

# Dokumen Teknik

---



20 JUNI

---

**Sistem Informasi Pembayaran Sekolah**  
**Ditulis oleh: Alief Cahyo Utomo – 2113030101**



Nama  
Logo

---

# Informasi Dokumen

## Tujuan

Aplikasi pembayaran digunakan untuk mengakomodasi transaksi pembayaran siswa dan bagian keuangan atau bendahara sekolah. Dengan adanya sistem ini diharapkan akan memberikan kemudahan bagi kepala sekolah, bendahara, siswa dan orang tua untuk memperoleh informasi terkait dengan pembayaran siswa.

***“Silahkan sesuaikan dengan kasus yang anda bahas atau ambail, hilangkan tulisan yang berwarna merah”***

Pihak-pihak yang terlibat dalam penggunaan aplikasi ini adalah :

1. **Kepala sekolah**, untuk melihat keuangan sekolah
2. **Bendahara**, untuk memasukkan data transaksi pembayaran siswa
3. **Wali Murid**, untuk memantau history pembayaran siswa
4. **Siswa**, untuk memantau history pembayaran siswa

## Ruang Lingkup

- |                    |  |
|--------------------|--|
| Overview sistem    | : Aplikasi ini dikembangkan untuk memberikan kemudahan kepada para pengguna yang membutuhkan informasi terkait dengan pembayaran sekolah.  |
| Deskripsi pengguna | : Dalam sistem ini secara umum memiliki 4 pengguna yaitu Kepala sekolah, bendahara, Orang tua / Wali murid, siswa. Dimana setiap pengguna memiliki hak akses atau menu yang berbeda-beda sesuai dengan kebutuhan masing-masing pengguna. |
| Overview produk    | : Aplikasi keuangan ini memiliki beberapa fitur :  |

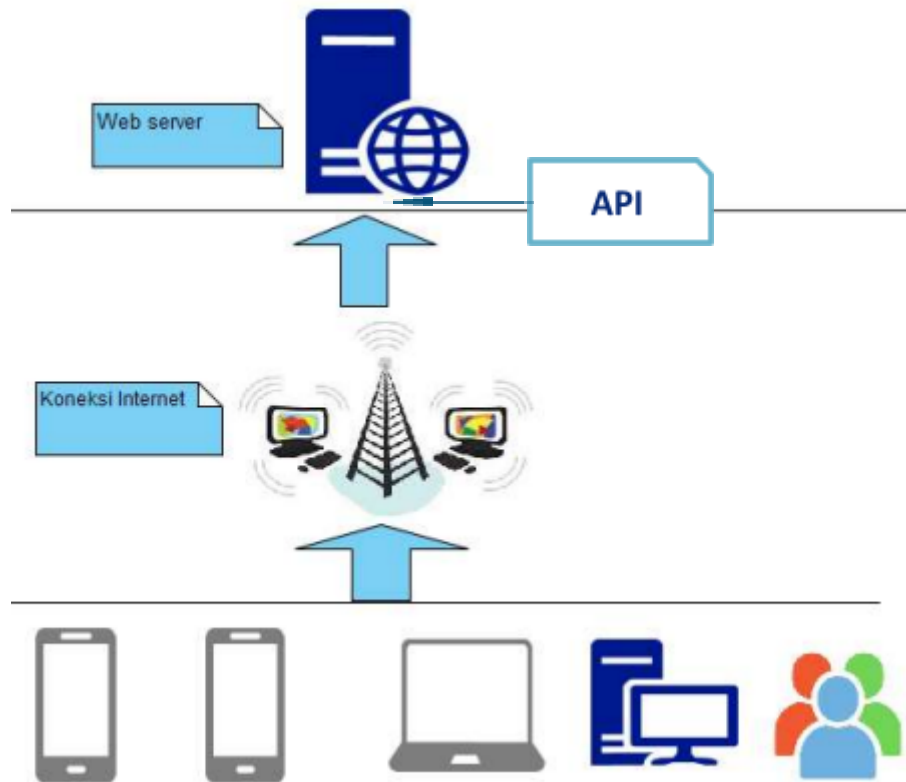
- 
1. **Data master**, Merupakan menu untuk manajemen data-data yang dibutuhkan oleh sistem ini
  2. **Transaksi**, Merupakan menu proses transaksi yang dibutuhkan oleh sistem
  3. **Laporan**, Menu yang digunakan untuk melihat laporan-laporan yang diperlukan oleh sistem dan masing-masing penggunaanya.

Kebutuhan sistem : Untuk menjalankan aplikasi ini dalam tahap deployment pengguna menggunakan spesifikasi berikut :

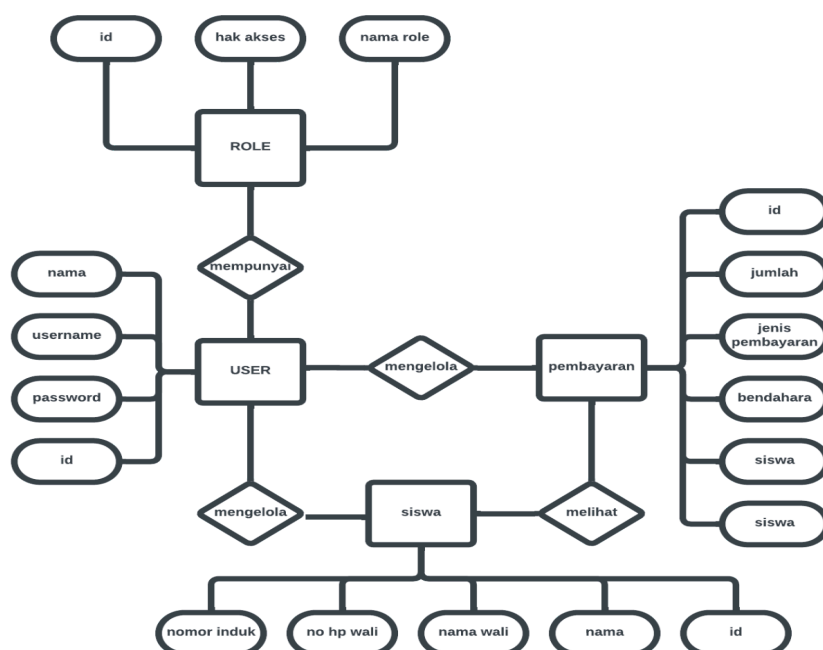
1. Sistem Operasi : Arch Linux
2. Laptop Thinkpad x230 RAM 10GB, SSD
3. Web Server : Backend Django , Frontend Svelte
4. Database Sqlite

