



REPUBLIK INDONESIA
KEMENTERIAN HUKUM DAN HAK ASASI MANUSIA

SURAT PENCATATAN CIPTAAN

Dalam rangka perlindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:

Nomor dan tanggal permohonan : EC00201977238, 21 Oktober 2019

Pencipta

Nama : **Tan Amelia, Bambang Hariadi, Budi Jatmiko, M.J. Dewiyani
Sunarto, Binar Kurnia Prahani, Tri Sagirani, Julianto
Lemantara, Nanda Rizka Maulana**

Alamat : **KMP Pondok Barat RT3 RW2 Pasanggrahan (Pesanggrahan)
Kwanyar, Bangkalan, Jawa Timur, 69163**

Kewarganegaraan : **Indonesia**

Pemegang Hak Cipta

Nama : **Universitas Dinamika**

Alamat : **Jalan Raya Kedung Baruk 98, Surabaya, Jawa Timur, 60298**

Kewarganegaraan : **Indonesia**

Jenis Ciptaan : **Program Komputer**

Judul Ciptaan : **Aplikasi Komputer Molearn V2 Versi Android**

Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia : **26 Agustus 2019, di Surabaya**

Jangka waktu perlindungan : **Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.**

Nomor pencatatan : **000159981**

adalah benar berdasarkan keterangan yang diberikan oleh Pemohon.

Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.

a.n. MENTERI HUKUM DAN HAK ASASI MANUSIA
DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL



Dr. Freddy Harris, S.H., LL.M., ACCS.
NIP. 196611181994031001

URAIAN CIPTAAN PROGRAM KOMPUTER MOLEARN (ANDROID) V2



Penyusun :

Tan Amelia, S.Kom, M.MT
Dr. Bambang Hariadi
Prof. Dr. Budi Jatmiko, M.Pd
Dr. M.J. Dewiyani Sunarto
Dr. Binar Kurnia Prahani, M.Pd
Tri Sagirani, S.Kom, M.MT
Julianto Lemantara, S.Kom, M.Eng
Nanda Rizka Maulana, S.Kom

UNIVERSITAS DINAMIKA

(d/h Institut Bisnis dan Informatika Stikom Surabaya)

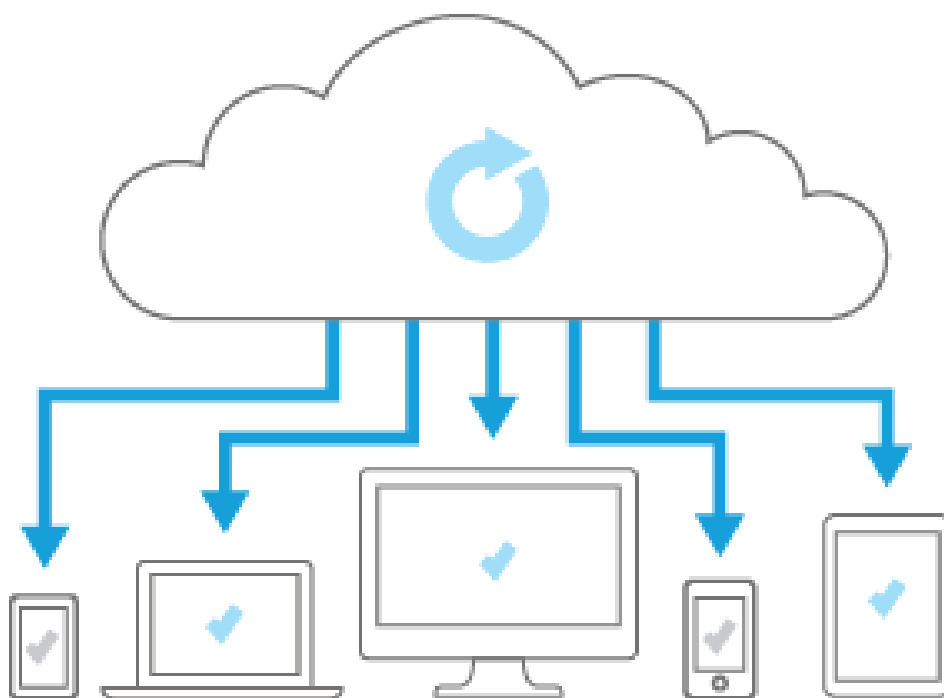
2019

URAIAN CIPTAAN PROGRAM KOMPUTER



A. Gambaran Umum MoLearn

MoLearn atau *Mobile Learning* adalah media yang dibuat untuk menunjang kegiatan belajar mengajar di sekolah. MoLearn yang artinya adalah belajar dimanapun memiliki keunggulan dapat diakses dimanapun melalui media *gadget* (laptop, *smartphone*, tablet, komputer) dengan dukungan koneksi internet. Media pembelajaran ini memfasilitasi interaksi antara guru dan siswa untuk menyediakan dan mendapatkan materi pembelajaran, memberikan tugas dan mengumpulkan tugas, pelatihan mengerjakan ujian online, serta melakukan diskusi. Selain untuk guru dan siswa, MoLearn dibuat untuk MGMP (Musyawarah Guru Mata Pelajaran) untuk mengontrol materi pembelajaran sehingga dengan media ini diharapkan siswa tidak mendapatkan materi yang menyimpang. Gambar 1 dibawah ini merupakan desain arsitektur aplikasi MoLearn:



Gambar 1. desain arsitektur MoLearn

B. Spesifikasi Teknis MoLearn

MoLearn adalah aplikasi berbasis **Web dan Android** dengan kebutuhan sistem sebagai berikut:

1. Android memiliki spesifikasi *software* sistem operasi minimum yaitu ICS (Ice Cream Sandwich).
2. Android telah memiliki *Google Play Service* (ditandai dengan adanya *Google Play Store* dan aplikasi *Google* lainnya).
3. Spesifikasi *minimum hardware* pada Android yaitu RAM 1 GB CPU dan *Processor Intel / Snap Dragon*.

C. Android MoLearn v2

Ada beberapa hal yang diubah/ditambahkan pada Android MoLearn v2 ini, yaitu:

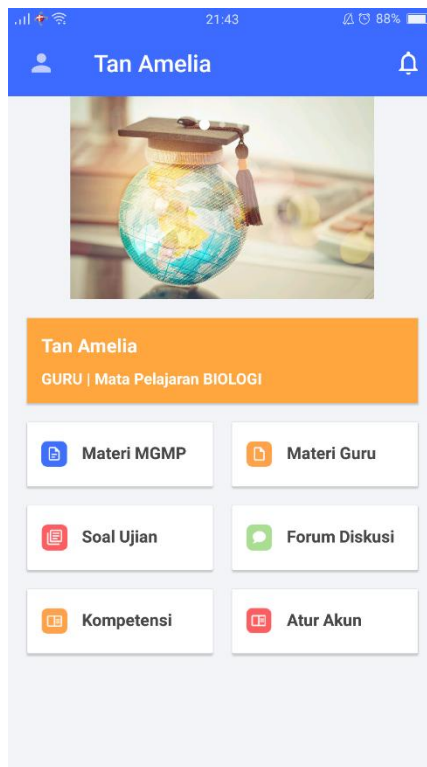
1. Perubahan tampilan UI/UX agar lebih mudah melakukan proses yang dibutuhkan, tampilan lebih menarik dan kenyamanan dalam menggunakannya.
2. Fitur siswa dapat melakukan ujian melalui perangkat Android.
3. Fitur siswa dapat mengumpulkan tugas melalui perangkat Android.

D. Android MoLearn Keseluruhan

Android MoLearn terbagi menjadi 3 bagian yaitu MGMP (administrator MGMP), guru, dan siswa. Masing-masing bagian memiliki menu yang berbeda sesuai dengan data dan informasi yang dibutuhkan serta apa saja yang bisa dilakukan pada android MoLearn.

1. MGMP (Versi Mobile)

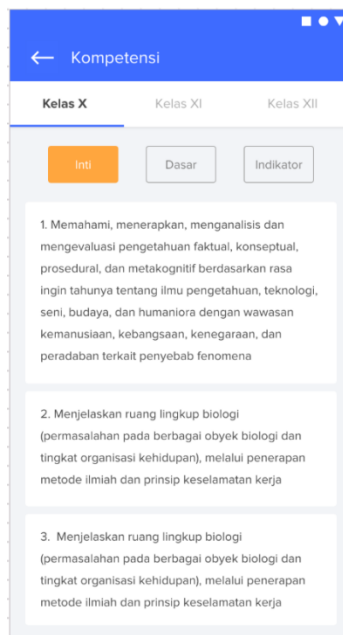
Aplikasi MoLearn berbasis Android hanya memiliki sebagian fitur yang ada pada versi website. Aplikasi MoLearn versi Android sebagai pengguna MGMP memiliki fitur untuk menampilkan materi kompetensi, materi MGMP, dan materi yang diunggah guru, menampilkan ujian, forum diskusi yang memiliki channel, dan pengumuman.



Gambar 2. Menu Utama MGMP

1.1. Materi Kompetensi

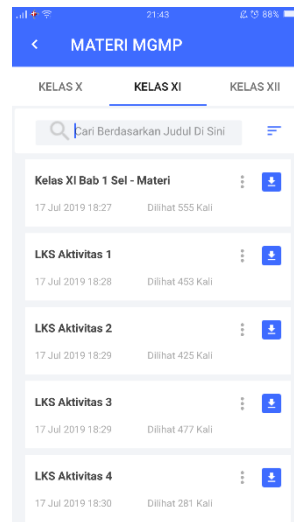
Materi kompetensi digunakan untuk menampilkan kompetensi yang telah di-entry-kan oleh mgmp. Semua kompetensi mulai dari kompetensi dasar, kompetensi inti hingga indikator ada pada menu ini.



