Website Based Student Absence Application Design Using Extreme Programming Method
(Study at A University in Bandung)

Mochamad Jejen Jaelani1, Soecipto2, Yenni Fatman3, Siti Nur4
Universitas Islam Nusantara, Bandung1,2,3,4
jejenmochamad@gmail.com1, cipto.bdg@gmail.com2, yennifatman@gmail.com3,
sitin.ur.uninus@gmail.com4

Abstract
Absenteeism application is one of the systems where attendance is critical data in every activity, be it lecture activities. Nevertheless, often, the data - this absence of data is less noticed. The proof is that many educational institutions still use conventional absenteeism. Conditions like this are certainly very vulnerable to fraud. Therefore, the Informatics Engineering Study Program of the Faculty of Engineering needs a system that can create an absence process that can produce information more efficiently and by its standards, of course, using methods that can solve the problems in the Informatics Engineering Study Program of the Faculty of Engineering—using the Extreme Programming Method. The extreme Programming method was chosen because the software to be created is simple and classified as small-scale software, and development takes little time. This system development method is one of the agile methods. Which has several stages, namely: 1. Planning, 2. Design, 3. Coding, and 4. Testing. Of course, with some design processes that use UML, such as Use Case Diagram, Activity Diagram, Flowchart & and DFD, the student attendance website application built has been tested for eligibility using Blackbox testing. The results of this study create a system that can help informatics engineering programs to help in terms of student retirement. Design a system that can produce reports that suit needs effectively and efficiently.

Keywords: Student Absenteeism, Extreme Programming Methods, UML, Blackbox

INTRODUCTION
The swift advancement of information technology in contemporary times has significantly influenced the efficacy and proficiency of all undertakings. Computing technology plays a pivotal role in the continuous progress of various fields. (Firliana & Rhohman, 2019) In contemporary times, items once considered luxurious or scarce have become ubiquitous and essential, especially in governmental and private establishments. The employment of computers has progressed into the swiftest method of communication and learning.

Technology integration in education has become widespread, including at Universitas Islam Nusantara. Most educational institutions have adopted computer technology for various academic purposes, such as processing student data and recording inventory, including attendance and other academic activities. Efficient compilation of student data is crucial for recapitulation.

Being present in a given activity is essential in diverse contexts, including academic and professional environments. This condition is because attendance data plays a critical role in various aspects. Nevertheless, attendance data should be more frequently noticed. Empirical data indicates that a significant number of educational establishments continue to utilize traditional methods of monitoring attendance.
These circumstances are susceptible to academic dishonesty since attendance holds a significant role in the overall management of the course. Attending classes holds significant importance as it encompasses crucial data, including students' attendance rates, which can influence their academic performance, thereby raising concerns regarding the precision of the resultant data. (Colclasure et al., 2020; Zhang et al., 2021)

An approach to tackle this matter involves digitizing student attendance and catering to students and lecturers at a university in Bandung. This initiative targets the Faculty of Engineering, Department of Informatics. The attendance procedure remains traditional, whereby students provide their signatures on attendance sheets corresponding to their ongoing courses. (Yusuf et al., 2016) The prescribed protocol for commencing a class entails the act of recording the presence of students and the retrieval of pertinent documentation. The manual attendance procedures also undergo rotation during class sessions. (Ariyanti et al., 2020)

At a university in Bandung, there is a prevalent occurrence of cheating in attendance records. Consequently, the Department of Informatics at a university in Bandung necessitates a system that can optimize the attendance process, leading to enhanced efficiency and uniformity of data, utilizing the Extreme Programming Methodology. (Setiawansyah et al., 2021; Shameem et al., 2020) In general terms, Extreme Programming (XP) is a methodology that facilitates swift system development and testing while upholding high-quality standards. (Hakak et al., 2019; Perkusich et al., 2020)

**METHOD**

Extreme Programming (XP) has been selected as the methodology for developing the Attendance system. (Jay et al., 2021; Melegati et al., 2019) The decision to adopt Extreme Programming as the software development methodology is based on the software's relatively low complexity and its classification as small-scale software, facilitating a more expedited development process. (Nurkholis et al., 2021; Zhou et al., 2021) As mentioned earlier, the approach to system development is classified as an agile methodology intended to facilitate the iterative and continuously evolving phases of business system development. (Han & Ghadimi, 2022; Ibrahim Ahmed Osman et al., 2021) The Extreme Programming methodology comprises a series of stages, which are as follows:

1. **Planning**

The planning stage of software development entails collecting preliminary user requirements, commonly called user stories. This condition facilitates the developers' comprehension of the business context, system output requirements, and essential software features. The initial stage of software development involves identifying and analyzing application requirements. This condition involves identifying issues, creating user stories as the fundamental aspect of Extreme Programming (XP) planning, establishing software objectives, identifying users,
determining necessary information, and analyzing application requirements.

