**Aplikasi rekap absensi file CSV ke *database* pada aplikasi Zoom**

Laporan Ujian Akhir Semester (Multi Channel Access)



Efraim William Solang 1805551148

Dosen Pengampu :

Dr. I Made Sukarsa, ST.,MT

Teknologi Informasi

Fakultas Teknik

Universitas Udayana

2020

1. Gambaran Umum Sistem

Berikut ini merupakan gambaran umum sistem yang dirancang berdasarkan dengan *project* yang telah di buat.

|  |
| --- |
|  |

Gambar 1. Gambaran umum sistem

Gambar di atas merupakan gambaran umum sistem Aplikasi rekap absensi file CSV ke *database* pada aplikasi Zoom. Gambaran umum sistem dibagi menjadi tiga bagian, yaitu User atau pengguna, PHP sebagai *processing* data dan Database untuk menyimpan data. Gambaran umum dimulai dari User yang menginputkan file rekapan Zoom dengan format CSV ke halaman PHP. Selanjutnya setelah di*import*, data tersebut akan masuk ke *database* “import”. Setelanjutnya PHP membuat *procedure* “get\_date” untuk mengambil kolom tanggal yang akan disimpan pada tabel “tb\_date”. Selanjutnya PHP akan membuat *procedure* “rekap” yang berfungsi untuk merekap data dengan mengambil kolom tertentu dan akan disimpan pada tabel “rekap” yang akan disamakan kecocokan nya pada tabel “tb\_date” sehingga tidak memiliki perulangan data atau duplikasi data. Setelah itu, PHP akan membuat *procedure* selanjutnya yaitu *procedure* “get\_username” dan “get\_username1” yang berfungsi untuk merekap user yang hadir dengan range jam 8 sampai jam 10 serta jam 15 sampai jam 18 pada tabel “tb\_kelas\_pertama” dan “tb\_kelas\_kedua” dengan mengambil referensi dari “tb\_master” yang bersikan topic dan *range* yang ada.

Setelah semua *procedure* telah dibuat dan dijalankan sesuai dengan urutan mereka, maka Langkah terakhir yaitu PHP akan mengambil data dari tabel “tb\_kelas\_pertama” dan “tb\_kelas\_kedua” untuk ditampilkan di layar PHP.

1. Teori
   1. CSV

File CSV (Nilai Berbatas Koma) adalah tipe file khusus yang dapat Anda buat atau edit di Excel. File CSV menyimpan informasi yang dipisahkan oleh koma, bukan menyimpan informasi dalam kolom. Saat teks dan angka disimpan dalam file CSV, mudah untuk memindahkannya dari satu program ke program lain. Misalnya, Anda dapat mengekspor kontak dari Google ke dalam file CSV, kemudian mengimpornya ke Outlook.

* 1. PHP

PHP adalah bahasa *scripting server-side*, Bahasa pemrograman yang digunakan untuk mengembangkan situs web statis atau situs web dinamis atau aplikasi Web. PHP disebut bahasa pemrograman server side karena PHP diproses pada komputer server. Hal ini berbeda dibandingkan dengan bahasa pemrograman client-side seperti JavaScript yang diproses pada web browser (client).

PHP merupakan singkatan dari *Hypertext Pre-processor*, yang sebelumnya disebut *Personal Home Pages*. Sesuai dengan namanya, PHP digunakan untuk membuat website pribadi. Dalam beberapa tahun perkembangannya, PHP menjelma menjadi bahasa pemrograman web yang powerful dan tidak hanya digunakan untuk membuat halaman web sederhana, tetapi juga website populer yang digunakan oleh jutaan orang seperti Wikipedia, Wordpress, Jomla, dll.

* 1. SQL

*Structured Query Language* (SQL) adalah sistem manajemen database relasional (RDBMS) yang dirancang untuk aplikasi dengan arsitektur *client*/*server*. Istilah *client*, *server*, dan *client*/*server* dapat digunakan untuk merujuk kepada konsep yang sangat umum atau hal yang spesifik dari perangkat keras atau perangkat lunak.

Pada umumnya terdapat 3 (tiga) jenis perintah SQL yang bisa digunakan oleh SQL, yaitu: DDL (*Data Definition Language*), DML (*Data Manipulation Language*), dan DCL (*Data Control Language*).

