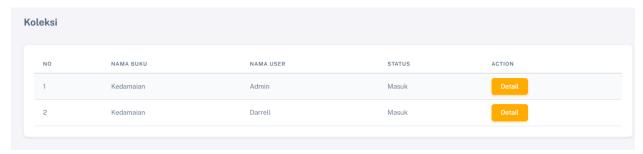


```
protected function isLoggedIn()
    return session()->has('id');
public function index()
    if ($this->isLoggedIn()) {
        return redirect()->to('dashboard');
    $data['title']='Login';
    echo view ('partial/header', $data);
    echo view('login/view');
    echo view('partial/footer');
 public function aksi_login()
    $u=$this->request->getPost('username');
    $p=$this->request->getPost('pswd');
    $model= new M_model();
    $data=array(
        'username'=>$u,
         'password'=>md5($p)
     );
    $cek=$model->getWhere2('user', $data);
    if ($cek>0) {
         session()->set('id', $cek['id_user']);
        session()->set('username', $cek['username']);
        session()->set('level', $cek['level']);
        return redirect()->to('dashboard');
        return redirect()->to('/');
public function log_out()
    session()->destroy();
    return redirect()->to('login');
```

Kode ini memiliki fungsi untuk manajemen login dan logout. Jika pengguna sudah login (sesi 'id' ada), pengguna akan diarahkan ke halaman dashboard. Jika belum login, halaman login akan ditampilkan. Fungsi aksi_login mengelola proses login dengan memeriksa username dan password dari formulir, kemudian membuat sesi jika informasi tersebut valid. Fungsi log_out menghancurkan semua sesi dan mengarahkan pengguna ke halaman login.



```
public function index()
    if (session()->get('id') > 0) {
        $model = new M_model();
        $userLevel = session()->get('level');
        if ($userLevel == 1 || $userLevel == 2) {
            $on = 'koleksi.book_id = book.id_book';
            $on1 = 'koleksi.user_id = user.id_user';
            $data['a'] = $model->join3('koleksi', 'book', 'user', $on, $on1);
        } elseif ($userLevel == 3) {
            $userId = session()->get('id');
            $data['a'] = $model->getKoleksiDataByIdUser($userId);
        $data['title'] = 'Koleksi';
        echo view('partial/header', $data);
        echo view('partial/menuutama');
        echo view('koleksi/view', $data);
        echo view('partial/footer');
        return redirect()->to('login');
```

Kode ini menentukan tampilan halaman koleksi berdasarkan level pengguna. Jika pengguna sudah login (sesi 'id' ada), kode memeriksa level pengguna. Jika level 1 atau 2, melakukan join tabel koleksi, book, dan user, kemudian menampilkan data koleksi. Jika level 3, langsung mengambil data koleksi berdasarkan ID pengguna. Selanjutnya, kode menampilkan halaman koleksi dengan header, menu utama, dan footer. Jika pengguna belum login, akan diarahkan ke halaman login.

```
public function index()
       if (session()->get('id') > 0) {
       $model=new M_model();
       $data['jojo']=$model->tampil('kategori');
       $data['title']='Data Kategori';
       echo view('partial/header', $data);
       echo view('partial/menuutama');
       echo view('kategori/view', $data);
       echo view('partial/footer');
       return redirect()->to('login');
public function tambah() {
       $model = new M_model();
       if (session()->get('id') > 0) {
       $data['title'] = 'Tambah Kategori';
       echo view('partial/header',$data);
       echo view('partial/menuutama');
       echo view('kategori/tambah', $data);
       echo view('partial/footer');
public function aksi_tambah()
   if(session()->get('level')== 1) {
       $a= $this->request->getPost('nama_k');
       $data1=array(
            'nama_k'=>$a,
            'created_at'=>date('Y-m-d H:i:s')
       );
       $model=new M_model();
       $model->simpan('kategori', $data1);
       return redirect()->to('kategori');
   }else {
       return redirect()->to('login');
oublic function edit($id)
       $model=new M_model();
       $where=array('id_kategori'=>$id);
       $data['a']=$model->getWhere('kategori',$where);
```

```
public function edit($id)
        $model=new M_model();
        $where=array('id_kategori'=>$id);
        $data['a']=$model->getWhere('kategori',$where);
        $data['title']='Data Kategori';
        echo view('partial/header', $data);
        echo view('partial/menuutama');
        echo view('kategori/edit',$data);
        echo view('partial/footer');
public function aksi_edit($id)
    if(session()->get('level')== 1) {
       $a= $this->request->getPost('nama_k');
       $id= $this->request->getPost('id');
        $where=array('id_kategori'=>$id);
       $data1=array(
           'nama_k'=>$a,
       $model=new M model();
       $model->qedit('kategori', $data1, $where);
        return redirect()->to('kategori');
      return redirect()->to('login');
  public function delete($id)
      if(session()->get('level')== 1) {
       $model=new m_model();
       $where=array('id_kategori'=>$id);
       $model->hapus('kategori',$where);
       return redirect()->to('kategori');
          return redirect()->to('login');
```

