

# LAMPIRAN A

UserController.php (Menampilkan form pemilihan jadwal kursus dan tampilan konfirmasi jadwal kursus)

```
public function showMeetingForm($student_real_name, $enrollment_id,
$meeting_number) {
    // Fetch enrollment and course details
    $enrollment = Enrollment::findOrFail($enrollment_id);
    $course = $enrollment->course;

    // Check if this is the last meeting
    $isLastMeeting = $meeting_number == $course->course_length;

    // Find the enrollment data for this student
    $enrollment = enrollment::with(['schedule', 'coursePayment'])->find($enrollment_id);

    // Manipulate and localize this page to Indonesian
    Carbon::setLocale('id');

    // Get the Open Time of the Admin's
    $openTime = \Carbon\Carbon::parse($enrollment->course->admin->open_hours_for_admin);
    // Get the Close Time of the Admin's
    $closeTime = \Carbon\Carbon::parse($enrollment->course->admin->close_hours_for_admin);
    // Get the course duration of the selected schedule
    $courseDuration = $enrollment->course->course_duration;

    // Generate the course time option until the start time is not more than close
    time
    while ($openTime->lessThan($closeTime)) {
```

```

// Generate the end time from adding the open time with course duration
$endOptionTime = $openTime->copy()->addMinutes($courseDuration);

// When the end time is passed the close time, end the generation
if ($endOptionTime->greaterThan($closeTime)) {
    break; // Exit the loop if it exceeds
}

// Add the slot to available slots
$availableSlots[] = [
    'start' => $openTime->format('H:i'),
    'end' => $endOptionTime->format('H:i'),
];

// Create new start time by adding the previous start time with course
duration
$openTime->addMinutes($courseDuration);
}

// Return the form for selecting the date and time for the current meeting
return view('student-page.user-choose-schedule', [
    'pageName' => "Pilih Jadwal Kursus | ",
    'enrollment' => $enrollment,
    'course' => $course,
    'meeting_number' => $meeting_number,
    'availableSlots' => $availableSlots,
    'isLastMeeting' => $isLastMeeting,
]);
}

public function showConfirmationPage($student_real_name, $enrollment_id) {

```

```

// Fetch the schedule data from the session
$enrollment = Enrollment::findOrFail($enrollment_id);
$course = $enrollment->course;
$student_real_name = $enrollment->student_real_name;
$meetings = [];

for ($i = 1; $i <= $course->course_length; $i++) {
    $meetings[] = [
        'date' => session()->get("meeting_{$i}_date"),
        'time' => session()->get("meeting_{$i}_time"),
    ];
}

Carbon::setLocale('id');

return view('student-page.user-confirm-schedule', [
    'pageName' => "Konfirmasi Jadwal Kursus | ",
    'enrollment' => $enrollment,
    'student_real_name' => $student_real_name,
    'meetings' => $meetings,
]);
}

```

CourseScheduleController.php (Memeriksa tabrakan jadwal dan menyimpan jadwal yang dipilih)

```

public function storeMeetingData(Request $request, $student_real_name,
$enrollment_id, $meeting_number) {
    $request->validate([
        'course_date' => 'required|date',
        'course_time' => 'required',

```

```
], [
    'course_date.required' => 'Silahkan Pilih Tanggal Kursus',
    'course_time.required' => 'Silahkan Pilih Salah Satu Opsi',
]);
```

Carbon::setLocale('id'); // This can be set in a service provider or at the start of your controller

```
// Get the Enrollment Data
```

```
$enrollmentData = enrollment::findOrFail($enrollment_id);
```

```
// Fetch the instructor_id from Enrollment Data
```

```
$instructor_id = $enrollmentData['instructor_id'];
```

```
// Get the course_length
```

```
$courseLength = $enrollmentData->course->course_length;
```

```
// dd($request);
```

```
// Extract start and end times from course_time
```

```
list($start_time_str, $end_time_str) = explode(' - ', $request->course_time);
```

```
$currentDate = \Carbon\Carbon::parse($request->course_date);
```

```
// Create full datetime for start and end times
```

```
$selectedStartTime = \Carbon\Carbon::parse($currentDate->format('Y-m-d')
```

```
. ' ' . $start_time_str);
```

```
$selectedEndTime = \Carbon\Carbon::parse($currentDate->format('Y-m-d') .
```

```
' ' . $end_time_str);
```

```
// dd($selectedStartTime, $selectedEndTime);
```

```
// Check for earlier date than the previous meeting
```

```
if ($meeting_number > 1) {
```

```

$previousMeetingDate = session()->get("meeting_" . ($meeting_number -
1) . "_date");