* 1. PLSQL

*Procedural Language*/*Structured Query Language* (PL/SQL) adalah generasi ke-4 (4GL: *Fourth–Generation Languages*) dari bahasa pemrograman dimana merupakan sebuah penggabungan antara bahasa pemrograman prosedural (PL) dan SQL syntax. Berbeda dengan SQL yg hanya mementingkan bagaimana mendapatkan hasil, dengan PL/SQL juga akan diketahui bagaimana proses untuk mendapatkan hasil tersebut. Dikarenakan PL/SQL merupakan bahasa prosedural, maka statement/syntax yg digunakan sangat tergantung dari jenis bahasa yg diadopsi.

Dalam PL/SQL dapat digunakan perintah untuk memanipulasi data yang ada dalam database MySQL ataupun Oracle dan membentuk pemrograman terstruktur dalam memproses data. Pada PL/SQL ditambahkan beberapa hal yang dikenal pada dunia pemrograman, seperti variabel, *loop*, pemrosesan kondisi, operasi *cursor*, modularisasai, dan hal-hal lainnya. Semua tambahan itu bertujuan untuk meningkatkan kinerja operasi-operasi SQL pada database sehingga manfaat dari kehandalannya menjadi maksimal.

* 1. Database

Database atau basis data adalah kumpulan data yang dikelola sedemikian rupa berdasarkan ketentuan tertentu yang saling berhubungan sehingga mudah dalam pengelolaannya. Melalui pengelolaan tersebut pengguna dapat memperoleh kemudahan dalam mencari informasi, menyimpan informasi dan membuang informasi.

Adapun pengertian lain dari *database* adalah sistem yang berfungsi sebagai mengumpulkan file, tabel, atau arsip yang terhubung dan disimpan dalam berbagai media elektronik.

1. Hasil

Berikut ini merupakan hasil dari dari aplikasi rekap absensi file CSV ke database pada aplikasi Zoom.

|  |
| --- |
|  |

Gambar 3.1 Tampilan utama program

Gambar di atas merupakan tampilan utama program aplikasi rekap absensi file CSV ke database pada aplikasi Zoom. User dapat melakukan upload file dengan memilih data pada lokal direktori user yang ingin di *import* dengan file CSV.

|  |
| --- |
|  |

Gambar 3.2 Hasil rekap absensi

Gambar di atas merupakan hasil rekap absensi pada aplikasi rekap absensi file CSV ke database pada aplikasi Zoom. Terlihat pada gambar di atas bahwa program telah menampilkan user yang terdapat pada kelas jam 8 sampai jam 10 dengan tanggal yang berbeda-beda dan dengan range waktu jam 8 sampai jam 10.

|  |
| --- |
|  |

Gambar 3.3 Hasil rekap absensi

Gambar di atas merupakan lanjutan dari hasil rekap absensi pada aplikasi rekap absensi file CSV ke *database* pada aplikasi Zoom. Terlihat pada gambar di atas bahwa program telah menampilkan *user* yang terdapat pada kelas jam 15 sampai jam 18 dengan tanggal yang berbeda-beda dan dengan range waktu jam 15 sampai jam 18. Terlihat bahwa pada gambar di atas menunjukan bahwa *user* tidak memiliki perulangan data atau duplikat data pada kolom *username*.

1. Database

Berikut ini merupakan database yang digunakan pada aplikasi rekap absensi file CSV ke database pada aplikasi Zoom.

|  |
| --- |
|  |

Gambar 4. Database aplikasi

Gambar di atas merupakan database pada aplikasi rekap absensi file CSV ke database pada aplikasi Zoom. Tabel yang digunakan pada aplikasi ini adalah tabel ”import” yang berfungsi untuk menyimpan seluruh data dari file CSV. Tabel ”rekap” berfungsi untuk merekap data dengan beberapa kolom sesuai dengan yang diinginkan untuk di rekap. Tabel ”tb\_date” berfungsi untuk mengambil kolom tanggal yang akan digunakan untuk mencocokan tanggal dengan *user* dari tabel ”import”. Tabel ”tb\_kelas\_pertama” dan ”tb\_kelas\_kedua” digunakan untuk merekap *user* yang hadir pada jam 8-10 sertai jam 15-18 dengan spesifikasi tanggal dan jam hadirnya. Tabel ”tb\_master” berfungsi sebagai tabel yang digunakan sebagai referensi *range* yang ada, sehingga ketika user sudah direkap ke ”tb\_kelas\_pertama” dan ”tb\_kelas\_kedua”, maka kedua tabel akan direkap berdasarkan dengan tabel *master* yang menunjukan *range* jam.

Database ”db\_absensi” juga terdapat *storage* *procedure* yang digunakan untuk merekap data. Procedure ”get\_date” digunakan untuk merekap data tanggal yang akan disimpan pada ”tb\_date”. Procedure ”rekap” digunakan untuk merekap data dari tabel ”import” yang disesuaikan dengan ”tb\_date”. Procedure ”get\_username” dan ”get\_username1” digunakan untuk merekap data user yang hadir ke tabel ”tb\_kelas\_pertama” dan ”tb\_kelas\_kedua”.

