

# SISTEM INFORMASI AKADEMIK SISWA BERBASIS WEBSITE PADA SMAN 1 NGABANG

Ari Setiawan Susanto <sup>1)</sup> Ery Dewayani <sup>2)</sup> Tri Sutrisno <sup>3)</sup>

<sup>1) 2) 3)</sup> Sistem Informasi Universitas Tarumanagara Jakarta Indonesia  
[ari.825180018@stu.untar.ac.id](mailto:ari.825180018@stu.untar.ac.id) <sup>1)</sup>, [eryd@fti.untar.ac.id](mailto:eryd@fti.untar.ac.id) <sup>2)</sup>, [tris@fti.untar.ac.id](mailto:tris@fti.untar.ac.id) <sup>3)</sup>

## ABSTRACT

Information system for Sman 1 Ngabang to process academic of students this web-based and mobile web system can help schools in learning features. The web-based is well designed using the PHP programming language and for the database using MySQL and phpMyAdmin. For testing using the white box method. The method used in software development is Rapid Application Development. The results obtained are Sman 1 Ngabang uses a web-based to obtain information system data in the form of reports. Admin also has full data access to add, update, and delete existing data in the database.

### Key words

Academic, PHP, Rapid Application Development, Website, Web Mobile.

## 1. Pendahuluan

Dalam perkembangan teknologi terus mengalami perkembangan, tentunya akan menyebabkan informasi semakin dicari melalui teknologi yang ada. Teknologi informasi saat ini sangat dibutuhkan untuk membantu perusahaan, instansi, dan sekolah untuk meningkatkan kinerjanya dalam teknologi informasi [1]. Informasi yang banyak dikenal oleh masyarakat adalah *website*. Karena fungsinya dapat mudah diakses dan dapat memberikan informasi dengan lengkap serta tidak memerlukan banyak waktu. Beberapa perusahaan atau sekolah yang dalam proses pengolahan data dan penyebaran informasi masih dilakukan manual [2]. Pemanfaatan internet untuk kepentingan akademik belum dilakukan pada Sman 1 Ngabang merupakan salah satu sekolah negeri yang terletak di Kalimantan Barat, Kabupaten Landak, Kecamatan Ngabang, yang memiliki status sekolah yang layak sehingga menjadi salah satu pilihan sekolah negeri yang baik untuk siswa. Meski demikian, Sman 1 Ngabang sebenarnya memiliki kelemahan dalam memproses pengolahan data. Sistem Akademik yang digunakan di sekolah ini yang sedang berjalan masih menggunakan aplikasi *Microsoft Excel* dan selanjutnya menggunakan pencatatan manual, tanpa menggunakan *database*. Sistem informasi yang dengan langkah tersebut mengakibatkan terjadinya kendala yang diantaranya dalam pengolahan data yang sangat lama dan memakan banyak waktu. Sistem informasi akademik ini akan membantu memudahkan sekolah untuk

mengolah suatu data seputar akademik agar lebih aman dan lebih *efektif* sehingga nantinya akan menghasilkan suatu laporan yang cepat dan tepat. Program ini juga dapat diakses melalui *web mobile* untuk membantu siswa yang tidak memiliki laptop dengan menggunakan *smartphone* [3].

## 2. Metode

### 2.1 Metode Pengembangan Sistem

Metode dalam pembuatan perancangan sistem informasi saat ini adalah menggunakan metode *Rapid Application Development*. Metode *Rapid Application Development* ini merupakan metode pengembangan *software* yang diciptakan untuk mempersingkat waktu yang dibutuhkan pengguna untuk mendesain serta mengimplementasikan sistem informasi sehingga dihasilkan siklus pengembangan yang sangat singkat dan mendapatkan hasil dengan kualitas yang baik [4]. Terdapat 4 tahap dalam RAD yaitu perencanaan kebutuhan, desain sistem, proses pengembangan dan implementasi yang melibatkan penganalisis dan *user* sebagai berikut :