```

```

$previousMeetingDateCarbon =
\Carbon\Carbon::parse($previousMeetingDate);

```

```

if ($currentDate->lt($previousMeetingDateCarbon)) {
    $request->session()->flash(
        'error',
        'Tanggal untuk Pertemuan ' . $meeting_number . ' tidak boleh lebih
awal dari tanggal Pertemuan ' . ($meeting_number - 1) . ' (' .
$previousMeetingDateCarbon->translatedFormat('d F Y') . ')'.
    );
    return redirect()->back();
}
}

```

```

// Check for conflicts here (but don't store in database yet)
$existingSchedule = courseSchedule::where(function ($query) use
($instructor_id, $selectedStartTime, $selectedEndTime) {
    $query->where('instructor_id', $instructor_id)
        ->where(function ($q) use ($selectedStartTime, $selectedEndTime) {
            $q->whereBetween('start_time', [$selectedStartTime,
$selectedEndTime])
                ->orWhereBetween('end_time', [$selectedStartTime,
$selectedEndTime])
                ->orWhere(function ($q) use ($selectedStartTime,
$selectedEndTime) {
                    $q->where('start_time', '<=', $selectedStartTime)
                        ->where('end_time', '>=', $selectedEndTime);
                });
        });
});

```

```

    })->first();

    // When there's a conflicting schedule, return error
    if ($existingSchedule) {
        $request->session()->flash('error', 'Instruktur ' . $existingSchedule->instructor->fullname . ' sudah terjadwal pada pukul ' . $request->course_time . ' di ' . $currentDate->translatedFormat('d F Y') . '. Silakan pilih waktu atau tanggal lain.');
```

return redirect()->back();

```
    }

    // Store the selected meeting data in session
    session()->put("meeting_{$meeting_number}_date", $request->course_date);
    session()->put("meeting_{$meeting_number}_time", $request->course_time);

    // dd(session()->get("meeting_{$meeting_number}_date"), session()->get("meeting_{$meeting_number}_time"));

    // Redirect to the next meeting or confirmation page
    $nextMeetingNumber = $meeting_number + 1;

    if ($nextMeetingNumber > $courseLength) {
        return redirect()->route('schedule.confirmation', [
            'student_real_name' => $student_real_name,
            'enrollment_id' => $enrollment_id,
        ]);
    } else {
        return redirect()->route('schedule.form', [
            'student_real_name' => $student_real_name,
```

```

        'enrollment_id' => $enrollment_id,
        'meeting_number' => $nextMeetingNumber,
    );
}
}

public function saveSchedule(Request $request, $student_real_name,
$enrollment_id) {
    // Save all meeting data to the database
    $enrollment = Enrollment::findOrFail($enrollment_id);
    $course = $enrollment->course;

    // Retrieve the submitted meetings from the request
    $meetings = $request->input('meetings');

    // Save each meeting
    foreach ($meetings as $meetingNumber => $meetingData) {
        $newSchedule = new CourseSchedule();
        $newSchedule->enrollment_id = $enrollment_id;
        $newSchedule->course_id = $course->id;
        $newSchedule->instructor_id = $enrollment->instructor_id;

        // Parse date and time
        $newSchedule->start_time = Carbon::parse($meetingData['date'] . ' ' .
$meetingData['start_time']);
        $newSchedule->end_time = Carbon::parse($meetingData['date'] . ' ' .
$meetingData['end_time']);
        $newSchedule->meeting_number = $meetingNumber; // Convert to 1-
based index
        $newSchedule->theoryStatus = 0;
        $newSchedule->quizStatus = 0;
    }
}