* 1. Isi tabel pada database

Berikut ini merupakan isi tabel pada database yang digunakan pada aplikasi rekap absensi file CSV ke database pada aplikasi Zoom.

|  |
| --- |
|  |

Gambar 4.1.1 Isi tabel ”import”

|  |
| --- |
|  |

Gambar 4.1.2 Isi tabel ”rekap”

|  |
| --- |
|  |

Gambar 4.1.3 Isi tabel ”tb\_date”

|  |
| --- |
|  |

Gambar 4.1.4 Isi tabel ”tb\_kelas\_pertama”

|  |
| --- |
|  |

Gambar 4.1.5. Isi tabel ”tb\_limabelas”

|  |
| --- |
|  |

Gambar 4.1.6. Isi tabel ”tb\_master”

1. Source Code

Berikut ini merupakan source code yang digunakan pada aplikasi rekap absensi file CSV ke database pada aplikasi Zoom. Source code yang digunakan juga dapat diunduh pada link berikut ini <https://github.com/efraimwilliam/MCARekapAbsensiZoom.git>.

testhampirfix.php

|  |
| --- |
| <?php  error\_reporting(0);  $conn = mysqli\_connect("localhost", "root", "", "db\_absensi");  if(isset($\_POST["import"])){  $filename = $\_FILES["file"]["tmp\_name"];  $index = 1;  if($\_FILES["file"]["size"] > 0){  $file = fopen($filename, "r");    while(($column = fgetcsv($file, 10000, ",")) !==FALSE){  if($index==1){  $index++;  continue;  }  $column1 = mysqli\_real\_escape\_string($conn, $column[0]);  $column2 = mysqli\_real\_escape\_string($conn, $column[1]);  $column3 = mysqli\_real\_escape\_string($conn, $column[2]);  $column4 = mysqli\_real\_escape\_string($conn, $column[3]);  $column5 = mysqli\_real\_escape\_string($conn, $column[4]);  $column6 = mysqli\_real\_escape\_string($conn, $column[5]);  $column7 = mysqli\_real\_escape\_string($conn, $column[6]);  $column8 = mysqli\_real\_escape\_string($conn, $column[7]);  $column9 = mysqli\_real\_escape\_string($conn, $column[8]);  $column10 = mysqli\_real\_escape\_string($conn, $column[9]);  $column11 = mysqli\_real\_escape\_string($conn, $column[10]);  $column12 = mysqli\_real\_escape\_string($conn, $column[11]);  $column13 = mysqli\_real\_escape\_string($conn, $column[13]);  $column14 = mysqli\_real\_escape\_string($conn, $column[14]);  $column15 = mysqli\_real\_escape\_string($conn, $column[15]);  $column16 = mysqli\_real\_escape\_string($conn, $column[16]);  $column17 = mysqli\_real\_escape\_string($conn, $column[17]);  $column18 = mysqli\_real\_escape\_string($conn, $column[18]);    //$sqlInsert = "Insert into data (nama, tipe) values ('" . $column1 . "', '" . $column2 ."')";  $sqlInsert = "Insert into import (topic, meeting\_id, username, email, department, group\_a, haszoom,  creation\_time, start\_time, end\_time, duration, participants, padma\_name, usermail, join\_time, leave\_time,  duration\_user, recording)  values  ('" . $column1 . "', '" . $column2 ."', '" . $column3 . "', '" . $column4 . "','" . $column5 . "',  '" . $column6 . "','" . $column7 . "','" .$column8 . "','" . $column9 . "','" . $column10 . "',  '" . $column11 . "','" . $column12 . "','" . $column13 . "','" . $column14 . "','" .$column15 . "','" . $column16 . "',  '" . $column17 . "','" . $column18 . "')";    //print\_r($column);  $result = mysqli\_query($conn, $sqlInsert);  if(!$result){  echo "Error";  }  }  }  }  ?