

## IMPLEMENTASI PENILAIAN HASIL BELAJAR SECARA DIGITAL BERBASIS APLIKASI E-RAPOR UNTUK DOKUMENTASI AKADEMIK

Pramudya Prima Insan<sup>1\*</sup>, Nasruddin Bin Idris<sup>2</sup>, Muhammad Kharisma Mahardika<sup>3</sup>

Universitas Mulia<sup>123</sup>

E-mail: [pramudyapi@gmail.com](mailto:pramudyapi@gmail.com)

### Abstrak

Rapor adalah nilai belajar siswa yang penting dan berfungsi sebagai komunikasi antara sekolah dan orang tua untuk memahami kemajuan belajar anak. Penilaian oleh pendidik bertujuan memantau kemajuan belajar siswa. Penggunaan aplikasi komputer dapat mempercepat dan memudahkan proses penilaian. Meskipun beberapa guru mungkin memiliki keterbatasan teknologi, hal itu tidak boleh menjadi alasan untuk tidak menggunakan aplikasi rapor. Oleh karena itu, merancang aplikasi untuk dokumentasi akademik siswa secara digital di sekolah diperlukan. Penelitian ini bertujuan untuk memahami fenomena pengalaman subjek penelitian secara holistik melalui implementasi MVT Django Framework. Hasilnya, aplikasi E-Rapor berhasil dibuat dengan pengelolaan data yang sistematis. Struktur MVT dalam Django Framework memisahkan tugas-tugas dengan jelas, dan penggunaan Model, View, Template membantu merancang aplikasi web yang terstruktur, efisien, mudah dipelihara, dan diperbarui. Aplikasi E-Rapor ini memungkinkan penyimpanan data hasil belajar siswa secara elektronik dan dapat mencetak dokumen E-Rapor, memungkinkan sekolah mendokumentasikan hasil belajar siswa secara digital.

**Kata Kunci:** E-Rapor; Framework Django; Model-View-Template; Hasil Belajar

### Abstract

*The report card is an important student learning score and serves as a communication between the school and parents to understand the child's learning progress. Assessment by educators aims to monitor student learning progress. The use of computer applications can speed up and facilitate the assessment process. Although some teachers may have technological limitations, it should not be an excuse not to use report card applications. Therefore, designing an application for digital student academic documentation in schools is necessary. This research aims to understand the phenomenon of the research subject's experience holistically through the implementation of MVT Django Framework. As a result, an E-Report card application was successfully created with systematic data management. The MVT structure in Django Framework separates tasks clearly, and the use of Model, View, Template helps design web applications that are structured, efficient, easy to maintain, and update. This E-Report application enables electronic storage of student learning outcomes data and can print E-Report documents, allowing schools to document student learning outcomes digitally.*

**Keywords:** E-Report Card; Django Framework; Model-View-Template; Learning Outcomes

105

Insan, P. P., Idris, N. B., & Kharisma Mahardika, M. (2024). IMPLEMENTASI PENILAIAN HASIL BELAJAR SECARA DIGITAL BERBASIS APLIKASI E-RAPOR UNTUK DOKUMENTASI AKADEMIK. *Jurnal Inovasi Pendidikan Dan Teknologi Informasi (JIPTI)*, 5(1), 105–113. <https://doi.org/10.52060/jipti.v5i1.1836>

## PENDAHULUAN

Dalam Peraturan Menteri Pendidikan dan Kebudayaan Nomor: 23 Tahun 2016 tentang Standar Penilaian Pendidikan menjelaskan bahwa penilaian pendidikan pada pendidikan dasar dan pendidikan menengah terdiri atas penilaian hasil belajar oleh pendidik, satuan pendidikan, dan pemerintah (Septiani, Aribbe, & Diansyah, 2020)

Penilaian hasil belajar oleh pendidik bertujuan untuk memantau dan mengevaluasi proses, kemajuan belajar, dan perbaikan hasil belajar peserta didik. Proses Penilaian hasil belajar peserta didik, baik oleh pendidik maupun oleh satuan Pendidikan, akan lebih sistematis, komprehensif, lebih akurat, dan cepat dilakukan apabila didukung dengan perangkat aplikasi komputer.