Untuk tahap penggunaan aplikasi keuangan ini bisa dijalankan di multi OS (Windows, Linux, MacOS). Untuk database bisa menggunakan sqlite ataupun database lain yang di support oleh framework django.

## Arsitektur Sistem



## Desain ERD



## Dokumentasi API :

### 1. Mendapatkan Akses dan Refresh Token.

URL	<a href="http://localhost:8000/api/v1/auth/token/">http://localhost:8000/api/v1/auth/token/</a>
Method	OPTION, POST
Permissions	-
Deskripsi : Mengambil kredensial pengguna dan mengembalikan pasangan token JSON web akses dan refresh untuk membuktikan autentikasi dari kredensial tersebut.	
Request	
username	username akun.
password	password akun.
Response	
access	access token.
refresh	refresh token.

### 2. Mendapatkan Akses token dari refresh token.

URL	<a href="http://localhost:8000/api/v1/auth/token/refresh">http://localhost:8000/api/v1/auth/token/refresh</a>
Method	OPTION, POST
Permissions	-
Mengambil refresh token dan mengembalikan akses token jika token refresh tersebut valid.	
Request	
refresh	refresh token.
Response	
access	access token

### 3. Mendapatkan User Object yang sedang login.

URL	<a href="http://localhost:8000/api/v1/auth/user/">http://localhost:8000/api/v1/auth/user/</a>
Method	GET
Permission	isAuthenticated
Mendapatkan User Object dari user yang sedang aktif.	
Request	
-	-
Response	
url	URL user object.
username	Username akun.
email	E-Mail akun.
groups	List URL Group object di mana akun terdaftar

#### 4. Mendapatkan Semua User Object.

URL	<a href="http://localhost:8000/api/v1/users/">http://localhost:8000/api/v1/users/</a>
Method	GET, HEAD, OPTION
Permission	isAuthenticated, isAdminUser
Mendapatkan semua User Object.	
Request	
-	-
Response	
count	jumlah dari user object yang diambil.
next	URL ke paginasi berikutnya.
previous	URL ke paginasi sebelumnya.
results	List User Object dengan fields, sebagai berikut: {




	<pre> url : URL user object. username : Username akun. email : E-Mail akun. groups : List URL Group object di mana akun terdaftar } </pre>
--	--


#### 5. Mendapatkan Semua Group Object.

URL	<a href="http://localhost:8000/api/v1/groups/">http://localhost:8000/api/v1/groups/</a>
Method	GET, HEAD, OPTION
Permission	isAuthenticated
Mendapatkan semua Group Object.	
Request	
-	-
Response	
count	jumlah dari Group object yang diambil.
next	URL kepaginasi berikutnya.
previous	URL ke paginasi sebelumnya.
results	List Group Object dengan fields sebagai berikut: <pre> { url : URL user object. name: nama group. } </pre>

#### 6. Mendapat Group Object.


URL	<a href="http://localhost:8000/api/v1/groups/[id]/">http://localhost:8000/api/v1/groups/[id]/</a>
Method	GET
Permission	isAuthenticated
Mendapatkan Group Object dengan id.	
Request	
-	-
Response	
url	URL Group Object.
name	nama <u>group.</u> 

7. Mendapatkan Siswa Object dengan id.

URL	<a href="http://localhost:8000/api/v1/siswa/[id]/">http://localhost:8000/api/v1/siswa/[id]/</a>
Method	GET
Permission	V1   pembayaran   Can View Siswa
Mendapatkan Siswa Object dengan id.	
Request	
-	-
Response	
url	URL siswa object.
created	tanggal dan waktu pembuatan siswa object.
nama	nama siswa. 
<u>namawali</u>	nama wali siswa.
<u>nohpwali</u>	no hp wali siswa.
<u>noinduk</u>	no induk nasional siswa.

8. Mendapatkan semua siswa object.



URL	<a href="http://localhost:8000/api/v1/siswa/">http://localhost:8000/api/v1/siswa/</a>	
Method	GET, HEAD, OPTION	
Permission	V1   pembayaran   Can View Siswa	
Mendapatkan semua siswa object		
Request		
-	-	
Response		
count	jumlah dari Group object yang diambil.	
next	URL kepaginasi berikutnya. 	
previous	URL ke paginasi sebelumnya.	
results	List siswa object.	

#### 9. Menambah Siswa Object.

URL	<a href="http://localhost:8000/api/v1/siswa/">http://localhost:8000/api/v1/siswa/</a>	
Method	POST	
Permission	V1   pembayaran   Can Add Siswa	
Menambahkan siswa object.		
Request		
nama	string	nama siswa.
<u>namawali</u>	string	nama wali siswa.
<u>nohpwali</u>	string	no hp wali.
<u>nomorinduk</u>	<u>interger</u>	nomor induk nasional siswa
Response		
url	URL siswa object.	

created	tanggal dan waktu pembuatan siswa object.
nama	nama siswa.
<u>namawali</u>	nama wali siswa.
<u>nohpwali</u>	no hp wali siswa.
<u>noinduk</u>	no induk nasional siswa.

