

## BAB III

### PELAKSANAAN KERJA MAGANG

#### 3.1 Kedudukan dan Koordinasi

Pada pelaksanaan Magang Merdeka di IT Digikidz, Divisi *Academic Operation* memiliki peran yang vital dalam menjamin kelancaran proses pembelajaran. Divisi ini secara garis besar dipimpin oleh Ibu Debie Christine, yang menjabat sebagai *Head of Operation Division*. Dalam divisi ini, terdapat dua tim utama yang bertanggung jawab atas berbagai aspek pendidikan, yakni *Tim Support* dan *Tim Curriculum*.

*Tim Support* memiliki fokus pada pendukung teknis dan administratif untuk memastikan para *coach*, atau instruktur, dapat menjalankan tugas *coaching* dengan baik. *coach* berperan dalam menangani berbagai permasalahan sehari-hari yang mungkin dihadapi oleh para *coach*, sehingga proses pembelajaran dapat berjalan lancar. Sementara itu, *Tim Curriculum* memiliki tanggung jawab yang lebih spesifik terkait dengan pengembangan materi pembelajaran. *Tim Curriculum* bertugas merancang, mengembangkan, dan menyesuaikan kurikulum sesuai dengan kebutuhan dan kemampuan para murid. Salah satu aspek penting dari tugas *Tim Curriculum* adalah menyesuaikan tingkat kesulitan materi agar sesuai dengan pemahaman dan kemampuan belajar para murid. Dengan demikian, *coach* berperan langsung dalam memastikan kualitas pembelajaran yang diberikan kepada para murid.

Divisi *Academic Operation* dan Divisi *Teacher* memiliki hubungan kerja sama yang kuat yang didasarkan pada perbedaan fokus tugas masing-masing divisi namun dengan tujuan akhir yang sama, yaitu memberikan pendidikan berkualitas kepada para murid IT Digikidz. Divisi *Academic Operation* memusatkan perhatiannya pada aspek internal dan sistem, termasuk

pengembangan kurikulum, penyesuaian materi pembelajaran, dan pengaturan sistem administratif. Sebaliknya, Divisi *Teacher* berfokus pada implementasi kurikulum di lapangan, di mana situasi dan kondisi dapat berubah. Divisi *Teacher* bertanggung jawab langsung dalam memberikan pembelajaran kepada para murid, berinteraksi dengan para murid secara langsung, dan menyesuaikan pendekatan pembelajaran sesuai dengan kebutuhan dan dinamika kelas.

Karena perbedaan fokus ini, kerja sama yang erat antara kedua divisi menjadi sangat penting. Divisi *Academic Operation* perlu memahami kondisi lapangan yang dihadapi oleh Divisi *Teacher* agar Tim *Academic* dapat merancang kurikulum yang relevan dan efektif. Di sisi lain, Divisi *Teacher* juga membutuhkan dukungan dari Divisi *Academic Operation* dalam hal penyediaan materi pembelajaran yang sesuai dengan standar kurikulum dan kemampuan para murid.



Gambar 3. 1 Team Teacher & Team AcademicOperational Digikidz Cabang BSD

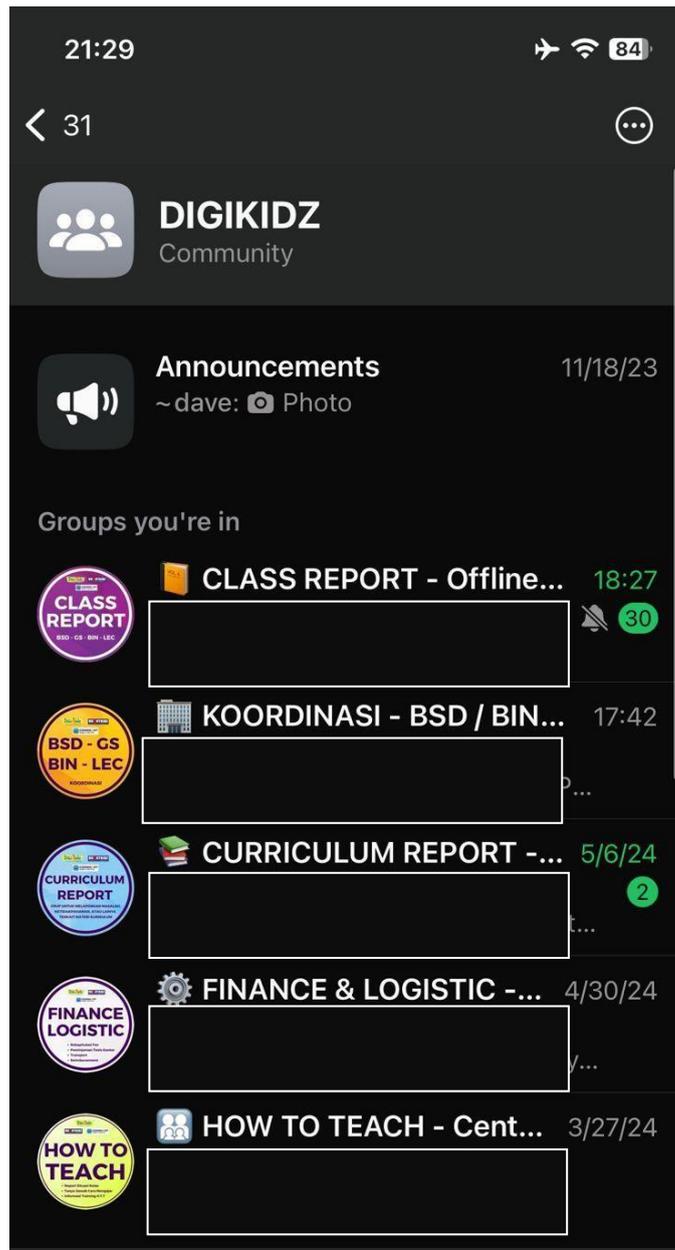
Hubungan komunikasi yang tinggi antara kedua divisi ini menjadi kunci dalam menjaga kelancaran proses pembelajaran. Dengan saling berbagi

informasi dan berkoordinasi secara teratur, sehingga sarana komunikasi yang dipilih harus terpercaya.

Pemilihan *Whatsapp* dan *Zoom* sebagai sarana komunikasi dalam pelaksanaan magang merupakan keputusan yang tepat dan didasarkan pada pertimbangan yang baik. Pertama-tama, *Whatsapp* adalah salah satu aplikasi komunikasi yang sangat populer dan luas digunakan oleh berbagai kalangan, termasuk dalam konteks profesional seperti pelaksanaan magang. Keberadaannya yang sudah teruji dan dipahami oleh banyak orang memudahkan dalam penggunaannya, sehingga meminimalkan hambatan dalam proses komunikasi antar anggota tim. Selain itu, *Whatsapp* merupakan produk Meta, perusahaan di balik platform besar seperti *Facebook*, yang telah terbukti memiliki reputasi yang baik dan dapat dipercaya dalam menyediakan layanan komunikasi. Keberadaan Meta sebagai perusahaan yang sudah mapan menambah kepercayaan terhadap keamanan dan kehandalan *Whatsapp* sebagai sarana komunikasi. Sementara itu, Penggunaan *Zoom* sebagai sarana komunikasi untuk memfasilitasi pertemuan virtual atau konferensi video dalam pelaksanaan magang IT Digikidz memperlihatkan kebijakan yang bijaksana, terutama dengan mempertimbangkan pengalaman positif yang dimiliki Divisi *Teacher* dalam penggunaannya.

Pada tahun 2020, ketika kasus COVID-19 sedang merebak, *Zoom* telah menjadi salah satu pilihan utama untuk menjalankan kegiatan pertemuan dan pembelajaran jarak jauh. Hal ini terutama terjadi karena *Zoom* telah terbukti dapat menyediakan layanan konferensi video berkualitas tinggi dengan fitur-fitur yang memadai, seperti kemampuan untuk mengadakan pertemuan dengan banyak peserta, fitur berbagi layar, dan lain sebagainya. Penggunaan *Zoom* memungkinkan pertemuan virtual dilakukan dengan efisien dan efektif, memungkinkan kolaborasi antar anggota tim secara langsung tanpa harus bertemu secara fisik. Dengan demikian, keputusan untuk menggunakan *Zoom* sebagai sarana komunikasi dalam pelaksanaan magang tidak hanya didasarkan

pada kemampuan teknis platform tersebut, tetapi juga mengambil manfaat dari pengalaman positif yang dimiliki oleh Divisi *Teacher*. Penggunaan *Zoom* sebelumnya oleh Divisi *Teacher* menunjukkan bahwa platform ini sudah diterima dengan baik dan dikuasai oleh anggota tim, sehingga meminimalkan hambatan dalam penerapannya dalam konteks pelaksanaan magang.



Gambar 3.2 Grup What's App DigikidZ

### 3.2 Tugas dalam Kerja Magang

Selama periode magang kerja di Divisi IT Digikidz dari tanggal 10 Agustus 2023 hingga 10 November 2023, peserta magang memiliki tanggung jawab yang jelas terkait dengan membantu dalam proses pembuatan sistem untuk mempermudah kerja para *coach* dalam Digikidz. Sistem yang harus dibuat bertujuan untuk membantu dalam pengaturan jadwal (*scheduling*) para *coach* guna menghindari keterlambatan dan ketidak hadiran dalam proses *coaching*.

