A web based e-archives information system design in Universitas PGRI Yogyakarta

S Oyama¹, A Wahana² and R Widagsa³
¹,²,³Universitas PGRI Yogyakarta, Indonesia
Email: oyama@upy.ac.id, aditya@upy.ac.id, widagsa@upy.ac.id

ABSTRACT. Overloaded space and the inefficiency of document finding are the weakness of the academic archive system in Universitas PGRI Yogyakarta (UPY). Drawing upon the background, this study attempts to design an academic e-archive information system in UPY. This study aims to generate a digital archive information system using Microsoft Solution Framework (MSF) with Oriented Development (OOD) approach assisted by UML (Unified Modeling Language) tools. The Web-based e-archive designed at UPY provides convenience and security in terms of academic document storage as well as the efficiency of data searching.

Keywords: e-archive, MSF, UML, OOD

1. Introduction
Archive management has a crucial role in the sustainability of an institution. Optimal archive management in an agency will be very helpful in determining an institutional policy and make a major contribution to the development of the institution [1]. Archive management properly and correctly allows employees to effectively process storage and retrieval of a document. In line with the statement, the Regulation of the Minister of Research, Technology and Higher Education Number 59 of 2018 requires every tertiary institution to prepare itself by making improvements, especially regarding student data at Directorate General of Higher Education Database (PDDIKTI).

Academic documents such as certificates, academic transcripts, and diploma supplements (SKPI) issued by higher education institutions must be necessarily archived in digital form. It is mandatory for every higher education institution. Therefore, Academic Administration Bureau of Universitas PGRI Yogyakarta (BAAk UPY) initiates the transformation form manual archiving to digital archiving. Currently, BAAk UPY uses manual archiving by storing the printed academic documents in a special storage room. Based on the retrieved data from BAAk, 13 data or 48.15% were recorded in the form of soft file (semi digital) and 14 or 51.85% were in the form on hard file (printed) as shown in Table 1.

| Table 1. Academic Administration Bureau Data |
|----------------------------------------------|
| No   | File Name                      | Type of Document |
|------|--------------------------------|------------------|
| 1    | Letter of course withdrawal   | soft file        |
| 2    | Letter of Course Transfer     | soft file        |
| 3    | Surat Cuti                    | soft file        |
|      | (Letter of Leave)             |                  |
| 4    | Letter of Student Activation  | soft file        |
| 5    | Letter of Graduation          | soft file        |
The ineffectiveness of the manual method includes limited space in terms of providing a cupboard or cabinet filling, the need for funds for photocopying, and purchasing expensive equipment. Along with the development of information systems, archiving is designed using computers where this system does not require a large area and does not require expensive costs. As a consequence of the old storage method, several files in the form of photocopies of certificates were damaged due to poor storage. Another disadvantage of the old method of data storage is the difficulty in finding data when the data is needed, especially the old one.

Based on the above constraints, this study is intended to create an information system that is able to overcome the problems. Additionally, this study is in line with the Regulation of the Ministry of Research, Technology and Higher Education no 59 [2]. The information system was developed to facilitate the academic staffs to access the archives needed in Universitas PGRI Yogyakarta.

This system was developed on a web base where the server was placed locally in BAAk. The design of this information system is adjusted to the needs of each section based on multi-user. However, the rights to access the system remains limited.

2. Method.
The research method of this study used the Microsoft Solution Framework (MSF). MSF is an application design and development method introduced by Microsoft. The system approach method used is object oriented development (OOD), namely the development of software / applications based on objects that exist in the real world. Generally, this method uses a tool, namely UML (Unified Modeling language). The data analysis method is the most decisive step in this study, the data analysis used includes the following stages:
2.1. Data Collecting Technique

The data collection method in this study includes:

- **Observation.** The method was carried out by direct observation of the object under study. The object is a digital archive that will be archived.
- **Interview.** This method was carried out by direct interviews with the parties involved in this study. Interviews were conducted with archive users in an academic setting.
- **Questionnaire.** This method was done by providing a questionnaire to system users to determine the quality level of the system that has been created.

2.2. Instrument for Data Collecting

2.2.1. Digital Data Checklist. The checklist in this study is used to measure the readiness of the system being developed in Table 2

| No | Aspek       |
|----|-------------|
| 1  | Type File   |
| 2  | Form File   |
| 3  | Zise file   |

2.2.2. Alpha Testing. This checklist includes several aspects and is used to determine the quality of the system that has been developed.

| No | Aspects | Items |
|----|---------|-------|
| 1  | View    | 2     |
3. Result and Discussion

3.1. System Requirements Design

This research was conducted from April to June 2020 by compiling questions or the needs to develop an E-Archive system. The results of the system requirements being developed are in the following details:

- Admin menu
- User menu
- User manage menu
- Access rights menu
- Upload data menu
- Data criteria menu

From the above requirements, a data flow diagram is made, here is a data flow diagram

![Data Flow Diagram](image)

**Figure 2. DAD Context**

The diagram above illustrates that the system consists of two entities, namely user and admin in which the user only has access to upload data, while in the admin menu, there are several access rights that can access all menus including creating a user, setting user access rights, viewing data, uploading data. From the context diagram above, the system was translated into a level 1 data flow diagram as follows:
The image above is a more detailed description of the process flow on the system to be developed where users can only add data to upload files. User settings are carried out by the admin, the admin here acts as a controller of all users and the admin can set any menus that can appear on the user page.

