

Perancangan Aplikasi Jadwal Konser Musik (Kick Gigs) Berbasis *Mobile*

Antonius Titis Adhitiyanto Bintang Timur¹⁾*, Magdalena A. Ineke Pakereng²⁾

¹⁾²⁾Teknik Informatika, Teknologi Informasi, Universitas Kristen Satya Wacana

Correspondence author: bintangtimur90an@gmail.com

DOI: <https://doi.org/10.37012/jtik.v7i2.599>

Abstrak

Permusikan di Indonesia kini telah mengalami perkembangan yang pesat dalam beberapa dekade ini. Jadwal konser yang sering digunakan saat ini hanya melalui social media. Seiring perkembangan teknologi, internet memunculkan perubahan ruang kreatifitas musisi dalam menyebarkan informasi panggungnya menjadi lebih mudah. Tujuan dari penelitian ini adalah pembuatan aplikasi jadwal konser *live performance* agar dapat digunakan untuk mempermudah masyarakat dalam memperoleh informasi jadwal panggung dengan lebih mudah dan cepat. Serta memudahkan musisi dalam menyebarkan informasi panggungnya ke masyarakat. Penelitian ini menghasilkan aplikasi jadwal konser bernama *Kick Gigs* berbasis *mobile* yang dapat berjalan sesuai dengan apa yang diharapkan yang dibangun menggunakan bahasa pemrograman *Dart 2.13* dengan *framework flutter 2.0* dan basis data *Firestore*. Aplikasi *Mobile* ini diharapkan dapat membantu musisi kecil dalam pengembangan karir musiknya lebih baik lagi. Selain itu dapat membantu pengusaha cafe dan restoran untuk turut berdamai bersama musisi dalam mengembangkan bisnis dan karya musik dengan pertunjukan *live performance* yang mengundang lebih banyak lagi pengunjung.

Kata Kunci: *Flutter, Android, Firebase Storage*

Abstract

Music in Indonesia has now experienced rapid development in recent decades. Concert schedules that are often used today are only through social media. Along with the development of technology, the internet has created a space to create creativity in disseminating stage information more easily. The purpose of this research is to make a live performance concert schedule application so that it can be used to facilitate the public in obtaining stage schedule information more easily and quickly. As well as facilitate musicians in disseminating stage information to the public. This research produces a mobile-based concert schedule application called Kick Gigs that can run as expected, built using the dart 2.13 programming language with the flutter 2.0 framework and the Firestore database. This Mobile Application is expected to help small musicians in developing their music career even better. In addition, it can help cafe and restaurant entrepreneurs to participate with musicians in developing their business and musical works with live performances that invite many visitors.

Keywords: *Flutter, Android, Firestore, Music*

PENDAHULUAN

Permusikan di Indonesia kini telah mengalami perkembangan yang pesat dalam beberapa dekade ini. Banyaknya jadwal musik di tiap daerah di Indonesia membuat banyak musisi muncul ke kancah Nasional hingga Internasional. Seiring perkembangan teknologi, internet memunculkan perubahan ruang kreatifitas untuk musisi yang harus selalu berinovasi untuk menghadirkan jadwal konser yang lebih cepat dan mudah ke masyarakat luas.

Internet memiliki peranan penting dalam kemajuan musisi dalam berkarya, dengan adanya Internet, kini semua orang dapat memiliki informasi mengenai jadwal panggung grup band serta musisi didalamnya.

Adanya perkembangan teknologi serta perkembangan dunia digital masa kini memudahkan masyarakat untuk memperoleh informasi dari manapun, dari *handphone* contohnya. Dengan teknologi, manusia dapat melakukan hal yang sulit menjadi lebih mudah, salah satu perkembangan teknologi yang populer adalah *Mobile App*. Teknologi *Mobile App* yang terus diperbaharui, sehingga pengguna dapat mengakses dan mengunggah informasi hanya dalam satu genggam.

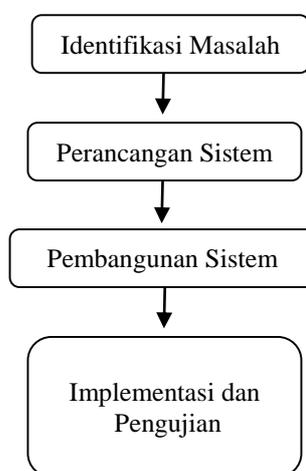
Live performance sekarang ini masih menjadi suatu pertunjukan yang memiliki peran sosial yang penting. Faktor pendukung selain dari sisi musisi, ada juga pemilik café dan restoran. Mereka seringkali mengadakan pertunjukan *live music* beberapa kali dalam seminggu untuk menarik pelanggan ke tempat mereka [1].

