

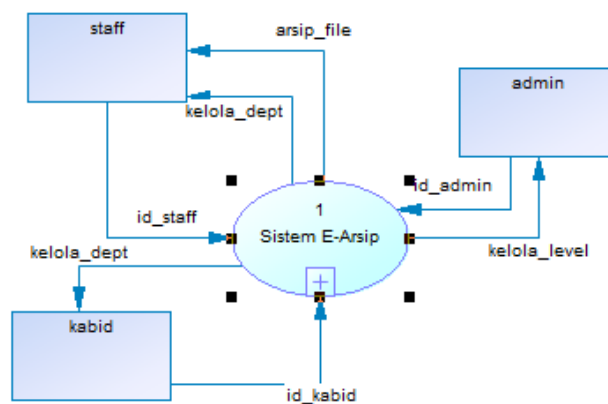
Sistem Elektronik Arsip Berbasis Web

1. Deskripsi Sistem

Sistem Elektronik Arsip berbasis web merupakan aplikasi untuk menyimpan arsip file dalam administrasi sebuah instansi/organisasi. Aplikasi ini bisa diterapkan di Organisasi dan Instansi. Sistem dapat mengelola pengarsipan file dan menyimpan file di website dengan cepat, aman dan nyaman sehingga memudahkan dalam *filtering* maupun *searching* ketika file tersebut akan digunakan kembali oleh pengguna.

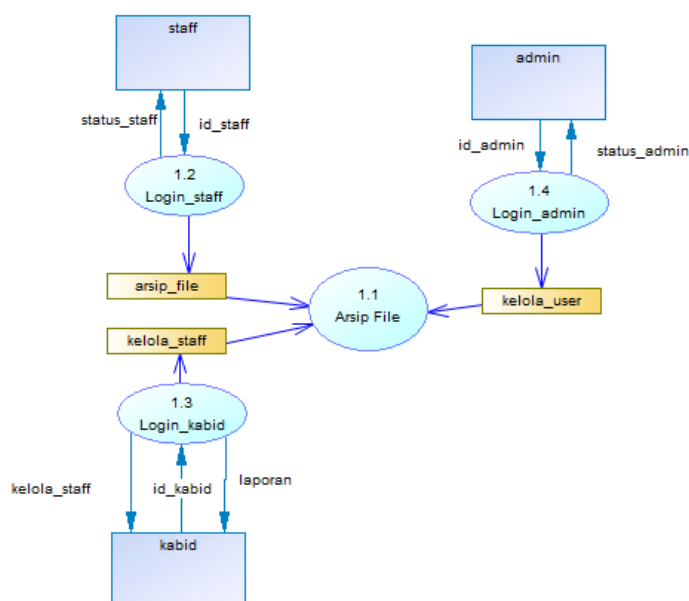
2. Perancangan Prosedur

2.1 Diagram Konteks



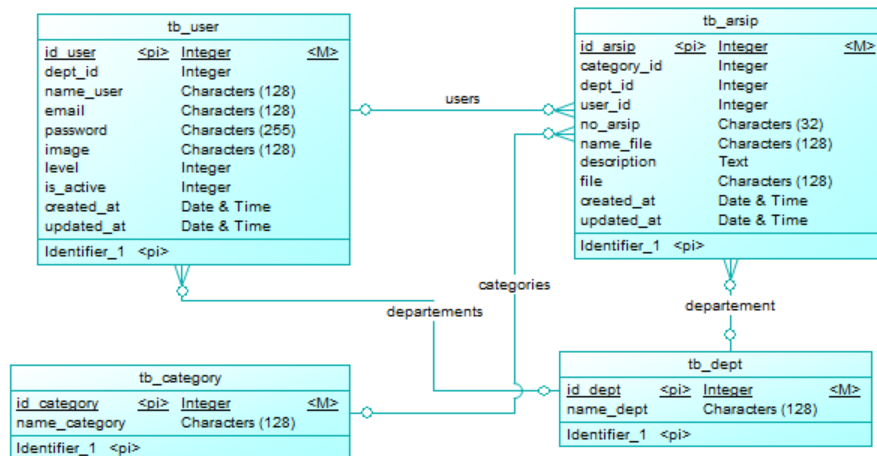
Gambar 1 Diagram Konteks E-arsip.

