

***PENGEMBANGAN APLIKASI DRIVER ONLINE
TRANSPORTATION TERJADWAL UNTUK OPERATOR
PLATFORM***

Skripsi
Untuk memenuhi sebagai persyaratan
Mencapai derajat S-1



Disusun oleh:
Deantari Dwikan Putri Aji
20140140102

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH YOGYAKARTA
2019**

**HALAMAN PENGESAHAN
SKRIPSI**

**PENGEMBANGAN APLIKASI DRIVER ONLINE
TRANSPORTATION TERJADWAL UNTUK OPERATOR
PLATFORM**

Disusun oleh:
**DEANTARI DWIKAN PUTRI AJI
20140140102**

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH YOGYAKARTA
2019**

Telah dipertanggungjawabkan dan disahkan pada tanggal :
Juli 2019

Susunan Dewan Penguji

Dosen Pembimbing I

Dosen Pembimbing II

Dr. Ir. Dwijoko Purbohadi, M.T.
NIDN : 0502026801

Aprilia Kurnianti, S.T., M. Eng.
NIDN : 0518048401

Penguji

Haris Setyawan S.T., M.Eng
NIDN : 0511116901

Skripsi Ini Telah Dinyatakan Sah Sebagai Salah Satu Persyaratan Untuk
Memperoleh Gelar Sarjana Teknik
Mengesahkan
Kepala Program Studi Teknik Informatika

Asroni, S.T., M.Eng.
NIDN : 052604740

PERNYATAAN

Dengan ini saya menyatakan bahwa skripsi ini tidak mengandung karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya tidak mengandung karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan di dalam daftar pustaka.

Yogyakarta, 26 Juli 2019

Deantari Dwikan Putri Aji

KATA PENGANTAR

Penulis bersyukur dengan segenap hati kepada Allah SWT yang telah memberikan segalanya sehingga penulisan karta Tugas Akhir yang berjudul Pengembangan Aplikasi Driver Online Transportation Terjadwal untuk Operator Platform telah terselesaikan dengan baik. Pengembangan aplikasi ini bertujuan untuk memberikan jadwal kerja Driver yang dibutuhkan Pihak Operator Platform.

Dalam melakukan pengembangan sistem dan penyusunan laporan skripsi penulis telah mendapatkan banyak dukungan dan bantuan dari berbagai pihak. Penulis mengucapkan terima kasih yang tak terhingga kepada :

1. Bapak Dr. Ir. Dwijoko Purbohadi, M.T. selaku pembimbing pertama yang telah membantu dalam proses pengembangan sistem dan proses penulisan laporan akhir dengan penuh kesabaran dan ketulusan.
2. Ibu Aprilia Kurnianti, S.T., M.Eng selaku pembimbing kedua yang telah meluangkan waktu membimbing penulis dalam penulisan skripsi.
3. Bapak Haris Setyawan S.T., M.Eng selaku dosen penguji yang telah memberikan masukan berarti, serta ilmu selama masa perkuliahan penulis.
4. Para dosen dan staf Teknik Informatika yang telah memberikan ilmunya selama penulis belajar di Universitas Muhammadiyah Yogyakarta.
5. Orang tua dan keluarga penulis yang telah mendukung semua proses perkuliahan hingga penyusunan tugas akhir hingga selesai.
6. Selvi, Erise, serta sahabat dan teman-teman yang selalu menyemangati penulis dalam menyelesaikan tugas akhir.
7. Semua pihak yang tidak dapat penulis sebutkan satu-persatu, yang telah membantu secara langsung maupun tidak langsung.

Yogyakarta, Juli 2019

Deantari Dwikan Putri Aji

INTISARI

Mobilitas merupakan aktivitas penting yang dilakukan manusia untuk memenuhi kebutuhan sehari-hari. Demi memenuhi kegiatan ini telah banyak dibuat cara untuk ditempuh. Di Indonesia terdapat transportasi yang sudah lama beroperasi seperti becak, delman, dan ojek yang membuktikan bahwa transportasi terus berkembang sesuai dengan jaman. Salah satu model terbaru yang digunakan adalah sewa kendaraan menggunakan aplikasi web atau *smartphone*. Karena dalam proses membutuhkan pemesanan kendaraan dilakukan sebelum waktu transportasi dibutuhkan, pengelola platform yang dalam kasus ini adalah operator membutuhkan aplikasi penjadwalan *driver* berdasar trip. Karena operator terus bekerja penerima trip dan menunjuk *driver* untuk mengerjakan trip setiap hari, jadwal yang dibutuhkan tidak boleh dalam jangka waktu yang terlalu lama. Itu sebabnya aplikasi penjadwalan yang dibuat berdasarkan kurun waktu mingguan.

