

# Perancangan dan Pengembangan Aplikasi Mobile untuk Pengelolaan Partitur Digital (STRATEGI)

Vardina Nava Madya Kasman<sup>#1</sup>, Sulaeman Santoso<sup>#2</sup>

<sup>#</sup>Program Studi Teknik Informatika, Universitas Kristen Maranatha  
Jln. Prof. Drg. Surya Sumantri No. 65, Sukawarna, Bandung, Indonesia

<sup>1</sup>1872025@maranatha.ac.id

<sup>2</sup>sulaeman.santoso@it.maranatha.edu

**Abstract** — Music sheet is a written form of a musical composition. With the sheet music, a musician can play or sing a song more accurately. However, physical music scores can be damaged or lost. To overcome this problem, an application for managing and viewing digital score documents was designed and developed. By using a mobile application that is connected to cloud storage, users can store and retrieve digital score data from anywhere and anytime as long as there is an internet connection. The app is built using the Flutter SDK and connected to a Firebase service. The result of this scientific work is a mobile application that can manage and display user's digital score data.

**Keywords**— digital music sheet, Firebase, Flutter SDK, mobile application.

## I. PENDAHULUAN

Partitur music adalah bentuk tertulis dari sebuah komposisi musik [1]. Pemusik menggunakan partitur musik untuk memainkan lagu dengan akurat. Umumnya partitur musik dicetak pada lembaran kertas atau buku. Hal tersebut menyebabkan partitur musik rentan rusak atau robek. Selain itu, untuk menyimpan partitur musik fisik diperlukan tempat penyimpanan dan perlu dibawa-bawa jika ingin digunakan di tempat lain. Setelah mempertimbangkan masalah tersebut, maka dikembangkanlah sebuah aplikasi untuk mengatur sekaligus membaca data partitur digital melalui perangkat *mobile*. Data partitur digital disimpan dalam bentuk dokumen PDF.

Aplikasi *mobile* dibuat untuk perangkat *android* dan menggunakan penyimpanan awan pada platform *Firebase*. Pengguna perlu membuat akun agar dapat menyimpan dan mengakses data partitur digital ke dalam penyimpanan awan. Keuntungan dari penyimpanan awan *Firebase* adalah data partitur digital tidak menempati memori penyimpanan perangkat *android* dan dapat diakses dari mana saja selama ada akun aplikasi dan koneksi internet. Sementara untuk pembuatan aplikasi pengelola dan pembaca partitur menggunakan SDK Flutter.

Berdasarkan hasil pengamatan terhadap hasil penelitian dengan topik serupa, belum ada aplikasi *mobile* yang secara khusus menangani pengelolaan data partitur digital. Oleh karena itu, dengan dibuatnya aplikasi *mobile* dalam penelitian ini diharapkan dapat menjadi solusi untuk permasalahan tersebut.

## II. KAJIAN LITERATUR

### A. Partitur Musik

Partitur musik adalah sebuah catatan atau panduan untuk memainkan sebuah komposisi musik. Partitur musik juga disebut dengan naskah atau tulisan musik [2]. Dengan adanya partitur musik, pemusik dapat membaca panduan untuk memainkan sebuah komposisi musik yang berupa not-not balok dan simbol-simbol lainnya.

### B. Portable Document File

*Portable Document File* atau yang umumnya disingkat sebagai PDF adalah file dokumen yang ringan atau mudah untuk dipindah-pindah dan dibawa-bawa. Tipe file tersebut dibuat oleh Adobe pada tahun 1992 dengan tujuan agar file dokumen berupa tulisan atau gambar dapat diakses tanpa memerlukan aplikasi, perangkat keras, atau sistem operasi tertentu [3].

### C. Flutter Software Development Kit

Flutter adalah sebuah *Software Development Kit*, yaitu satu paket untuk keperluan pengembangan perangkat lunak. Paket pengembangan aplikasi ini bersifat *open-source*. Flutter memungkinkan pengembangan aplikasi yang berkualitas native yang artinya adalah aplikasi yang dibangun dengan bahasa spesifik yang ditujukan untuk platform tertentu [4].

### D. Flutter Widget

Dalam pembuatan GUI (*Graphical User Interface*), Flutter menggunakan konsep *widget*. Setiap komponen GUI Flutter adalah *widget-widget*. Ada dua jenis *widget* pada Flutter, yaitu *Stateless* dan *Stateful*. *Stateless widget* adalah widget yang bersifat statis dan tidak dapat digambar ulang. *Stateful widget* adalah *widget* yang dapat digambar ulang oleh Flutter. Contohnya adalah bila ada perubahan data pada basis data, *widget* tersebut dapat diperbaharui isinya dengan basis data yang baru [5].