```



```
// dd($newSchedule);

$newSchedule->save();
}

// Clear the session
session()->forget('meeting_*');

$request->session()->flash('success', 'Jadwal berhasil dibuat. Silahkan
hubungi Admin Kursus untuk proses lebih lanjut.');
```

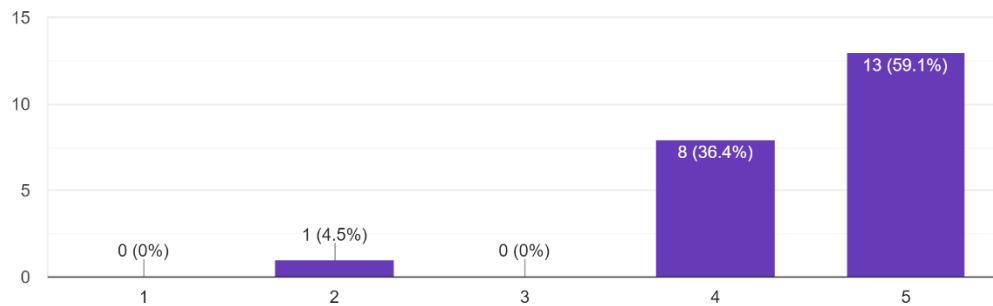
```
return redirect(url('/user-course-progress/' . $student_real_name . '/' .
$enrollment_id));
}
```

# **LAMPIRAN B**

## Pertanyaan dan hasil jawaban untuk Faktor *Functional Suitability*

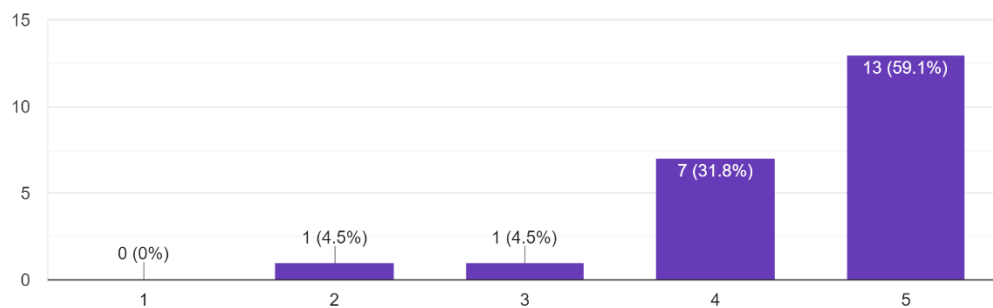
1. Apakah KEMUDI menyediakan semua fungsi yang dibutuhkan oleh Anda untuk mengelola, mendaftar atau melakukan proses kursus mengemudi yang diinginkan?

22 responses



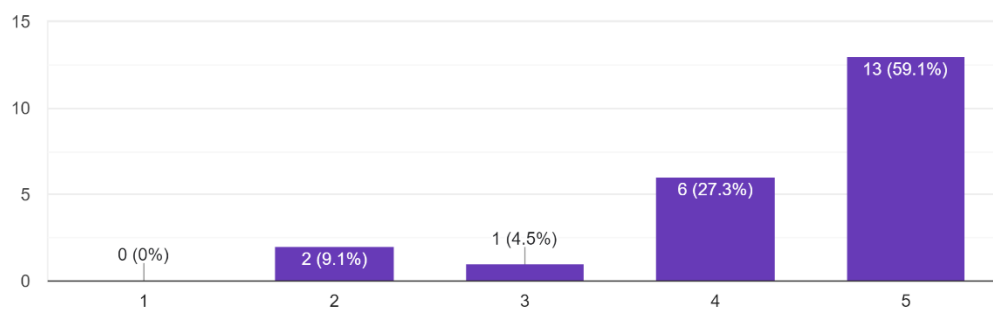
2. Apakah KEMUDI memberikan hasil yang benar dan konsisten sesuai dengan harapan Anda?

22 responses



3. Apakah fungsi-fungsi yang disediakan oleh KEMUDI membantu Anda menyelesaikan mengelola, mendaftar atau melakukan proses kursus mengemudi dengan efisien?

