BAB III

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Mahasiswa yang mengikuti program U-Tapis berjumlah 26 orang yang berasal dari 3 program studi berbeda. Mahasiswa dari program studi Jurnalistik berjumlah 5 orang, mahasiswa dari program studi Sistem Informasi berjumlah 6 orang, mahasiswa dari program studi Teknik Informatika berjumlah 15 orang yang dapat dibagi ke dalam 2 mahasiswa skripsi dan 13 mahasiswa magang. Berikut ini adalah beberapa kegiatan yang dilakukan oleh mahasiswa magang dari masing-masing program studi:

Mahasiswa dari program studi Jurnalistik akan melakukan kegiatan antara lain:

- Membangun korpus data artikel berita mentah dari Tribun melalui pengecekan sampel naskah berita mentah dari redaksi Tribun dan mengidentifikasi kategori kesalahan Bahasa.
- 2. Membuat berita dan dokumentasi kegiatan terkait proses pengembangan aplikasi U-Tapis, uji coba, dan peluncuran aplikasi U-Tapis.
- 3. Mengumpulkan sampel artikel-artikel berita untuk penelitian jurnalistik terkait kesalahan berbahasa dalam karya jurnalistik.
- 4. Menjadi pendamping dalam pelatihan penggunaan aplikasi U-Tapis dalam kegiatan uji coba pada redaksi Tribunnews.com dalam masyarakat umum.

Mahasiswa dari program studi Teknik Informatika akan melakukan kegiatan antara lain:

- 1. Pengembangan pemodelan berbasis Machine Learning untuk aplikasi pengecekan kesalahan Bahasa Jurnalistik dalam artikel berita yang diproduksi rekadsi Tribunnews.com dengan menggunakan komputasi berbasis *Natural Language Processing* (NLP)
- 2. Menyusun panduan penggunaan aplikasi U-Tapis bagi user, redaksi Tribunnews dan masyarakat umum

3. Menyusun dokumentasi pengembangan *user interface* aplikasi U-Tapis yang akan digunakan pada tahap pengembangan aplikasi berikutnya.

Mahasiswa dari program studi Sistem Informasi akan melakukan kegiatan antara lain:

- 1. Pengembangan *user interface* aplikasi U-Tapis berbasis *web application* untuk perangkat PC/Desktop dan berbasis *mobile application* untuk seluler/*smart phone*.
- 2. Menyusun dokumentasi pengembangan *user interface* aplikasi U-Tapis yang akan digunakan pada tahap pengembangan aplikasi berikutnya.

Sebagai mahasiswa dari Program Studi Sistem Informasi maka tugas utamanya adalah menyajikan U-Tapis dalam bentuk situs web yang dapat digunakan oleh pihak reporter/jurnalis bersama dengan editor dari Tribun dalam melakukan proses input dan menyunting berita. Setiap 2 minggu atau 3 minggu sekali, mahasiswa magang dari Prodi Sistem Informasi mengadakan pertemuan via Zoom dengan pihak PT. Tribun Digital Online yang diwakili oleh Bapak Abdullah (Badi) selaku Lead Data Engineer sekaligus supervisi magang untuk memberikan masukan atau saran terkait situs web U-Tapis yang dibuat.

3.2 Tugas dan Uraian Kerja Magang

Mahasiswa Sistem Informasi terdiri dari 6 orang. Masing-masing mengerjakan bagiannya dengan tugas sebagai berikut:

- 1. Bohdi Virya Chandra mengerjakan bagian *back-end* web U-Tapis untuk role editor.
- 2. Naufal Akmal mengerjakan bagian *front-end* web U-Tapis untuk role reporter.
- 3. Owen Billiam mengerjakan bagian *front-end* web U-Tapis untuk role editor.
- 4. Ricardo Cuatanto mengerjakan bagian *back-end* web U-Tapis untuk role.
- 5. Ruddy Kusuma Jaya Mengerjakan bagian *front-end* web U-Tapis untuk role admin.

6. Wilson Manuel (saya) mengerjakan bagian *back-end* web U-Tapis untuk bagian admin.

Berikut ini adalah tugas yang dijalankan oleh mahasiswa selama 100 hari kerja atau setara dengan 800 jam kerja di PT. Tribun Digital Online:

Tabel 3.1 Uraian pelaksanaan magang

l	No	Jenis Pekerjaan	Mulai	Selesai				
Pembahasan Mengenai U-Tapis dan Rencana Pelaksanaan Magang.								
1	1.A	Pengenalan awal U-Tapis dan rencana jalannya proyek tersebut	16 Agustus 2022	16 Agustus 2022				
	1.B	Pembagian tugas proyek U-Tapis untuk mahasiswa magang	17 Agustus 2022	31 Agustus 2022				
Pembelajaran Cara Pembuatan <i>Website</i> dengan Figma, HTML, CSS, Javascript, dan termasuk <i>tools-tools</i> yang digunakan								
2	2.A	Mempelajari front- end dalam sebuah webstie menggunakan HTML, CSS, Javascript	22 Agustus 2022	12 September 2022				
	2.B	Mempelajari tools- tools yang digunakan untuk pembuatan sebuah website	12 Agustus 2022	30 September 2022				
Proses Pembuatan Website U-Tapis dengan Framework Python Flask								
3	3.A	Framework Python Flask untuk pembuatan Back- end website	1 Oktober 2022	20 Oktober 2022				
	3.B	Pembuatan <i>back- end website</i> U- Tapis bagian halaman Admin	20 Oktober 2022	19 November 2022				
Penggabungan Website dengan Database Menggunakan MongoDB Atlas								
4	4.A	Membuat database menggunakan MongoDB Atlas	31 Oktober 2022	4 November 2022				

	4.B	Mengkoneksikan website dengan MongoDB Atlas	31 Oktober 2022	4 November 2022			
Per	enggabungan Website dengan 3 Algoritma Machine Learning						
		Mempelajari		O			
5	5.A	penggunaan API untuk mengakses ketiga algoritma machine learning	5 November 2022	20 November 2022			
	5.B	Penggabungan website U-Tapis dengan algoritma Sintaksis Kalimat	21 November 2022	3 Desember 2022			
	5.C	Penggabungan website U-Tapis dengan algoritma Konjungsi	5 Desember 2022	23 Desember 2022			
	5.D	Penggabungan website U-Tapis dengan algoritma Peluluhan Kata	26 Desember 2022	30 Desember 2022			
Uji	Coba	Website U-Tapis di 3	Tempat				
6	6.A	Melaksanakan Uji Coba U-Tapis di Jakarta	22 November 2022	8 Desember 2022			
	6.B	Melaksanakan Uji Coba U-Tapis di Makassar	10 Desember 2022	10 Desember 2022			
	6.C	Melaksanakan Uji Coba U-Tapis di Solo	10 Desember 2022	10 Desember 2022			
Re	visi W	lebsite U-Tapis					
7	7.A	Melakukan revisi website U-Tapis sesuai dengan masukan dari dosen dan pengguna U-Tapis di 3 tempat uji coba sebelumnya	11 Desember 2022	30 Desember 2022			
		NÚ5	ANIA	KA			

G ② ▲ ▶ ⊕ ⊖ ⊕ 119% -H. T P 2 ⊕ A A Q ① ① 2 /2 d 13 m Q 2 0 Create PDF Combine Files Edit PDF Proposal Usulan Matching Fund A Fill & Sign Gramatikal (Part-of-speech tagging Pembuatan program komputasi untuk analisis teks dan penapis bahasa dengan pendekatan I. metode klasifikasi teks berbasis kecerdasan buatan dengan Natural Language Processing (NLP) Export PDF Comment Comment Scan & OCR Pembuatan User Experience (UX) aplikas analisis teks dan penapi Protect More Tools

3.2.1 Pembahasan Mengenai U-Tapi dan Rencana Pelaksanaan Magang

Gambar 3.10 Pertemuan pertama untuk membahas program U-Tapis

Gambar 3.10 merupakan pembahasan pertama kali mengenai program magang MBKM U-Tapis ini dilaksanakan pada tanggal 16 Agustus 2022 secara daring melalui Zoom. Pertemuan tersebut dihadiri oleh para mahasiswa magang, dosen-dosen terkait, dan juga beberapa perwakilan dari PT. Tribun Digital Online. Bahasan pada pertemuan pertama ini terkait hal-hal secara umum seperti gambaran program U-Tapis, asal mula munculnya program U-Tapis dan perkembangannya, *timeline* secara garis besar mulai dari bulan Agustus hingga Desember, dan tujuan atau hasil akhir yang akan dicapai yaitu dalam bentuk sebuah *website* yang sudah dapat digunakan.



Gambar 3.11 Pertemuan pertama kali dengan Mas Badi selaku pembimbing lapangan/supervisi

Selanjutnya, pada awal bulan September, mahasiswa magang kembali mengadakan pertemuan dengan dosen-dosen yang terlibat dalam program U-Tapis untuk membahas mengenai pembagian tugas yang akan dikerjakan oleh masing-masing mahasiswa magang MBKM. Tugas yang didapatkan oleh mahasiswa magang Program Studi Sistem Informasi adalah pembuatan website dengan menggunakan framework Python Flask. Berikutnya kami juga mengadakan pertemuan secara daring bersama dengan pembimbing lapangan atau supervisi magang yaitu Mas Abdullah (Badi) seperti pada gambar 3.11 yang bekerja sebagai Lead Data Engineer di PT Tribun Digital Online sekaligus membuat jadwal pertemuan setiap 2 minggu sekali.

3.2.2 Pembelajaran Cara Pembuatan *Website* dengan HTML, CSS, Javascript, dan Termasuk *Tools-tools* yang Akan Digunakan

Pembuatan situs web oleh mahasiswa Sistem Informasi sebagai Website Developer meliputi front-end dan back-end. Front-end akan berhubungan dengan pengguna website U-Tapis nantinya secara langsung, misalnya tampilan antar muka pengguna atau disebut juga user interface di mana user melakukan aktivitasnya di situs web tersebut. Oleh karena itu, front-end berhubungan erat dengan estetika dan desain dari situs web yang penting untuk diperhatikan agar dapat memudahkan pengguna dalam menggunakan situs web tersebut [3]. Di sisi lain, back-end akan berfokus pada cara kerja dari situs web tersebut, termasuk di dalamnya seperti bagaimana sebuah situs web bekerja mulai dari cara penanganan permintaan dari front-end yang masuk, mengolahnya, hingga mengirimkan respon balik. Peran back-end dalam hal ini juga tidak kalah penting mengingat tampilan antarmuka pengguna tidak akan berguna apabila tidak ada respon balik yang didapatkan dari proses back-end. Baik front-end maupun back-end keduanya merupakan bidang yang tidak dapat dipisahkan karena masing-masing memiliki peran yang saling mendukung demi terciptanya sebuah situs web yang baik.