### E. Flutter Plugin

Plugin adalah program tambahan yang dapat menambahkan fungsionalitas dari kode program utama. Plugin yang digunakan pada penelitian ini adalah *plugin* untuk menampilkan PDF dan beberapa *plugin* dari *Firebase* untuk mengelola data [6].

### F. Firebase

*Firebase* menyediakan berbagai kemudahan untuk pengembang aplikasi yang didesain untuk mempercepat integrasi layanan berbasis penyimpanan awan dengan aplikasi *mobile* dan web. Setelah akuisisi, Google menggabungkan fitur-fitur *Firebase* dengan beberapa fitur dari *Google Cloud Platform*. Gabungan keduanya-lah yang saat ini dikenal sebagai *Firebase* [7].

*Firebase* dapat menyimpan dan secara *non-relational*, artinya antar tipe data tidak ada keterkaitan. Hal ini memungkinkan pengambilan dan penyimpanan data yang fleksibel, karena tidak membutuhkan bahasa kueri terstruktur (SQL) untuk mengambil dan mengelola data. Selain itu, penyimpanan data secara *non-relational* juga dapat mengurangi kompleksitas penyimpanan data. Untuk menambahkan data baru pun tidak perlu memikirkan data yang lain karena setiap data tidak ada keterkaitan sehingga penyimpanan data secara *non-relasional* memiliki skalabilitas yang tinggi [8].

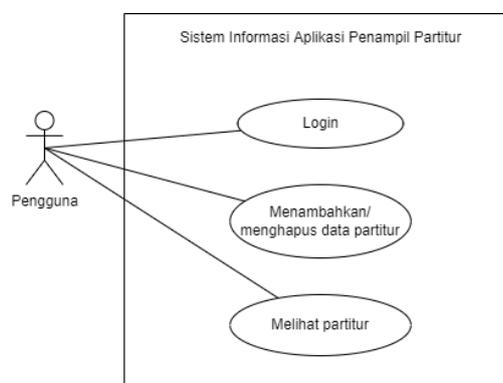
## III. ANALISIS DAN RANCANGAN

### A. Analisis Aplikasi

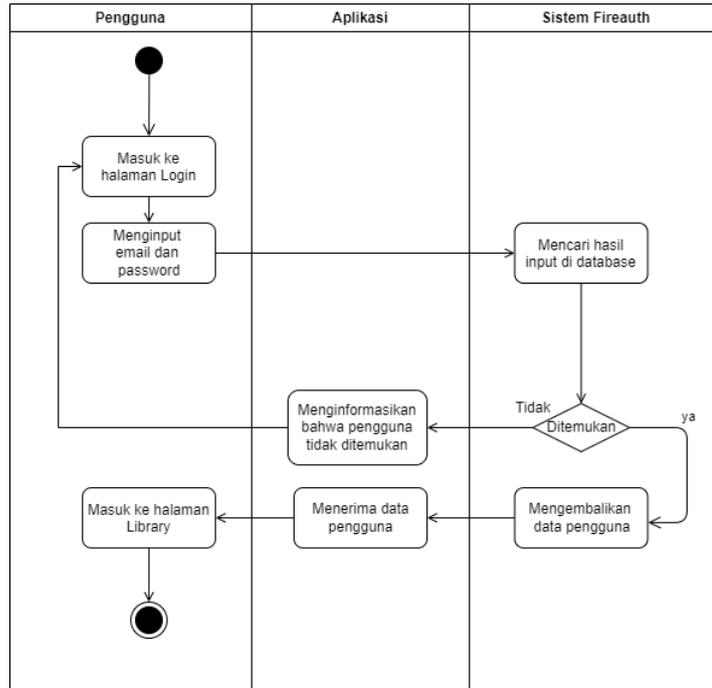
Aplikasi *mobile* pembaca dan pengelola file partitur digital dirancang dan dibangun untuk membantu para pemusik mengelola file partitur digital mereka. Aplikasi dapat membantu dengan menyediakan fitur untuk menampilkan file partitur dan fitur untuk mengelola file partitur secara daring. Data yang disimpan secara daring memungkinkan pengguna untuk mengakses file-file partiturnya dari perangkat *mobile* yang berbeda selama masih menggunakan akun yang sama.

