

# AIRCRAFT RELIABILITY DAHSBOARD DEVELOPER'S GUIDE

## FUNDAMENTALS

Aircraft Reliability Dashboard pada dasarnya hanya menggunakan PHP dan HTML Native. Disamping itu, Website ini didukung dengan css dari Bootstrap dan jquery untuk membuat website lebih menarik dan informatif.

## ARSTITEKTUR

Aircraft Reliability Dashboard dibuat berdasarkan Arsitektur Client-Server. Database yang digunakan adalah database milik Aircraft Reliability Management. Setiap tabel merupakan tabel tanpa relasi.

## Modul yang kami gunakan adalah

### 1. Datatables

Datatables merupakan modul *open source* yang telah lazim digunakan, karena tampilannya yang bagus dan tabel yang responsive. Selengkapnya mengenai datatables dapat diakses [disini](#). Dan plugin ini merupakan plugin online yang dapat digunakan dengan menuliskan jquery di akhir script. Lalu membuat id sesuai dengan jquery yang telah dibuat, yang nantinya digunakan sebagai id untuk tabel tersebut.

Show  entries Search:

Name	Position	Office	Age	Start date	Salary
Airi Satou	Accountant	Tokyo	33	2008/11/28	\$162,700
Angelica Ramos	Chief Executive Officer (CEO)	London	47	2009/10/09	\$1,200,000
Ashton Cox	Junior Technical Author	San Francisco	66	2009/01/12	\$86,000
Bradley Greer	Software Engineer	London	41	2012/10/13	\$132,000
Brenden Wagner	Software Engineer	San Francisco	28	2011/06/07	\$206,850
Brielle Williamson	Integration Specialist	New York	61	2012/12/02	\$372,000
Bruno Nash	Software Engineer	London	38	2011/05/03	\$163,500
Caesar Vance	Pre-Sales Support	New York	21	2011/12/12	\$106,450
Cara Stevens	Sales Assistant	New York	46	2011/12/06	\$145,600
Cedric Kelly	Senior Javascript Developer	Edinburgh	22	2012/03/29	\$433,060

**Name**                      **Position**                      **Office**                      **Age**                      **Start date**                      **Salary**

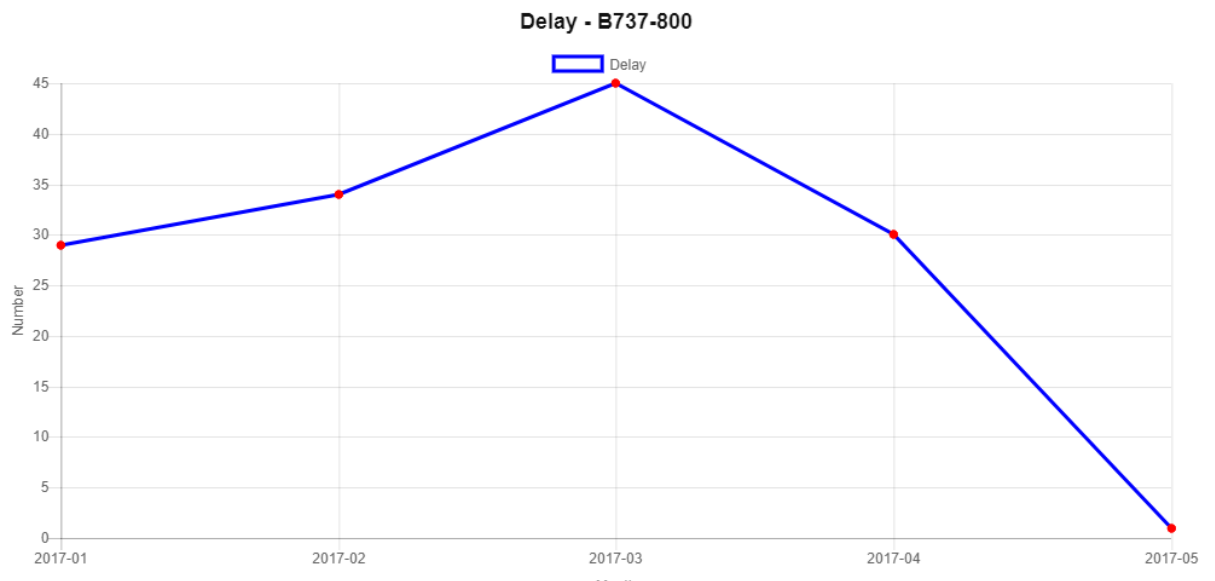
Showing 1 to 10 of 57 entries Previous  2 3 4 5 6 Next