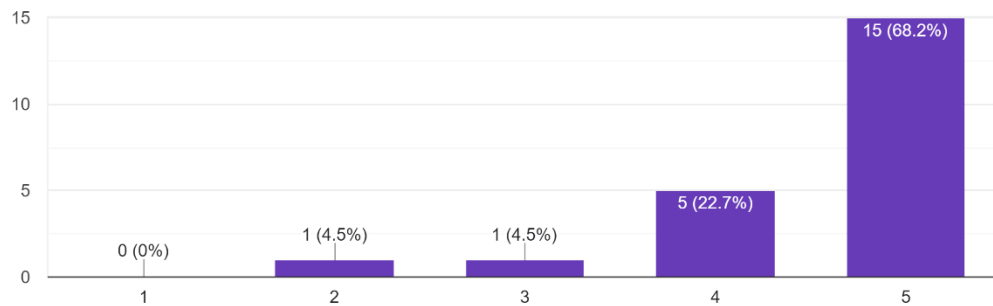
22 responses



## Pertanyaan dan hasil jawaban untuk Faktor *Interaction Capability*

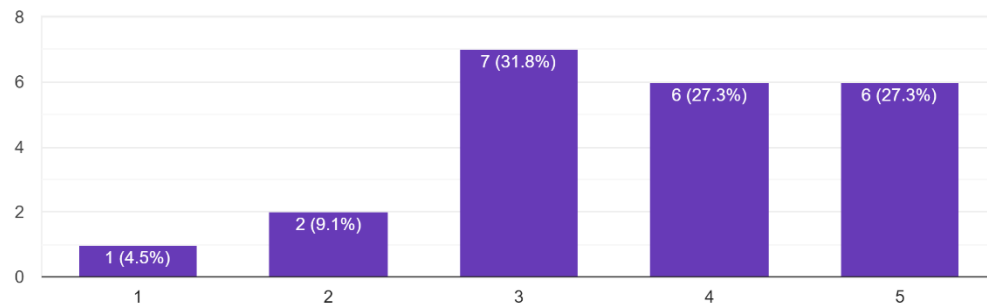
1. Apakah Anda setuju bahwa KEMUDI membantu anda dalam mengelola, mendaftar atau melakukan kursus mengemudi?

22 responses



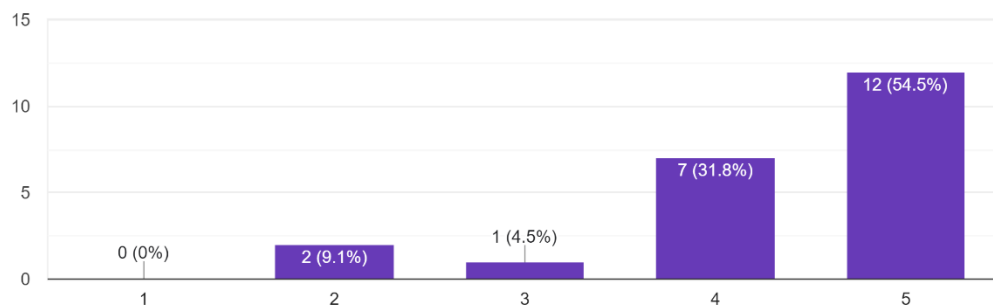
2. Apakah Anda merasa bahwa Anda membutuhkan waktu untuk membiasakan diri dengan KEMUDI?

22 responses



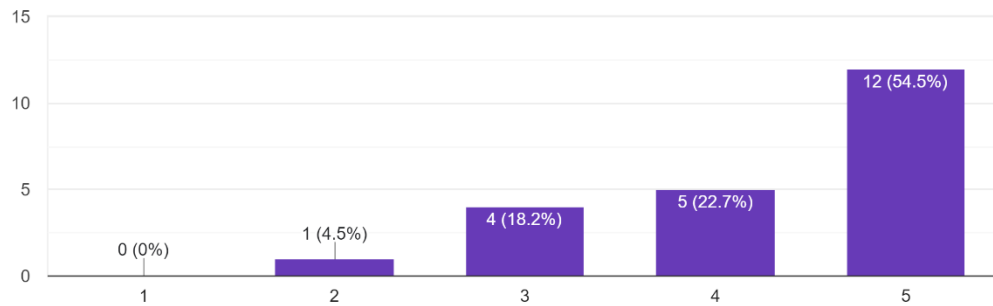
3. Apakah Anda dapat mengoperasikan KEMUDI dengan mudah dan tanpa kesulitan?