2.2 Data Flow Diagram



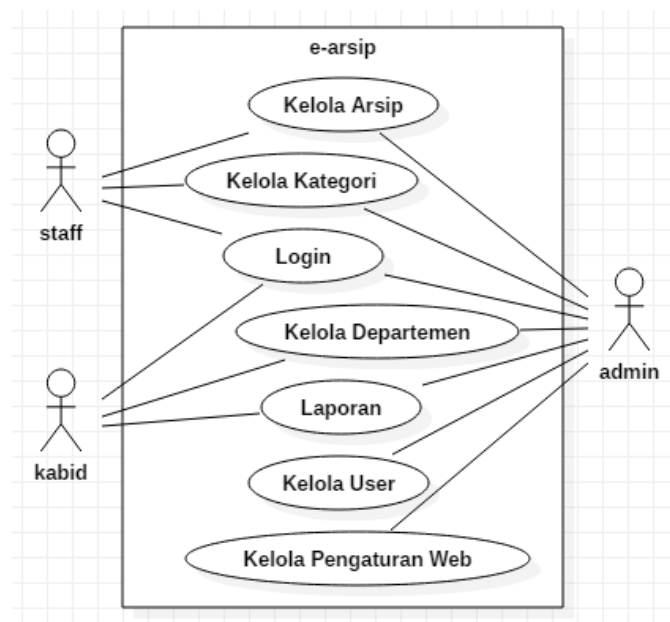
Gambar 2 DFD level 0

2.3 Conceptual Data Model



Gambar 3 CDM E-arsip

2.4 Use Case Diagram



Use Cas Diagram E-arsip

a. Deskripsi Aktor

Pada Use case diagram sistem e-arsip ini terdapat 3 aktor yaitu:

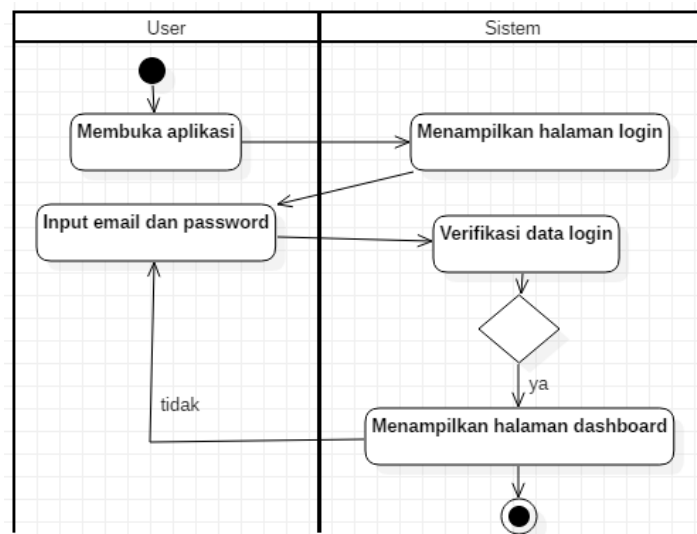
No	Aktor	Deskripsi
1	Staff	Pihak yang melakukan pengarsipan file
2	Kabid	Pihak yang mendapat laporan dari staff
3	Admin	Pihak yang mengelola data dan level user pada sistem