Gambar 1. Data Tables

Karena plugin tersebut adalah plugin *open source*, maka dokumentasi dari plugin tersebut sama seperti dokumentasi yang tertera pada [website Datatables](#).

## 2. Chart JS

Chart JS merupakan modul *open source* yang lazim digunakan karena tampilannya menarik dan informatif. Selengkapnya mengenai ChartJS dapat dilihat [disini](#). Dan plugin ini merupakan plugin online yang dapat digunakan dengan menuliskan jquery di akhir script. Lalu membuat id sesuai dengan jquery yang telah dibuat, yang nantinya digunakan sebagai id untuk *Canvas* tersebut.



Gambar 2. ChartJS pada halaman graph\_delay.php

## 3. JSPdf

JSPdf merupakan modul *open source* yang lazim digunakan untuk men ekspor data dari website ke format .pdf. Selengkapnya mengenai JSPdf dapat dilihat [disini](#). Sama seperti modul sebelumnya, JSPdf merupakan modul online yang dapat digunakan dengan menuliskan jquery di akhir script. Lalu memberi id pada div yang nantinya akan diekspor.

## 4. Jsonwrapper.php

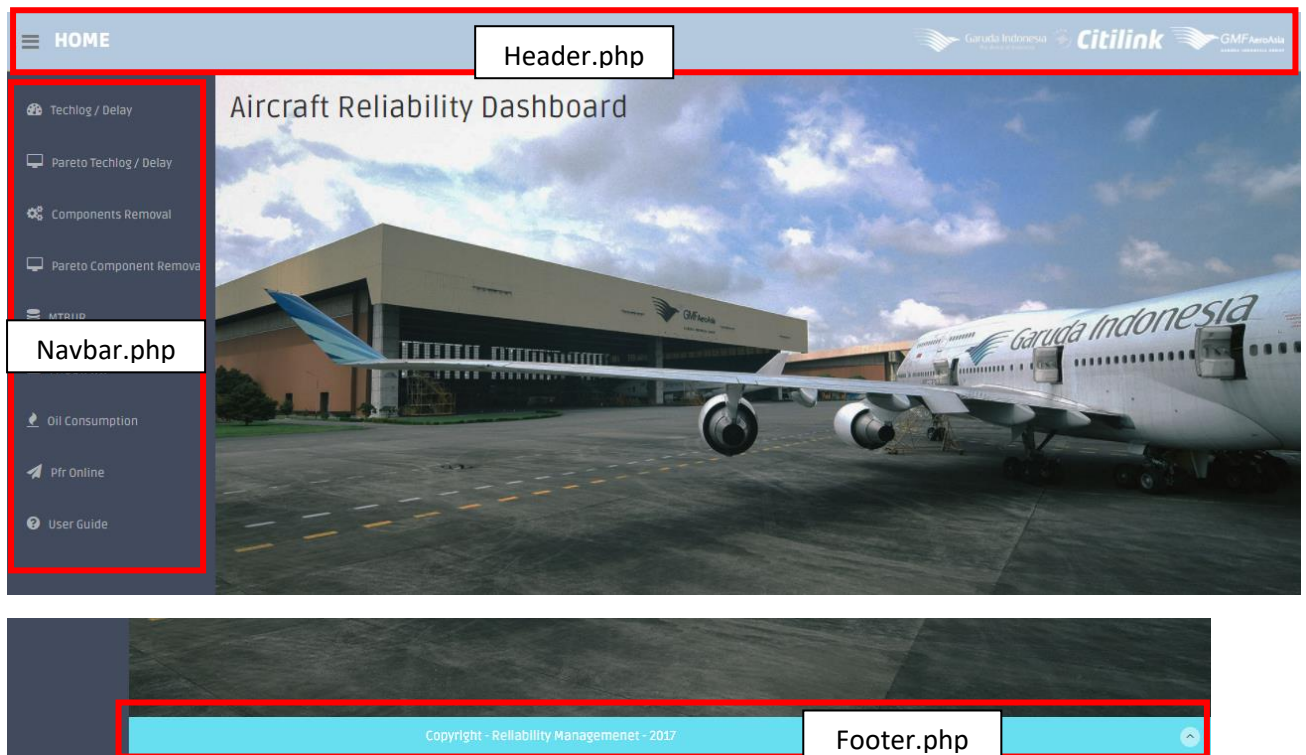
Jsonwrapper.php merupakan modul *open source* yang berguna untuk php versi dibawah 5.1 tapi dalam code terdapat fungsi yang memerlukan fungsi json. Fungsi tersebut seperti *json\_encode*, *json\_decode*, atau fungsi json lainnya yang dapat digunakan untuk mempermudah mendvelop website tersebut.