2. Design
The present study employs UML models, including use case and activity diagrams, to illustrate the system design. The design functions as a visual or conceptual model of the system, aiding in streamlining the development process. The software development methodology known as Extreme Programming employs Class Responsibility Collaborator (CRC) cards to identify and organize the classes involved in the process. The application design process encompasses the development of introductory classes utilizing the Class Responsibility Collaborator (CRC) technique, the design of a graphical user interface (GUI), and the establishment of a database design.

3. Coding
During the coding stage, the development process entails the direct implementation of the system design that was previously established. The activities encompassed in the project involve the development of the database and the execution of the user interface. The study's system development employs programming languages, namely PHP, HTML, and MySQL, with the support of XAMPP and the Laravel Framework.

4. Testing
The testing phase is intended to assess the functionality of the developed website application and ascertain its proper operation. Iterative testing is conducted in order to detect possible errors or issues. The evaluation procedure involves assessing the web application software's design, functionality, and performance.

Figure 1. Depicts the Extreme Programming Method
The utilization of the Extreme Programming approach in software development presents various benefits, such as:

1. Enhanced customer contentment.
2. Expedited duration of development.
3. The topic of interest is the efficacy of client communication.
4. Decreased expenses associated with creating a product or service.
5. The methodology employed in this study is semi-formal.

Nonetheless, the Extreme Programming (XP) development approach has its drawbacks:

1. The principle of simplicity advocates for addressing immediate needs, which may result in an inability to create detailed code from the outset.
2. The development process relies primarily on the initial documentation provided by users, with minimal formal documentation.

RESULTS AND DISCUSSION
Attending is a means of evaluating the degree of work ethic and compliance with relevant
policies. The objective is to enhance the quality and service of an institution by improving discipline. The utilization of attendance serves as an indicator of discipline within a given setting and assesses the efficacy of its operational framework. Therefore, attending can also contribute to positively evaluating institutions that have implemented it.

Following the requirements analysis, the subsequent phase of system development entails the generation of system and software design. The system and software design phase marks the primary stage in the development of the Student Attendance System.

1. Context Diagram
   The Context Diagram provides a comprehensive overview of the system, depicting the system's inbound and outbound data flow.

   ![Figure 2. Context Diagram](image)

2. Data Flow Diagram (DFD)
   Data Flow Diagrams are utilized to depict the progression of data and procedures within a system, as well as the entities implicated in it.

   ![Figure 3. DFD Level 1](image)
   ![Figure 4. DFD Level 2](image)
   ![Figure 5. DFD Level 3](image)

The design of this attendance system can improve the old system by providing a clear picture or view of the system design process from start to finish of the study.

1. Use Case Diagram
   Use Case Diagrams are used to model the interaction process based on the perspective of system users. Use Case This diagram consists of diagrams for use cases and actors.
2. Flowchart Diagram
Flowchart diagrams that describe the sequence of one process with another process so that it is easy to understand and understand.

a. Flowchart of Student Attendance Diagram

Figure 7. Flowchart of Student Attendance Diagram

b. Lecturer Absenteeism Flowchart Diagram

Figure 8. Flowchart of Lecturer Attendance Diagram

c. Schedule Input Diagram Flowchart

Figure 9. Schedule Input Diagram Flowchart

d. Flowchart Diagram of Student Data Input
The coding implementation stage, also known as programming, is the subsequent phase of software development following system design activities. Once the system is prepared for operation, it comprises a description of the implementation of the database and the interface.

1. Database Implementation

The attendance application utilized by the student population employs a MySQL database, which bears the nomenclature "absent online." The table structure, which has been devised, is executed on the MySQL server using the XAMPP web server utilizing the Apache network. Utilizing the programming language PHP version 7.0. The following table is present in the absence line database, encompassing:
Process and generate the expected output per the design specifications. The categorization of system testing will be based on user roles, specifically the Admin master (Tendik), Lecturers, and Students.