Gambar 3. Kompetensi Dasar

1.2. Materi MGMP

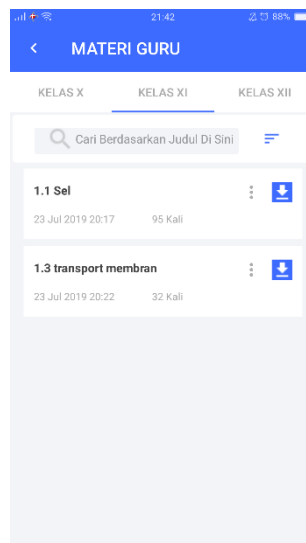
Materi MGMP adalah menu yang berisikan materi-materi yang telah diunggah melalui website molearn.net. Materi yang ditampilkan terbagi menjadi tiga macam yaitu materi dalam bentuk video, dalam bentuk dokumen yang dapat diunduh, dan materi dalam bentuk ringkasan materi yang dapat langsung dibaca pada aplikasi [molearn](http://molearn.net).



Gambar 4. Materi MGMP

1.3. Materi yang diunggah Guru

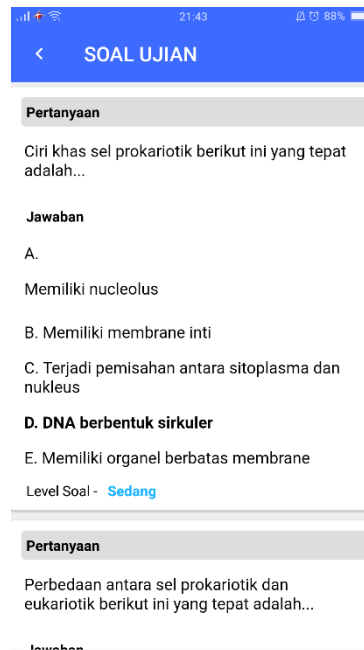
Materi yang diunggah guru adalah materi yang telah diunggah oleh guru pada website molearn.net dan dapat dilihat oleh MGMP. Sama seperti halnya dengan materi yang ada pada MGMP, materi guru juga dibagi menjadi tiga jenis.



Gambar 5. Materi Guru

1.4. Menampilkan Ujian

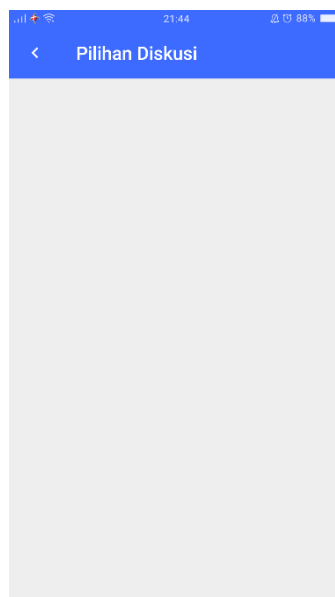
Admin MGMP dapat melihat soal berdasarkan pilihan kompetensi dasar, kompetensi inti, dan indikator. Setelah dipilih maka aplikasi molearn akan menampilkan soal.



Gambar 6. Pilihan Soal

1.5. Forum diskusi

Forum diskusi dapat digunakan oleh tim MGMP untuk saling bertukar informasi dan pendapat melalui sebuah wadah yang terintegrasi juga dengan forum yang ada di website.



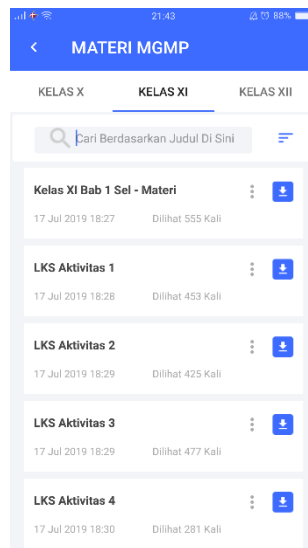
Gambar 7. Forum Diskusi

2. Guru

Aplikasi Molearn versi Android untuk guru memiliki fitur untuk menampilkan materi MGMP, materi guru, buat tugas, diskusi kelas, dan menampilkan soal ujian.

2.1. Menampilkan materi MGMP

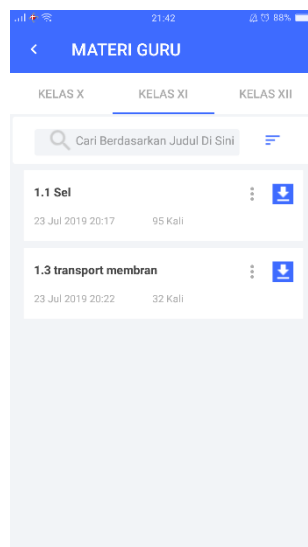
Sama seperti user sebagai MGMP, pengguna sebagai guru juga dapat melihat materi yang diunggah oleh mgmp sebagai referensi pembelajaran.