## ANATOMI CODE

Untuk mempermudah developer mendvelop website ini, ada beberapa code yang sengaja kami buat statis. Tetapi Code tersebut selalu dipanggil saat muncul halaman baru. Code itu adalah:

1. Header.php
2. Navbar.php
3. Footer.php

Berikut apabila diimplementasikan dalam website



Gambar 3. Homepage website

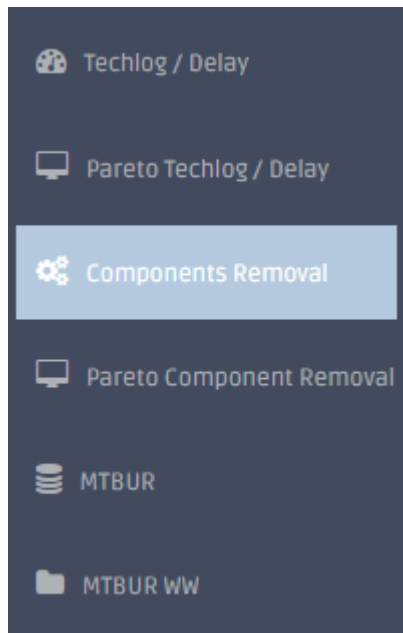
Penjelasan lengkap mengenai isi dari code diatas:

1. Header.php

Halaman ini hanya menampilkan bagian atas setiap halaman saja. Logo yang digunakan merupakan gambar statis yang kami rancang agar tidak dapat ditekan oleh user. Pada Line 2 yang berisikan `<div class="sidebar-toggle-box">` hingga `</div>` merupakan code untuk menyembunyikan atau menampilkan sidebar.

2. Navbar.php

Halaman ini merupakan halaman navigation bar yang selalu melekat di sebelah kiri setiap halaman. Terdapat variable `$page_now` yang berfungsi untuk menandai sekarang user ada di halaman apa pada navbar. Sehingga apabila user berada di salah satu halaman, maka halaman tersebut akan berubah warna



Gambar 4. Halaman user terkini berwarna biru muda

### 3. Footer.php

Halaman ini merupakan bagian bawah website yang menunjukkan bahwa website ini adalah hak milik dari Aircraft Reliability Management. Dan tahun *copyright* yang menunjukkan bahwa website ini adalah website yang terbaru atau *up to date*.

Terdapat Sembilan tombol yang berada pada navbar. Pada bagian *Techlog/Delay*, *Pareto Techlog/Delay*, *Components Removal*, *Pareto Component Removal*, dan *MTBUR*, halaman pertama setelah user menekan tombol tersebut adalah halaman filter. Halaman filter dinamai dengan format “awal\_” dan diikuti dengan nama halaman selanjutnya, dan diakhiri dengan “.php”. Contoh: “awal\_component.php”, yang artinya halaman ini adalah halaman pertama saat user menekan tombol component. Format penamaan ini berfungsi, sebagai pembeda antara halaman pertama yang hanya berisi filter, dengan halaman setelahnya yang menampilkan data secara keseluruhan. Berikut mengenai penjelasan setiap halaman.