Gambar 3.12 3 pilar utama dalam front-end sebuah website

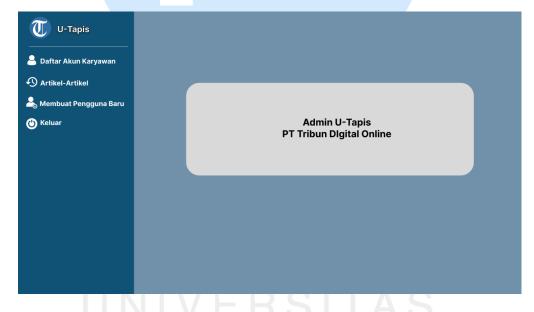
Seperti pada gambar 3.12, front-end sebuah website terdiri dari HyperText Markup Language (HTML), Cascading Style Sheet (CSS), dan Javascript. Ketiga hal tersebut merupakan pilar utama dalam pembentukan sebuah website yang baik di mana HTML memiliki peran sebagai penopang utama dari struktur website yang akan dibuat, Javascript berperan sebagai motor penggerak website tersebut agar website terlihat dinamis, dan CSS berperan sebagai komponen yang mempercantik tampilan sebuah situs web. Pada bagian back-end beberapa bahasa yang sering digunakan antara lain PHP, Pyhton, Ruby, dan lainnya yang dapat digunakan sendiri maupun secara bersamaan sekaligus dalam pembentukan sebuah situs web.



Gambar 3.13 Logo Figma

Sebelum masuk ke dalam pembahasan lebih lanjut mengenai *front-end* dan *back-end* maka penting untuk mengetahui rancangan desain secara visual mengenai *website* yang akan dibuat. Terdapat sebuah aplikasi berbasis web yang dapat digunakan oleh pengguna secara gratis tanpa mengeluarkan biaya

untuk mendesain web secara visual. Namanya aplikasi tersebut adalah Figma yang memiliki logo seperti gambar 3.13. Dengan menggunakan Figma, pengguna dapat secara luas mengeksplorasi ide-ide tampilan website yang dimilikinya dan mencoba untuk mendesainnya secara langsung. Hingga saat ini, Figma sudah banyak sekali digunakan termasuk di kalangan mahasiswa UMN yang ingin melakukan desain user interface (UI) dan user experience (UX) terhadap suatu website atau aplikasi seluler. Pada periode awal magang, mahasiswa magang bersama dengan rekan-rekan kerja dari Prodi Sistem Informasi mencoba untuk membuat gambaran awal secara visual website U-Tapis untuk masing-masing peran antara lain halaman reporter, editor, dan admin. Akan tetapi, mahasiswa lebih memfokuskan pada pengerjaan halaman admin saja.



Gambar 3.14 Halaman utama admin

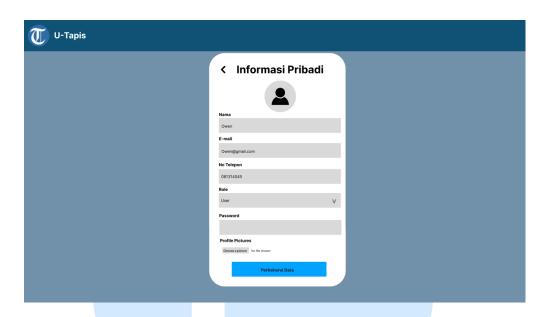
M U L T I M E D I A N U S A N T A R A



Gambar 3.15 Halaman daftar akun

Gambar 3.14 merupakan desain Figma untuk halaman utama pengguna admin. Pada halaman tersebut ditampilkan kata sambutan dengan format "nama admin" dilanjutkan dengan "PT Tribun Digital Online". Berikutnya gambar 3.15 adalah desain Figma untuk halaman daftar akun. Pada halaman tersebut ditampilkan daftar nama-nama akun yang terdaftar di web U-Tapis dalam bentuk tabel. Isi tabel terdiri dari nomor akun, nama akun, email, nomor telepon, dan role atau peran pengguna seperti reporter, editor, atau admin. Terakhir terdapat tulisan "Edit" yang jika ditekan akan mengarahkan pengguna ke dalam halaman edit akun.

UNIVERSITAS MULTIMEDIA NUSANTARA



Gambar 3.16 Halaman edit akun



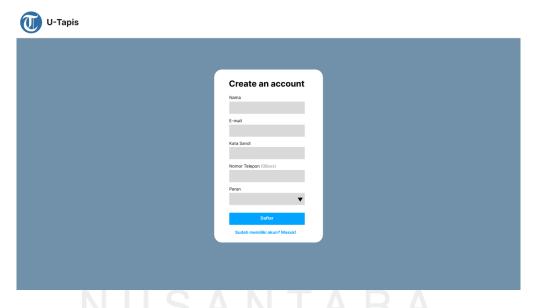
Gambar 3.17 Halaman artikel

Gambar 3.16 merupakan desain Figma untuk halaman edit akun. Pada halaman tersebut admin dapat melakukan perubahan informasi akun seperti nama, email, nomor telepon, role atau peran, dan password atau kata sandi. Jika sudah selesai maka admin hanya perlu menekan tombol "Perbaharui Data". Halaman berikutnya yaitu gambar 3.17 adalah halaman artikel. Halaman ini akan memuat berita-berita yang sudah dimasukkan oleh reporter dan ditampilkan dalam bentuk tabel. Kolom yang terdapat pada tabel antara lain

adalah nomor berita, judul berita, nama reporter yang memasukkan berita, tanggal berita tersebut diunggah, dan status dari berita tersebut apakah sudah di *edit* atau belum. Pada kolom paling kanan tabel terdapat tulisan "Lihat" yang jika ditekan akan mengalihkan pengguna ke halaman lihat artikel.



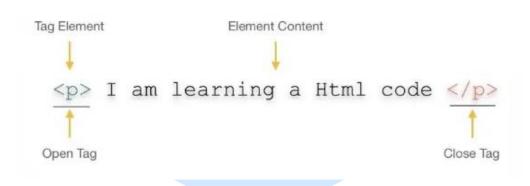
Gambar 3.18 Halaman lihat artikel



Gambar 3.19 Halaman pembuatan akun baru

Gambar 3.18 merupakan halaman lihat artikel yang memungkinkan pengguna dengan peran admin untuk menyunting berita. Disertai dengan

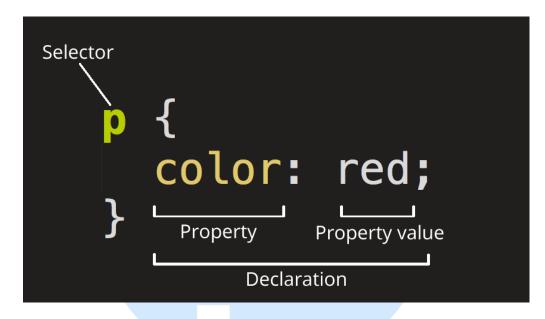
pengecekan terhadap kata-kata yang terdapat di dalam isi berita tersebut melalui algoritma, kemudian akan diberikan tanda dalam bentuk perubahan warna pada kata tertentu apabila terdapat kesalahan. Pengguna dapat menekan tulisan "Perbaharui Teks" untuk melakukan pengecekan ulang jika terdapat perubahan pada isi berita dan juga tulisan "Selesai" apabila pengguna telah selesai menyunting berita. Gambar 3.19 merupakan halaman pembuatan akun baru. Pada halaman tersebut, admin dapat membuat akun baru untuk melakukan login pada web U-Tapis.



Gambar 3.20 Contoh penulisan HTML (rifqimulyawan.com)

Pada bagian *front-end*, bahasa yang digunakan untuk menampilkan sebuah *website* dinamakan sebagai HTML. HTML merupakan kepanjangan dari Hyper Text Markup Language yang akan mendeskripsikan struktur dari sebuah *website* [3]. Struktur dari *website* tersebut dikemas dalam bentuk *tag* pembuka yaitu disimbolkan dengan tanda lebih kecil dari dan diakhiri dengan tanda lebih besar dari (<*>) dan *tag* penutup yang memiliki simbol mirip dengan *tag* pembuka namun ditambahkan garis miring setelah tanda lebih kecil dari (</*>). Sebagai contoh seperti pada gambar 3.20, *tag* dinamakan sebagai *paragraph*. Oleh karena itu, teks yang berada di antara *tag* dan akan dikategorikan ke dalam paragraf yang akan di cetak di *website*. Keberadaan struktur yang jelas penting untuk diperhatikan agar *website* dapat ditampilkan oleh berbagai macam program berbasis web seperti Google Chrome, Internet Explorer, Mozilla Firefox, dan lainnya. HTML tidak cukup

berdiri sendiri, kehadiran CSS sangat diperlukan untuk memberikan visual yang lebih baik di mata pengguna.



Gambar 3.21 Contoh penulisan CSS (developer.mozilla.org)

CSS merupakan kepanjangan dari Cascading Style Sheet. CSS ini bekerja dengan cara membuat aturan-aturan mengenai bagaimana sebuah elemen dalam dokumen HTML akan ditampilkan [4]. Penggunaan CSS sebagai contoh adalah latar belakang sebuah website yang berwarna biru, kata yang dicetak dengan garis miring, jenis fon spesifik yang ingin digunakan pada paragraf tertentu, jarak atau spasi antara dinding halaman website dengan isi konten, dan lain sebagainya. Contoh penulisannya seperti gambar 3.21 adalah menggunakan selector untuk mengidentifikasi bagian di HTML yang dituju kemudian ditambahkan dengan kurung kurawal buka dan diakhiri dengan kurung kurawal tutup. Isi di dalam kurung kurawal tersebut disebut declaration yang terdiri dari properti yang ingin di ubah dan nilainya. Misalnya jika kita ingin mengubah warna konten dalam tag paragraf dari HTML menjadi merah maka kita dapat menuliskan "p {color: red;}".



