

Design and Development of the MosqPray Application Using Extreme Programming Method

Satrio Mukti Prayoga¹, Meriska Defriani², Muhamad Agus Sunandar³

^{1,2,3}Teknik Informatika, Sekolah Tinggi Teknologi Wastukencana, Jalan Cikopak No.53, Kabupaten Purwakarta, Jawa Barat, 41151, Indonesia
e-mail : satriomukti37@wastukencana.ac.id

Abstract: Digital technology advancement has not been fully utilized to support timely prayer for Muslims, especially when traveling or in unfamiliar locations. Access to accurate prayer times and nearby mosque locations is crucial in such situations. This study aims to design and develop an Android application called MosqPray, which provides automatic prayer schedules based on the user's location and helps locate nearby mosques. The application is developed using the Extreme Programming (XP) method, which emphasizes short iterations and active user involvement. Observations of three similar apps revealed limited implementation of native mosque search features and several issues related to interface and advertisement distractions. MosqPray is built using Kotlin, Jetpack Compose, and integrated APIs including Aladhan, Google Places, and Maps SDK for Android. Blackbox testing conducted on 12 core features confirmed that all functions worked successfully and the app performed reliably on Android version 8.0 and above. The application is deemed functional and effective in assisting Muslims to fulfill their prayer obligations on time, especially in new or unfamiliar environments.

Keywords: MosqPray; Android; Prayer Schedule; Nearby Mosque; Google Maps API

Abstrak: Kemajuan teknologi digital belum sepenuhnya dimanfaatkan untuk mendukung pelaksanaan ibadah salat secara tepat waktu, khususnya bagi umat Muslim yang sedang bepergian atau berada di lokasi yang tidak familiar. Padahal, ketersediaan informasi jadwal salat dan lokasi masjid sangat dibutuhkan dalam situasi seperti ini. Oleh karena itu, penelitian ini bertujuan untuk merancang dan membangun aplikasi Android bernama MosqPray sebagai solusi berbasis lokasi untuk menyediakan informasi jadwal salat otomatis dan pencarian masjid terdekat. Aplikasi ini dikembangkan menggunakan metode Extreme Programming (XP), yang mengutamakan iterasi pendek dan umpan balik pengguna secara aktif. Observasi terhadap tiga aplikasi serupa menunjukkan adanya keterbatasan fitur pencarian masjid secara native, serta kendala antarmuka dan iklan. Aplikasi MosqPray dibangun dengan bahasa Kotlin, Jetpack Compose, serta integrasi API seperti Aladhan, Google Places, dan Maps SDK for Android. Hasil pengujian Blackbox terhadap 12 fitur utama menunjukkan bahwa seluruh fungsi berjalan dengan baik dan stabil pada Android 8.0 ke atas. Aplikasi ini dinilai layak digunakan serta dapat mendukung umat Muslim dalam memperoleh informasi ibadah secara akurat dan efisien, terutama di tempat baru.

Kata kunci: MosqPray; Android; Jadwal Salat; Masjid Terdekat; Google Maps API

PENDAHULUAN

Kemajuan teknologi dan informasi yang pesat telah memudahkan masyarakat dalam berbagai aktivitas, termasuk dalam mendukung ibadah umat muslim seperti salat (Akbar et al., 2023). Salat merupakan kewajiban yang harus dilaksanakan tepat waktu, namun dalam kondisi tertentu, seperti bepergian atau berada di tempat yang belum dikenal, umat muslim kerap mengalami kesulitan mengetahui jadwal salat dan mencari masjid terdekat.

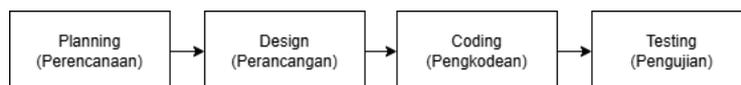
Menurut data Direktorat Jenderal Bimas Islam, terdapat 303.932 masjid di Indonesia pada tahun 2023. Jumlah ini dapat mendukung pelaksanaan salat, namun belum terintegrasi dalam sistem berbasis lokasi yang dapat memandu pengguna secara efektif. Oleh karena itu, diperlukan solusi teknologi untuk menyediakan informasi jadwal salat dan lokasi masjid secara akurat.