22 responses



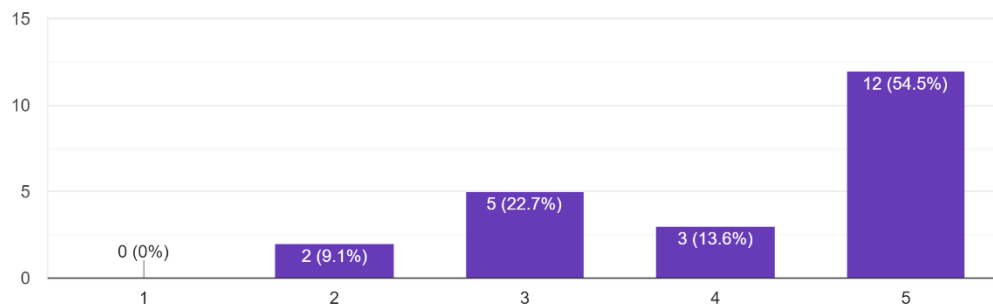
4. Apakah KEMUDI membantu Anda mencegah kesalahan dan memberikan bantuan saat terjadi kesalahan?

22 responses



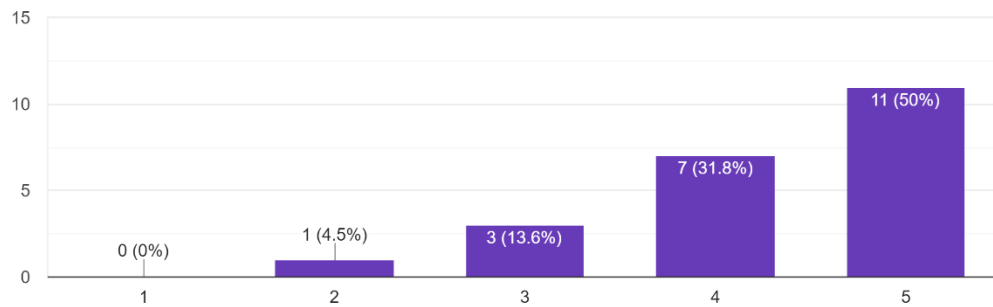
5. Apakah KEMUDI berhasil membuat Anda ingin kembali menggunakannya secara terus menerus?

22 responses



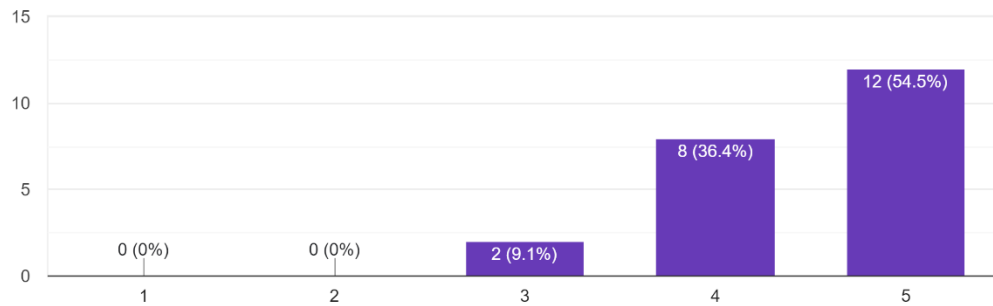
6. Seberapa mudahkan Anda bernavigasi dan memahami berbagai fitur yang ditawarkan oleh KEMUDI, terlepas dari keahlian teknis Anda?