#### 1. Halaman inti

Halaman inti merupakan halaman yang murni dikerjakan dengan native php. Seperti yang dituliskan pada penjelasan diatas, halaman inti adalah halaman yang bersangkutan dengan *Techlog/Delay*, *Pareto Techlog/Delay*, *Components Removal*, *Pareto Component Removal*, dan *MTBUR*. Dalam code kami telah kami sisipkan comment yang berguna untuk membantu developer mendvelop website tersebut.

Terdapat kemiripan pada setiap awal halaman tersebut (halaman yang memiliki nama berawalan “awal\_”) yaitu memanggil code lain yang berisi <form>. Hal ini dikarenakan, setelah user menginputkan data sesuai filter yang dicari, mereka tidak lupa akan apa yang tadinya mereka ingin cari. Berikut elemen yang ditemui dalam halaman <form> tersebut

No	Nama Elemen	Fungsi
1.	<code>mysql_ctype \$res_ctype \$row</code>	Merupakan query yang akan mendapatkan nama tipe Aircraft dari tabel <code>tbl_master_ctype</code>
2.	<code>&lt;select name="ctype"&gt;</code>	Merupakan fungsi dropdown yang akan menampilkan hasil dari query <code>\$res_ctype</code>
3.	<code>&lt;input type="date" class="form-control" name="datefrom" id="id_datefrom"&gt;</code>	Merupakan field yang nantinya akan mendapatkan tanggal mulai data sesuai input user. Bagian ini akan berfungsi optimal apabila menggunakan <i>browser</i> Google Chrome. User apabila menggunakan broser selain Google Crome <b>HARUS</b> mengikuti arahan format. Karena data yang disimpan memiliki format <code>yyyy-mm-dd</code>
4.	<code>&lt;input type="date" class="form-control" name="datefrom" id="id_dateto"&gt;</code>	Merupakan field yang nantinya akan mendapatkan tanggal akhir data sesuai input user. Bagian ini akan berfungsi optimal apabila menggunakan <i>browser</i> Google Chrome. User apabila menggunakan broser selain Google Crome <b>HARUS</b> mengikuti arahan format. Karena data yang disimpan memiliki format <code>yyyy-mm-dd</code>
5.	<code>&lt;input type="submit" value="Display Report" class="btn btn-default"&gt;</code>	Merupakan tombol submit yang menjadi pelatuk action form

Untuk bebrapa halaman, terdapat beberapa input yang unik. Setiap input memiliki peranan dan nama masing masing sesuai dengan code yang telah kami rancang. Input tersebut dapat berupa radio button, checkbox atau sekedar field seperti biasa. Untuk lebih jelasnya dapat dilihat setiap comment pada masing halaman.

Selanjutnya adalah tabel. Tabel yang kami gunakan merupakan plugin dari Datatables. Referensi Datatables dapat diakses melalui [halaman ini](#).

Cara kerja datatables yang kami implementasikan adalah:

#### a) Inisiasi

Pada tahap inisiasi salah satu tabel harus memiliki id yang sesuai seperti code DataTables yang telah developer siapkan di akhir atau tengah script. Sebagai contoh ada pada `component_removal.php`.

```

<div class="panel-body">
  <section id="unseen" style="padding: 10px">
    <button id="exportButton" onclick="generate()" type="button" class="btn btn-default">Export</button>
    <table id="comp_table" class="table table-bordered table-striped">
      <thead>
        <tr>
          <th>No</th>
          <th>Name</th>
          <th>Age</th>
          <th>Gender</th>
          <th>Address</th>
        </tr>
      </thead>
      <tbody>
        <tr>
          <td>1</td>
          <td>John</td>
          <td>25</td>
          <td>Male</td>
          <td>New York</td>
        </tr>
        <tr>
          <td>2</td>
          <td>Jane</td>
          <td>30</td>
          <td>Female</td>
          <td>Los Angeles</td>
        </tr>
      </tbody>
    </table>
  </section>
</div>