b. Deskripsi Use Case

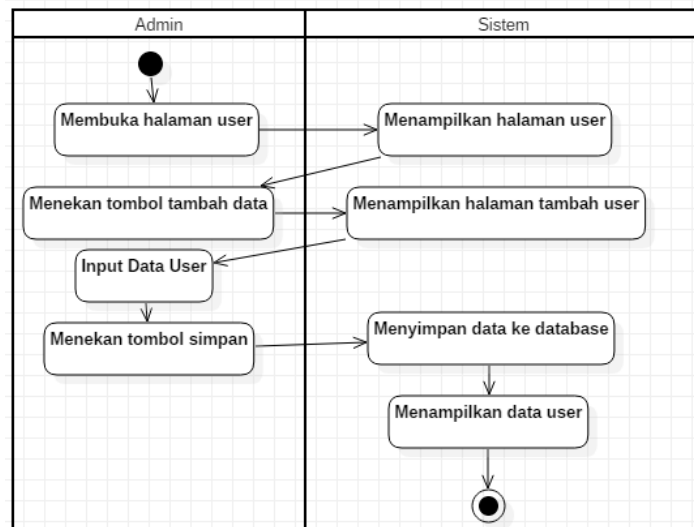
Pada Use case diagram sistem e-arsip ini terdapat beberapa use case yaitu:

No	Use case	Deskripsi
1	Kelola Arsip	Proses mengelola file arsip ke sistem
2	Kelola Kategori	Proses mengelola kategori untuk file arsip
3	Login	Proses masuk aplikasi untuk tiap user
4	Kelola Departemen	Proses mengelola departemen dan staff
5	Laporan	Proses mendapatkan laporan dari data pengarsipan
6	Kelola User	Proses mengelola pengguna sistem
7	Kelola Pengaturan Web	Proses mengelola pengaturan sistem