22 responses



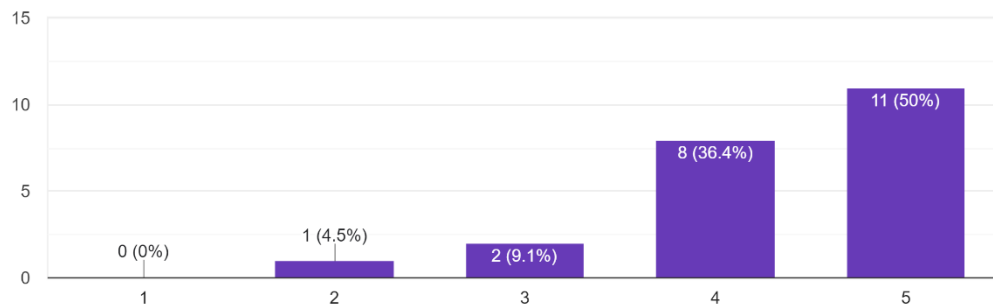
7. Seberapa puaskah Anda dengan ketersediaan dan kualitas bantuan (tutorial, pertanyaan-pertanyaan umum tentang aplikasi, dll.) yang diberikan oleh KEMUDI?

22 responses



8. Apakah label-label, petunjuk, instruksi, dan pesan kesalahan yang ditampilkan oleh KEMUDI sudah jelas dan mudah dipahami?

22 responses



# LAMPIRAN C

## **SURAT TUGAS**

Nomor : 33.S/FTETI-ITATS/VII/2024

Dasar :

1. SKEP Rektor ITATS Nomor : SKEP/22/ITATS/III/2009 tentang Bimbingan, Sidang Proposal, Sidang Progress & Sidang Akhir / Tugas Akhir Program Strata 1 Tahun 2009 Institut Teknologi Adhi Tama Surabaya.
2. SKEP Rektor ITATS Nomor : SKEP/25/ITATS/VII/2019 tentang Pedoman Akademik Program Strata I Tahun 2019/2020 Institut Teknologi Adhi Tama Surabaya.
3. Kalender Akademik Institut Teknologi Adhi Tama Surabaya Tahun 2019/2020

### **Dekan Fakultas Teknik Elektro dan Teknologi Informasi**

Dengan ini menugaskan kepada :

Nama Pembimbing I : Rachman Arief, S.Kom., M.Kom.

Nama Pembimbing II : -

Tugas : Membimbing Skripsi Mahasiswa,

Nama : Yodanis Erlandi Sutantio

N.P.M. : 13.2020.1.00905

Alamat : Perumahan Taman Puspa Sari Blok L Nomor 11, Ds. Klurak, Kec. Candi,  
Kabupaten Sidoarjo, Jawa Timur

Telp. : 082145649388

Judul Skripsi : Rancang Bangun Aplikasi Untuk Penyedia Jasa Kursus Mengemudi Berbasis Web Dengan  
*Incremental Model*

Tugas :

1. Memecahkan masalah Seminar Proposal - Skripsi dengan musyawarah mufakat yang mengedepankan obyektifitas keilmuan serta kemanfaatan bagi peserta seminar.
2. Membimbing, menguji, memberi masukan dan penilaian secara obyektif dan tidak berpihak.
3. Memonitor kelancaran proses pembimbingan skripsi.

Tempat : Program Studi Sistem Informasi

Pelaksanaan : Sesuai dengan ketentuan yang ada pada Buku Panduan Skripsi jurusan Sistem Informasi

Demikian surat tugas ini agar dilaksanakan sebaik – baiknya dengan rasa tanggung jawab.



Surabaya, 19 Juli 2024

Fakultas Teknik Elektro dan Teknologi Informasi

Dekan,

Dr. H. Hari Agus Sujono, M.Sc.