Gambar 3.22 Logo Javascript

Javascript atau juga sering disingkat JS merupakan sebuah bahasa pemrograman yang dapat membuat sebuah website menjadi lebih dinamis dan memiliki logo seperti pada gambar 3.22. Dinamis yang dimaksud adalah mampu membuat sebuah website menjadi lebih hidup karena adanya interaksi atau hubungan timbal balik antara pengguna dan website itu sendiri [5]. Javascript memiliki fitur yang dinamakan sebagai event handler yang berguna untuk menangani suatu kejadian yang dilakukan oleh pengguna. Sebagai contoh, event "onClick" terjadi apabila pengguna mengeklik tombol yang ditampilkan di sebuah website maka hal yang dilakukan berikutnya yang disebut handler akan menyimpan pesan yang sudah ditulis oleh pengguna ke dalam database. Terdapat banyak sekali event yang dapat digunakan di Javascript. Beberapa event tersebut antara lain adalah "onmouseover" yang akan diaktifkan ketika kursor pengguna berada di atas elemen sebuah HTML, "onmouseout" yang akan dijalankan ketika kursor pengguna berada di luar elemen HTML, "onkeydown" yang akan dijalankan ketika pengguna menekan sebuah tombol pada papan tombol, "onload" yang akan dijalankan ketika sebuah halaman HTML selesai dimuat, dan masih banyak lagi event yang tersedia. Sebagai informasi tambahan, Javascript tidak hanya dapat digunakan untuk keperluan pembuatan website melainkan dapat digunakan juga untuk hal lain seperti pembuatan aplikasi.



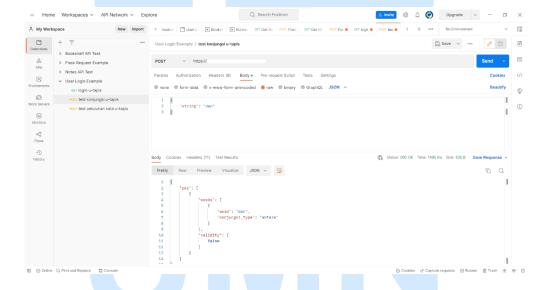
Gambar 3.23 Logo Visual Studio Code

Selain dari bahasa pemrograman yang digunakan untuk membuat sebuah website, dibutuhkan juga beberapa tools atau alat yang dapat digunakan untuk menunjang proses pembuatan website. Salah satu alat yang paling sering digunakan untuk proses pembuatan website adalah Visual Studio Code yang memiliki logo seperti pada gambar 3.23. Visual Studio Code merupakan sebuah aplikasi yang berfungsi untuk membuat, memodifikasi, dan menghapus kode suatu program. Visual Studio Code merupakan produk Microsoft yang saat ini bisa digunakan untuk PC/Desktop dengan operating system Windows, Linux, dan MacOS. Beberapa fitur canggih yang dimiliki oleh Visual Studio Code antara lain adalah menguasai berbagai macam format dan bahasa pemrograman khususnya dalam bidang web development seperti JavaScript, React, HTML, CSS, SCSS, JSON, Python, dan lainnya, IntelliSense yang mampu menyelesaikan kode secara otomatis, penyorotan sintaks, pencocokan tanda kurung, indentasi secara otomatis, bahkan hingga pengaturan jenis fon yang digunakan untuk membuat kode, warna latar belakang aplikasi, dan besaran spasi yang digunakan.



POSTMAN

Gambar 3.24 Logo Postman



Gambar 3.25 Tampilan dashboard aplikasi Postman

Aplikasi berikutnya adalah Postman yang memiliki logo seperti gambar 3.24. Postman dapat di gunakan untuk melakukan uji coba terhadap sebuah website atau API untuk mengetahui apakah sebuah tautan website atau API tersebut dapat merespon request yang dikirmkan dari Postman. Gambar 3.25 merupakan tampilan utama dari aplikasi Postman dan cara untuk menggunakannya adalah pengguna hanya perlu memasukkan tipe request API yang dituju seperti get, post, put, patch, delete, dan yang lainnya. Selanjutnya, pengguna perlu memasukan tautan atau link API yang dituju. Setelah itu, pengguna dapat memasukkan data yang dibutuhkan oleh website atau API

tersebut dan menekan tombol *send* atau kirim. Hasil atau respon balik dari *website* atau API akan ditampilkan dibagian bawah aplikasi.



Gambar 3.26 Logo GitHub

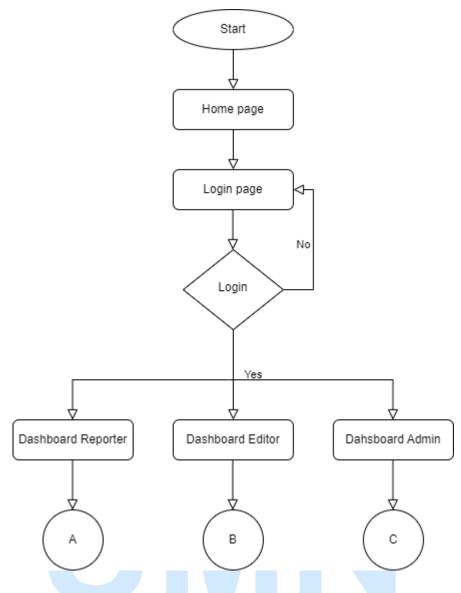
Gambar 3.26 adalah logo dari GitHub. GitHub merupakan sebuah alat dalam bentuk website yang dapat digunakan untuk membantu pekerjaan para developer. Melalui GitHub, pengguna dapat mengunggah dokumen hasil kerjanya dan mengunduhnya kembali di lain waktu. Beberapa keunggulan yang dimiliki adalah GitHub menawarkan fitur pembuatan versi sehingga, setiap kali pengguna mengunggah dokumen kerjanya maka versi yang sebelumnya tidak terhapus begitu saja melainkan tetap tersimpan. Hal tersebut memungkinkan pengguna untuk kembali ke versi tertentu apabila dibutuhkan. Keunggulan berikutnya adalah GitHub memberikan fitur keamanan yang baik. Dokumen kerja yang sudah di unggah dapat dibatasi aksesnya sesuai dengan keinginan pengguna. Misalnya, sebuah dokumen dapat dibatasi aksesnya hanya untuk diri sendiri, dapat diakses oleh beberapa orang yang diijinkan saja, atau dibuka secara publik. GitHub juga dapat digunakan oleh orang-orang yang bekerja dalam satu proyek secara bersamaan sehingga dokumen pekerjaan langsung dapat dikerjakan secara bersamaan dan dokumen-dokumen yang mengalami perubahaan dengan segera dapat diunggah ke GitHub tanpa harus di pindahkan secara manual antar tim.

3.2.3 Proses Pembuatan Website U-Tapis dengan Framework Python Flask

Sesuai dengan namanya, U-Tapis adalah sebuah website yang dibuat dengan tujuan sebagai penapis kesalahan berbahasa Indonesia. Saat ini situs web U-Tapis dikhususkan untuk bidang jurnalistik di PT. Tribun Digital Online dalam proses pembuatan beritanya. Namun, harapan kedepannya U-Tapis juga dapat digunakan untuk menunjang berbagai macam bidang keahlian di Indonesia. Hasil akhir U-Tapis saat ini berupa sebuah situs web yang dapat digunakan oleh reporter/jurnalis dan editor dari PT. Tribun Digital Online untuk membantu mengurangi kesalahan dalam penulisan berita dengan cara mendeteksi kesalahan pada berita yang akan disunting.

Terdapat 3 peran di dalam situs web U-Tapis. Pertama adalah reporter yang hanya memiliki kemampuan untuk memasukkan berita dan melihat hasil akhir dari berita yang sudah di sunting. Kedua adalah editor yang memiliki kemampuan untuk menyunting berita yang sudah dimasukkan sebelumnya oleh reporter. Peran terakhir adalah admin yang memiliki kemampuan untuk menyunting berita, membuat akun baru, menghapus akun, dan mengubah informasi akun. Oleh karena itu, admin sebenarnya memiliki peran yang mirip dengan editor tetapi ditambahkan dengan kemampuan untuk membuat, mengubah, serta menghapus akun.

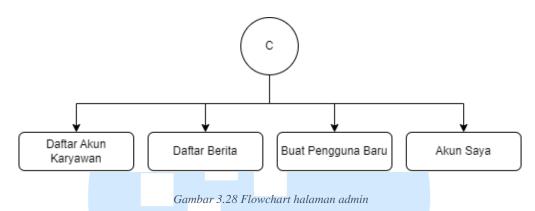
UNIVERSITAS MULTIMEDIA NUSANTARA



Gambar 3.27 Flowchart penggunaan website U-Tapis

Gambar 3.27 adalah gambaran diagram alur dari penggunaan website U-Tapis. Alur aplikasi U-Tapis dimulai dari halaman utama pada saat pengguna pertama kali masuk ke laman U-Tapis. Selanjutnya, pengguna dapat masuk ke halaman login. Apabila login gagal maka pengguna harus mengulangi langkah login dengan menggunakan email dan kata sandi yang sesuai dengan akun yang sudah terdaftar. Setelah itu, pengguna akan diarahkan ke halaman yang sesuai dengan peran yang tersimpan di akun pengguna. Jika pengguna adalah reporter maka pengguna akan diarahkan ke halaman dashboard reporter, jika pengguna

adalah editor maka akan diarahkan ke halaman *dashboard* editor, dan jika pengguna adalah admin maka akan diarahkan ke halaman *dashboard* admin.



arunakan digaram alur yang menggambark

Gambar 3.28 merupakan diagram alur yang menggambarkan jalannya web U-Tapis setelah pengguna admin melakukan login. Pengguna dengan peran admin dapat memilih 4 halaman yang berbeda antara lain halaman Daftar Akun Karyawan, halaman Daftar Berita, halaman Buat Pengguna Baru, dan halaman Akun Saya. Pada halaman Daftar Akun Karyawan, pengguna dapat melihat seluruh daftar akun yang sudah terdaftar di website U-Tapis dalam bentuk tabel. Pengguna juga bisa melakukan perubahan terhadap akun-akun yang sudah terdaftar, baik itu mengubah nama, email, nomor telepon, kata sandi, dan bahkan menghapus akun tersebut. Secara visual, halaman Daftar Berita tidak jauh berbeda dengan Daftar Akun Karyawan. Perbedaan hanya terdapat pada isi tabel yaitu berita-berita yang sudah dimasukkan oleh reporter, baik yang sudah di sunting maupun yang belum di sunting. Halaman Buat Pengguna Baru merupakan tempat di mana pengguna dapat membuat akun baru, baik itu akun reporter, editor, maupun admin. Perlu diingat bahwa pembuatan akun hanya dapat dilakukan oleh pengguna dengan peran admin di halaman ini. Halaman terakhir yaitu Akun Saya merupakan halaman yang memungkinkan pengguna untuk mengubah data pribadinya, seperti nama, email, nomor telepon, dan kata sandi.



Gambar 3.29 Logo Python Flask

Proses pembuatan *back-end* situs web U-Tapis menggunakan bahasa pemrograman Python dengan *framework* Flask yang memiliki logo seperti pada gambar 3.29. Python dipilih oleh tim U-Tapis sebagai bahasa pemrograman dengan tujuan memudahkan penggabungan dengan algoritma yang nantinya akan dibuat oleh mahasiswa magang dari Prodi Informatika. Sementara itu, *framework* Flask dipilih karena menawarkan beberapa keunggulan jika dibandingkan dengan *framework-framework* yang lainnya. Salah satu keunggulannya adalah Flask dikategorikan sebagai *micro-framework* karena bentuknya yang sangat sederhana sehingga, menawarkan kemudahan pagi pengembang untuk menggunakannya [6].

