* baik pada sisi bagian *backend* hingga sisi *frontend*.

### e. *Testing*

Tahap testing merupakan tahap untuk memastikan bahwa perangkat lunak yang dibuat dapat berfungsi dengan baik. Penelitian menggunakan *blackbox testing* untuk memastikan apakah perangkat lunak yang dihasilkan dapat berjalan sesuai dengan fungsinya tanpa perlu mengetahui *code* program [8][9]. Dalam *blackbox* ini, pengujian dilakukan untuk setiap *test case* yang dihasilkan dari deskripsi eksternal perangkat lunak [10].

Selain lima fase dalam metodologi pengembangan sistem, penelitian ini juga menggunakan kombinasi model pengembangan sistem, yaitu *Traditional Iterative* model dengan Scrum model. Gambar 1 menggambarkan keterkaitan metodologi dan model pengembangan perangkat lunak yang digunakan pada penelitian ini.



Keterangan:  
 P= fase Planning,  
 A= fase Analysis,  
 D= fase Design,  
 I= fase Implementation,  
 T= fase Testing yang menjadi bagian dari maintenance.

Gambar 1. Kombinasi Model Iteratif dan Scrum

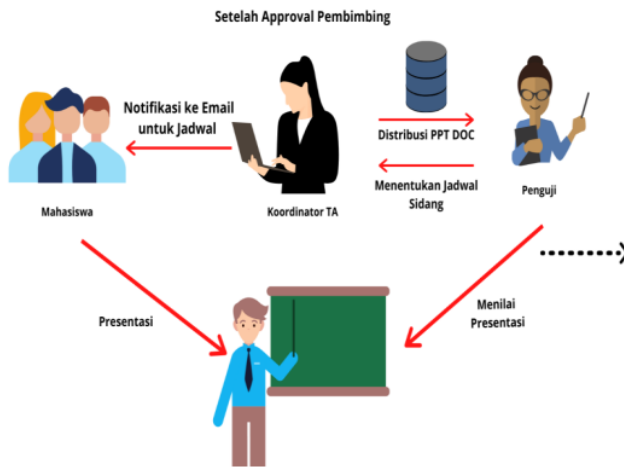
Dalam setiap iterasi, fase-fase yang diperlukan akan tetap dikerjakan. Proses iterasi dilakukan sampai durasi waktu selesai atau perangkat lunak dianggap sudah berfungsi dengan baik sesuai dengan perencanaan awal. Umpanbalik yang fleksibel dikomunikasikan secara regular disiapkan pada tiap fase yang dilakukan oleh setiap blok iterasi.

### III. Hasil dan Pembahasan

Pembahasan akan dimulai dengan melihat pada proses bisnis yang ada saat ini dan menjadi landasan bagi dilakukannya sistem digitalisasi administrasi yang diusulkan. Selama ini, pihak Koordinator TA melakukan penentuan dosen pembimbing dan dosen penguji secara manual sehingga subjektivitas mempengaruhi penentuan ini. Mahasiswa mengumpulkan berkas TA dalam file berformat dokumen ppt dan docs, selanjutnya Koordinator TA berperan untuk mereview file-file tersebut. Sedangkan dosen pembimbing menerima distribusi file dan setelah proses bimbingan terjadi maka menyetujui mahasiswa mengikuti sidang. Proses bisnisnya ditunjukkan pada Gambar 2. Proses bisnis selanjutnya adalah Gambar 3, dimana Koordinator Ta akan menginformasikan mahasiswa tentang jadwal sidang dan mendistribusikan file yang telah dikumpulkan ke dosen penguji. Hanya mahasiswa yang sudah mengumpulkan berkas lengkap yang diverifikasi oleh koordinator TA dan sudah mengisi KRS TA sebagai syarat administrasi yang akan mengikuti proses bisnis ini.