Tujuan dari penelitian ini adalah pembuatan aplikasi jadwal konser *live performance* yang dapat digunakan seluruh musisi sebagai user dalam aplikasi ini. Adapun aplikasi berbasis *mobile* yang dibangun memberikan beberapa manfaat seperti mempermudah musisi untuk menyebarluaskan informasi panggungnya ke penikmat musik. Aplikasi ini juga bisa diakses kapanpun dan dimanapun [2]. Selain itu, aplikasi *mobile* yang dibangun dapat memfasilitasi musisi kecil untuk memperluas jaringan antar musisi yang tersebar di Indonesia [3].

Berdasarkan latar belakang yang ada, maka dilakukan penelitian yang bertujuan untuk membuat suatu Aplikasi *Mobile* yang dapat membantu masyarakat dalam memperoleh informasi jadwal panggung dengan lebih mudah dan cepat.

METODE PENELITIAN

Dalam penelitian ini menerapkan metode SDLC (*System Development Life Cycle*) dengan model waterfall. Model Waterfall yaitu dengan perancangan suatu sistem informasi yang diterapkan secara sistematis dan sekuensial [8]. Penelitian ini bertujuan untuk membuat sebuah aplikasi Jadwal konser (*Kick Gigs*). Dalam penelitian ini dibagi menjadi 4 tahapan informasi. Penelitian ini bertujuan untuk membuat sebuah aplikasi Jadwal Konser (*Kick Gigs*) yang dalam penelitian ini dibagi menjadi 4 tahapan, yaitu, (1) Identifikasi masalah, (2) Perancangan dan desain sistem, (3) Pembangunan sistem, (4) Implementasi dan Pengujian sistem. Pada perancangan sistem informasi menggunakan tahapan penelitian seperti pada Gambar 1.



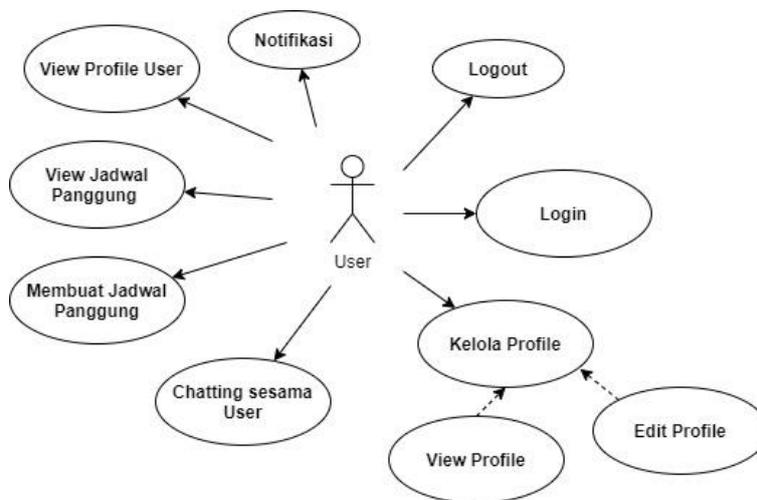
Gambar 1 Tahapan Penelitian

Tahapan awal penelitian adalah identifikasi masalah. Tahapan penulisan mengidentifikasi masalah yang sering terjadi pada kalangan musisi kecil bahwa mereka sangat sulit meramaikan pertunjukan mereka saat *Live Performance*. Karena kurangnya persebaran informasi yang merata yang biasanya hanya terjadi jika musisi tersebut belum memiliki nama yang besar.

Tahapan kedua dilakukan perancangan aplikasi mobile untuk menjawab permasalahan yang ada musisi kecil dalam aktivitas *live performance*-nya. Perancangan ini akan menggunakan metode pemodelan yang berorientasi objek yaitu UML. UML adalah Bahasa standar yang banyak digunakan untuk menentukan kebutuhan, menganalisis, dan desain, serta menjelaskan arsitektur pemrograman berorientasi objek di dunia permusikan. UML adalah pemodelan komputer dan Bahasa komunikasi visual yang menggunakan diagram dan dukungan teks serta penggunaan UML tidak hanya terbatas pada satu metode saja, tetapi sebagian besar digunakan dalam metodologi berorientasi objek [9].

