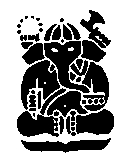
**Dokumen Pembangunan**

***Web Service***

**II3160 Pemrograman Integratif**

**Project Timeline Management System**



untuk:

I Gusti Bagus Baskara Nugraha ST,MT,Ph.D.

Dipersiapkan oleh:

Fahmi Satria Aji /18215021

Program Studi Sistem dan Teknologi Informasi

STEI - ITB

Jl. Ganesha 10, Bandung 40132

2017

Daftar Isi

[Daftar Isi 2](#_Toc501053460)

[Daftar Tabel 2](#_Toc501053461)

[Daftar Gambar 3](#_Toc501053462)

[BAB I Pendahuluan 4](#_Toc501053463)

[1.1. Latar Belakang 5](#_Toc501053464)

[1.2. Deskripsi *Web Service* 5](#_Toc501053465)

[1.3. Deskripsi API 6](#_Toc501053466)

[BAB II Spesifikasi Kebutuhan Perangkat Lunak 8](#_Toc501053467)

[2.1. Kebutuhan Fungsional 8](#_Toc501053470)

[2.2. Kebutuhan Non Fungsional 9](#_Toc501053471)

[2.3. Batasan Perancangan 9](#_Toc501053472)

[BAB III Deskripsi Perancangan *Web Service* 11](#_Toc501053473)

[3.1. Lingkungan Implementasi 11](#_Toc501053474)

[3.2. *E-R Diagram* 11](#_Toc501053475)

[3.3. *Use Case Diagram* 12](#_Toc501053476)

[3.4. Implementasi Web Service 13](#_Toc501053477)

[BAB IV Hasil Implementasi dan Pengujian *Web Service* 16](#_Toc501053478)

[4.1. Hasil Pengujian 16](#_Toc501053480)

[4.2. Evaluasi Hasil Pengujian 21](#_Toc501053481)

Daftar Tabel

[Tabel 1. Deskripsi API 6](#_Toc501053514)

[Tabel 2. Kebutuhan Fungsional 8](#_Toc501053515)

[Tabel 3. Kebutuhan Non Fungsional 9](#_Toc501053516)

[Tabel 4. Lingkungan Implementasi 11](#_Toc501053517)

Daftar Gambar

[Gambar 1. Cara Kerja *Web Service* 8](#_Toc501053551)

[Gambar 2. *Entity Relationship Diagram* 12](#_Toc501053552)

[Gambar 3. *Use Case Diagram* 13](#_Toc501053553)

[Gambar 4. Tampilan Fungsi *GetAllKejaran* 14](#_Toc501053554)

[Gambar 5. Tampilan Fungsi *GetKejaranByNamaDivisi* 14](#_Toc501053555)

[Gambar 6. Tampilan Fungsi *GetKejaranByPenanggungJawab* 14](#_Toc501053556)

[Gambar 7. Tampilan Fungsi *GetKejaranByNamaPekerjaan* 15](#_Toc501053557)

[Gambar 8. Tampilan Fungsi *GetKejaranByTanggalTenggat* 15](#_Toc501053558)

[Gambar 9. Tampilan Fungsi *PostKejaran* 15](#_Toc501053559)

[Gambar 10. Tampilan Fungsi *UpdateDataKejaran* 16](#_Toc501053560)

[Gambar 11. Tampilan Fungsi *DeleteDataKejaran* 16](#_Toc501053561)

BAB I

Pendahuluan

* 1. Latar Belakang

Institut Teknologi Bandung memiliki bermacam-macam organisasi. Setiap organisasi memiliki visi, misi, tujuan, dan program kerja yang berbeda. Dalam merealisasikan program kerja, setiap organisasi akan membuat rencana kerja dalam bentuk *roadmap* atau *timeline*. Setiap rencana kerja tersebut memiliki deskripsi pekerjaan, penanggung jawab, dan tanggal tenggat. Namun, masalah yang sering timbul pada suatu organisasi adalah seringkali terjadi penundaan pekerjaan dikarenakan kurangnya pengelolaan terhadap rencana kerja yang dimiliki.

