Student Service Corner

Puput Budi Wintoro [1]✉), **Deny Budiyanto** [2]), **Resty Annisa** [3])

[1)2)3)]*Program Studi Teknik Informatika, Universitas Lampung, Lampung-Indonesia*

[1)]`budi.wintoro@eng.unila.ac.id`

[2)]`deny.budiyanto@eng.unila.ac.id`

[3)]`resty.annisa@eng.unila.ac.id`

Abstract— Student service corner is an application designed as a student service system by utilizing a laravel framework in the Informatics Engineering study program, University of Lampung. The choice of Laravel as an application development framework at this time is because this application will continue to be developed with several programmers and will be integrated with other systems into a larger informatics study program. Data were collected by conducting interviews and literature studies. Data modeling in this system uses an entitty relationship diagram / ERD. The test result of this system is that the level of student satisfaction with services in the Information Engineering study program increases because it is always available because it is online. The admin of the study program can manage and request services for follow up. The results of this test show that the student service corner application can meet the system requirements as a communication between the administration and students in a value service system and correspondence requests.

Keywords— Student Service Corner, Laravel, student administration services

I. INTRODUCTION

Informatics Engineering, University of Lampung already has Standard Operating Procedures in the implementation of student services. Services to students so far are still done manually, students need to come to the administration department to take care of all campus administrative needs [1]. Problems arise when students carry out service arrangements together, so that it takes a long time to manage a service. Of course, this can interfere with running academic activities [2].

Informatics Engineering program needs a system that can serve students online to meet the national standards of College in quality of services [3]. Student service corner is a student service system that is able to assist study program administration tasks in managing student service needs. Students are expected to be faster in getting the services they need while still following all existing Standard Operating Procedures. This system is built on a web basis using the Laravel framework using the MVC (Model View Controller) concept [4], so that it is easy to develop further because this system is a small part of a larger system, namely the informatics engineering study program portal.

II. RESEARCH METHODS

A. Data Collection

We collect all data in this research using some important method, they are:

a. Interview

Interviews were conducted with the administration, students, lecturers, and the head of the study program as actors and potential users of the application.

b. Observation and Documentation

Observations are made by observing the administrative service process that is taking place and reading SOPs related to student administrative services. The documentation carried out is in the form of photographs and videos taken at the research location.

c. Questionnaire

This questionnaire was conducted by giving several questions to the respondent to be answered. This questionnaire was designed to analyze the needs of the user which aims to determine how important the student service corner application is made and used [5].

B. Agile Method

Software Development is a type of information technology project that focuses on creating or developing software. One of the methods of software development is the Agile Development Method [6]. Agile Development Methods are a collection of software development methods based on iterative and incremental development where terms and solutions change through collaboration between self-organizing and cross-functional teams. This encourages adaptive planning, an iterative approach that is time-boxed, and is flexible and responds quickly to change [7].
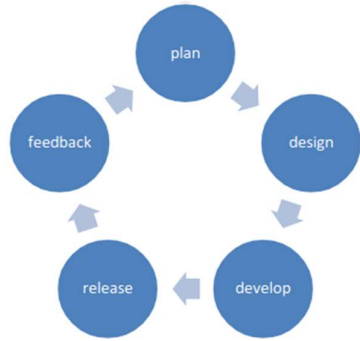
Figure 1. Agile Method Life Cycle

Registration: the admin section creates an account for the Head of Study Program user, the administration section, and students.
Login: the admin and the students both can enter the system using the account that the admin has created
Requesting services: students can request services such as grades and correspondence services.
Processing services: admin will process service requests made by students.
Service monitoring: students can see the requested service process journey.
Validating services: the administration and head of study programs validate the services submitted by students.