Tanggung jawab peserta magang dalam proyek ini mencakup beberapa aspek kunci:

1. Analisis Kebutuhan.

Peserta magang perlu melakukan analisis mendalam terhadap kebutuhan dan persyaratan sistem yang akan dibuat. Ini melibatkan berinteraksi dengan para *coach* dan pihak terkait lainnya untuk memahami secara menyeluruh proses *scheduling* yang sedang berlangsung, serta kendala-kendala yang mungkin dihadapi dalam proses tersebut.

2. Perancangan Sistem.

Berdasarkan analisis kebutuhan, peserta magang harus merancang sistem secara cermat. Ini mencakup merancang antarmuka pengguna (*user interface*) yang intuitif dan fungsional, serta merancang algoritma atau metode pengaturan jadwal yang efisien dan dapat diandalkan.

3. Pengembangan.

Setelah perancangan selesai, peserta magang akan terlibat dalam proses pengembangan sistem menggunakan berbagai teknologi dan bahasa pemrograman yang relevan. Para *coach* perlu memiliki keterampilan teknis yang baik dalam pengembangan perangkat lunak untuk mengeksekusi rencana pengembangan dengan lancar.

Selain membantu dalam pengembangan sistem untuk mempermudah kerja para *coach*, para peserta magang juga memiliki tanggung jawab untuk membagikan pengetahuan yang didapatkan selama kuliah kepada para *coach* sesuai dengan bidang keahlian masing-masing. Ini merupakan kesempatan bagi para peserta magang untuk memberikan kontribusi yang lebih luas dalam pengembangan keterampilan dan pengetahuan para *coach*, seperti membagikan pengetahuan dalam bidang pemrograman dan pengembangan perangkat lunak akan membagikan pengetahuan dasar-dasar coding kepada para *coach* yang terlibat dalam program studi Teknik Informatika (TI) dan Sistem Informasi (SI). Peserta magang prodi tersebut akan memberikan pelatihan tentang konsep dasar pemrograman, bahasa pemrograman populer, dan praktik-praktik pengembangan perangkat lunak.

Dengan membagikan pengetahuan dan pengalaman kepada para *coach*, para peserta magang dapat membantu meningkatkan kualitas pembelajaran dan pengembangan keterampilan para *coach* dalam bidang yang relevan. Hal ini juga dapat menciptakan lingkungan kerja yang kolaboratif dan mendukung antara Divisi IT dan Divisi *teacher* di Digikidz. Selain memberikan manfaat langsung bagi para *coach*, membagikan pengetahuan juga merupakan kesempatan bagi para peserta magang untuk mengasah kemampuan komunikasi dan kepemimpinan, serta memperluas jaringan profesional. Dengan demikian, tanggung jawab ini tidak hanya memberikan manfaat bagi para *coach*, tetapi juga memberikan pengalaman dan pembelajaran yang berharga bagi para peserta magang di Digikidz.

Berdasarkan penjelasan secara garis besar di atas tugas-tugas yang dilaksanakan dalam Digikidz dalam divisi IT dapat dijabarkan dalam Tabel 3.1 yang akan dijabarkan dibawah ini. Tabel ini akan menjabarkan kegiatan relasi kerja magang yang telah dilakukan selama proses kerja magang yang dimulai pada tanggal 10 Agustus 2023 sampai dengan 10 November 2023.

Tabel 3. 1 Jadwal MBKM

Nomor	Tugas	Tanggal Mulai	Tanggal Selesai
1	<p>1) Pengenalan Digikidz beserta dengan para <i>coach</i> cabang BSD.</p> <p>a) Mendaftarkan diri dan menyiapkan hal yang diperlukan seperti: Member Parkir, Identitas Kartu Nama, beserta Seragam Magang.</p> <p>b) Mengikuti Pertemuan <i>Kick Off</i> dari Digikidz yang berisi tentang: Kerja sama Digikidz dengan sekolah-sekolah, Sejarah Pendirian Digikidz, Pengenalan Susunan Jabatan dalam Digikidz, dan Pengenalan peserta Magang serta para <i>coach</i> dari berbagai cabang.</p>	10 Agustus 2023	14 Agustus 2023
2	<p>1) Pengenalan terhadap Divisi IT.</p> <p>a) Mengikuti sesi pelatihan mengenai tugas dan pembagian tugas para peserta magang.</p> <p>b) Melakukan brain storming terhadap analisis dari sistem yang dibutuhkan.</p>	15 Agustus 2023	21 Agustus 2023
3	<p>1) Melakukan Perancangan dari hasil Brain Storming.</p> <p>a) Melakukan survey mengenai sistem-sistem yang sudah dipakai oleh Digikidz.</p>	21 Agustus 2023	31 Agustus 2023

	b) Melakukan diskusi terhadap jenis dari sistem yang akan didesign.		
4	1) Pembuatan Skema <i>Database</i> . a) Melakukan review ulang dari cara mengembangkan <i>database</i> . b) Membuat design <i>flowchart</i> before dan after class dari sistem.	1 September 2023	5 September 2023
5	1) Membuat design <i>Database</i> . a) Membuat design tabel dan isi tabel dari <i>database</i> . b) Membuat design Relasi dari tabel-tabel. c) Membuat revisi dari design tabel. d) Merevisi ulang design dari relasi tabel.	6 September 2023	20 September 2023
6	1) Mengembangkan <i>Database</i> . a) Menunggu approve dari design <i>database</i> . b) Membuat tabel, beserta relasi-relasi tabel sesuai design. c) Memasukan data	21 September 2023	10 Oktober 2023
7	1) Melakukan Design <i>Website</i> a) Melakukan review ulang atas cara membuat design <i>website</i> . b) Mempelajari tata cara mendesign ui dan ux.	11 Oktober 2023	16 Oktober 2023
8	Mengikuti perpisahan dari Digikidz BSD	21 Oktober 2023	21 Oktober 2023

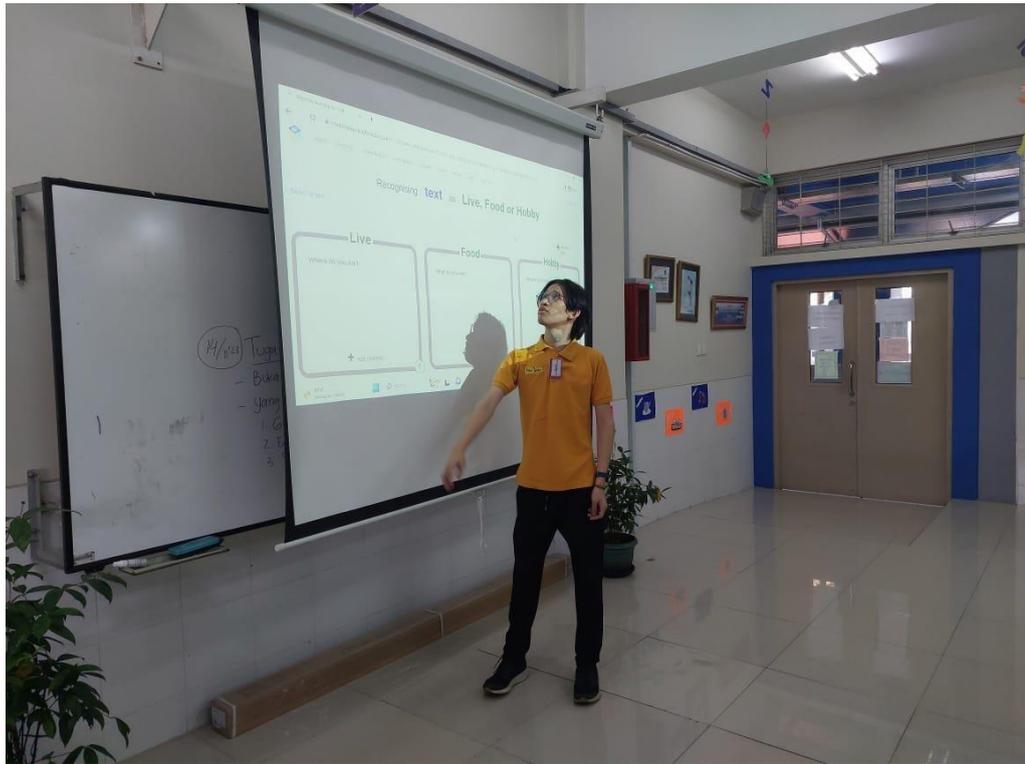
### **3.3 Uraian Kerja Magang**

#### **3.3.1 Pengenalan Digikidz BSD**

Pada awal pelaksanaan magang di Digikidz BSD, dilakukan sesi pengenalan yang diselenggarakan secara *offline* dan diberi nama sesi "*Kickoff*". Istilah "*Kickoff*" diambil dari dunia olahraga sepak bola yang menandakan dimulainya pertandingan. Digikidz BSD mengadaptasi istilah ini sebagai simbol dimulainya awal ajaran baru, yang mempunyai makna bahwa semua *coach* dan tim harus fokus untuk menghindari hal-hal yang tidak diinginkan selama proses pembelajaran berlangsung. Sesi "*Kickoff*" ini merupakan momen penting di mana semua anggota tim, termasuk para *coach* dan peserta magang, berkumpul untuk memulai perjalanan pembelajaran yang baru. Tujuan dari sesi ini adalah untuk menyatukan visi, misi, dan tujuan bersama, serta mengkomunikasikan harapan dan ekspektasi yang diperlukan selama magang berlangsung. Selain itu, sesi "*Kickoff*" juga menjadi waktu yang tepat untuk memperkenalkan semua anggota tim satu sama lain, membangun hubungan yang solid, dan menetapkan kerangka kerja yang jelas untuk kolaborasi dan kerjasama selama periode magang. Dengan adanya sesi "*Kickoff*" ini, Digikidz BSD menekankan pentingnya fokus dan dedikasi dalam mencapai tujuan bersama dalam lingkungan pembelajaran. Ini adalah langkah awal yang kuat untuk memastikan bahwa semua anggota tim siap dan berkomitmen untuk memberikan yang terbaik dalam proses pembelajaran yang akan datang.