Kode ini adalah bagian dari suatu aplikasi yang menangani operasi CRUD untuk entitas 'kategori'.

Fungsi index menampilkan daftar kategori. Jika pengguna sudah login, menampilkan daftar kategori dengan menggunakan model untuk mengambil data dari tabel 'kategori'. Jika belum login, mengarahkan ke halaman login.

Fungsi tambah menampilkan halaman untuk menambahkan kategori baru. Hanya dapat diakses oleh pengguna yang sudah login (level 1).

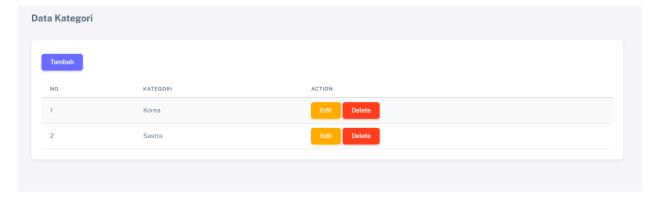
Fungsi aksi_tambah menangani aksi penambahan kategori baru. Hanya dapat diakses oleh pengguna dengan level 1.

Fungsi edit menampilkan halaman untuk mengedit kategori berdasarkan ID-nya. Hanya dapat diakses oleh pengguna yang sudah login (level 1).

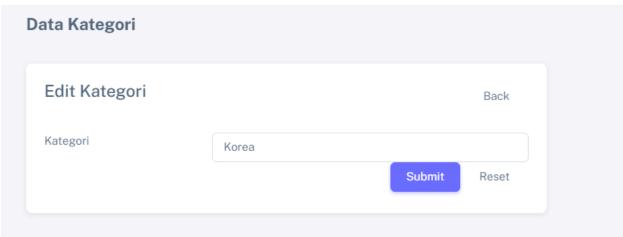
Fungsi aksi_edit menangani aksi pengeditan kategori. Hanya dapat diakses oleh pengguna dengan level 1.

Fungsi delete menangani penghapusan kategori berdasarkan ID-nya. Hanya dapat diakses oleh pengguna dengan level 1.

Semua fungsi tertentu hanya dapat diakses oleh pengguna dengan level 1. Jika pengguna belum login, dia akan diarahkan ke halaman login.







```
public function batalkan_koleksi($id_book)
    $model = new KoleksiModel();
    $id_user = session()->get('id');
$condition = ['book_id' => $id_book, 'user_id' => $id_user];
     $model->deleteKoleksi($condition);
    return redirect()->to('/book');
    public function masukkan_koleksi($id_book)
         $model = new KoleksiModel();
$id_user = session()->get('id'); // Ganti sesuai dengan bagaimana Anda mengelola user session
             'book_id' => $id_book,
             'status' => 'Masuk',
'created_at'=>date('Y-m-d H:i:s'),
         $model->insertKoleksi($data);
         return redirect()->to('/book');
    public function commentForm($postId)
         $postModel = new M_book();
         $commentModel = new CommentModel();
         $post = $postModel->getPostById($postId); // Menggunakan $postId
$comments = $commentModel->getCommentsByPost($postId); // Menggunakan $postId
         $data = [
   'book_id' => $postId,
              'post' => $post,
               'comments' => $comments.
         echo view('partial/header',$data);
         echo view('partial/menuutama');
```

```
public function submitComment()
{
    $commentModel = new CommentModel();
    $userId = session()->get('id');
    $postId = $this->request->getPost('post_id');
    $commentText = $this->request->getPost('comment');

    // Simpan komentar ke database menggunakan CommentModel
    $commentModel->addComment(['user_id' => $userId, 'book_id' => $postId, 'comment' => $commentText]);

    // Redirect kembali ke halaman viewAlbum dengan menyertakan id_album
    $postModel = new M_book();
    $post = $postModel->getPostById($postId);

    return redirect()->to(base_url('/book/commentForm/' . $postId));
}
```