Gambar 2. Proses Bisnis 1



Gambar 3. Proses Bisnis 2

Ketika proses sidang telah selesai, maka **17** koordinator TA akan mengumpulkan nilai baik dari dosen pembimbing, maupun dosen penguji TA. Proses ini **ditunjukkan pada Gambar 4.**



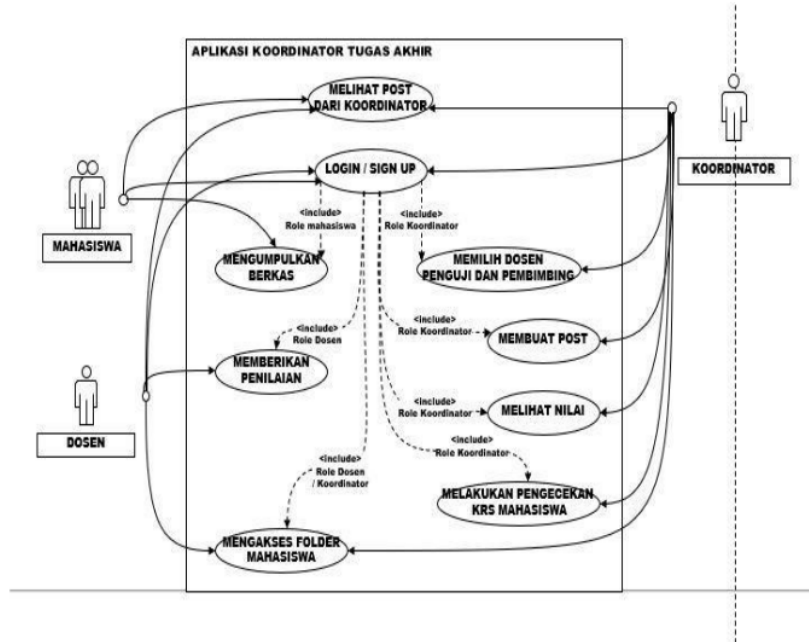
Gambar 4. Proses Bisnis 3

Permasalahan yang ada pada proses bisnis saat ini adalah karena alurnya yang panjang dan membutuhkan verifikasi dari beberapa pihak, dimana seluruh proses verifikasi masih dilakukan semi manual melalui koordinator

TA. Hal ini dapat menimbulkan delay yang cukup lama serta cukup rentan akan *human error*, apalagi pada saat jumlah mahasiswa TA yang cukup banyak. Oleh karena itulah diperlukan sistem digitalisasi administrasi TA yang dapat memotong proses antrian yang bisa panjang dan mempermudah verifikasi dengan tetap memastikan keamanan berkas TA. Untuk memperjelas alur dari sistem digitalisasi administrasi TA yang diusulkan, maka dibuatlah *Use Case diagram*, *Activity diagram*, serta *Schema* dari sistem yang dibuat.

A. *Use Case Diagram*

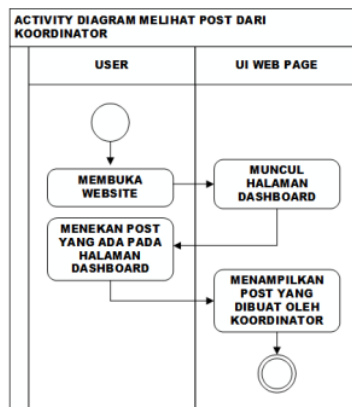
*Use case diagram* mencerminkan hasil analisis atas penggambaran kebutuhan sistem, dimana aktor dalam sistem digitalisasi administrasi TA terbagi menjadi 3 (tiga) *role* pengguna yaitu *user*, admin (koordinator), dan admin kedua (Dosen Pembimbing dan Dosen Penguji), digambarkan pada Gambar 5 dengan hak akses proses yang berbeda-beda.