##### **3.3.1.2 Dasar dari Pemograman**

Setelah sesi bonding kickoff, para peserta magang di Digikidz BSD melakukan proses pengajaran tentang dasar-dasar dari jurusan masing-masing. Setelah diberikan waktu untuk menyiapkan diri dan menyusun materi yang ingin dibawa oleh para peserta magang. Proses ini bertujuan untuk mengevaluasi seberapa mampu dan sejauh mana pengetahuan dasar telah dikuasai oleh para peserta magang.



Gambar 3. 3 Sesi pengajaran dasar Pemrograman

Salah satu topik yang dipilih untuk pembahasan adalah dasar-dasar pemrograman. Peserta magang akan membahas konsep-konsep fundamental dalam pemrograman yang menjadi dasar bagi semua bahasa pemrograman. Beberapa topik yang dibahas antara lain:

- 1) *Algoritma*: *Algoritma* adalah dasar dari pemrograman, karena semua program dibuat berdasarkan algoritma tertentu.
- 2) *Bahasa Pemrograman*: Bahasa pemrograman adalah bahasa yang digunakan oleh para developer untuk berkomunikasi.
- 3) *Variabel dan Tipe Data*: *Variabel* adalah tempat untuk menyimpan data dalam program, setiap variabel harus diberi nama dan memiliki tipe data tertentu.
- 4) *Pengendalian Alur Program (Control Flow)*: Pengendalian alur program adalah proses mengevaluasi sebuah kondisi dan menjalankan blok kode tertentu tergantung pada hasil.

- 5) *Syntax*: *Syntax* adalah aturan atau tata bahasa yang digunakan dalam penulisan kode program.
- 6) Pengulangan: Pengulangan adalah proses mengulang sebuah blok kode untuk menjalankan tugas yang sama berkali-kali.
- 7) Percabangan: Percabangan adalah proses mengevaluasi sebuah kondisi dan menjalankan blok kode tertentu tergantung pada hasil.
- 8) *Input*, Proses, dan *Output*: *Input* merupakan proses memasukan data ke dalam komputer, proses adalah kumpulan proses yang diproses, dan output adalah informasi yang ditampilkan.

Dengan membahas dasar-dasar pemrograman, para peserta magang memiliki kesempatan untuk menunjukkan pemahaman tentang konsep-konsep fundamental dalam dunia pemrograman. Selain itu, proses ini juga memberikan kesempatan bagi peserta magang untuk berlatih kemampuan menyampaikan materi secara efektif kepada orang lain, yang merupakan keterampilan yang penting dalam karir di bidang teknologi informasi.

### **3.3.2 Perancangan Sistem**

Selama proses perancangan sistem dalam proses magang, ada beberapa hal yang wajib diperhatikan untuk memastikan bahwa sistem yang dibuat dapat memenuhi kebutuhan. Berikut adalah beberapa hal yang dilakukan selama proses magang di dalam Digikidz:

- 1) Analisis Kebutuhan: Langkah awal dalam perancangan sistem adalah melakukan analisis kebutuhan secara menyeluruh. Ini melibatkan identifikasi kebutuhan fungsional dan non-fungsional dari sistem yang akan dibangun, serta pemahaman mendalam tentang kebutuhan pengguna dan pemangku kepentingan lainnya. Berdasarkan analisis dari tim pengembangan Digikidz membutuhkan sistem yang dapat membantu untuk mengatur jadwal, dan penghubung antara orang tua dan tim *coach*. Digikidz juga harus dapat mengakses sistem ini dari mana saja dan kapan saja, hal ini disebabkan tim

*coach* Digikidz selain melakukan sesi *coaching* dari *Center*, selain itu *coach* juga harus melakukan sesi *coaching* dari sekolah-sekolah.

- 2) Perencanaan Sistem: Setelah kebutuhan sistem teridentifikasi, langkah selanjutnya adalah merencanakan arsitektur dan struktur keseluruhan sistem. Ini termasuk pemilihan teknologi dan platform yang sesuai, serta merancang antarmuka pengguna yang ramah dan intuitif. Sehingga untuk menjawab permasalahan dari Digikidz, tim dari Digikidz memutuskan untuk membuat sistem berbasis *website*. Keputusan untuk membuat sistem berbasis *website* adalah langkah yang tepat dalam menanggapi permasalahan yang dihadapi oleh Digikidz. Dengan menggunakan platform *website*, Digikidz dapat memberikan solusi yang fleksibel, mudah diakses, dan dapat diakses dari berbagai perangkat dengan koneksi internet

### **3.3.3 Design Database**

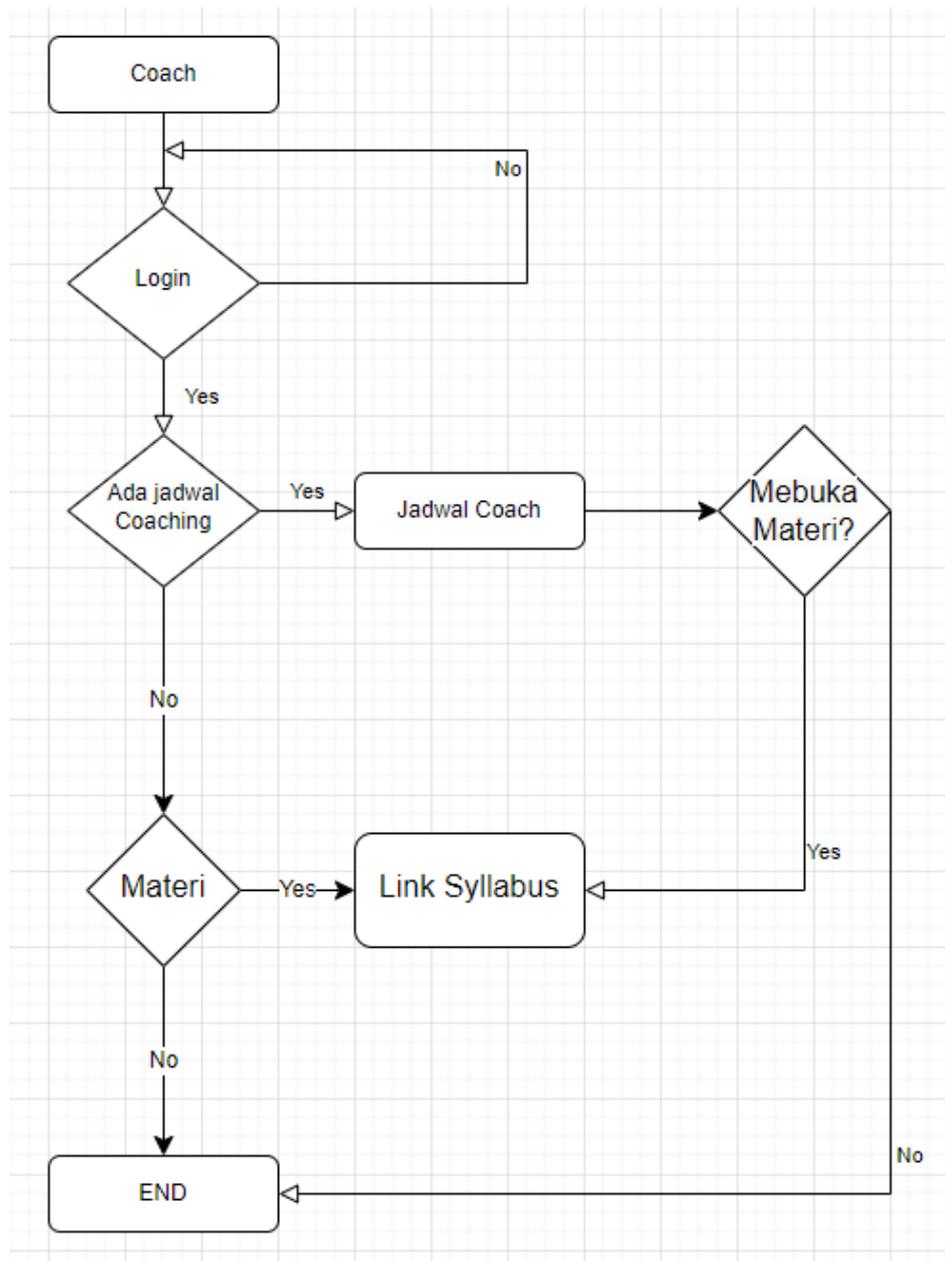
Pada tahap pengembangan sistem Digikidz selama proses magang mendapat tugas untuk mendesign dan mengembangkan *database* untuk sistem Digikidz BSD. Dalam tahap pertama untuk mengembangkan *database* diperlukan untuk menggambar skema terlebih dahulu. Penggambaran skema sebelum membuat *database* memiliki beberapa fungsi penting dalam proses pengembangan sistem. Berikut adalah beberapa fungsi utamanya:

- 1) Visualisasi Struktur Data. Penggambaran skema *database* memungkinkan pengembang untuk secara visual memahami struktur data yang akan disimpan dalam *database*. Dengan melihat diagram skema, pengembang dapat dengan mudah melihat entitas-entitas apa yang ada, atribut apa yang dimiliki oleh setiap entitas, dan hubungan antara entitas.
- 2) Klarifikasi Kebutuhan Bisnis. Proses penggambaran skema *database* memungkinkan tim pengembang untuk mendiskusikan dan merumuskan kebutuhan bisnis dengan lebih jelas. Dengan melihat representasi visual dari skema *database*, tim pengembang dapat dengan mudah mengidentifikasi

kebutuhan data yang spesifik dan menentukan struktur yang tepat untuk memenuhi kebutuhan tersebut.