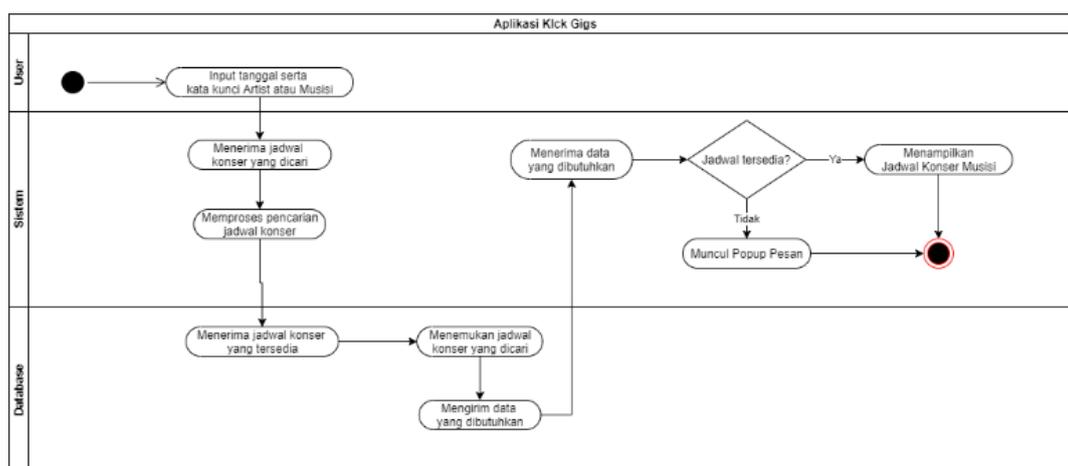
Pada tahap ketiga, dilakukan pembuatan aplikasi yang sudah dirancang pada tahap sebelumnya. Aplikasi ini dibangun dengan basis *mobile* dengan *framework Flutter 2.2*, yaitu sebuah *Framework* yang dikembangkan oleh perusahaan teknologi bernama *Google.inc* yang dapat dikembangkan serta dijalankan pada platform besar Android [6]. Bahasa pemrograman yang dipakai adalah *Dart 2.13* menggunakan basis data *Firestore*.

Use Case dideskripsikan secara tekstual dalam bentuk *use case scenario* untuk menjelaskan interaksi yang terjadi antara actor dengan sistem. Selanjutnya, *use case scenario* di ilustrasikan secara visual dalam bentuk *use case diagram* untuk menggambarkan konteks dari sistem yang dikembangkan[10]. *Use Case diagram* dibuat sebagai analisa awal membangun suatu aplikasi untuk menjelaskan fungsi secara sederhana dari sudut pandang *user*. Pada Gambar 2 menjelaskan, seorang *user* dapat membuat suatu jadwal *live performancenya* sendiri dengan mudah. Memberikan keterangan waktu, poster, nama Artis/Musisi dan tanggal. Jadi, setiap *user* akan memiliki hak untuk membuat jadwal *live performance* nya dengan mudah dan cepat. Jadwal *user* ini akan ditampung menjadi 1 tempat yakni di kolom *Explore*



Gambar 2 Use Case Diagram Sistem

Activity Diagram merupakan rancangan aliran kerja dan aktivitas dalam sebuah sistem yang akan dijalankan. *Activity Diagram* juga digunakan menampilkan proses model sistem. *Activity Diagram* adalah salah satu yang paling mudah pada UML diagram karena mereka menggunakan simbol-simbol yang mirip dengan notasi *flowchart* [11]. *Activity Diagram* ini menggambarkan rangkaian aktivitas dalam sistem yang berjalan pada sistem aplikasi Jadwal Konser *Kick Gigs*.



Gambar 3 Activity Diagram Sistem Aplikasi Mobile Kick Gigs

Pada Gambar 3 menunjukkan proses diagram alur dari aplikasi *mobile Kick Gigs*, pada diagram tersebut dijelaskan, pengguna dapat mencari Jadwal Konser favoritnya dengan kata kunci Artis/Musisi beserta tanggal konser itu berlangsung.