Gambar 5. *Use Case Diagram*

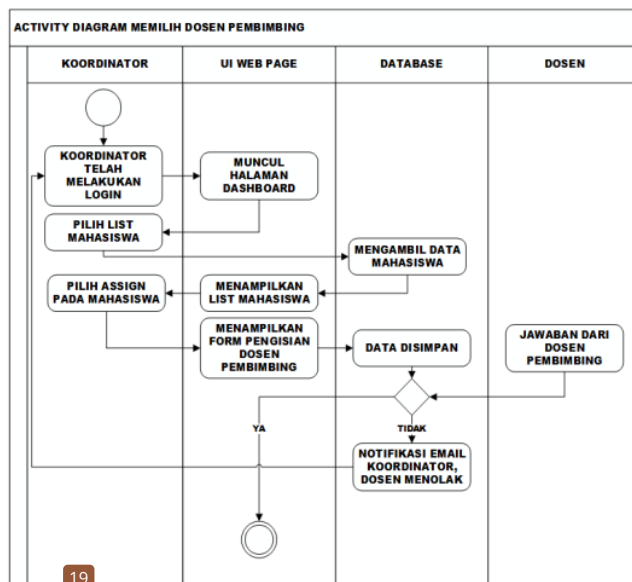
B. *Activity Diagram*

*Activity diagram* merupakan representasi perilaku sistem, dimana aktivitas pertama pengguna adalah validasi login untuk memasuki sistem. Sistem juga memfasilitasi perilaku aktivitas sign up pengguna, apabila pengguna masih belum terdata dalam repositori yang telah disediakan. Aktivitas pengguna untuk melihat pesan notifikasi dari Koordinator TA digambarkan pada Gambar 6. Koordinator TA juga dapat melakukan posting informasi penting untuk seluruh mahasiswa melalui sistem ini.



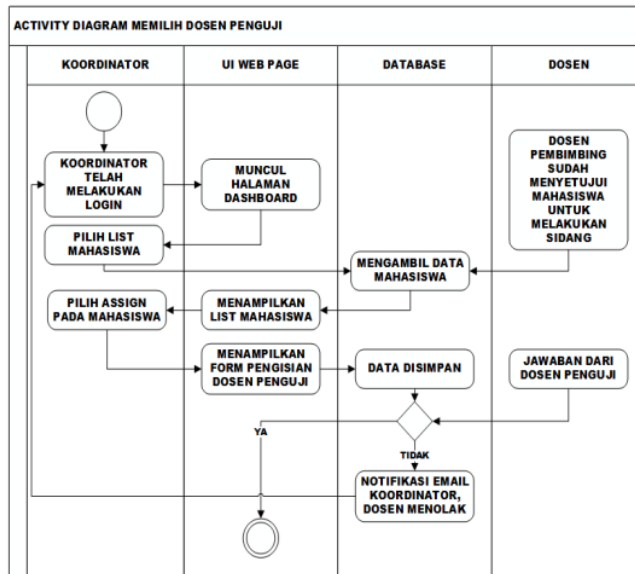
Gambar 6. Activity Diagram Post Koordinator

Seluruh dokumen mahasiswa akan disimpan dalam folder masing-masing sesuai nama mahasiswa peserta TA. Dosen penguji TA, dosen pembimbing TA, dan juga koordinator TA dapat mengakses langsung dokumen yang diperlukan pada setiap folder mahasiswa tersebut. Komunikasi dan mekanisme proses pemilihan dosen penguji dan dosen pembimbing dapat dilakukan melalui sistem seperti yang digambarkan pada Gambar 7 dan Gambar 8. Dosen penguji dan pembimbing juga dapat melakukan input penilaian terkait TA melalui sistem.



