DESIGN OF STAFF PAYROLL SYSTEM USING DATABASE AT XYZ UNIVERSITY

Fahmi Adhila Fadhila Ruspendi
Polytechnic Piksi Ganesha
Bandung, Indonesia

Yuda Syahidin
Polytechnic Piksi Ganesha
Bandung, Indonesia

Abstract. The development of the staff payroll system aims to streamline and enhance the accuracy and efficiency of payroll and attendance data management for administrative staff. This system is designed to help administrators quickly and reliably compile and process attendance and payroll information. A robust database is essential to support the implementation of the payroll system at XYZ University. The database design process is carried out in four key stages: Requirement Collection and Analysis, Conceptual Database Design, Logical Database Design, and Physical Database Design. Once the design is complete, the database undergoes rigorous testing, including access policy validation, anomaly detection, and view checks, to ensure its reliability and functionality.

Keywords: *Attendance Data, Payroll Data Management, Logical Database Design*

1.1 Background of the Study

XYZ University, a private institution, comprises several faculties, each employing its own unique payroll system for lecturers, assistant lecturers, and administrative staff. The Informatics Study Program, part of the Faculty of Informatics, currently relies on a manual payroll process, which has proven to be inefficient and prone to errors.

This manual system primarily uses Microsoft Excel for data management, requiring the input of various data points such as attendance records, courses taught, and daily class schedules, particularly for lecturers and assistant lecturers. This approach increases the risk of data loss and calculation errors, ultimately compromising the accuracy and reliability of the payroll system. As a result, there is a pressing need to develop a more effective and efficient payroll system to address these issues.

To address these shortcomings, a web-based staff payroll information system will be developed to streamline the recapitulation of payroll data quickly and accurately. This study focuses on designing and implementing a database to support the system, enabling administrators to efficiently manage and utilize data. The database will store comprehensive information, including details about lecturers, instructors, teaching assistants, courses taught, classes, and attendance records.

The proposed system is expected to assist all relevant stakeholders in managing the monthly payroll process, ensuring that salary calculations and data recapitulation are performed efficiently and accurately, significantly reducing processing time and minimizing errors.

1.2 Problem Formulation

Based on the background outlined, the core problem can be formulated as follows: *How can staff payroll data be managed accurately and efficiently?*

1.3 Research Objectives

The objective of this study is to design and implement a database system to support the payroll process within the Informatics Study Program at XYZ University. This system aims to enhance data accuracy, streamline payroll management, and improve overall efficiency.

1.4 Problem Scope and Limitations

The scope of the study, titled *"Designing a Staff Payroll System for the Informatics Study Program at XYZ University,"* is defined as follows:

a. Development of a database system tailored specifically for the Informatics Study Program at XYZ University.

b. The database will manage data related to lecturers, assistant lecturers, and instructors.

1.5 Research Benefits

The outcomes of this research are expected to offer the following benefits:

a. Enhancing the accuracy of data management processes.

b. Accelerating the processing and management of payroll-related data.

c. Establishing standardized procedures and functionalities for the payroll system, ensuring consistency and efficiency.

1.6 Structure of the Report

The structure of this report is designed to provide clarity and facilitate the writing process of the final assignment. The organization is as follows:

CHAPTER I: INTRODUCTION

Chapter I outlines the general framework for the final assignment, including the background of the study, problem formulation, research objectives, research benefits, and the structure of the report.

CHAPTER II: LITERATURE REVIEW AND THEORETICAL FRAMEWORK

Chapter II presents the relevant theories and existing literature that serve as the foundation for the research. These theories provide the necessary context and support for the development of the thesis.

CHAPTER III: SYSTEM ANALYSIS AND DESIGN

Chapter III details the research methodology employed in the study. It covers the research tools and materials, the steps taken throughout the research process, the system design, and the test design, all of which are informed by relevant supporting theories and prior explanations.

CHAPTER IV: RESULTS AND DISCUSSION

Chapter IV presents the implementation of the developed system and discusses the results of the system testing. This chapter highlights the effectiveness and performance of the system in addressing the research objectives.

CHAPTER V: CONCLUSIONS AND RECOMMENDATIONS

Chapter V summarizes the key findings of the research, offering conclusions drawn from the results. Additionally, this chapter provides recommendations based on the study’s findings, suggesting areas for improvement or further exploration.

Basic Theory

2.1 Research Tools and Materials

This study requires several tools and materials to successfully design the Staff Payroll System database. These resources include both hardware and software components essential for the development and implementation process.