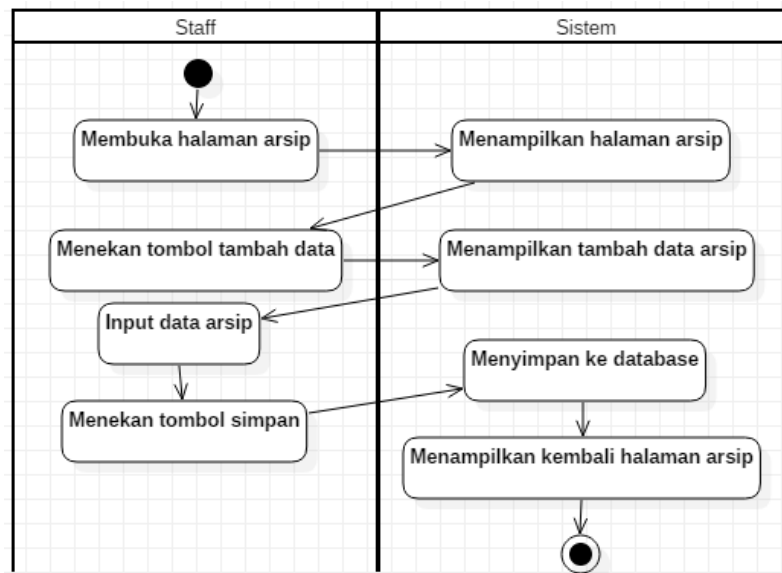
2.5 Activity Diagram



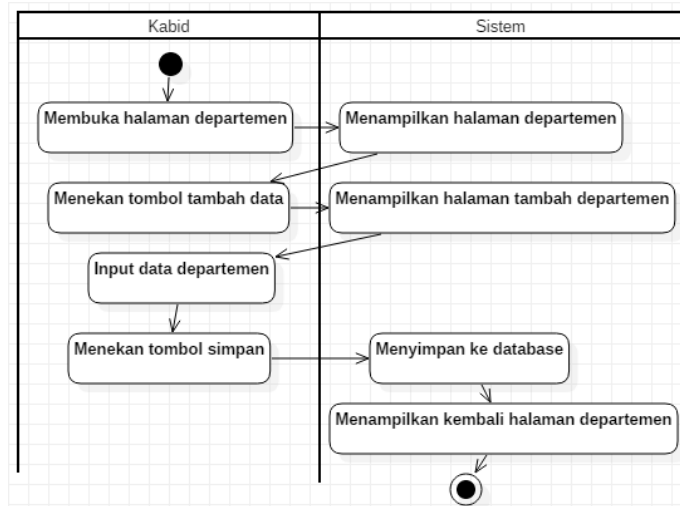
Activity Diagram Login



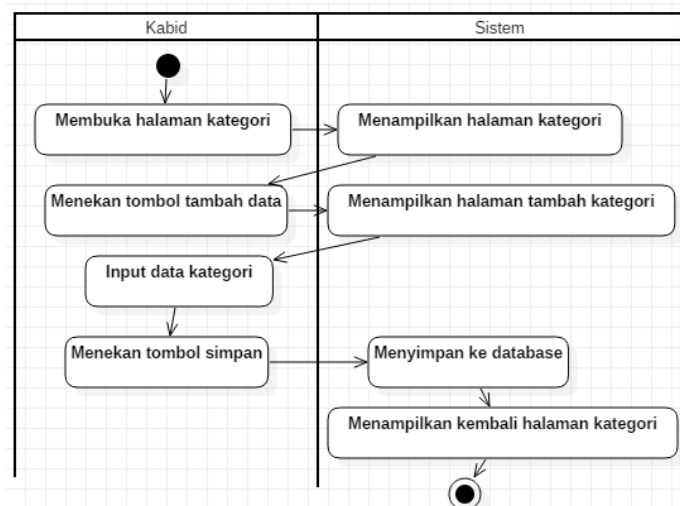
Activity Diagram Halaman User



Activity Diagram Halaman Arsip



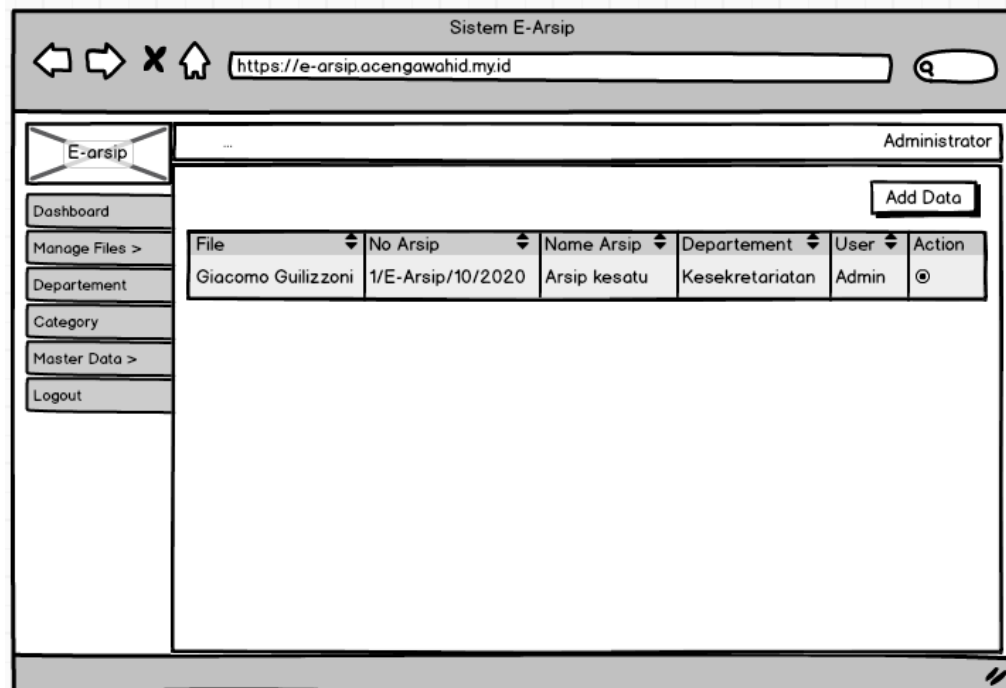
Activity Diagram Departement



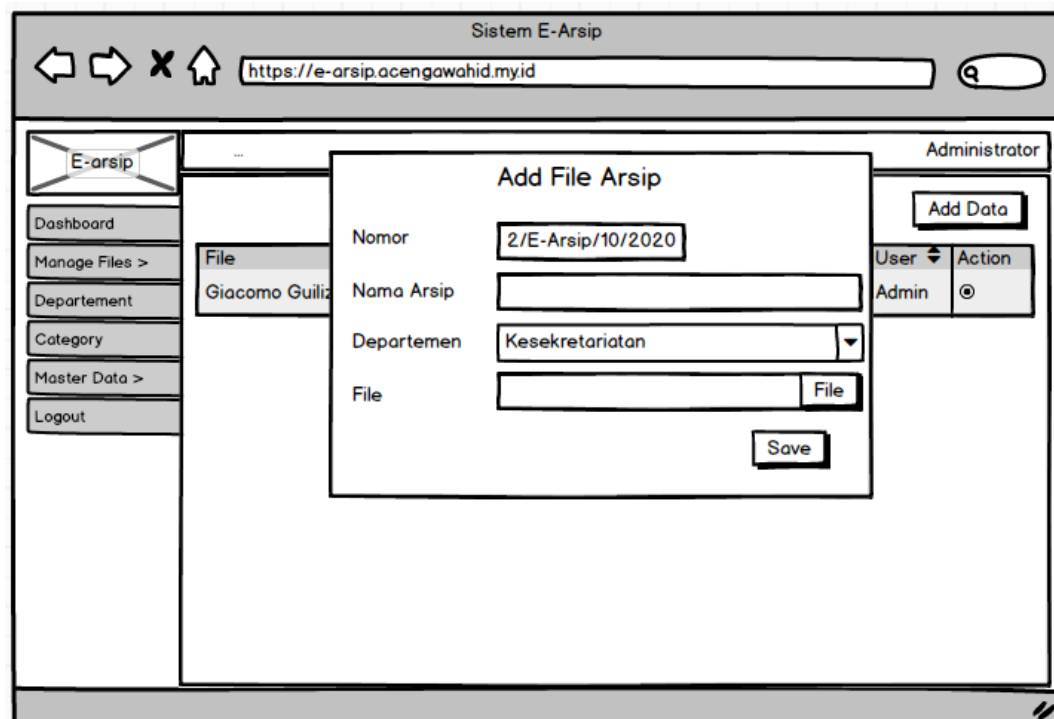
Activity Diagram Kategori

2.6 Mockup Aplikasi

Merupakan gambaran dari tampilan dalam aplikasi



Mockup halaman Arsip



Mockup tambah Arsip

2.7 Kode Program

Controller Arsip

```
<?php

namespace App\Controllers\Admin;

use App\Controllers\BaseController;
use App\Models\ArsipModel;
use App\Models\CategoryModel;
use App\Models\DepartementModel;

class Arsip extends BaseController
{
    protected $ArsipModel;
    public function __construct()
    {
        $this->ArsipModel = new ArsipModel();
        $this->CategoryModel = new CategoryModel();
        $this->DepartementModel = new DepartementModel();
    }
    public function index()
    {
        $data = [
            'title' => 'Arsip Data',
            'setting' => $this->SettingModel->getData(),
            'arsip' => $this->ArsipModel->getAllData(),
            'category' => $this->CategoryModel->getAllData(),
            'departement' => $this->DepartementModel->getAllData(),
            'count_arsip' => $this->ArsipModel->getCount(),
            'validation' => \Config\Services::validation()
        ];
        return view('admin/arsip', $data);
    }

    public function create()
    {
        if ($this->validate([