Rapor adalah buku yang berisi nilai kemampuan dan nilai siswa di sekolah yang dapat digunakan sebagai acuan penilaian siswa selama mengikuti pembelajaran di sekolah. Dengan adanya buku rapor guru dan orang tua dapat memantau bagaimana perkembangan siswa selama menerima proses belajar mengajar di sekolah. Pentingnya rapor bagi pendidikan di Indonesia adalah secara tolak ukur seorang siswa sudah berhasil menyelesaikan kegiatan belajarnya di sekolah dalam bentuk penilaian dan fungsi sebagai penunjang siswa guna melanjutkan jenjang yang lebih tinggi.

Mengenai keterbatasan alat elektronik dan kurangnya pengetahuan guru dalam pengetahuan komputer di sekolah bukan berarti mereka menjadikan suatu alasan

untuk tidak menerapkan aplikasi rapor dalam penilaian siswa. Dalam segala bidang pekerjaan pada umumnya mereka telah menggunakan aplikasi dengan dasar elektronik.

Setelah dilakukan penelitian ini diharapkan framework Django memiliki waktu respon yang cepat, dibandingkan dengan penelitian yang telah dilakukan sebelumnya sebagian besar menggunakan framework laravel dan tidak adanya penerapan MVT. Dengan adanya penelitian ini akses para guru maupun peserta didik sekolah lebih cepat. Integrasi pemodelan sistem dengan Model-view-template (MVT) jauh lebih tepat dikarenakan adanya pembuatan data didalam sebuah class dan object dalam sebuah pemrograman. Selain itu sebuah pembaharuan diterapkan dalam penelitian ini dengan tidak adanya sebuah controller dalam pengembangan sistem berbasis website.

Sehingga penelitian ini dinilai perlu dilakukan dan merekomendasikan kepada pihak sekolah untuk membangun aplikasi e-rapor dengan framework Django dan penerapan MVT. Berdasarkan latar belakang di atas, tujuan peneliti untuk mendeskripsikan Efektivitas penggunaan aplikasi rapor digital

## METODE PENELITIAN

### A. Pengumpulan Data

Penelitian kualitatif adalah penelitian yang bermaksud untuk memahami fenomena tentang apa saja yang telah dialami oleh subjek penelitian secara holistik dan dengan

cara deskripsi dalam bentuk kata-kata dan bahasa, pada suatu konteks khusus yang alamiah dan dengan memanfaatkan berbagai metode alamiah.

Metode pengumpulan data yang digunakan oleh penulis dalam penelitian observasi, wawancara dan studi pustaka (Shalahuddin & Sukamto, 2018).

### 1. Observasi

Melakukan observasi langsung terhadap proses penggunaan E-Rapor dalam konteks pengambilan data akademik siswa.

### 2. Wawancara

Melakukan wawancara dengan guru dan staf administrasi untuk mendapatkan pandangan mereka tentang implementasi, manfaat, tantangan, dan pengalaman mereka dalam menggunakannya.

### 3. Studi Pustaka

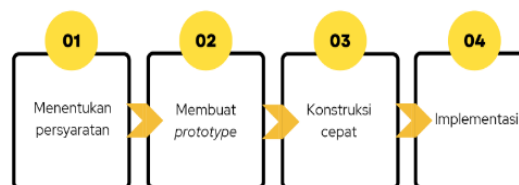
Melakukan studi pustaka dengan membaca buku-buku literatur Framework Django, MVT, dan laporan/jurnal dengan topik penelitian yang sesuai untuk kebutuhan penelitian.

## B. Pengembangan Sistem

Dalam penelitian ini, pengembangan aplikasi menggunakan kerangka pengembangan perangkat lunak. Penerapan sebuah teknologi web cocok untuk integrasi data pada sistem informasi merupakan teknologi yang dapat menghubungkan sistem informasi yang berbeda platform serta database yang berbeda (Idris & Mutiara,

2017). Tahapan pengembangan sistem versi awal dari perangkat lunak yang digunakan untuk mewakili percobaan desain yang bertujuan untuk memudahkan pengembang dan user dalam menemukan solusi perancangan sistem yang sesuai dengan kebutuhan (Agustinus Noertjahyana, 2002).