- a. Perencanaan Kebutuhan  
Tahap ini merupakan tahap yang mendasari dalam perbaikan *framework*, perencanaan kebutuhan dilakukan pembuktian terhadap permasalahan dan bermacam-macam informasi yang didapat dari pengguna yaitu kepala sekolah dan kepala tata usaha yang bertujuan untuk mencari kebutuhan yang diinginkan.
- b. Desain Sistem  
Tahap ini merupakan perancangan sistem, keaktifan pengguna yang terlibat penting untuk mencapai tujuan karena tahap ini merupakan proses perancangan dan proses perbaikan dilakukan secara berulang-ulang jika masih terdapat ketidaksesuaian desain dengan kebutuhan pengguna.
- c. Proses Pengembangan  
Tahap ini *framework plan* yang telah dibuat dan diselesaikan, diubah menjadi jenis aplikasi versi beta ke bentuk terakhir. Pada tahap ini, pengembang atau *programmer* harus terus melakukan pengembangan.

d. Implementasi

Tahap ini merupakan tahap terakhir untuk menerapkan suatu desain dari suatu sistem. Dilakukannya proses testing program terlebih dahulu untuk mencari kesalahan yang ada pada sistem yang dibuat sebelum sistem diterapkan.

2.2 Metode Analisis Dan Perancangan

Dalam perancangan Sistem Informasi Akademik Siswa Berbasis Web Pada Sman 1 Ngabang ini menggunakan metode perancangan yaitu *Unified Modeling Language* (UML). UML merupakan beberapa diagram yang digunakan untuk melakukan perancangan terhadap suatu sistem yang berbasis objek. UML dapat digunakan untuk mempermudah pengembangan aplikasi. UML juga dapat menampilkan bahasa pemodelan di bidang rekayasa perangkat lunak pemrograman yang diharapkan dapat memberikan metode standar untuk menggambarkan perancangan suatu sistem. UML terdiri dari banyak komponen grafis yang digabungkan dalam bentuk suatu diagram [5]. Alasan penggambaran komponen grafis dalam garis besar adalah untuk memperkenalkan sebuah sistem berdasarkan fungsi suatu diagram. Perancangan diagram yang digunakan pada sistem ini adalah *usecase diagram*.

*Usecase diagram* merupakan proses penggambaran yang dilakukan untuk menunjukkan hubungan antara pengguna dengan sistem yang dirancang. Hasil representasi dari skema tersebut dibuat secara sederhana dan bertujuan untuk memudahkan pengguna dalam membaca informasi yang diberikan. *Usecase diagram* dapat dilihat pada gambar 1.

1. Proses Login

Pada progress login semua *actor* diharuskan untuk melakukan *login* terlebih dahulu dengan memasukan *username* dan *password* yang sudah terdaftar. Setelah *login* setiap *actor* akan mendapatkan tampilan *web* yang berbeda-beda fungsinya sesuai dengan akses yang diberikan.

2. Proses input

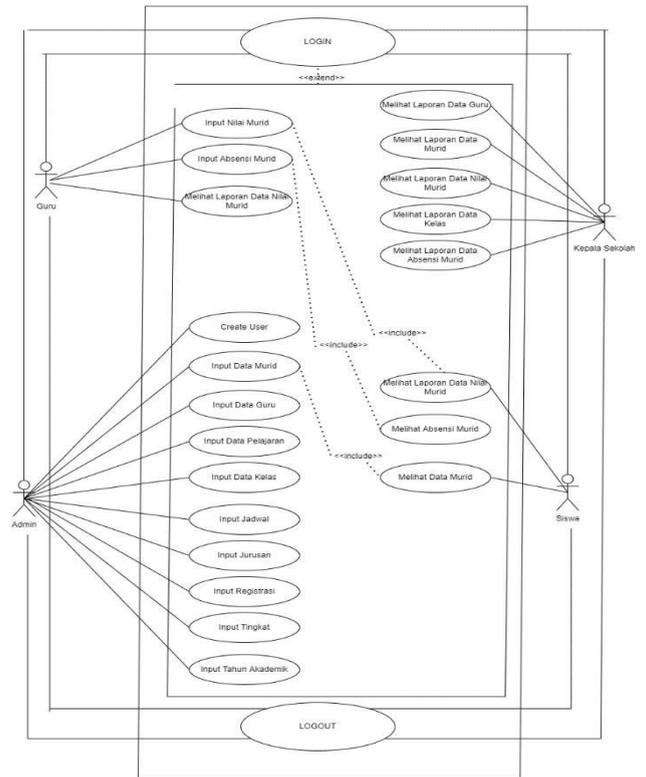
Proses *input* ini dilakukan oleh admin dan guru. Admin melakukan *create user*, *input data murid*, *input data guru*, *input data pelajaran*, *input data kelas*, *input absensi murid*, *input jurusan*, *input registrasi*, *input tingkat*, dan *input tahun akademik*. Guru melakukan *input nilai murid* dan *input absensi murid*.