Gambar 8. Materi MGMP

2.2. Materi Guru

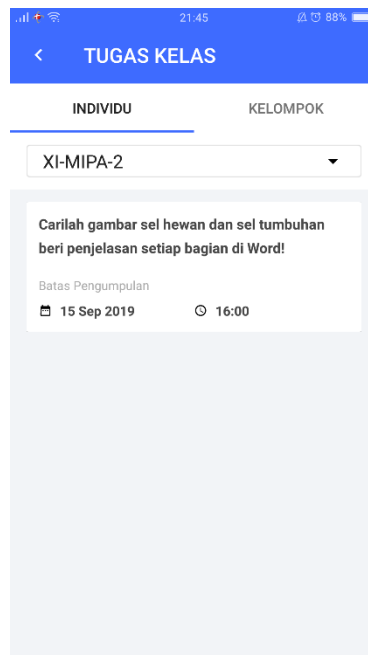
Materi guru ini adalah materi yang diunggah oleh guru itu sendiri. Materi guru sama seperti materi yang lainnya yang memiliki tiga bentuk materi yaitu materi dalam bentuk video, materi dalam bentuk dokumen, dan materi dalam bentuk tulisan.



Gambar 9. Materi Guru

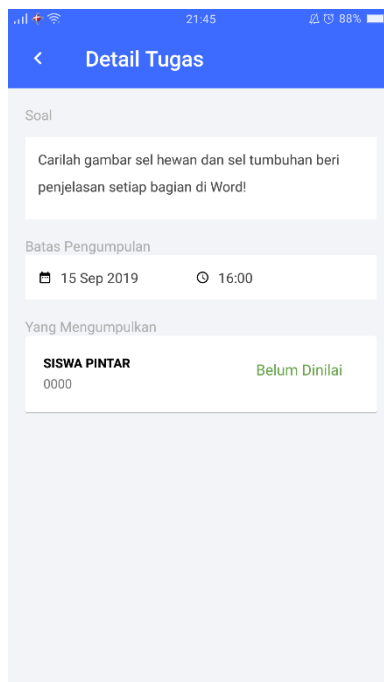
2.3. Buat Tugas

Sebagai guru dapat langsung membuat tugas melalui Aplikasi Android Molearn. Fitur ini digunakan oleh guru untuk memberikan tugas kepada para siswa di kelas ajar.



Gambar 10. Buat tugas

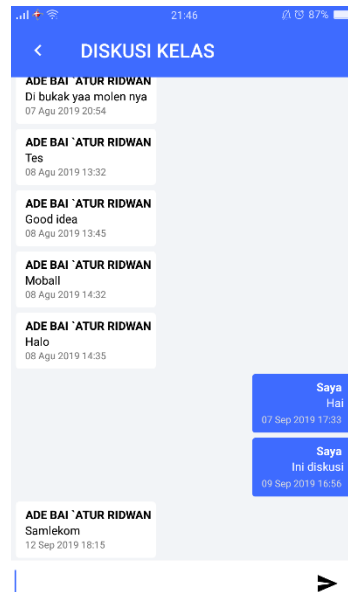
Selain membuat tugas, guru juga dapat melihat tugas yang pernah diberikan pada kelas yang diajar.



Gambar 11. Daftar tugas yang diberikan kepada siswa

2.4. Diskusi kelas

Diskusi yang dilakukan oleh guru dibagi menjadi dua bagian yaitu diskusi kelas dan diskusi bersama MGMP. Fitur diskusi ini adalah fitur yang digunakan untuk saling berkomunikasi dengan siswa ataupun mgmp melalui pesan tulisan.



Gambar 12. Diskusi Kelas

2.5. Melihat Jawaban Ujian Siswa

Guru dapat melihat jawaban ujian siswa



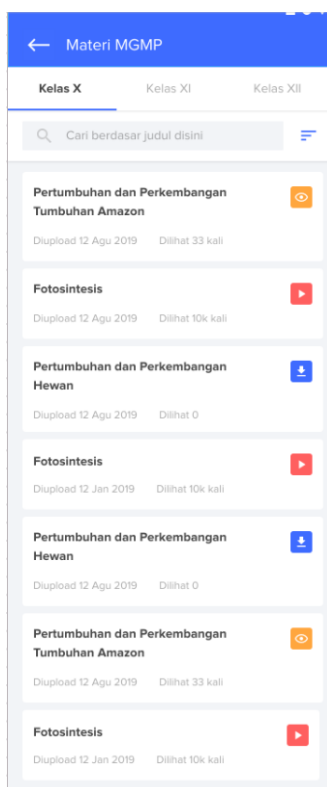
Gambar 13. Jawaban Ujian Siswa

3. Siswa (Versi Mobile)

Selain pengguna sebagai MGMP dan guru, Aplikasi Molearn Android juga dapat digunakan oleh pengguna sebagai siswa dengan beberapa fitur yaitu menampilkan materi MGMP, menampilkan materi guru, forum diskusi, dan menampilkan daftar tugas yang diberikan oleh guru.

3.1. Menampilkan materi MGMP

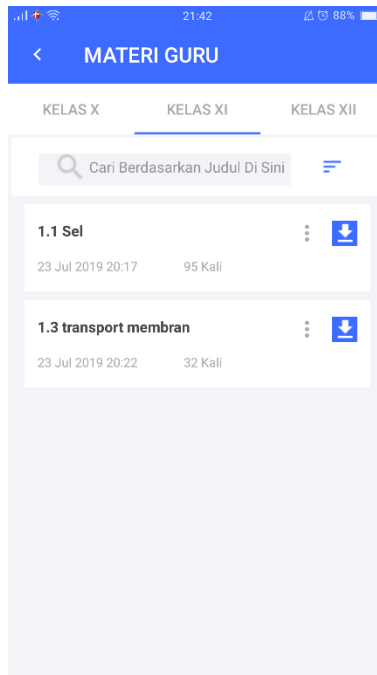
Sama halnya seperti pengguna guru dan MGMP, siswa juga dapat melihat materi yang diberikan oleh MGMP yang terbagi menjadi tiga jenis.