Model pengembangan *Rapid Application Development* (RAD) dalam langkah pertama dilakukan planning dan analisis kebutuhan dapat membuat desain sistem lebih cepat dengan waktu pencapaian hasil produksi aplikasi tidak kurang dari 30 Hari (Agustinus Noertjahyana, 2002).



Gambar 1. *Rapid Application Development*

Sumber: <https://revou.co/kosakata/rad>

Adapun keterangan pada tahapan metode yang dapat dilihat pada gambar 1 diatas. Berikut penjelasan mengenai tahapan-tahapan metode:

#### 1. Planning

Tahap awal dalam suatu pengembangan sistem, dimana pada tahap ini dilakukan identifikasi masalah dan pengumpulan data.

#### 2. Prototyping

Tahap perancangan desain sistem dilakukan berdasarkan hasil analisis pengguna yang terlibat dalam pengelolaan

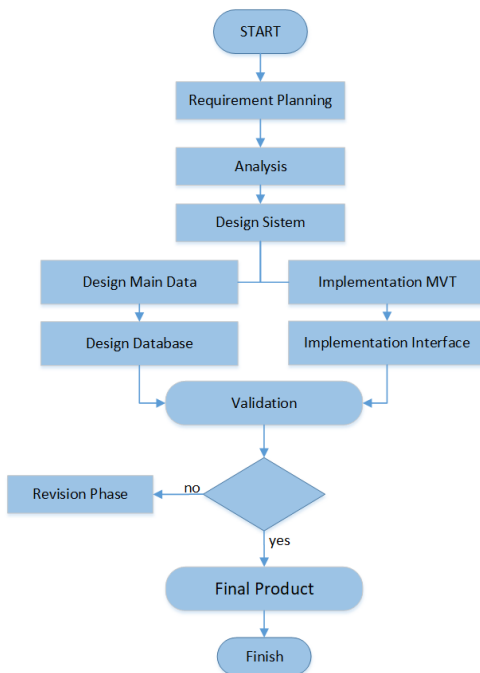
data dan desain terhadap kebutuhan pengguna.

### 3. Development

Desain sistem yang telah dibuat dan disepakati, diubah dari model rancangan kedalam pengkodean program.

### 4. Implementation

Tahap menerapkan desain dari suatu sistem yang telah disepakati pada tahap sebelumnya terhadap pihak perusahaan harus melakukan evaluasi terlebih dahulu. Kemudian pada metode pengembangan sistem ini juga dibutuhkan untuk mendeskripsikan langkah-langkah yang akan dilakukan untuk menyelesaikan penelitian. Berikut gambar 2 tahapan pengembangan sistem E-Rapor.

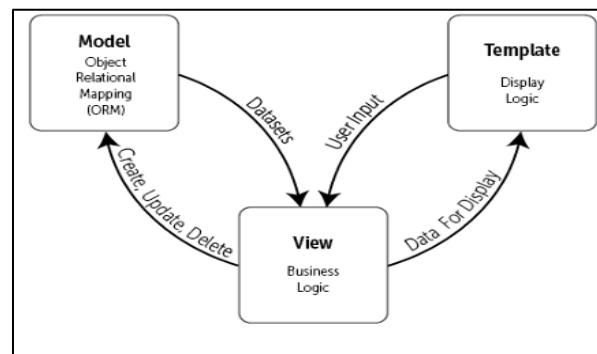


Gambar 2. Tahapan Pengembangan Sistem

### C. Model-View-Template

Django framework dikembangkan dengan mengadopsi arsitektur MVT (*Model View Template*), yang merupakan pola desain perangkat lunak yang digunakan dalam pengembangan aplikasi berbasis web.

Struktur MVT terdiri dari tiga komponen utama yang saling terhubung. Struktur MVT memiliki fungsi-fungsi yang berkesinambungan seperti yang terlihat pada gambar 2 berikut ini.



Gambar 3. Arsitektur Model-View-Template (Sumber: Matt Layman, 2019)

Ketiga struktur MVT yang terdapat pada gambar 2 diatas memiliki fungsi yang berbeda. Terdiri dari tiga komponen utama yang saling terhubung dengan deskripsi masing-masing sebagai berikut:

#### 1. Model

Model berfungsi untuk mengelola data aplikasi dan menyediakan interaksi dengan database. Model ini menyimpan informasi mengenai objek, hubungan antara objek, dan aturan validasi data.