Kemudahan Python Flask untuk digunakan membuat mahasiswa magang tidak memerlukan waktu yang lama untuk mempelajarinya sehingga pengembangan web dapat dengan segera dimulai. Keunggulan berikutnya adalah fleksibilitas dari *framework* Flask sangat tinggi sehingga mampu dimodifikasi sedemikian rupa untuk mengikuti tingkat kompleksitas dari *website* itu sendiri. Fleksibilitas tersebut diperoleh dari desain *framework* Flask yang memiliki banyak sekali ekstensi dalam bentuk paket dan dapat dipilih sesuai dengan keperluan dari masing-masing penggunanya. *Framework* Flask juga menawarkan fitur untuk melakukan *testing* secara langsung pada sebuah *website* sehingga, pengembang web dapat melihat apakah semuanya sudah berjalan dengan baik atau masih ada beberapa kesalahan yang perlu diperbaiki.

```
app = Flask(__name__)

if __name__ == "__main__":
    app.run(debug=True)
```

Gambar 3.30 Contoh komponen pembuka dan penutup pada framework Flask

Penggunaan framework Flask sebagai sarana membangun website U-Tapis memiliki beberapa syarat untuk dapat berjalan dengan baik. Salah satu syaratnya adalah keharusan untuk melakukan instalasi terlebih dahulu framework Flask lewat Python yang sudah dimiliki oleh PC/Desktop mmasing-masing. Setelah itu, mahasiswa harus memiliki sebuah file dengan ekstensi Python yang dapat dijadikan sebagai file utama untuk menjalankan website U-Tapis di mesin pencari dan harus memiliki komponen pembuka dan penutup di dalamnya seperti yang digambarkan pada gambar 3.30. File tersebut tidak harus memiliki nama tertentu tetapi perlu di definisikan terlebih dahulu bahwa file tersebut adalah file utamanya. Berikutnya, file-file dengan ekstensi HTML yang memuat struktur halaman pada website U-Tapis harus dimasukkan ke dalam folder dengan nama "templates" sedangkan, file-file dengan ekstensi Javascript dan CSS harus dimasukkan ke dalam folder dengan nama "static". Kedua hal tersebut wajib dilakukan agar Flask mengetahui di mana letak file-file pendukung yaitu file dengan ekstensi HTML, CSS, dan Javascript berada.

Pembuatan akun hanya dapat dilakukan oleh admin. Oleh karena itu, sebelum seseorang dapat memasuki website U-Tapis, ia perlu meminta admin untuk membuatkan akun terlebih dahulu. Id dari pengguna baru akan dibuat secara acak menggunakan fungsi uuid4 untuk membuat id tersebut unik nilainya. Setelah itu, nama sandi, email, kata sandi, nomor telepon, dan role atau peran akun tersebut akan diinput secara manual oleh admin. Selanjutnya, data akun akan dimasukkan ke dalam database dengan mengenkripsi kata sandi terlebih dahulu menggunakan fungsi encrypt sehingga, kata sandi tidak dapat dibaca oleh manusia.

Sistem *login* pada *website* U-Tapis dilakukan pada file Python dengan melakukan pengecekan terhadap email dan kata sandi pengguna. Fungsi find_one akan digunakan untuk menemukan apakah email yang di tuliskan pada halaman login terdapat di dalam *database*. Apabila tidak ditemukan maka akan muncul notifikasi "Email atau kata sandi salah". Namun, apabila ditemukan maka akan dilanjutkan dengan pengecekan kata sandi. Kata sandi yang terdapat di dalam *database* sudah di enkripsi sehingga untuk membandingkan kata sandi yang dimasukkan dengan kata sandi yang terdapat di dalam *database* dibutuhkan fungsi verify. Jika kata sandi yang digunakan tidak cocok maka notifikasi "Email atau kata sandi salah" akan ditampilkan. Namun, apabila kata sandi sudah tepat maka pengguna akan diarahkan ke dalam halaman dashboard utama sesuai dengan peran pengguna tersebut.

Setelah pengguna menyelesaikan *login* maka sistem akan menyimpan data pengguna tersebut sehingga, pengguna tersebut dapat melakukan aktifitas di dalam web tanpa harus memasukkan ulang data pribadinya. Hal tersebut dapat dilakukan dengan menggunakan fungsi start_session. Sesi akan berhenti apabila pengguna melakukan *logout* atau tidak memiliki aktivitas selama 1 hari sejak melakukan *login* dengan menggunakan fungsi session.clear(). Setelah sesi berhenti maka ketika pengguna membuka kembali *website* U-Tapis, ia harus melakukan *login* terlebih dahulu.

```
@app.route('/daftarberita')
@login_required
def daftarberita():
    all_news = collection2.find()
    return render_template('A_daftarberita.html', all_news=all_news)
```

Gambar 3.31 Contoh penggunaan route dari Python Flask

Sebuah *website* hanya dapat diakses apabila memiliki tautan. Oleh karena itu, tautan harus dibuat terlebih dahulu menggunakan "route" yang diisi dengan tanda garis miring diikuti dengan tautan yang dituju seperti pada gambar 3.31. Misalnya, jika ingin membuat tautan untuk halaman *login* maka contoh kurang lebihnya adalah "@app.route('/login')". Setelah mendefinisikan rute tersebut, di

bawahnya dapat ditambahkan hal-hal apa saja yang ingin dilakukan ketika pengguna *website* mengakses rute untuk *login* tersebut termasuk juga menyebutkan *file* HTML yang akan ditampilkan.

Fungsi-fungsi yang dapat digunakan untuk menampilkan file HTML sebagai struktur web nya adalah render_template, redirect, dan url_for. Fungsi render_template dapat digunakan dengan cara menyebutkan secara langsung file HTML yang dituju. Sementara itu, fungsi redirect dapat digunakan untuk mengarahkan secara langsung ke tautan yang sudah disebutkan sebelumnya. Misalnya apabila route login sudah dibuat sebelumnya dan ingin digunakan kembali maka pengguna hanya perlu memanggilnya kembali dengan "redirect('/login')". Fungsi terakhir yaitu url_for memiliki kemiripan dengan fungsi redirect, bedanya fungsi url_for hanya memerlukan penggunanya untuk memanggil nama route yang sudah di spesifikasi sebelumnya. Sebagai contoh, apabila pengguna ingin mengembalikan halaman website ke halaman daftar berita maka penulisannya adalah "url for('daftarberita')".

Flask menawarkan fitur *testing* yang dapat digunakan secara langsung baik melalui aplikasi Visual Studio Code maupun dari Command Prompt untuk Windows dan terminal untuk MacOS dan linux. Oleh karena itu, untuk melihat perkembangan *back-end* untuk pembuatan web U-Tapis, uji coba secara *local* di laptop pribadi mahasiswa sering dilakukan. Caranya mahasiswa hanya perlu menuliskan "set FLASK_APP=<<iisi nama file utama Python>>" dan "set FLASK_ENV=development" untuk mendefinisikan file yang harus dijalankan ketika *testing* secara *local* dilakukan. Setelah itu, perintah "flask run" dapat segera dijalankan dan *website* U-Tapis dapat dilihat melalui alamat IP localhost yaitu 127.0.0.1:5000 untuk Windows.



Gambar 3.32 Contoh notifikasi/peringatan "Berita berhasil disimpan"

Gambar 3.32 adalah situasi ketika admin selesai melakukan pekerjaan yang akan merubah isi *database* seperti merubah data akun karyawan, mengubah berita, membuat akun baru, dan memodifikasi akun sendiri yang disertai notifikasi atau peringatan di bagian atas konten halaman tersebut. Misalnya pada saat admin menekan tombol "selesai" setelah melakukan *edit* berita maka admin akan dikembalikan ke halaman daftar berita disertai dengan notifikasi berisi "Berita berhasil disimpan". Fitur notifikasi tersebut ditampilkan dengan menggunakan *flash message* yang berfungsi untuk menampilkan pesan di HTML setelah sebuah aksi selesai dilakukan.

Website U-Tapis sudah memiliki penanganan terhadap error atau kesalahan yang mungkin saja terjadi selama pengguna menggunakan website U-Tapis. Bentuk kesalahan yang akan ditampilkan masih tergolong sederhana yaitu error dengan kode 400 dan 500 saja. Kesalahan dengan kode 400 akan ditampilkan apabila terdapat kesalahan dari sisi client atau pengguna. Pada web U-Tapis misalnya seperti pengguna salah memasukkan email atau kata sandi saat melakukan login. Disisi lain, kode kesalahan 500 akan muncul apabila terjadi kesalahan dari sisi server atau penyedia layanan. Misalnya pada web U-Tapis adalah gagal melakukan update berita atau akun.sss

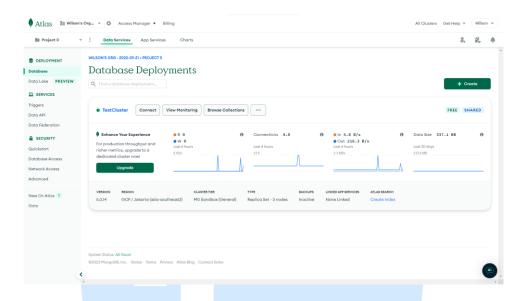
NUSANTARA

3.2.4 Penggabungan *Website* dengan Database Menggunakan MongoDB Atlas



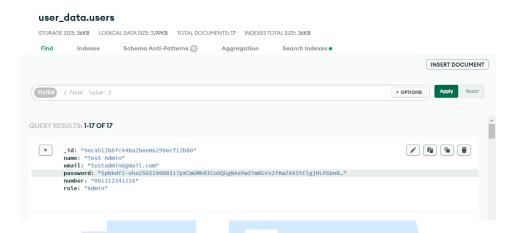
Gambar 3.33 Logo MongoDB

Database NoSQL adalah data yang tidak memiliki struktur sehingga, pengguna dapat menyimpan data dengan berbagai tipe data yang berbeda. Pada tahap awal data yang disimpan dari website U-Tapis hanya berupa teks berita namun, tidak menutup kemungkinan adanya penambahan gambar atau datadata dengan tipe yang lain untuk menunjang kebutuhan penyajian berita yang lebih baik. Bentuk database NoSQL dapat ditemukan pada beberapa layanan seperti MarkLogic, ScyllaDB, Apache Cassandra, Couchbase, MongoDB, dan lain lain. Mahasiswa magang memilih layanan database NoSQL dari MongoDB karena MongoDB menawarkan layanan gratis baik untuk digunakan secara local maupun cloud. Gambar 3.33 merupakan logo dari database MongoDB. Layanan database secara *local* dapat diperoleh dalam bentuk aplikasi PC/desktop dengan nama MongoDB Compass. Melalui MongoDB Compass, pengguna dapat melakukan input data ke dalam database tanpa melalui koneksi internet namun, database hanya bisa diakses melalui masing-masing PC/desktop tempat MongoDB Compass terpasang. Layanan database *cloud* di MongoDB dinamakan MongoDB Atlas yang memungkinkan database untuk diakses oleh seluruh pengguna dari mana saja yang memiliki akses terhadap database tersebut. Oleh karena alasan-alasan tersebut, database MongoDB Atlas dipilih sebagai tempat untuk menyimpan data dari website U-Tapis.



