# **BAB III**

# **ANALISIS DAN PERANCANGAN**

## **Analisis**

### **Identifikasi Masalah**

Pada kegiatan pendakian, informasi menjadi hal yang penting untuk didapatkan guna kelancaran dan keselamatan dalam pendakian. Untuk mendapatkan informasi pendakian Gunung Panderman-Buthak, para pendaki harus mengunjungi secara langsung pos pendakian yang berada di Kota Batu. Karena hingga saat ini masih belum terdapat suatu sistem yang menyediakan informasi untuk pendaki, seperti pembukaan atau penutupan jalur. Proses pendaftaran yang secara langsung juga menyebabkan informasi tentang persyaratan pendakian hanya bisa diketahui dari lokasi pos pendakian. Buruknya penyimpanan kartu identitas pendaki di pos pendakian juga sering terjadi dimana terdapat keluhan dari pendaki atas kartu identitasnya yang kotor dan rusak.

Kasus hilangnya pendaki di jalur pendakian juga disebabkan oleh sedikitnya pengetahuan yang dimiliki oleh pendaki, terutama oleh pendaki pemula yang banyak mendaki Gunung Panderman-Buthak. Tidak adanya kemampuan navigasi didalam hutan menyebabkan pendaki salah memilih jalur pendakian. Selain itu, banyak pendaki yang melakukan perbuatan melanggar peraturan pendakian seperti tindakan vandalisme, tindakan asusila, pemetikan bunga edelweiss, dan pendakian di jalur illegal.

### **Pemecahan Masalah**

Berdasarkan pada identifikasi masalah yang telah dilakukan diatas, maka akan dibuat sebuah Aplikasi Manajemen Pendakian Gunung Panderman-Buthak Berbasis Android Menggunakan Metode *Scrum*. Adanya sistem ini, diharapkan akan menyelesaikan masalah dengan beberapa kelebihan diantaranya yaitu:

1. Mempermudah proses manajemen data pendaki.
2. Mempermudah pihak pengelola menerapkan sanksi *blacklist* kepada pendaki.
3. Mempermudah menyampaikan informasi kepada pendaki.
4. Mempermudah pengecekan data pendakian.
5. Membantu pendaki navigasi dengan fitur *tracking* pendakian.
6. Membantu pendaki mengirimkan sinyal darurat dan posisi terakhir di jalur pendakian saat terdapat kendala dalam pendakian.

## **Perancangan**

Berdasarkan analisis permasalahan yang telah dilakukan, maka dibuat perancangan dengan menyesuaikan kebutuhan masalah. Dalam merancang pembuatan aplikasi pendakian Gunung Panderman-Buthak menggunakan metode *scrum*, dimana terdapat tiga langkah dalam perancangan sistem yang meliputi *product backlog, sprint backlog,* dan *sprint planning*.

1. Product Backlog

Dalam metode pengembangan scrum, langkah pertama yang dilakukan adalah menyusun *product backlog*. *Product backlog* disusun berdasarkan kebutuhan yang didapatkan melalui observasi dan pengumpulan data yang telah dilakukan dan dapat berubah di sepanjang waktu pembuatan aplikasi. Berikut tabel *product backlog* yang terdapat pada aplikasi pendakian Gunung Panderman-Buthak.

**Tabel 3.1** Tabel Product Backlog User Aplikasi

|  |  |
| --- | --- |
| **No** | **Backlog Item User** |
| 1 | Menampilkan halaman awal aplikasi |
| 2 | Menampilkan halaman *log in* |
| 3 | Menampilkan halaman registrasi user |
| 4 | Menampilkan halaman utama aplikasi yang berisi informasi jalur pendakian |
| 5 | Menampilkan halaman pendaftaran pendakian oleh user |
| 6 | Menampilkan halaman pembayaran pendakian oleh user |
| 7 | Menampilkan halaman informasi pendakian (*Guide*) |
| 8 | Menampilkan halaman peralatan pendakian |
| 9 | Menampilkan halaman keselamatan pendakian |
| 10 | Menampilkan halaman kompas |
| 11 | Menampilkan halaman tracking pendakian yang sesuai dengan gunung yang didaki |
| 12 | Menampilkan halaman pengiriman sinyal SOS untuk meminta bantuan |
| 13 | Menampilkan halaman history pendakian user |
| 14 | Menampilkan halaman *qrcode* pendaki yang sesuai dengan pemesanan pendakian |
| 15 | Menampilkan halaman profil akun pendaki |

**Tabel 3.2** Tabel Product Backlog Admin Aplikasi

|  |  |
| --- | --- |
| **No** | **Backlog Item Admin** |
| 1 | Menampilkan halaman awal aplikasi |
| 2 | Menampilkan halaman *log in* |
| 3 | Menampilkan halaman utama yang berisi data pendakian |
| 4 | Menampilkan halaman pengecekan registrasi pendakian |
| 5 | Menampilkan halaman pengecekan pembayaran pendakian |
| 6 | Menampilkan halaman scan *qrcode* pendaki |
| 7 | Menampilkan halaman detail data pendakian tiap kelompok |
| 8 | Menampilkan halaman *update* jalur dan kuota pendakian |
| 9 | Menampilkan halaman *blacklist* pendaki yang melakukan pelanggaran |
| 10 | Menampilkan halaman sinyal SOS untuk pendaki yang butuh bantuan |
| 11 | Menampilkan halaman laporan pendakian |
| 12 | Menampilkan halaman profil akun Admin |
| 13 | Menampilkan halaman tambah Admin |

1. Sprint Backlog

Dalam perancangan aplikasi pendakian Gunung Panderman-Buthak dengan metode scrum, langkah kedua yang dilakukan adalah menyusun sprint backlog. Sprint backlog merupakan daftar product backlog item yang terpilih dan terurut berdasarkan prioritasnya untuk dikerjakan dalam sprint dengan perencanaan dan perkiraan mengenai fungsionalitasnya untuk menghasilkan produk yang diinginkan. Berikut tabel sprint backlog pada pembuatan aplikasi pendakian Gunung Panderman-Buthak.

**Tabel 3.1** Tabel Sprint 1 Backlog User

|  |  |  |  |
| --- | --- | --- | --- |
| **Story ID** | **Story** | **Estimasi** | **Prioritas** |
| 3 | User dapat membuat akun baru | 2 | 1 |
| 2 | User dapat melakukan *log in* aplikasi | 1 | 2 |
| 4 | User dapat melihat halaman utama yang berisi informasi jalur pendakian | 3 | 3 |
| 5 | User dapat melakukan pendaftaran pendakian | 5 | 4 |
| 7 | User dapat melakukan perubahan *leader* dan anggota pendakian | 3 | 5 |
| 8 | User dapat melakukan pembatalan pendakian | 2 | 6 |
| 6 | User dapat melakukan *upload* bukti pembayaran pendakian | 3 | 7 |
| 9 | User dapat melakukan pembatalan *upload* bukti pembayaran | 2 | 8 |
| 17 | User dapat melakukan pengecekan history pendakian | 3 | 9 |
| 10 | User dapat melihat *qrcode* dari konfirmasi pendaftaran pendakian | 2 | 10 |
| 15 | User dapat melakukan tracking pendakian sesuai dengan jalur yang di daki | 10 | 11 |
| 16 | User dapat mengirimkan sinyal SOS ke admin untuk meminta bantuan di pendakian | 3 | 12 |
| 11 | User dapat melihat informasi pendakian | 1 | 13 |
| 12 | User dapat melihat informasi peralatan yang dibutuhkan dalam pendakian | 1 | 14 |
| 13 | User dapat melihat informasi untuk keselamatan selama pendakian | 1 | 15 |
| 14 | User dapat menggunakan kompas | 2 | 16 |
| 18 | User dapat melihat profil akun | 3 | 17 |
| 19 | User dapat melakukan perubahan profil akun | 2 | 18 |
| 20 | User dapat melihat informasi tentang aplikasi | 1 | 19 |
| 1 | User dapat melihat halaman awal aplikasi (splash screen) | 1 | 20 |