#### 2. View

View bertanggung jawab untuk mengatur logika bisnis dan menangani

permintaan dari pengguna. View ini mengambil data dari model dan memformatnya ke dalam bentuk yang sesuai untuk ditampilkan kepada pengguna

### 3. Template

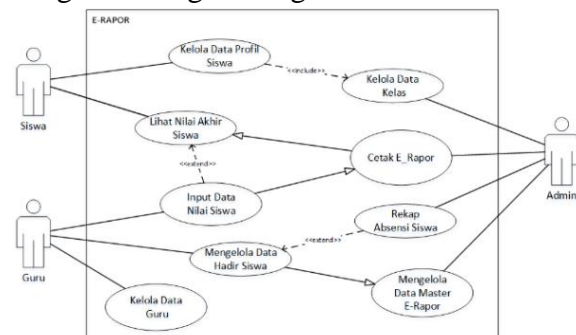
Template bertugas untuk menghasilkan tampilan yang diberikan kepada pengguna. Template menggunakan bahasa template yang terintegrasi dengan Django untuk menggabungkan data dari model dengan markup HTML, sehingga menghasilkan tampilan yang dinamis (Gore et al., 2021). Dengan adanya struktur MVT, Django framework menyediakan kerangka kerja yang kuat dan terstruktur untuk mengembangkan aplikasi web yang efisien dan mudah dipelihara.

### D. Analisis Perancangan Sistem

Pada tahap analisis perancangan sistem dilakukan untuk identifikasi kebutuhan-kebutuhan dari user pengguna dari aplikasi E-Rapor. Dalam perancangannya penjelasan proses interaksi user, proses alur sistem dan mekanisme penggunaan aplikasi E-Rapor menggunakan pemodelan Unified Modeling Language (UML) agar lebih mempermudah proses pembentukan proses didalam sistem (Arianti, Fa'izi, Adam, & Mira Wulandari, 2022). Teknik pemodelan UML tersebut merupakan sebuah teknik standar yang memberikan representasi grafis yang dapat membuat pemahaman dari setiap proses pengembangan sistem (Hakim, Harefa, & Widodo, 2019).

Dengan representasi grafis yang dihasilkan menggunakan teknik UML diagram, telah dibuat use case diagram

berdasarkan analisis kebutuhan sistem pada aplikasi E-Rapor. Hasil rancangan use case dapat dilihat gambar 3 dengan masing-masing user disertai interaksi terhadap sistem dengan masing-masing level.



Gambar 4. Use Case Diagram

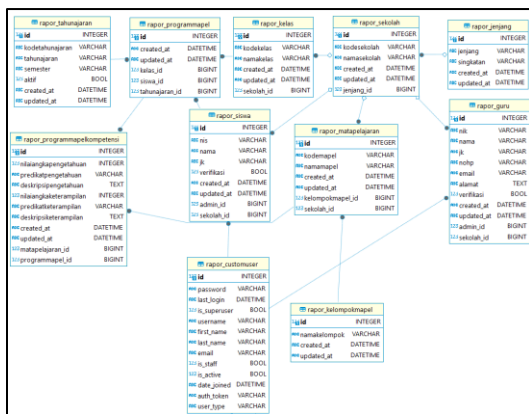
Berdasarkan gambar 3 use case diagram diatas, terdapat tiga user yang memiliki interaksi terhadap sistem dengan level yang berbeda-beda. Berikut keterangan aktor dapat dilihat pada tabel 1 sebagai berikut.

Tabel 1. Deskripsi Use Case Diagram

Use Case	Aktor	Keterangan
Data Profile	Siswa, Guru	Siswa dan guru dapat melakukan update data profile
Data Master	Admin	Admin mengelola data master (jenis penilaian, mata pelajaran)
Kelola data nilai siswa	Guru	Guru dan siswa dapat melihat data nilai.
Rekap kehadiran	Admin, Guru	Admin dapat melihat dan cetak laporan kehadiran siswa
Cetak E-Rapor	Admin	Admin dapat mengunduh dan cetak e-rapor berdasarkan siswa yang telah terselesaikan penilaiannya

Deskripsi use case diagram pada tabel 1 diatas, dapat diketahui bahwa pada setiap level user memiliki interaksi dan fitur serta fungsi yang berbeda. Dengan keseluruhan kebutuhan pengguna telah didefinisikan sesuai untuk kebutuhan pengolahan data yang akan digunakan pada E-Rapor.