NIP. 410109940007

Tembusan :













1. Dosen yang bersangkutan
2. Mahasiswa yang dibimbing
3. Koordinator Skripsi
4. Jurusan Sistem Informasi
5. Arsip



No : 1.6.3/ITATS/FORM/SPM/C-26.2  
Tanggal Terbit : 15 September 2021  
Revisi : 00

### KEMAJUAN PENYELESAIAN SKRIPSI

Nama : Yodanis Erlandi Sutantio  
NPM : 13.2020.1.00905  
Judul Skripsi : Rancang Bangun Aplikasi Untuk Penyedia Jasa Kursus Mengemudi Berbasis Web Dengan *Incremental Model*  
Dosen Pembimbing : Rachman Arief, S.Kom., M.Kom.

No	Tanggal	Catatan Pembimbing	Tanda Tangan Pembimbing
1	24 Juli '24	Siklus incremental dibuat Per Pengguna, buatkan fitur	
2	29 Juli '24	Instructor tidak boleh ubah jadwal	
3	1 AGS '24	Hapus fitur ubah jadwal untuk Pengguna	
4	8 AGS '24	Yang Pilih jadwal biar Siswa saja.	
5	13 AGS '24	Rancangan database Pascal class diagram	
6	5 SEP '24	Proses 'Login' masukkan use case	
7	16 SEP '24	Tambah fitur 'Lupa Password'	
8	24 SEP '24	Black box testing, pakai hasil kerosi pertama	
9	3 OKT '24	Untuk tampilan aplikasi di bab 4 dipisah berdasarkan Pengguna	
10	16 OKT '24	Jadwal kursus tampilkan jadi 1 bulan	
11	21 OKT '24	Untuk jadwal di dashboard Penitk/Admin & Instructor diubah ke hijau.	
12	24 OKT '24	Tambah ringkasan bug yang ditemui setelah tabel black box di bab 4	

**Catatan:**

Persyaratan sidang minimal melaksanakan tatap muka bimbingan sebanyak 12 x dengan Dosen Pembimbing Utama + Pembimbing Pendamping



## LEMBAR REVISI

Nama : Yodanis Erlandi Sutantio  
N P M : 13.2020.1.00905  
Hari : Jumat  
Tanggal Sidang : 29 November 2024

### Aplikasi:

- Belum ada batasan umur y mjd siswa kampus
- Jk tgl lahir & tgl lulus, maka munculkan warning message yg sesuai
- Y pendaftaran alim penyedia kampus, jika izin kampusnya exp date bgm?

### Laporan:

- Tambahkan batasan masalah
- Tambahkan teori  $\geq$  dibab 2 (sda di-note dibuku)
- Berikan kesimpulan Hg hasil penguluran kualitas PL

Dosen Penguji,



**Sulistyowati, S.T., M.Kom**

NIP 411112000026

## LEMBAR REVISI

Nama : Yodanis Erlandi Sutantio  
N P M : 13.2020.1.00905  
Hari : Jumat  
Tanggal Sidang : 29 November 2024

Perbaiki sesuai catatan penguji

Dosen Penguji,



**Rachman Arief, S.Kom., M.Kom**  
NIP 411109120048

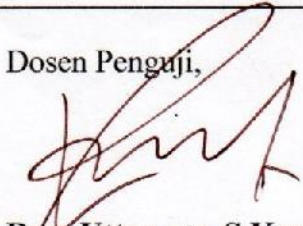


## LEMBAR REVISI

Nama : Yodanis Erlandi Sutantio  
N P M : 13.2020.1.00905  
Hari : Jumat  
Tanggal Sidang : 29 November 2024

- o Perbaiki tombol Back pada Homeisteri
- o Perbaiki ~~batas~~ menu pada program.
- o ~~Rombak ulang~~ Perbaiki. Batasan jadwal prog.
- o Perbaiki tgl izin kursus.
- o Perbaiki Search inputan data.

Dosen Penguji,

  
**Resa Uttungga, S.Kom., M.Kom**  
NIP 412112200150

## LEMBAR REVISI

Nama : Yodanis Erlandi Sutantio  
N P M : 13.2020.1.00905  
Hari : Jumat  
Tanggal Sidang : 29 November 2024

- 1) umur peserta kursus
- 2) tanggal izin penyelenggaraan
- 3) Perbaiki Aplbasi

Dosen Penguji,



**Adib Pakarbudi, S.Kom., M.Kom**  
NIP 411802190128