3. Proses Cetak Laporan

Proses cetak laporan ini bisa dilakukan oleh kepala sekolah, guru, dan murid. Kepala sekolah dapat melakukan cetak laporan data

guru, laporan data murid, laporan data nilai murid, laporan data kelas, dan laporan absensi murid. Guru dapat melakukan cetak laporan data nilai murid. Sedangkan murid dapat melakukan cetak laporan nilai murid, absensi murid, dan data diri murid.

USECASE DIAGRAM



Gambar 1. Usecase Diagram

3. Hasil Dan Pembahasan

Dalam pengujian Sistem Informasi Akademik Siswa Berbasis Web Pada Sman 1 Ngabang ini menggunakan metode pengujian *white box*.

*White box* merupakan metode pengujian yang di dasarkan kepada pengecekan ke dalam detail program yang dilakukan struktur kontrol dari suatu desain pemograman untuk dapat membagi pengujian ke beberapa *form* yang telah dibuat. *White box* menggunakan petunjuk untuk memeriksa aliran *input* dan *output* pada program yang sedang dibuat berjalan dengan baik dan tidak ada nya kendala atau kesalahan [6]. Pengujian *white box* pada sistem ini terdiri dari 4 yaitu, admin, kepala sekolah, guru, dan murid. Pengujian *white box* dapat dilihat dari pada gambar berikut :