Peneliti memanfaatkan teknologi Location-Based Services (LBS) dan API eksternal. Menurut Pranatawijaya (2021), LBS adalah layanan yang menyediakan informasi berdasarkan lokasi pengguna. Aplikasi MosqPray dirancang untuk perangkat Android dengan memanfaatkan Google Maps API, Google Places API, dan Aladhan API. Aplikasi ini menyediakan informasi jadwal salat berdasarkan lokasi, pencarian masjid terdekat, dan fitur penyimpanan lokasi masjid. Dengan MosqPray, diharapkan pengguna dapat menjalankan salat tepat waktu di mana pun mereka berada.

KAJIAN PUSTAKA

Berikut adalah kajian Pustaka yang digunakan dalam melakukan penelitian:

Extreme Programming



Gambar 1. Metode Extreme Programming

Extreme Programming merupakan salah satu metode Agile yang banyak digunakan, dengan tahapan utama meliputi planning, design, coding, dan testing. Metode ini cocok untuk tim kecil hingga menengah serta dirancang untuk menghadapi kebutuhan sistem yang dinamis dan cepat berubah. Penelitian ini mengadopsi tahapan Extreme Programming sebagaimana ditampilkan pada Gambar 1 (Coyanda et al., 2022).

UML (Unified Modelling Language)

Unified Modeling Language (UML) merupakan bahasa berbasis visual yang digunakan untuk memodelkan, menentukan spesifikasi, membangun, serta mendokumentasikan sistem dalam pengembangan perangkat lunak berbasis objek (Object-Oriented). UML bukan sekadar bahasa pemrograman visual, namun dapat diintegrasikan secara langsung dengan berbagai bahasa pemrograman seperti Java, C++, Visual Basic, bahkan dengan basis data berorientasi objek (Sumirat et al., 2023).

Usecase Diagram

Use Case Diagram digunakan untuk menggambarkan interaksi antara aktor dan sistem. Diagram ini berfungsi untuk menunjukkan jenis-jenis interaksi yang dapat terjadi antara pengguna dan sistem. Sebagai langkah awal dalam proses pemodelan, diperlukan sebuah diagram yang mampu mengilustrasikan tindakan-tindakan yang dilakukan oleh aktor serta respons yang diberikan oleh sistem, sebagaimana dijelaskan dalam use case diagram (Sumirat et al., 2023).

Activity Diagram

Activity diagram digunakan untuk memvisualisasikan alur kerja (workflow) atau rangkaian aktivitas dalam suatu sistem, proses bisnis, atau menu dalam perangkat lunak. Penting untuk dicatat bahwa diagram ini merepresentasikan aktivitas yang dilakukan oleh sistem, bukan tindakan dari aktor (Sukamto, R. Ariani, & M. Shalahuddin, 2013 dalam Suharni et al., 2023).

Sequence Diagram

Sequence diagram bertujuan utama untuk menggambarkan urutan peristiwa yang terjadi guna menghasilkan output tertentu. Selain itu, sequence diagram memiliki fungsi yang serupa dengan activity diagram, yaitu menggambarkan alur proses dari suatu aktivitas. Namun, sequence diagram mampu menyajikan aliran data secara lebih rinci, termasuk informasi mengenai data atau perilaku yang dikirim maupun diterima dalam interaksi antar komponen sistem (Sumirat et al., 2023).

Class Diagram

Class diagram adalah diagram struktural dalam UML yang menunjukkan struktur sistem, termasuk kelas, atribut, metode, dan relasi antar objek. Bersifat statis, diagram ini hanya menampilkan jenis hubungan antar kelas tanpa menggambarkan alur kejadian. Class diagram cocok digunakan dalam proyek berbasis object-oriented karena mudah dipahami (Sumirat et al., 2023).

LBS (Location Based Service)

Location Based Service (LBS) adalah layanan yang menyampaikan informasi berdasarkan lokasi geografis pengguna secara real-time, dengan dukungan teknologi Geographic Information System (GIS). LBS terdiri dari dua komponen utama: Location Manager (API Maps) untuk menampilkan dan mengelola peta, serta Location Providers (API Location) yang mendeteksi posisi perangkat menggunakan data GPS dan lokasi aktual. Sistem ini memungkinkan pelacakan posisi, pergerakan, dan kedekatan perangkat dengan lokasi tertentu (Pranatawijaya, 2021).

API (Application Programming Interface)

Application Programming Interface (API) adalah seperangkat aturan yang memungkinkan komunikasi antar aplikasi. Saat aplikasi dijalankan, permintaan dikirim ke server melalui API dan direspons dengan data yang diperlukan. API memudahkan pengembang dalam mengintegrasikan sistem dengan layanan atau sumber data lain tanpa perlu membangun dari awal (Tinambunan et al., 2024).