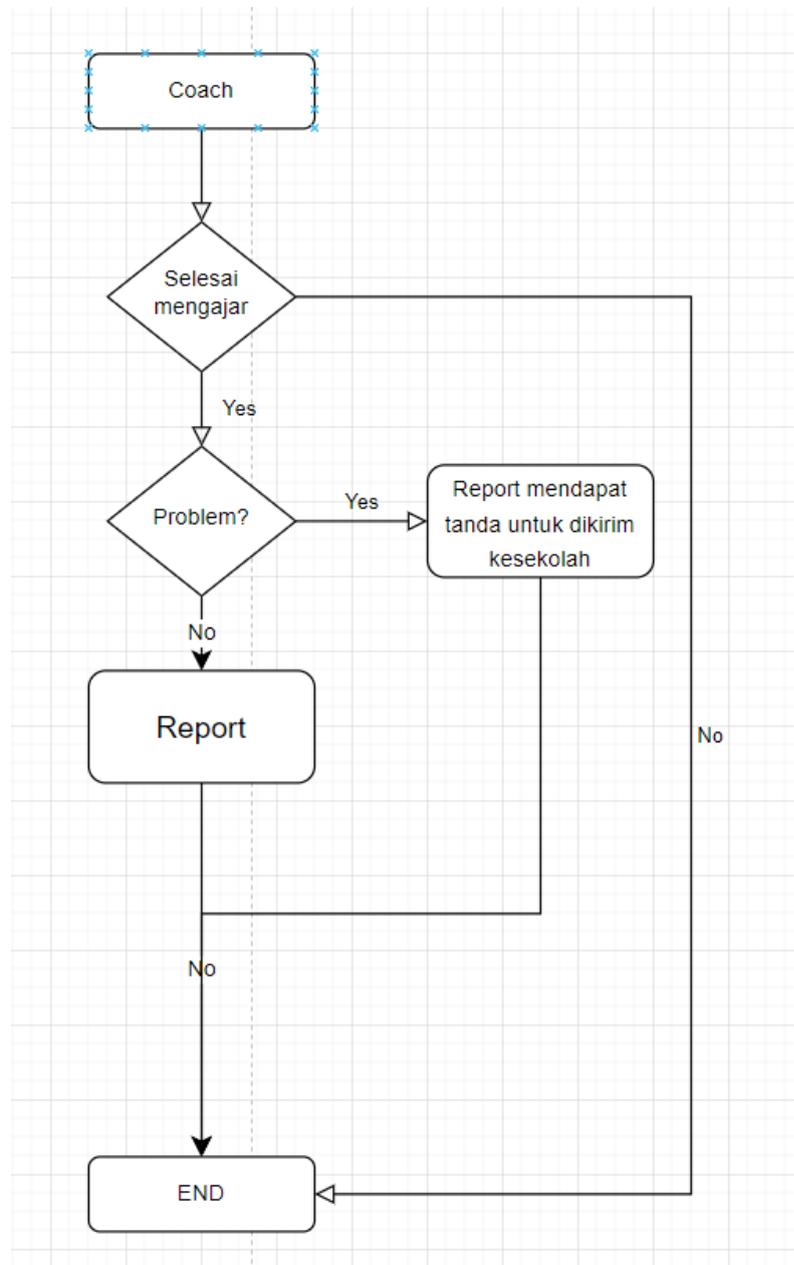
- 3) Deteksi Kesalahan Desain Awal. Melalui penggambaran skema *database*, tim pengembang dapat secara proaktif mendeteksi kesalahan atau kekurangan dalam desain awal. Misalnya, tim pengembang mengidentifikasi hubungan yang tidak tepat antara entitas, atribut yang tidak diperlukan, atau kunci primer yang tidak cocok. Dengan mendeteksi masalah ini sejak awal, tim dapat menghindari kesalahan yang lebih besar dan menghemat waktu serta upaya di tahap selanjutnya.
- 4) Pembagian Tugas. Penggambaran skema *database* juga memungkinkan tim pengembang untuk membagi tugas secara lebih efisien. Setelah skema *database* digambarkan, pengembang dapat mengidentifikasi bagian-bagian yang berbeda dari sistem yang harus dikembangkan dan menetapkan tugas kepada anggota tim yang sesuai dengan keahlian masing-masing.
- 5) Konsultasi dan Persetujuan. Skema *database* dapat digunakan sebagai alat komunikasi antara tim pengembang dan pemangku kepentingan lainnya, seperti manajemen atau pengguna akhir. Dengan menunjukkan diagram skema, tim pengembang dapat menjelaskan struktur data yang akan digunakan dalam sistem kepada pemangku kepentingan dan mendapatkan persetujuan sebelum memulai pengembangan.

Pertama-tama, diperlukan sebuah panel login yang kuat untuk memastikan keamanan *web*, sehingga hanya pihak yang memiliki otorisasi yang dapat mengaksesnya. Keberadaan panel jadwal menjadi esensial untuk memberikan visualisasi yang jelas terhadap jadwal setiap *coach*, memungkinkan perencanaan yang lebih baik dan manajemen waktu yang efektif. Selain itu, keberadaan panel materi menjadi krusial bagi para *coach*, memungkinkan untuk dengan mudah mengakses dan memahami materi yang akan diajarkan. Ini tidak hanya mempermudah persiapan pengajaran, tetapi juga meningkatkan efisiensi dalam manajemen konten pembelajaran.



Gambar 3. 4 Design Diagram sebelum proses *coaching*

Setelah proses *coaching* selesai, semua *coach* diwajibkan untuk membuat progress report, fungsinya adalah agar pihak dari sekolah beserta Digikidz dapat mengetahui apakah terjadi sebuah masalah pada saat proses *coaching* serta membantu orang tua serta pihak sekolah untuk tetap mengetahui sampai mana anak atau murid-murid dari sekolah sudah belajar.



Gambar 3. 5 Diagram Report Kelas

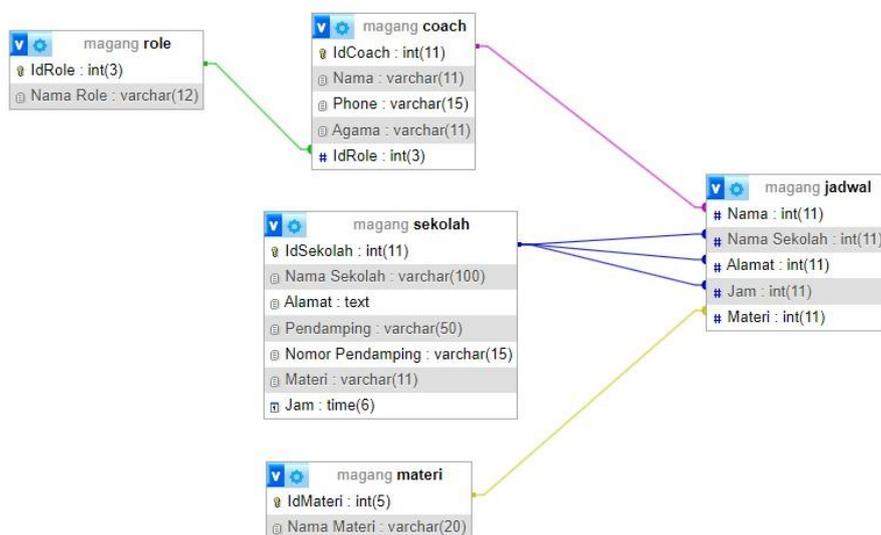
- a) Setelah penggambaran skema selesai maka tahap selanjutnya dalam pengembangan sistem untuk Digikidz BSD adalah membuat design *database*. Desain *database* melibatkan konversi skema yang telah digambar menjadi struktur *database* yang sesungguhnya, termasuk tabel, kolom, kunci

primer, kunci asing, dan indeks. Berikut adalah langkah-langkah yang dapat diambil dalam membuat desain *database*:

- 1) Pemilihan Tipe Data. Setiap atribut dalam skema *database* harus diberi tipe data yang sesuai. Misalnya, atribut nama mungkin memiliki tipe data *VARCHAR*, sedangkan atribut tanggal lahir mungkin memiliki tipe data *DATE*. Pemilihan tipe data yang tepat penting untuk memastikan integritas data dan efisiensi penyimpanan.
- 2) Penentuan Kunci Primer. Setiap tabel dalam *database* harus memiliki satu atau beberapa kolom yang berfungsi sebagai kunci primer. Kunci primer adalah atribut atau kombinasi atribut yang unik mengidentifikasi setiap baris dalam tabel. Penentuan kunci primer yang tepat penting untuk memastikan keunikan setiap entitas dalam *database*.
- 3) Pengaturan Kunci Asing. Setelah kunci primer ditentukan, kunci asing dapat ditambahkan untuk menghubungkan tabel-tabel yang berbeda dalam *database*. Kunci asing adalah atribut yang menunjukkan hubungan antara tabel dengan tabel lainnya. Ini membentuk dasar dari hubungan antar entitas dalam *database*.
- 4) Pembuatan Tabel. Setelah struktur kunci primer dan kunci asing ditentukan, tabel-tabel sekarang dapat dibuat dalam *database*. Setiap entitas dalam skema *database* akan menjadi sebuah tabel, dengan setiap atribut menjadi kolom dalam tabel tersebut.
- 5) Definisi Indeks. Indeks adalah struktur data tambahan yang digunakan untuk mempercepat proses pencarian data dalam *database*. Indeks biasanya dibuat untuk kolom-kolom yang sering digunakan sebagai kriteria pencarian atau pengurutan data. Definisi indeks harus diperhatikan dalam desain *database* untuk meningkatkan kinerja operasi *database*.
- 6) Penentuan Batasan (Constraints). Batasan dapat ditambahkan ke tabel untuk memastikan integritas data dalam *database*. Contohnya, batasan

unik dapat digunakan untuk memastikan bahwa tidak ada nilai duplikat dalam kolom tertentu, sementara batasan ketergantungan referensial (*foreign key constraint*) digunakan untuk memastikan integritas referensial antara tabel.