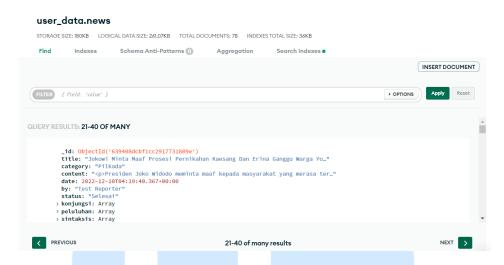
Gambar 3.34 Tampilan dashboard MongoDB Atlas

MongoDB Atlas dapat diakses dengan membuka website mongodb.com dan mengikuti proses login. Setelah login berhasil, pengguna akan diarahkan ke sebuah halaman utama/dashboard dari MongoDB Atlas seperti pada gambar 3.34. Pada halaman dashboard, pengguna akan diberikan informasi mengenai berapa cluster (yaitu tempat di mana database akan dijalankan) yang dimiliki akun tersebut. Selain itu, pada setiap cluster juga akan diberikan informasi berupa grafik mengenai seberapa sering cluster tersebut menerima perintah baca atau tulis, besaran jumlah baca atau tulis yang sudah dilakukan, dan besaran database yang sudah digunakan. U-Tapis hanya menggunakan satu cluster yang diberi nama TestCluster dan dijalankan melalui layanan cloud Google Cloud Platform di Jakarta. Database yang dijalankan pada cluster TestCluster memiliki dinamakan user_data. Di dalam database user_data, terdapat 2 buah collection. Collection adalah cara pengelompokan dokumen yang ingin dibedakan dan dapat diistilahkan sebagai tabel pada database dengan tipe relasional atau disebut juga sebagai database SQL.



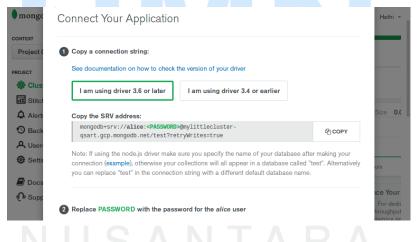
Gambar 3.35 Collection users pada database user_data

Gambar 3.35 merupakan Collection yang pertama dinamakan users karena akan dijadikan tempat penyimpanan seluruh akun atau pengguna yang terdaftar di U-Tapis. Data tersimpan dalam bentuk format JavaScript Object Notation (JSON). JSON adalah format data yang sudah seringkali digunakan untuk peyimpanan atau pengiriman data melalui koneksi internet seperti antar server maupun aplikasi web. Bentuk format JSON memiliki key value di mana key yang dimaksud adalah nama dari datanya dan value adalah isi dari datanya. Sebagai contoh, {"name": "Test Admin"} artinya name adalah key dan Test Admin adalah value. Beberapa key yang terdapat pada collection users adalah id, name, email, password, number, dan role. Key id berisikan nilai unik yang tidak akan sama satu dengan yang lain untuk membedakan setiap data yang terdapat di data akun/pengguna, key name memuat nama akun/pengguna, key email memuat alamat email akun/pengguna, key password memuat kata sandi akun/pengguna tetapi sudah di proteksi dengan enkripsi tambahan sehingga, kata sandi tidak akan terbaca, key number untuk menyimpan nomor telepon akun/pengguna, dan key role yang akan membedakan peran masing-masing akun/pengguna seperti reporter, editor, atau admin.



Gambar 3.36 Collection news pada database user_data

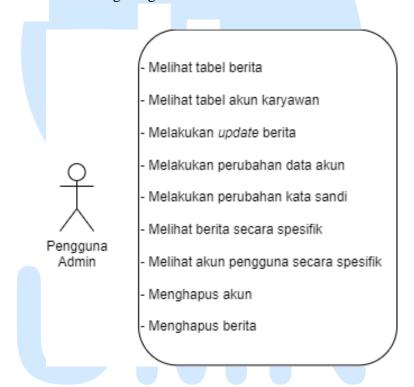
Gambar 3.36 merupakan *Collection* yang kedua dinamakan news karena akan memuat data berita yang diinput oleh pengguna terdaftar dengan peran reporter pada U-Tapis. Berita-berita tersebut nantinya juga dapat diubah oleh pengguna dengan peran editor atau admin. Beberapa *key* yang terdapat di *collection* news pada saat pertama kali berita diinput oleh reporter adalah id, title, category, content, date, by, dan status. Setelah itu, apabila berita ditapis oleh editor atau admin maka akan ditambahkan *key* konjungsi, peluluhan, dan sintaksis yang memuat hasil proses ketiga algoritma pada *website* U-Tapis.



Gambar 3.37 Contoh alamat SRV untuk melakukan koneksi

Cara untuk melakukan koneksi *database* MongoDB adalah dengan terlebih dahulu menginstal Pymongo di dalam Visual Studio Code karena dokumen di dalam Python Flask hanya dapat terkoneksi dengan MongoDB

apabila menggunakan Pymongo. Setelah terinstal maka alamat SRV yang tersedia di *website* MongoDB Atlas seperti pada gambar 3.37 dapat dimasukkan ke dalam *file* Python utama yang digunakan untuk menjalankan aplikasi U-Tapis. Selanjutnya, kata sandi yang tersedia pada bagian "<PASSWORD>" perlu diganti dengan kata sandi yang terdaftar pada MongoDB tersebut. Ketika semua sudah selesai dilakuan maka koneksi antara *database* di MongoDB Atlas dengan *website* U-Tapis dapat segera di uji coba untuk memastikan apakah koneksi sudah tersambung dengan baik.



Gambar 3.38 Beberapa kegiatan admin yang membutuhkan fungsi Pymongo

Gambar 3.38 merupakan beberapa fungsi dari Pymongo yang sering digunakan untuk melakukan koneksi dengan *database* MongoDB khususnya admin adalah fungsi find, find_one, update_one, dan delete_one. Fungsi find berguna untuk menampilkan seluruh data dari dalam *database* yang memiliki kesamaan dengan ketentuan yang ingin dicari. Contoh penggunaan fungsi find adalah ketika admin ingin melihat tabel berita dan tabel akun karyawan karena data yang ditampilkan sekaligus berjumlah banyak. Berbeda dengan fungsi find, fungsi find_one juga akan menampilkan data dari *database* akan tetapi, data

yang ditampilkan hanya satu saja. Contoh penggunaan fungsi find_one adalah ketika admin ingin melihat berita atau akun pengguna secara spesifik maka pencarian dari database untuk menampilkan satu data saja.

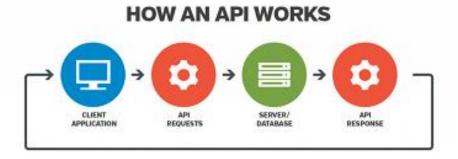
Gambar 3.39 Contoh penggunaan fungsi update_one untuk mengubah data karyawan

Fungsi berikutnya yaitu update_one digunakan untuk melakukan perubahan data dan menyimpannya ke dalam *database*. Seperti pada gambar 3.39, cara kerjanya adalah dengan menemukan id unik yang sesuai dengan yang disebutkan kemudian menyimpan perubahan data yang baru. Contoh penggunaan fungsi update_one adalah ketika admin ingin melakukan perubahan akun karyawan, perubahan kata sandi, dan perubahan berita. Fungsi terakhir adalah delete_one yang akan digunakan ketika ingin melakukan penghapusan data. Cara kerja fungsi delete_one memiliki kemiripan dengan fungsi update_one yaitu dengan menemukan id yang unik lalu menghapus data dengan id tersebut. Contoh penggunaan fungsi delete_one adalah ketika admin ingin menghapus berita atau akun karyawan.

CRUD atau Create, Read, Update, dan Delete pada web U-Tapis dapat dicontohkan pada alur mulai dari input berita hingga selesai. Proses *create* terjadi pada saat pengguna dengan peran reporter membuat sebuah berita dan menekan tombol "Simpan". Pada saat itu, berita yang dibuat akan dimasukkan ke dalam database. Selanjutnya, proses *read* akan dilakukan pada saat editor atau admin membuka daftar berita dan melihat berita yang baru dimasukkan oleh reporter sebelumnya. Pada proses tersebut, *website* akan membaca *database* dan menampilkan berita yang sudah disimpan di dalamnya. Kemudian

proses *update* akan dilakukan pada saat pengguna dengan peran editor atau admin melakukan perubahan terhadap berita tersebut dan menyimpannya kembali dengan menekan tombol "Tapis" atau "Selesai". Terakhir, proses *delete* akan dilakukan apabila editor atau admin memutuskan untuk menghapus berita tersebut sehingga, berita tersebut tidak dapat ditemukan lagi di dalam *database*.

3.2.5 Penggabungan Website dengan 3 Algoritma Machine Learning



Gambar 3.40 Cara kerja API

Untuk mengkoneksikan website U-Tapis dengan machine learning, maka digunakan Application Programming Interface (API). API adalah cara agar dua buah perangkat lunak dapat berkomunikasi satu dengan yang lain. Gambar 3.40 menjelaskan mengenai cara kerja API yang dapat diaplikasikan pada proses website U-Tapis. Ketika pengguna website menekan tombol "Tapis" maka berita yang sedang di sunting akan dikirimkan ke tautan/URL tempat algoritma machine learning di jalankan. Method API yang digunakan adalah POST karena sesuai dengan fungsinya yaitu mengirimkan data yang dalam hal ini adalah berita yang sedang di sunting. Algoritma konjungsi di deploy di server tersendiri sedangkan, algoritma sintaksis kalimat dan algoritma peluluhan kata di deploy dalam sebuah server yang terletak di UMN. Sebagai bentuk keamanan tambahan situs web U-Tapis saat ini juga diletakkan di dalam VPS atau Virtual Private Server yang disiapkan oleh UMN sehingga, akses ke dalam website U-Tapis hanya bisa dilakukan dengan bantuan Virtual Private Network (VPN).