Kode ini adalah bagian dari suatu aplikasi yang menangani operasi CRUD untuk entitas 'buku' dan juga melibatkan fungsi terkait dengan koleksi dan komentar. Berikut adalah penjelasan singkat:

index: Menampilkan data buku dengan informasi kategori. Jika pengguna belum login, akan diarahkan ke halaman login.

tambah: Menampilkan halaman untuk menambahkan buku baru. Hanya dapat diakses oleh pengguna dengan level 1 atau 2.

aksi_tambah: Menangani aksi penambahan buku baru. Hanya dapat diakses oleh pengguna dengan level 1 atau 2.

edit: Menampilkan halaman untuk mengedit buku berdasarkan ID-nya. Hanya dapat diakses oleh pengguna dengan level 1.

update: Menangani aksi pembaruan buku berdasarkan ID-nya. Hanya dapat diakses oleh pengguna dengan level 1 atau 2.

delete: Menangani penghapusan buku berdasarkan ID-nya. Hanya dapat diakses oleh pengguna dengan level 1.

detail: Menampilkan halaman detail buku berdasarkan ID-nya.

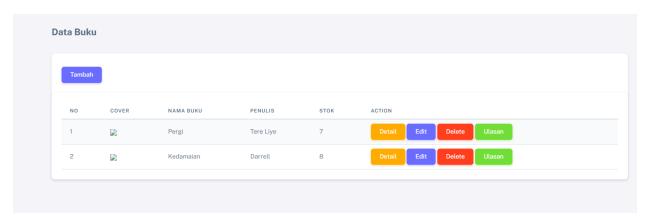
batalkan_koleksi: Menghapus buku dari koleksi pengguna.

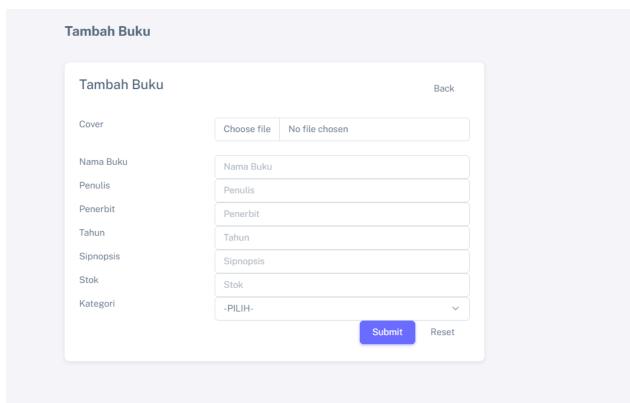
masukkan_koleksi: Menambahkan buku ke dalam koleksi pengguna.

commentForm: Menampilkan halaman formulir komentar untuk suatu buku.

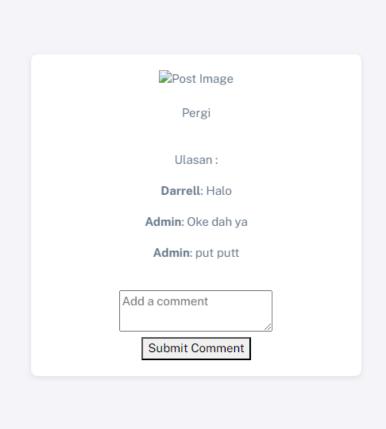
submitComment: Menangani penambahan komentar untuk suatu buku.

Sebagian besar fungsi memeriksa tingkat akses pengguna sebelum melanjutkan aksi CRUD tertentu. Jika pengguna belum login, mereka akan diarahkan ke halaman login.