### B. Cara Kerja Aplikasi

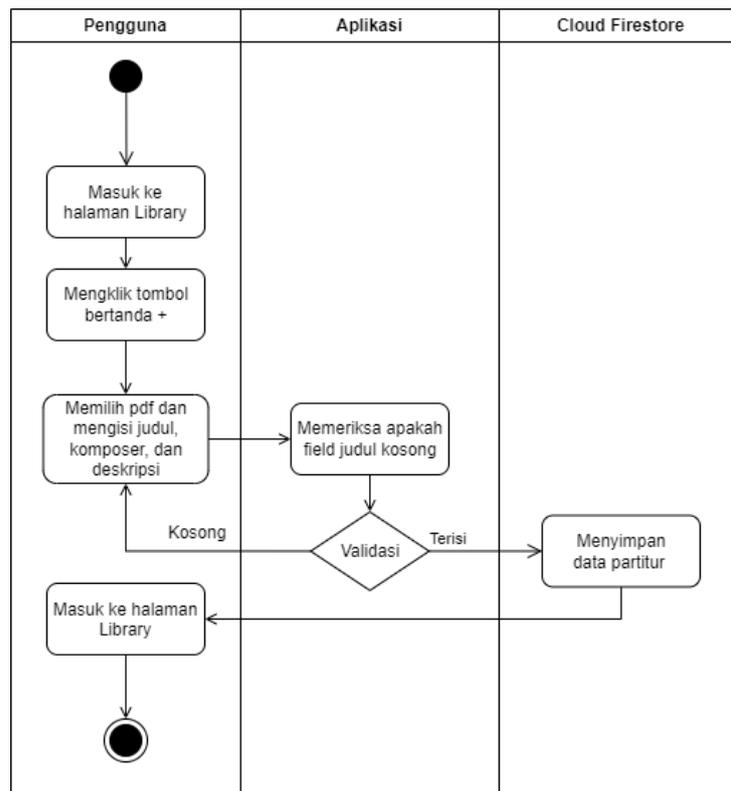
Aplikasi bertujuan untuk memungkinkan pengguna menyimpan dan menampilkan file partitur musik yang bertipe PDF. Alur kerja aplikasi dan fitur ditampilkan melalui Gambar 1. Pada diagram tersebut, tertera tiga fungsi utama dari aplikasi, yaitu login, menambahkan/menghapus data partitur, dan melihat partitur. Gambar 2, 3, dan 4 adalah diagram yang menjelaskan alur kerja dari masing-masing fitur tersebut.