III. RESULT AND DISCUSSION

The concept of this application is to build a web-based administrative service application.
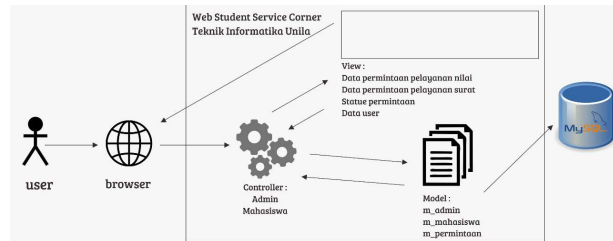
Figure 2. System Design

Users in this application are students, study program administrators, the head of the study program as the authorized official in correspondence. To be able to access services on the student service corner application, users need to use a web browser. After that the user can log in using a username and password. Then the user will be taken to a web page in accordance with the user's status. On the web, it consists of models, views, controllers. Controllers consist of admins and students. View consists of value service request data, service request data correspondence, request status, user data and others. The model consists of m_admin, m_mahasiswa, m_request and others. Then the model is connected directly to the MySQL database which consists of several tables such as admin, student, and request tables.
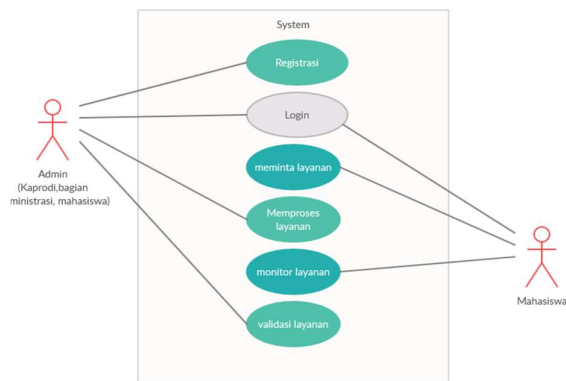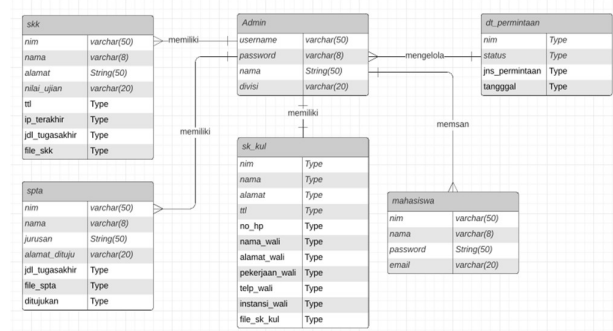

Figure 4. ER Diagram Database

System Testing
The test method used in this study is the black box testing method. Black box testing functions on the functional specifications of the application, testers or application testers can define a set of input conditions and perform tests on the functional specifications of the application [8]. If the system of the application gives inappropriate results, it means that there is an error from the application system. The criterion against which the success of the system is measured is if the system can provide output in accordance with the application flow.
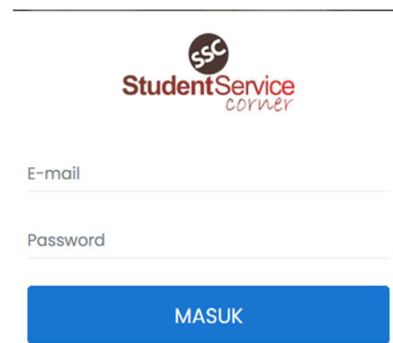
System Testing Results
Login page

Figure 5. Login Page

Figure 3. Use Case Diagram System

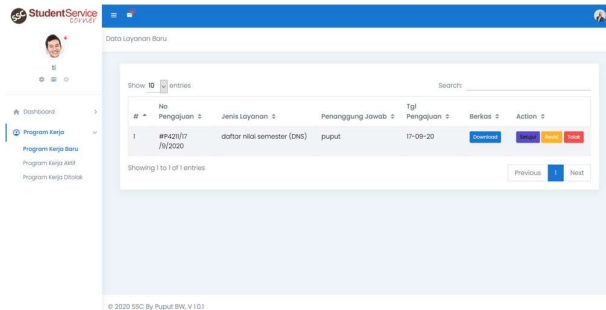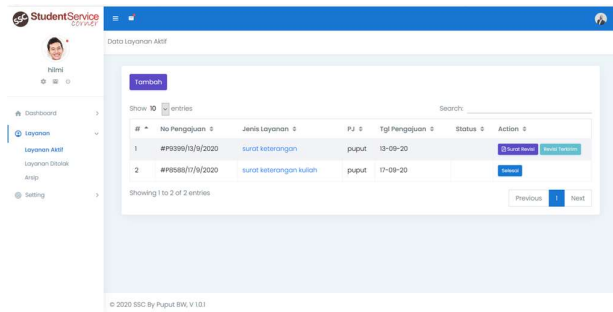| Input	: Input Username & Password Correctly
Output	: Success Login to System
Information	: If username & password both correct, page will go to User Page(admin/student) depend on privileges