Penundaan pekerjaan akan menyebabkan penambahan biaya dan waktu. Selain itu, kemunduran suatu pekerjaan akan menyebabkan penurunan kinerja staf organisasi dan mengakibatkan suatu program kerja organisasi tidak dapat berjalan sesuai tujuan yang ingin dicapai. Maka dari itu, kebutuhan akan pengelolaan *timeline* dalam suatu proyek atau program kerja kemudian mendorong ide untuk menciptakan sebuah *web service* yang dapat memberikan layanan pengelolaan *timeline* sebuah proyek atau program kerja organisasi di ITB. *Web service* Project Timeline Management System diharapkan dapat meningkatkan efisiensi dan efektivitas pengelolaan kejaran pada suatu proyek.

* 1. Deskripsi *Web Service*

Project Timeline Management System adalah sebuah *web service* yang menyediakan layanan untuk *web application* dalam melakukan pengelolaan *timeline* suatu proyek atau acara. Layanan ini memanfaatkan basis data yang berisi data divisi dan data pekerjaan. Data divisi berisi tentang informasi mengenai divisi suatu proyek. Sedangkan data pekerjaan berisi informasi tentang kejaran yang dimiliki oleh masing-masing divisi. *Web service* yang dikembangkan akan berbasis pada standar protokol RESTful API yang memiliki *method* GET, PUT, POST, dan DELETE.

Berikut ini adalah layanan yang ditawarkan oleh Project Timeline Management System.

1. Melihat Informasi Kejaran Divisi

Layanan ini dapat menampilkan informasi kejaran dengan melakukan pencocokkan dengan informasi nama divisi, penanggung jawab, nama pekerjaan, dan tanggal tenggat yang terdapat pada basis data.

1. Menambahkan Informasi Kejaran Divisi

Layanan ini dapat menambahkan *record* baru pada tabel pekerjaan. Informasi yang harus dimasukkan untuk menambahkan atau membuat *record* baru pada tabel pekerjaan adalah ID Pekerjaan, ID Divisi, Nama Pekerjaan, Tanggal Mulai, Tanggal Tenggat, Deskripsi Pekerjaan, dan Penanggung Jawab.

1. Memodifikasi Informasi Kejaran Divisi

Layanan ini dapat merubah *record* pada tabel pekerjaan. Informasi yang dapat diubah pada tabel pekerjaan adalah Nama Pekerjaan, Tanggal Tenggat, Deskripsi Pekerjaan, dan Penanggung Jawab Pekerjaan. Informasi kejaran divisi diubah berdasarkan ID Pekerjaan yang merupakan identitas yang membedakan suatu pekerjaan dengan pekerjaan lain.

1. Menghapus Kejaran Divisi

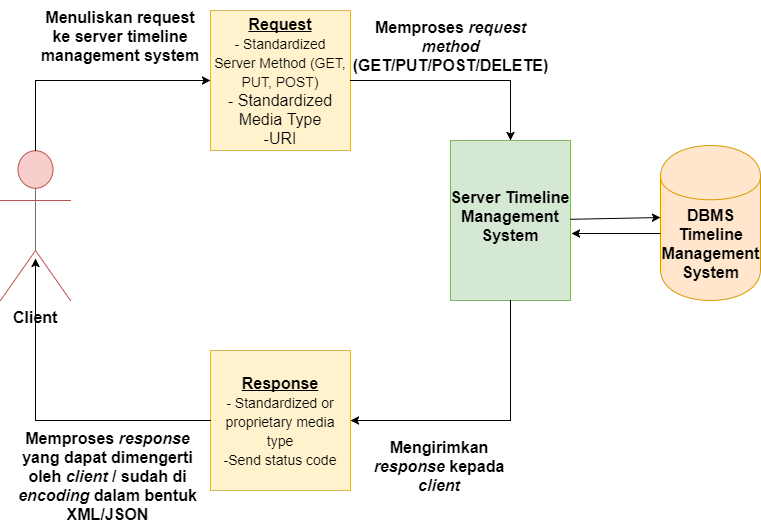
Layanan ini dapat menghapus *record* pada tabel pekerjaan. *Record* dihapus ketika kejaran tersebut telah dicapai atau dirasa tidak perlu. Informasi kejaran divisi dihapus berdasarkan ID Pekerjaan yang merupakan identitas yang membedakan suatu pekerjaan dengan pekerjaan lain.