Tahap terakhir, aplikasi yang telah selesai dibangun dilakukan pengujian sistem untuk mengetahui apakah aplikasi berhasil berjalan sesuai dengan perancangan yang dilakukan. Pengujian sistem menggunakan metode *black box testing* yang merupakan salah satu teknik pengujian perangkat lunak. *Black box testing* digunakan untuk menentukan fungsionalitas dan output suatu sistem. Fokus utama dari pengujian *black box* adalah menyesuaikan standar program sebuah aplikasi apabila terdapat *bug* pada program aplikasi tersebut.

HASIL DAN PEMBAHASAN

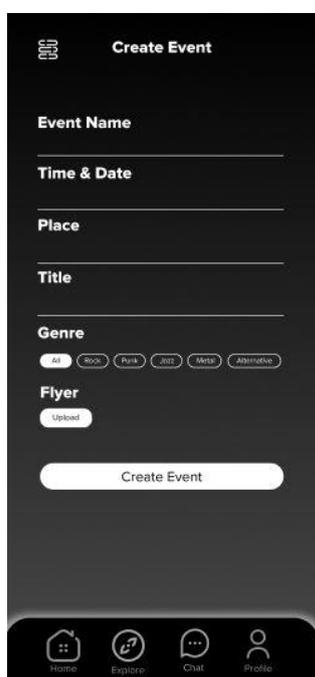


Gambar 4 Login



Gambar 5 Menu Utama

Pada tahap pengimplementasian sistem rancangan yang sudah dibuat, berikut beberapa tampilan aplikasi *mobile Kick Gigs* yang telah di bangun. Gambar 4 merupakan tampilan halaman *form login* melalui *Google*. Pada halaman *login* ini, *user* harus mengkoneksikan *email google* untuk dapat masuk ke dalam menu utama. Gambar 5 merupakan menu utama yang menampilkan jadwal *live performance* (konser) yang akan dicari *user*. Terdapat *search artist* dan *tanggal* penyelenggaraan *event* itu akan digelar. Menu utama juga menampilkan “*Previous Gigs Event*” yang artinya menampilkan *event* yang sebelumnya sudah selesai digelar.



Gambar 6 Create Event

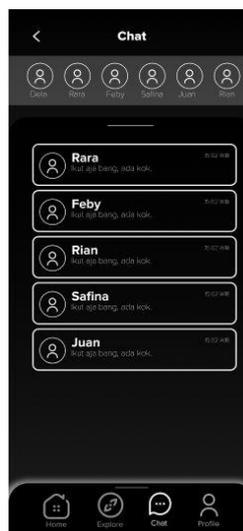


Gambar 7 Menu Utama Explore

Gambar 6 merupakan tampilan dari “*Create Event*” yakni membuat acara panggung. Fitur ini digunakan musisi untuk memberikan informasi jadwal panggungnya. Tersedia Nama *Event*, Tanggal, Tempat, dan Caption, Genre, serta *Upload* Gambar. Gambar 7 merupakan *Explore* dari aplikasi *Kick Gigs*. Disini pengguna sekaligus musisi dapat mengetahui jadwal *event* sesama musisi melalui kolom *explore* ini. Dalam halaman ini, *user* bisa melihat *explore* Aplikasi *Kick Gigs* dengan filter berdasarkan *genre* musik. *Explore* sebagai wadah seluruh data yang diinputkan pada *create event* dan fitur *search* Artis/Musisi dengan tanggal di menu *Homepage*.



Gambar 8 Menu Utama Profile



Gambar 9 Menu Utama Chat

Gambar 8 merupakan *Menu Utama Profile*. Dimenu ini *user/musisi* dapat melihat *postingan* yang diunggah oleh *user/musisi* lain, bahkan mereka juga dapat melihat event yang telah di favoritkan. Di kolom profil ini juga disediakan *button chat* untuk memudahkan pengguna dalam berkomunikasi jika ada informasi yang kurang jelas dari penyelenggara *event* melalui user yang telah memposting acaranya. Gambar 9 merupakan halaman *chat* dari aplikasi *Kick Gigs*. *Widget* atas adalah sebagai daftar kontak pengguna aplikasi *Kick Gigs*. Mereka dapat saling berinteraksi tanpa batasan.