19

Gambar 7. Activity Diagram Pemilihan Dosen Pembimbing



Gambar 8. Activity Diagram Pemilihan Dosen Penguji

C. Schema

MongoDB merupakan salah satu manajemen sistem basis data non relational yang menggunakan konsep key-value dengan pemahaman bahwa setiap dokumen pasti memiliki key. Mongoose digunakan sebagai Object Document Model (ODM) yang memetakan objek pada basis data MongoDB dengan model JavaScriptnya. Schema user, dosen dan koordinator pada Gambar 9 berfungsi untuk menyimpan data profil dari setiap user, dosen dan koordinator yang login maupun sign up. Nantinya data tersebut akan masuk ke dalam basis data MongoDB dalam bentuk Schema. Setiap melakukan login akan ada status yang akan menentukan user akan pindah ke dashboard yang dituju.

```
const mongoose = require("mongoose");
const uniqueValidator = require("mongoose-unique-validator");

const userSchema = mongoose.Schema({
  email: { type: String, required: true, unique: true},
  password: { type: String, required: true },
  nim: { type:String },
  status: { type: String },
  path: { type: String },
  krs: {type:String},
  judul:{type:String},
  dosen_pembimbing:{type:String},
  dosen_penguji:{type:String},
  dosen_penguji_2:{type:String},
  pending:{type:String}
});

userSchema.plugin(uniqueValidator);

module.exports = mongoose.model("user", userSchema);
```

Gambar 9. Schema User, Dosen, dan Koordinator

Gambar10 merupakan Schema untuk dosen dalam mengupload penilaiannya untuk mahasiswa yang dibimbing maupun diuji. Hasil penilaian dari dosen akan masuk ke database, maka koordinator akan mengetahui dan mengecek nilai mahasiswa beserta keterangan untuk membuktikan kejelasan nilai tersebut.

```
const mongoose = require("mongoose");

const nilai = mongoose.Schema({
  mhsEmail :{type:String},
  dosenEmail :{type:String},
  nilai :{type:String},
  keterangan : {type:String}
});

module.exports = mongoose.model("nilai", nilai);
```

Gambar 10. *Schema* Penilaian Dosen

Meskipun Koordinator TA memiliki kuasa untuk menentukan dosen pembimbing TA dan dosen penguji bagi mahasiswa peserta TA, tetapi koordinator juga tidak sepenuhnya memiliki wewenang memaksa dosen tersebut untuk menyetujuinya. Oleh karena itu dosen difasilitasi rubrik 'answer' dalam *Schema* terkait jawaban (menolak atau menerima) penunjukkan dosen oleh Koordinator, seperti terlihat pada Gambar 11.

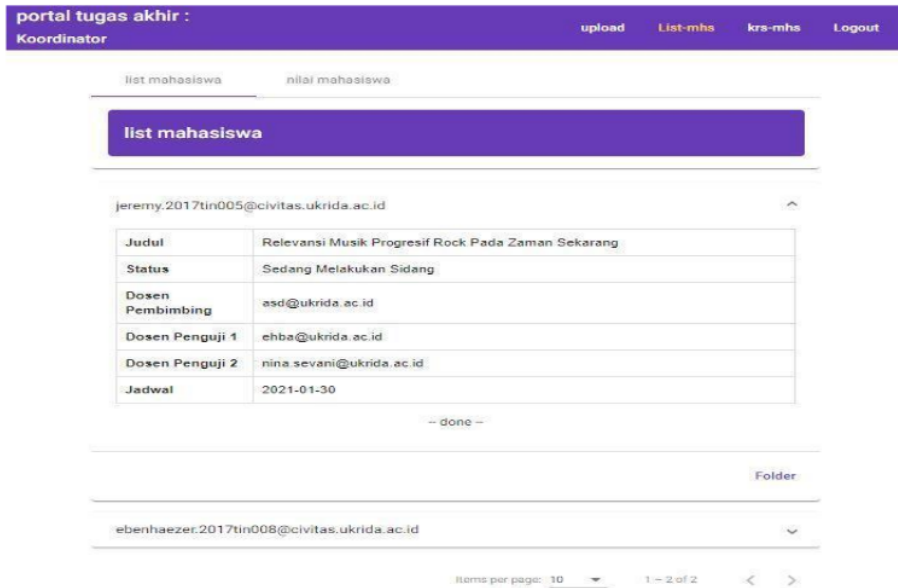
```
const mongoose = require("mongoose");

const assign = mongoose.Schema({
  mhsEmail :{type:String},
  dosenId :{type:String},
  status :{type:String},
  answer : {type:String},
  judul: {type:String}
});

module.exports = mongoose.model("Assign", assign);
```