**Tabel 3.2** Tabel Sprint 2 Backlog Admin

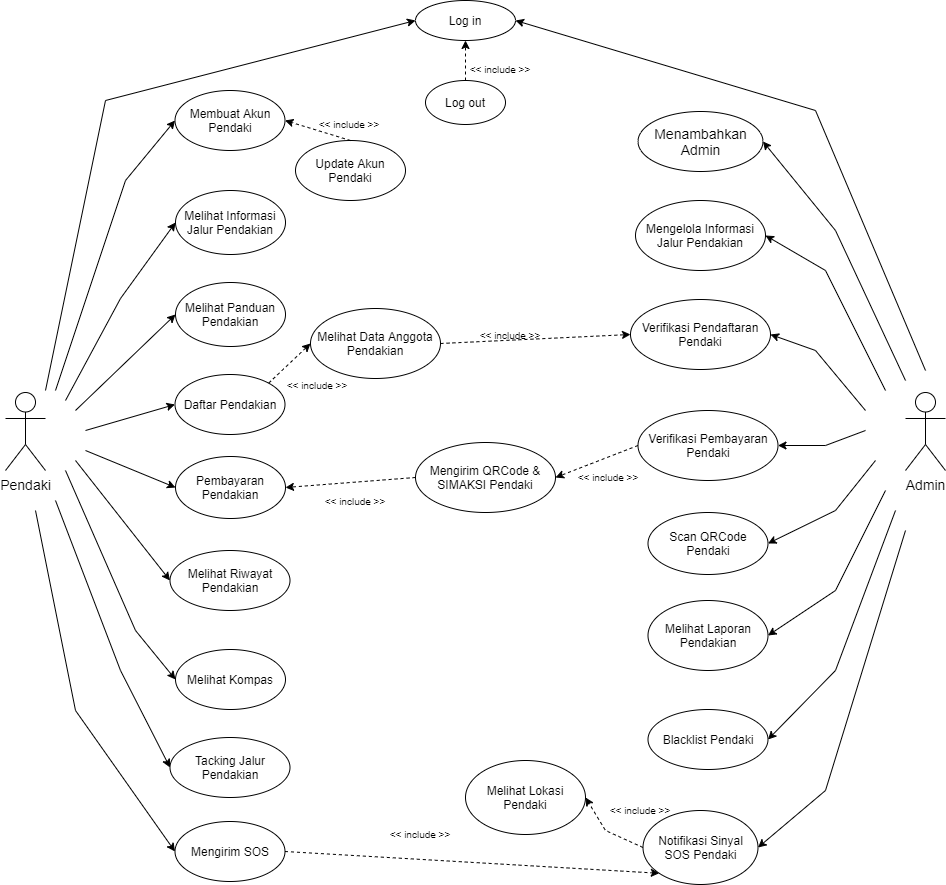
|  |  |  |  |
| --- | --- | --- | --- |
| **Story ID** | **Story** | **Estimasi Hari** | **Prioritas** |
| 2 | Admin dapat *log in* ke aplikasi | 1 | 1 |
| 3 | Admin dapat melakukan pengecekan pendaftaran pendakian | 4 | 2 |
| 5 | Admin dapat melakukan pembatalan pendaftaran pendakian | 3 | 3 |
| 4 | Admin dapat melakukan pengecekan pembayaran pendakian | 4 | 4 |
| 6 | Admin dapat melakukan pembatalan pembayaran pendakian | 3 | 5 |
| 7 | Admin dapat melihat halaman informasi jalur pendakian | 3 | 6 |
| 8 | Admin dapat melakukan *update* jalur dan kuota pendakian | 3 | 7 |
| 9 | Admin dapat melakukan pengecekan (*scan*) *qrcode* pendaki pada *check in* dan *check out* pendakian | 4 | 8 |
| 11 | Admin dapat melakukan *blacklist* pendaki yang melakukan pelanggaran | 3 | 9 |
| 10 | Admin dapat melihat pendaki yang masuk daftar *blacklist* | 1 | 10 |
| 12 | Admin dapat menerima sinyal SOS dari pendaki yang membutuhkan bantuan | 4 | 11 |
| 13 | Admin dapat melihat laporan pendakian per bulan | 4 | 12 |
| 14 | Admin dapat melihat profil akun admin | 2 | 13 |
| 15 | Admin dapat merubah profil akun admin | 2 | 14 |
| 16 | Admin dapat menambah admin baru | 2 | 15 |
| 17 | Admin dapat melihat halaman informasi aplikasi | 1 | 16 |
| 1 | Admin dapat melihat halaman awal aplikasi (*splash screen*) | 1 | 17 |

### **Perancangan Sistem**

Pada penelitian ini, peneliti menggunakan model UML (*Unified Modeling Language*) untuk perancangan sistemnya. UML ialah alat untuk membuat visualisasi dan dokumentasi dari hasil analisis dan desain yang berisi sintak dalam memodelkan sistem yang dibangun. Model UML yang digunakan dalam menggambarkan sistem yang akan dibangun yaitu *use case diagram* dan *activity diagram*.

#### **Use Case Diagram**

*Use case diagram* adalah sebuah interaksi antara aktor dengan sistem. *Use case diagram* yang dibuat pada sistem yang akan dibangun sebagai berikut:



**Gambar 3.1** Use Case Diagram

Pada sistem ini terdapat dua aktor yang meliputi Pendaki dan Admin yang mempunyai fungsi berbeda. Deskripsi dari fungsi aktor keduanya sebagai berikut:

1. Pendaki

* Pendaki dapat melakukan daftar akundi aplikasi.
* Pendaki dapat melakukan *update* akun di aplikasi
* Pendaki dapat melakukan *login* di aplikasi.
* Pendaki dapat melakukan pendaftaran pendakian di aplikasi.
* Pendaki dapat mengelola data anggota dalam pendakian.
* Pendaki dapat melakukan upload bukti pembayaran dari aplikasi.
* Pendaki dapat menerima verifikasi pembayaran, qrcode, dan SIMAKSI untuk pendakian.
* Pendaki dapat melihat riwayat selama mendaki ke Gunung Panderman-Buthak.
* Pendaki dapat melihat informasi buka/tutup jalur pendakian.
* Pendaki dapat melihat panduan untuk pendakian, seperti peralatan, keselamatan, dan peta jalur pendakian.
* Pendaki dapat melihat informasi cuaca di pos pendakian.
* Pendaki dapat menggunakan kompas untuk membaca arah mata angin.
* Pendaki dapat navigasi rute jalur dengan *tracking* pendakian.
* Pendaki dapat mengirimkan sinyal darurat ke pos pendakian saat membutuhkan pertolongan di jalur pendakian.
* Pendaki dapat melakukan *logout* akun di aplikasi.

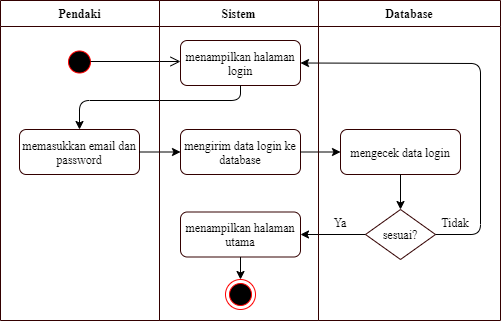
1. Admin

* Admin dapat melakukan *login* di aplikasi.
* Admin dapat melakukan verifikasi pendaftaran pendakian dengan melakukan pengecekan identitas dan persyaratan pendakian.
* Admin dapat melakukan verifikasi pembayaran pendakian dan mengirimkan qrcode ke pendaki.
* Admin dapat melakukan *scanning* qrcode saat pendaki akan melakukan pendakian.
* Admin dapat melihat laporan pendakian bulanan yang meliputi jumlah pendaki, jumlah pendapatan pendakian, jenis kelamin pendaki, dan usia pendaki sebagai bahan evaluasi untuk pihak pengelola.
* Admin dapat melakukan *blacklist* kepada pendaki.
* Admin dapat memberikan informasi buka/tutup jalur pendakian.
* Admin dapat mengetahui lokasi terakhir pendaki di jalur pendakian.
* Admin dapat menerima sinyal darurat dari pendaki yang membutuhkan pertolongan.
* Admin dapat melakukan *logout* di aplikasi.

#### **Activity Diagram**

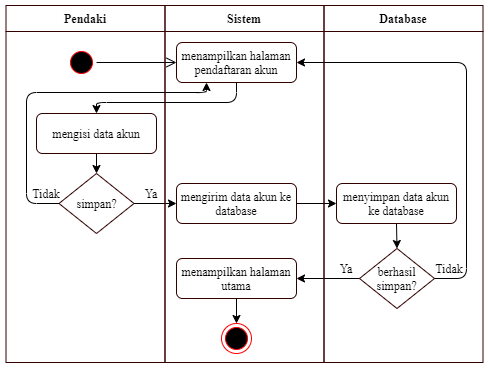
*Activity diagram* digunakan untuk setiap aktivitas atau aliran kerja yang terdapat pada sistem yang akan dibangun. *Activity diagram* mencakup alur kerja mulai dari *user* login aplikasi, manajemen akun, pendaftaran dan pembayaran pendakian, riwayat pendakian, *tracking* jalur pendakian, dan sinyal darurat dari pendaki sebagai berikut:

1. *Activity diagram Login* Pendaki



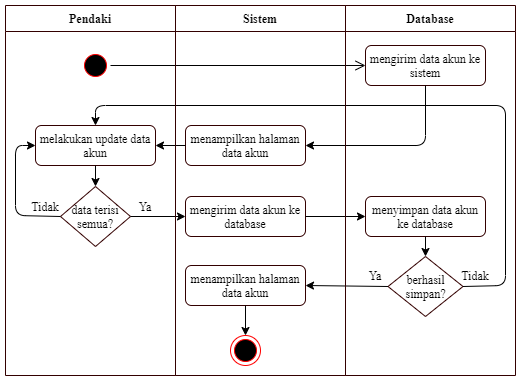
**Gambar 3.2** Activity Diagram Login Pendaki

1. *Activity Diagram* Daftar Akun



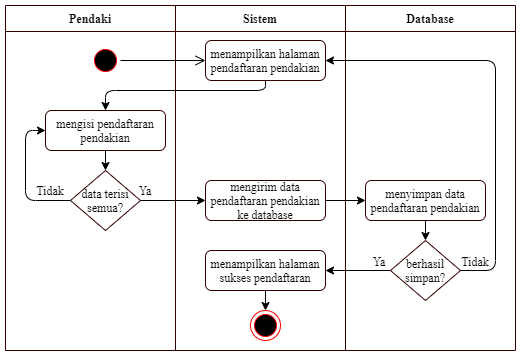
**Gambar 3.3** Activity Diagram Daftar Akun

1. *Activity Diagram Update* Akun



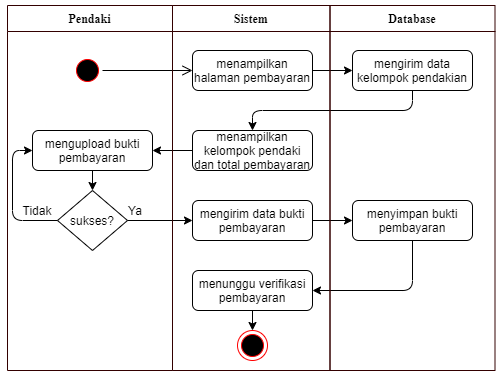
**Gambar 3.4** Activity Diagram Update Akun

1. *Activity Diagram* Daftar Pendakian



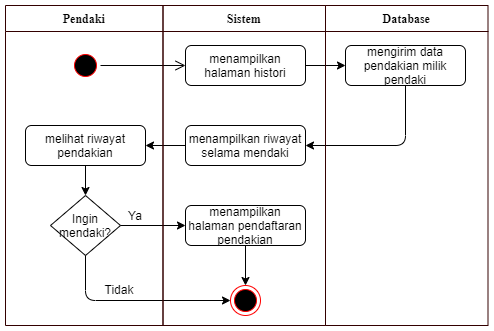
**Gambar 3.5** Activity Diagram Daftar Pendakian