#### 10. Mengedit Siswa Object.

URL	<a href="http://localhost:8000/api/v1/siswa/[id]">http://localhost:8000/api/v1/siswa/[id]</a>	
Method	PUT, PATCH	
Permission	V1   pembayaran   Can change siswa	
Mengedit siswa object sesuai dengan id.		
Request		
nama	string	nama siswa.
<u>namawali</u>	string	nama wali siswa.
<u>nohpwali</u>	string	no hp wali.
<u>nomorinduk</u>	<u>interger</u>	nomor induk nasional siswa
Response		
url	URL siswa object.	
created	tanggal dan waktu pembuatan siswa object.	
nama	nama siswa.	
<u>namawali</u>	nama wali siswa.	
<u>nohpwali</u>	no hp wali siswa.	
<u>noinduk</u>	no induk nasional siswa.	

#### 11. Menghapus Siswa Object.

URL	<a href="http://localhost:8000/api/v1/siswa/[id]">http://localhost:8000/api/v1/siswa/[id]</a>	
Method	DELETE	
Permission	V1   pembayaran   Can delete siswa	
Menghapus siswa object sesuai dengan id.		
Request		
-	-	-
Response		
-	-	

## 12. Mendapatkan Pembayaran dengan id.

URL	<a href="http://localhost:8000/api/v1/pembayaran/[id]/">http://localhost:8000/api/v1/pembayaran/[id]/</a>
Method	GET
Permission	V1   pembayaran   Can View Pembayaran
Mendapatkan Pembayaran Object dengan id.	
Request	
-	-
Response	
url	URL siswa object.
created	tanggal dan waktu pembuatan siswa object.
jumlah	Jumlah pembayaran.
detail	detail dari pembayaran.
siswa	URL ke siswa object.
bendahara	URL ke bendahara object.
<u>tglbayar</u>	tanggal pembayaran di input.
siswaname	nama siswa
bendahara	nama bendahara

## 13. Mendapatkan semua Pembayaran.


URL	<a href="http://localhost:8000/api/v1/pembayaran/">http://localhost:8000/api/v1/pembayaran/</a>
Method	GET, HEAD, OPTION
Permission	V1   pembayaran   Can View Pembayaran
Mendapatkan semua pembayaran object	
Request	
-	-
Response	
count	jumlah dari pembayaran object yang diambil.
next	URL kepaginasi berikutnya.
previous	URL ke paginasi sebelumnya.
results	List pembayaran object.

#### 14. Menambahkan Pembayaran.

URL	<a href="http://localhost:8000/api/v1/pembayaran/">http://localhost:8000/api/v1/pembayaran/</a>	
Method	POST	
Permission	V1   pembayaran   Can add pembayaran	
Menambahkan pembayaran object.		
Request		
jumlah	integer	Jumlah pembayaran.
detail	string	detail dari pembayaran.
siswa	string	URL ke siswa object.
bendahara	string	URL ke bendahara object.
<u>tglbayar</u>	string	tanggal pembayaran di input, format mm/dd/yyyy

Response	
url	URL siswa object.
created	tanggal dan waktu pembuatan siswa object.
jumlah	Jumlah pembayaran.
detail	detail dari pembayaran.
siswa	URL ke siswa object.
bendahara	URL ke bendahara object.
<u>tglbayar</u>	tanggal pembayaran di input.
siswaname	nama siswa
bendahara	nama bendahara

#### 15. Mengedit pembayaran.

URL		<a href="http://localhost:8000/api/v1/pembayaran/id/">http://localhost:8000/api/v1/pembayaran/id/</a>
Method		PUT, PATCH
Permission		V1   pembayaran   Can edit pembayaran
Mengedit pembayaran object. 		
Request		
jumlah	integer	Jumlah pembayaran.
detail	string	detail dari pembayaran.
siswa	string	URL ke siswa object.
bendahara	string	URL ke bendahara object.
<u>tglbayar</u>	string	tanggal pembayaran di input, format mm/dd/yyyy

Response	
url	URL siswa object.
created	tanggal dan waktu pembuatan siswa object.
jumlah	Jumlah pembayaran.
detail	detail dari pembayaran.
siswa	URL ke siswa object.
bendahara	URL ke bendahara object.
<u>tglbayar</u>	tanggal pembayaran di input.
siswaname	nama siswa
bendahara	nama bendahara

16. Menghapus pembayaran.

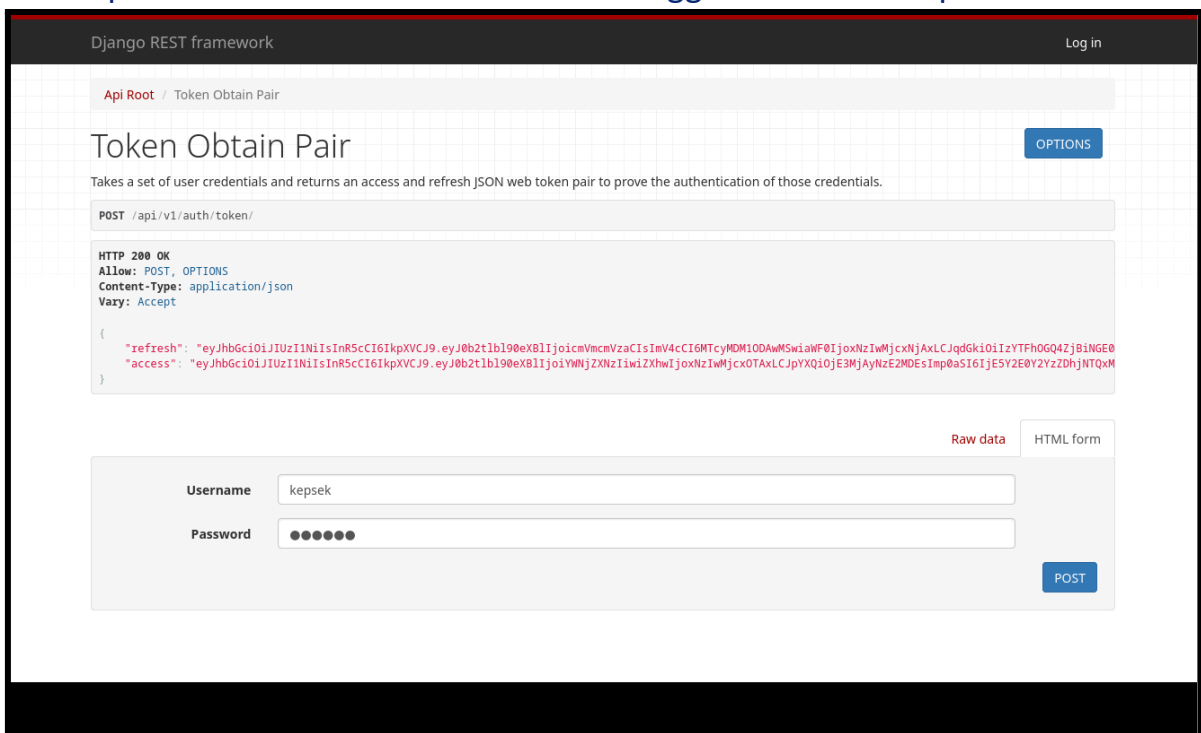
URL	<a href="http://localhost:8000/api/v1/pembayaran/[id]/">http://localhost:8000/api/v1/pembayaran/[id]/</a>	
Method	DELETE	
Permission	V1   pembayaran   Can delete <u>pembayaran</u>	
Menghapus pembayaran object sesuai dengan id.		
Request		
-	-	-
Response		
-	-	