* 1. Deskripsi API

Timeline Project Management System API ini dirancang dengan menggunakan API GO CRUD. Berikut ini adalah fungsi API yang disediakan untuk *web application*.

Tabel 1. Deskripsi API

| Method | URL | Deskripsi |
| --- | --- | --- |
| GetAllKejaran | 167.205.67.246:7601/  timelinemanagement/ | Memanggil seluruh data kejaran yang terdapat dalam basis data |
| GetKejaran  ByNamaDivisi | 167.205.67.246:7601  /timelinemanagement/  ?Nama\_Divisi=MSDM | Memanggil data kejaran sesuai dengan nama divisi |
| GetKejaran  ByPenanggungJawab | 167.205.67.246:7601/  timelinemanagement/  ?Penanggung\_Jawab=Fahmi | Memanggil data kejaran sesuai dengan penanggung jawab pekerjaan |
| GetKejaran  ByTanggalTenggat | 167.205.67.246:7601/  timelinemanagement/?  Tanggal\_Tenggat=2017 | Memanggil data kejaran sesuai dengan tanggal tenggat |
| GetKejaran  ByPekerjaan | 167.205.67.246:7601/  timelinemanagement/  ?Nama\_Pekerjaan=sponsor | Memanggil data kejaran sesuai dengan pekerjaan divisi |
| PostPekerjaan | 167.205.67.246:7601/  timelinemanagement/ | Membuat *record* baru dalam tabel pekerjaan |
| UpdateTabelPekerjaan | 167.205.67.246:7601/  timelinemanagement/  ?ID\_Pekerjaan= | Memodifikasi *record* dalam tabel pekerjaan |
| DeleteKejaranDivisi | 167.205.67.246:7601/  timelinemanagement/  ?ID\_Pekerjaan= | Menghapus *record* dalam tabel pekerjaan |

Berikut ini adalah cara kerja dari *web service* Project Timeline Management System.

Gambar 1. Cara Kerja Web Service

BAB II

Spesifikasi Kebutuhan Perangkat Lunak

2. 1. Kebutuhan Fungsional

Berikut ini adalah kebutuhan fungsional dari Project Timeline Management System.

Tabel 2. Kebutuhan Fungsional

| **ID** | **Kebutuhan** | **Deskripsi** |
| --- | --- | --- |
| FR-01 | *Web service* dapat mengumpulkan data divisi dan kejaran tiap divisi | *Web service* dapat menerima masukin data divisi dan kejaran tiap divisi |
| FR-02 | *Web service* dapat menyimpan data divisi dan kejaran dari tiap divisi dalam *database* | *Web service* terhubung dengan *database*  “projecttimeline  managementsystem” |
| FR-03 | *Web service* dapat menampilkan data dari *database*. *Web service* dapat menerima *request* GET | *Web service* akan menerima *request* GET berupa nama divisi, kejaran divisi, penanggung jawab divisi, dan tanggal tenggat kejaran divisi |
| FR-04 | *Web service* dapat menambahkan data ke *database*. *Web service* dapat menerima request POST | *Web service* akan menerima *request* POST berupa ID Pekerjaan, ID Divisi, nama pekerjaan, tanggal mulai, tanggal tenggat, deskripsi pekerjaan, dan penanggung jawab divisi |
| FR-05 | *Web service* dapat menambahkan data ke *database*. *Web service* dapat menerima request PUT | *Web service* akan menerima *request* PUT berupa nama pekerjaan, tanggal tenggat, deskripsi pekerjaan, dan penanggung jawab divisi |
| FR-06 | *Web service* dapat menambahkan data ke *database*. *Web service* dapat menerima request DELETE | *Web service* akan menghapus informasi satu *record* berdasarkan masukin ID Pekerjaan |