Admin					
No	Form	Skenario	Test Case	Hasil	Kesimpulan
1	Halaman Login	Mengisi username dan password yang benar	Klik tombol login	Berhasil masuk kehalaman dashboard	Valid
		Mengisi username dan password yang salah	Klik tombol login	Tidak berhasil masuk kehalaman dashboard	Valid
		Tidak mengisi username dan password	Klik tombol login	Tidak dapat melakukan login	Valid
2	Data Murid	Klik input data	Klik menu input data murid	Menampilkan halaman data murid	Valid
		Tambah murid memasukkan data murid	Klik tombol simpan	Data murid berhasil disimpan	Valid
		Edit data murid	Klik tombol edit	Data murid berhasil diubah	Valid
		Hapus data murid	Klik tombol hapus	Data murid berhasil dihapus	Valid
3	Data guru	Klik input data	Klik menu input data guru	Menampilkan halaman data guru	Valid
		Tambah guru memasukkan data guru	Klik tombol simpan	Data guru berhasil disimpan	Valid
		Edit data guru	Klik tombol edit	Data guru berhasil diubah	Valid
		Hapus data guru	Klik tombol hapus	Data guru berhasil dihapus	Valid
4	Data pelajaran	Klik input data	Klik menu input data pelajaran	Menampilkan halaman data pelajaran	Valid
		Tambah pelajaran memasukkan data pelajaran	Klik tombol simpan	Data pelajaran berhasil disimpan	Valid
		Edit data pelajaran	Klik tombol edit	Data pelajaran berhasil diubah	Valid
		Hapus data pelajaran	Klik tombol hapus	Data pelajaran berhasil dihapus	Valid
5	Data kelas	Klik input data	Klik menu input data kelas	Menampilkan halaman data kelas	Valid
		Tambah kelas memasukkan data kelas	Klik tombol simpan	Data kelas berhasil disimpan	Valid
		Edit data kelas	Klik tombol edit	Data kelas berhasil diubah	Valid
		Hapus data kelas	Klik tombol hapus	Data kelas berhasil dihapus	Valid
6	Data jadwal	Klik input data	Klik menu input data jadwal	Menampilkan halaman data jadwal	Valid
		Tambah jadwal memasukkan data jadwal	Klik tombol simpan	Data jadwal berhasil disimpan	Valid
		Edit data jadwal	Klik tombol edit	Data jadwal berhasil diubah	Valid
		Hapus data jadwal	Klik tombol hapus	Data jadwal berhasil dihapus	Valid
7	Data jurusan	Klik input data	Klik menu input data jurusan	Menampilkan halaman data jurusan	Valid
		Tambah jurusan memasukkan data jurusan	Klik tombol simpan	Data jurusan berhasil disimpan	Valid
		Edit data jurusan	Klik tombol edit	Data jurusan berhasil diubah	Valid
		Hapus data jurusan	Klik tombol hapus	Data jurusan berhasil dihapus	Valid
8	Data registrasi	Klik input data	Klik menu input data registrasi	Menampilkan halaman data registrasi	Valid
		Tambah registrasi memasukkan data registrasi	Klik tombol simpan	Data registrasi berhasil disimpan	Valid
		Edit data registrasi	Klik tombol edit	Data registrasi berhasil diubah	Valid
		Hapus data registrasi	Klik tombol hapus	Data registrasi berhasil dihapus	Valid
9	Data tingkat	Klik input data	Klik menu input data tingkat	Menampilkan halaman data tingkat	Valid
		Tambah tingkat memasukkan data tingkat	Klik tombol simpan	Data tingkat berhasil disimpan	Valid
		Edit data tingkat	Klik tombol edit	Data tingkat berhasil diubah	Valid
		Hapus data tingkat	Klik tombol hapus	Data tingkat berhasil dihapus	Valid
10	Data user	Menu data user	Klik create user	Menampilkan halaman create user	Valid
		Tambah user memasukkan data user	Klik tombol simpan	data user berhasil disimpan	Valid
		Edit data user	Klik tombol edit	Data user berhasil diubah	Valid
		Hapus data user	Klik tombol hapus	Data user berhasil dihapus	Valid
11	Data Tahun Akademik	Menu data Tahun Akademik	Klik Tahun Akademik	Menampilkan halaman Tahun Akademik	Valid
		Tambah tahun akademik memasukkan tahun akademik	Klik tombol simpan	data tahun akademik berhasil disimpan	Valid

		Edit data tahun akademik	Klik tombol edit	Data tahun akademik berhasil diubah	Valid
		Hapus data tahun akademik	Klik tombol hapus	Data tahun akademik berhasil dihapus	Valid

Gambar 2. Pengujian White Box Role Admin

Kepala Sekolah					
No	Form	Skenario	Test Case	Hasil	Kesimpulan
1	Data guru	Laporan data guru	Klik data guru	Menampilkan data guru	Valid
		Print laporan data guru	Klik tombol cetak laporan guru	Mencetak laporan guru	Valid
2	Data murid	Laporan data murid	Klik data murid	Menampilkan data murid	Valid
		Print laporan data murid	Klik tombol cetak laporan murid	Mencetak laporan murid	Valid
3	Data nilai murid	Laporan data nilai murid	Klik laporan nilai murid	Menampilkan nilai murid	Valid
		Print laporan nilai murid	Klik tombol cetak nilai murid	Mencetak nilai murid	Valid
4	Data kelas	Laporan data kelas	Klik laporan data kelas	Menampilkan data kelas	Valid
		Print laporan data kelas	Klik tombol cetak data kelas	Mencetak data kelas	Valid
5	Data absensi murid	Laporan absensi murid	Klik laporan absensi murid	Menampilkan absensi murid	Valid
		Print laporan absensi murid	Klik tombol cetak absensi murid	Mencetak absensi murid	Valid