17. Melihat history pembayaran siswa dengan nomor induk.

URL	<a href="http://localhost:8000/api/v1/lihatpembayaran/[nomor induk]/">http://localhost:8000/api/v1/lihatpembayaran/[nomor induk]/</a>	
Method	GET	
Permission	-	
Mendapatkan history pembayaran dari siswa berdasarkan nomor induk nya.		
Request		
-	-	-
Response		
results	list pembayaran obj	

## API Screenshot

1. Mendapatkan Akses dan Refresh Token menggunakan akun kepek.



2. Mendapatkan akses token.





Django REST framework admin ▾

Api Root / User List

## User List

API endpoint that allows users to be viewed.

OPTIONS GET ▾

GET /api/v1/users/

HTTP 200 OK  
Allow: GET, HEAD, OPTIONS  
Content-Type: application/json  
Vary: Accept

```
{
  "count": 3,
  "next": null,
  "previous": null,
  "results": [
    {
      "url": "http://localhost:8000/api/v1/users/3/",
      "username": "bendahara",
      "email": "bendahara@example.com",
      "groups": [
        "http://localhost:8000/api/v1/groups/2/"
      ]
    },
    {
      "url": "http://localhost:8000/api/v1/users/2/",
      "username": "kepsek",
      "email": "kepsek@example.com",
      "groups": [
        "http://localhost:8000/api/v1/groups/1/"
      ]
    },
    {
      "url": "http://localhost:8000/api/v1/users/1/",
      "username": "admin",
      "email": "admin@example.com",
      "groups": [
        "http://localhost:8000/api/v1/groups/3/"
      ]
    }
  ]
}
```

## 5. Mendapatkan semua group object.

Django REST framework admin ▾

Api Root / Group List

## Group List

API endpoint that allows groups to be viewed.

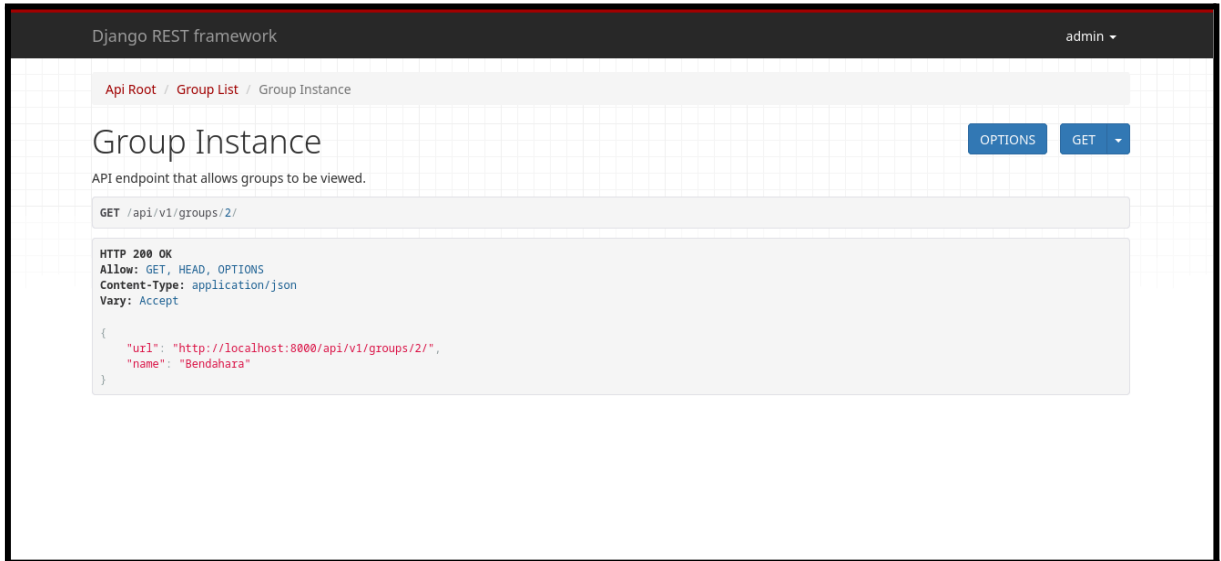
OPTIONS GET ▾

GET /api/v1/groups/

HTTP 200 OK  
Allow: GET, HEAD, OPTIONS  
Content-Type: application/json  
Vary: Accept