Gambar 14. Materi MGMP yang ada pada siswa

3.2. Menampilkan materi guru

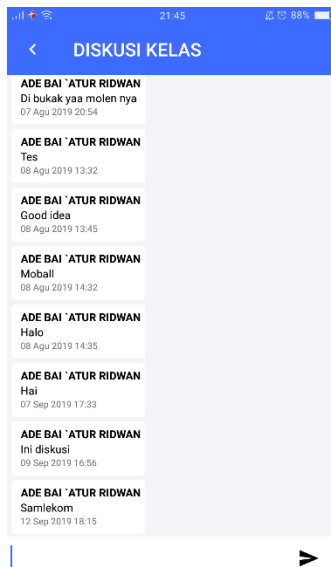
Sama halnya dengan pengguna sebagai guru dan MGMP, siswa juga dapat melihat materi yang diberikan oleh guru sesuai dengan kelas ajarnya.



Gambar 15. Menampilkan materi guru

3.3. Forum Diskusi

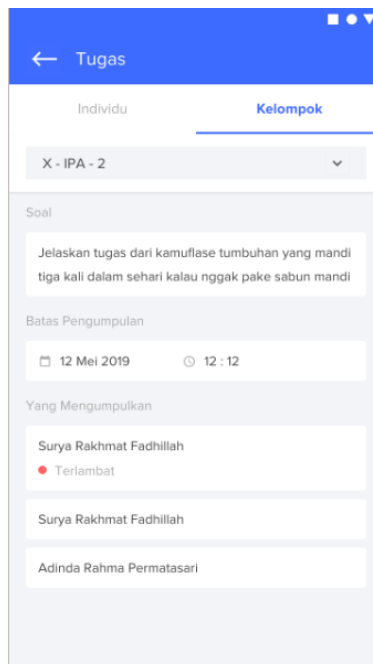
Forum diskusi pada siswa adalah forum untuk saling berdiskusi permasalahan pada mata pelajaran yang berisikan siswa lainnya satu kelas beserta guru ajar.



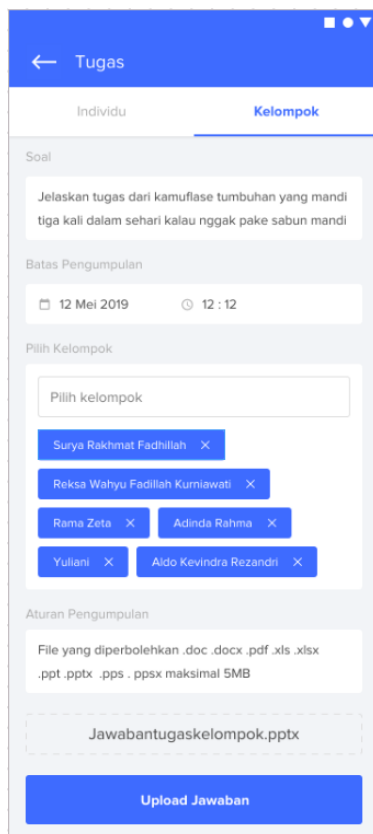
Gambar 16. Diskusi Kelas

3.4. Menampilkan daftar tugas yang diberikan oleh guru

Fitur untuk melihat tugas yang diberikan oleh guru ajar. Setiap tugas memiliki status yaitu status telah mengumpulkan dan belum mengumpulkan.