1. *Activity Diagram* Pembayaran Pendakian



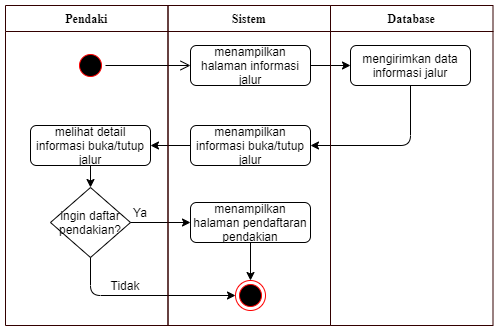
**Gambar 3.6** Activity Diagram Pembayaran Pendakian

1. *Activity Diagram* Riwayat Pendakian



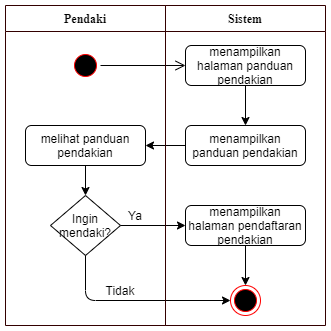
**Gambar 3.7** Activity Diagram Riwayat Pendakian

1. *Activity Diagram* Informasi Buka/Tutup Pendakian



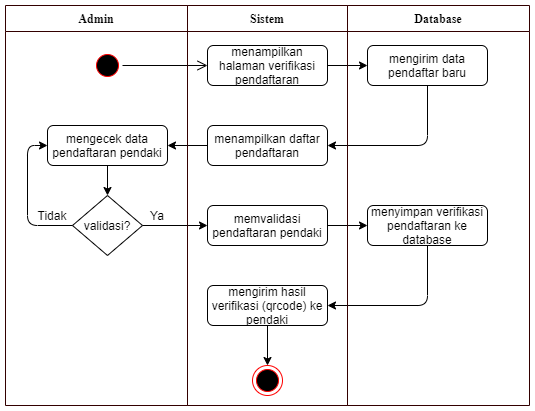
**Gambar 3.8** Activity Diagram Informasi Buka/Tutup Pendakian

1. *Activity Diagram* Panduan Pendakian



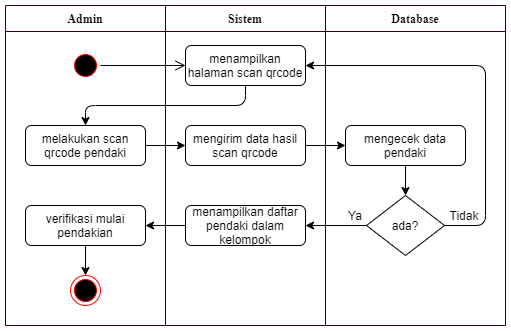
**Gambar 3.9** Activity Diagram Panduan Pendakian

1. *Activity Diagram* Verifikasi Pendaftaran Pendakian



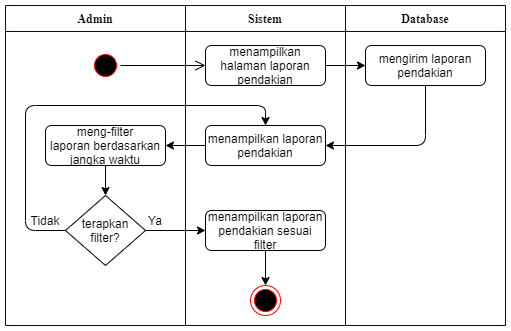
**Gambar 3.10** Activity Diagram Verifikasi Pendaftaran Pendakian

1. *Activity Diagram Scan* QRCode Pendaki



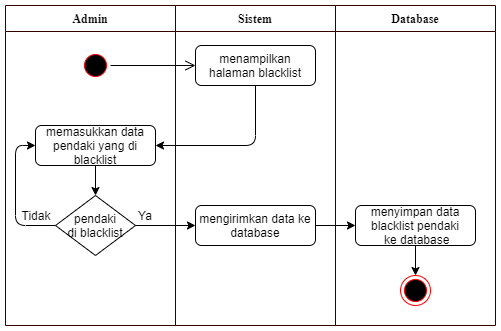
**Gambar 3.11** Activity Diagram Scan QRCode Pendaki

1. *Activity Diagram* Laporan Pendakian



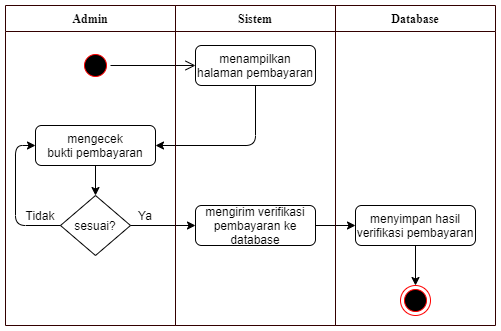
**Gambar 3.12** Activity Diagram Laporan Pendakian

1. *Activity Diagram Blacklist* Pendakian



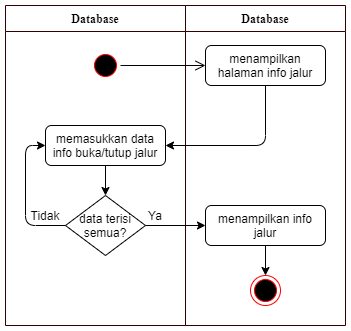
**Gambar 3.13** Activity Diagram Blacklist Pendakian

1. *Activity Diagram* Verifikasi Pembayaran



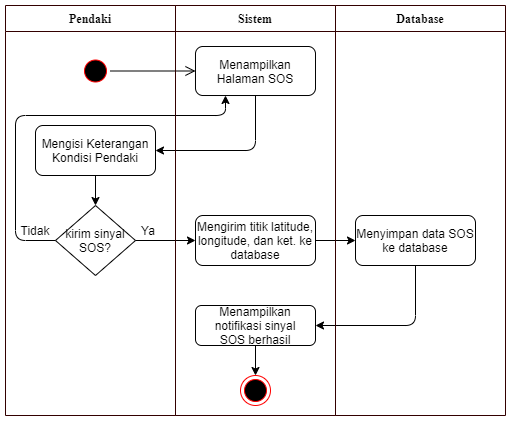
**Gambar 3.14** Activity Diagram Verifikasi Pembayaran

1. *Activity Diagram* Informasi Jalur Pendakian



**Gambar 3.15** Activity Diagram Informasi Jalur Pendakian

1. *Activity Diagram* Sinyal SOS Pendaki



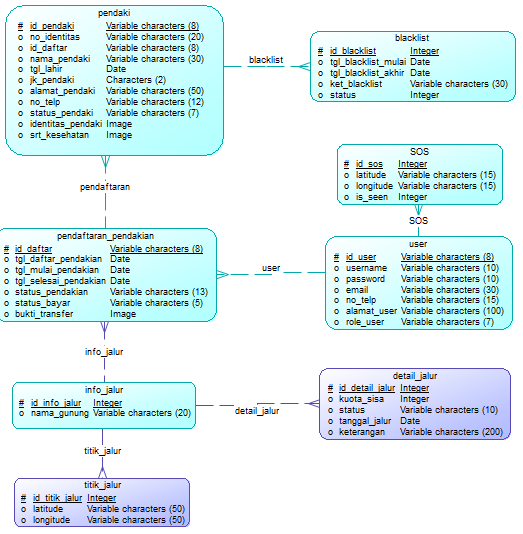
**Gambar 3.16** Activity Diagram Sinyal SOS Pendaki

### **Perancangan Data**

Pada penelitian ini, rancangan basis data atau *data*base digunakan untuk menentukan isi dan pengaturan data yang diperlukan yang mencakup CDM (*Conceptual Data Model*), PDM (*Physical Data Model*), dan Desain Tabel.

#### **CDM (Conceptual Data Model)**

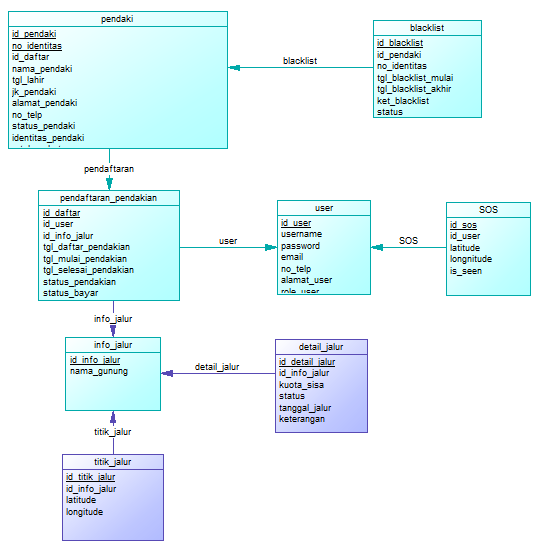
CDM atau *Conceptual Data Model* adalah konsep gambaran secara detail mengenai basis data dalam bentuk *logic*. CDM dibuat dalam bentuk tabel-tabel tanpa tipe data yang menggambarkan relasi antar tabel pada implementasi ke *database*. Pada sistem yang akan dibuat, *conceptual data model* sebagai berikut:



**Gambar 3.17** Conceptual Data Model (CDM)

#### **PDM (Physical Data Model)**

PDM atau *Physical Data Model* adalah gambaran secara detail mengenai basis data dalam bentuk fisik. Pada penerapan PDM dapat disamakan dengan skema relasi yang memiliki fungsi untuk memodelkan struktur fisik dan secara detail dari suatu basis data. PDM dalam sistem yang akan dibuat sebagai berikut:



**Gambar 3.18** Physical Data Model (PDM)

#### **Desain Tabel**

1. Tabel User

Tabel user digunakan untuk menyimpan data *user* atau pengguna yang menggunakan aplikasi yang mencakup id\_user, username, password, email, no\_telp, alamat\_user, dan role\_user. Role\_user berfungsi untuk menentukan hak akses pengguna dalam aplikasi yang meliputi admin dan pendaki. Tabel user pada sistem yang akan dibangun sebagai berikut:

**Tabel 3.3** Tabel User

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_user | Varchar (8) | Primary Key |
| username | Varchar (10) | - |
| password | Varchar (10) | - |
| email | Varchar (30) | - |
| no\_telp | Varchar (15) | - |
| alamat\_user | Varchar (100) | - |
| role\_user | Varchar (7) | - |

1. Tabel Pendaki

Tabel Pendaki berfungsi untuk menyimpan data identitas pendaki. Pada tabel pendaki, terdapat data status\_pendaki yang digunakan untuk menentukan status pada pendakian yaitu pendaki sebagai *leader* atau anggota. Data no\_identitas berisi tentang nomor identitas pendaki seperti nomor KTP.