>  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>  <form class = "form-horizoontal" action= "" method = "post" name = "uploadCsv" enctype = "multipart/form-data">  <div class = "container">  <h2 align="center">Upload Absensi CSV</a></h2>  <p><label>Choose File</label></p>  <input type ="file" name = "file" accept = ".csv">  <p><button type = "submit" name = "import" id = "first-button">Import</button></p>  </div>  </form>  <?php  $servername = "localhost";  $username = "root";  $password = "";  $dbname = "db\_absensi";  // Create connection  $conn = new mysqli($servername, $username, $password, $dbname);  $sqlSelect = "SELECT \* from tb\_kelas\_pertama";  $sqlSelect1 = "SELECT \* from tb\_kelas\_kedua";  $result = mysqli\_query($conn, $sqlSelect);  $result1 = mysqli\_query($conn, $sqlSelect1);  if(mysqli\_num\_rows($result)>0){  ?>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>    <div class = "container">  <table class = "table">  <h2 align="center">Hasil Rekapan</h2>  <p><h3 align="left">Kelas Jam 8-10</h3></p>  <thead>  <tr>  <th>Username</th>  <th>Date</th>  <th>Start</th>  <th>Finished</th>  </tr>  </thead>  <?php  while ($row = mysqli\_fetch\_array($result)){  ?>  <tbody>  <tr>  <td><?php echo $row["username"];?></td>  <td><?php echo $row["time"];?></td>  <td><?php echo $row["waktu"];?></td>  <td><?php echo $row["waktu1"];?></td>  </tr>  </tbody>  <?php } ?>  </table>  </div>    <?php  }  if(mysqli\_num\_rows($result1)>0){  ?>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>    <div class = "container">  <table class = "table">  <h2 align="center">Hasil Rekapan</h2>  <p><h3 align="left">Kelas Jam 15-18</h3></p>  <thead>  <tr>  <th>Username</th>  <th>Date</th>  <th>Start</th>  <th>Finished</th>  </tr>  </thead>  <?php  while ($row = mysqli\_fetch\_array($result1)){  ?>  <tbody>  <tr>  <td><?php echo $row["username"];?></td>  <td><?php echo $row["time"];?></td>  <td><?php echo $row["waktu"];?></td>  <td><?php echo $row["waktu1"];?></td>  </tr>  </tbody>  <?php } ?>  </table>  </div>    <?php  }  ?>  <?php  $mysqli = new mysqli("localhost", "root", "", "db\_absensi");  if ($mysqli->connect\_errno) {  echo "Failed to connect to MySQL: (" . $mysqli->connect\_errno . ") " . $mysqli->connect\_error;  }  if (!$mysqli->query("DROP PROCEDURE IF EXISTS get\_date") ||  !$mysqli->query('CREATE PROCEDURE get\_date()    BEGIN  DELETE FROM tb\_date;  INSERT INTO tb\_date (`date`)  SELECT DISTINCT DATE(STR\_TO\_DATE(creation\_time, "%m/%d/%Y %H:%i:%s")) AS q FROM `import`;    END;')){  echo "Stored procedure creation failed: (" . $mysqli->errno . ") " . $mysqli->error;  }  if (!$mysqli->query("CALL get\_date()")) {  echo "CALL failed: (" . $mysqli->errno . ") " . $mysqli->error;  }  ?>  <?php  $mysqli = new mysqli("localhost", "root", "", "db\_absensi");  if ($mysqli->connect\_errno) {  echo "Failed to connect to MySQL: (" . $mysqli->connect\_errno . ") " . $mysqli->connect\_error;  }  if (!$mysqli->query("DROP PROCEDURE IF EXISTS rekap") ||  !$mysqli->query('CREATE PROCEDURE rekap()    BEGIN  DECLARE cek\_date VARCHAR (60);  DECLARE finished INTEGER DEFAULT 0;  DECLARE cur\_date CURSOR FOR SELECT tb\_date.`date` FROM tb\_date;  DECLARE CONTINUE HANDLER FOR NOT FOUND SET finished = 1;  OPEN cur\_date;  TRUNCATE TABLE rekap;  date\_loop: LOOP  FETCH cur\_date INTO cek\_date;  IF finished = 1 THEN  LEAVE date\_loop;  END IF;    INSERT INTO rekap (topic, username, durations, finished, DAY)  SELECT topic, padma\_name,  HOUR(STR\_TO\_DATE(join\_time, "%m/%d/%Y %H:%i:%s")),  HOUR(STR\_TO\_DATE(end\_time, "%m/%d/%Y %H:%i:%s")),  DATE(STR\_TO\_DATE(creation\_time, "%m/%d/%Y %H:%i:%s"))  FROM `import` WHERE creation\_time != 0 AND DATE(STR\_TO\_DATE(creation\_time, "%m/%d/%Y %H:%i:%s")) = cek\_date  GROUP BY padma\_name;  END LOOP;  CLOSE cur\_date;      END;')){  echo "Stored procedure creation failed: (" . $mysqli->errno . ") " . $mysqli->error;  }  if (!