            'id_category' => [
                'label' => 'Category',
                'rules' => 'required',
            ],
            'id_dept' => [
                'label' => 'Departement',
```

```

        'rules' => 'required',
    ],
    'no_arsip' => [
        'label' => 'Number',
        'rules' => 'required',
    ],
    'name_file' => [
        'label' => 'Name File',
        'rules' => 'required',
    ],
    'file' => [
        'label' => 'File',
        'rules' => 'mime_in[file,application/pdf,application/doc,application/docx]',
    ],
D)) {
    // Jika valid
    $fileDoc = $this->request->getFile('file');
    $nameDoc = $fileDoc->getName();
    $data = [
        'id_category' => $this->request->getPost('id_category'),
        'id_dept' => $this->request->getPost('id_dept'),
        'id_user' => session()->get('id_user'),
        'no_arsip' => $this->request->getPost('no_arsip'),
        'name_file' => $this->request->getPost('name_file'),
        'description' => $this->request->getPost('description'),
        'file' => $nameDoc,
    ];
    $fileDoc->move('assets/file/arsip', $nameDoc);
    $this->ArsipModel->insert($data);
    session()->setFlashdata('message', 'Data has been successfully');
    return redirect()->to('/admin/arsip');
} else {
    // Jika tidak valid
    session()->setFlashdata('error', \Config\Services::validation()->listErrors());
    return redirect()->to('/admin/arsip');
}
}

public function update($id)
{
    if ($this->validate([
        'id_category' => [
            'label' => 'Category',