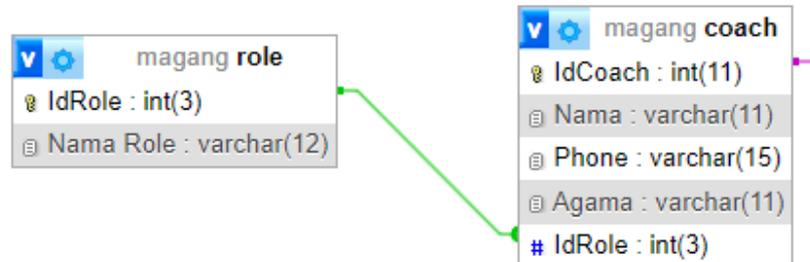
- 7) Optimasi Kinerja. Selama proses desain *database*, perlu dipertimbangkan strategi optimasi kinerja untuk meningkatkan kinerja operasi *database*, seperti pemilihan indeks yang tepat, normalisasi *database*, dan denormalisasi selektif.



Gambar 3. 6 Struktur relasi dari *database* MBKM

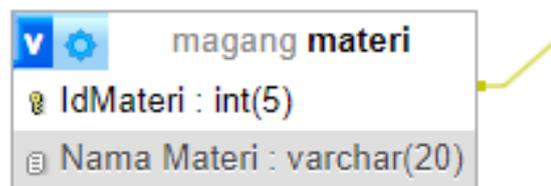
Dalam gambar relasi dari tabel *database* yang telah dirancang, tampak bahwa rancangan ini mencakup beberapa tabel kunci, salah satunya adalah tabel "Role." Tabel ini memiliki peran penting sebagai penanda kedudukan dan pembatas kemampuan pengguna terkait dengan pengelolaan jadwal. Tabel "Role" ini memiliki tiga kategori utama, yaitu Admin yang bertanggung jawab atas pengelolaan *web*, Manajemen yang memiliki kewenangan untuk menambah dan

mengubah jadwal serta materi, dan yang terakhir adalah *coach* yang memiliki hak akses terbatas hanya untuk melihat jadwal dan materi yang akan diajarkannya.



Gambar 3. 7 Tabel database coach

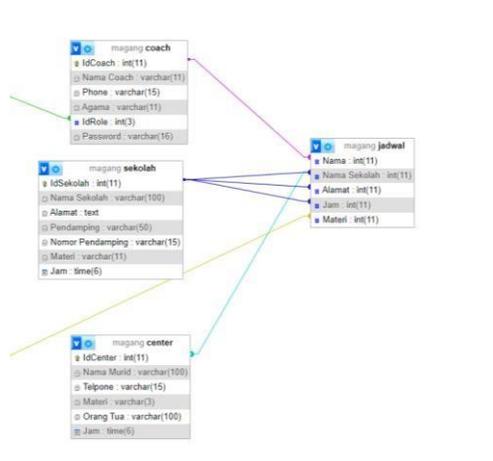
Tabel "*coach*" menyimpan data pribadi dari setiap *coach*, seperti nama, nomor telepon, agama, dan peran (role) yang dimilikinya. Data ini menjadi informasi penting untuk mengidentifikasi *coach* dan mengelola hak akses sesuai dengan perannya dalam proses pengajaran.



Gambar 3. 8 Tabel database Materi

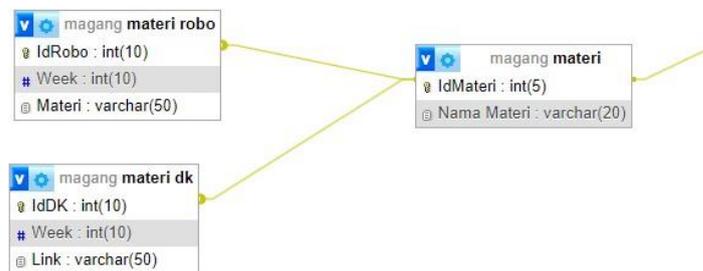
Di samping itu, terdapat tabel "*Materi*" pada Gambar 3.9 yang terdiri dari dua kolom utama, yaitu ID dan nama materi. Tabel ini berfungsi sebagai penyimpanan informasi terkait materi-materi yang akan diajarkan dalam program pelatihan.

Selanjutnya, tabel-tabel tersebut akan dihubungkan ke dalam tabel "*Jadwal*" untuk menyajikan informasi terstruktur dalam sistem *web*. Rancangan ini memastikan integrasi data yang baik antar berbagai elemen, sehingga sistem *web*



Gambar 3. 9 Relasi Tabel Jadwal

dapat memberikan tampilan dan fungsionalitas yang sesuai dengan kebutuhan dan peran masing-masing pengguna. Dengan demikian, gambaran relasi tabel ini memberikan pandangan komprehensif tentang bagaimana data terkait jadwal, materi, dan peran diintegrasikan dalam desain *database*.



Gambar 3. 10 Tabel Materi setelah revisi

Terdapat peningkatan rinci dalam struktur tabel, khususnya pada bagian materi. Penyempurnaan ini disebabkan oleh adanya dua jenis materi yang diajarkan di Digikidz, yaitu Digikidz yang mencakup *Coding*, *3D Design*, *Editing Photo & Video*, dan lainnya, serta Robotkidz yang memberikan pengajaran tentang cara merakit, membuat *script*, dan menjalankan robot. Untuk mengakomodasi perbedaan ini, tabel materi dipecah menjadi dua bagian.

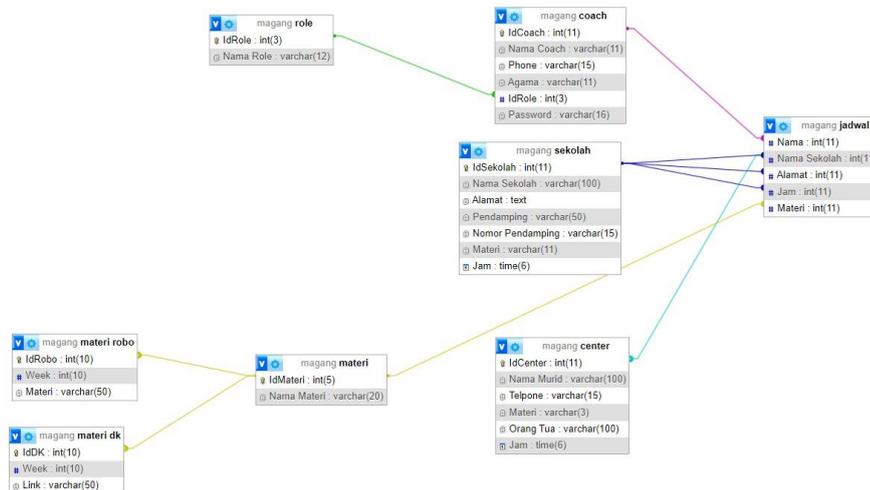
magang center	
IdCenter	int(11)
Nama Murid	varchar(100)
Telpone	varchar(15)
Materi	varchar(3)
Orang Tua	varchar(100)
Jam	time(6)

Gambar 3.11 Tabel Center

Selain itu, penambahan tabel baru juga diperkenalkan, yaitu tabel "Center." Tabel ini berisi data mengenai murid-murid yang mendaftar dalam program *Fun Learning* di cabang Digikidz BSD. Penambahan ini memberikan dimensi lebih lanjut terkait dengan lokasi atau pusat pembelajaran yang dapat diintegrasikan dengan informasi lainnya.

Setelah perancangan tabel selesai, langkah selanjutnya adalah mendesain relasi antar tabel. Dari gambar tabel di bawah, terlihat bahwa terdapat 8 tabel, sehingga diperlukan sebuah hubungan yang menghubungkan kedelapan tabel tersebut. Ada beberapa langkah yang perlu dilakukan terlebih dahulu sebelum menggabungkan tabel-tabel tersebut, yaitu mengidentifikasi kolom dari setiap tabel. Misalnya, pada tabel *coach*, terdapat kolom yang sudah dibuat yang berfungsi sebagai penghubung ke role yang akan dimiliki oleh *coach*, yaitu "IdRole". Kolom "IdRole" ini akan menjadi sebuah *foreign key* yang menghubungkan tabel *coach* dengan tabel *Role*. Foreign key ini memastikan bahwa setiap *coach* memiliki role yang sesuai, dan hubungan antara tabel *coach* dan tabel *Role* dapat terjaga dengan baik. Serupa dengan tabel *coach* dan *Role*, terdapat beberapa kolom tabel lainnya yang telah disediakan untuk berfungsi sebagai penghubung. Kolom-kolom ini diberi awalan "Id" dan diikuti oleh nama tabel yang sesuai, seperti. *Idcoach*, *IdMateri*, *IdSekolah*, dan lainnya. Setiap kolom ini akan menjadi *foreign key* yang menghubungkan tabel terkait, menciptakan relasi yang memungkinkan data untuk saling berinteraksi. Proses identifikasi kolom-kolom yang akan digunakan sebagai *foreign key* adalah

langkah penting dalam mendesain relasi antar tabel. Kolom-kolom ini menentukan bagaimana data dari satu tabel dapat diakses dan diintegrasikan dengan data dari tabel lain.