HASIL PENAPIS

Sintaksis Kalimat:

Warna Hitam: Tidak ada kesalahan Warna Merah: Terdapat kesalahan sintaksis

Wuling mengeluarkan mobil terbarunya yaitu Wuling Air EV dan dinyatakan sebagai mobil yang paling layak dibeli di tahun 2022 ini.
Wuling mengeluarkan mobil terbarunya yaitu Wuling Air EV dan dinyatakan sebagai mobil yang paling layak dibeli di tahun 2022 ini

Gambar 3.41 Contoh deteksi kesalahan dengan algoritma Sintaksis Kalimat

Gambar 3.41 adalah Algoritma Sintaksis Kalimat yang akan membaca apakah sebuah kalimat sudah memenuhi struktur kalimat yang baik dan benar. Sebagai contoh, sebuah kalimat harus ditutup dengan tanda titik sebagai penanda kalimat berakhir. Apabila kalimat di akhir tidak diakhiri dengan titik maka kalimat tersebut akan ditandai sebagai "False". Namun, apabila kalimat sudah sesuai maka kalimat tersebut akan ditandai dengan nilai "True". Nilai *Boolean* dalam bentuk "True" atau "False" pada setiap kalimat tersebut yang dijadikan sebagai acuan untuk memberikan warna pada tampilan *website* U-Tapis. Kalimat yang memiliki nilai "True" akan ditampilkan dengan warna hitam sedangkan kalimat dengan nilai "False" akan ditampilkan dengan warna merah.

Gambar 3.42 Contoh hasil uji coba API algoritma Konjungsi

NUSANTARA

Konjungsi:

Warna Hitam: Bukan konjungsi

Warna Merah: Terdapat kesalahan konjungsi

Presiden Joko Widodo meminta maaf kepada masyarakat yang merasa terganggu kenyamanannya selama proses pernikahan putra bungsunya, Kaesang Pangarep dengan Erina Gudono

Hal tersebut khsususnya disampaikan Presiden untuk warga Yogyakarta dan Solo

Kemudian, Jokowi menyampaikan rasa syukur setelah proses siraman putra bungsungnya berlangsung lancar, pada Jumat (9/12/2022) pagi Dirinya juga meminta doa restu kepada masyarakat **agar** rangkaian prosesi perkawinan Kaesang dan Erina lancar Sebelumnya, Jokowi memimpin prosesi siraman Kaesang di kediaman keluarga di Sumber, Solo

Gambar 3.43 Contoh deteksi kesalahan dengan algoritma Konjungsi

Algoritma berikutnya adalah Konjungsi. Algoritma Konjungsi akan menganalisa apakah terdapat kata konjungsi di dalam kalimat tersebut. Setelah ditemukan kata konjungsinya maka kata tersebut akan ditandai dengan nilai "konjungsi type" yang diisi dengan tipe dari konjungsi tersebut sedangkan, apabila kata tersebut tidak termasuk konjungsi maka kata tersebut memiliki nilai "konjungsi type" yang kosong. Sebagai contoh seperti pada gambar 3.42, jika teks berita memiliki kata konjungsi "dan" maka nilai dari "konjungsi type" adalah "antara". Hasil dari API ke website juga diberikan informasi mengenai "validity" yang akan memberitahu apakah kata tersebut bernilai "true" atau "false". Apabila sebuah kata yang bukan konjungsi maka nilai "validity" dipastikan "false". Akan tetapi, jika kata yang dianalisis adalah kata konjungsi, maka nilai "validity" tersebut tergantung dari apakah penempatan konjungsi pada kalimat sudah tepat penggunaannya. Jika penggunaan kalimat sudah benar maka nilai "validity" adalah "true" dan jika tidak maka nilainya adalah "false". Melalui nilai validity tersebut jika kata tersebut merupakan konjungsi dengan nilai "validity: true" maka akan diwarnai dengan hijau namun, jika nilai "validity: false" maka akan diwarnai dengan warna merah seperti pada gambar

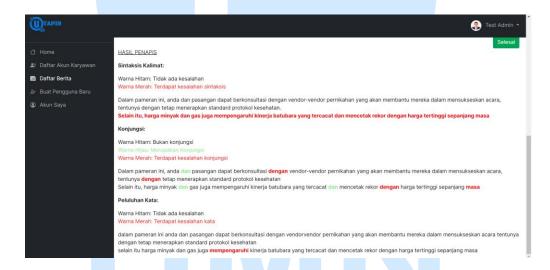
Peluluhan Kata:

Warna Hitam: Tidak ada kesalahan Warna Merah: Terdapat kesalahan kata

dalam pameran ini anda dan pasangan dapat berkonsultasi dengan vendorvendor pernikahan yang akan membantu mereka dalam mensukseskan acara tentunya dengan tetap menerapkan standard protokol kesehatan selain itu harga minyak dan gas juga mempengaruhi kinerja batubara yang tercacat dan mencetak rekor dengan harga tertinggi sepanjang masa

Gambar 3.44 Contoh deteksi kesalahan dengan algoritma Peluluhan Kata

Gambar 3.44 merupakan algoritma yang terakhir yaitu algoritma Peluluhan Kata. Algoritma Peluluhan Kata akan mendeteksi kata berimbuhan yang memiliki peluluhan dengan awalan huruf s, p, t, dan k atau disebut juga dengan aturan KPST. Contoh nya adalah kata dasar "pengaruh" jika diberikan imbuhan me- maka akan menjadi "memengaruhi" dan bukan "mempengaruhi" karna huruf p didalamnya akan diluruhkan. Respon yang dikirimkan kembali dari API memiliki nilai "is_true". Jika sebuah kata memiliki pasangan nilai "is_true: true" maka kata tersebut akan ditampilkan dengan warna hitam karena tidak termasuk ke dalam kata imbuhan atau tidak ditemukan kesalahan. Namun, apabila pasangan kata memiliki kesalahan imbuhan maka nilai "is_true" menjadi "false" dan kata tersebut akan ditampilkan dengan warna merah seperti pada gambar 3.45.



Gambar 3.45 Tampilan hasil ketiga algoritma pada halaman admin

Setelah menggabungkan 3 algoritma *machine learning* tersebut, mahasiswa juga mencoba untuk melakukan uji kecepatan proses dan membandingkannya. Skenario yang pertama adalah dengan menggunakan sebuah kalimat contoh dari sebuah berita yaitu "Dokter spesialis bedah plastik rekonstruksi dari Siloam Hospitals Balikpapan, dr. Arie Wibisono, Sp.BP-RE. mengatakan, RS Siloam Balikpapan telah menerapkan pelayanan terapi laser." dan menghitung waktu yang dibutuhkan mulai dari menekan tombol "Tapis" hingga hasil proses algoritma selesai dikirimkan kembali dan ditampilkan pada *website* U-Tapis. Berikut ini adalah lama waktu yang diperoleh:

- 1. Menggunakan 1 kalimat membutuhkan waktu 12 detik.
- 2. Menggunakan 2 kalimat membutuhkan waktu 16 detik.
- 3. Menggunakan 3 kalimat membutuhkan waktu 17 detik.
- 4. Menggunakan 4 kalimat membutuhkan waktu 19 detik.

Dapat disimpulkan dari hasil tes tersebut bahwa semakin banyak kalimat yang di proses maka waktu yang dibutuhkan juga lebih banyak. Akan tetapi, tentunya terdapat faktor lain yang dapat mempercepat atau memperlambat waktu yang dibutuhkan. Faktor tersebut adalah kekuatan sinyal dari pengguna maupun server dalam mengirimkan pesan dan juga menerima pesan. Apabila tidak terdapat gangguan sinyal maka waktu yang dibutuhkan akan cenderung lebih cepat sebaliknya, jika terdapat gangguan sinyal maka waktu yang dibutuhkan cenderung lebih lama. Skenario tes yang kedua memiliki sistem perhitungan waktu yang sama seperti skenario pertama. Perbedaannya adalah skenario kedua akan menghitung berdasarkan banyaknya kata dan bukan kalimat. Berikut ini adalah lama waktu yang diperoleh:

- 1. 1 kata "mengatakan" memerlukan waktu 8 detik.
- 2. 10 kata "mengatakan" memerlukan waktu 8 detik.
- 3. 100 kata "mengatakan" memerlukan waktu 8 detik.
- 4. 1.000 kata "mengatakan" memerlukan waktu 17 detik.
- 5. 10.000 kata "mengatakan" memerlukan waktu 1 menit 38 detik.

Berdasarkan hasil dari pengetesan skenario kedua dapat disimpulkan bahwa jumlah kata juga berpengaruh terhadap lama waktu yang dibutuhkan oleh algoritma untuk memproses isi berita yang ditampilkan. Akan tetapi, bedanya tidak terlalu signifikan hingga 100 kata dalam satu kalimat. Perbedaan waktu baru terasa pada saat kata yang dikirimkan dalam satu kalimat melebihi 100 kata seperti 1.000 kata dan 10.000 kata.

3.2.6 Uji Coba Website U-Tapis di 3 Tempat



Gambar 3.46 Presentasi perkembangan U-Tapis dengan Bapak Dr. Ninok Leksono M.A.

Gambar 3.46 merupakan situasi sebelum melaksanakan uji coba di mana tim rekan magang dari Prodi Sistem Informasi dan Prodi Jurnalistik beserta dengan 2 dosen yaitu Ibu Niknik Mediyawati S.Pd., M.Hum. dan Bapak Samiaji Bintang Nusantara S.T., M.A. mengadakan pertemuan dengan Bapak Dr. Ninok Leksono M.A. untuk membahas mengenai proyek U-Tapis dan apa saja perkembangannya mulai dari bulan Agustus hingga November. Pada saat itu juga diadakan perekaman video untuk dokumentasi U-Tapis bersama dengan Bapak Dr. Ninok Leksono M.A., serta pengambilan foto dan video ucapan dari dosen Fakultas Teknik dan Informatika yang terlibat dalam program U-Tapis yaitu Bapak Moeljono Widjaja, B.Sc., M.Sc., Ph.D. dan Bapak Rudi Sutomo, S.Kom., M.Si., M.Kom..