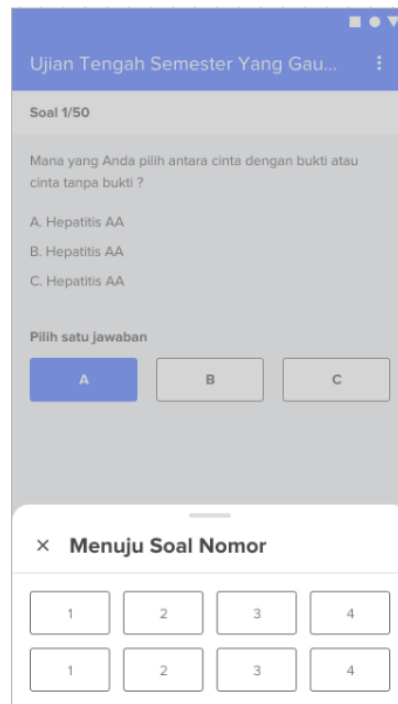
Gambar 17. Daftar Tugas



Gambar 18. Siswa Mengumpulkan Tugas

3.5. Ujian Siswa

Fitur siswa melakukan ujian



Gambar 19. Ujian Siswa

E. SOURCE CODE

1. Kompetensi

Source code di bawah digunakan untuk melakukan request data kompetensi pada server untuk ditampilkan pada aplikasi.

```
StringRequest stringRequest = new StringRequest(Request.Method.GET,
    helperLink.END_POINT_GET_STANDAR_KOMPETENSI
    +"?ki_tingkat="+kelas_tingkat+"&mp_id="+mataPelajaranID, new
    Response.Listener<String>() {
        @Override
        public void onResponse(String response) {
            Log.d(TAG, "onResponse: " + response);
            try {
                JSONArray jsonArray = new JSONArray(response);
                if(jsonArray.length() < 0) {
                    progressBarFragment.setVisibility(View.GONE);
                    lv.setVisibility(View.GONE);
                } else {
                    for (int i = 0; i < jsonArray.length(); i++) {
                        JSONObject jsonObject = new
                        JSONObject(jsonArray.get(i).toString());

                        controllerKompetensi.addKompetensi(jsonObject.getString(helperKompetensi.ki_id),
                            jsonObject.getString(helperKompetensi.mp_id),
                            jsonObject.getInt(helperKompetensi.ki_urutan),
                            jsonObject.getString(helperKompetensi.ki_desc),
                            jsonObject.getInt(helperKompetensi.ki_tingkat),
                            "");
                    }
                    customAdapterKompetensiKelas10 customAdapterKompetensi = new
                    customAdapterKompetensiKelas10(getContext().getApplicationContext(),
                    kelas_tingkat, mataPelajaranID, controllerKompetensi);
                    lv.setAdapter(customAdapterKompetensi);
                }
            }
        }
    });
```

```

        progressBarFragment.setVisibility(View.GONE);
        nodata.setVisibility(View.GONE);
        lv.setVisibility(View.VISIBLE);
        lv.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> parent, View view, int
position, long id) {
                Log.e("trace","itemclick kelas 10 inti");
                HashMap<String,String> map = new HashMap<>();

                map.put("ki_id",controllerKompetensi.getArrayListModelKompetensi().get(position).getKi_id());
                map.put("ki_urutan",
String.valueOf(controllerKompetensi.getArrayListModelKompetensi().get(position).getKi_urutan()));
                map.put("first_data",
String.valueOf(controllerKompetensi.getArrayListModelKompetensi().get(position).getKi_urutan())+"."+controllerKompetensi.getArrayListModelKompetensi().get(position).getKi_desc());
                ((KompetensiActivity)
getActivity()).doCommunicate(fragmentKompetensiIntiKelas10.class.getSimpleName(),
fragmentKompetensiDasarKelas10.class.getSimpleName(),
                map);
                //Toast.makeText(c, "I am clicked at kompetensi inti for "
+ controllerKompetensi.getArrayListModelKompetensi().get(position).getKi_id(),
Toast.LENGTH_SHORT).show();
            }
        });
    } catch (JSONException e) {
        e.printStackTrace();
    }
}
}, new Response.ErrorListener() {
    @Override
    public void onErrorResponse(VolleyError error) {
        Log.d(TAG, "onErrorResponse: " + error);
    }
}) {
    @Override
    protected Map<String, String> getParams() throws AuthFailureError {
        Map<String, String> params = new HashMap<>();
        params.put("ki_tingkat", kelas_tingkat);
        params.put("mp_id", mataPelajaranID);
        return params;
    }
};
AppController.getInstance().addToRequestQueue(stringRequest,
"req_standar_kompetensi_kelas10");

```

2. Download File Materi

Source Code di bawah digunakan untuk melakukan proses download materi jika tipe materi merupakan file.

```

if (controllerMateriMgmp.getArrayListModelMateriMgmp().get(position).getMf_tipe() ==
1 && controllerMateriMgmp.getArrayListModelMateriMgmp().get(position).getMf_share()
== 1) {
    new controllerDownloadFile(context, pDialog, getActivity(), new CallbackFile() {
        @Override
        public void openFile(File file, String pathFile) {
            Log.d(TAG, "openFile: pathFile -> " + pathFile);
            Intent intent = new Intent(Intent.ACTION_VIEW);
            String substring3 = pathFile.substring(pathFile.length() - 3,
pathFile.length());
            String substring4 = pathFile.substring(pathFile.length() - 4,
pathFile.length());
            if (substring3.equalsIgnoreCase("doc") ||