1. Login User Admin (Tendik)

Table 1. Login User Admin (Tendik)

<table>
<thead>
<tr>
<th>No</th>
<th>Objective</th>
<th>Input</th>
<th>Expected output</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Know the response of the login display if the user username and password are correct</td>
<td>Username and passwords</td>
<td>Displays the dashboard page of the system</td>
</tr>
<tr>
<td>2</td>
<td>Know the response of the login display if the password is wrong and to change the password or reset</td>
<td>Passwords</td>
<td>Displays the display when changing the password or resetting</td>
</tr>
<tr>
<td>3</td>
<td>Know the response when the email has not been registered</td>
<td>E-mail or users</td>
<td>View where the email error occurred</td>
</tr>
<tr>
<td>4</td>
<td>Know the response when you want to reset the password again</td>
<td>Reset Passwords</td>
<td>Display when resetting password and setting link to reset password to email page</td>
</tr>
</tbody>
</table>
Manik,
Exploring Leadership Style on Employee Competence:
Authentic and Servant Leadership on Workability

<table>
<thead>
<tr>
<th>No</th>
<th>Objective</th>
<th>Input</th>
<th>Expected output</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Login and Reset Password section</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Know the response when you have entered the email intended to reset the password</td>
<td>Reset Passwords</td>
<td>Display for reset password</td>
</tr>
<tr>
<td>6</td>
<td>Know the response when to change the password</td>
<td>Reset Passwords</td>
<td>Display to change the password</td>
</tr>
<tr>
<td>2. Lecturer Data Section</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Knowing the response when importing lecturer data</td>
<td>Import lecturer data</td>
<td>Display for importing lecturer data</td>
</tr>
<tr>
<td>8</td>
<td>Knowing the response when importing lecturer data is successful</td>
<td>Import lecturer data</td>
<td>Display when successfully adding lecturer data</td>
</tr>
<tr>
<td>9</td>
<td>Knowing the response when successfully editing lecturer data</td>
<td>Lecturer data</td>
<td>Displays information when the lecturer edits data successfully changes data</td>
</tr>
<tr>
<td>10</td>
<td>Knowing the response when editing lecturer data</td>
<td>Edit lecturer data</td>
<td>Displays data for editing lecturer data</td>
</tr>
</tbody>
</table>

This function logs in for lecturers who have a system test scenario like the following:

Table 2. Login User Lecturer

<table>
<thead>
<tr>
<th>No</th>
<th>Objective</th>
<th>Input</th>
<th>Expected output</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Knowing the response of the lecture schedule display in the lecturer user section</td>
<td>Lecture schedule</td>
<td>Displays the class schedule section</td>
</tr>
<tr>
<td>2</td>
<td>Knowing the response of the display to change the status of the lecture where the red color indicates the lecture is closed and the green color indicates the lecture is opened by the lecturer concerned</td>
<td>Change schedule status</td>
<td>Displays information when changing class schedule status</td>
</tr>
<tr>
<td>3</td>
<td>Knowing the response when opening lecture classes</td>
<td>Open lecture</td>
<td>Displays a display where successfully opened lectures in the lecturer user section</td>
</tr>
<tr>
<td>4</td>
<td>Know the response when closing</td>
<td>Close lecture</td>
<td>Displays information when closing</td>
</tr>
</tbody>
</table>
### CONCLUSION

Based on the conducted research, the following conclusions were drawn. The Extreme Programming methodology was effectively employed to develop a student attendance website, which involved a comprehensive process comprising four key stages: planning, design, coding, and testing. The feasibility of the Student Attendance Website application assessed via Blackbox testing methodology. The feasibility testing outcomes suggest that the application exhibits satisfactory performance, with all features of the student attendance website operating correctly. Hence, the website can serve as a platform for students to mark their attendance during lectures in the Informatics Engineering Program offered by the Faculty of Engineering. This system aims to aid the Informatics Engineering Program manage student attendance and devise a streamlined system that produces reports tailored to particular requirements.
REFERENCES


Perkusich, M., Chaves e Silva, L., Costa, A., Ramos, F., Saraiva, R., Freire, A., Dilorenzo, E., Dantas, E., Santos, D.,