**Tabel 3.4** Tabel Pendaki

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_pendaki | Varchar (8) | Primary Key |
| no\_identitas | Varchar (20) | Primary Key |
| id\_daftar | Varchar (8) | Foreign Key |
| nama\_pendaki | Varchar (30) | - |
| tgl\_lahir | Date | - |
| jk\_pendaki | Char (1) | - |
| alamat\_pendaki | Varchar (50) | - |
| no\_telp | Varchar (12) | - |
| status\_pendaki | Varchar (7) | - |
| identitas\_pendaki | Image | - |
| srt\_kesehatan | Image | - |

1. Tabel Pendaftaran Pendakian

Tabel Pendaftaran Pendakian berfungsi sebagai tabel penyimpanan transaksi pendaftaran pendakian. Pada tabel status\_pendakian berisi tentang verifikasi pendaftaran pendakian yang meliputi sudah, belum, dan batal pendakian. Sedangkan status\_bayar berisi tentang verifikasi pembayaran pendakian.

**Tabel 3.5** Tabel Pendaftaran Pendakian

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_daftar | Varchar (8) | Primary Key |
| id\_user | Varchar (8) | Foreign Key |
| id\_info\_jalur | Integer | Foreign Key |
| tgl\_daftar\_pendakian | Date | - |
| tgl\_mulai\_pendakian | date | - |
| tgl\_selesai\_pendakian | Date | - |
| status\_pendakian | Varchar (13) | - |
| status\_bayar | Varchar (5) | - |
| bukti\_transfer | Image | - |

1. Tabel Blacklist

Tabel Blacklist digunakan untuk menyimpan data pendaki yang di *blacklist* atau pemblokiran pendaki yang melakukan pelanggaran dalam pendakian. Data pada tabel ket\_blacklist berisi tentang keterangan penyebab pendaki di *blacklist*.

**Tabel 3.6** Tabel Blacklist

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_blacklist | Integer | Primary Key |
| id\_pendaki | Varchar (8) | Foreign Key |
| no\_identitas | Varchar (20) | Foreign Key |
| tgl\_blacklist\_mulai | Date | - |
| tgl\_blacklist\_akhir | Date | - |
| ket\_blacklist | Varchar (30) | - |
| Status | integer | - |

1. Tabel Info Jalur

Tabel Info Jalur digunakan untuk menyimpan data nama gunung pada jalur pendakian.

**Tabel 3.7** Tabel Info Jalur

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_info\_jalur | Integer | Primary Key |
| nama\_gunung | Varchar (20) | - |

1. Tabel SOS

Tabel SOS digunakan untuk menyimpan titik latitude dan longitude dari pendaki yang membutuhkan pertolongan di jalur pendakian. Tabel SOS memiliki relasi dengan tabel user karena semua pendaki yang mempunyai aplikasi dapat mengirimkan sinyal SOS.

**Tabel 3.8** Tabel SOS

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_sos | Integer | Primary Key |
| id\_user | Varchar (8) | *Foreign Key* |
| latitude | Varchar (15) |  |
| longitude | Varchar (15) |  |
| is\_seen | Integer |  |

1. Tabel Detail Jalur

Tabel Detail Jalur digunakan untuk menyimpan data jadwal dan kuota dari jalur pendakian. Tabel Detail Jalur berisi field id\_detail\_jalur, id\_info\_jalur, kuota\_sisa, status, tanggal\_jalur, dan keterangan dari jalur pendakian.

**Tabel 3.9** Tabel Detail Jalur

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_detail\_jalur | Integer | Primary Key |
| id\_info\_jalur | Integer | Foreign Key |
| kuota\_sisa | Integer | - |
| Status | Varchar (10) | - |
| tanggal\_jalur | Date | - |
| keterangan | Varchar (200) | - |

1. Tabel Titik Jalur

Tabel Titik Jalur digunakan untuk menyimpan titik latitude dan longitude dari jalur pendakian. Titik latitude dan longitude dari tabel titik jalur akan membentuk rute dari jalur pendakian di Gunung Panderman dan Gunung Buthak.

**Tabel 3.10** Tabel Titik Jalur

|  |  |  |
| --- | --- | --- |
| **Field** | **Tipe Data** | **Keterangan** |
| id\_titik\_jalur | Integer | Primary Key |
| id\_info\_jalur | Integer | Foreign Key |
| Latitude | Varchar (50) | - |
| longitude | Varchar (50) | - |

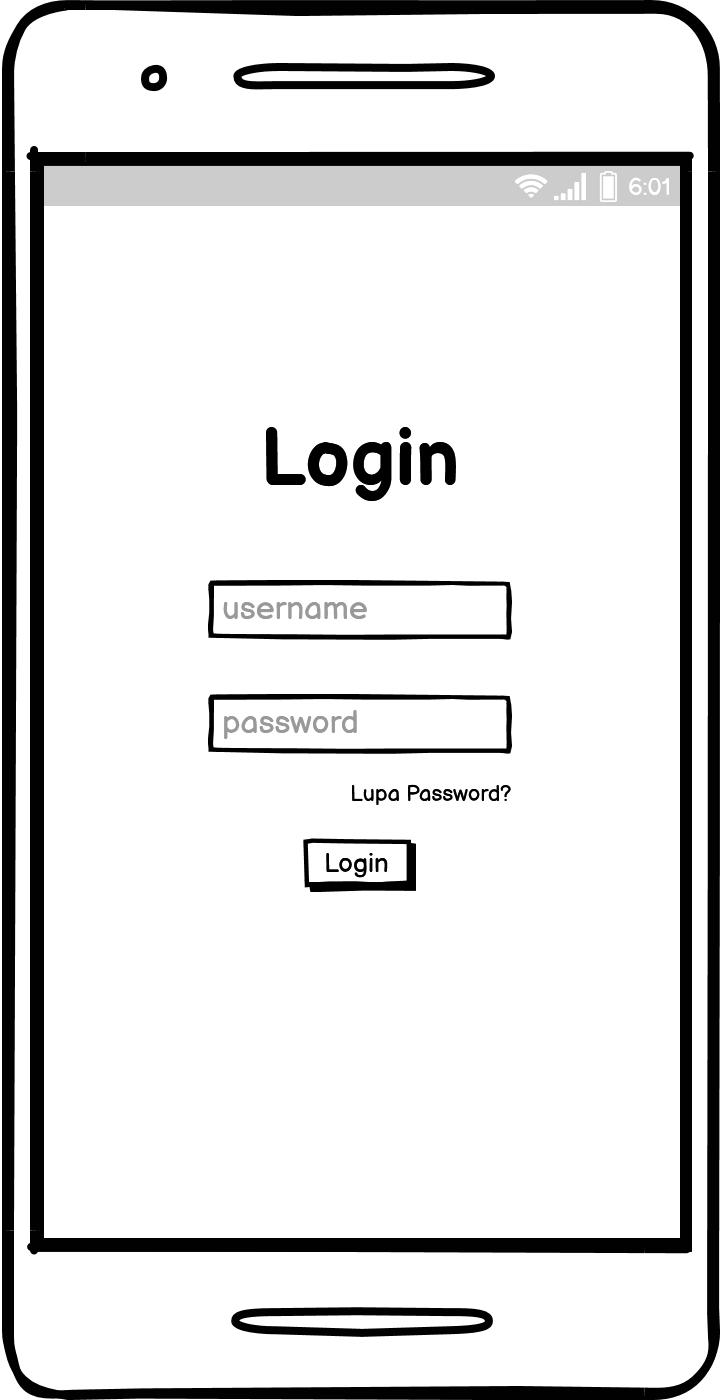
### **Perancangan User Interface**

Dibawah ini adalah rancangan tampilan dari Aplikasi Manajemen Pendakian Gunung Panderman-Buthak Berbasis Android Menggunakan Metode *Scrum*:

#### **Admin**

Rancangan *user interface* yang dibuat adalah rancangan untuk user Admin Pos Pendakian. *User interface* pertama adalah tampilan dari halaman Splash Screen dan halaman Login aplikasi.





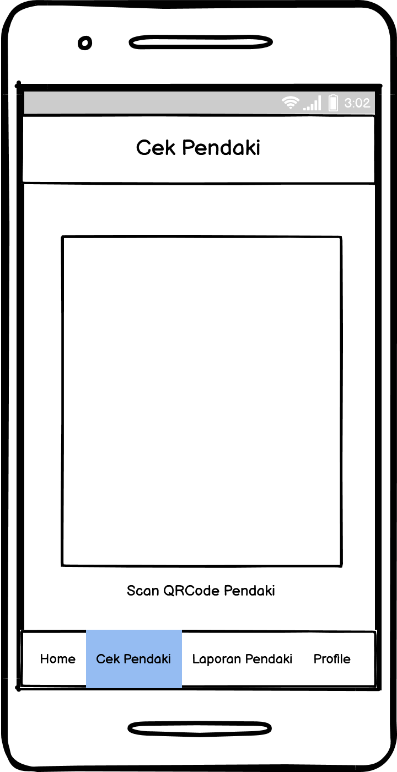
**Gambar 3.19** Halaman Splash Screen dan Login Admin

Rancangan *user interface* akan memiliki empat menu utama yaitu menu Home, menu Cek Pendaki, menu Laporan Pendaki, dan menu Profile. Dari keempat menu tersebut, penulis merancang untuk halaman Home yang dapat dilihat pada Gambar 3.20. Halaman Home Admin memiliki fitur sebagai tambah jadwal dan kuota pendakian.