* 1. Kebutuhan Non Fungsional

Berikut ini adalah kebutuhan non fungsional dari *project timeline management system.*

Tabel 3. Kebutuhan Non Fungsional

| **ID** | **Kebutuhan** | **Deskripsi** |
| --- | --- | --- |
| NFR-01 | *Web service* memiliki *failure rate* tidak lebih dari 3% | *Web service* dapat menerima *request* sebanyak 5-7 |
| NFR-02 | *Web service* memiliki *reusability* tinggi | *Web service* dapat digunakan kembali untuk kepanitiaan lain selain arkavidia |
| NFR-03 | *Web service* dapat dioperasikan 24/7 | *Web service* tiap menit dapat menerima 5-7 *request* |
| NFR-04 | *Web service* memiliki *user interface* yang mudah untuk pengguna | *Web service* memiliki kemudahan untuk digunakan atau dihubungkan dengan *web application* |

* 1. Batasan Perancangan

Berikut ini adalah batasan dalam perancangan *web service* Timeline Management System.

1. *Server database* sistem menggunakan MYSQL
2. Data yang terdapat dalam basis data merupakan data *dummy*
3. Sistem hanya dapat dioperasikan menggunakan VPN Institut Teknologi Bandung
   1. *Hardware*

Berikut ini adalah *hardware* yang digunakan dalam pengembangan *web service* Timeline Management System.

1. System Model: Vostro 14-5459
2. Machine name: Desktop- T42MAIE
3. Processor: Intel® Core™ i7-6500U CPU @2.50 GHz (4 CPUs), ~2,6 GHz.
4. Memory: 8192 MB RAM
5. Page File: 7136MB used, 4154MB available
   1. *Software*

Berikut ini adalah *software* yang digunakan dalam pengembangan *web service* Timeline Management System.

1. OS: Windows 10 Home Single Language 64-bit (10.0, Build 15063) (15063.rs2\_release.170317-1834)
2. *Text Editor*: Sublime Text 3
3. *Programming Languange*: Prioritas pertama akan menggunakan bahasa GO, prioritas kedua akan menggunakan bahasa pemrograman PHP
4. Web Browser (Google Chrome)
5. Microsoft Word (*Documentation every progress*)
6. Trello (*Managing Project Web Service*)
7. XAMPP Versi 3.2.2
8. *Database*: MYSQL
9. *E*-*mail*
10. *Github:* Untuk menyimpan dokumentasi implementasi *web service*
11. *Slack*

BAB III

Deskripsi Perancangan *Web Service*

* 1. Lingkungan Implementasi

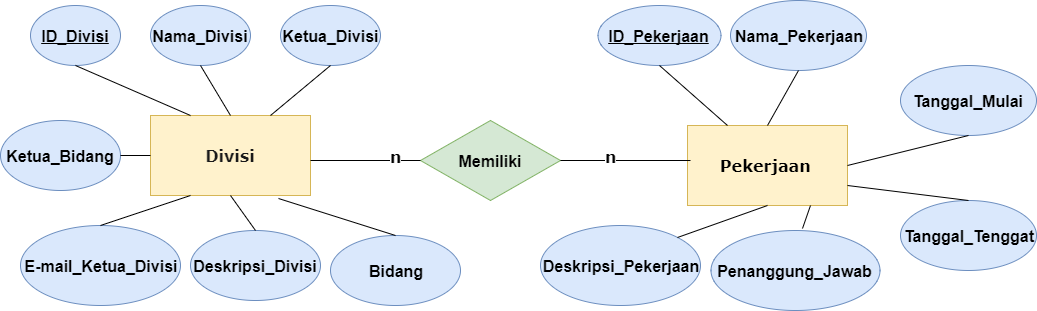
Berikut ini adalah rincian informasi mengenai lingkungan implementasi dari *web servic*e Project Timeline Management System.