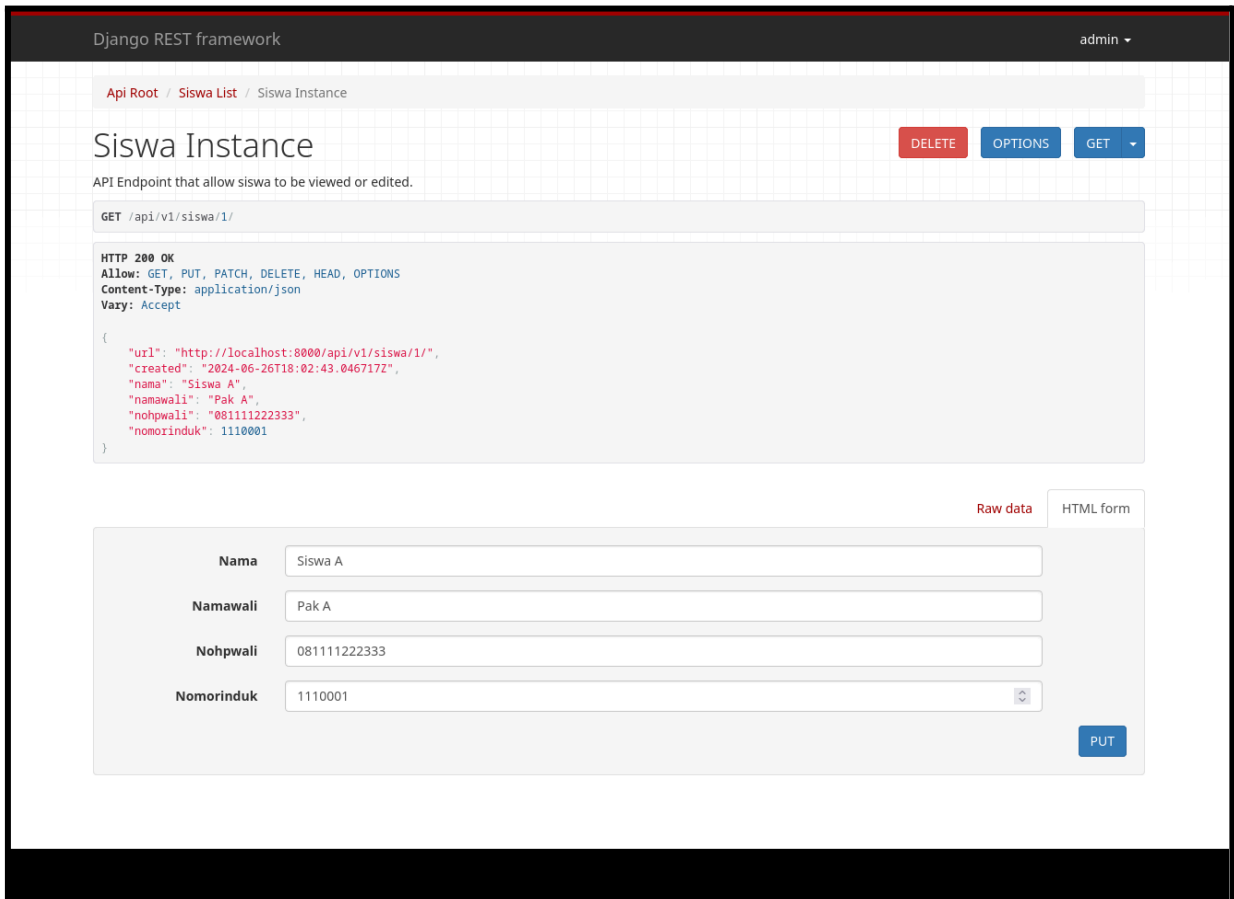
```
{
  "count": 3,
  "next": null,
  "previous": null,
  "results": [
    {
      "url": "http://localhost:8000/api/v1/groups/2/",
      "name": "Bendahara"
    },
    {
      "url": "http://localhost:8000/api/v1/groups/1/",
      "name": "Kepala Sekolah"
    },
    {
      "url": "http://localhost:8000/api/v1/groups/3/",
      "name": "admin"
    }
  ]
}
```

## 6. Mendapatkan group object dengan id.



The screenshot shows the Django REST framework interface for the 'Group Instance' view. The breadcrumb trail is 'Api Root / Group List / Group Instance'. The view title is 'Group Instance' with 'OPTIONS' and 'GET' buttons. The description is 'API endpoint that allows groups to be viewed.' The URL is 'GET /api/v1/groups/2/'. The response is an HTTP 200 OK with headers: 'Allow: GET, HEAD, OPTIONS', 'Content-Type: application/json', and 'Vary: Accept'. The JSON body is: { "url": "http://localhost:8000/api/v1/groups/2/", "name": "Bendahara" }.

## 7. Mendapatkan siswa obj dengan id 1.



The screenshot shows the Django REST framework interface for the 'Siswa Instance' view. The breadcrumb trail is 'Api Root / Siswa List / Siswa Instance'. The view title is 'Siswa Instance' with 'DELETE', 'OPTIONS', and 'GET' buttons. The description is 'API Endpoint that allow siswa to be viewed or edited.' The URL is 'GET /api/v1/siswa/1/'. The response is an HTTP 200 OK with headers: 'Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS', 'Content-Type: application/json', and 'Vary: Accept'. The JSON body is: { "url": "http://localhost:8000/api/v1/siswa/1/", "created": "2024-06-26T18:02:43.046717Z", "nama": "Siswa A", "namawali": "Pak A", "nohpwali": "081111222333", "nomorinduk": "1110001" }. Below the response, there are tabs for 'Raw data' and 'HTML form'. The 'HTML form' tab is active, showing a form with fields: 'Nama' (Siswa A), 'Namawali' (Pak A), 'Nohpwali' (081111222333), and 'Nomorinduk' (1110001). A 'PUT' button is at the bottom right.

## 8. Mendapatkan semua siswa object.

Django REST framework

admin

Api Root / Siswa List

Siswa List

OPTIONSGET

API Endpoint that allow siswa to be viewed or edited.