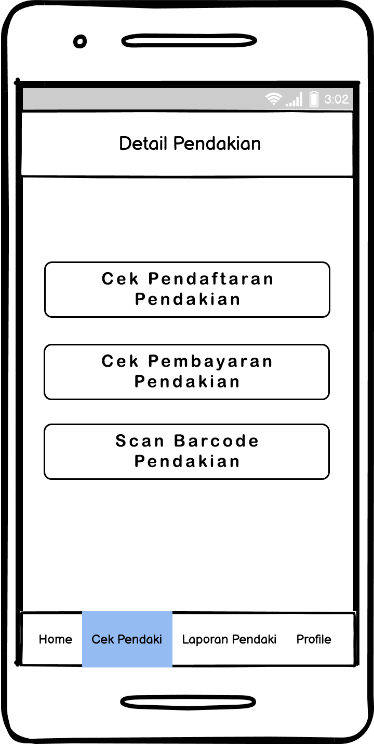
 ****

**Gambar 3.20** Halaman Utama Admin Pendakian

Rancangan *user interface* selanjutnya yaitu halaman Cek Pendaki. Halaman Cek Pendaki akan digunakan sebagai pemindai (*scan*) QR Code pendaki.



**Gambar 3.21** Halaman Scan qrcode Pendaki

****

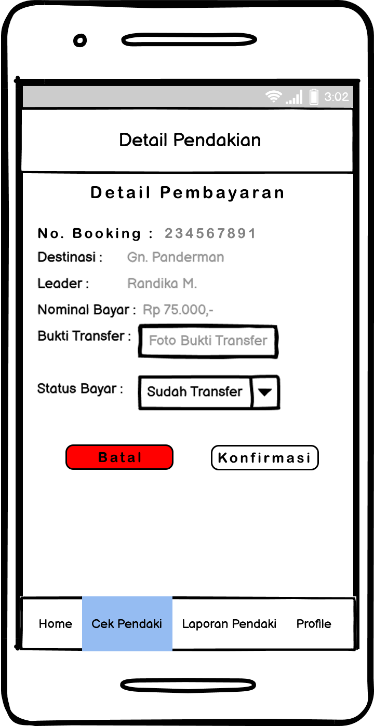
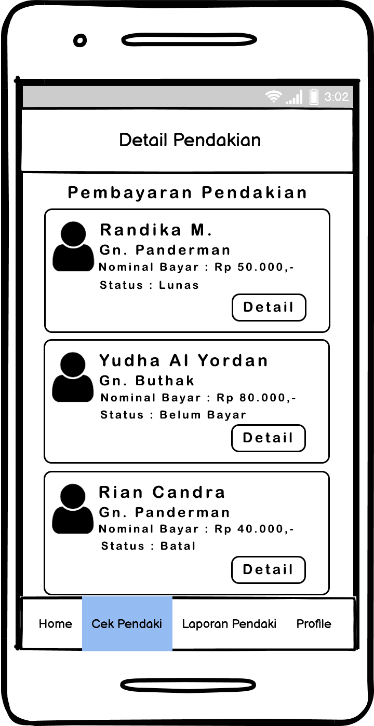
**Gambar 3.22** Halaman Menu Admin Cek Pendaki

Rancangan *user interface* pada Gambar 3.22 merupakan halaman Cek Pendaki yang memiliki tiga menu utama, yaitu menu Cek Pendaftaran Pendakian, menu Cek Pembayaran Pendakian, dan menu Scan Barcode Pendakian. Rancangan *user interface* menu Cek Pendaftaran Pendakian dapat dilihat pada Gambar 3.23.

**** ****

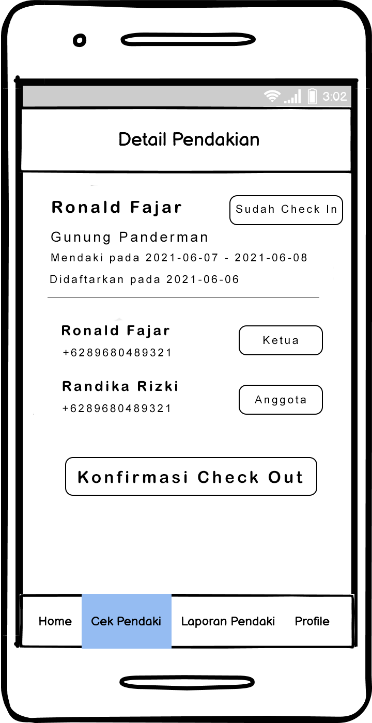
**Gambar 3.23** Halaman Admin Pendaftaran Pendakian

Rancangan *user interface* untuk menu Cek Pembayaran Pendakian dapat dilihat pada Gambar 3.24. Pada halaman ini Admin melakukan pengecekan dan validasi pembayaran yang dilakukan oleh pendaki.



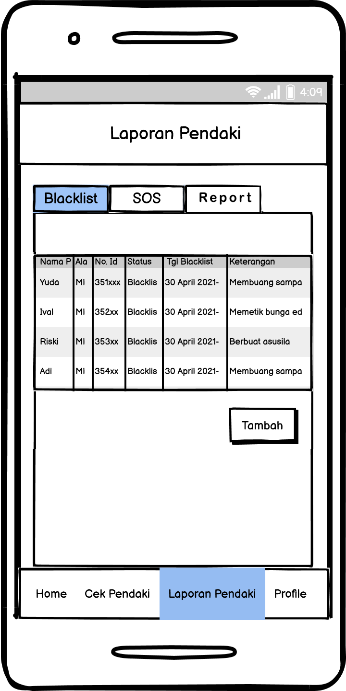
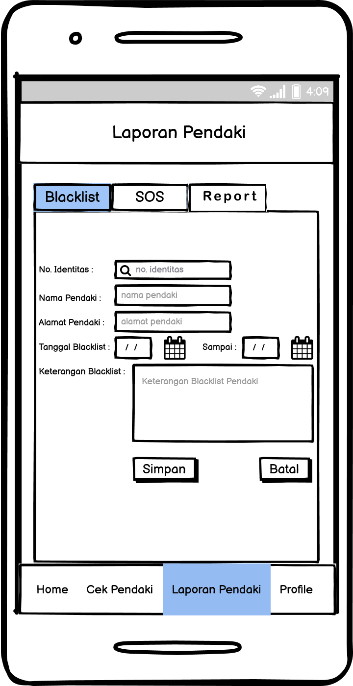
**Gambar 3.24** Halaman Admin Pembayaran Pendakian

Pada Gambar 3.25 dibawah ini merupakan halaman dari hasil pemindaian QR Code pendaki. Pada halaman ini, Admin akan melakukan validasi pendakian yaitu *check in* dan *check out*.



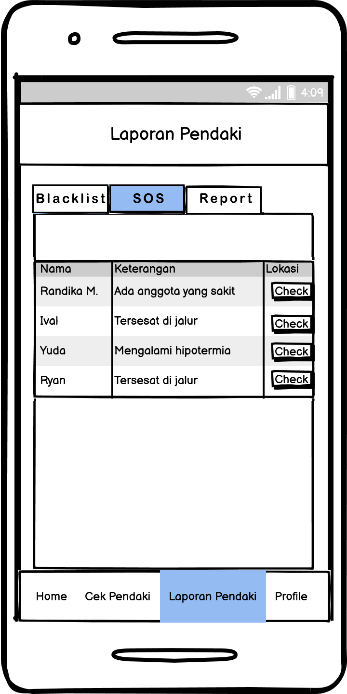
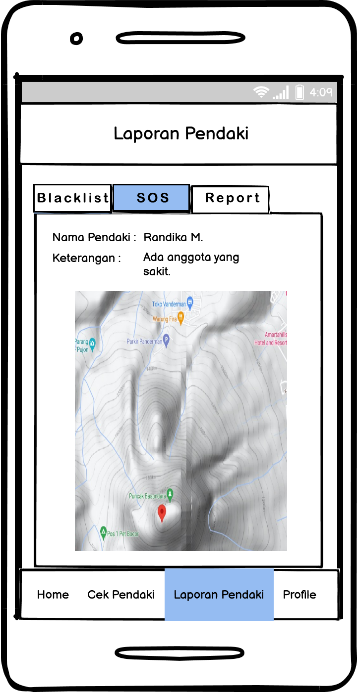
**Gambar 3.25** Halaman Detail Pendakian Hasil Scan qrcode Pendaki

Rancangan *user interface* pada menu Laporan Pendaki memiliki tiga fitur, yaitu Blacklist, SOS, dan Report. Fitur Blacklist dapat digunakan Admin untuk melakukan pemblokiran pendakian kepada pendaki yang melakukan pelanggaran. Rancangan fitur Blacklist dapat dilihat di Gambar 3.26 dibawah ini.

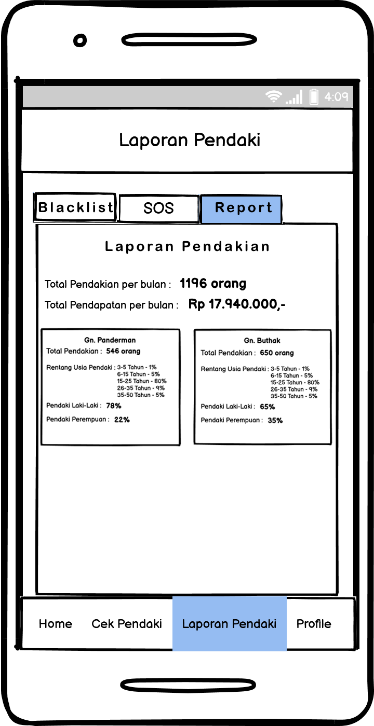
**Gambar 3.26** Halaman Blacklist Pendaki di Aplikasi

Rancangan *user interface* pada fitur SOS dapat digunakan Admin untuk memperoleh informasi lokasi ketika terdapat pendaki yang membutuhkan pertolongan dalam pendakian. Rancangan tampilan untuk fitur SOS dapat dilihat pada Gambar 3.27.