Tabel 4. Lingkungan Implementasi

|  |  |
| --- | --- |
| *Operating System* | Linux Ubuntu Server |
| *Web Browser* | Google Chrome |
| DBMS | Phpmyadmin, MYSQL (1 Kelas Pemrograman Integratif) |
| *Database server* | MySQL (1 Kelas Pemrograman Integratif) |
| *Development tools* | Sublime Text 3 |
| Bahasa Pemrograman | GO Programming Languange |

* 1. *E-R Diagram*

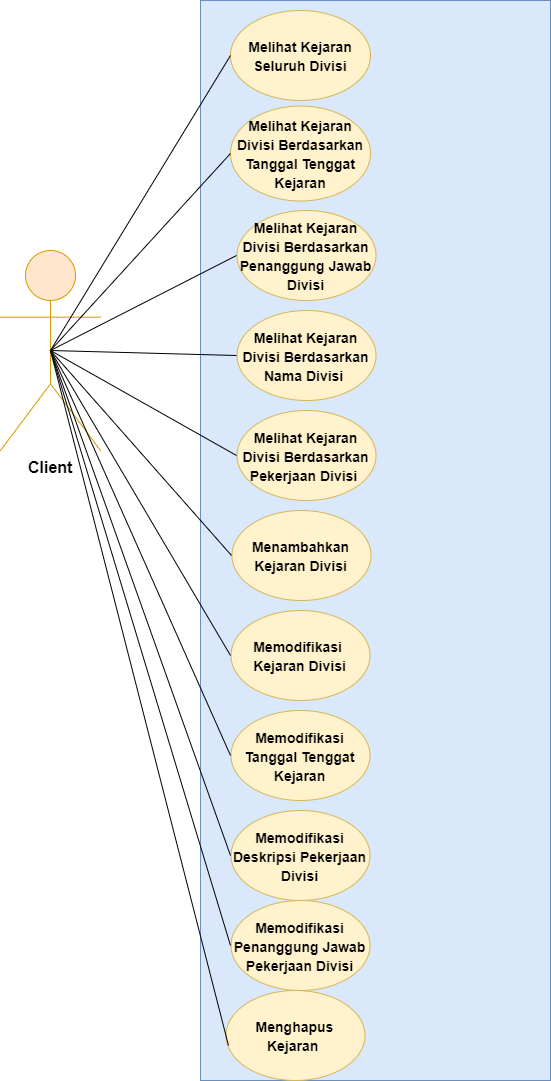
Berikut ini adalah *entity relationship diagram* dari *web service* Project Timeline Management System.

**

Gambar 2. Entity Relationship Diagram

* 1. *Use Case Diagram*

Berikut ini adalah *use case diagram* dari *web service* Project Timeline Management System.



Gambar 3. Use Case Diagram

* 1. Implementasi Web Service

Berikut ini adalah beberapa gambar tangkapan layar yang menampilkan hasil pemanggilan fungsi *web service* Project Timeline Management System.