2.1.1 Hardware

The hardware used for designing the database consists of a laptop with the following specifications:

1. Intel Core i3 Processor

2. 4 GB RAM

3. 500 GB Hard Disk

4. Windows 10 Pro

2.1.2 Software

In addition to the hardware, the following software tools are utilized to support the design of the IT Staff Payroll System database:

1. XAMPP (version 3.2.2)

2. Microsoft Visio 2016

3. PHPMyAdmin (version 4.5.1)

4. MySQL

2.2 Database Design Method

This study employs a four-stage database design method, which includes the following steps:

1. Requirement Collection and Analysis

2. Conceptual Database Design

3. Logical Database Design

4. Physical Database Design

These stages provide a structured approach to designing a robust and efficient database system, ensuring that all aspects of the database are carefully planned and implemented.

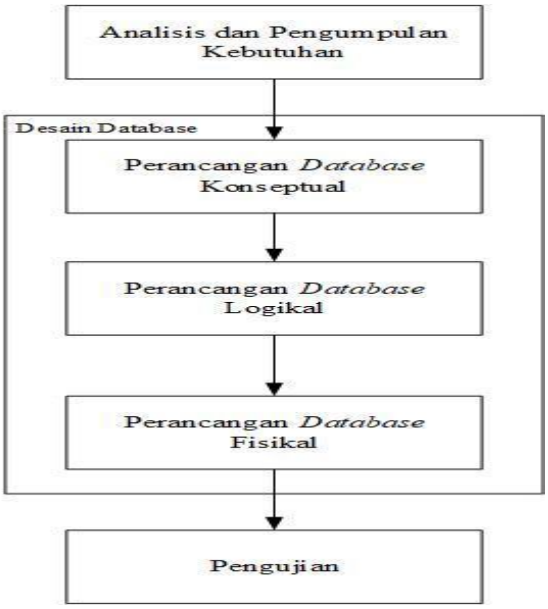

Analisis dan Pengumpulan Kebutuhan
Desain Database
Perancangan *Database* Konseptual
Perancangan *Database* Logikal
Perancangan *Database* Fisikal
Pengujian

Figure 2. 1 Database Design Stage

2.2.1 Requirement Analysis and Collection

The Requirement Analysis and Collection phase involves gathering and analyzing information to identify the system's user needs. This process includes reviewing existing documentation and conducting interviews with key users to gain insights into their requirements.

2.2.1.1 Interviews

In this study, interviews were conducted through direct question-and-answer sessions with relevant stakeholders. Three individuals involved in the payroll process were interviewed—two responsible for the attendance section and one for payroll calculation. The purpose of these interviews was to gather information about the existing attendance system and identify the features required for the new payroll website. This information is crucial for designing a system that meets user expectations and addresses current challenges.

2.2.1.2 Documentation

The documentation phase involves gathering data through observations and summarizing financial information from various sources. This includes reviewing relevant files, references, and observation results to compile all necessary information. The collected data serves as the foundation for creating the system to be developed, ensuring that all relevant details are incorporated into the design process.

2.2.2 Conceptual Database Design

Conceptual Database Design is the process of creating a model that reflects the information needs of the organization, without considering the physical aspects of implementation. This phase focuses on understanding and organizing the data structure at a high level. Key steps in conceptual database design include:

1. Identifying the entities within the system.

2. Defining the attributes for each entity.

3. Determining the primary key for each entity.

4. Establishing the relationships between different entities.

These steps help ensure that the database design accurately represents the data requirements of the organization, laying the groundwork for logical and physical design stages.

2.2.2.1 Normalization

The normalization stage involves analyzing and refining database tables to ensure they meet specific design standards. The primary goals of normalization are as follows:

a. Eliminate Data Redundancy: Reducing duplicate data to optimize storage and prevent unnecessary memory usage.

b. Reduce Complexity: Simplifying the database structure to make it easier to understand and manage.

c. Facilitate Data Modification: Streamlining data updates, deletions, and insertions to minimize errors and improve efficiency.

d. Prevent Anomalies: Avoiding data inconsistencies such as insertion, update, and deletion anomalies.

Normalization is a systematic, formal technique used in database design, employing a bottom-up approach to validate the relational structure. This process ensures that tables are well-structured, enhancing data integrity and storage efficiency. The goal is to correct poorly designed tables, resulting in a database that is both efficient and free from anomalies.

For a clearer understanding of the normalization process, refer to the following diagram:

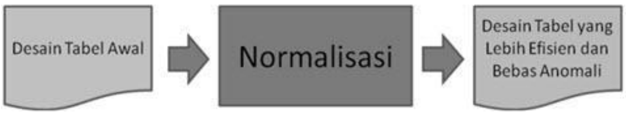


Figure 2. 2 Normalitation Process

2.2.3 Logical Database Design

Logical database design involves translating the conceptual design into a database model that can be implemented. This process maps the Entity Relationship Diagram (ERD) into a set of relational tables. The relational model is used to structure the data logically, defining tables, columns, and relationships between them to accurately represent the system's data requirements.