**Gambar 3.27** Halaman Penerimaan Sinyal SOS di Aplikasi

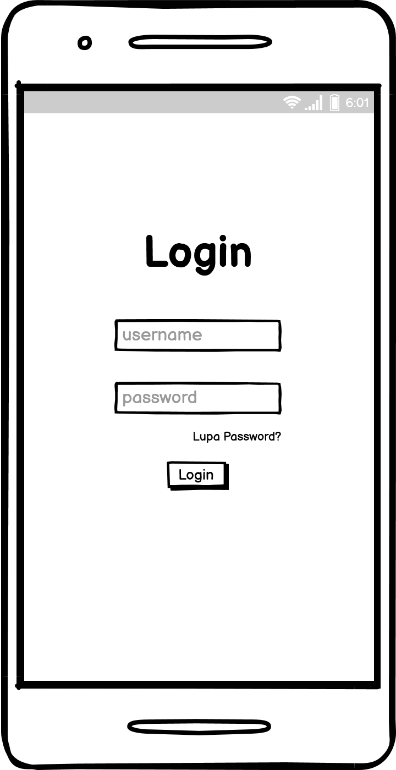
Rancangan *user interface* fitur Report digunakan Admin sebagai laporan pendakian selama satu bulan yang meliputi total pendakian, total pendapatan pendakian, dan lainnya. Rancangan tampilan fitur Report dapat dilihat pada Gambar 3.28.

****

**Gambar 3.28** Halaman Admin Laporan Pendakian

#### **Pendaki**

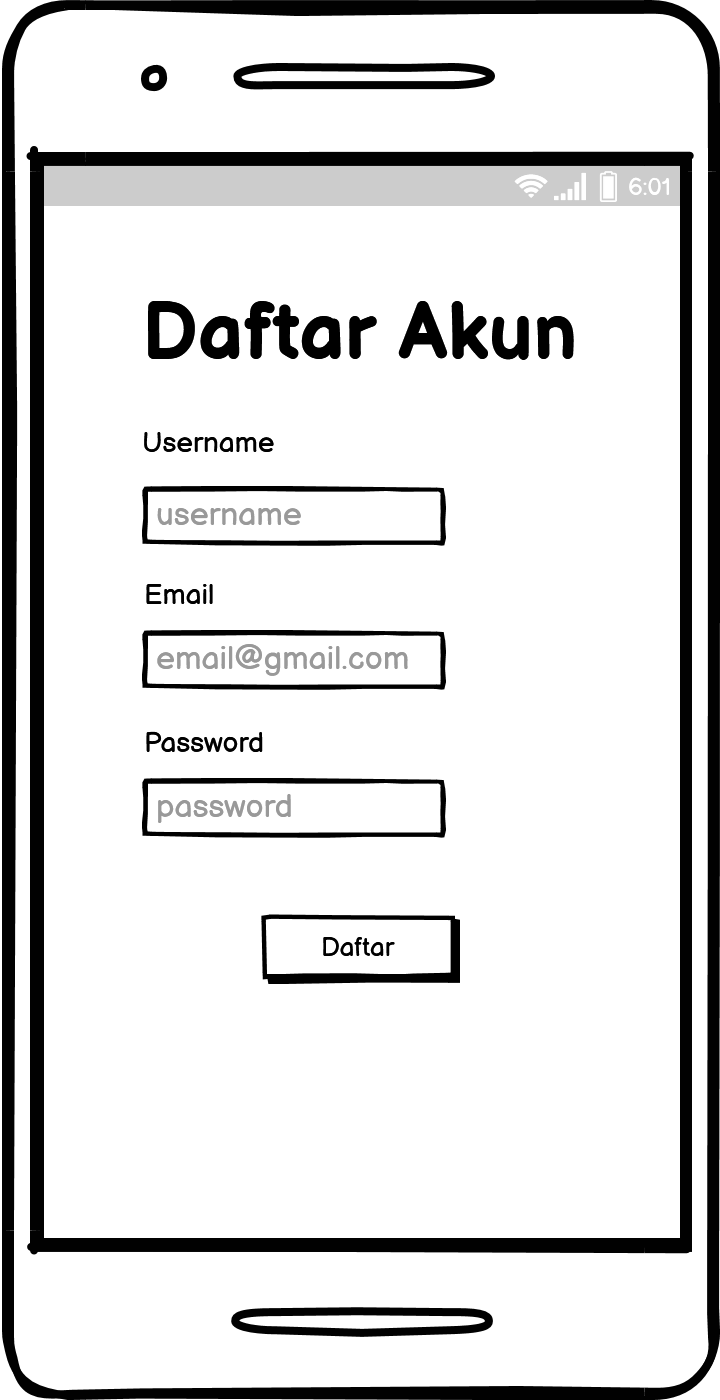
Pada penelitian yang dilakukan oleh penulis, rancangan *user interface* juga dilakukan untuk user pendaki. Halaman awal dari *user interface* pendaki sama dengan admin yaitu terdapat halaman Splash Screen dan halaman Login





**Gambar 3.29** Halaman Splash Screen dan Login Pendaki

Rancangan user interface selanjutnya yaitu halaman Daftar Akun. Halaman ini akan digunakan pendaki untuk melakukan pendaftaran akun aplikasi agar dapat masuk kedalam aplikasi. Rancangan user interface halaman Daftar Akun dapat dilihat pada Gambar 3.30.



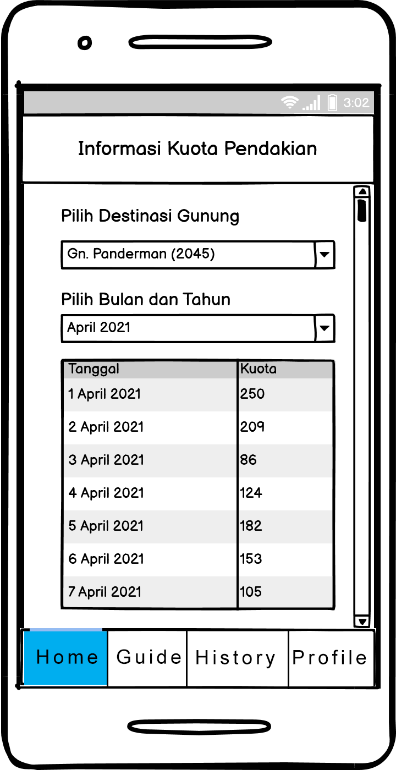
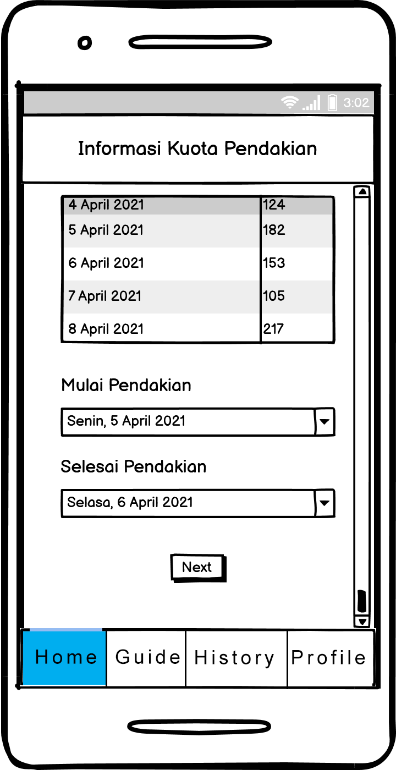
**Gambar 3.30** Halaman Daftar Akun Pendaki

Rancangan *user interface* untuk pendaki selanjutnya yaitu halaman Home yang digunakan ketika pendaki sudah melakukan *login* akan masuk ke halaman ini. Rancangan halaman Home dapat dilihat pada Gambar 3.31.



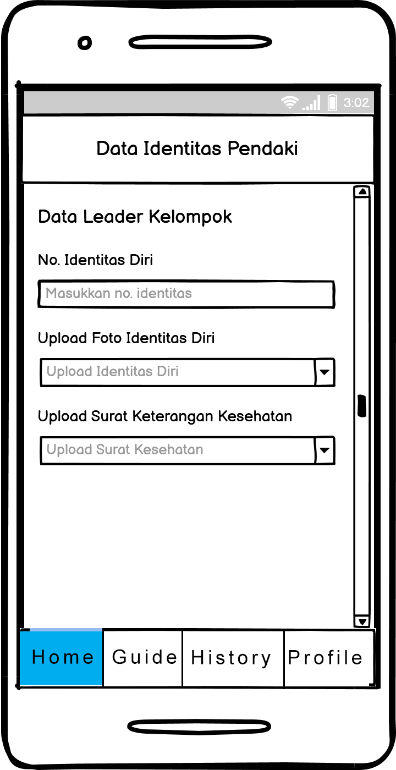
**Gambar 3.31** Halaman Home Pendaki

Rancangan *user interface* yang dibuat adalah menu Booking yang digunakan pendaki untuk melakukan pendaftaran pendakian. Pendaki dapat melihat jadwal dan kuota pendakian sesuai dengan jalur yang dipilih. Rancangan tersebut dapat dilihat pada Gambar 3.32

**Gambar 3.32** Halaman Booking Pendakian

Setelah halaman booking, penulis merancang halaman Data Identitas Pendaki yang akan meliputi nama, alamat, nomor identitas, hingga persyaratan pendakian. Rancangan user interface tersebut dapat dilihat pada Gambar 3.33 untuk halaman Data Identitas Leader dan Gambar 3.34 untuk halaman Data Identitas Anggota.