UNIVERSITAS MULTIMEDIA NUSANTARA



Gambar 3.47 Pelaksanaan uji coba pertama di Tribun Jakarta, 22 November 2022

Uji coba website U-Tapis pertama kali dilaksanakan pada tanggal 22 November 2022 dan bertempat di Tribun Jakarta, Palmerah yang digambarkan pada gambar 3.47. Pertemuan tersebut dihadiri oleh rekan-rekan kerja magang dari Prodi Sistem Informasi dan Prodi Jurnalistik, 2 dosen Prodi Jurnalistik yaitu Ibu Niknik Mediyawati S.Pd., M.Hum. dan Bapak Samiaji Bintang Nusantara S.T., M.A., dan beberapa Editor dari pihak PT. Tribun Digital Online. Pertemuan pertama di kantor Tribun Jakarta bertujuan untuk menyampaikan progres U-Tapis kepada rekan dari Tribun sekaligus juga meminta pendapat dan saran mengenai web U-Tapis. Sesi pertama dimulai dengan presentasi dari rekan kerja magang Prodi Jurnalistik mengenai kebahasaan yang baik dan benar. Sesi berikutnya dilanjutkan dengan presentasi web U-Tapis. Tampilan U-Tapis saat itu masih tergolong sederhana dengan hasil keluaran yang terbatas pada "True" atau "False" dengan 1 algoritma yaitu Sintaksis Kalimat. Uji coba kemudian dilanjuti oleh Denn Sebastian selaku rekan mahasiswa magang dari Prodi Informatika yang menjelaskan mengenai algoritma Sintaksis Kalimat yang dibuat olehnya secara Zoom. Pertemuan

tersebut ditutup dengan beberapa saran dan masukan dari rekan yang bekerja di PT. Tribun Digital Online.



Gambar 3.48 Uji coba kedua tanggal 8 Desember 2022 di Tribun Jakarta

Uji coba kedua dilakukan pada bulan Desember tepatnya tanggal 8 untuk Jakarta. Uji coba di Jakarta kembali diadakan di tempat Tribun yang sama seperti uji coba bulan November yang lalu tepatnya di Tribun Palmerah yang digambarkan pada gambar 3.48. Awal sesi uji coba diambil alih oleh Balai Bahasa Banten yang menjelaskan pentingnya penggunaan Bahasa Indonesia yang baik dan benar sesuai dengan kaidah KBBI. Sesi berikutnya dilanjutkan dengan uji coba U-Tapis. Pada sesi kali ini, tampilan website U-Tapis sudah mendapatkan banyak perubahan jika dibandingkan dengan uji coba yang pertama kali pada tanggal 22 November 2022. Algoritma yang sudah digabungkan ke dalam web U-Tapis masih sama yaitu hanya algoritma Sintaksis Kalimat saja. Akan tetapi, hasil keluaran dari yang sebelumnya hanya "True" atau "False" kini sudah bisa ditampilkan setiap kalimatnya dengan perbedaan warna. Kalimat berwarma hitam melambangkan "True" sehingga

tidak terdapat kesalahan sintaksis dan kalimat berwarna merah melambangkan "False" yang artinya ditemukan kesalahan sintaksis kalimat.



Gambar 3.49 Uji coba di Makassar tanggal 10 Desember 2022

Uji coba berikutnya dilaksanakan pada tanggal 10 Desember 2022 dilaksanakan secara serentak di 2 kota yaitu Solo dan Makassar yang dimuat pada gambar 3.49 dan 3.50. Perwakilan tim U-Tapis yang pergi ke Makassar adalah Ibu Veronika, S.Sos., M.Si. selaku dosen dari Prodi Jurnalistik, dan 2 rekan magang yaitu Ruddy Kusuma Jaya dari Prodi Sistem Informasi dan Cantya Kirana Dewi dari Prodi Jurnalistik. Sementara itu, perwakilan yang melaksanakan uji coba ke Solo adalah Bapak Samiaji Bintang Nusantara S.T., M.A., bersama dengan 2 rekan magang yaitu Denn Sebastian dari Prodi Informatika dan Disty Laurent Audry dari Prodi Jurnalistik. Total tim U-Tapis yang mengikuti uji coba adalah 2 dosen dan 4 mahasiswa sementara rekanrekan magang yang lain ikut memperhatikan via zoom di Jakarta. Sama seperti uji coba di Jakarta sebelumnya, pelaksanaan uji coba U-Tapis dimulai oleh balai bahasa selama hampir 1 jam dan baru dilanjutkan dengan uji coba web U-Tapis.

Kegiatan uji coba juga diakhiri dengan penerimaan masukan dan saran dari para rekan jurnalis maupun editor di Tribun Solo dan Makassar.



Gambar 3.50 Uji coba di Solo tanggal 10 Desember 2022

3.2.7 Revisi Website U-Tapis



Gambar 3.51 Halaman Utama website U-Tapis

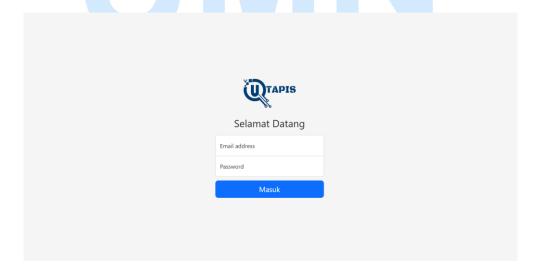
Setelah pelaksanaan uji coba dilaksanakan, sisa waktu yang ada akan digunakan untuk melakukan revisi. Revisi yang pertama dilakukan adalah

mengubah tampilan halaman paling utama sebelum melakukan *login*. Gambar 3.51 merupakan tampilan halaman utama U-Tapis yang telah ditambahkan logonya sehingga terdapat logo UMN, U-Tapis, dan Tribun untuk memberikan makna bahwa proyek ini dimiliki bersama yaitu UMN dan juga Tribun.



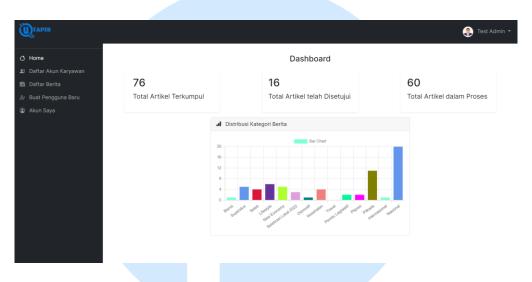
Gambar 3.52 Tampilan "Tentang U-Tapis" pada halaman utama U-Tapis

Selain itu seperti pada gambar 3.52, ditambahkan juga informasi mengenai U-Tapis pada bagian "Tentang U-Tapis" yang akan menjelaskan mengenai latar belakang singkat lahirnya U-Tapis. Berikutnya, pada bagian *sidebar* dan *navbar* dari *website* U-Tapis menjadi berwarna hitam dan juga menambahkan logo U-Tapis yang terbaru.



Gambar 3.53 Halaman Login

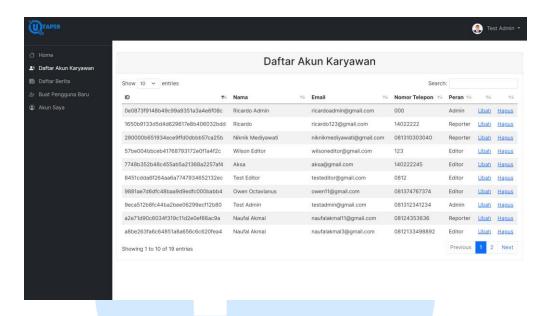
Gambar 3.53 merupakan halaman login dari web U-Tapis yang harus dilakukan sebelum pengguna dapat memasuki halaman *dashboard*. Beberapa hal yang diubah dari desain awalnya adalah latar belakang berwarna abu-abu, logo U-Tapis yang ditampilkan di atas, serta kata sambutan "Selamat Datang".



Gambar 3.54 Halaman awal admin

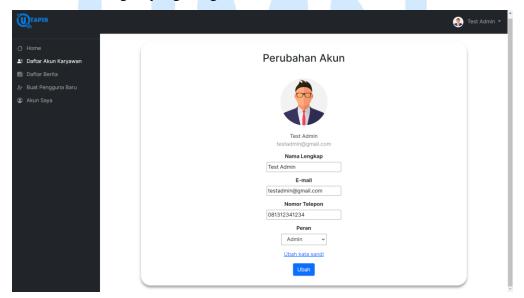
Gambar 3.54 merupakan halaman admin atau dashboard admin yang akan ditampilkan pada saat pengguna dengan peran admin melakukan *login*. Beberapa perubahan yang dilakukan antara lain adalah menambahkan gambar grafik distribusi kategori berita yang dinamis mengikuti perubahan data dari *database* setiap kali halaman tersebut dimuat, total artikel terkumpul atau jumlah artikel yang sudah dikumpulkan oleh reporter, total artikel yang telah disetujui atau artikel yang memiliki status "Selesai", dan total artikel dalam proses atau artikel yang masih memiliki status "Pending".

MULTIMEDIA NUSANTARA



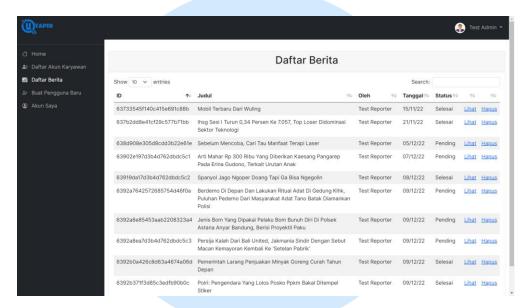
Gambar 3.55 Halaman daftar akun karyawan

Gambar 3.55 merupakan halaman daftar akun karyawan. Tabel pada halaman daftar akun karyawan dan daftar berita mengalami perubahan visual seperti gradasi warna putih dan abu-abu pada setiap barisnya agar dapat dibedakan dengan lebih mudah. Selain itu, saat ini tabel juga memiliki fitur pencarian data dan fitur untuk menyortir nilai tabel yang ditampilkan berdasarkan kategori yang diinginkan.



Gambar 3.56 Halaman perubahan akun karyawan

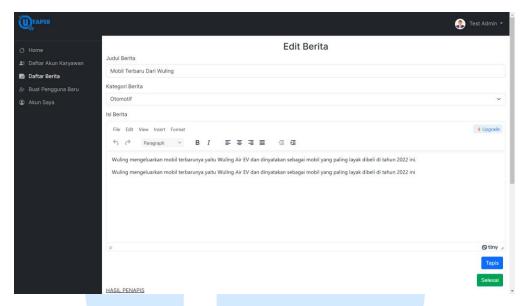
Gambar 3.56 merupakan halaman perubahan akun. Pada halaman ini, admin dapat melakukan perubahan informasi akun lain seperti mengubah nama, email, nomor telepon, dan peran akun tersebut. Setelah selesai maka admin dapat menekan tombol "Ubah".