Gambar 3. Pengujian White Box Role Kepala Sekolah

Guru					
No	Form	Skenario	Test Case	Hasil	Kesimpulan
1	Halaman Login	Mengisi username dan password yang benar	Klik tombol login	Berhasil masuk kehalaman dashboard	Valid
		Mengisi username dan password yang salah	Klik tombol login	Tidak berhasil masuk kehalaman dashboard	Valid
		Tidak mengisi username dan password	Klik tombol login	Tidak dapat melakukan login	Valid
2	Data Nilai Murid	Klik input data	Klik menu input nilai murid	Menampilkan halaman nilai murid	Valid
		Tambah nilai murid memasukkan nilai murid	Klik tombol simpan	nilai murid berhasil disimpan	Valid
		Edit nilai murid	Klik tombol edit	nilai murid berhasil diubah	Valid
		Hapus nilai murid	Klik tombol hapus	nilai murid berhasil dihapus	Valid
3	Data Absensi Murid	Klik input data	Klik menu input absensi murid	Menampilkan halaman data absensi murid	Valid

Gambar 4. Pengujian White Box Role Guru

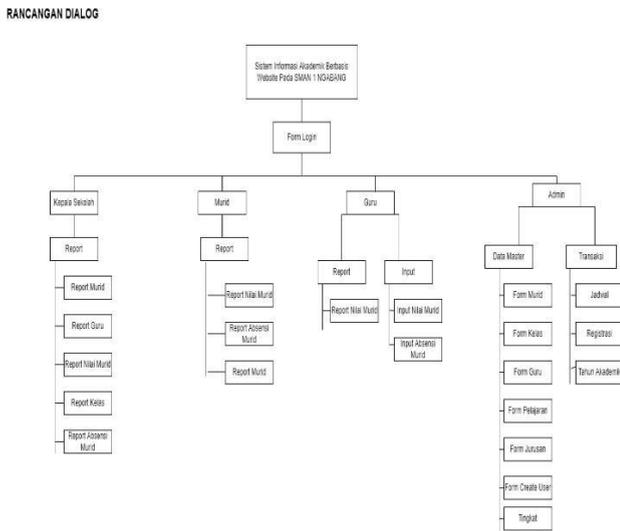
Murid					
No	Form	Skenario	Test Case	Hasil	Kesimpulan
1	Nilai	Laporan nilai murid	Klik nilai murid	Menampilkan halaman nilai murid	Valid
		Print laporan nilai murid	Klik tombol cetak nilai murid	Mencetak laporan nilai murid	Valid
2	Absensi	Laporan absensi murid	Klik absensi murid	Menampilkan halaman absensi murid	Valid
		Print laporan absensi murid	Klik tombol cetak absensi murid	Mencetak laporan absensi murid	Valid
3	Data murid	Laporan data murid	Klik data murid	Menampilkan halaman data murid	Valid
		Print data murid	Klik tombol cetak data murid	Mencetak data murid	Valid

Gambar 5. Pengujian White Box Role Murid

#### 4. Tampilan Interface

Rancangan dialog merupakan rancangan bangunan dari percakapan antara pengguna sistem *user* dengan komputer. Percakapan ini dapat terdiri dari proses pemasukan data ke sistem, menampilkan informasi kepada *user*.

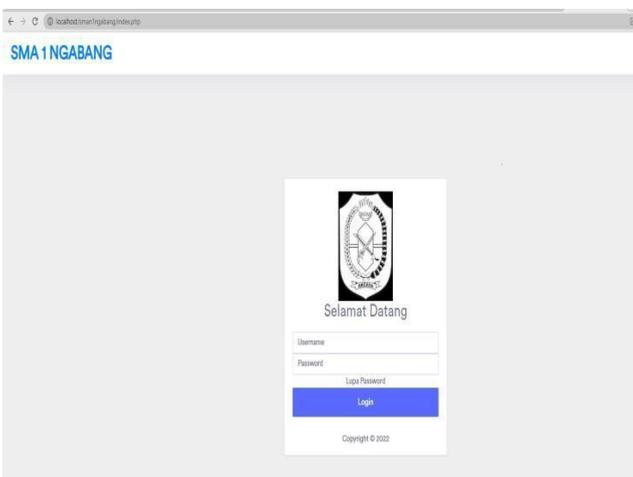
Rancangan Dialog dapat dilihat pada gambar 6.



Gambar 6. Rancangan Dialog

Untuk tampilan *form login website* di mana *user* dapat masuk *form login* tersebut agar bisa mengakses programnya, dan harus melakukan *login* terlebih dahulu menggunakan *username* dan *password* yang sudah didaftarkan oleh admin sekolah. Tampilan awal saat ingin memasuki program ini adalah *form login*.