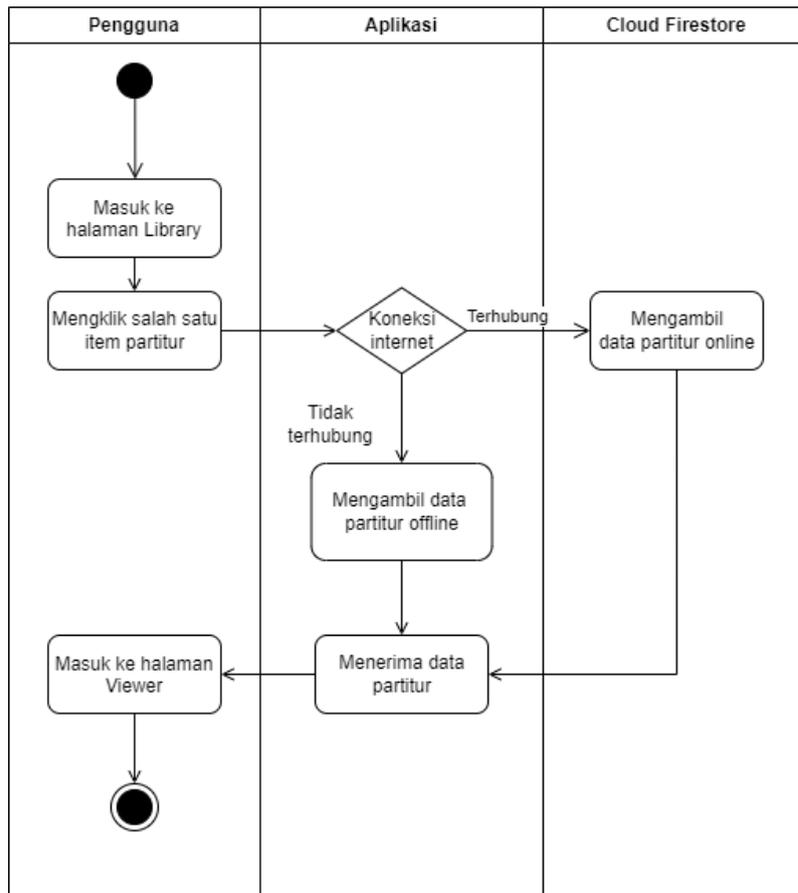
Gambar 1. Use Case Diagram



Gambar 2 Activity Diagram Login



Gambar 3. Activity Diagram Kelola Partitur



Gambar 4. Activity Diagram Melihat Partitur

C. Rancangan Basis Data

Untuk menyimpan data pengguna dan file partitur digital, aplikasi menggunakan layanan *Firestore*. Layanan *Firestore* yang digunakan adalah *Firestore* dan *Cloud Firestore*. Koleksi diberi nama dengan id unik pengguna (UID) dan pada setiap dokumen terdapat data. Dokumen diberi id berupa nilai acak yang dibuat secara otomatis oleh *Firestore*. Detail koleksi dan data yang disimpan dalam setiap dokumen ditunjukkan pada Tabel I.

TABEL I  
RANCANGAN PENYIMPANAN DOKUMEN DALAM KOLEKSO

Koleksi: (id unik pengguna)		
Nama Data	Tipe Data	Fungsi
Judul	String	Menyimpan judul partitur
Komposer	String	Menyimpan nama komposer dari partitur
Catatan	String	Menyimpan catatan dari partitur
Pdf	String	Menyimpan url dimana file partitur digital disimpan

#### IV. IMPLEMENTASI

##### A. Implementasi Aplikasi

Berikut adalah hasil implementasi dari analisis dan rancangan yang telah dijabarkan:

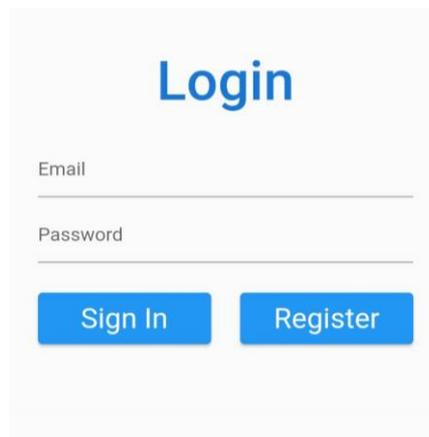
- Aplikasi dapat mengelola data partitur digital, yaitu menambah, mengubah, dan menghapus data partitur digital.
- Aplikasi sudah berhasil tersambung dengan penyimpanan awan Firebase, sehingga dapat menyimpan data ke dalam basis data dan menampilkan data yang terdapat di dalam basis data penyimpanan awan.
- Pengguna dapat membuat akun dengan cara memasukkan alamat email dan kata sandi.
- Pada penampil partitur, pengguna dapat menggeser, memperbesar, atau memperkecil tampilan dengan gestur jari.

Aplikasi *mobile* pembaca dan pengelola file partitur digital pada penelitian ini hanya menerima data dengan ekstensi PDF. Sebelum pengguna dapat menambahkan data partitur digital ke dalam aplikasi, pengguna perlu mengubah ekstensi file partitur digital menjadi PDF terlebih dahulu, bila ekstensi file partitur digital yang dimiliki bukan PDF.

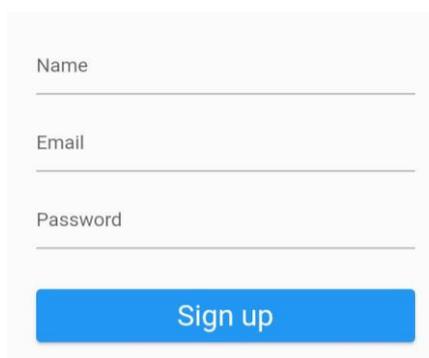
##### B. Login dan Register Akun

Ketika pengguna pertama kali masuk ke dalam aplikasi, halaman *login* akan muncul. Pada halaman *login* terdapat dua buah tombol, yaitu tombol untuk *login* (*Sign In*) dan tombol untuk registrasi akun (*Register*). Sebelum pengguna dapat masuk ke dalam aplikasi, pengguna tersebut perlu melakukan registrasi terlebih dahulu. Pada halaman *login* terdapat dua buah *TextField* dan dua buah tombol. Hasil tampilan halaman *login* ditunjukkan pada Gambar 5.

Pengguna yang sudah mengisi *TextField Email* dan *Password* dengan benar akan diarahkan ke halaman *Library* dan jika pengguna mengklik tombol *Register* maka akan diarahkan ke halaman registrasi. Hasil tampilan untuk halaman registrasi terdapat pada Gambar 6.