GET /api/v1/siswa/

HTTP 200 OK  
Allow: GET, POST, HEAD, OPTIONS  
Content-Type: application/json  
Vary: Accept

```
{
  "count": 2,
  "next": null,
  "previous": null,
  "results": [
    {
      "url": "http://localhost:8000/api/v1/siswa/1/",
      "created": "2024-06-26T18:02:43.046717Z",
      "nama": "Siswa A",
      "namawali": "Pak A",
      "nohpwali": "081111222333",
      "nomorinduk": 1110001
    },
    {
      "url": "http://localhost:8000/api/v1/siswa/2/",
      "created": "2024-06-26T18:03:34.232996Z",
      "nama": "Siswa B",
      "namawali": "pak B",
      "nohpwali": "081111222333",
      "nomorinduk": 1110002
    }
  ]
}
```

Raw dataHTML form

Nama

Namawali

Nohpwali

Nomorinduk

POST

9. Menambahkan siswa object, dengan nama “Siswa C”.

Django REST framework

admin

Api Root / Siswa List

Siswa List

OPTIONSGET

API Endpoint that allow siswa to be viewed or edited.

POST /api/v1/siswa/

HTTP 201 Created

Allow: GET, POST, HEAD, OPTIONS

Content-Type: application/json

Location: http://localhost:8000/api/v1/siswa/4/

Vary: Accept

```
{
  "url": "http://localhost:8000/api/v1/siswa/4/",
  "created": "2024-07-06T13:29:22.631287Z",
  "nama": "Siswa C",
  "namawali": "sis",
  "nohpwali": "081111222333",
  "nomorinduk": "1110003"
}
```

Raw data

HTML form

Nama

Siswa C

Namawali

sis

Nohpwali

081111222333

Nomorinduk

1110003

POST

10. Mengedit “Siswa C” menjadi “Siswa D”

Django REST framework

kepek

Api Root / Siswa List / Siswa Instance

Siswa Instance

DELETEOPTIONSGET

API Endpoint that allow siswa to be viewed or edited.

PUT /api/v1/siswa/4/

HTTP 200 OK

Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS

Content-Type: application/json

Vary: Accept

```
{
  "url": "http://localhost:8000/api/v1/siswa/4/",
  "created": "2024-07-06T13:29:22.631287Z",
  "nama": "Siswa D",
  "namawali": "sis",
  "nohpwali": "081111222333",
  "nomorinduk": "1110003"
}
```

Raw data

HTML form

Nama

Siswa D

Namawali

sis

Nohpwali

081111222333

Nomorinduk

1110003

PUT

## 11. Menghapus “Siswa D”

Django REST framework

kepek

Api Root / Siswa List / Siswa Instance

Siswa Instance

DELETEOPTIONSGET

API Endpoint that allow siswa to be viewed or edited.

GET /api/v1/siswa/4/

HTTP 404 Not Found

Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS

Content-Type: application/json

Vary: Accept

```
{
  "detail": "No Siswa matches the given query."
}
```

Raw data

HTML form

Nama

Namawali

Nohpwali

Nomorinduk

PUT

## 12. Mendapatkan pembayaran object dengan id 1.

Django REST framework bendahara ▾

Api Root / Pembayaran List / Pembayaran Instance

### Pembayaran Instance

API endpoint that allows pembayaran be viewed or edited.

DELETE OPTIONS GET ▾

GET /api/v1/pembayaran/1/

**HTTP 200 OK**  
Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS  
Content-Type: application/json  
Vary: Accept

```
{
  "url": "http://localhost:8000/api/v1/pembayaran/1/",
  "created": "2024-06-26T18:07:02.726446Z",
  "jumlah": 500000,
  "detail": "SPP",
  "siswa": "http://localhost:8000/api/v1/siswa/1/",
  "bendahara": "http://localhost:8000/api/v1/users/3/",
  "tglbayar": "2024-06-27",
  "siswaname": "Siswa A",
  "bendaharaname": "bendahara"
}
```

Raw data HTML form

Jumlah	<input type="text" value="500000"/>
Detail	<input type="text" value="SPP"/>
Siswa	<input type="text" value="Siswa A"/>
Bendahara	<input type="text" value="bendahara"/>
Tglbayar	<input type="text" value="06 / 27 / 2024"/>

PUT

## 13. Mendapatkan semua pembayaran object.



Django REST framework

bendahara

Api Root / Pembayaran List

Pembayaran List

OPTIONSGET

API endpoint that allows pembayaran be viewed or edited.

GET /api/v1/pembayaran/

HTTP 200 OK

Allow: GET, POST, HEAD, OPTIONS

Content-Type: application/json

Vary: Accept

```
{
  "count": 1,
  "next": null,
  "previous": null,
  "results": [
    {
      "url": "http://localhost:8000/api/v1/pembayaran/1/",
      "created": "2024-06-26T18:07:02.726446Z",
      "jumlah": 500000,
      "detail": "SPP",
      "siswa": "http://localhost:8000/api/v1/siswa/1/",
      "bendahara": "http://localhost:8000/api/v1/users/3/",
      "tglbayar": "2024-06-27",
      "siswaname": "Siswa A",
      "bendaharaname": "bendahara"
    }
  ]
}
```

Raw data

HTML form

Jumlah

Detail

Siswa

Siswa A

Bendahara

admin

Tglbayar

mm / dd / yyyy

POST

14. Menambahkan pembayaran object, atas “Siswa B”.

Django REST framework

bendahara

Api Root / Pembayaran List

Pembayaran List

OPTIONSGET

API endpoint that allows pembayaran be viewed or edited.