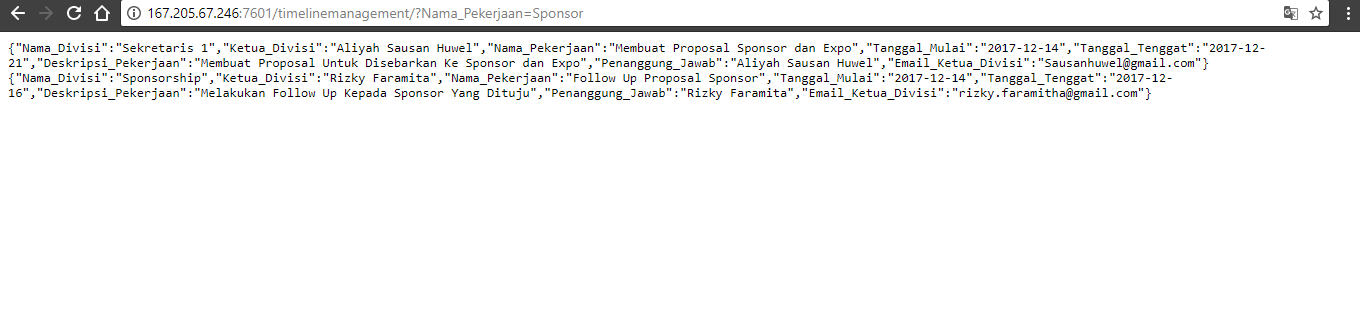
Gambar 4. Tampilan Fungsi GetAllKejaran



Gambar 5. Tampilan Fungsi GetKejaranByNamaDivisi

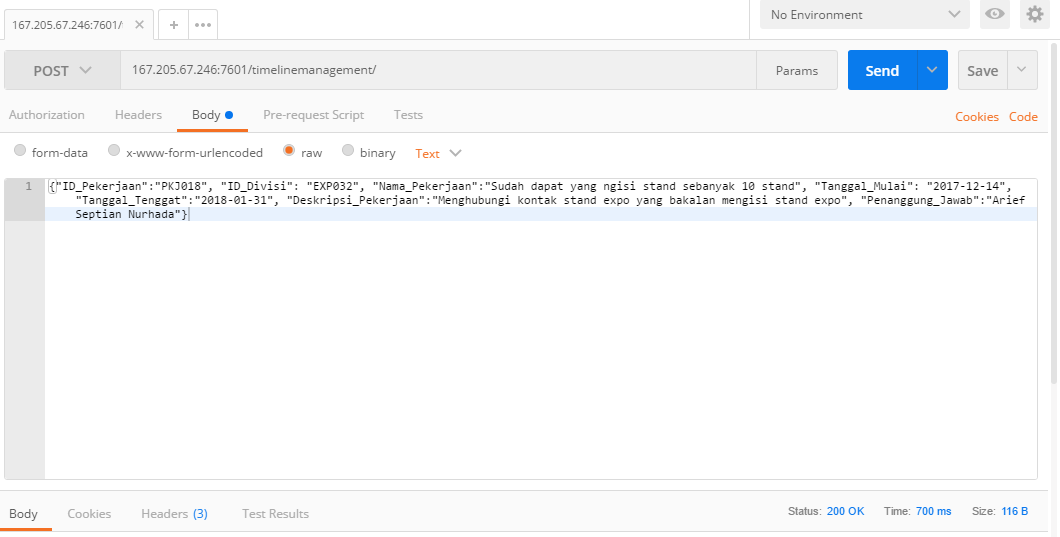


Gambar 6. Tampilan Fungsi GetKejaranByPenanggungJawab

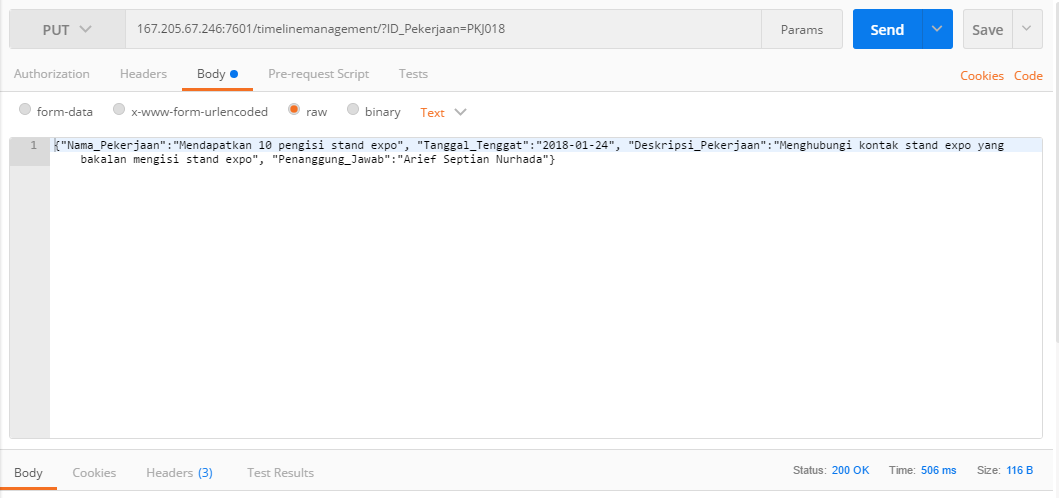


Gambar 7. Tampilan Fungsi GetKejaranByNamaPekerjaan

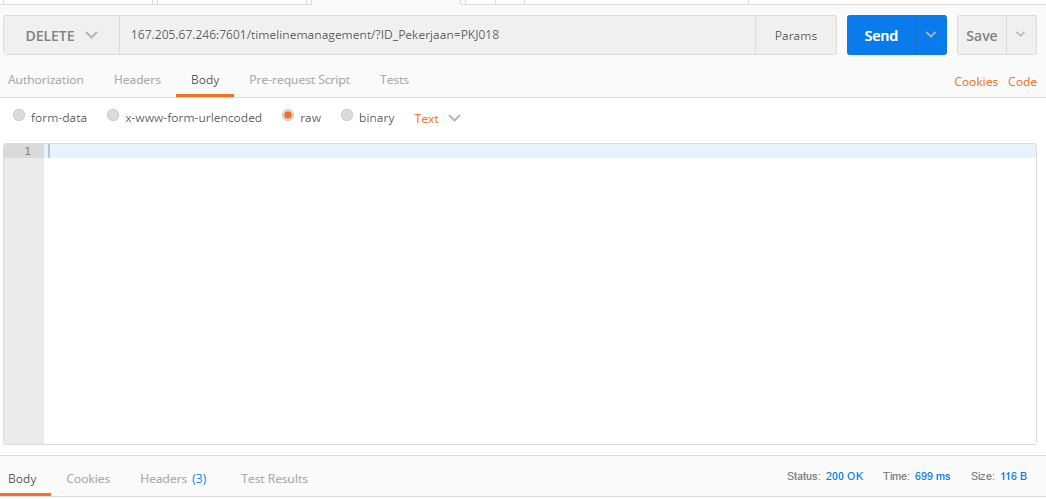


Gambar 8. Tampilan Fungsi GetKejaranByTanggalTenggat

Gambar 9. Tampilan Fungsi PostKejaran



Gambar 10. Tampilan Fungsi UpdateDataKejaran



Gambar 11. Tampilan Fungsi DeleteDataKejaran

BAB IV Hasil Implementasi dan Pengujian *Web Service*

1. 1. Hasil Pengujian

Berikut ini adalah hasil pengujian dari *web service* Project Timeline Management System.