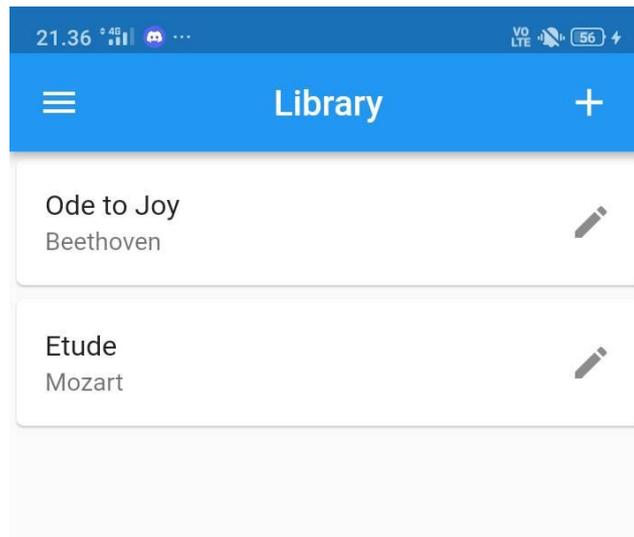
Gambar 5. Halaman Login



Gambar 6. Halaman Registrasi

##### C. Halaman Library

Halaman ini adalah halaman utama dari aplikasi setelah pengguna melakukan login. Pada halaman tersebut, pengguna dapat melihat partitur digital yang sudah diunggah ke basis data. Jika data partitur pengguna tidak ditemukan, maka akan muncul pesan bahwa data tidak ditemukan. Gambar 7 adalah hasil dari tampilan halaman *library*.

Gambar 7. Halaman *Library*

#### D. Implementasi Firebase

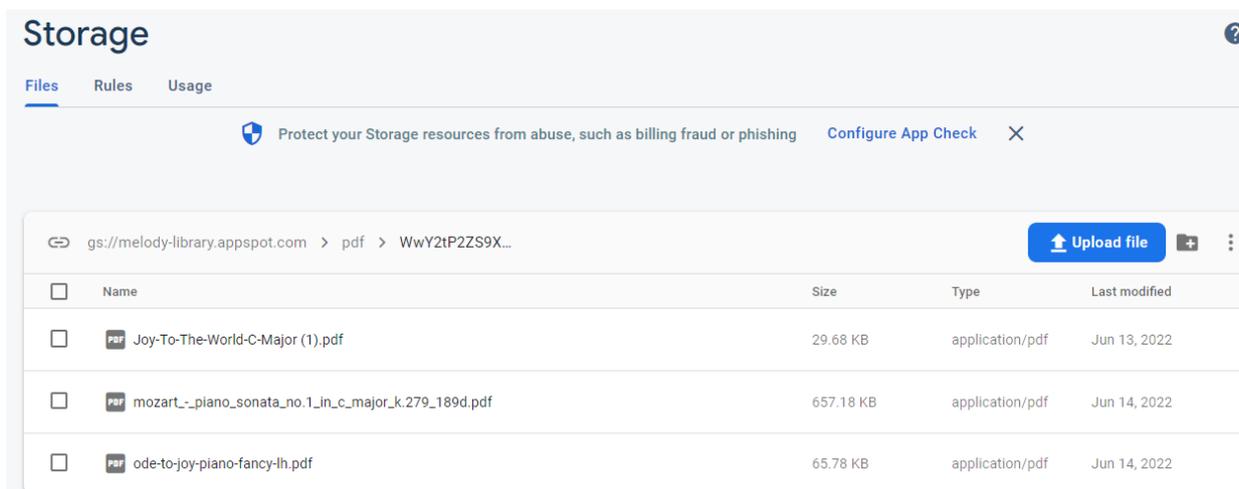
*Firebase* menyediakan berbagai layanan dalam kegiatan penyimpanan data ke basis data. Layanan yang digunakan oleh aplikasi adalah pengelola data pengguna *firebase auth*, penyimpanan media *firebase storage*, dan penyimpanan waktu nyata *cloud firestore*. Untuk detail data yang disimpan pada masing-masing layanan akan ditunjukkan melalui gambar. Gambar 8 menampilkan data pengguna yang disimpan ke *firebase auth*.

Identifier	Providers	Created ↓	Signed In	User UID
admin@gmail.com	✉	Jun 4, 2022	Jun 4, 2022	AdtNAnn66dblkjjjPFihZeF03KG2
vardina.nava@gmail.com	✉	Apr 11, 2022	Jun 14, 2022	WwY2tP2ZS9XQIJPHdOTMdiM9P...