POST /api/v1/pembayaran/

HTTP 201 Created

Allow: GET, POST, HEAD, OPTIONS

Content-Type: application/json

Location: http://localhost:8000/api/v1/pembayaran/2/

Vary: Accept

```
{
  "url": "http://localhost:8000/api/v1/pembayaran/2/",
  "created": "2024-07-06T14:03:06.040233Z",
  "jumlah": 200000,
  "detail": "SPP",
  "siswa": "http://localhost:8000/api/v1/siswa/2/",
  "bendahara": "http://localhost:8000/api/v1/users/3/",
  "tglbayar": "2024-07-06",
  "siswaname": "Siswa B",
  "bendaharaname": "bendahara"
}
```

Raw dataHTML form

Jumlah

200000

Detail

SPP

Siswa

Siswa B

Bendahara

bendahara

Tglbayar

07 / 06 / 2024

POST

15. Mengedit pembayaran object, mengubah detail menjadi "SPP 2024"

Django REST framework

bendahara

Api Root / Pembayaran List / Pembayaran Instance

Pembayaran Instance

DELETEOPTIONSGET

API endpoint that allows pembayaran be viewed or edited.

PUT /api/v1/pembayaran/2/

HTTP 200 OK  
Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS  
Content-Type: application/json  
Vary: Accept

```
{
  "url": "http://localhost:8000/api/v1/pembayaran/2/",
  "created": "2024-07-06T14:03:08.040233Z",
  "jumlah": 200000,
  "detail": "SPP 2024",
  "siswa": "http://localhost:8000/api/v1/siswa/2/",
  "bendahara": "http://localhost:8000/api/v1/users/3/",
  "tglbayar": "2024-07-06",
  "siswaname": "Siswa B",
  "bendaharaname": "bendahara"
}
```