| Identifikasi | Deskripsi | Prosedur Pengujian | Masukan | Keluaran yang Diharapkan | Kriteria Evaluasi Hasil | Hasil yang Didapat | Kesimpulan |
| --- | --- | --- | --- | --- | --- | --- | --- |
| PF-01 | Pengujian menampilkan seluruh data kejaran | Menuliskan *request* pada aplikasi postman | 167.205.67.246:7601/  timelinemanagement/ | Data Kejaran Divisi dalam format JSON | Data Kejaran Seluruh Divisi dapat ditampilkan dan *response status*: 200 OK | Data Kejaran Seluruh Divisi dapat ditampilkan dan *response status*: 200 OK | Berhasil |
| PF-02 | Pengujian menampilkan data kejaran divisi berdasarkan nama divisi | Menuliskan *request* pada aplikasi postman | 167.205.67.246:7601  /timelinemanagement/  ?Nama\_Divisi=MSDM | Data Kejaran Divisi dalam format JSON | Data Kejaran Seluruh Divisi dapat ditampilkan berdasarkan nama divisi dan *response status*: 200 OK | Data Kejaran Seluruh Divisi dapat ditampilkan berdasarkan nama divisi dan *response status*: 200 OK | Berhasil |
| PF-03 | Pengujian menampilkan data kejaran divisi berdasarkan penanggung jawab | Menuliskan *request* pada aplikasi postman | 167.205.67.246:7601/  timelinemanagement/  ?Penanggung\_Jawab=Fahmi | Data Kejaran Divisi dalam format JSON | Data Kejaran Seluruh Divisi dapat ditampilkan dan berdasarkan penanggung jawab *response status*: 200 OK | Data Kejaran Seluruh Divisi dapat ditampilkan dan berdasarkan penanggung jawab *response status*: 200 OK | Berhasil |
| PF-04 | Pengujian menampilkan data kejaran divisi berdasarkan tanggal tenggat | Menuliskan *request* pada aplikasi postman | 167.205.67.246:7601/  timelinemanagement/?  Tanggal\_Tenggat=2017 | Data Kejaran Divisi dalam format JSON | Data Kejaran Seluruh Divisi dapat ditampilkan dan berdasarkan tanggal tenggat *response status*: 200 OK | Data Kejaran Seluruh Divisi dapat ditampilkan dan berdasarkan tanggal tenggat *response status*: 200 OK | Berhasil |
| PF-05 | Pengujian menampilkan data kejaran divisi berdasarkan nama pekerjaan | Menuliskan *request* pada aplikasi postman | 167.205.67.246:7601/  timelinemanagement/  ?Nama\_Pekerjaan=sponsor | Data Kejaran Divisi dalam format JSON | Data Kejaran Seluruh Divisi dapat ditampilkan dan berdasarkan nama pekerjaan *response status*: 200 OK | Data Kejaran Seluruh Divisi dapat ditampilkan dan berdasarkan nama pekerjaan *response status*: 200 OK | Berhasil |
| PF-06 | Pengujian untuk menambah data ke tabel pekerjaan | Menuliskan *request* pada aplikasi postman | 167.205.67.246:7601/  timelinemanagement/ | Data Kejaran Divisi ditambahkan ke tabel pekerjaan | Data Kejaran Divisi dapat ditambah-kan ke tabel pekerjaan *response status*: 200 OK | Data Kejaran Divisi dapat ditambahkan ke tabel pekerjaan *response status*: 200 OK | Berhasil |
| PF-07 | Pengujian untuk memodifikasi data di tabel pekerjaan | Menuliskan *request* pada aplikasi postman | 167.205.67.246:7601/  timelinemanagement/  ?ID\_Pekerjaan= | Data Kejaran Divisi dimodifikasi dan masuk ke tabel pekerjaan | Data Kejaran Divisi dapat dimodifikasi ke tabel pekerjaan *response status*: 200 OK | Data Kejaran Divisi dapat dimodifikasi ke tabel pekerjaan *response status*: 200 OK | Berhasil |
| PF-08 | Pengujian untuk menghapus data di tabel pekerjaan | Menuliskan *request* pada aplikasi postman | 167.205.67.246:7601/  timelinemanagement/  ?ID\_Pekerjaan= | Data Kejaran Divisi dihapus ke tabel pekerjaan dan tabel pekerjaan terbarui | Data Kejaran Divisi dapat dihapus dari tabel pekerjaan *response status*: 200 OK | Data Kejaran Divisi dapat dihapus dari tabel pekerjaan *response status*: 200 OK | Berhasil |

* 1. Evaluasi Hasil Pengujian

Hasil pengujian menunjukkan bahwa 100% fitur *web service* Project Timeline Management System sudah lolos kasus uji. Namun, keterbatasan dari web service tidak menangani *error handling request* GET, POST, PUT, dan METHOD.