Perancangan database menggunakan MySQL. Data yang digunakan merupakan hasil analisis data pada saat melakukan identifikasi kebutuhan pengguna. Rancangan tabel database untuk aplikasi E-Rapor ini dapat dilihat pada gambar 4 berikut ini.



Gambar 5. Rancangan Database Sistem

Pada gambar 4 diatas, dapat dilihat hasil sebuah rancangan database dengan total 11 tabel data. Dengan adanya tabel data tersebut masing-masing tabel memiliki primary key dan foreign key. Beberapa tabel juga diberikan fungsi relasi data dalam fungsionalitas yang sama. Seperti pada tabel siswa yang memiliki relasi dengan beberapa tabel lainnya guna membentuk output E-Rapor berdasarkan hasil pembelajaran siswa.

### HASIL DAN PEMBAHASAN

Implementasi arsitektur *Model View Template (MVT)* pada penelitian ini

diterapkan struktur MVT dimulai dari membuat konfigurasi model pada direktori yang sudah disediakan, dilanjutkan dengan implementasi view dengan membuat sebuah fungsi yang menerima dan mengembalikan parameter (Gore et al., 2021). Selanjutnya menghubungkan fungsi model dengan view dan template dengan melakukan import model yang sebelumnya telah dibuat. Berikut gambar 6a, gambar 6b, dan gambar 6c hasil pengkodean dari implementasi MVT dibawah ini.

```
class ProgramMapelKompetensi(models.Model):
    programmapel = models.ForeignKey(ProgramMapel, on_delete=models.CASCADE)
    matapelajaran = models.ForeignKey(Matapelajaran, on_delete=models.CASCADE)
    nilaiangkepgetahuan = models.IntegerField()
    predikatpengetahuan = models.CharField(max_length=20, null=True)
    deskripsipengetahuan = models.TextField(null=True)
    nilaiangkaterampilan = models.IntegerField()
    predikataterampilan = models.CharField(max_length=20, null=True)
    deskripsikaterampilan = models.TextField(null=True)
    created_at = models.DateTimeField(auto_now_add=True)
    updated_at = models.DateTimeField(auto_now=True)

    def __str__(self):
        return str(self.programmapel_id)
```

Gambar 6a. Hasil Pengkodean “model”

```
def tambahprogrammapel(request):
    tahunajaran = Tahunajaran.objects.filter(aktif=True).first()
    taid = tahunajaran.id
    print(taid)
    if request.method == 'POST':
        form = CreateProgramMapel(request.POST)
        if form.is_valid():
            formpromapel = form.save(commit=False)
            formpromapel.tahunajaran_id = taid
            formpromapel.save()
            messages.success(request, 'Data Ditambah')
            return redirect('manage_programmapel')
    else:
        form = CreateProgramMapel()
        #print(form)
    context = {
        'title': 'TAMBAH DATA PROGRAM MATA PELAJARAN',
        'form': form,
    }
    return render(request, 'admin_template/tambahprogrammapel.html', context)
```

Gambar 6b. Hasil Pengkodean “view”

```
def tambahprogrammapel(request):
    tahunajaran = Tahunajaran.objects.filter(aktif=True).first()
    taid = tahunajaran.id
    print(taid)
    if request.method == 'POST':
        form = CreateProgramMapel(request.POST)
        if form.is_valid():
            formpromapel = form.save(commit=False)
            formpromapel.tahunajaran_id = taid
            formpromapel.save()
            messages.success(request, 'Data Ditambah')
            return redirect('manage_programmapel')
    else:
        form = CreateProgramMapel()
        #print(form)
    context = {
        'title': 'TAMBAH DATA PROGRAM MATA PELAJARAN',
        'form': form,
    }
    return render(request, 'admin_template/tambahprogrammapel.html', context)
```

Gambar 6c. Hasil Pengkodean “template”

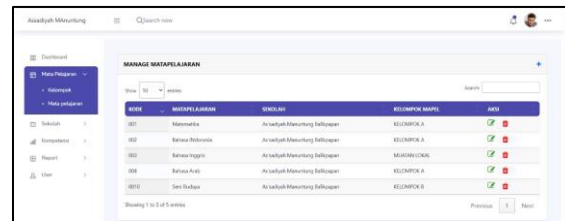
Hasil implementasi MVT yang telah dibuat tersebut, dapat dilihat setelah terbentuknya beberapa menu yang dapat dilakukan management oleh admin diantaranya untuk pengelolaan data sekolah, kelompok mata pelajaran, dan kompetensi. Keseluruhan data tersebut memiliki struktur penggunaan *object model* yang telah diterapkan sesuai dengan arsitektur Model View Template (MVT).