```



```

        'rules' => 'required',
    ],
    'id_dept' => [
        'label' => 'Departement',
        'rules' => 'required',
    ],
    'no_arsip' => [
        'label' => 'Number',
        'rules' => 'required',
    ],
    'name_file' => [
        'label' => 'Name File',
        'rules' => 'required',
    ],
    'file' => [
        'label' => 'File',
        'rules' => 'mime_in[file,application/pdf,application/doc,application/docx]',
    ],
)) {
    // Jika valid
    $fileDoc = $this->request->getFile('file');
    if ($fileDoc->getError('file') == 4) {
        $nameDoc = $this->request->getFile('oldFile');
    } else {
        // Hapus foto
        $arsip = $this->ArsipModel->getAllData($id);
        if ($arsip['file'] != "") {
            unlink('assets/file/arsip/' . $arsip['file']);
        }
        $nameDoc = $fileDoc->getName();
        $fileDoc->move('assets/file/arsip', $nameDoc);
    }
    $data = [
        'id_arsip' => $id,
        'id_category' => $this->request->getPost('id_category'),
        'id_dept' => session()->get('id_dept'),
        'id_user' => session()->get('id_user'),
        'no_arsip' => $this->request->getPost('no_arsip'),
        'name_file' => $this->request->getPost('name_file'),
        'description' => $this->request->getPost('description'),
        'file' => $nameDoc,
    ];
    $this->ArsipModel->updateData($data);
}

```

```

        session()->setFlashdata('message', 'Data has been updated');
        return redirect()->to('/admin/arsip');
    } else {
        // Jika tidak valid
        session()->setFlashdata('error', \Config\Services::validation()->listErrors());
        return redirect()->to('/admin/arsip');
    }
}

public function delete($id)
{
    $arsip = $this->ArsipModel->getAllData($id);
    unlink('assets/file/arsip/' . $arsip['file']);

    $this->ArsipModel->deleteData($id);
    session()->setFlashdata('message', 'Data has been deleted. ');
    return redirect()->to('/admin/arsip');
}

public function excel()
{
    $data = [
        'arsip' => $this->ArsipModel->getAllData(),
    ];
    return view('report/excel_arsip', $data);
}
}

```

Model Arsip

```

<?php

namespace App\Models;

use CodeIgniter\Model;

class ArsipModel extends Model
{
    protected $table = 'tb_arsip';
    protected $primaryKey = 'id_arsip';
    protected $useTimestamps = TRUE;
    protected $allowedFields = ['id_category', 'id_dept', 'id_user', 'no_arsip', 'name_file',
    'description', 'file'];
}

```

```
public function __construct()
{
    $this->db = db_connect();
    $this->builder = $this->db->table($this->table);
}

public function getAllData($id = null)
{
    if ($id == null) {
        return $this->builder->join('tb_category', 'tb_category.id_category =
tb_arsip.id_category', 'left')->join('tb_user', 'tb_user.id_user = tb_arsip.id_user', 'left')-
>join('tb_dept', 'tb_dept.id_dept = tb_arsip.id_dept', 'left')->get()->getResultArray();
    } else {
        $this->builder->where('id_arsip', $id);
        return $this->builder->get()->getRowArray();
    }
}

public function updateData($data)
{
    return $this->builder->update($data, ['id_arsip' => $data['id_arsip']]);
}

public function deleteData($id)
{
    return $this->builder->delete(['id_arsip' => $id]);
}

public function getCount()
{
    return $this->builder->countAll();
}
}
```