Aladhan API

Aladhan API adalah layanan yang menyediakan informasi waktu salat berdasarkan tanggal tertentu, dengan data yang dapat diakses melalui HTTP/HTTPS dalam format JSON. Meski praktis bagi pengembang, waktu salat yang ditampilkan bisa berbeda dengan jadwal resmi masjid atau otoritas lokal karena perbedaan metode perhitungan (Aladhan, 2025).

Places API

Places API adalah layanan yang menyediakan data lokasi seperti tempat usaha dan titik penting melalui protokol HTTP. API ini memungkinkan pencarian lokasi berdasarkan teks, kategori, atau lokasi terdekat, serta dilengkapi fitur autocomplete untuk mempermudah input pengguna (Google, 2025b).

Maps SDK for Android

Maps SDK for Android memungkinkan integrasi peta Google ke dalam aplikasi Android, termasuk Wear OS. SDK ini mendukung berbagai tampilan dan interaksi peta, serta memungkinkan penambahan elemen seperti marker dan poligon. Kompatibel dengan Kotlin dan Java, SDK ini juga menyediakan pustaka tambahan untuk fitur lanjutan (Google, 2025a).

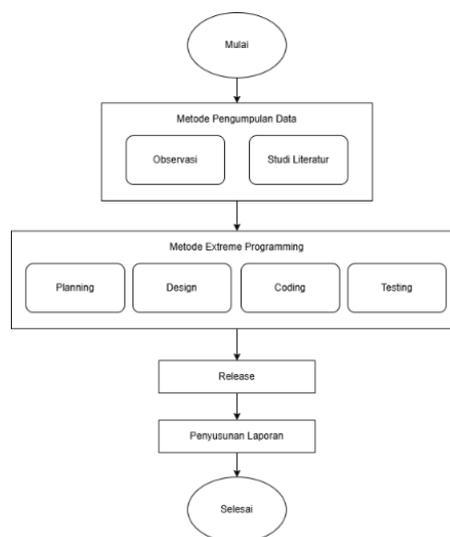
Kotlin

Kotlin adalah bahasa pemrograman open-source dengan penyetikan statis yang mendukung paradigma fungsional dan berorientasi objek. Mengadopsi konsep dari bahasa seperti Java dan C#, Kotlin dirancang sebagai evolusi dari berbagai bahasa sebelumnya. Kotlin tersedia dalam beberapa bentuk: Kotlin/JVM, Kotlin/JS, dan Kotlin/Native. Dikembangkan oleh Kotlin Foundation—kolaborasi antara JetBrains dan Google—bahasa ini kini direkomendasikan secara resmi oleh Google untuk pengembangan aplikasi Android (Developer, 2025b).

Jetpack Compose

Jetpack Compose adalah toolkit UI modern berbasis deklaratif yang mempermudah pengembangan antarmuka dengan Kotlin. Menggunakan pendekatan reaktif, Compose secara otomatis memperbarui UI saat data berubah, cukup dengan mendefinisikan tampilan melalui fungsi (Developer, 2025a).

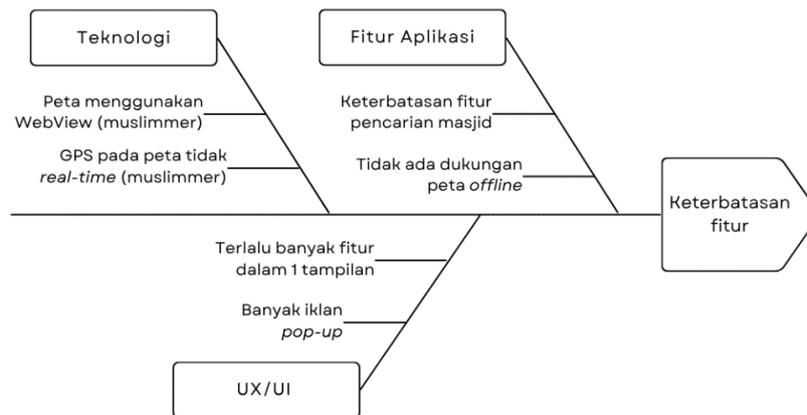
METODE PENELITIAN



Gambar 2 Kerangka Penelitian

Observasi dilakukan dengan cara mencari aplikasi di Play Store yang sejenis dengan aplikasi MosqPray, aplikasi yang diobservasi yaitu Muslimmer oleh Vittenssi, Prayer Times - Qibla & Salah oleh ZeeDev Islam, dan Jadwal Sholat, Kiblat & Adzan oleh PT Dalfindo Cipta Karya. Ketiga aplikasi tersebut dipilih karena tidak terlalu kompleks dan memiliki fitur yang mirip dengan Aplikasi yang akan dibuat yaitu MosqPray.