Selanjutnya proses transformasi dari pengkodean menjadi user interface dari masing-masing arsitektur MVT. Berikut tampilan hasil implementasi MVT dalam bentuk user interface dapat dilihat pada gambar 7 dibawah ini.



Gambar 7. Hasil Implementasi *Object*

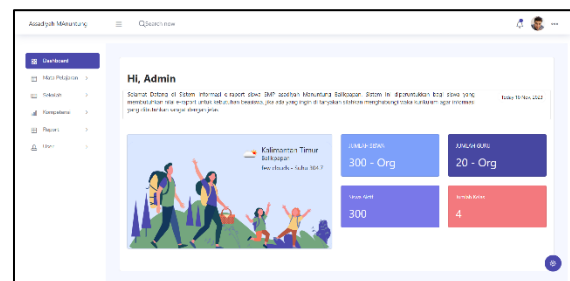
Pada gambar 7 tersebut merupakan bentuk dari implementasi objek dari proses pengkodean model yang berkesinambungan dengan database sistem. Dengan objek sebagai contoh yaitu pada tampilan hasil pengelompokan mata pelajaran (Mapel) seperti yang terlihat dalam kategori mata pelajaran yang berbeda-beda seperti pada gambar diatas. Setelah itu pembentukan view yang dimaksud adalah penambahan data tabel mapel sesuai dengan kategorinya. Adapun tampilan *view* yang terdapat beberapa objek lainnya dapat dilihat pada gambar 8 berikut ini.



Gambar 8. Implementasi *view*

Tampilan *view* untuk masing-masing kategori mapel yang terlihat pada gambar 10 adalah salah satu bentuk implementasi untuk membentuk *field* data tabel “mapel” dengan pembentukan kategori mata pelajaran. Dengan begitu jenis mata pelajaran dapat dibuat lebih detail untuk memudahkan pengelolaan data nilai hasil belajar lebih lengkap. Pengelompokan data mata pelajaran ini merupakan kebutuhan pengguna yang ingin disesuaikan dengan kondisi di sekolah.

Sedangkan pada gambar 9 berikut ini, adalah tampilan untuk dashboard yang akan ditampilkan oleh sistem. Halaman dashboard dapat diakses ketika user telah melakukan login terlebih dahulu.



Gambar 9. Halaman Dashboard

Tampilan dashboard pada gambar 9 tersebut, merupakan sebuah contoh hasil implementasi sistem E-Rapor untuk user admin. Halaman ini digunakan untuk mengelola data master diantaranya: Mata Pelajaran, Sekolah, Kompetensi, dan user.

Kemudian untuk pengelolaan data Report untuk admin adalah proses cetak e-rapor berdasarkan data siswa yang telah berhasil diselesaikan penilaiannya.

## IMPLEMENTASI E-RAPOR

Berikut proses report oleh user admin yang dilakukan adalah adanya pemilihan semester aktif berdasarkan tahun akademik. Pembuatan tahun akademik ini dibuat adalah untuk proses penyimpanan data dapat dilihat pada gambar 10 dibawah ini.



Gambar 10. Fitur Pilih Tahun Akademik

Setelah proses pemilihan tahun akademik. Admin dapat melakukan cetak rapor berdasarkan identitas siswanya. Adapun tampilan aplikasinya dapat dilihat pada gambar 11 berikut.



Gambar 11. Tampilan Data Siswa

Pada gambar 11 diatas, admin dapat melakukan cetak rapor secara langsung dengan menekan tombol aksi yang terlihat pada kolom sebelah kanan. Sehingga proses pembuatan e-rapor dapat dilihat secara *preview* dalam bentuk pdf. Untuk hasil cetak dari e-rapor dapat dilihat pada gambar 12 merupakan hasil cetak E-Rapor berdasarkan data siswa.