Gambar 8. Implementasi Firebase Auth

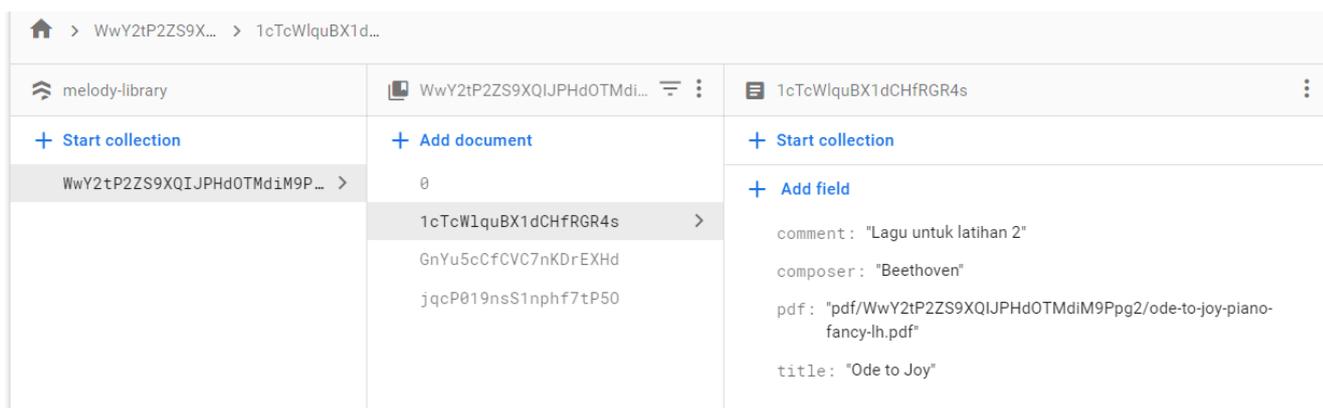
#### E. Penyimpanan Data Partitur Digital

Data yang dapat disimpan pada layanan *firebase storage* adalah berupa dokumen dengan berbagai ekstensi. Pada aplikasi ini, data yang disimpan ke dalam *storage* adalah file PDF dari partitur digital. Semua file partitur digital yang diunggah oleh pengguna diletakkan ke dalam satu folder yang dinamai dengan id unik pengguna, sehingga tidak akan tercampur dengan data partitur digital pengguna lain. Id unik pengguna dibuat secara otomatis oleh *firebase auth*. Gambar 9 adalah hasil penerapan *firebase storage*.



Gambar 8. Implementasi Firebase Auth

Data partitur digital berupa judul, komposer, dan keterangan disimpan ke dalam *cloud firestore*. Alamat dari lokasi PDF partitur digital juga disimpan ke dalam *cloud firestore*. Hasil implementasi dari *cloud firestore* dalam aplikasi ditunjukkan melalui Gambar 9. Nama koleksi yang digunakan adalah id unik pengguna dan id adalah nomor acak yang dibuat oleh *cloud firestore*.

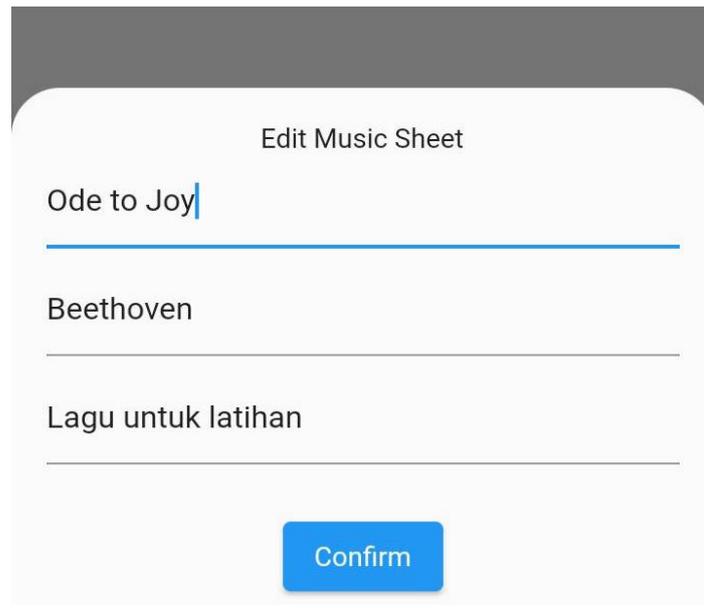


Gambar 9. Implementasi Cloud Firestore

#### F. Fitur Pengelola Data Partitur Digital

Fitur untuk mengelola data partitur digital adalah fitur untuk menambah, mengubah, atau menghapus data partitur digital. Fitur untuk menambah data partitur digital adalah melalui tombol dengan ikon (+) dan untuk mengubah data partitur digital adalah dengan menggunakan tombol dengan ikon pensil. Fitur untuk menghapus data partitur digital dapat ditemukan pada jendela apung yang akan muncul jika pengguna mengklik salah satu data partitur digital.