Gambar 10 Notification

Gambar 10 merupakan Notifikasi. Notifikasi berisi informasi jadwal konser yang sebelumnya telah difavoritkan oleh pengguna. Notifikasi akan muncul saat 1 hari sebelum acara itu dimulai.

Kode Program 1 *Add Event* Pada Aplikasi *Kick Gigs*

```
addEvent(  
    {String picture,  
     String artist,  
     String description,  
     String genre,  
     String title}) {  
  
    Uuid uuid = Uuid();  
    String id = uuid.v4().toString();  
  
    firebase.addChat(id, {  
        "id": id,  
        "genre": genre,  
        "title": title,  
        "picture": picture,  
        "artist": artist,  
        "description": description,  
        "bookmark": false,  
        "created_at": FieldValue.serverTimestamp(),  
    });  
}
```

Pada Kode Program 1 merupakan kode untuk menjalankan fungsi *Add Event*. Setiap variable dapat memiliki nilai *null* {*String variable*}, sehingga *user* dapat memilih untuk tidak mengisi data sesuai *field* pada *form* yang nanti akan dijadikan informasi *event*. Lalu data yang telah dikirim ke *firestore* sesuai dengan data model yang tertera pada *function*. variable *'id'* didapat dari *generate ID* oleh *library package Uuid* untuk menentukan *ID* secara *dynamic* setiap kali fungsi *addEvent* dijalankan, *id* yang ter-generate digunakan untuk pada *firestore* yang menentukan pemanggilan data *query event*, *query data* pada *firestore* juga dapat menyimpan *link path* seperti pada data variable *String Picture* untuk memanggil data *path* pada *firebase storage* nantinya. Lalu data yang telah dikirim ke *firestore* sesuai dengan data model yang tertera pada *function*. Misalnya di *firestore* terdapat data *picture* : *"user/profile-picture/gambar.jpg"* dari *string* akan di *hit* ke *storage* yang telah tersedia *folder user* ke *folder profile-picture*, untuk diambil link *download* ke *gambar.jpg*.

Tahapan terakhir adalah pengujian sistem yang dilakukan menggunakan *black box testing*. Pengujian pada suatu aplikasi bertujuan untuk memeriksa apakah sebuah program telah berjalan dengan semestinya atau masih ada kesalahan yang harus diperbaiki agar program yang dibuat akan menjadi program yang memiliki kualitas yang baik [12]. *Software Testing* adalah elemen kritis dari jaminan kualitas perangkat lunak yang merepresentasikan

kajian pokok dari spesifikasi, desain, dan pengkodean. Peningkatan visibilitas (kemampuan perangkat lunak sebagai suatu elemen sistem dan ”biaya” yang muncul akibat kegagalan perangkat lunak, dilakukannya perencanaan yang baik melalui pengujian yang teliti [13]. Hasil pengujian aplikasi *mobile* Kick Gigs yang dibuat dengan *black box testing* dapat dilihat Pada Tabel 1.