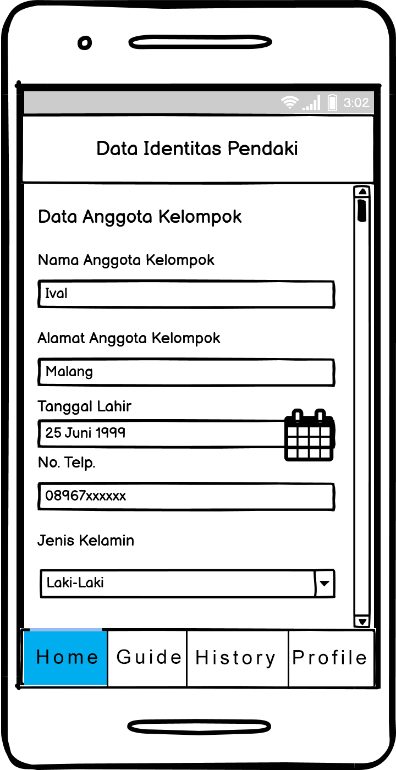
 

**Gambar 3.33** Halaman Pendaftaran Leader Pendakian

**Gambar 3.34** Halaman Tambah Anggota Pendakian

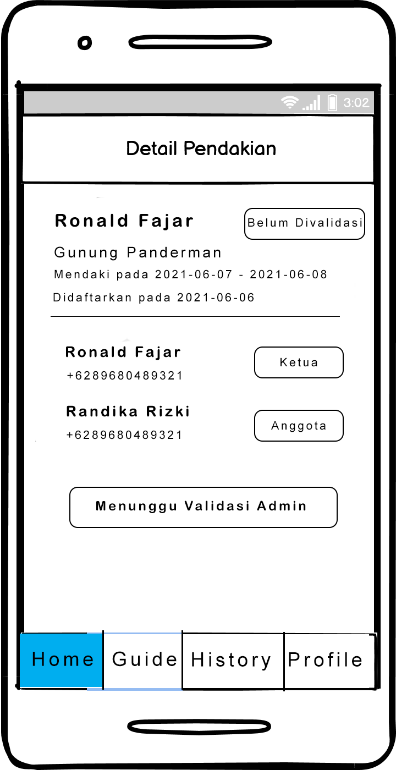
Pendaki juga dapat merubah data identitas ketika pendaftaran, oleh karena itu dirancang halaman Ubah Data Pendaki seperti pada Gambar 3.35 dibawah ini.





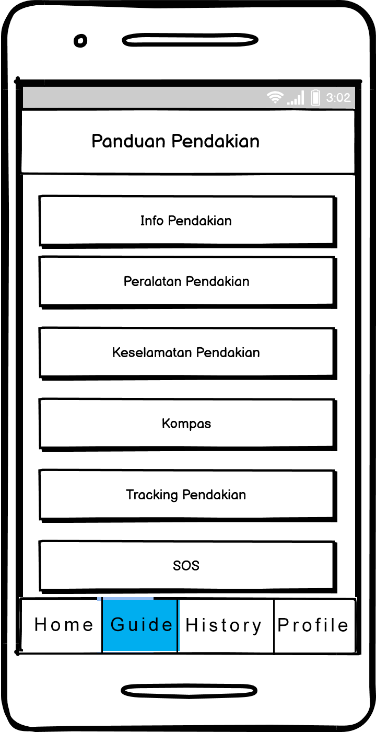
**Gambar 3.35** Halaman Ubah Data Pendaki

Penulis juga merancang halaman Detail Pendakian yang akan menampilkan detail dari pendaftaran yang sudah dilakukan seperti jalur pendakian, tanggal pendakian, data identitas pendaki, dan status pendaftaran yang dapat dilihat pada Gambar 3.36.



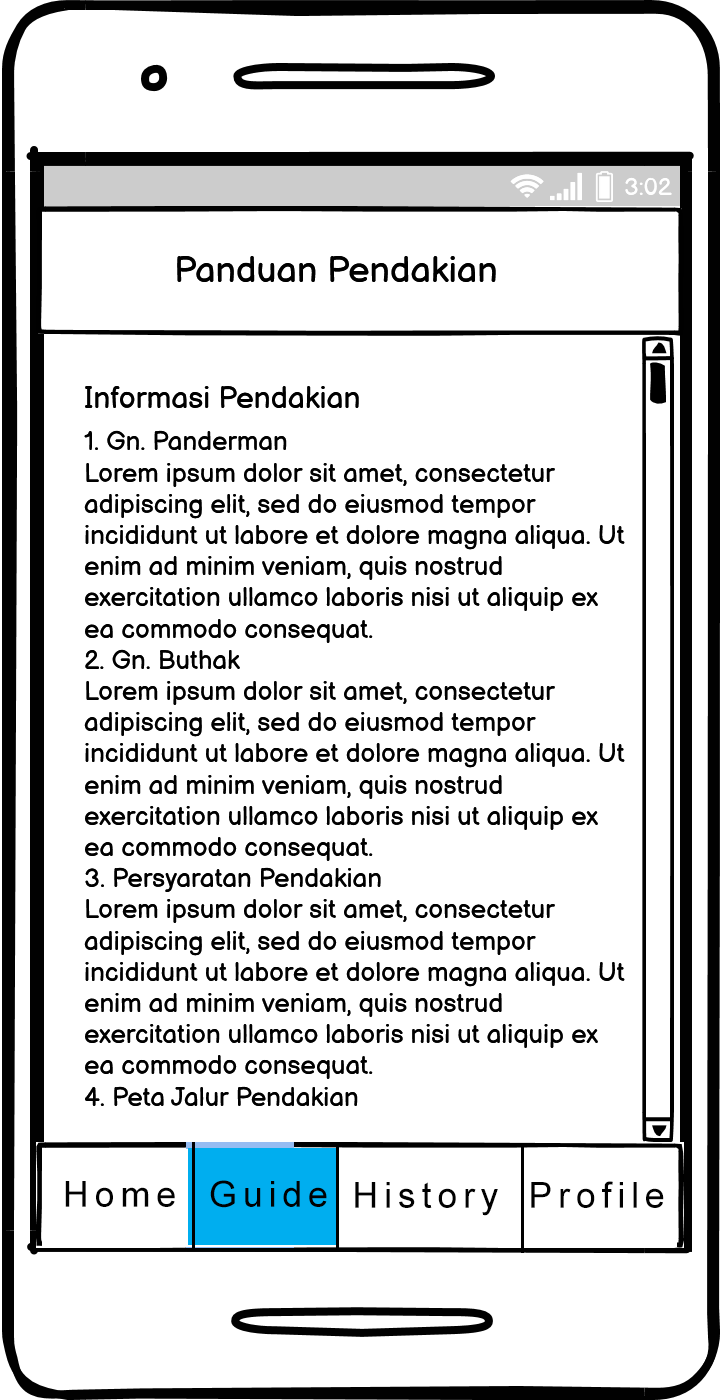
**Gambar 3.36** Halaman Pendaftaran Pendakian

Pada menu Guide, penulis merancang *user interface* yang akan meliputi fitur Info Pendakian, Peralatan Pendakian, Keselamatan Pendakian, Kompas, Tracking Pendakian, dan SOS.



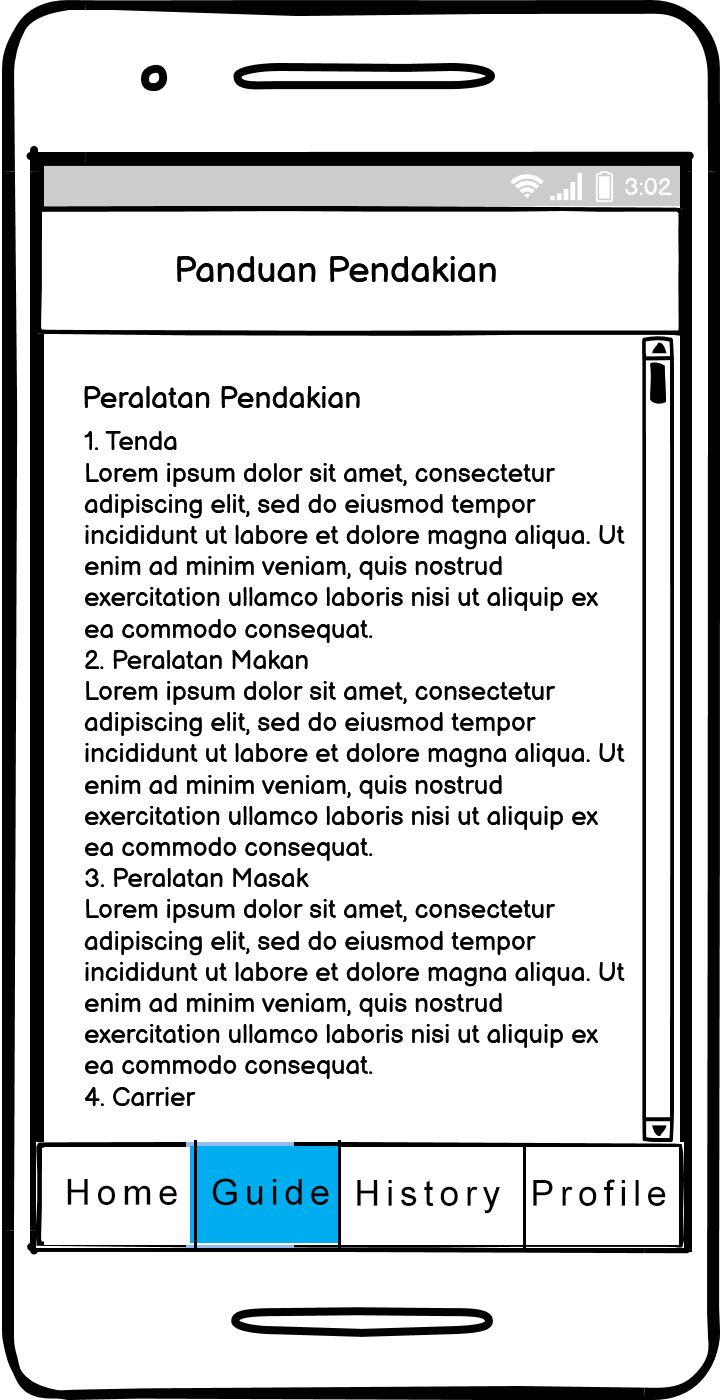
**Gambar 3.37** Halaman Guide

Rancangan halaman Info Pendakian akan berisi informasi tentang Pos Pendakian Gunung Panderman-Buthak.