No	Nama Pelajaran	Pengetahuan		Keterampilan		Rata-rata
		Angka	Prekuis	Angka	Prekuis	
1	Matematika	85	A	70	A	77,5
2	Bahasa Inggris	90	B	80	B	85
Rata-rata		87,5		75		81,25

Gambar 12. Hasil Cetak E-Rapor

Pada hasil implementasi yang telah berhasil dilakukan, proses selanjutnya adalah pengujian aplikasi menggunakan black box testing (Jaya, 2018). Hasil pengujian dengan parameter uji coba aplikasi e-rapor diantaranya pengujian interface, pengujian hasil implementasi MVT, pengujian menu aplikasi, pengujian input-output seperti cetak e-rapor dengan menggunakan pdf-viewer. Dengan hasil cetak rapor yang dicetak dalam bentuk dokumen tersebut pihak admin dapat melanjutkan proses pemberkasan secara digital dengan mengunduh halaman aplikasi pada gambar 12 sebelumnya menjadi portable document dalam format *pdf*.

## KESIMPULAN

Berdasarkan hasil penelitian, analisis, pemodelan dan implementasi MVT pada aplikasi E-Rapor berhasil dibuat dengan proses pengelolaan data diantaranya dapat diambil kesimpulan bahwa Struktur MVT dalam Django framework memisahkan tugas-tugas yang berbeda secara jelas.

Secara keseluruhan, penggunaan Model, View, Template dalam Django framework membantu pengembang dalam merancang dan mengembangkan aplikasi berbasis web yang terstruktur, efisien, dan mudah dipelihara serta dilakukan



pembaharuan. Setelah adanya aplikasi untuk pengelolaan data rapor untuk seluruh siswa ini dapat dilakukan penyimpanan data akademik siswa secara digital

#### DAFTAR PUSTAKA

- Agustinus Noertjahyana. (2002). Studi Analisis Rapid Application Development Sebagai Salah Satu Alternatif Metode Pengembangan Perangkat Lunak. *Jurnal Informatika*, 3(2), 64–68. Retrieved from <http://puslit2.petra.ac.id/ejournal/index.php/inf/article/view/15819>
- Arianti, T., Fa'izi, A., Adam, S., & Mira Wulandari. (2022). Perancangan Sistem Informasi Perpustakaan Menggunakan Diagram Uml (Unified Modelling Language). *Jurnal Ilmiah Komputer*, 1(1), 19–25. Retrieved from <https://journal.polita.ac.id/index.php/politati/article/view/110/88>
- Gore, H., Singh, R. K., Singh, A., Singh, A. P., Shabaz, M., Singh, B. K., & Jagota, V. (2021). Django: Web Development Simple & Fast. *Annals of the Romanian Society for Cell Biology*, 25(6), 4576–4585. Retrieved from <https://www.annalsofrscb.ro/index.php/journal/article/view/6301>
- Hakim, A. R., Harefa, K., & Widodo, B. (2019). Pengembangan Sistem Informasi Akademik Berbasis Android Menggunakan Flutter Di Politeknik. *SCAN - Jurnal Teknologi Informasi Dan Komunikasi*, 14(3), 27–32. <https://doi.org/10.33005/scan.v14i3.1684>
- Idris, N. Bin, & Mutiara, A. B. (2017). Penerapan Teknologi Web Service Dan Soa Untuk Integrasi Data Pada Sistem Informasi. *Metik Jurnal*, 1(2), 47–52.
- Jaya, T. S. (2018). Pengujian Aplikasi dengan Metode Blackbox Testing Boundary Value Analysis (Studi

- Kasus: Kantor Digital Politeknik Negeri Lampung). *Jurnal Informatika: Jurnal Pengembangan IT*, 3(1), 45–48. <https://doi.org/10.30591/jpit.v3i1.647>
- Septiani, Y., Aribbe, E., & Diansyah, R. (2020). Analisis Sistem. *Jurnal Teknologi Dan Open Source*, 3(1), 131–143.
- Shalahuddin, M., & Sukanto, R. A. (2018). Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek Edisi Revisi, 25–41.