$mysqli->query("CALL rekap()")) {  echo "CALL failed: (" . $mysqli->errno . ") " . $mysqli->error;  }  ?>  <?php  $mysqli = new mysqli("localhost", "root", "", "db\_absensi");  if ($mysqli->connect\_errno) {  echo "Failed to connect to MySQL: (" . $mysqli->connect\_errno . ") " . $mysqli->connect\_error;  }  if (!$mysqli->query("DROP PROCEDURE IF EXISTS get\_username") ||  !$mysqli->query('CREATE PROCEDURE get\_username()    BEGIN  DECLARE cek\_master VARCHAR (60);  DECLARE cek\_master1 VARCHAR (60);  DECLARE finished INTEGER DEFAULT 0;  DECLARE cur\_master CURSOR FOR SELECT tb\_master.`start` FROM tb\_master WHERE tb\_master.`id` = 1;  DECLARE cur\_master1 CURSOR FOR SELECT tb\_master.`finish` FROM tb\_master WHERE tb\_master.`id` = 1;  DECLARE CONTINUE HANDLER FOR NOT FOUND SET finished = 1;  OPEN cur\_master;  OPEN cur\_master1;  TRUNCATE TABLE tb\_kelas\_pertama;  date\_loop: LOOP  FETCH cur\_master INTO cek\_master;  IF finished = 1 THEN  LEAVE date\_loop;  END IF;    FETCH cur\_master1 INTO cek\_master1;  IF finished = 1 THEN  LEAVE date\_loop;  END IF;  INSERT INTO tb\_kelas\_pertama (username, `time`, `waktu`, waktu1)  SELECT username,`day`, durations, rekap.`finished`  FROM `rekap` WHERE durations >= cek\_master AND rekap.`finished`<=cek\_master1;  END LOOP;  CLOSE cur\_master;  CLOSE cur\_master1;        END;')){  echo "Stored procedure creation failed: (" . $mysqli->errno . ") " . $mysqli->error;  }  if (!$mysqli->query("CALL get\_username()")) {  echo "CALL failed: (" . $mysqli->errno . ") " . $mysqli->error;  }  ?>  <?php  $mysqli = new mysqli("localhost", "root", "", "db\_absensi");  if ($mysqli->connect\_errno) {  echo "Failed to connect to MySQL: (" . $mysqli->connect\_errno . ") " . $mysqli->connect\_error;  }  if (!$mysqli->query("DROP PROCEDURE IF EXISTS get\_username1") ||  !$mysqli->query('CREATE PROCEDURE get\_username1()    BEGIN  DECLARE cek\_master VARCHAR (60);  DECLARE cek\_master1 VARCHAR (60);  DECLARE finished INTEGER DEFAULT 0;  DECLARE cur\_master CURSOR FOR SELECT tb\_master.`start` FROM tb\_master WHERE tb\_master.`id` = 2;  DECLARE cur\_master1 CURSOR FOR SELECT tb\_master.`finish` FROM tb\_master WHERE tb\_master.`id` = 2;  DECLARE CONTINUE HANDLER FOR NOT FOUND SET finished = 1;  OPEN cur\_master;  OPEN cur\_master1;  TRUNCATE TABLE tb\_kelas\_kedua;  date\_loop: LOOP  FETCH cur\_master INTO cek\_master;  IF finished = 1 THEN  LEAVE date\_loop;  END IF;    FETCH cur\_master1 INTO cek\_master1;  IF finished = 1 THEN  LEAVE date\_loop;  END IF;  INSERT INTO tb\_kelas\_kedua (username, `time`, `waktu`, waktu1)  SELECT username,`day`, durations, rekap.`finished`  FROM `rekap` WHERE durations >= cek\_master AND rekap.`finished`<=cek\_master1;  END LOOP;  CLOSE cur\_master;  CLOSE cur\_master1;    END;')){  echo "Stored procedure creation failed: (" . $mysqli->errno . ") " . $mysqli->error;  }  if (!$mysqli->query("CALL get\_username1()")) {  echo "CALL failed: (" . $mysqli->errno . ") " . $mysqli->error;  }  ?> |

Kode Program 5.1. Kode program PHP

1. Saran atau Perbaikan

Saran dan perbaikan yang dapat diberikan pada program aplikasi ini adalah sebagai berikut :

1. Memperbaiki kode program sehingga memberikan ketepatan data
2. Memperbaiki API dari program aplikasi
3. Memperbaiki aplikasi dengan menggunakan tabel master yang berisikan jadwal kelas berdasarkan waktu, dan kemudian mengacu pada master tabel tersebut.
4. Memperbaiki kode program PHP menggunakan JSON