Gambar 11. *Schema* Persetujuan Dosen Pembimbing dan Penguji

Implementasi sistem dilakukan menggunakan JavaScript yang mendasari pemrograman *MEAN Stack*. *MEAN Stack* akan memproses setiap lapisan, dimulai dari pengaksesan basis data, pembuatan RESTful API, pengelolaan routing, hingga implementasi Single Page Application (SPA). NodeJS memiliki komponen yang digunakan untuk mengakses basis data Mongoose. Gambar 12 merupakan contoh tampilan pada halaman Koordinator, dimana terdapat data mahasiswa beserta nama pembimbing dan penguji yang sudah disetujui oleh dosen yang bersangkutan.



Gambar 12. Implementasi Halaman List Mahasiswa-Dosen Koordinator



Gambar 13. Notifikasi Penolakan Dosen

Dosen dapat memberikan penolakan atas bimbingan dan pengujian melalui aplikasi, dan secara otomatis notifikasi penolakan tersebut akan masuk dalam email yang dikelola oleh koordinator seperti pada Gambar13. Selanjutnya koordinator TA dapat mengelola kembali penentuan dosen pembimbing atau dosen penguji pada mahasiswa yang bersangkutan. Dosen juga dapat melihat list mahasiswa bimbingan seperti pada Gambar14.



Gambar 14. Implementasi halaman list mahasiswa-dosen

Setelah implementasi, sistem juga diuji menggunakan *blackbox testing* yang dilakukan oleh tim pengembang. Pengujian ini diperlukan untuk melihat apakah sistem yang dikembangkan sudah siap dipakai, serta sesuai dengan tujuan pembuatannya, dan hal ini dapat dilakukan dengan menyiapkan sejumlah *test case* untuk melihat validitas hasilnya [11]. Oleh karena itu, pengujian pada sistem ini difokuskan pada input dan output sistem, serta tidak menekankan pada proses kontrol yang dilakukan dalam sistem [8]. Tabel 1 berikut merupakan hasil pengujian untuk beberapa *test case*.

Tabel 1. Hasil Uji *Blackbox*

Test Case	Deskripsi Pengujian	Hasil yang Diharapkan	Hasil	Kesimpulan
Halaman Login	Mengetikkan email, dan password tidak diisi atau kosong kemudian klik tombol Login	Sistem akan menolak dan menampilkan pesan "Password belum diisi"	Sesuai harapan	Berhasil
Halaman Post Koordinator	Mengetikkan Title, content dan tempat upload file diisi kemudian klik tombol submit	Sistem akan menyimpan data pindah ke halaman utama	Sesuai harapan	Berhasil
Halaman Pilih Dosen Pembimbing	Memilih dosen kemudian klik tombol submit	Sistem akan menyimpan data pindah ke halaman utama	Sesuai harapan	Berhasil
Halaman Mahasiswa Mengumpulkan Berkas	Mengumpulkan berkas file yang akan diuji dan dinilai disebut kolom 'choose file'	File akan otomatis masuk ke sebuah folder dan terdistribusi ke sistem koordinator	Sesuai harapan	Berhasil
Halaman Pengisian Nilai	Mengisi Form hasil penilaian Mahasiswa yang diuji	Dosen dapat mengisi penilaian di form tersebut, lalu nilai bisa masuk ke sistem Koordinator	Sesuai harapan	Berhasil

#### IV. Kesimpulan

Pendigitalisasian administrasi TA dengan pengembangan web berteknologi *MEAN (MongoDB, Express, Angular, dan NodeJS) stack* dengan menggunakan JavaScript pada sisi *backend* dan *frontend* mengakomodir penyederhanaan dan transparansi koordinasi bertingkat antar mahasiswa peserta TA, dosen pembimbing TA, dosen penguji TA dan koordinator TA. Dampak lain dari pendigitalisasian administrasi TA, maka keputusan yang tepat dapat diambil mengingat terpusatnya media penyimpanan data dan teroganisirnya keseragaman format dokumen.