### 3.2. Interface Design

The next step is to make a system design, namely by making a system interface design; here is the system interface design:

**3.2.1. Login Page Design.** This page is used as the initial door or gateway to enter the E-Archive system.

This page is designed to appear the first time a user accesses the system, the user must type a username and password. If the username and password are correct, the user will enter the system, if they are incorrect, a username or password error will appear.
3.2.2. **Admin start page design**

![Admin Page](image)

**Figure 5. Admin Page**

3.2.3. **Setting Page Design**

![Setting Page](image)

**Figure 6. Setting Page**

3.2.4. **Access Rights Setting Page Design**

![Access Rights Setting](image)

**Figure 7. Access Rights Setting**
The design of the interface on the access rights settings is used by the admin to set the menu for users and the admin has full access rights to the system. Meanwhile, user permissions can only view and upload files.

### 3.2.5. User Page Design

![User Page Design](image)

#### Figure 8. User Page

### 3.3. Database Design

#### 3.3.1. Table Design

Table design is the process to create databases and tables that is used in the UPY E-Archive system. The tables will be used to store data and upload files. The following tables are the names of the tables used in the system creation.

#### Table 4. Table app_role

| Field      | Type     | Size | Atribut            |
|------------|----------|------|--------------------|
| Role_id    | Integer  | 10   | Primary key        |
| Role_code  | Varchar  | 25   |                    |
| Role_desc  | Varchar  | 50   |                    |

#### Table 5. Table edok_kategori_meta

| Field                  | Type     | Size       | Atribut        |
|------------------------|----------|------------|----------------|
| Kategorimeta_id        | integer  | 11         | Primary key    |
| Kategorimeta_kode      | varchar  | 50         |                |
| Kategorimeta_label     | varchar  | 255        |                |
| Kategorimeta_deskripsi| Varchar  | 255        |                |

#### Table 6. Table app_config

| Field             | Type     | Size       | Atribut        |
|-------------------|----------|------------|----------------|
| Appconfig_id      | integer  | 11         | Primary key    |
| Appconfig_name    | varchar  | 255        |                |
| Appconfig_desc    | varchar  | 255        |                |
| Appconfig_key     | Varchar  | 255        |                |
| Appconfig_value   | Varchar  | 400        |                |
| Appconfig_type    | Enum     |            |                |

#### Table 7. Table edok_berkas

| Field       | Type   | Size   | Atribut        |
|-------------|--------|--------|----------------|
| Berkas_id   | integer| 11     | Primary key    |
| Profil_id   | Integer| 11     |                |
| Kategori_id | Integer| 11     |                |
Table 8. Table app_roleprivilege

| Field            | Type    | Size | Atribut     |
|------------------|---------|------|-------------|
| Roleprivilege_id | integer | 11   | Primary key |
| Role_id          | Integer | 11   |             |
| Menu_id          | Integer | 11   |             |
| Roleprivilege_isshow | Enum   |      | T,f         |

Table 9. Table edok_profil

| Field            | Type    | Size | Atribut     |
|------------------|---------|------|-------------|
| Profil_id        | integer | 11   | Primary key |
| Profil_identitas | Varchar | 100  |             |
| Profil_nama      | Varchar | 255  |             |
| Profil_email     | Varchar | 100  |             |
| Profil_phone     | Varchar | 20   |             |
| Profil_flag      | Varchar | 20   |             |

Table 10. Table app_pengguna

| Field            | Type    | Size | Atribut     |
|------------------|---------|------|-------------|
| Pengguna_id      | integer | 11   | Primary key |
| Profil_id        | Integer | 11   |             |
| Role_id          | Integer | 10   |             |
| Pengguna_username | Varchar | 255  |             |
| Pengguna_password | Varchar | 255  |             |
| Pengguna_email   | Varchar | 255  |             |
| Pengguna_password_ | Varchar | 255  |             |
| Pengguna_status  | Enum    |      | F,T         |
| Pengguna_created | Datetime |      |             |
| Pengguna_modified | Datetime |      |             |
| Pengguna_lastlogin | Datetime |      |             |

Table 11. Table edok_kategori

| Field            | Type    | Size | Atribut     |
|------------------|---------|------|-------------|
| Kategori_id      | integer | 11   | Primary key |
| Kategori_kode    | Varchar | 50   |             |
| Kategori_nama    | Varchar | 255  |             |
| Kategori_syscreate | Datetime |      |             |
| Kategori_sysupdate | Datetime |      |             |
| Pengguna_idlast  | Integer | 10   |             |