2.2.4 Physical Database Design

Physical database design focuses on implementing the database in a specific storage environment. This stage includes organizing files and indexes to optimize data access and storage efficiency. Additionally, it involves defining integrity constraints and setting up access controls to ensure data security and proper usage.

2.2.4.1 Access Policy

Access policy refers to the rules and restrictions implemented to control user access to the database. These policies ensure that users have the appropriate level of access based on their roles, protecting sensitive data and maintaining system integrity. Below are some access policies assigned to different user roles:

1. Admin Access Policy

Admins have comprehensive access to manage the system and perform the following tasks:

1. Log in, view, add, and modify data in the admin table.

2. View, add, modify, and delete staff data.

3. View, add, modify, and delete course data.

4. View and update staff status data.

5. View, add, modify, and delete class data.

6. View, add, modify, and delete attendance data.

7. View, add, modify, and delete payroll data.

2. Staff Access Policy

Staff have limited access focused on their personal data and payroll information:

1. Log in to the system.

2. View and update their password.

3. View their payroll data.

3. Master Access Policy

The Master has full administrative rights over the database and can perform all actions, including:

1. Viewing, adding, modifying, and deleting any data within the Staff Payroll System database.

This hierarchical access structure ensures secure and efficient data management while maintaining appropriate levels of access for different user roles.

2.2.5 Testing

The testing methods used in developing this database are anomaly testing and view check. Anomaly testing and view check are done through the admin page. Anomaly testing functions to find out whether the database process has unexpected side effects. For example, it causes data inconsistency or makes data disappear when data is deleted and view check functions to validate data.

Results and Discussion

3.1 Analysis and Requirement Gathering

Based on an interview with Mrs. Aprilia Kurnianti, who is directly responsible for payroll calculations, the essential data includes information on lecturers, instructors, and assistant lecturers. This data comprises NIP (Employee Identification Number), name, address, phone number, email, and password. Additionally, attendance records are required, detailing the teacher’s name, courses taught, and session meetings. Payroll data will be calculated based on the number of attendance records.

Similarly, an interview with Mr. Andhy Kurniawan, who oversees the attendance process, revealed comparable requirements. He emphasized the need for data on lecturers, instructors, and assistants, along with attendance records, which are capped at a maximum of 24 sessions per course. Course data and payroll information were also highlighted as critical.

From these interviews, it can be concluded that the key data required for developing the payroll database includes:

1. Lecturer, instructor, and assistant lecturer data.

2. Course data.

3. Attendance records.

4. Payroll data.

3.2 Conceptual Database Design

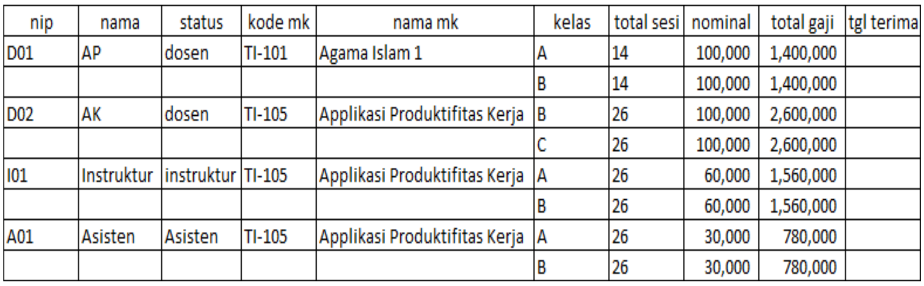
The conceptual database design process begins with normalization, aimed at ensuring the database structure is efficient and free from anomalies. In this study, the salary data table is the primary focus for normalization.