Hasil dari observasi kepada tiga aplikasi ditemukan beberapa kelemahan, dapat digambarkan pada gambar 3 berikut :



Gambar 3 Hasil analisis observasi aplikasi

Berdasarkan hasil observasi terhadap tiga aplikasi, ditemukan sejumlah kelemahan yang dapat menjadi peluang inovasi dalam pengembangan aplikasi MosqPray. Gambar 3 menunjukkan hubungan antara penyebab utama dan dampaknya terhadap keterbatasan fitur dalam aplikasi. Beberapa aspek yang menjadi perhatian meliputi teknologi, fitur aplikasi, dan desain antarmuka pengguna (UI/UX). Pertama, fitur pencarian masjid terdekat belum sepenuhnya diimplementasikan secara native. Hanya aplikasi Muslimmer yang menyediakan fitur ini, namun masih menggunakan WebView, sementara dua aplikasi lainnya belum menyediakannya. Kedua, keberadaan iklan yang mengganggu juga menjadi masalah. Muslimmer menampilkan terlalu banyak iklan pop-up yang mengurangi kenyamanan pengguna, meskipun dua aplikasi lainnya juga memiliki iklan, namun dalam jumlah yang lebih terbatas. Ketiga, dari sisi UI/UX, Muslimmer dirasa kurang optimal karena banyaknya iklan dan tombol pada halaman utama yang membingungkan, sedangkan dua aplikasi lainnya memiliki tampilan yang lebih sederhana dan ringan.

Untuk mengatasi hal tersebut, pengembangan MosqPray menggunakan metode Extreme Programming (XP), sesuai dengan teori Kent Beck (Beck, 2004), yang menekankan fleksibilitas, kolaborasi intensif, dan pengembangan berbasis iterasi pendek. XP dinilai cocok dalam penelitian ini karena pengembangan dilakukan dalam lingkup kecil, dengan kebutuhan yang dapat berubah cepat selama iterasi.

Extreme Programming (XP) adalah salah satu metode Agile yang dirancang untuk tim kecil hingga menengah, untuk menghadapi kebutuhan yang tidak pasti dan perubahan yang cepat. Metode

ini terdiri dari empat tahap utama: planning, design, coding, dan testing. Dalam penelitian ini, XP digunakan sebagai metode pengembangan perangkat lunak, seperti pada Gambar 1 (Coyanda, Ariati, Sunardi, & K. Ghazali, 2022). Berikut penjelasan tiap tahap dalam metode XP :

1. Planning

Dilakukan analisis kebutuhan melalui observasi fitur aplikasi sejenis dan literatur. Fitur penting seperti deteksi lokasi otomatis, jadwal salat berbasis lokasi, serta pencarian masjid digunakan sebagai dasar pengembangan.

2. Design

Perancangan sistem dilakukan menggunakan pendekatan UML, seperti Use Case, Activity, Sequence, dan Class Diagram. Untuk perancangan UI digunakan Balsamiq sebagai alat bantu perancangan antarmuka berbasis prinsip user-centered design.

3. Coding

Implementasi kode dilakukan menggunakan Kotlin dan Jetpack Compose, yang sesuai dengan standar modern Android . API eksternal seperti Aladhan API dan Places API diintegrasikan menggunakan Retrofit.

4. Testing

Pengujian dilakukan dengan metode Blackbox Testing, yaitu menguji fungsi dari sisi pengguna tanpa melihat struktur kode. Evaluasi dilakukan berdasarkan kecocokan input-output sesuai dengan skenario. Sebagai bagian dari validasi kelayakan, dilakukan evaluasi terhadap seluruh fitur aplikasi.

HASIL DAN PEMBAHASAN

Berikut ini adalah Hasil dan Pembahasan dari penelitian:

Planning

Perencanaan didasarkan pada observasi terhadap tiga aplikasi serupa. Ditemukan bahwa fitur-fitur seperti pencarian masjid, jadwal salat otomatis, dan pengingat salat sangat penting. Teori LBS (Location-Based Services) yang dijelaskan oleh Pranatawijaya (2021) menjadi dasar pemanfaatan lokasi pengguna secara real-time.