Gambar 3.57 Halaman daftar berita

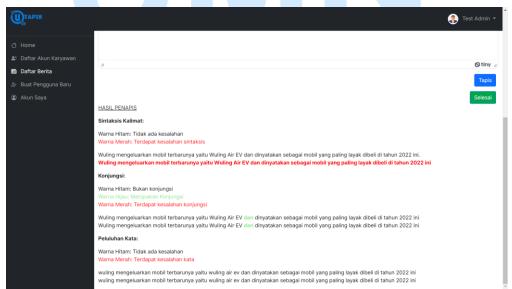
Gambar 3.57 adalah halaman daftar berita. Pada halaman tersebut, admin dapat melihat daftar berita yang sudah dimasukkan oleh reporter sebelumnya. Sama seperti tabel pada halaman daftar akun karyawan, terdapat perubahan visual seperti gradasi warna putih dan abu-abu. Kemudian, tabel daftar berita saat ini memiliki fitur pencarian data beserta dengan fitur sortir yang dapat menampilkan tabel berdasarkan urutan kategori naik atau turun.

UNIVERSITAS MULTIMEDIA NUSANTARA

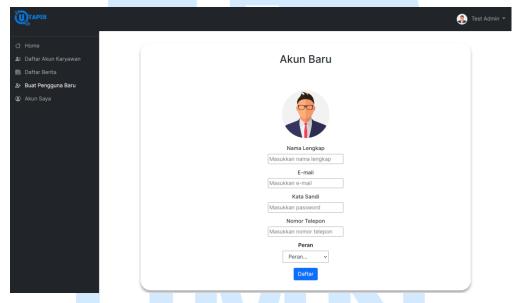


Gambar 3.58 Halaman edit berita

Gambar 3.58 merupakan halaman edit berita yang memungkinkan admin untuk melakukan perubahan berita terhadap berita-berita yang sudah di input sebelumnya oleh reporter baik itu perubahan untuk judul berita, kategori berita, maupun isi beritanya. Selain itu, admin juga dapat menekan tombol "Tapis" untuk mengirimkan isi berita kepada API untuk melakukan pengecekan apakah isi berita tersebut sudah sesuai dengan kaidah Bahasa Indonesia atau belum. Apabila admin sudah selesai melakukan edit berita maka admin dapat menekan tombol "Selesai" untuk kembali ke halaman daftar berita sekaligus mengubah status berita tersebut dari "Pending" menjadi "Selesai".

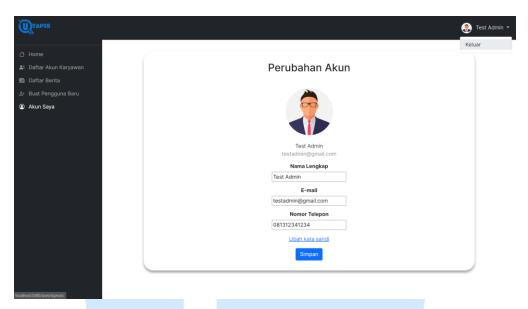


Gambar 3.59 merupakan halaman edit berita pada bagian hasil penapis yang mengalami sedikit perubahan jika dibandingkan dengan rencana awal yang dibuat dengan aplikasi Figma. Pada mulanya mahasiswa dan rekan-rekan yang lain merencanakan untuk langsung memberikan tanda kesalahan pada kolom teks berita akan tetapi, karena ternyata fitur tersebut tidak dimungkinkan untuk digunakan maka hasil tapis dari berita yang sedang disunting dipindahkan ke bawah kolom teks berita.



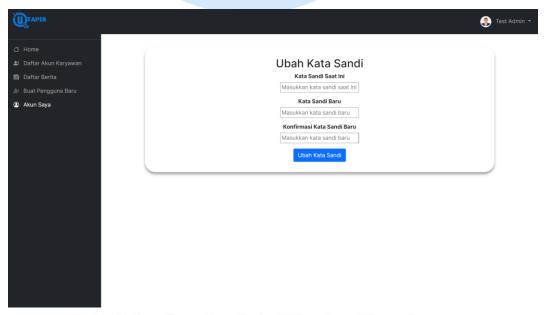
Gambar 3.60 Halaman pembuatan akun baru

Gambar 3.60 adalah halaman pembuatan akun baru. Mirip dengan halaman perubahan akun karyawan, halaman ini juga memiliki tampilan yang hampir identik. Bedanya hanya pada fungsinya saja di mana halaman pembuatan akun baru digunakan untuk pembuatan akun dan hanya dimiliki oleh pengguna dengan *role* admin.



Gambar 3.61 Halaman perubahan akun

Gambar 3.61 merupakan halaman perubahan akun pribadi. Pada halaman ini, admin dapat mengubah informasi akun pribadinya seperti mengganti nama, email, nomor telepon, dan kata sandi yang dipisah pada halaman tersendiri. Setelah selesai maka admin hanya perlu menekan tombol "Simpan".



Gambar 3.62 Halaman perubahan kata sandi

Gambar 3.62 merupakan halaman perubahan kata sandi yang memiliki 3 isian yaitu kata sandi saat ini, kata sandi baru, dan konfirmasi kata sandi baru. Halaman ini tidak ada sebelumnya pada desain Figma. Akan tetapi, karena

pertimbangan keamanan yang lebih maka dibuatkan halaman khusus beserta dengan 3 isian tersebut. setelah mengisi semuanya, admin dapat menekan tombol "Ubah Kata Sandi" untuk mengkonfirmasi perubahan tersebut.

3.3 Kendala yang Ditemukan

Beberapa kendala yang ditemukan selama proses kerja magang antara lain:

6. Kurangnya komunikasi

Pada saat pelaksanaan program magang U-Tapis, seringkali terjadi kesalahan atau kurangnya komunikasi antar rekan kerja. Sebagai contoh, saat pembagian tugas kerja di minggu pertama dan kedua magang, rekan kerja dari program studi Sistem Informasi mendapatkan tugas untuk membuat algoritma U-Tapis. Sedangkan, rekan kerja dari program studi Teknik Informatika mendapat tugas untuk mengerjakan website U-Tapis. Akan tetapi, setelah 2 minggu berjalan tugas tersebut ternyata tertukar sehingga, rekan dari program studi Sistem Informasi beralih tugas sebagai pembuat website U-Tapis dan rekan dari program studi Teknik Informatika yang akan membuat algoritmanya.

7. Waktu kerja terbatas

Sebenarnya, akan lebih baik apabila program ini dilaksanakan sejak bulan Juli atau awal Agustus. Akan tetapi, karena komunikasi yang tergolong lamban dan tidak sering dilakukan menyebabkan program magang ini secara efektif mulai di awal September. Oleh karena itu, terdapat kurang lebih 1 bulan lebih waktu yang terbuang dan tidak dapat dimanfaatkan dengan baik.

8. Pengalaman yang kurang memadai

Tugas yang diperoleh untuk rekan kerja program studi Sistem Informasi adalah pembuatan situs web U-Tapis dengan menggunakan *framework* Python Flask. Akan tetapi, *framework* Python Flask tergolong ke dalam bahasa pemrograman situs web yang belum banyak digunakan. Oleh karena itu, rekan-rekan dari program studi Sistem Informasi belum pernah mempelajarinya. Hal tersebut menyebabkan rekan-rekan kerja magang dari

program studi Sistem Informasi harus mempelajari terlebih dahulu pembuatan website dengan *framework* Python Flask dari dasar sehingga, pembuatan situs web U-Tapis tidak dapat dilakukan secepat mungkin.

9. Latar belakang rekan yang berbeda satu sama lain

Mahasiswa yang mengikuti program magang U-Tapis berasal dari jurusan dan latar belakang yang berbeda. Total terdapat 3 jurusan yaitu program studi Jurnalistik, Sistem Informasi, dan Teknik Informatika. Seluruh mahasiswa program studi Jurnalistik dan Sistem Informasi merupakan mahasiswa semester 7 yaitu mahasiswa angkatan 2019. Sedangkan pada jurusan Teknik Informatika, terdapat mahasiswa semester akhir yang sedang melakukan skripsi, mahasiswa semester 7, dan mahasiswa semester 3 yaitu mahasiswa dari angkatan 2021. Hasilnya, rekan-rekan kerja magang memiliki kemampuan atau keahlian yang berbeda-beda.

10. Tempat kerja yang berbeda-beda

Hanya sekitar separuh dari tim U-Tapis yang mengikuti program magang secara *offline* di kampus UMN. Sisa separuhnya mengikuti magang secara daring. Alhasil, komunikasi antar sesama rekan kerja magang mengalami beberapa kendala.

3.4 Solusi atas Kendala yang Ditemukan

Berikut ini adalah solusi yang dapat dilakukan untuk mengatasi kendalakendala yang sudah disebutkan sebelumnya:

- 1. Sering melakukan komunikasi antara mahasiswa dan dosen, dosen dan dosen, dan mahasiswa dengan supervisi magang untuk menghindari terjadinya miskomunikasi sekaligus mempersingkat jangka waktu yang dibutuhkan untuk pengambilan keputusan.
- 2. Seharusnya setelah proyek U-Tapis sepakat untuk dijadikan sebagai program magang maka pihak dosen dan Tribun sesegera mungkin mengadakan pertemuan untuk membahas mengenai nasib proyek kedepannya, apa saja yang dibutuhkan, dan pembagian tugas mahasiswa. Harapannya pada saat program magang dimulai maka pertemuan perdana dapat langsung dilaksanakan untuk

penyampaian ulang saja terkait tugas magang yang akan dilaksanakan oleh mahasiswa dan bukan baru menerka-nerka dari awal mengenai pembagian tugas mahasiswa sehingga, program magang dapat dimulai secepatnya sekaligus mahasiswa masih memiliki jangka waktu kerja yang cukup untuk memenuhi kriteria minimal kerja magang yaitu 800 jam atau setara dengan 100 hari kerja untuk magang MBKM.

- Mahasiswa harus mencari tahu terlebih dahulu dengan jelas pekerjaan yang akan dilaksanakan selama magang sehingga mahasiswa dapat mencari informasi terlebih dahulu untuk mempersiapkan diri sebelum mengikuti program kerja magang.
- 4. Penting bagi para rekan kerja magang untuk sering berkomunikasi satu sama lain selama program kerja magang berlangsung agar masing-masing pihak mengetahui dengan jelas sejauh mana batasan-batasan kontribusi yang akan diberikan.
- 5. Menetapkan peraturan bahwa mahasiswa yang memang tinggal dekat dengan area kampus UMN harus mengikuti kegiatan magang secara *offline* atau setidaknya *hybrid*. Sementara itu, bagi mahasiswa yang tinggal jauh dari kampus UMN solusinya adalah dengan sering mengadakan pertemuan secara daring sehingga, komunikasi antara mahasiswa magang terjalin dengan baik dan tidak ada yang ketinggalan informasi.

UNIVERSITAS MULTIMEDIA NUSANTARA