3.2.1 Normalization

1. Normalization of the Salary Data Table

The initial step involves normalizing the salary data table. Below is the preliminary design:

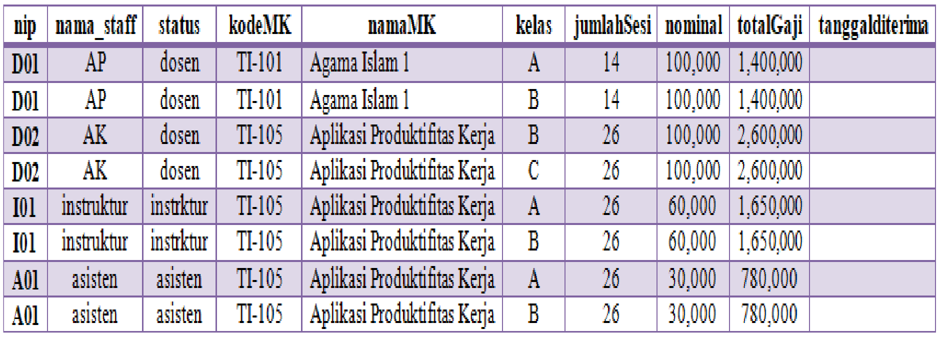
Table 3.1 Initial Design of the Staff Payroll Table



nip	nama	status	kode mk	nama mk	kelas	total sesi	nominal	total gaji	tgl terima
D01	AP	dosen	TI-101	Agama Islam 1	A	14	100,000	1,400,000	
					B	14	100,000	1,400,000	
D02	AK	dosen	TI-105	Applikasi Produktifitas Kerja	B	26	100,000	2,600,000	
					C	26	100,000	2,600,000	
I01	Instruktur	instruktur	TI-105	Applikasi Produktifitas Kerja	A	26	60,000	1,560,000	
					B	26	60,000	1,560,000	
A01	Asisten	Asisten	TI-105	Applikasi Produktifitas Kerja	A	26	30,000	780,000	
					B	26	30,000	780,000	

In Table 3.1, duplicate data is present, necessitating normalization to achieve First Normal Form (1NF). The normalized structure is shown in

Table 3.2: Staff Payroll Table in 1NF



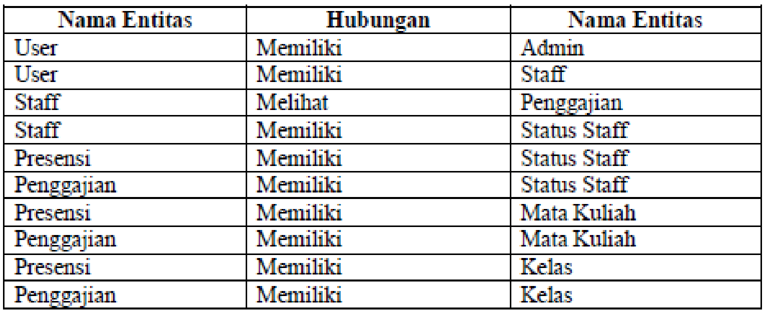
nip	nama_staff	status	kodeMK	namaMK	kelas	jumlahSesi	nominal	totalGaji	tanggalditerima
D01	AP	dosen	TI-101	Agama Islam 1	A	14	100,000	1,400,000	
D01	AP	dosen	TI-101	Agama Islam 1	B	14	100,000	1,400,000	
D02	AK	dosen	TI-105	Aplikasi Produktifitas Kerja	B	26	100,000	2,600,000	
D02	AK	dosen	TI-105	Aplikasi Produktifitas Kerja	C	26	100,000	2,600,000	
I01	instruktur	instrktur	TI-105	Aplikasi Produktifitas Kerja	A	26	60,000	1,650,000	
I01	instruktur	instrktur	TI-105	Aplikasi Produktifitas Kerja	B	26	60,000	1,650,000	
A01	asisten	asisten	TI-105	Aplikasi Produktifitas Kerja	A	26	30,000	780,000	
A01	asisten	asisten	TI-105	Aplikasi Produktifitas Kerja	B	26	30,000	780,000	

In the Staff Payroll Table (Table 3.2), dependencies still exist.

Relationship Identification

The next step involves identifying the relationships between the previously defined entities. Each entity is associated with others through specific relationships. The relationships within the staff payroll system are outlined in

Table 3.3: Staff Payroll System Relationships



Nama Entitas	Hubungan	Nama Entitas
User	Memiliki	Admin
User	Memiliki	Staff
Staff	Melihat	Penggajian
Staff	Memiliki	Status Staff
Presensi	Memiliki	Status Staff
Penggajian	Memiliki	Status Staff
Presensi	Memiliki	Mata Kuliah
Penggajian	Memiliki	Mata Kuliah
Presensi	Memiliki	Kelas
Penggajian	Memiliki	Kelas

The relationships shown in Table 3.3 are explained as follows:

a. User has Admin: This means that the Admin entity includes an `id_user` attribute, which corresponds to the admin role.

b. User has Staff: This indicates that the Staff entity includes an `id_user` attribute, which corresponds to the staff role.

c. Staff views Payroll: This means that in the Payroll entity, staff can view their payroll for each month. Staff can also see the number of teaching sessions they have conducted and the status of their payroll delivery, including the delivery date.