Gambar 3. 12 Relasi database Jadwal setelah revisi.

Berikutnya, dalam teks tersebut, dilakukan pembuatan tabel menggunakan bahasa *SQL* dengan menambahkan beberapa data dummy. Hal ini bertujuan untuk memastikan bahwa struktur tabel yang dirancang sudah sesuai dan dapat berfungsi dengan baik. Implementasi ini memberikan gambaran lebih lanjut tentang bagaimana data riil nantinya akan disimpan dan diakses dalam *database* Digikidz. Dengan demikian, gambaran ini menggambarkan evolusi desain *database*, mengakomodasi kompleksitas materi yang diajarkan dan menambahkan dimensi baru dalam hal data murid yang terdaftar di pusat pembelajaran tertentu.

```

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";
/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;
--
-- Database: `magang`
--
--
-- Table structure for table `center`
--
CREATE TABLE `center` (
  `IdCenter` int(11) NOT NULL,
  `Nama Murid` varchar(100) NOT NULL,
  `Telpone` varchar(15) NOT NULL,
  `Materi` varchar(3) NOT NULL,
  `Orang Tua` varchar(100) NOT NULL,
  `Jam` time(6) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
--
-- Table structure for table `coach`
--
CREATE TABLE `coach` (
  `Idcoach` int(11) NOT NULL,
  `Nama coach` varchar(11) NOT NULL,
  `Phone` varchar(15) NOT NULL,
  `Agama` varchar(11) NOT NULL,
  `IdRole` int(3) NOT NULL,
  `Password` varchar(16) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
--
-- Table structure for table `coach`
--

```

```

CREATE TABLE `coach` (
  `IdCoach` int(11) NOT NULL,
  `Nama Coach` varchar(11) NOT NULL,
  `Phone` varchar(15) NOT NULL,
  `Agama` varchar(11) NOT NULL,
  `IdRole` int(3) NOT NULL,
  `Password` varchar(16) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `coach`
--

INSERT INTO `coach` (`IdCoach`, `Nama Coach`, `Phone`, `Agama`, `IdRole`, `Password`)
VALUES
(1, 'Hendry Avan', '081510007809', 'Buddha', 1, '');

--
-----
--
-- Table structure for table `jadwal`
--

CREATE TABLE `jadwal` (
  `Nama` int(11) NOT NULL,
  `Nama Sekolah` int(11) NOT NULL,
  `Alamat` int(11) NOT NULL,
  `Jam` int(11) NOT NULL,
  `Materi` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-----
--
-- Table structure for table `materi`
--

CREATE TABLE `materi` (
  `IdMateri` int(5) NOT NULL,
  `Nama Materi` varchar(20) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

```

```
--
-- Dumping data for table `materi`
--
INSERT INTO `materi` (`IdMateri`, `Nama Materi`) VALUES
(1, 'DK'),
(2, 'Robo');

--
-- Table structure for table `materi dk`
--
CREATE TABLE `materi dk` (
  `IdDK` int(10) NOT NULL,
  `Week` int(10) NOT NULL,
  `Link` varchar(50) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Table structure for table `materi robo`
--
CREATE TABLE `materi robo` (
  `IdRobo` int(10) NOT NULL,
  `Week` int(10) NOT NULL,
  `Materi` varchar(50) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Table structure for table `role`
--
CREATE TABLE `role` (
  `IdRole` int(3) NOT NULL,
  `Nama Role` varchar(12) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Table structure for table `role`
--
```

```

CREATE TABLE `role` (
  `IdRole` int(3) NOT NULL,
  `Nama Role` varchar(12) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `role`
--

INSERT INTO `role` (`IdRole`, `Nama Role`) VALUES
(1, 'Admin'),
(2, 'Manejer'),
(3, 'Coach'),
(4, 'Murid');

-----

--
-- Table structure for table `sekolah`
--

CREATE TABLE `sekolah` (
  `IdSekolah` int(11) NOT NULL,
  `Nama Sekolah` varchar(100) NOT NULL,
  `Alamat` text NOT NULL,
  `Pendamping` varchar(50) NOT NULL,
  `Nomor Pendamping` varchar(15) NOT NULL,
  `Materi` varchar(11) NOT NULL,
  `Jam` time(6) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `sekolah`
--

INSERT INTO `sekolah` (`IdSekolah`, `Nama Sekolah`, `Alamat`, `Pendamping`, `Nomor
Pendamping`, `Materi`, `Jam`) VALUES
(1, 'Stella Maris', 'Kencana Loka', 'Ibu Sisil ', '081212345678', 'DK',
'13:00:00.000000');

--
-- Indexes for dumped tables
--

```

```

--
-- Indexes for table `center`
--
ALTER TABLE `center`
  ADD PRIMARY KEY (`IdCenter`);

--
-- Indexes for table `coach`
--
ALTER TABLE `coach`
  ADD PRIMARY KEY (`IdCoach`),
  ADD KEY `IdRole` (`IdRole`);

--
-- Indexes for table `jadwal`
--
ALTER TABLE `jadwal`
  ADD KEY `Alamat` (`Alamat`),
  ADD KEY `Jam` (`Jam`),
  ADD KEY `Materi` (`Materi`),
  ADD KEY `Nama` (`Nama`),
  ADD KEY `Nama Sekolah` (`Nama Sekolah`);

--
-- Indexes for table `materi`
--
ALTER TABLE `materi`
  ADD PRIMARY KEY (`IdMateri`);

--
-- Indexes for table `materi dk`
--
ALTER TABLE `materi dk`
  ADD PRIMARY KEY (`IdDK`);

--
-- Indexes for table `materi robo`
--
ALTER TABLE `materi robo`
  ADD PRIMARY KEY (`IdRobo`);

```

```

--
-- Indexes for table `role`
--
ALTER TABLE `role`
  ADD PRIMARY KEY (`IdRole`);

--
-- Indexes for table `sekolah`
--
ALTER TABLE `sekolah`
  ADD PRIMARY KEY (`IdSekolah`);

--
-- AUTO_INCREMENT for dumped tables
--
--
-- AUTO_INCREMENT for table `center`
--
ALTER TABLE `center`
  MODIFY `IdCenter` int(11) NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `coach`
--
ALTER TABLE `coach`
  MODIFY `IdCoach` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;

--
-- AUTO_INCREMENT for table `materi`
--
ALTER TABLE `materi`
  MODIFY `IdMateri` int(5) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=3;

--
-- AUTO_INCREMENT for table `materi dk`
--
ALTER TABLE `materi dk`
  MODIFY `IdDK` int(10) NOT NULL AUTO_INCREMENT;

```

```

--
-- AUTO_INCREMENT for table `materi robo`
--
ALTER TABLE `materi robo`
  MODIFY `IdRobo` int(10) NOT NULL AUTO_INCREMENT;
--
-- AUTO_INCREMENT for table `role`
--
ALTER TABLE `role`
  MODIFY `IdRole` int(3) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=5;
--
-- AUTO_INCREMENT for table `sekolah`
--
ALTER TABLE `sekolah`
  MODIFY `IdSekolah` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;
--
-- Constraints for dumped tables
--
--
-- Constraints for table `center`
--
ALTER TABLE `center`
  ADD CONSTRAINT `center_ibfk_1` FOREIGN KEY (`IdCenter`) REFERENCES `jadwal` (`Nama
Sekolah`);
--
-- Constraints for table `coach`
--
ALTER TABLE `coach`
  ADD CONSTRAINT `coach_ibfk_1` FOREIGN KEY (`IdRole`) REFERENCES `role` (`IdRole`);
--
-- Constraints for table `jadwal`
--

```

```

ALTER TABLE `jadwal`
  ADD CONSTRAINT `jadwal_ibfk_1` FOREIGN KEY (`Alamat`) REFERENCES `sekolah`
(`IdSekolah`),
  ADD CONSTRAINT `jadwal_ibfk_2` FOREIGN KEY (`Jam`) REFERENCES `sekolah`
(`IdSekolah`),
  ADD CONSTRAINT `jadwal_ibfk_3` FOREIGN KEY (`Materi`) REFERENCES `materi`
(`IdMateri`),
  ADD CONSTRAINT `jadwal_ibfk_4` FOREIGN KEY (`Nama`) REFERENCES `coach` (`IdCoach`),
  ADD CONSTRAINT `jadwal_ibfk_5` FOREIGN KEY (`Nama Sekolah`) REFERENCES `sekolah`
(`IdSekolah`);

--
-- Constraints for table `materi dk`
--
ALTER TABLE `materi dk`
  ADD CONSTRAINT `materi dk_ibfk_1` FOREIGN KEY (`IdDK`) REFERENCES `materi`
(`IdMateri`);

--
-- Constraints for table `materi robo`
--
ALTER TABLE `materi robo`
  ADD CONSTRAINT `materi robo_ibfk_1` FOREIGN KEY (`IdRobo`) REFERENCES `materi`
(`IdMateri`);
COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