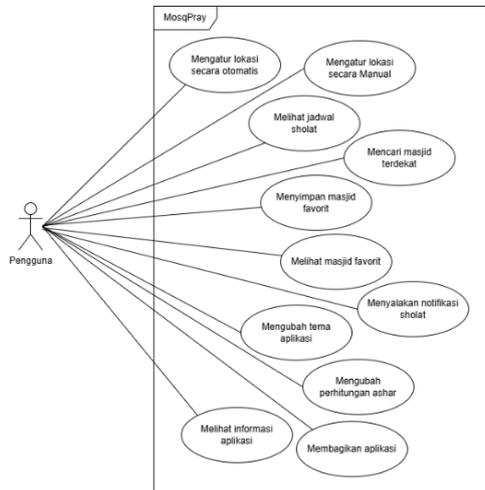
Design

Pada tahap Design, sistem aplikasi MosqPray dirancang menggunakan Unified Modeling Language (UML), yang merupakan bahasa visual standar untuk memodelkan dan mendokumentasikan sistem berbasis objek (Sumirat et al., 2023). UML membantu pengembang menyusun representasi sistem secara terstruktur dan lintas bahasa pemrograman. Diagram yang digunakan meliputi:

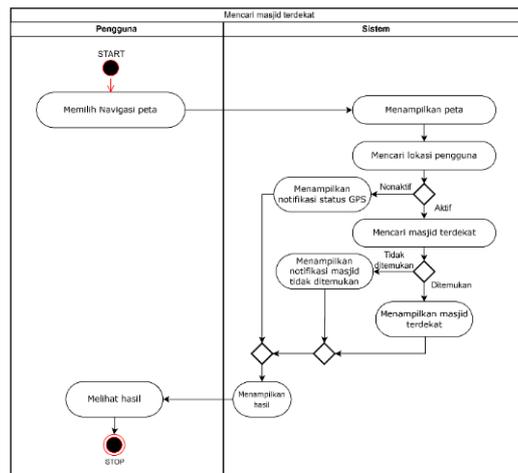
1. Use Case Diagram: menggambarkan interaksi antara pengguna dan sistem (Sumirat et al., 2023).
2. Activity Diagram: memvisualisasikan alur kerja sistem (Sukamto et al., 2013 dalam Suharni et al., 2023).

3. Sequence Diagram: menunjukkan urutan interaksi antar objek secara rinci (Sumirat et al., 2023).
4. Class Diagram: menggambarkan struktur kelas dan relasinya (Sumirat et al., 2023).

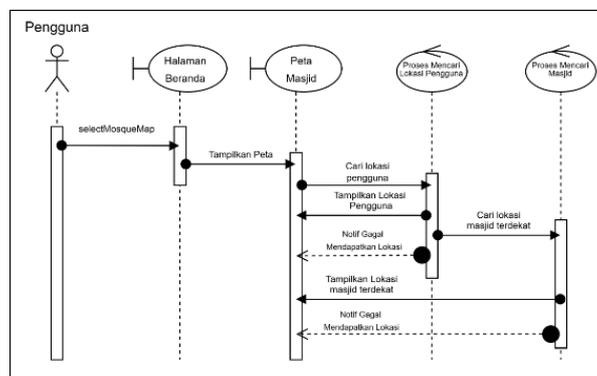
Diagram tersebut ditampilkan pada Gambar 4, Gambar 5, Gambar 6, dan Gambar 7. Perancangan antarmuka pengguna (UI/UX) dibuat dengan Balsamiq, mengacu pada prinsip user-centered design agar navigasi aplikasi lebih sederhana dan mudah dipahami (Gambar 8).



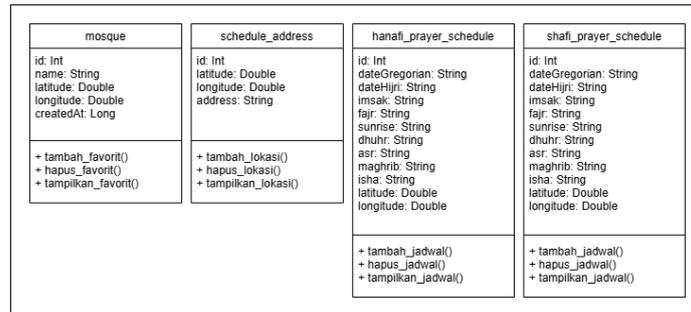
Gambar 4 Usecase Diagram aplikasi MosqPray