d. Staff has Staff Status: This means that the Staff entity includes a `status` attribute, where each staff member is assigned a single status to classify their employment status.

e. Presence has Staff Status: This means that the Attendance entity includes a `status` attribute, where each attendance record is linked to a status that classifies the attendance status of the staff member.

f. Payroll has Staff Status: This means that the Payroll entity includes a `status` attribute, where each payroll record is assigned a status that classifies the staff member’s payroll status.

g. Attendance has a Course: This means that the Attendance entity includes a `codeMK` attribute, which refers to the course code taught by staff.

h. Payroll has a Course: This means that the Payroll entity includes a `codeMK` attribute, which refers to the course code taught by staff.

i. Attendance has a Class: This means that the Attendance entity includes an `id_kelas` attribute, which refers to the class taught by staff.

j. Payroll has a Class: This means that the Payroll entity includes an `id_kelas` attribute, which refers to the class taught by staff.

Based on the above relationships, the foreign keys for the related entities can be identified. The following are the foreign keys found in the entities within the staff payroll system:

1. User has Admin

   a. The User entity is the parent table.

   b. The Admin entity is the child table.

   c. The foreign key is `id_user`, referencing the User table (parent).

2. User has Staff

   a. The User entity is the parent table.

   b. The Staff entity is the child table.

   c. The foreign key is `id_user`, referencing the User table (parent).

3. Staff views Payroll

   a. The Staff entity is the parent table.

   b. The Payroll entity is the child table.

c. The foreign key is `nip`, referencing the Staff table (parent).

4. Staff has Staff Status

a. The Staff Status entity is the parent table.

b. The Staff entity is the child table.

c. The foreign key is `status`, referencing the Staff Status table (parent).

5. Attendance has Staff Status

a. The Staff Status entity is the parent table.

b. The Attendance entity is the child table.

c. The foreign key is `status`, referencing the Staff Status table (parent).

6. Payroll has Staff Status

a. The Staff Status entity is the parent table.

b. The Payroll entity is the child table.

c. The foreign key is `status`, referencing the Staff Status table (parent).

7. Attendance has Courses

a. The Course entity is the parent table.

b. The Attendance entity is the child table.

c. The foreign key is `codeMK`, referencing the Course table (parent).

8. Payroll has Courses

a. The Course entity is the parent table.

b. The Payroll entity is the child table.

c. The foreign key is `MKcode`, referencing the Course table (parent).

9. Attendance has Classes

a. The Class entity is the parent table.

b. The Attendance entity is the child table.

c. The foreign key is `id_kelas`, referencing the Class table (parent).

10. Payroll has Classes

a. The Class entity is the parent table.

b. The Payroll entity is the child table.

c. The foreign key is `id_kelas`, referencing the Class table (parent).

Conclusion

The following conclusions can be drawn from the design and development of the database for the Staff Payroll System:

1. Based on data collection through interviews and documentation, a database design for the Staff Payroll System was developed, which can assist in creating an efficient staff payroll system.

2. The analysis and design process resulted in a database consisting of eight tables: the User table, Admin table, Staff table, Staff Status table, Course table, Class table, Attendance table, and Payroll table.

3. The initial table design was normalized to Third Normal Form (3NF).

4. The Staff Payroll System database design includes three user roles: Admin, Staff, and Master, each with its own access policy.

5. Tests performed on the database design included anomaly testing and view testing.

6. No anomalies were found during the anomaly testing of the tables.

Suggestions

The following suggestions are offered for enhancing the database in future developments:

1. Expand the database to include other study programs beyond Informatics.

2. Enhance the attendance database to allow classification based on the academic year.

References

Abdillah, L. A. (2006). Perancangan Basisdata Sistem Informasi Penggajian (Studi Kasus pada Universitas 'XYZ').

Fathansyah. (2015). *Basis Data (Revisi Kedua).* Bandung: Informatika Bandung.

Indrajani, S. M. (2017). *Database Design (Theory, Pratice, and Case Study).* Jakarta: PT Elex Media Komputindo.

McLeon, dkk. (2001).Mulyadi, S. M. (2014). Implementasi Sistem Basis Data Untuk Mendukung

Pengembangan Sistem Informasi Manajemen Sumber Data Manusia (Studi Kasus : STIKOM Dinamika Bangsa).

Purbadian, Y. (2016). *FRAMEWORK CODEIGNITER 3; MEMBANGUN APLIKASI PENGGAJIAN UNTUK PADUAN SKRIPSI.* Cirebon: CV.

ASFA Solution, Software Development, IT & Publishing.