Form login dapat dilihat pada gambar 7.



Gambar 7. Form Login

Pada pembuatan *web mobile* adalah melakukan pengaturan pada *service worker*. Langkah pertama untuk pembuatan *web mobile* yaitu mempersiapkan *manifest.json*. Manifest ini dapat berfungsi untuk mengatur tampilan aplikasi, mengatur nama aplikasi, *background color*, *theme*, *color*, *display* dan *icon*.

Source code untuk manifest dapat dilihat pada gambar 8.

```

manifest.json X
pwa > src > {} manifest.json > ...
1  {
2    "name": "SMANSA",
3    "short_name": "SMANSA",
4    "start_url": "../index.php",
5    "background_color": "#6dcdb1",
6    "theme_color": "#009578",
7    "display": "standalone",
8    "icons": [
9      {
10     "src": "../img/icons.png",
11     "sizes": "64x64 32x32 24x24 16x16",
12     "type": "image/png"
13   },
14   {
15     "src": "../img/icons.png",
16     "sizes": "192x192",
17     "type": "image/png"
18   },
19   {
20     "src": "../img/icons.png",
21     "sizes": "512x512",
22     "type": "image/png"
23   }
24 ]
25 }
26
    
```

Gambar 8. Manifest

Setelah itu dapat melakukan pengaturan pada *service worker* yang membuat aplikasi ini dapat di *install*.

Source code untuk *Service Worker* dapat dilihat pada gambar 9.

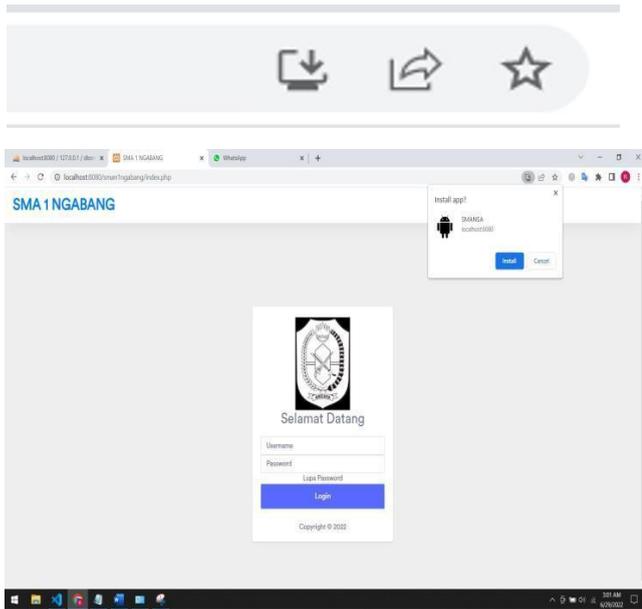
```

JS sw.js > ...
1  self.addEventListener("install", (e) => {
2    console.log("Install!");
3    e.waitUntil();
4  });
5
6  self.addEventListener("fetch", (e) => {
7    e.respondWith(
8      caches.match(e.request).then((response) => {
9        return response || fetch(e.request);
10     });
11   );
12 });
13
14 self.addEventListener("activate", (e) => {
15   self.clients.claim();
16 });
17
    
```

Gambar 9. Service Worker

Website yang menggunakan PWA (Progressive Web Application) akan dapat di install.

Icon install dapat dilihat pada gambar 10.



Gambar 10. Progressive Web Application

Penginputan data murid hanya bisa diakses oleh admin sekolah yang mana data murid tersebut akan ditampilkan pada tabel admin dan data tersebut akan tersimpan di database. Tampilan penginput data murid bisa dilihat pada gambar 11.

**Form Input Data Murid**

ID Murid	<input type="text" value="ID"/>
Nama Murid	<input type="text" value="Nama"/>
Alamat	<input type="text" value="Alamat"/>
Email	<input type="text" value="example@mail.com"/>
Telp	<input type="text" value="08123xxxxxxx"/>

Save
  
Cancel

Copyright © 2022

Gambar 11. Form Input Data Murid

Pada output nilai murid yang sudah diinput oleh guru dapat dilihat pada gambar 12.