Table 12. Table app_menu

| Field            | Type    | Size | Atribut     |
|------------------|---------|------|-------------|
| Menu_id          | Integer | 11   | Primary key |
| Menu_urutan     | Integer | 11   |             |
3.3.2 Relation Table

| Field                | Type      | Size  | Attribut        |
|----------------------|-----------|-------|-----------------|
| Berkasdata_id        | integer   | 11    | Primary key     |
| Berkasdata_kode      | Varchar   | 255   |                 |
| Berkasdata_label     | Varchar   | 255   |                 |
| Berkasdata_isi       | Text      |       |                 |
| Berkasdata_syscreate | Datetime  |       |                 |
| Berkasdata_sysupdate | Datetime  |       |                 |
| Pengguna_idlast      | Integer   | 11    |                 |

![Figure 9. Relation Table]
Table relations are relationships that occur in a table with the other tables that represent object relationships in the real world and function to regulate the operation of a database. The relationship aims to connect Table 1 with other tables, therefore, the table no longer stands alone and can be connected to one another so that it becomes a unit. The relationship between tables is also called normalization so that the table looks easy to navigate. The purpose of data relations is to emphasize data independence and to overcome inconsistencies and duplication of data.

3.4. System Implementation

System implementation is the stage where the system was created and transformed into a real application based on the results of the previous design so that the system is ready to be operated. Thus, it can be seen that the system that has been created can be used in accordance with the objectives of the design.

3.4.1. Login Page Implementation. Login means giving access rights to the following menus; therefore, each user has limitations in accessing the menu. The following figure is the implementation of the login page figure:

![Login Page Image](image)

**Figure 10. Login Page**

3.4.2. Admin Page Implementation. The admin page was displayed for users who logged in as administrators, the page has several menus, including: settings menu, access menu, user roles menu, file menu, category menu, personal and archive. The following figure is the implementation of the admin page.
3.4.3. **Menu Setting Page Implementation.** This page is used by the administrator to manage what menus given to users who use the UPY E-Archive system. Some of the access menus include user permissions, file categories, personal and manage. The following figure is the display of the access settings page.

![Figure 11. Admin Page](image)

**Figure 11. Admin Page**

3.4.4. **Access Rights Page Setting Implementation.** This page is used by administrators to control user access to the system. Admin menu and user menu have different views; here is a view of access rights for administrators.

![Figure 12. Menu Page](image)

**Figure 12. Menu Page**
3.4.5. **User Page Implementation.** This page is used by administrators to add new users and delete users who are no longer active. Here is the implementation of the user page.

![User Page](image1.png)

**Figure 14. User Page**

3.4.6. **Category Page Implementation.** This page is used by administrators to input categories of files to be stored in the UPY E-Archive, including: certificate of graduation, academic transcripts, SKPI, graduation letters. Here is the result of the category page implementation.

![Data Pengguna](image2.png)
3.4.7. **Profile Page Implementation.** This page can be accessed by administrators and users, this page aims to add students who have graduated and also letters that are uploaded. The following is the result of implementing the profile page.

3.4.8. **File Page Implementation.** This page is used by the user to upload data according to the file name, as students can have more than one file. Here is the implementation of the file page.
3.5. **System Device Specifications**

In order for the application to run properly, supporting devices are needed, one of the devices is a computer. Computers are devices that have components consisting of hardware and software. The two components cannot be separated from one another. The following are the specifications to make the application run properly.

3.5.1. **Hardware.** The minimum hardware specifications needed for this e-archive application are as follows:

- Intel Pentium 4 2.8 GHZ or above
- Memory (RAM) 512 MB
- Free disk space capacity is better above 2 GB
- Color monitor at least 14 inch
- Memory (VGA) at least 64 MB
- Keybord
- Mouse
- Speakers

3.5.2. **Software.** The minimum specifications of the software required in the process of building and operation of the application system as follows:

- OS : Above Microsoft windows XP
- Web Server : Apache
- Database Server : MySQL
- Programming language : PHP and ActionScript 2.0
4. Conclusion
The E-archive information system in Universitas PGRI Yogyakarta was built on a web-based, which is a media that can be used and can convey information online. Therefore, it can be concluded that:
- E-archive information system simplifies the files storage without requiring a large space.
- By implementing the E-archive information system, the stored files are safer since they are stored in digital form.
- This system facilitates the data searching in finding archives.

5. References
[1] Abubakar, hadi (1997). Cara-cara Pengolahan Kearsipan yang Praktis dan Efisien. Jakarta:Djambatan
[2] BAN PT. (2017). Peraturan BAN PT No 2 tahun 2017 tentang SAN-SIKTI. Jakarta: BAN-PT.
[3] Kementrian Dalam negeri. (2018, 5 4). Kementrian Dalam negeri. Dipetik 12 30, 2018, dari Kementrian Dalam negeri: https://www.kemendagri.go.id/
[4] Kermenristekdikti. (2012). UU No 12 Tahun 2012 Tentang Pendidikan Tinggi. Jakarta: Republik Indonesia.
[5] Amsyah, Zulkifli.(1996). Manajemen Kearsipan. Jakarta: Gramedia Pustaka Utama
[6] Laudon, C. Kenneth. (2008). Management Information System: Managing The Digital Firm. Pearson Prentice Hill.