Pergi Cover Buku Penulis: Tere Liye Penerbit: da.co.book Tahun: 2022 Sipnopsis: Ya Stok: 7 Kategori: Korea



```
public function index()
    if (session()->get('id') > 0) {
        $model = new M_model();
         $userLevel = session()->get('level');
         if ($userLevel == 1 || $userLevel == 2) {
             $on = 'peminjaman.id_book = book.id_book';
             $on1 = 'peminjaman.id_user = user.id_user';
             $data['a'] = $model->join3('peminjaman', 'book', 'user', $on, $on1);
         } elseif ($userLevel == 3) {
             $userId = session()->get('id');
             $data['a'] = $model->getDataByIdUser($userId);
         $data['title'] = 'Data Peminjaman';
        echo view('partial/header', $data);
        echo view('partial/menuutama');
        echo view('peminjaman/view', $data);
echo view('partial/footer');
        return redirect()->to('login');
public function tambah()
    if (session()->get('id') > 0) {
         $model=new M_model();
         $data['a']=$model->tampil('user');
        $data['c']=$model->tampil('book');
$data['title']='Data Peminjaman';
        echo view('partial/header', $data);
        echo view('partial/menuutama');
echo view('peminjaman/tambah', $data);
         echo view('partial/footer');
        return redirect()->to('login');
```

```
public function aksi_tambah()
    if (session()->get('id') > 0) {
       $a= $this->request->getPost('user');
       $b= $this->request->getPost('book');
       $c= $this->request->getPost('tgl_kembali');
$d= $this->request->getPost('jumlah');
        //Yang ditambah ke user
        $data1=array(
            'id_user'=>$a,
            'id_book'=>$b,
            'tgl_kembali'=>$c,
            'jumlah'=>$d,
            'tgl_pinjam'=>date('Y-m-d'),
            'status'=>'Dipinjam'
        $model=new M model();
        $model->simpan('peminjaman', $data1);
        return redirect()->to('peminjaman');
        return redirect()->to('login');
public function delete($id)
    $model=new m_model();
    $where=array('id_peminjaman'=>$id);
    $model->hapus('peminjaman',$where);
     return redirect()->to('peminjaman');
public function aksi($id_peminjaman)
    $model = new m model();
   $peminjaman = $model->getpeminjamanById($id_peminjaman);
    $newStatus = ($peminjaman->status === 'Dipinjam') ? 'Dikembalikan' : 'Dipinjam';
    $data = array('status' => $newStatus);
    $where = array('id_peminjaman' => $id_peminjaman);
    $model->qedit('peminjaman', $data, $where);
    return redirect()->to('/peminjaman');
```

Kode ini adalah bagian dari suatu aplikasi yang menangani operasi peminjaman buku. Berikut adalah penjelasan singkat:

index: Menampilkan data peminjaman buku. Jika pengguna belum login, akan diarahkan ke halaman login.

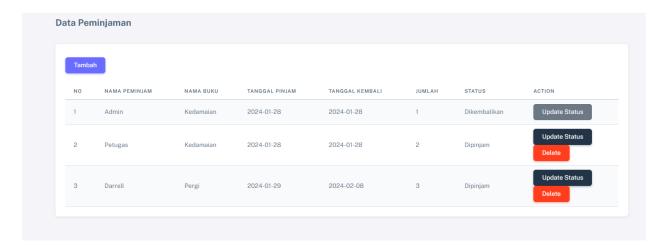
tambah: Menampilkan halaman untuk menambahkan data peminjaman buku. Hanya dapat diakses oleh pengguna yang sudah login.

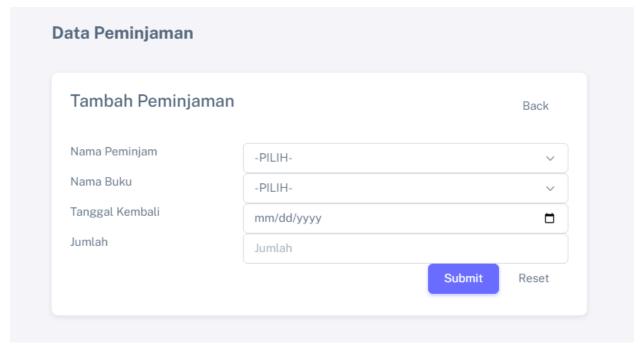
aksi_tambah: Menangani aksi penambahan data peminjaman buku. Hanya dapat diakses oleh pengguna yang sudah login.

delete: Menghapus data peminjaman buku berdasarkan ID-nya. Hanya dapat diakses oleh pengguna yang sudah login.

aksi: Mengubah status peminjaman buku antara 'Dipinjam' dan 'Dikembalikan'. Hanya dapat diakses oleh pengguna yang sudah login.