Inisiasi proyek aplikasi penjadwalan dimulai dari clone proyek ke dalam komputer dari repository. Setelah proyek tersimpan di dalam computer, dilakukan koneksi service dengan MongoDB selaku database yang digunakan. Deklarasi tersebut dilakukan di dalam file *config.js*. Selanjutnya dibutuhkan deklarasi skema database pada bagian service trip. Setelah membuat sebuah file bernama *index.js* di dalam *folder trips/model*, selanjutnya diberi isi berupa skema yang akan digunakan dalam database trip.

Bahasa pemrograman yang digunakan untuk membuat aplikasi penjadwalan adalah Node.js. Ketika menjalankan Service, Express.js sebagai *framework* akan bekerja menggunakan Bahasa Javascript dan menerima *input* dari *frontend* untuk selanjutnya diproses di dalam mesin. Di dalam mesin ini dibutuhkan Node.js untuk operasi pemrograman berjalan. Data trip akan diambil dari database MongoDB lalu dilempar ke dalam service untuk selanjutnya diproses oleh *code* yang dibuat. Di dalam *folder trip/service* terdapat file *index.js* berisi fungsi yang ada pada service trip. Di dalamnya dilakukan coding dengan hasil akhir berupa data terstruktur JSON berisi jadwal *driver* serta trip yang dikerjakan selama satu minggu.

Keyword: transportasi, javascript, Node.js, Express.js, MongoDB, JSON

ABSTRACT

Mobility is an important act which human use in their daily life. In the purpose of fulfilling this necessity there are a lot of ways that already been taken. In Indonesia, there even a transportation which also became part of the culture like becak, delman and ojek. This is a proof that transportation has always evolved within the change of the time. One of the newest kind of transportation is rent a car using web or mobile application. The client of this service need to book their order within a period of time before they actually using this transportation. In regards of that matter, the operator which is the person responsible to manage the trips of the application needs a scheduler application based by trip.

The project initiation starts with a project clone form a repository into the computer. After downloaded into the computer, it is a mandatory to make a connection between the service and the database which is MongoDB. This declaration taken place in a file named config.js. After the connection established, it is necessary to create a file declaring the database scheme in the trip service section. This file will be called index.js and put into trips/model folder. Inside the file is a declaration of the trip scheme in the database.

Node.js is the programming language chosen for developing this scheduler application. When the service starts, Express.js as the framework will starts to work using javascript to receive input data from the frontend as for work it later. For the application to work, Node.js is the language for the tools. The data retrieved from the Mongo Database will be send to service so it could be worked on. Inside trips/service folder there is a file named index.js. In this particular file is the code for processing the retrieved data into a structured JSON which contain a driver schedule within a week time.

Keyword: transportation, javascript, Node.js, Express.js, MongoDB, JSON

DAFTAR ISI

HALAMAN PENGESAHAN	<i>i</i>
PERNYATAAN	<i>ii</i>
KATA PENGANTAR	<i>iii</i>
INTISARI	<i>iv</i>
ABSTRACT	<i>v</i>
DAFTAR ISI	<i>vi</i>
DAFTAR GAMBAR	<i>viii</i>
DAFTAR LAMPIRAN	<i>x</i>
BAB I PENDAHULUAN	<i>1</i>
1.1. Latar Belakang	<i>1</i>
1.2. Rumusan Masalah	<i>2</i>
1.3. Batasan Masalah	<i>2</i>
1.4. Tujuan	<i>2</i>
1.5. Manfaat	<i>3</i>
1.6. Sistematika Penulisan	<i>3</i>
BAB II TINJAUAN PUSTAKA	<i>5</i>
2.1 Tinjauan Pustaka	<i>5</i>
2.2 Dasar Teori	<i>6</i>
2.2.1 Visual Studio Code	<i>6</i>
2.2.2 Arsitektur Aplikasi	<i>6</i>
2.2.3 Arsitektur Aplikasi Web	<i>7</i>
2.2.4 Backend	<i>7</i>
2.2.5 Node.js	<i>7</i>
2.2.6 Express.js	<i>8</i>
2.2.7 Express.js Middleware	<i>10</i>
2.2.8 NPM	<i>12</i>
2.2.9 Path	<i>12</i>
2.2.10 Moment	<i>13</i>
2.2.11 MongoDB	<i>15</i>

2.2.12	Mongoose	16
2.2.13	JSON	17
2.2.14	Git	20
2.2.15	Mercurial	22
2.2.16	Bitbucket.....	23
2.2.17	Insomnia	23
BAB III TATA CARA PENGEMBANGAN APLIKASI.....		25
3.1	Metodologi Pengembangan Aplikasi.....	25
3.2	Analisa Kebutuhan	25
3.3	Kebutuhan Fungsional	26
3.4	Mendefinisikan Database yang Digunakan	26
3.5	Mendefinisikan Implementasi dan Pengujian.....	27
BAB IV IMPLEMENTASI DAN PENGUJIAN		28
4.1	Alur Proses	28
4.2	Inisiasi Service.....	29
4.3	Inisiasi Program	32
4.4	Instalasi NPM dan Package yang Dibutuhkan	41
4.5	Koneksi Service ke Database	44
4.6	Koneksi Program ke Service.....	46
4.7	Deklarasi Model Database ke Dalam Database	46
4.8	Model Data	47
4.9	Pengambilan dan Seleksi Data.....	51
4.10	Enkapsulasi Data Menjadi JSON.....	55
4.11	Pengujian	63
BAB V KESIMPULAN		66
DAFTAR PUSTAKA		67