## Referensi

- [1] Muhyadi, "Kajian Ilmu Administrasi," *Efisiensi, Kaji. Ilmu Adm.*, vol. XIV, no. 1, pp. 98–109, 2016.
- [2] R. Jain and U. Suman, "A Project Management Framework for Global Software Development," *ACM SIGSOFT Softw. Eng. Notes*, vol. 43, no. 1, pp. 1–10, 2018.
- [3] R. Datta and H. U. Sharif, "An integrated approach to software engineering," *Int. J. Adv. Electron. Comput. Sci.*, vol. 7, no. 6, pp. 30–35, 2020.
- [4] S. Nugroho, S. Hadi, and L. Hakim, "Comparative Analysis of Software Development Methods between Parallel, V-Shaped and Iterative," *Int. J. Comput. Appl.*, vol. 169, no. 11, pp. 7–11, 2017.
- [5] M. B. Legowo, B. Indiarjo, and D. Prayitno, "Implementation of Scrum Work Framework in the Development of Quality Assurance Information System," *J. Penelit. Pos dan Inform.*, vol. 9, no. 2, p. 125, 2019.
- [6] M. Hron and N. Obwegeser, "Scrum in practice: An overview of Scrum adaptations," *Proc. Annu. Hawaii Int. Conf. Syst. Sci.*, vol. 2018–January, no. February, pp. 5445–5454, 2018.
- [7] N. Nirgudkar and P. Singh, "The MEAN Stack," *Int. Res. J. Eng. Technol.*, pp. 2395–56, 2017.
- [8] T. Snadhika Jaya, "Pengujian Aplikasi dengan Metode Blackbox Testing Boundary Value Analysis (Studi Kasus: Kantor Digital Politeknik Negeri Lampung)," *J. Inform. J. Pengemb. IT*, vol. 03, no. 02, pp. 45–48, 2018.
- [9] F. C. Ningrum, D. Suherman, S. Aryanti, H. A. Prasetya, and A. Saifudin, "Pengujian Black Box pada Aplikasi Perpustakaan Menggunakan Teknik Equivalence Partitions," *J. Inform. Univ. Pamulang*, vol. 4, no. 4, pp. 125–130, 2019.
- [10] P. Ammann and J. Offutt, *Introduction to Software Testing*. Cambridge University Press, 2016.
- [11] A. N. A. Thohari and A. E. Amalia, "Implementasi Test Driven Development Dalam Pengembangan Aplikasi Berbasis Web," *SITECH J. Sist. Inf. dan Teknol.*, vol. 1, no. 1, pp. 1–10, 2018.

## **SURAT PERNYATAAN KEABSAHAN KARYA ILMIAH**

Yang bertanda tangan di bawah ini:

Nama semua Penulis :  
Judul Artikel :  
Bidang Ilmu :  
Fakultas/Program studi :  
Universitas :  
Alamat Universitas :

Dengan ini kami menyatakan bahwa :

1. Artikel yang telah diserahkan ke Jurnal Ilmiah ilmiah FIFO p-ISSN: 2085-4315 e-ISSN: 2502-8332 adalah benar karya ilmiah kami atau bukan plagiat hasil karya orang lain
  2. Apabila dikemudian hari terbukti bahwa karya ilmiah ini bukan karya ilmiah kami sendiri atau plagiat hasil karya orang lain, maka kami bersedia menerima sanksi sesuai ketentuan perundang-undangan yang berlaku.
  3. Artikel kami pada saat ini tidak sedang dipertimbangkan untuk diterbitkan oleh jurnal lain dan tidak akan dikirimkan ke jurnal lain untuk dinilai, pada saat artikel kami sedang dinilai oleh jurnal Ilmiah FIFO.
  4. Jika pada saat proses review, kami menarik/membatalkan artikel maka kami bersedia untuk mengikuti aturan pembatalan artikel yang sudah ditetapkan oleh jurnal ilmiah FIFO.
- Demikian surat pernyataan ini kami buat untuk dipergunakan sebagaimana mestinya.