```

```

        substring4.equalsIgnoreCase("docx") ||
        substring3.equalsIgnoreCase("ppt") ||
        substring4.equalsIgnoreCase("pptx")) {
            intent.setDataAndType(FileProvider.getUriForFile(context,
"edu.stikom.molearn.util.GenericFileProvider", file), "text/*");
            intent.setFlags(Intent.FLAG_GRANT_READ_URI_PERMISSION);
            startActivity(Intent.createChooser(intent, "Pilih Untuk Membuka
File"));
        } else if (substring3.equalsIgnoreCase("jpg") ||
        substring4.equalsIgnoreCase("jpeg") ||
        substring3.equalsIgnoreCase("png") ||
        substring3.equalsIgnoreCase("bmp")) {
            intent.setDataAndType(FileProvider.getUriForFile(context,
"edu.stikom.molearn.util.GenericFileProvider", file), "image/*");
            intent.setFlags(Intent.FLAG_GRANT_READ_URI_PERMISSION);
            startActivity(Intent.createChooser(intent, "Pilih Untuk Membuka
File"));
        } else if (substring3.equalsIgnoreCase("pdf")) {
            intent.setDataAndType(FileProvider.getUriForFile(context,
"edu.stikom.molearn.util.GenericFileProvider", file), "application/pdf");
            intent.setFlags(Intent.FLAG_GRANT_READ_URI_PERMISSION);
            startActivity(Intent.createChooser(intent, "Pilih Untuk Membuka
File"));
        } else {
            Toast.makeText(getActivity(), "File tidak dapat dibuka karena tidak
ada aplikasi yang membuka", Toast.LENGTH_LONG).show();
        }
    }
    }.execute(helperLink.END_POINT_MATERI_MGMP +
controllerMateriMgmp.getArrayListModelMateriMgmp().get(position).getMf_file(),
controllerMateriMgmp.getArrayListModelMateriMgmp().get(position).getMf_file());
    Log.i("ActionActivity", "File Downloaded...");
}
}

```

3. Menampilkan tugas siswa

Source code di bawah ini merupakan fungsi yang digunakan untuk menampilkan daftar tugas terbaru pada siswa.

```

public void readTugas() {
    progressBar.setVisibility(View.VISIBLE);
    new Handler().postDelayed(new Runnable() {
        @Override
        public void run() {
            cTugasKelas.getTugasKelasVolley(new CallbackTugasKelas() {
                @Override
                public void onSuccess(List<modelTugasKelas> mTugasKelas) {
                    progressBar.setVisibility(View.GONE);
                    Log.d(TAG, "onSuccess: " + mTugasKelas.toString());
                    nodata.setVisibility(View.VISIBLE);
                    if (mTugasKelas.size() > 0) {
                        nodata.setVisibility(View.GONE);
                    }
                    rvTugasKelas = new RecyclerViewTugasKelas(cTugasKelas,
getActivity());
                    rvTugas.setHasFixedSize(true);
                    RecyclerView.LayoutManager mLayoutManager = new
LinearLayoutManager(context);
                    rvTugas.setLayoutManager(mLayoutManager);
                    rvTugas.setItemAnimator(new DefaultItemAnimator());
                    rvTugas.setAdapter(rvTugasKelas);
                    rvTugasKelas.notifyDataSetChanged();
                }
            });
        }
    });
}

```



```

        public void onSuccessResponse(String response) {
            Log.d(TAG, "onSuccessResponse: " + response);
        }

        @Override
        public void onFail(String msg) {
            progressBar.setVisibility(View.GONE);
            Log.d(TAG, "onFail: " + msg);
            nodata.setVisibility(View.VISIBLE);
        }
    }, cSiswa.getSessionSiswa().get(helperSiswa.kp_id),
    cSiswa.getSessionSiswa().get(helperSiswa.siswa_id),
    cMataPelajaran.getSesionMataPelajaran().get(helperMataPelajaran.MP_ID),
    cSiswa.getSessionGuruForSiswa().get(helperGuru.model_guru), view,
    cSiswa.getSessionSiswa().get(helperSiswa.sekolah_id));
    }
}, 500);
}

```

4. Mengirim pesan diskusi

Source code di bawah berfungsi untuk mengirimkan pesan diskusi kelas siswa.

```

if (v == holder.buttonSend) {
    String isi_pesan = holder.editTextIsiPesan.getText().toString();
    if (cSiswa.isLoggedIn()) {
        holder.KP_ID = cSiswa.getSessionSiswa().get(helperSiswa.kp_id);
        holder.SISWA_OR_GURU_ID = cSiswa.getSessionSiswa().get(helperSiswa.siswa_id);
        holder.SISWA_OR_GURU_NAMA =
        cSiswa.getSessionSiswa().get(helperSiswa.siswa_nama);
    } else if (cGuru.isLoggedIn()) {
        holder.KP_ID = intentBefore.getStringExtra(helperKelasPembelajaran.KP_ID);
        holder.SISWA_OR_GURU_ID = cGuru.getSesionGuru().get(helperGuru.guru_id);
        holder.SISWA_OR_GURU_NAMA = cGuru.getSesionGuru().get(helperGuru.guru_nama);
    }
    final controllerChatKelas newChat = new controllerChatKelas();

    LayoutInflater vi = (LayoutInflater)
    getApplicationContext().getSystemService(Context.LAYOUT_INFLATER_SERVICE);
    View view1;
    TextView textViewChat;
    TextView textViewChatName;
    TextView textViewChateDate;

    view1 = vi.inflate(R.layout.view_chat_me, null);
    textViewChat = (TextView) view1.findViewById(R.id.textViewChatMe);
    textViewChatName = (TextView) view1.findViewById(R.id.textViewChatName);
    textViewChateDate = (TextView) view1.findViewById(R.id.textViewChateDate);
    textViewChat.setText(isi_pesan);
    if (cSiswa.isLoggedIn()) {
        textViewChatName.setText(cSiswa.getSessionSiswa().get(helperSiswa.siswa_nama));
    } else if (cGuru.isLoggedIn()) {
        textViewChatName.setText(cGuru.getSesionGuru().get(helperGuru.guru_nama));
    }

    textViewChateDate.setText(helperSembarang.getNewFormatDateTime(helperSembarang.getDat
    eNowFortmat()));
    holder.linearLayout.addView(view1);
    holder.editTextIsiPesan.setText("");
    requestBottomFocus();

    newChat.setChatDiskusiKelasVolley(new CallbackChatKelas() {
        @Override
        public void onSuccess(List<modelChatDiskusiKelas> modelChatDiskusiKelasList)
    {
        Log.d(TAG, "onSuccess: newChat:" +
        modelChatDiskusiKelasList.get(0).getKdiPosting());
    }
}
}