Tabel 1 Hasil *Black Box Testing*

| Fungsi Yang Diuji | Kondisi | Output Yang Diharapkan | Output Yang Dihasilkan | Status |
|-----------------------|--|--|--|--------|
| Login | <i>Email</i> dan <i>password</i> benar | Sistem menerima akses <i>login</i> lalu masuk ke halaman menu utama | Sistem menerima akses <i>login</i> lalu masuk ke halaman menu utama | Valid |
| | <i>Email</i> salah | Sistem menolak akses <i>login</i> lalu menampilkan pesan “ <i>Couldn’t find your Google Account</i> ” | Sistem menolak akses <i>login</i> lalu menampilkan pesan “ <i>Couldn’t find your Google Account</i> ” | Valid |
| | <i>Email</i> benar dan <i>password</i> salah | Sistem menolak akses <i>login</i> lalu menampilkan pesan “ <i>Wrong Password. Try again or click Forgot password to reset it</i> ” | Sistem menolak akses <i>login</i> lalu menampilkan pesan “ <i>Wrong Password. Try again or click Forgot password to reset it</i> ” | Valid |
| Icon Home | Klik icon pada Homepage. | Home Menu Sistem menampilkan Homepage berisi kolom search Musisi, Tanggal dan Previous Gigs Event | Sistem berhasil menampilkan data yang telah diharapkan | Valid |
| Icon Explore | Klik icon pada Homepage | Explore menu Sistem menampilkan halaman explore yang berisi beranda profil musisi yang telah memposting jadwal panggunanya | Sistem berhasil menampilkan nama user, flyer event, judul, tanggal acara, Icon Favorit serta button See More | Valid |
| | Cek button See more | Sistem berhasil menampilkan Detail halaman Explore | Sistem berhasil menampilkan Detail halaman explore detail | Valid |
| | Cek icon Favourite | Event yang dipilih berhasil di favoritkan dan masuk ke halaman Profil | Event yang dipilih berhasil di favoritkan dan masuk ke halaman Profil | Valid |
| Icon Chat | Klik Icon Chat pada menu Homepage | Sistem menampilkan halaman chat | Sistem berhasil menampilkan widget daftar kontak dan kolom chatting | Valid |
| Icon Notification | Cek button Notification | Sistem menampilkan Notification artist yang telah di favoritkan. | Sistem berhasil menampilkan Notification artist yang telah di favoritkan sebelumnya. | Valid |
| Ubah Data <i>User</i> | Form diisi dengan benar | Sukses tambah data dan menampilkan pesan “ <i>Data user</i> berhasil diubah” | Sukses tambah data dan menampilkan pesan “ <i>Data user</i> berhasil diubah” | Valid |

| | | | | |
|-------------------------|---|---|--|-------|
| | Input <i>username</i> yang sudah terdaftar di <i>database</i> | Gagal tambah data dan menampilkan pesan “Username sudah dipakai” | Gagal tambah data dan menampilkan pesan “Username sudah dipakai” | Valid |
| Hapus Data <i>User</i> | Data yang akan dihapus dipilih | Sistem menghapus data dan menampilkan pesan “ <i>User</i> berhasil dihapus” | Sistem menghapus data dan menampilkan pesan “ <i>User</i> berhasil dihapus” | Valid |
| Create Event | Cek Button Create Event | Sistem menambah data dan menampilkan pesan “Event created” | Sistem menambah data dan menampilkan pesan “Event created” | Valid |
| | Cek Field Event Name, Time Data, Place Title, Genre. | Field akan terisi sesuai dengan yang user inginkan | Field terisi sesuai dengan yang user inginkan | Valid |
| | Cek button Upload Foto Flyer | User berhasil menambahkan foto untuk keperluan informasi jadwal panggung | User mampu menambahkan foto untuk keperluan informasi jadwal panggung | Valid |
| Cari Jadwal Musisi | Data ditemukan | Sistem menampilkan Jadwal menurut Artist dan Tanggal yang dicari | Sistem berhasil menampilkan informasi jadwal panggung artis sesuai tanggal yang dicari | Valid |
| | Data tidak ditemukan | Muncul respon “Event Not found” | Muncul respon “Event Not found” | Valid |
| Icon Profile | Cek button chat | Sistem akan merespon mengarahkan ke halaman chat. | Sistem akan merespon mengarahkan ke halaman chat. | Valid |
| | Cek button Edit Profile | Sistem akan merespon dengan mengarahkan ke halaman Settings | Sistem berhasil merespon dengan mengarahkan user ke halaman Settings | Valid |
| | Cek button Favourite | Sistem akan menampilkan list favourite dari profile. | Sistem berhasil menampilkan list favorit profile | Valid |
| | Cek button Post | Sistem akan menampilkan list postingan dari profile pengguna. | Sistem berhasil menampilkan list postingan dari profil pengguna | Valid |
| Icon Menu pada Homepage | Cek button settings | Sistem akan merespon mengarahkan pada ke halaman Settings | Sistem berhasil merespon dengan mengarahkan user ke halaman settings. | Valid |
| | Cek button Contact Us | Sistem akan merespon, mengarahkan user ke halaman contact us. | Sistem berhasil merespon dengan mengarahkan user ke halaman contact us. | Valid |

SIMPULAN

Penelitian ini menghasilkan Aplikasi Jadwal Konser bernama Kick Gigs berbasis mobile yang dapat berjalan sesuai dengan apa yang diharapkan yang dibangun menggunakan *framework flutter* dan basis data *Firestore*. Di dalam aplikasi jadwal konser ini, *user* sebagai musisi dapat melakukan pembuatan *event* dengan cepat dan mudah. Aplikasi *Mobile* ini diharapkan dapat membantu musisi kecil dalam pengembangan karir musiknya lebih baik lagi. Selain itu dapat membantu pengusaha cafe dan restoran untuk turut berdampingan bersama musisi dalam mengembangkan bisnis dan karya musik dengan pertunjukan *live performance* yang mengundang banyak pengunjung.