<b>SMA NEGERI 1 NGABANG</b> Jl. Veteran, Hilir Ktr. Kec. Ngabang, Kabupaten Landak, Kalimantan Barat 79357							
Nama		: Ari					
ID		: MRD001					
Kelas		: X IPA A					
Tahun Ajaran		: Genap 2021/2022					
Tanggal Cetak: 01 / 07 / 2022							
No	ID Pelajaran	Mata Pelajaran	KKM	Nilai Pengetahuan	Nilai Keterampilan	Nilai Akhir	Predikat
1	MTK001	Matematika	70	72	75	73.5	B
2	EK001	Ekonomi	70	80	88	84	A
3	FS001	Fisika	75	80	81	80.5	A
4	PIK001	Penjaskes	70	90	85	87.5	A
5	BI001	Bahasa Inggris	75	80	81	80.5	A

Gambar 12. Output Nilai Murid

Pada output absensi murid yang sudah diinput oleh guru dapat dilihat pada gambar 13.

<b>SMA NEGERI 1 NGABANG</b> Jl. Veteran, Hilir Ktr. Kec. Ngabang, Kabupaten Landak, Kalimantan Barat 79357			
Kelas		: X IPA A	
Tanggal		: 2022-07-05	
Mata Pelajaran		: Bahasa Indonesia 1	
Tahun		: Genap / 2021/2022	
Tanggal Cetak: 05 / 07 / 2022			
No	ID Murid	Nama Murid	Status
1	MRD001	Ari	Tidak Hadir
2	MRD002	Bara	Tidak Hadir
3	MRD014	bambang	Tidak Hadir
4	MRD015	Enzi	Hadir

Gambar 13. Output Absensi Murid

## 5. Kesimpulan

Kesimpulan pada penelitian ini yaitu Sistem Informasi Akademik Siswa Berbasis Web Pada SMAN 1 NGABANG sebagai berikut :

1. Sistem Informasi Akademik Siswa Berbasis Web Pada Sman 1 Ngabang ini dibuat sebagai solusi bagi pihak sekolah dalam penyajian laporan, setiap data murid, data guru, data pelajaran, data nilai murid data kelas, dan data absensi.
2. Sistem Informasi Akademik Siswa Berbasis Web Pada Sman 1 Ngabang ini dibuat hanya untuk kalangan tertentu yang memiliki akses terhadap sistem yaitu admin, kepala sekolah, guru, dan siswa Sman 1 Ngabang.

## **REFERENSI**

- [1] Dicky, S., & Tri, S. (2021). Sistem Informasi Akademik Berbasis Web Pada Sekolah Menengah Atas Negeri 20 Kabupaten Tangerang. Tangerang: Jurnal Teknologi Informasi.
- [2] Hafiz, R., Arliyana, & Agus, W. (2019). Sistem Informasi Akademik Siswa Berbasis Web Mobile. Kutowangin Barat: Jurnal Sains Komputer dan Teknologi Informasi.
- [3] Imaniawan, F. F., & Riyanto, A. D. (2015). Perancangan Sistem Informasi Akademik Berbasis Web Pada Akademik Pertanian Hkti Banyumas. Banyumas: Telematika.
- [4] Mulyani, S. (2016). Metode Analisis dan Perancangan Sistem . Bandung: Abdi Sistematika.
- [5] Recky, T. D., Rizal, S., & Oktavian, A. L. (2015). Perancangan Sistem Informasi Akademik Sekolah Berbasis Web Studi Kasus Sekolah Menengah Atas Kristen 1 Tomohon. Manado: Jurnal Teknik Elektro dan Komputer.
- [6] Susanto, M. (2016). Perancangan Sistem Informasi Akademik Berbasis Web Pada Smk Pasar Minggu Jakarta. Jakarta: Jurnal Informatika.

Ari Setiawan Susanto. Mahasiswa tingkat akhir Program Studi Sistem Informasi, Universitas Tarumanagara, Jakarta.

Ery Dewayani. Dosen Program Studi Sistem Informasi, Universitas Tarumanagara, Jakarta.

Tri Sutrisno. Dosen Program Studi Sistem Informasi, Universitas Tarumanagara, Jakarta.