Kota, tgl/bulan/tahun  
Disetujui oleh Penulis

Materai 10000 & ttd

Nama Penulis (perwakilan)

\* Surat discan dan diupload ke supplementary file

# Pemanfaatan Mean Stack Dalam Digitalisasi Administrasi Tugas Akhir Menggunakan Kombinasi Iteratif dan Scrum Model

## ORIGINALITY REPORT

12%

SIMILARITY INDEX

11%

INTERNET SOURCES

7%

PUBLICATIONS

0%

STUDENT PAPERS

## PRIMARY SOURCES

1	<a href="http://publikasi.mercubuana.ac.id">publikasi.mercubuana.ac.id</a> Internet Source	5%
2	<a href="http://repository.bsi.ac.id">repository.bsi.ac.id</a> Internet Source	1%
3	<a href="http://widuri.raharja.info">widuri.raharja.info</a> Internet Source	1%
4	<a href="http://ejournal.ukrida.ac.id">ejournal.ukrida.ac.id</a> Internet Source	1%
5	<a href="http://etheses.uin-malang.ac.id">etheses.uin-malang.ac.id</a> Internet Source	1%
6	Nina Sevani, Iwan Aang Soenandi, Fajar Saputra. "Implementation of backpropagation artificial neural network for early detection of vitamin and mineral deficiency", IOP Conference Series: Materials Science and Engineering, 2020 Publication	<1%
7	Yuliadi Idrus, Rodianto Rodianto, M. Julkarnaen. "REKAYASA PERANGKAT LUNAK	<1%

APLIKASI INVENTARIS BARANG PADA BADAN USAHA MILIK DESA (BUMDES) BINA MADANI DESA DASAN LEKONG", TEKNIMEDIA: Teknologi Informasi dan Multimedia, 2021

Publication

---

8	<a href="https://repository.upi.edu">repository.upi.edu</a> Internet Source	<1 %
9	Muhammad Sholeh Bathin, Desi Ramayanti. "SOBATHUNI : Aplikasi Rumah Sewa Berbasis Web", Jurnal Edukasi dan Penelitian Informatika (JEPIN), 2019 Publication	<1 %
10	<a href="https://fr.scribd.com">fr.scribd.com</a> Internet Source	<1 %
11	<a href="https://jurnal.lpkia.ac.id">jurnal.lpkia.ac.id</a> Internet Source	<1 %
12	<a href="https://www.neliti.com">www.neliti.com</a> Internet Source	<1 %
13	<a href="https://dokument.pub">dokument.pub</a> Internet Source	<1 %
14	<a href="https://ejournal.unkhair.ac.id">ejournal.unkhair.ac.id</a> Internet Source	<1 %
15	<a href="https://pakarkomunikasi.com">pakarkomunikasi.com</a> Internet Source	<1 %
16	<a href="https://pt.scribd.com">pt.scribd.com</a> Internet Source	<1 %

---

17	<a href="https://repository.uksw.edu">repository.uksw.edu</a> Internet Source	<1 %
18	<a href="https://repository.widyatama.ac.id">repository.widyatama.ac.id</a> Internet Source	<1 %
19	<a href="https://zenodo.org">zenodo.org</a> Internet Source	<1 %
20	Budi Prasetyo, Dimas Fanny Hebrasianto Permadi. "TKJ Laboratory Inventory Monitoring System Using Web-based AHP (Analytical Hierarchy Process) Method at SMK Negeri 1 Doko", <i>Procedia of Engineering and Life Science</i> , 2021 Publication	<1 %
21	<a href="https://www.researchgate.net">www.researchgate.net</a> Internet Source	<1 %

---

Exclude quotes      Off

Exclude matches      Off

Exclude bibliography      On