```

Gambar 3. Code Untuk memunculkan table pada halaman component\_removal.php

Kemudian code datatables sesuai instruksi pada halaman referensi.

```

<script type="text/javascript">
  $(document).ready(function() {
    $('#comp_table').DataTable({
      "lengthMenu": [[10, 25, 50, 100, -1], [10, 25, 50, 100, "All"]],
      dom: 'Blfrtip',
      buttons: [
        {
          extend: 'excelHtml5', text: 'Export As Excel', className: 'btn btn-default'
        }
      ],
      responsive: true
    });
  });
</script>

```

Gambar 5. Code JQuery untuk datatables halaman component\_removal.php

## b) Customize

Agar developer dapat men-customize DataTable sesuai keinginan, maka disarankan kepada developer untuk membaca referensinya terlebih dahulu. Sebagai contoh, pada bagian component\_removal.php kami menambahkan panjang penyajian data yang dapat diubah, dan adanya tombol untuk men ekspor menjadi format .xls

Kemudian, agar datatables dapat berfungsi dengan sempurna perlu diperhatikan **Urutan Memasukkan Script** pada halaman tersebut. Urutan yang kami gunakan adalah:

```

<script src="https://cdn.datatables.net/1.10.15/js/jquery.dataTables.min.js"></script>
<script
src="https://cdn.datatables.net/buttons/1.3.1/js/dataTables.buttons.min.js"></script>
<script src="//cdnjs.cloudflare.com/ajax/libs/jszip/3.1.3/jszip.min.js"></script>
<script
src="//cdn.rawgit.com/bpampuch/pdfmake/0.1.27/build/pdfmake.min.js"></script>
<script
src="//cdn.rawgit.com/bpampuch/pdfmake/0.1.27/build/vfs_fonts.js"></script>
<script src="//cdn.datatables.net/buttons/1.3.1/js/buttons.html5.min.js"></script>

```

Disamping itu, developer diharapkan juga untuk **Tidak Lupa** menambahkan css dari Datatables itu sendiri.

```
<link rel="stylesheet" type="text/css"
href="//cdn.datatables.net/1.10.15/css/jquery.dataTables.css">
```

Untuk memudahkan penyajian data, kami menggunakan plugin ChartJS untuk membuat Chart sesuai dengan data yang ditampilkan pada tabel. Referensi ChartJS dapat dilihat pada [halaman ini](#). Tahapannya serupa yaitu:

a) Inisiasi

Pada tahap inisiasi salah satu *canvas* harus memiliki id yang sesuai seperti code ChartJS yang telah developer siapkan di tengah atau akhir script. Sebagai contoh ada pada `graph_component.php`.

```
if($row_cnt>0){
    echo "<canvas id=grafik_pareto style='height: 250px; margin-t
}
else {
    echo "<h2>Tidak ada data</h2>";
}
```

Gambar 6. Code untuk memunculkan chart pada halaman `graph_component.php`

Kemudian code ChartJS sesuai instruksi pada halaman referensi, dan menyamakan nama id supaya grafik muncul.

```
Chart.plugins.register({
    beforeDraw: function(chartInstance) {
        var ctx = chartInstance.chart.ctx;
        ctx.fillStyle = "white";
        ctx.fillRect(0, 0, chartInstance.chart.width, chartInstance.chart.height);
    }
});

var ctx = document.getElementById('grafik_pareto').getContext("2d");

var data = {
    labels: part_no,
    datasets: [{
        label: "Number of Component Removal",
        backgroundColor: "red",
        strokeColor: "black",
```

Gambar 7. Potongan code JQuery untuk menampilkan chartJS

Lalu, untuk membuat chart dapat menggunakan code:

```
var myBarChart = new Chart(ctx, {
    type: 'bar',
    data: data,
    options: options
});
```

## b) Customize

Agar developer dapat men-customize Chart sesuai keinginan, maka disarankan kepada developer untuk membaca referensinya terlebih dahulu. Sebagai contoh, pada bagian `graph_component.php` kami memcustomize label, data, title, dan warna chart.