Fungsi-fungsi ini mengelola operasi CRUD dan mengakses data dari tabel 'peminjaman', 'user', dan 'book'. Sebagian besar fungsi memeriksa apakah pengguna sudah login sebelum melanjutkan operasi. Jika belum login, mereka akan diarahkan ke halaman login.





```
public function index()
     if(session()->get('level')==1 || session()->get('level')==2){
         $model=new M_model();
$data['title']='Data Peminjaman';
$kui['kunci']='print_in';
echo view('partial/header',$data);
         echo view('partial/menuutama');
echo view('filter_peminjaman',$kui);
echo view('partial/footer');
     }else{
         return redirect()->to('/login');
public function print_in()
     if(session()->get('level')==1 || session()->get('level')==2 ){
$model = new M_model();
$awal = $this->request->getPost('awal');
$akhir = $this->request->getPost('akhir');
$status = "Dipinjam";
$kui['duar'] = $model->filter22('peminjaman', $awal, $akhir, $status);
echo view('v_pm', $kui);
return redirect()->to('/login');
public function pdf_in()
     if(session()->get('level')==1 || session()->get('level')==2 ){
         $model = new M_model();
$awal = $this->request->getPost('awal');
         $akhir = $this->request->getPost('akhir');
         $kui['duar'] = $model->filter22('peminjaman', $awal, $akhir, $status);
     $dompdf = new\Dompdf\Dompdf();
     $dompdf->loadHtml(view('v_pm',$kui));
$dompdf->setPaper('A4','potrait');
    $dompdf->render();
$dompdf->stream('my.pdf', array('Attachment'=>0));
     return redirect()->to('/login');
```

```
public function excel in()
    if(session()->get('level')==1 || session()->get('level')==2 ){
        $model = new M model();
        $awal = $this->request->getPost('awal');
        $akhir = $this->request->getPost('akhir');
        $status = "Dipinjam";
        $data = $model->filter22('peminjaman', $awal, $akhir, $status);
    $spreadsheet=new Spreadsheet();
    $spreadsheet->setActiveSheetIndex(0)
    ->setCellValue('A1', 'Nama Peminjam')
    ->setCellValue('B1', 'Nama Buku')
   ->setCellValue('C1', 'Tanggal Pinjam')
->setCellValue('D1', 'Tanggal Kembali')
->setCellValue('E1', 'Jumlah')
->setCellValue('F1', 'Status');
    $column=2;
    foreach($data as $data){
         $spreadsheet->setActiveSheetIndex(0)
         ->setCellValue('A'. $column, $data->nama)
        ->setCellValue('B'. $column, $data->nama_b)
        ->setCellValue('C'. $column, $data->tgl_pinjam)
        ->setCellValue('D'. $column, $data->tgl_kembali)
         ->setCellValue('E'. $column, $data->jumlah)
         ->setCellValue('F'. $column, $data->status);
        $column++;
    $writer = new XLsx($spreadsheet);
    $fileName = 'Data Laporan Buku Dipinjam';
    \begin{tabular}{ll} header (\begin{tabular}{ll} Content-type: vnd. openxml for mats-office document. spread sheetml. sheet'); \\ \end{tabular}
    header('Content-Disposition:attachment;filename='.$fileName.'.xls');
    header('Cache-Control: max-age=0');
    $writer->save('php://output');
    return redirect()->to('/login');
```

Kode ini merupakan bagian dari suatu aplikasi yang menghasilkan laporan peminjaman buku. Berikut adalah penjelasan singkat:

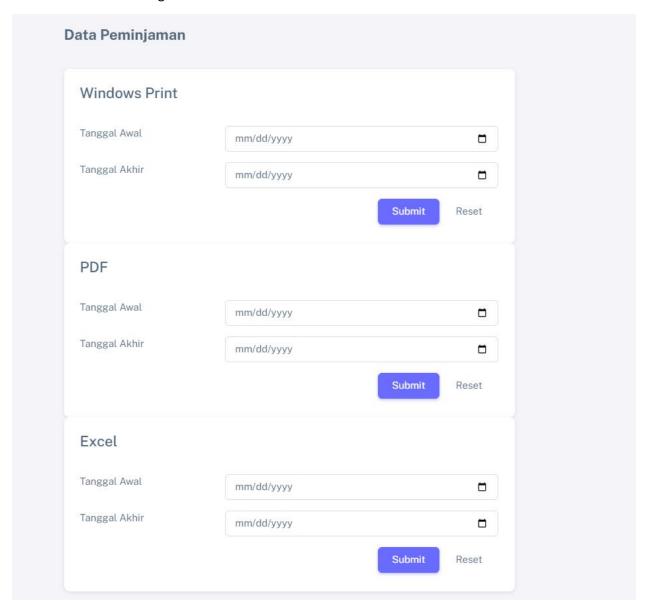
index: Menampilkan halaman untuk mengatur filter laporan peminjaman buku. Hanya dapat diakses oleh pengguna dengan level 1 atau 2.

print_in: Menangani filter dan menampilkan hasil laporan peminjaman buku dalam bentuk tabel HTML. Hanya dapat diakses oleh pengguna dengan level 1 atau 2.

pdf_in: Menangani filter dan menampilkan hasil laporan peminjaman buku dalam format PDF menggunakan Dompdf. Hanya dapat diakses oleh pengguna dengan level 1 atau 2.

excel_in: Menangani filter dan menampilkan hasil laporan peminjaman buku dalam format Excel menggunakan PhpSpreadsheet. Hanya dapat diakses oleh pengguna dengan level 1 atau 2.

Semua fungsi memeriksa apakah pengguna sudah login dan memiliki tingkat akses yang sesuai sebelum melanjutkan operasi. Jika belum login atau memiliki tingkat akses yang tidak sesuai, pengguna akan diarahkan ke halaman login.



```
public function index()
       if(session()->get('level')==1 || session()->get('level')==2){
           $model=new M model();
           $data['title']='Data Pengembalian';
           $kui['kunci']='print_in';
           echo view('partial/header',$data);
           echo view('partial/menuutama');
           echo view('filter_pengembalian',$kui);
           echo view('partial/footer');
           return redirect()->to('/login');
   public function print_in()
       if(session()->get('level')==1 || session()->get('level')==2 ){
   $model = new M_model();
   $awal = $this->request->getPost('awal');
   $akhir = $this->request->getPost('akhir');
   $status = "Dikembalikan";
   $kui['duar'] = $model->filter222('peminjaman', $awal, $akhir, $status);
   echo view('v_pm', $kui);
}else{
   return redirect()->to('/login');
   public function pdf_in()
       if(session()->get('level')==1 || session()->get('level')==2 ){
           $model = new M_model();
           $awal = $this->request->getPost('awal');
           $akhir = $this->request->getPost('akhir');
           $status = "Dikembalikan";
           $kui['duar'] = $model->filter222('peminjaman', $awal, $akhir, $status);
       $dompdf = new\Dompdf\Dompdf();
       $dompdf->loadHtml(view('v_pm',$kui));
       $dompdf->setPaper('A4','potrait');
       $dompdf->render();
       $dompdf->stream('my.pdf', array('Attachment'=>0));
       return redirect()->to('/login');
```

Kode ini adalah bagian dari suatu aplikasi yang menghasilkan laporan pengembalian buku. Berikut adalah penjelasan singkat:

index: Menampilkan halaman untuk mengatur filter laporan pengembalian buku. Hanya dapat diakses oleh pengguna dengan level 1 atau 2.

print_in: Menangani filter dan menampilkan hasil laporan pengembalian buku dalam bentuk tabel HTML. Hanya dapat diakses oleh pengguna dengan level 1 atau 2.

pdf_in: Menangani filter dan menampilkan hasil laporan pengembalian buku dalam format PDF menggunakan Dompdf. Hanya dapat diakses oleh pengguna dengan level 1 atau 2.

excel_in: Menangani filter dan menampilkan hasil laporan pengembalian buku dalam format Excel menggunakan PhpSpreadsheet. Hanya dapat diakses oleh pengguna dengan level 1 atau 2.

Semua fungsi memeriksa apakah pengguna sudah login dan memiliki tingkat akses yang sesuai sebelum melanjutkan operasi. Jika belum login atau memiliki tingkat akses yang tidak sesuai, pengguna akan diarahkan ke halaman login.