Gambar 5 Activity Diagram mencari masjid terdekat



Gambar 6 Sequence Diagram Mencari masjid terdekat



Gambar 7 Class Diagram aplikasi MosqPray



Gambar 8 Rancangan tampilan peta masjid terdekat

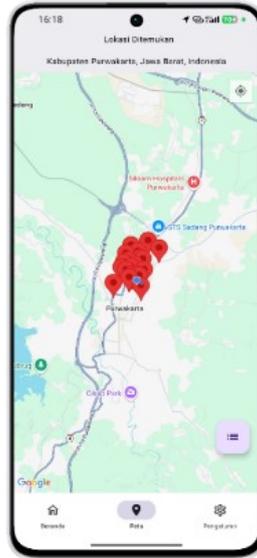
Coding

API Aladhan digunakan untuk mengambil data jadwal salat berdasarkan lokasi dan metode perhitungan. Integrasi dengan Places API digunakan untuk mendeteksi masjid terdekat, dan Maps SDK for Android untuk menampilkan peta. Pemrograman dilakukan dengan Jetpack Compose yang mendukung deklaratif dan reaktif UI. Endpoint yang digunakan untuk mendapatkan jadwal salat adalah <https://api.aladhan.com/v1/calendarByAddress/{year}>, data waktu salat ditentukan berdasarkan parameter lokasi (alamat), tahun, metode perhitungan, serta mazhab. Pada gambar 9 merupakan kode implementasi endpoint menggunakan retrofit.

```
interface ApiService {
    @GET("{year}")
    suspend fun getPrayerTimes(
        @Path("year") year: Int,
        @Query("address") address: String,
        @Query("method") method: Int,
        @Query("tune") tune: String,
        @Query("calendarMethod") calendar: String,
        @Query("school") school: Int
    ): AnnualScheduleResponse
}
```

Gambar 9 Implementasi Endpoint API Aladhan

Pada gambar 10 menunjukkan implementasi tampilan peta masjid terdekat, menampilkan nama lokasi, *marker* masjid, dan tombol daftar masjid:



Gambar 10 Implementasi tampilan peta masjid terdekat

Testing

Berikut tabel hasil pengujian aplikasi menggunakan metode Blackbox Testing:

No	Fungsi	Skenario Pengujian	Output	Kesimpulan
1	Melihat jadwal salat	Membuka aplikasi	Menampilkan jadwal salat	Berhasil
2	Mengatur lokasi secara otomatis	Memilih tombol “Deteksi lokasi secara otomatis”	Mendapatkan dan menyimpan lokasi pengguna dan jadwal salat	Berhasil
3	Mengatur lokasi secara Manual	Memilih tombol “Pilih lokasi Manual”, Memasukkan nama lokasi dan memilih lokasi	Menyimpan lokasi pengguna dan mendapatkan jadwal salat	Berhasil
4	Mencari masjid terdekat	Memilih tombol navigasi “Peta”	Mendapatkan lokasi pengguna dan masjid terdekat	Berhasil
5	Menyimpan masjid favorit	Memilih tombol “Favorit”	Menyimpan data masjid ke database	Berhasil
6	Melihat masjid favorit	Memilih tombol “Daftar Masjid”	Menampilkan daftar masjid favorit	Berhasil
7	Menyalakan notifikasi sholat	menekan tombol toggle notifikasi	Menjadwalkan notifikasi salat	Berhasil
8	Mengubah tema aplikasi	Memilih tombol “tema”, dan memilih tema	Mengganti tema aplikasi	Berhasil
9	Mengubah perhitungan ashar	Memilih tombol “Perhitungan Ashar”, dan memilih metode	Menampilkan jadwal salat ashar sesuai metode yang dipilih	Berhasil
10	Membagikan aplikasi	Memilih tombol “Bagikan”	Menampilkan opsi bagikan	Berhasil

11	Melihat informasi aplikasi	Memilih tombol “Tentang”	Menampilkan informasi aplikasi	Berhasil
12	Mendapatkan notifikasi salat	Menunggu waktu salat tiba setelah notifikasi diaktifkan	Notifikasi muncul sesuai jadwal salat yang ada	Berhasil

Dari total 12 fitur yang diuji, seluruh fitur berhasil dijalankan sesuai dengan fungsinya. Hasil ini menunjukkan bahwa aplikasi MosqPray telah memenuhi aspek fungsional dengan baik dan dapat digunakan sebagaimana mestinya oleh pengguna tanpa kendala pada fitur utama. Keberhasilan semua skenario pengujian menandakan bahwa sistem bekerja stabil dan sesuai dengan tujuan pengembangan.

Release

Aplikasi yang telah dibangun dan telah berhasil melalui tahap pengujian, aplikasi akan dirilis atau di publikasikan di Google Play Store, sehingga pengguna dapat mengunduh aplikasi dengan aman. Aplikasi MosqPray dirilis di Google Play Store karena platform ini merupakan tempat distribusi aplikasi Android yang paling umum dan aman bagi banyak pengguna.