Karena hasil query sudah sesuai, maka hasil query tersebut disimpan dalam array javascript, agar data tersebut dapat ditampilkan pada chart. Karena berbeda format kami menggunakan `jsonwrapper` sebagai alat bantu untuk memanggil fungsi `json_encode`. Lalu data tersebut diisikan ke array yang telah disiapkan untuk mengisi data pada chart.

```
var part_no = [];  
var jumlah_comp = [];  
var z=0;  
  
var arr_comp = <?php echo json_encode($arr_pareto); ?>;  
  
for ( tot=arr_comp.length; z < tot; z++) {  
    part_no.push(arr_comp[z][0]);  
    jumlah_comp.push(arr_comp[z][2]);  
};
```

Gambar 8. Potongan code untuk mendistribusikan data antar array

Selanjutnya adalah menambahkan JSPdf. JSPdf adalah plugin yang dapat mengekspor seluruh halaman ke dalam format .pdf. Untuk lebih lengkapnya, ada referensi yang dapat dirujuk pada [halaman ini](#). Cara pengerjaannya hampir sama seperti plugin sebelumnya yaitu:

### a) Inisiasi

Pada tahap inisiasi salah satu *button* harus memiliki nama fungsi yang sesuai seperti code JSPdf yang telah developer siapkan di tengah atau akhir script. Sebagai contoh ada pada `graph_component.php`.

```
<div class="panel-body">  
<section id="unseen" style="padding: 10px">  
<table id="comp_table" class="table table-borde  
<button id="exportButton" onclick="generate()  
<br>  
<br>  
<br>
```

Gambar 9. Code untuk memunculkan tombol ekspor pada halaman `graph_component.php`

Kemudian code JSPdf sesuai instruksi pada halaman referensi, dan menyamakan nama id dan nama fungsi agar tombol muncul dan berfungsi.



```

function generate() {
    var pdfsize = 'a4';
    var columns = ["No", "Code", "Name"];
    var data = tableToJson($("#comp_table").get(0), columns);
    console.log(data);
    var canvas = document.querySelector("#grafik_pareto");
    var canvasImg = canvas.toDataURL("image/jpeg", 1.0);
    var doc = new jsPDF('l', 'pt', pdfsize);
    var width = doc.internal.pageSize.width;
    doc.autoTable(columns, data, {
        theme: 'grid',
        styles: {
            overflow: 'linebreak'
        },
        pageBreak: 'always',
        tableWidth: 'auto'
    });
    let finalY = doc.autoTable.previous.finalY;
    doc.addPage();
    doc.drawImage(canvasImg, 'JPEG', 40, 40, width-80, 400);
    doc.save("Pareto Component.pdf");
}

```

Gambar 10. Inisiasi pada halaman graph\_component.php

b) Customize

Agar developer dapat men-customize Data yang akan diekspor sesuai keinginan, maka disarankan kepada developer untuk membaca referensinya terlebih dahulu.

2. Halaman Tambahan

Kami memberi sebutan demikian, karena website tersebut sudah ada, dan kami *embed* halaman tersebut ke website dashboard. Halaman tersebut adalah halaman *Oil Consumption* dan *PFR Online*. Dokumentasi kedua halaman tersebut ada pada Aircraft Reliability Management.

3. Halaman pendukung

Halaman pendukung yang kami maksud adalah halaman *User Guide* dan *MTBUR WW*. Kedua halaman tersebut merupakan hasil *embed* dari Google drive. Dimana *MTBUR World Wide* adalah halaman yang berisi dokumen dokumen pendukung. Sedangkan *User Guide* adalah halaman yang berisi tentang cara menggunakan website ini.

Mungkin cukup sekian yang bisa kami jelaskan. Penjelasan lebih lanjut terdapat pada comment pada script kami. Apabila masih ada yang kurang jelas, diharapkan untuk membaca Referensi yang telah kami *Hyperlink* dengan web yang bersangkutan. Apabila memerlukan penjelasan lebih lanjut, dapat menghubungi kami pada:

1. [Rilokukuh@gmail.com](mailto:Rilokukuh@gmail.com) (Kukuh Rilo Pambudi)
2. [Mhabiburr17@gmail.com](mailto:Mhabiburr17@gmail.com) (M. Habibur Rahman)