```

Gambar 3. 13 Coding SQL Database

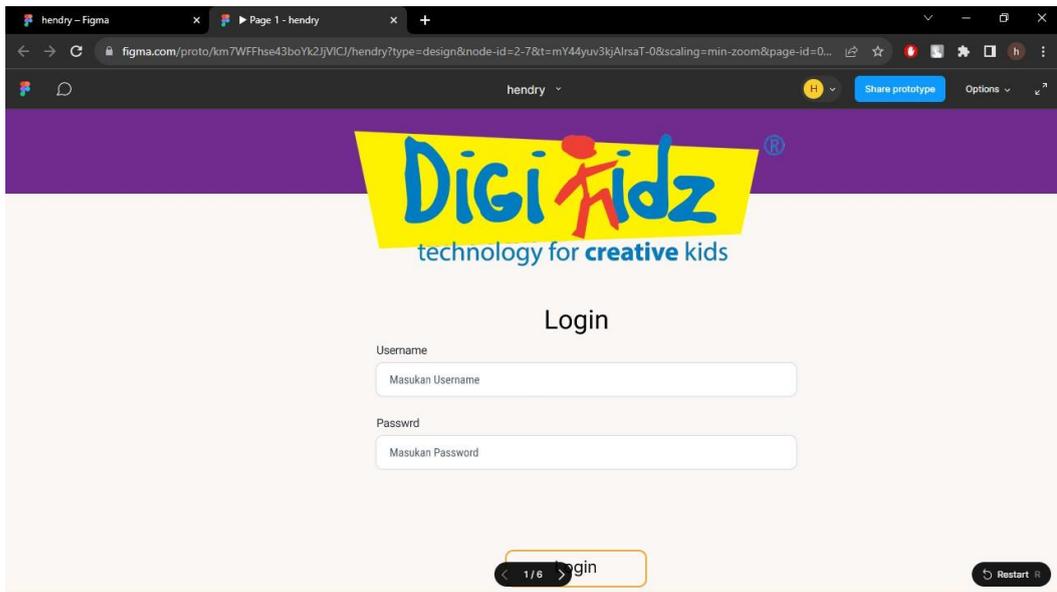
### 3.3.4 Design Web

Setelah selesai mengembangkan *database*, proses selanjutnya adalah mendesign *website* yang akan digunakan sebagai interface dari sistem. Dalam mendesain *website* sebagai antarmuka untuk sistem yang telah dikembangkan, ada beberapa hal dasar yang perlu diperhatikan terlebih dahulu untuk memastikan bahwa *website* tersebut dapat memenuhi kebutuhan pengguna dengan baik. Berikut adalah beberapa hal yang perlu dipertimbangkan.

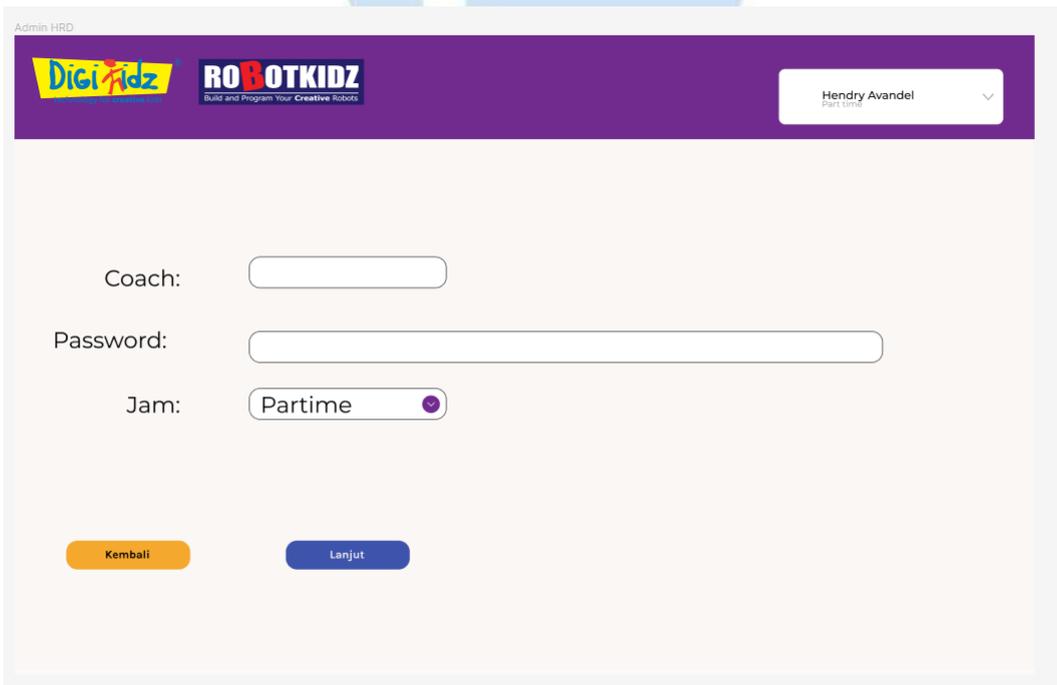
- 1) **Target Pengguna.** Mengidentifikasi target pengguna *website*. Dalam kasus ini adalah para *coach*, murid, dan administrator. Sehingga sangat diperlukan untuk memahami secara baik profil dan kebutuhan para pengguna yang dimana akan membantu dalam merancang *interface* yang sesuai dan mudah digunakan.
- 2) **User Experience (UX).** Merupakan pengalaman pengguna (user experience) dalam merancang antarmuka *website*. Perlu dipastikan bahwa navigasi, tata letak, dan elemen-elemen desain lainnya dirancang dengan baik untuk memastikan pengguna dapat dengan mudah menavigasi dan menggunakan *website* dengan lancar, dan tidak membingungkan para pengguna.
- 3) **Responsiveness.** *Website* yang harus dirancang responsif, artinya dapat menyesuaikan tata letak dan tampilan sesuai dengan berbagai ukuran layar dan perangkat yang digunakan oleh pengguna. Ini penting untuk memastikan pengalaman pengguna yang konsisten dan optimal, baik dari komputer desktop maupun perangkat mobile. Ditambah alat untuk mengakses dari para *coach* masing-masing juga berbeda-beda.
- 4) **Konsistensi Desain.** Pertahankan konsistensi dalam desain antarmuka *website*, termasuk pemilihan warna, tipografi, dan elemen-elemen desain lainnya. Konsistensi ini membantu menciptakan identitas merek yang kuat dan memudahkan pengguna untuk berorientasi di dalam *website*.
- 5) **Accessibility.** *Website* dirancang untuk dapat diakses oleh semua pengguna, termasuk orang yang memiliki keterbatasan dalam menggunakan perangkat elektronik. Ini disebabkan oleh banyaknya orangtua yang ingin mengakses untuk melihat perkembangan dari anaknya.

Pada Gambar 3.14 yang bernama Halaman Login, merupakan halaman pertama yang akan di akses oleh para *coach*. Para *coach* akan diminta untuk memasukan *Username* yang berasal dari *code IdRole*, dan *Idcoach* masing-masing. Kemudian akan dilakukan proses verifikasi oleh sistem apakah *Username* dan *Password* yang dimasukan sudah benar. Pada Gambar 3.16 merupakan halaman

kedua yang akan dilihat dari para *coach*. Pada halaman ini *coach* dapat melihat jadwal selama seminggu kedepan. Data yang akan ditampilkan merupakan hasil data yang diambil dari inputan Admin. Pada Halaman Materi dalam Gambar 3.17 merupakan deskripsi tambahan dari apa yang akan dilihat dari *coach* jika membuka tampilan tabel pada Jadwal. Terdapat beberapa Informasi tambahan yaitu link Materi, dan jenis materi yang akan diajarkan pada *coaching* nanti. Pada Halaman bernama Admin Materi dalam Gambar 3.19, dimana hanya dapat diakses oleh *coach* yang memiliki *Role* Admin. Hal ini berfungsi untuk menjaga para *coach* lainnya untuk mengganti-ganti Link atau nama Sekolah. Dalam hal ini Admin dapat memasukan nama Sekolah dan Link dari Materi, yang nanti akan ditampilkan pada Halaman Materi. Pada Halaman Admin Jadwal dalam Gambar 3.11, dimana hanya dapat diakses oleh *coach* yang memiliki *Role* Admin. Hal ini berfungsi untuk menjaga para *coach* lainnya untuk mengganti-ganti Link atau nama Sekolah. Dalam hal ini Admin dapat memilih sekolah, hari mengajar, nama dari *coach*, jam mulai, dan jam selesai. Setelah selesai menginput sistem akan menyimpannya dan mengirimkan pada Hal Jadwal untuk ditampilkan pada semua *coach*. Pada Halaman Admin HRD, dimana hanya dapat diakses oleh *coach* yang memiliki *Role* Admin. Hal ini berfungsi untuk menjaga para *coach* lainnya untuk mengganti-ganti *Password* dan *Username* dari *coach*. Dalam hal ini Admin dapat mengubah password, dan username dari *coach*.