KESIMPULAN DAN SARAN

Pada penelitian ini telah berhasil dirancang dan dibangun aplikasi MosqPray berbasis Android menggunakan metode Extreme Programming, dengan bahasa pemrograman Kotlin, Jetpack Compose, Google Maps SDK for Android, API Places, dan API Aladhan, serta perancangan sistem menggunakan UML. Aplikasi ini menyediakan fitur utama berupa jadwal salat otomatis dan pencarian masjid terdekat berdasarkan lokasi pengguna. Hasil pengujian Blackbox menunjukkan bahwa seluruh fungsi berjalan sesuai harapan. Aplikasi ini diharapkan dapat membantu umat Muslim dalam memperoleh informasi waktu salat dan lokasi masjid terdekat, terutama saat berada di lokasi baru. Untuk pengembangan lebih lanjut, disarankan penambahan fitur seperti pengingat salat dengan suara adzan yang dapat dikustomisasi, dukungan multiplatform, fitur login untuk menyimpan data favorit ke cloud, serta fitur tambahan seperti arah kiblat dan kalender hijriah.

DAFTAR PUSTAKA

- Aditya, S. Kumar., & Karn, V. Kumar. (2014). *Android SQLite essentials: develop android applications with one of the most widely used database engines, SQLite*. Birmingham: Packt Publishing Ltd.
- Akbar, M. L., Usman, A., & Budiman, A. (2023). Rancang Bangun Desain UI/UX Pada Pembuatan Startup Aplikasi Selfcare Berbasis Website. *Jurnal Ilmu Komputer Dan Sistem Informasi (JIRSI)*, 2(1), 42–56.
- Aladhan. (2025, March 18). Prayer Times API - Prayer Times and Adhan Player. Retrieved March 18, 2025, from <https://aladhan.com/prayer-times-api>
- Balsamiq Studios. (2025, June 12). Balsamiq: About Us. Retrieved June 12, 2025, from <https://balsamiq.com/company/>