Fitur untuk mengubah data juga menggunakan formulir yang serupa. Dari atas ke bawah, daya yang diminta adalah judul, komposer, dan komentar/keterangan. Formulir *bottom sheet* untuk mengisi data partitur baru atau mengubah data partitur yang sudah ada adalah seperti pada Gambar 10.



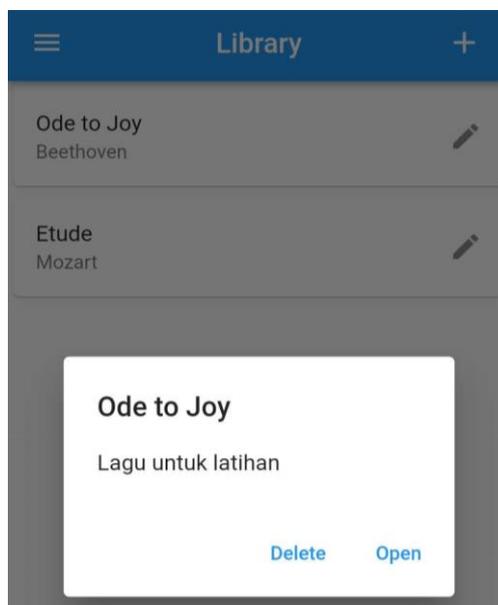
Form titled "Edit Music Sheet" with three input fields and a "Confirm" button.

- Field 1: Ode to Joy
- Field 2: Beethoven
- Field 3: Lagu untuk latihan
- Button: Confirm

Gambar 10. Formulir Penambahan/Perubahan Data

#### G. Fitur Penampil Partitur

Pengguna dapat mengklik data partitur digital pada halaman library untuk memunculkan sebuah jendela apung. Pada jendela apung tersebut terdapat dua buah tombol, yaitu tombol untuk menghapus data partitur digital yang diklik dan tombol untuk menampilkan file PDF partitur digital. Tampilan jendela apung adalah seperti pada Gambar 11.



Gambar 11. Tampilan Jendela Apung

Jika tombol "Open" diklik maka tampilan layar akan berpindah ke penampil file PDF partitur digital. Pada halaman ini, orientasi layar adalah horizontal/*landscape* agar partitur digital lebih mudah dibaca dari kiri ke kanan. Aplikasi akan mengambil data partitur digital dari *cloud firestore*, kemudian membuat url dari data lokasi yang disimpan pada *cloud*

firestore. Aplikasi kemudian menyimpan PDF dalam bentuk cache yang dapat dihapus oleh pengguna dengan aman. Hasil implementasi fitur penampil partitur digital ditunjukkan pada Gambar 12.



Gambar 12. Implementasi Halaman Viewer

## V. PENGUJIAN

Berikut ini merupakan hasil pengujian dari aplikasi mobile pembaca partitur:

### A. Pengujian Fitur Login

Tabel II adalah pengujian yang dilakukan terhadap fitur *login* yang ada di halaman *login*. yang telah dijabarkan:

TABEL II  
HASIL PENGUJIAN FITUR PADA HALAMAN LOGIN

No	Kasus Uji	Hasil Harapan	Hasil Pengamatan	Simpulan
1	Email : Username Password : 123456	Muncul peringatan Email	Muncul peringatan Email tidak sesuai	Sesuai
2	Email : (spasi) Password : 123456	Muncul peringatan Email	Muncul peringatan Email tidak sesuai	Sesuai
3	Email : (email tidak terdaftar) Password 123456	Kembali lagi ke halaman awal	Ketika login gagal, diarahkan ke halaman awal	Sesuai
4	Email : (spasi) Password : (spasi)	Muncul peringatan	Muncul pesan email tidak sesuai dan password minimal 6 digit	Sesuai
5	Email : (kosong) Password : (kosong)	Muncul peringatan	Muncul pesan email tidak boleh kosong dan password minimal 6 digit	Sesuai
6	Email : admin@gmail.com Password : 123456	Diarahkan ke halaman Library	Halaman Library muncul	Sesuai

### B. Pengujian Fitur pada Halaman Library

Tabel III adalah pengujian yang dilakukan terhadap fitur *insert* dan *update* data partitur digital yang ada di halaman *Library*.