```

```

@Override
public void onSuccessResponse(String response) {
    Log.d(TAG, "onSuccessResponse: newChat JSONObject : " + response);
}

@Override
public void onFail(String msg) {
    Log.d(TAG, "onFail: newChat" + msg);
}
}, holder.KP_ID, isi_pesan, holder.SISWA_OR_GURU_ID, holder.SISWA_OR_GURU_NAMA,
cFirebase.getTokenNumber(),
cMataPelajaran.getSesionMataPelajaran().get(helperMataPelajaran.MP_ID));
}

```

5. Melihat hasil soal ujian

Source code di bawah digunakan untuk melakukan request pada server untuk menampilkan data hasil ujian pada aplikasi.

```

cSoal.getSoalJawabanSiswaVolley(new CallbackSoalJawabanSiswa() {
@Override
public void onSuccess(final List<modelSoal> mSoal) {
    int TIME_OUT = 500;
    tempNilai = 0;
    new Handler().postDelayed(new Runnable() {
@Override
public void run() {
        nilaiKeseluruhan = new ArrayList<>();
        Log.d(TAG, "onSuccess banyak soal: " + mSoal.size());
        RecyclerViewJawabanSoalSiswaNew rvJawabanSoalSiswa = new
RecyclerViewJawabanSoalSiswaNew(cSoal,
cNilaiUjian.getModelKelompokUjianSiswa().getUs_id(), getApplicationContext(), new
CallbackNilai() {
@Override
public void nilaiPerSoal(int nilaiPersoal, String soalID) {
        //Log.d(TAG, "nilaiPerSoal: " + nilaiPersoal);
        Log.d(TAG, "nilaiPerSoal: " + nilaiPersoal + " soal id " +
soalID);

        tempNilai = tempNilai + nilaiPersoal;
        nilaiKeseluruhan.add(new classHelperNilai(nilaiPersoal,
soalID));

        Holder.textViewNilai.setText(String.valueOf(tempNilai));
    }

@Override
public void nilaiEssay(int nilaiPersoal, String soalID) {
        try {
            int temporaryNilai = 0;
            for(int i=0;i<nilaiKeseluruhan.size();i++) {
                Log.d(TAG, "nilaiEssay: posisi soal " +
nilaiKeseluruhan.get(i).getSoalID());
                Log.d(TAG, "nilaiEssay: nilai persoal " +
nilaiKeseluruhan.get(i).getNilaiPersoal());

if (nilaiKeseluruhan.get(i).getSoalID().equals(soalID)) {
nilaiKeseluruhan.get(i).setNilaiPersoal(nilaiPersoal);
                }
            }

            for(classHelperNilai row:nilaiKeseluruhan) {
                temporaryNilai = temporaryNilai +
row.getNilaiPersoal();
            }

Holder.textViewNilai.setText(String.valueOf(temporaryNilai));
        } catch (IndexOutOfBoundsException exx) {

```

```

        Log.d(TAG, "nilaiEssay: nilai masih
IndexOutOfBoundsException" + exx.getMessage());
    }
    });

    Holder.recyclerView.setHasFixedSize(true);
    RecyclerView.LayoutManager layoutManager = new
LinearLayoutManager(getApplicationContext());
    Holder.recyclerView.setLayoutManager(layoutManager);
    Holder.recyclerView.setItemAnimator(new DefaultItemAnimator());
    Holder.recyclerView.setAdapter(rvJawabanSoalSiswa);
    Holder.recyclerView.setOnItemClickListener(new
RecyclerViewTouchListener(getApplicationContext(), Holder.recyclerView, new
ClickListener() {
        @Override
        public void onClick(View view, int position) {

        }

        @Override
        public void onLongClick(View view, int position) {

        }
    }));
    rvJawabanSoalSiswa.notifyDataSetChanged();
    Holder.progreBar.setVisibility(View.GONE);
}
}, TIME_OUT);

}

@Override
public void onSuccessResponse(String response) {
    Log.d(TAG, "onSuccessResponse: " + response);
}

@Override
public void onFail(String msg) {
    Log.d(TAG, "onFail: " + msg);
}
}, cNilaiUjian.getModelKelompokUjianSiswa().getUs_id(),
cNilaiUjian.getModelKelompokUjianSiswa().getKu_id());

```

F. Struktur Database