Gambar 3. 14 Halaman Login



Gambar 3. 15 Admin HRD

Admin Jadwal

**DigiKidz** **ROBOTKIDZ**  
Build and Program Your Creative Kidz

Hendry Avandel

Tempat: Penabur GS

Hari: Senin

Coach: Adit

Jam Mulai: 10:30

Jam Selesai: 14:30

Kembali Lanjut

Gambar 3. 17 Admin Jadwal

Materi

**DigiKidz** **ROBOTKIDZ**  
Build and Program Your Creative Kidz

Hendry Avandel

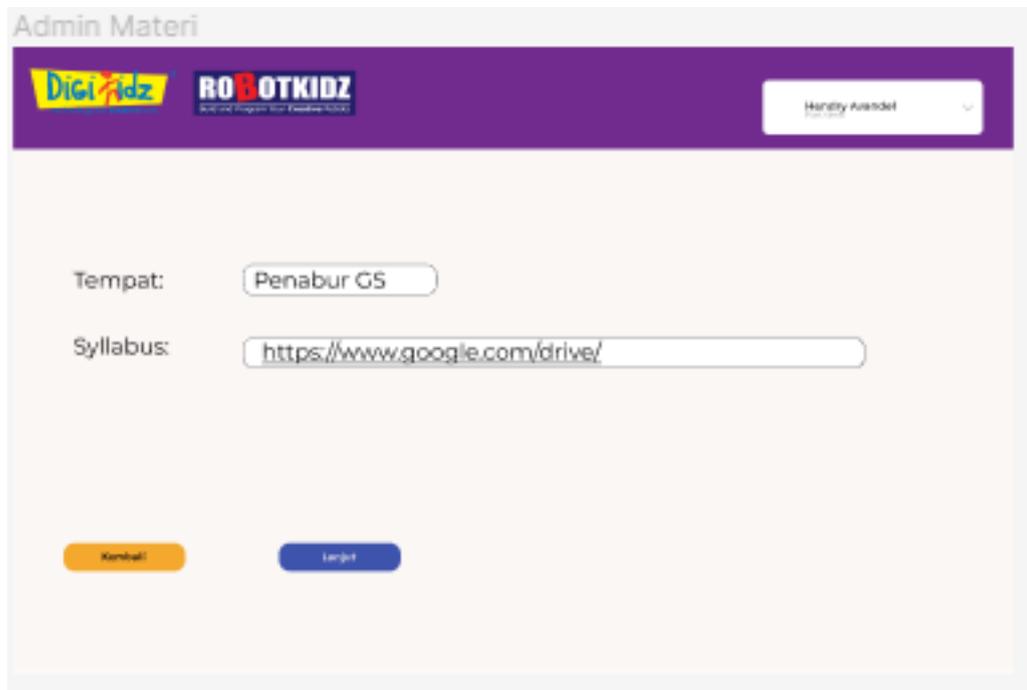
Digikidz RobotKidz

Penabur Modernland Coding

Syllabus: <https://www.google.com/drive/>

Jam: 11.00 - 12.00

Gambar 3. 16 Halaman Materi



Gambar 3. 18 Halaman Admin Materi

### 3.4 Kendala yang Ditemukan

Dalam proses melakukan kerja magang sebagai Intern di IT departement di Digikidz BSD, berikut beberapa kendala atau masalah yang dihadapi.

#### 1. Keterbatasan Pengetahuan dan Waktu.

Masalah umum yang sering dihadapi oleh mahasiswa yang baru mengikuti magang di bidang IT untuk pertama kalinya adalah keterbatasan ilmu. Namun, sistem kerja yang memberikan kebebasan kepada anggota intern untuk mencari dan belajar sendiri dapat menjadi sebuah kesempatan untuk mengatasi kendala ini. Namun, waktu menjadi pembayaran atas proses pembelajaran ini sehingga proses pengembangan menjadi terhambat. Selama masa magang, mahasiswa sering kali dihadapkan pada tugas dan proyek yang memerlukan pemahaman mendalam tentang teknologi dan konsep-konsep terkait. Keterbatasan ilmu dan pengalaman membuat mahasiswa merasa tertekan dan cemas, terutama ketika harus menghadapi tantangan-tantangan teknis yang baru. Meskipun demikian,

sistem kerja yang memberikan kebebasan untuk belajar mandiri juga dapat menjadi keuntungan. Dengan kemajuan teknologi dan akses mudah ke sumber daya online, mahasiswa dapat memanfaatkan waktu luang untuk memperdalam pengetahuan melalui kursus online, tutorial, atau sumber belajar lainnya. Proses pembelajaran ini tidak hanya meningkatkan pengetahuan teknis mahasiswa, tetapi juga mengembangkan keterampilan mandiri, ketekunan, dan kemampuan pemecahan masalah. Namun, tidak dapat dipungkiri bahwa waktu yang diperlukan untuk mempelajari materi baru dapat menjadi kendala. Mahasiswa harus mengimbangi antara tugas-tugas magang, tanggung jawab akademis, dan usaha untuk memperluas pengetahuan. Terkadang, hal ini dapat menyebabkan proses pengembangan terhambat atau keterlambatan dalam menyelesaikan proyek-proyek magang.

## 2. Pengurusan Waktu Kerja Operasional.

Ketika perusahaan menerapkan sistem kerja jarak jauh seperti *Work From Anywhere (WFA)*, seperti yang dilakukan di Digikidz BSD, kendala waktu kerja operasional yang tidak selalu sesuai dengan jam kerja yang telah ditetapkan seringkali muncul. Meskipun WFA memberikan fleksibilitas kepada karyawan untuk bekerja dari mana saja, tantangan dalam menjaga disiplin waktu dan mengatur jadwal kerja tetap ada. Fleksibilitas yang diberikan oleh WFA dapat menyebabkan karyawan sulit untuk memisahkan waktu kerja dan waktu pribadi. Tanpa batasan fisik lokasi kerja, para karyawan mungkin cenderung untuk bekerja di luar jam kerja yang telah ditetapkan atau bahkan selama waktu-waktu yang seharusnya dihabiskan bersama keluarga atau untuk kegiatan non-kerja lainnya.

## 3. Masalah Komunikasi.

Dikarenakan alat komunikasi utama yang dipakai selama kerja magang merupakan pesan teks dengan aplikasi *Whatsapp* dan komunikasi berbasis video dengan Zoom, membuat proses komunikasi menjadi tidak maksimal karena keterbatasan dalam menyampaikan informasi secara lengkap dan jelas

melalui pesan teks, serta kesulitan dalam menangkap ekspresi dan nuansa komunikasi yang disampaikan melalui komunikasi video. Selain itu, terkadang terjadi masalah teknis seperti koneksi internet yang tidak stabil atau gangguan pada aplikasi, yang dapat mengganggu kelancaran komunikasi. Kurangnya interaksi langsung juga dapat mengurangi rasa keterlibatan dan keterhubungan antara anggota tim, sehingga mempengaruhi efektivitas kolaborasi dan pemecahan masalah secara tim.

### **3.5 Solusi atas Kendala yang Ditemukan**

Untuk mengatasi kendala-kendala yang dihadapi selama pelaksanaan magang, berikut beberapa solusi yang diharapkan dapat meningkatkan efektivitas dan kelancaran proyek.

#### **1. Keterbatasan Pengetahuan dan Waktu.**

Mahasiswa yang baru mengikuti magang di bidang IT sering kali menghadapi keterbatasan dalam pengetahuan dan pengalaman. Mahasiswa mungkin merasa tertekan dan cemas ketika dihadapkan pada tugas atau proyek yang membutuhkan pemahaman mendalam tentang teknologi. Namun, dalam konteks sistem kerja yang memberikan kebebasan untuk belajar mandiri, ada kesempatan untuk mengatasi kendala ini. Mahasiswa dapat memanfaatkan waktu luang untuk memperdalam pengetahuan melalui berbagai sumber belajar online seperti kursus daring, tutorial, atau sumber daya belajar lainnya. Walaupun waktu yang dibutuhkan untuk mempelajari materi baru bisa menjadi kendala, manajemen waktu yang efektif dan penyesuaian prioritas tugas membantu memaksimalkan efisiensi pembelajaran. Selain itu memanfaatkan mentor atau rekan kerja yang memiliki pengalaman lebih dalam bidang IT. Mentor dapat memberikan wawasan berharga dan bimbingan yang mempercepat proses pembelajaran, dan memotong waktu untuk mencari jawaban.

#### **2. Pengurusan Waktu Kerja Operasional.**

Dalam sistem kerja jarak jauh seperti *Work From Anywhere* (WFA), fleksibilitas yang diberikan kepada karyawan sering kali menimbulkan tantangan dalam menjaga disiplin waktu dan mengatur jadwal kerja yang sesuai. Tanpa batasan fisik lokasi kerja, karyawan bisa cenderung untuk bekerja di luar jam kerja yang telah ditetapkan. Penting untuk menetapkan komunikasi yang jelas mengenai jam kerja yang diharapkan, batasan waktu, dan ekspektasi ketersediaan. Alat manajemen waktu seperti kalender digital atau reminder juga dapat membantu karyawan mengatur jadwal kerja dan memastikan produktivitas tetap terjaga.

### 3. Masalah Komunikasi.

Selama kerja magang, alat komunikasi utama yang digunakan adalah pesan teks dan komunikasi video. Namun, seringkali terjadi kendala dalam menyampaikan informasi secara lengkap melalui pesan teks dan kesulitan dalam menangkap ekspresi melalui komunikasi video. Masalah teknis seperti koneksi internet yang tidak stabil juga dapat mengganggu kelancaran komunikasi. Untuk mengatasi hal ini, penting bagi perusahaan untuk memastikan bahwa semua anggota tim memiliki akses yang memadai ke alat komunikasi dan koneksi internet yang stabil. Pelatihan atau panduan tentang penggunaan alat komunikasi dengan efektif juga membantu meningkatkan kemampuan komunikasi tim. Selain itu, menciptakan kesempatan untuk interaksi langsung melalui pertemuan tatap muka atau kegiatan kolaboratif dapat memperkuat keterhubungan dan keterlibatan anggota tim.