Raw dataHTML form

Jumlah

200000

Detail

SPP 2024

Siswa

Siswa B

Bendahara

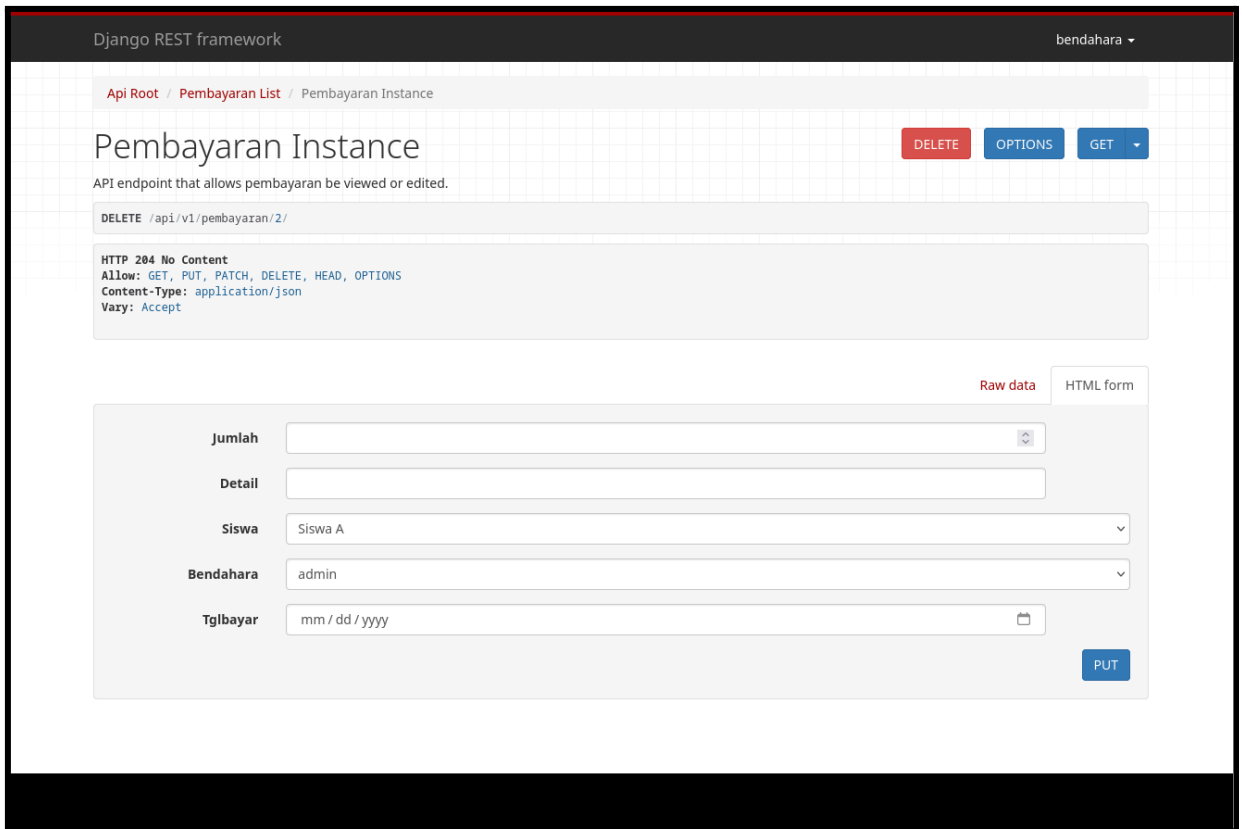
bendahara

Tglbayar

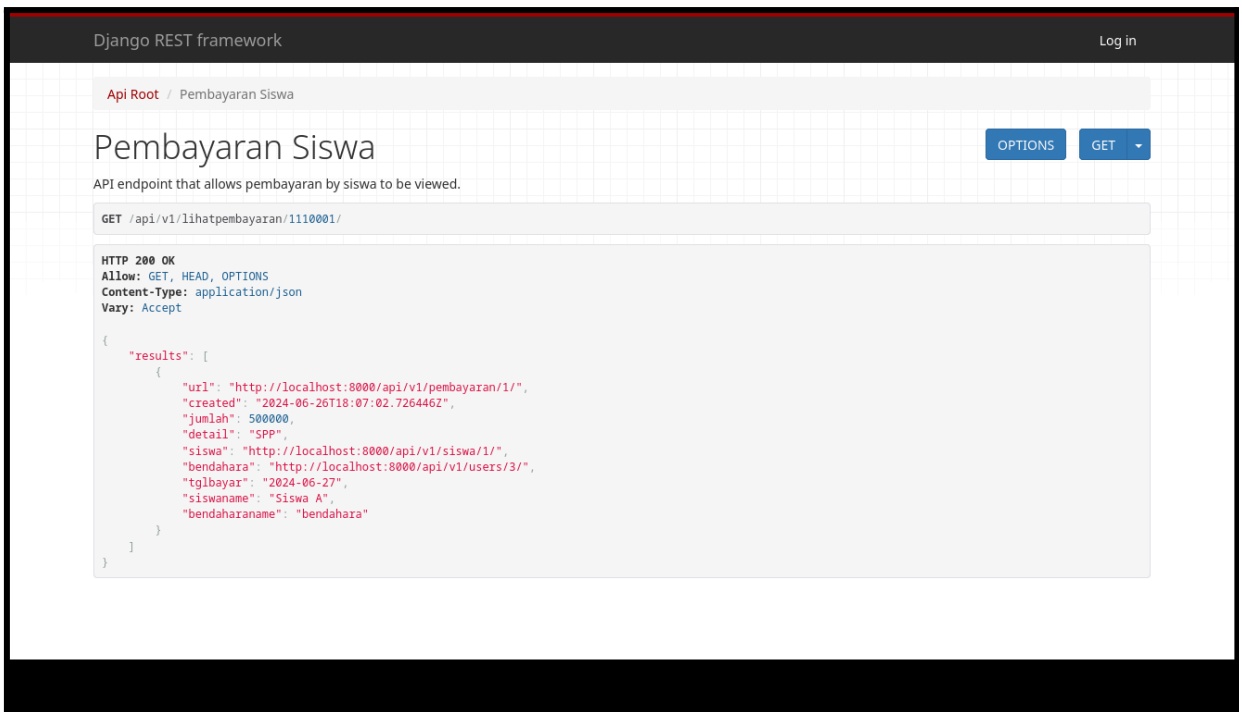
07 / 06 / 2024

PUT

16. Menghapus Pembayaran dengan id 2.

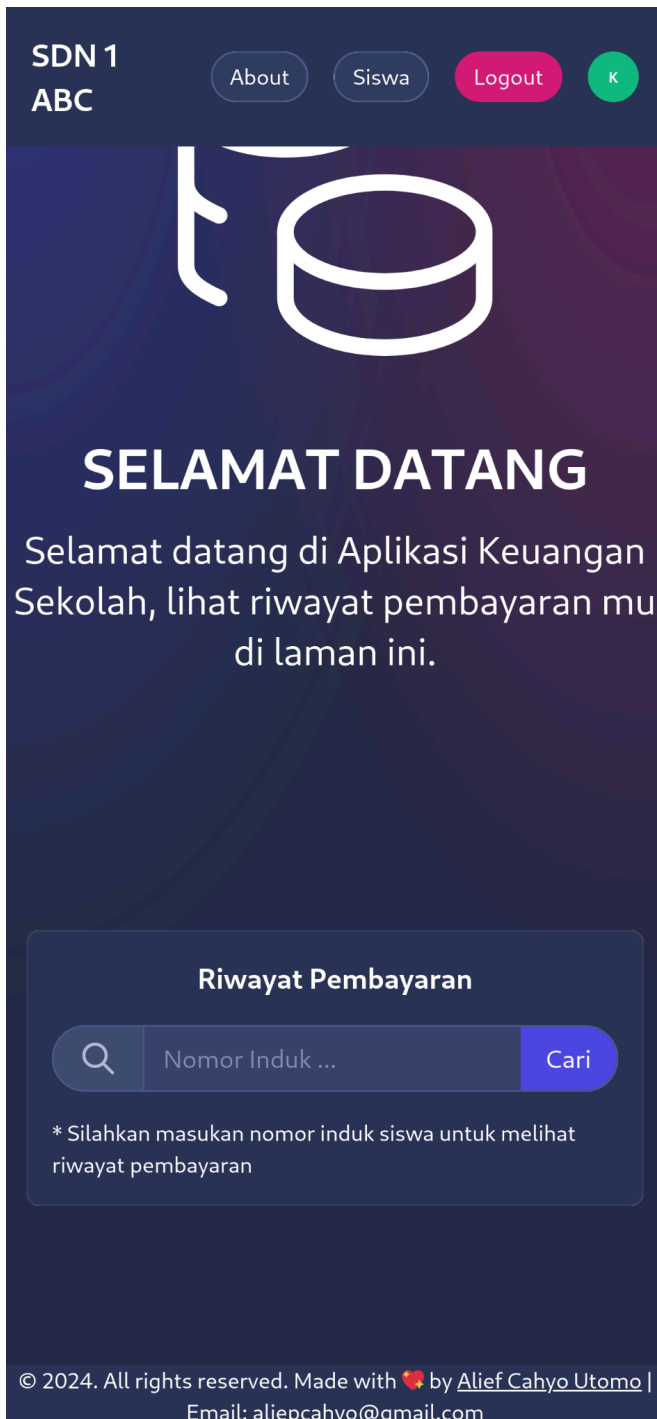


### 17. Mendapatkan history pembayaran siswa dengan nomor induk "1110001"



## Mobile Screenshot

1. Landing page, dimana siswa/wali bisa mengecek riwayat pembayaran.



2. Tabel siswa.



3. form siswa.

SDN 1  
ABC

AboutSiswaLogoutK

Tambah Siswa

Nomor Induk Siswa

Nomor Induk

Nama

Nama


Nama Wali

Nama Wali

No Handphone Wali

No Handphone Wali

Tambahkan

© 2024. All rights reserved. Made with  by [Alief Cahyo Utomo](#) |  
Email: [aliepcahyo@gmail.com](mailto:aliepcahyo@gmail.com)

4. tabel pembayaran.