Figure 6. Student Page

Input	: Choose "services" then click "add services"
Output	: list of student services page
Information	: admin can edit, approve, reject services

Figure 7. Admin Page

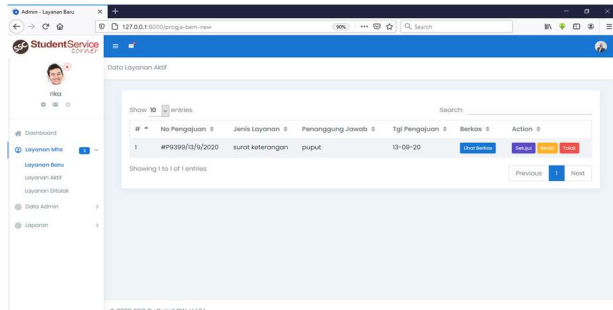| Input	: Select student services then click archives to open the services archive or click receive service
Output	: service application form page
Information	: after submitted, the services will send to administration and student page

Figure 8. Head of program page

Input	: Select student services then click archives to open the services archive or click receive service
Output	: service application form page
Information	: after submitted, the services will send to administration and student page

IV. CONCLUSION

Based on the discussion that has been done on the system, there are several conclusions

1. The Laravel framework can be used to build a student service corner student service system

2. The use of the Laravel framework makes it easier to develop systems, especially when working with teams

3. The service monitoring feature found in the student service corner application really helps students in monitoring the progress of their submissions

4. The list of semester scores can be viewed directly on the student service corner system and can be downloaded directly

REFERENCES

[1] S. Putri *et al.*, "Rancang Bangun Sistem Antrian Pelayanan Mahasiswa Berbasis Android Di Fakultas Teknik Universitas Persada Indonesia Y.a.I," vol. 4, no. 2, p. 2, 2019.

[2] A. Sidik and N. T. Sunggono, "Aplikasi Layanan Mahasiswa Berbasis Web Menggunakan Radio Frequency Identification (RFID)," *J. SISFOTEK Glob.*, vol. 3, no. 2, 2013.

[3] Saniya, "Emitor : jurnal teknik elektro.," *Emit. J. Tek. Elektro*, vol. 16, no. 1, pp. 16–22, 2016, [Online]. Available: http://journals.ums.ac.id/index.php/emitor/article/view/2678.

[4] G. Gunawan, A. Lawi, and A. Adnan, "Analisis Arsitektur Aplikasi Web Menggunakan Model View Controller (MVC) pada Framework Java Server Faces," *Sci. J. Informatics*, vol. 3, no. 1, pp. 55–67, 2016, doi: 10.15294/sji.v3i1.5958.

[5] S. Roopa and M. Rani, "Questionnaire Designing for a Survey," *J. Indian Orthod. Soc.*, vol. 46, no. December, pp. 273–277, 2012, doi: 10.5005/jp-

journals-10021-1104.

[6]   I. Mahendra and D. T. Eby Yanto, "Sistem Informasi Pengajuan Kredit Berbasis Web Menggunakan Agile Development Methods Pada Bank Bri Unit Kolonel Sugiono," *J. Teknol. Dan Open Source*, vol. 1, no. 2, pp. 13–24, 2018, doi: 10.36378/jtos.v1i2.20.

[7]   I. Sommerville, *Software engineering (10th edition)*. 2016.

[8]   A. A. Arwaz, T. Kusumawijaya, R. Putra, K. Putra, and A. Saifudin, "Pengujian Black Box pada Aplikasi Sistem Seleksi Pemenang Tender Menggunakan Teknik Equivalence Partitions," *J. Teknol. Sist. Inf. dan Apl.*, vol. 2, no. 4, p. 130, 2019, doi: 10.32493/jtsi.v2i4.3708.