Saran dari penelitian ini yaitu dapat mengembangkan aplikasi ini dengan menambahkan fitur-fitur lain yang dapat terintegrasi dengan aplikasi *mobile* ini supaya dapat lebih lengkap, misalnya menambahkan fitur pemesanan tiket dengan pembayaran transfer lewat bank. Dengan adanya fitur tersebut diharapkan ekosistem musik di Indonesia semakin baik lagi.

REFERENSI

- A. Ramadhanti, M. Nursaif, and A. M. I. Taufik, "Motivasi Penggunaan Spotify Sebagai Media Penyebarluasan Karya Musik Musisi Indie Lokal," *Pros. Ind. Res. Work. Natl. Semin.*, vol. 10, no. 1, pp. 904–916, 2019.
- Nawasasi Parantopo Haryo Prakoso, "Perlindungan Konsumen Terhadap Pembatalan Konser Musik Oleh Pihak Promotor," p. 2014, 2014, [Online]. Available: <http://repository.unair.ac.id/13736/>.
- D. E. Vrenita and E. Sedyono, "Perancangan Aplikasi M-Market (Mobile Market) Berbahasa Jawa untuk Penjualan Smartphone (Studi Kasus : Kota Salatiga)," no. 672013148, 2017.
- T. A. S. Prasida, "Aplikasi Pengajuan Kredit Berbasis Android (Studi Kasus Bank Rakyat Indonesia Tondano).," 2012.
- N. Maulana, T. A. S. Prasida, and A. H. Lasso, "Aplikasi Untuk Panduan Wisata Kuliner dengan SAW pada Android Mobile (Studi Kasus : Kota Magelang)."
- I. PUJI SAPUTRA, "Rancang Bangun Aplikasi Siska (Sistem Informasi Karier) Berbasis Android," *J. Manaj. Inform.*, vol. 10, no. 2, pp. 21–28, 2020.
- F. Enggar Krisnada and R. Tanone, "Aplikasi Penjualan Tiket Kelas Pelatihan Berbasis Mobile menggunakan Flutter," *J. Tek. Inform. dan Sist. Inf.*, vol. 5, no. 3, pp. 281–295, 2020, doi: 10.28932/jutisi.v5i3.1865.
- G. W. Sasmito, "Penerapan Metode Waterfall Pada Desain Sistem Informasi Geografis Industri Kabupaten Tegal," *J. Inform. Pengemb. IT*, vol. 2, no. 1, pp. 6–12, 2017.
- S. S. Vallery, Happy Novita, "UML Modeling and Black Box Testing Methods in the School Payment Information System," *J. Mantik*, vol. 3, no. January, pp. 31–38, 2019.

-
- T. A. Kurniawan, “Pemodelan Use Case (UML): Evaluasi Terhadap beberapa Kesalahan dalam Praktik,” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 1, p. 77, 2018, doi: 10.25126/jtiik.201851610.
- Ovi Marzuki, “Perancangan Sistem Informasi Pengelolaan Surat Perintah Membayar Pada Perum Peruri Berbasis Web Dengan Metode Pendekatan OBJECT ORIENTED ANALYSIS AND DESIGN (OOAD),” vol. 1, no. 1, pp. 41–57, 2019.
- B. A. Priyaungga, D. B. Aji, M. Syahroni, N. T. S. Aji, and A. Saifudin, “Pengujian Black Box pada Aplikasi Perpustakaan Menggunakan Teknik Equivalence Partitions,” *J. Teknol. Sist. Inf. dan Apl.*, vol. 3, no. 3, p. 150, 2020, doi: 10.32493/jtsi.v3i3.5343.
- G. W. Setiawan, “Pengujian Perangkat Lunak Menggunakan Metode Black Box Studi Kasus Exelsa Universitas Sanata Dharma,” p. 286, 2011, [Online]. Available: https://repository.usd.ac.id/32377/2/055314010_Full.pdf.