- Coyanda, J. R., Ariati, N., Sunardi, H., & K.Ghazali. (2022). Sistem Informasi Ekspedisi Barang Pada PT. New Power Global Energy Dengan Menggunakan Metode Extreme Programming. *Jurnal Ilmiah Informatika Global*, 13(2), 94–99. <https://doi.org/10.36982/jiig.v13i2.2295>
- Developer. (2025a, March 17). Dasar-Dasar Jetpack Compose. Retrieved March 17, 2025, from <https://developer.android.com/codelabs/jetpack-compose-basics?hl=id#0>
- Developer. (2025b, March 17). Ringkasan Kotlin. Retrieved March 17, 2025, from <https://developer.android.com/kotlin/overview?hl=id>
- Dicoding. (2025, March 19). Apa itu Activity Diagram? Beserta Pengertian, Tujuan, Komponen - Dicoding Blog. Retrieved March 19, 2025, from <https://www.dicoding.com/blog/apa-itu-activity-diagram/>
- Dzulqarnain, F., & Tukino. (2023). Rancang Bangun Aplikasi Belajar Arab Untuk Android Menggunakan Jetpack Compose Dan Kotlin. *CBIS JOURNAL*, 11(01). Retrieved from <http://ejournal.upbatam.ac.id/index.php/cbishttp://ejournal.upbatam.ac.id/index.php/cbis>
- Farhan, NM Faizah, & Koryanto, L. (2024). Rancang Bangun Aplikasi Sistem Informasi Geografis untuk Pencarian Bengkel Resmi Honda di Kota Depok Berbasis Android dengan Metode Location-Based Service (LBS). *Design Journal*, 2(2), 79–88. <https://doi.org/10.58477/dj.v2i2.194>
- Google. (2025a, March 18). Maps SDK for Android. Retrieved March 18, 2025, from <https://developers.google.com/maps/documentation/android-sdk/overview>
- Google. (2025b, March 18). Places API. Retrieved March 18, 2025, from <https://developers.google.com/maps/documentation/places/web-service/overview>
- Gustiawan, A., & Trisianto, C. (2022). PERANCANGAN SISTEM INFORMASI PENGGAJIAN KARYAWAN BERBASIS WEB MENGGUNAKAN METODE EXTREME PROGRAMMING PADA PT. PRADANA ENERGI GEMILANG. *Jurnal Ilmu Komputer JIK*, 5, 1–6.
- Imamsyah, R., Sari, N. N. K., & Lestari, A. (2023). Rancang Bangun Aplikasi Angkotkita Menggunakan Location Based Service Dengan Metode Haversine Berbasis Android. *JOINTECOMS (Journal of Information Technology and Computer Science)*, 3(1), 28–38.
- Kaplan, D. E., & J. Hegarty, C. (2006). *Understanding GPS Principles and Applications Second Edition*. Norwood: ARTECH HOUSE, INC.
- Muslimmer Apps. (2025, May 27). Waktu Sholat, Adzan, Kiblat. Retrieved May 27, 2025, from Google Play Store website: <https://play.google.com/store/apps/details?id=com.muslimmer.prayer.times.qibla.compass.azan.holy.quran&hl=id>
- P. Sitorus, J. H., & Sakban, M. (2021). Perancangan Sistem Informasi Penjualan Berbasis Web Pada Toko Mandiri 88 Pematangsiantar. *Jurnal Bisantara Informatika (JBI)*, 5(2).
- Pasaribu, J. S. (2021). PEMBUATAN APLIKASI PEMESANAN BANNER DI WARNA PRINT KOTA CIMAHI. *Jurnal Ilmiah Teknologi Informasi Terapan*, 7(2), 138–147.
- Pranatawijaya, V. H. (2021). *PENERAPAN LOCATION BASED SERVICED (LBS) DALAM PROTOTIPE PENGENALAN RUANGAN DENGAN METODE EXTREME PROGRAMMING*. 15(1). <https://doi.org/10.47111/JTI>
- PT Dalfindo Cipta Karya. (2025, June 2). Jadwal Sholat, Kiblat & Adzan. Retrieved June 2, 2025, from Google Play Store website: <https://play.google.com/store/apps/details?id=com.andi.waktusholatdankiblat&hl=id>
- Sayfulloh, A. (2021). Perancangan Program Penjualan Mainan Berbasis Web Menggunakan Metode Extreme Programming. *Riset Dan E-Jurnal Manajemen Informatika Komputer*, 5(2), 306–312. <https://doi.org/10.33395/remik.v4i1.11150>
- Silberschatz, A., Galvin, P. B., & Gagne, G. (2018). *Operating System Concepts 10th Edition* (10th ed., Vol. 1278; D. Fowley & R. Dannelly, Eds.). John Wiley & Sons, Inc.
- Siregar, D. R. S., Koryanto, L., & Faizah, N. (2023). Aplikasi Pencarian Hotel di Kota Jakarta Berbasis Android dengan Metode Location Based Service (LBS) Menggunakan Android Studio. *Computer Journal*, 1(1), 64–72. <https://doi.org/10.58477/cj.v1i1.65>

- Suharni, Susilowati, E., & Pakusadewa, F. (2023). PERANCANGAN WEBSITE RUMAH MAKAN NINIK SEBAGAI MEDIA PROMOSI MENGGUNAKAN UNIFIED MODELLING LANGUAGE. *Jurnal Rekayasa Informasi*, 12(1).
- Sumirat, L. P., Cahyono, D., Kristyawan, Y., & Kacung, S. (2023). *DASAR-DASAR Rekayasa Perangkat Lunak* (1st ed.). Madza Media. Retrieved from www.madzamedia.co.id
- Suzana, A. (2021). ANALISIS DAN PERANCANGAN APLIKASI PEMESANAN TIKET BUS BERBASIS ANDROID. *Jurnal Informatika Dan Rekayasa Perangkat Lunak (JATIKA)*, 2(3), 353–360.
- Tinambunan, M. H., Siregar, A. H., & Ginting, S. (2024). *FULLSTACK PROGRAMMING : Membangun Application Programming Interface (API) DENGAN LARAVEL* (1st ed.). Tahta Media Group.
- Uminingsih, Ichsanudin, M. N., Yusuf, M., & Suraya. (2022). PENGUJIAN FUNGSIONAL PERANGKAT LUNAK SISTEM INFORMASI PERPUSTAKAAN DENGAN METODE BLACK BOX TESTING BAGI PEMULA INFO ARTIKEL ABSTRAK. *Jurnal Ilmiah Teknik Dan Ilmu Komputer*, 1(2), 1–8. <https://doi.org/10.55123>
- ZeeDev Apps. (2025, May 27). Prayer Times - Qibla & Salah. Retrieved May 27, 2025, from [Google Play Store website:](https://play.google.com/store/apps/details?id=com.zeedev.islamprayertime&hl=id)