TABEL III  
HASIL PENGUJIAN FITUR PADA HALAMAN LIBRARY

No	Kasus Uji	Hasil Harapan	Hasil Pengamatan	Simpulan
<i>1) Pengujian Fitur Insert</i>				
1	Tidak mengisi apa-apa, memilih file partitur, lalu klik tombol	Data ditambah ke dalam basis data	Muncul data kosong pada halaman Library dan pada basis data	Sesuai
2	Mengisi field lalu tidak memilih file partitur	Menutup bottom sheet untuk menambah data	Bottom sheet hilang dan kembali ke halaman Library	Sesuai
3	Mengisi field lalu memilih file partitur	Data ditambah ke dalam basis data	Muncul data kosong pada halaman Library dan pada basis data	Sesuai
<i>2) Pengujian Fitur Update</i>				
4	Tidak merubah apa-apa lalu klik tombol	Tidak terjadi perubahan pada tabel maupun basis data	Tidak terjadi perubahan pada tabel maupun basis data	Sesuai
5	Title: (kosong) Composer: (kosong) Comment: (kosong)	Data menjadi kosong	Muncul data tanpa judul, composer, dan komentar	Sesuai
6	Title: A Composer: B Comment: C	Data berubah sesuai dengan nilai yang di-inputkan	Muncul data dengan judul A, composer B, dan komentar C	Sesuai

## C. Pengujian Fitur AlertDialog

Tabel IV adalah pengujian yang dilakukan terhadap fitur jendela baru yang muncul ketika pengguna mengklik salah satu data pada ListView. Terdapat 2 tombol pada jendela tersebut, yaitu tombol *Delete* dan tombol *Open*.

TABEL IV  
HASIL PENGUJIAN FITUR ALERTDIALOG

No	Kasus Uji	Hasil Harapan	Hasil Pengamatan	Simpulan
<i>1) Pengujian Tombol Delete</i>				
1	Klik tombol delete, lalu tidak mengklik apa-apa lagi	Muncul snackbar dan data tidak dihapus	Muncul snackbar konfirmasi penghapusan dan Data masih ada pada basis data dan ListView	Sesuai
2	Klik tombol delete, lalu konfirmasi delete pada snackbar	Data partitur dan file partitur digital akan dihapus	Data partitur pada Firestore dan Storage dihapus dan tidak muncul pada ListView	Sesuai
<i>2) Pengujian Tombol Open</i>				
3	Klik tombol open	Viewer untuk file pdf tersebut akan muncul	File pdf muncul di layar	Sesuai
4	Mengklik dimana saja di luar AlertDialog	AlertDialog akan ditutup	Kembali ke halaman Library	Sesuai

## VI. KESIMPULAN

Berdasarkan pengujian dari hasil implementasi rancangan, dapat disimpulkan bahwa aplikasi yang dihasilkan telah berjalan sesuai dengan rancangan yang telah dijabarkan, yaitu dapat mengelola file partitur digital dan menampilkan file partitur digital yang disimpan ke dalam penyimpanan awan *Firebase*.

Aplikasi mobile tersebut sudah berjalan sesuai dengan rancangan, tetapi masih memiliki banyak ruang untuk pengembangan lebih lanjut. Contohnya adalah membuat versi IOS dari aplikasi, meningkatkan tampilan aplikasi, dan pengembangan terhadap fitur penampil partitur digital agar aplikasi dapat menjadi lebih baik lagi.

## DAFTAR PUSTAKA

- [1] A. M. W. Aru, E-modul seni budaya kelas XI, 2018.
- [2] R. J. Iskandar, "PERANCANGAN APLIKASI PEMBELAJARAN NOTASI DAN TANGGA NADA DENGAN PARTITUR BERBASIS ANDROID," *INTEKSIS*, no. 2, p. 2, 2016.
- [3] Adobe, "What is a PDF? Portable Document Format | Adobe Acrobat DC," [www.adobe.com](http://www.adobe.com), [Online]. Available: <https://www.adobe.com/acrobat/about-adobe-pdf.html>. [Accessed 20 May 2022].
- [4] L. Dagne, Flutter for cross-platform App and SDK development, 2019.
- [5] N. Smyth, *Flutter Essentials-Android Edition*, Payload Media, Inc., 2017.
- [6] M. Fahrurrozi and S. N. Azhari, "Rancang Bangun Plugin Protégé Menggunakan Ekspresi SPARQL-DL Dengan Masukan Bahasa Alami," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, vol. 11, no. 2, pp. 155-164, 2017.
- [7] E. Windmill, *Flutter in action*, Simon and Schuster, 2020.
- [8] Wicaksono, R. F. Tri, E. K. Nurnawati and R. Y. Ariyana, "Perancangan Sistem Informasi Bank Darah Menggunakan Basis Data NoSQL," *SATESI: Jurnal Sains Teknologi dan Sistem Informasi*, vol. 2, no. 1, pp. 38-46, 2022.