DAFTAR GAMBAR

Gambar 2. 1 Ruang Lingkup Node.js	8
Gambar 2. 2. Express.js Life Cycle	10
Gambar 2. 3. Express.js Middleware	11
Gambar 2. 4. Hasil Pemanggilan Hari Menggunakan Javascript	13
Gambar 2. 5. Hasil Pemanggilan Hari Menggunakan Moment.....	14
Gambar 2. 6. Contoh JSON	15
Gambar 2. 7. Pemetaan Objek Mongo Menggunakan Mongoose	16
Gambar 2. 8. Objek dalam JSON.....	18
Gambar 2. 9. Array dalam JSON	18
Gambar 2. 10. Value dalam JSON.....	19
Gambar 2. 11. String Dalam JSON.....	20
Gambar 2. 12. Logo Git	21
Gambar 2. 13. Logo Mercurial	22
Gambar 2. 14. Logo Bitbucket.....	23
Gambar 4. 1. Struktur Enkapsulasi Data yang Diinginkan.....	29
Gambar 4. 2. Clone Proyek dari Bitbucket.....	30
Gambar 4. 3. Merubah Direktori Terminal Men Document.....	30
Gambar 4. 4. Cloning Proyek dari Terminal.....	31
Gambar 4. 5. Proyek Tersimpan di Direktori Dokumen.....	31
Gambar 4. 6. Tampilan Proyek Dibuka dari Visual Studio Code.....	32
Gambar 4. 7. Membuat file index.js	32
Gambar 4. 8. Membuat Koneksi Menuju Database	33
Gambar 4. 9. Deklarasi Skema Agar Bisa Dipakai di Luar File.....	40
Gambar 4. 10. Terminal Integrasi Visual Studio Code.....	41
Gambar 4. 11. Perintah NPM Dalam Terminal	42
Gambar 4. 12. Instalasi Package Menggunakan NPM	43
Gambar 4. 13. Isi package.json	44
Gambar 4. 14. Fungsi trimChar	45
Gambar 4. 15. Fungsi Module Exports pada config.js	45

Gambar 4. 16. Koneksi Program untuk Mengakses Package	46
Gambar 4. 17. Deklarasi path Menuju File Skema.....	47
Gambar 4. 18. Collections dalam Database	48
Gambar 4. 19. Pembuatan konstruktor	52
Gambar 4. 20. Code Untuk Mengecek Hari Dalam Satu Minggu	52
Gambar 4. 21. Hasil Code Mendapatkan Hari Dalam Satu Minggu	53
Gambar 4. 22. Query Pertama	53
Gambar 4. 23. Query Kedua	53
Gambar 4. 24. Query Ketiga.....	54
Gambar 4. 25. Query Keempat	54
Gambar 4. 26. Query Untuk Memanggil Data Dari Collections Lain	55
Gambar 4. 27. Membuat Fungsi Group By Driver	56
Gambar 4. 28. Looping Untuk Mendapatkan List Driver.....	56
Gambar 4. 29. Mengambil Data Driver yang Sesuai Trip	57
Gambar 4. 30. Membuat Fungsi Mapping Driver Schedule.....	58
Gambar 4. 31. Membuat Fungsi Penambahan Total Trip Per Hari	58
Gambar 4. 32. Fungsi Penambahan Trip Berdasarkan Tipe Trip	59
Gambar 4. 33. Menyimpan Value Hasil Jadi.....	59
Gambar 4. 34. Menghapus Data yang Sudah Sekali Diproses	60
Gambar 4. 35. Hasil Jadi Dalam Variabel driverScheduleInAweek	60
Gambar 4. 36. Menggabungkan Dua Format Data Menjadi Satu.....	61
Gambar 4. 37. Seluruh Code Yang Digunakan Untuk Memproses Data	62
Gambar 4. 38. Memasukkan Hasil Proses Ke Dalam Express	63
Gambar 4. 39. Pembuatan Endpoint Agar Bisa Diakses Client.....	63
Gambar 4. 40. Perintah Memulai Service.....	64
Gambar 4. 41. Memasukkan Tujuan Endpoint Service Penjadwalan.....	64
Gambar 4. 42. Hasil Data JSON yang Dikirim Service Sebagai Respon.....	65

DAFTAR LAMPIRAN

Lampiran 1. Data Hasil Akhir JSON	68
---	----