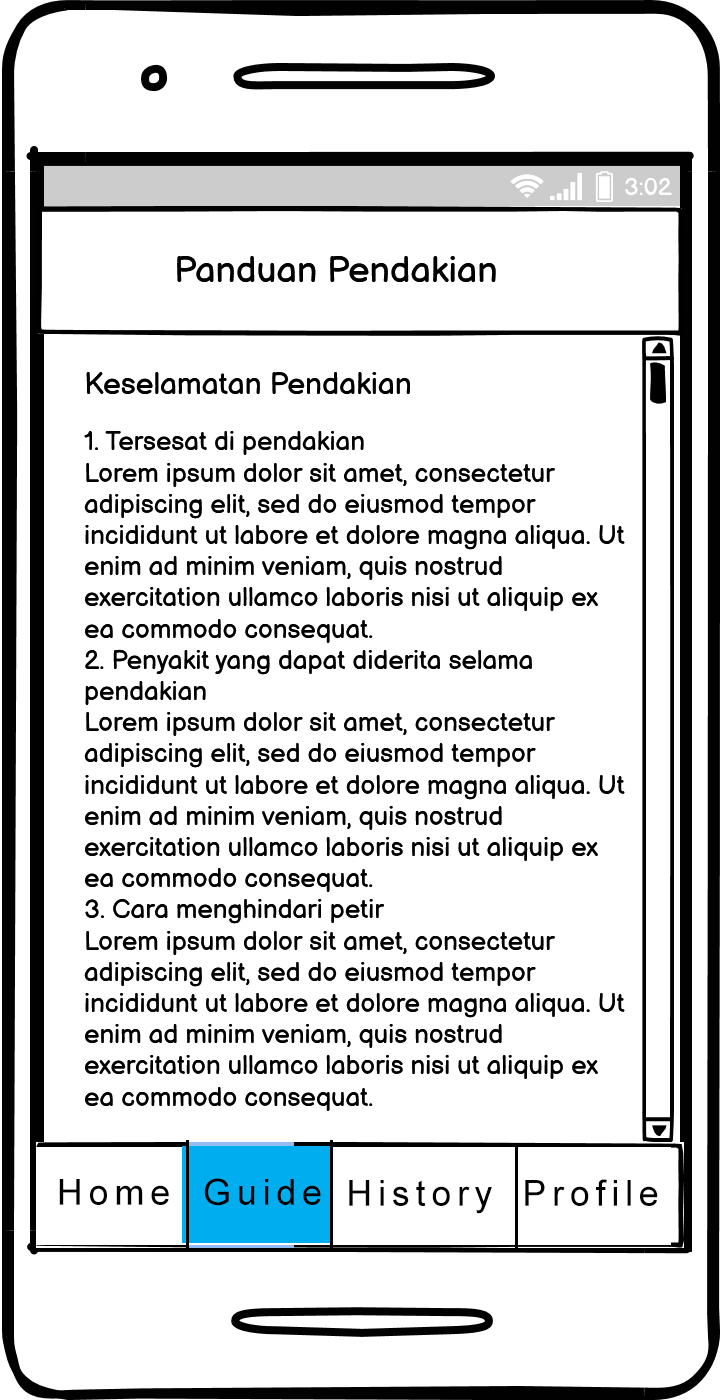
**Gambar 3.38** Halaman Informasi Pendakian

Rancangan halaman Peralatan Pendakian berisi tentang informasi peralatan pendakian yang disarankan untuk para pendaki.



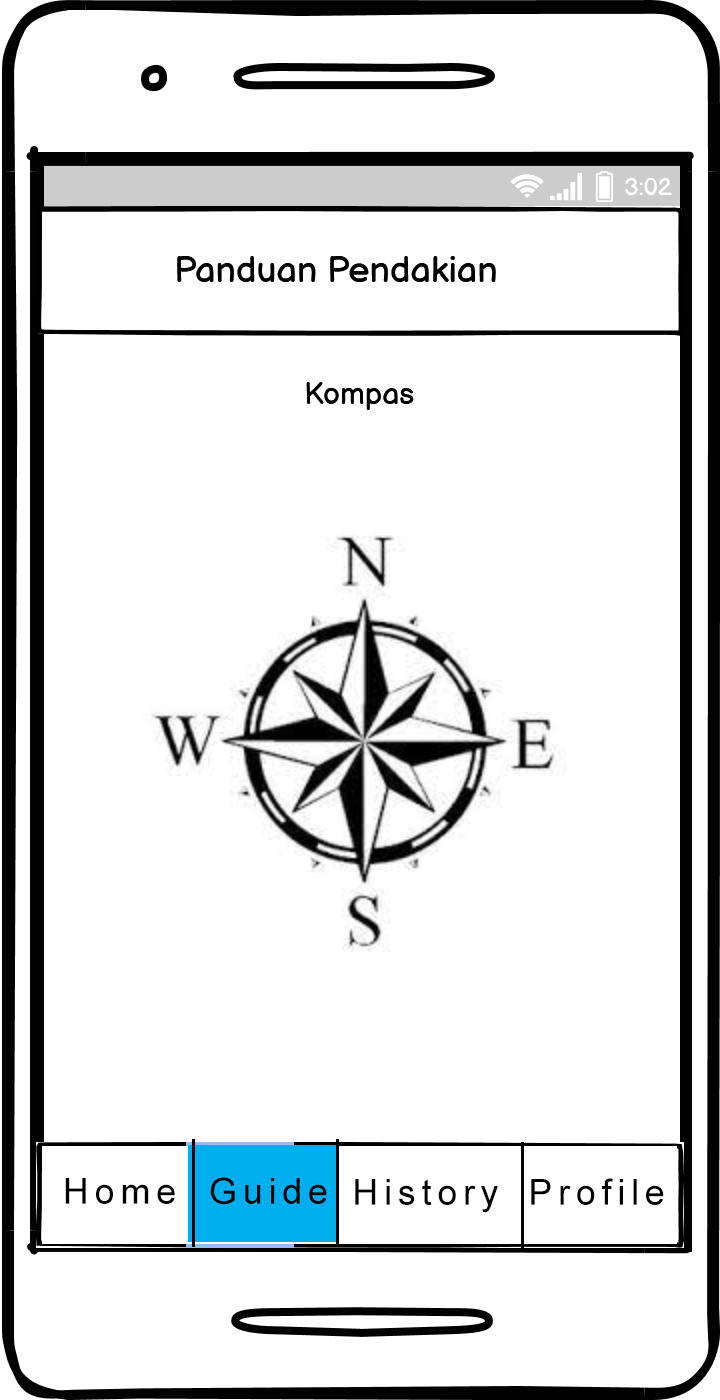
**Gambar 3.39** Halaman Peralatan Pendakian

Rancangan pada halaman Keselamtan Pendakian akan berisi tentang informasi keselamatan dalam pendakian.



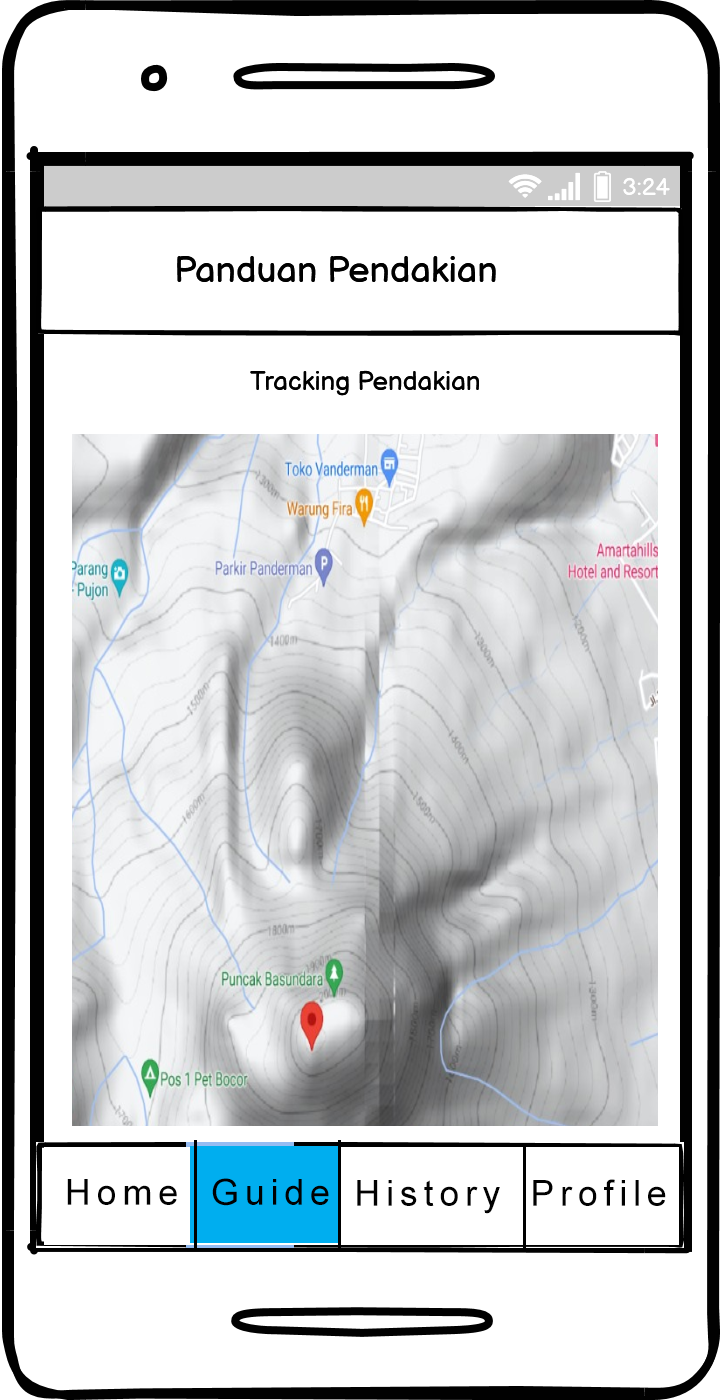
**Gambar 3.40** Halaman Keselamatan Pendakian

Rancangan halaman Kompas berisi fitur penunjuk arah mata angin untuk membantu pendaki dalam jalur pendakian.



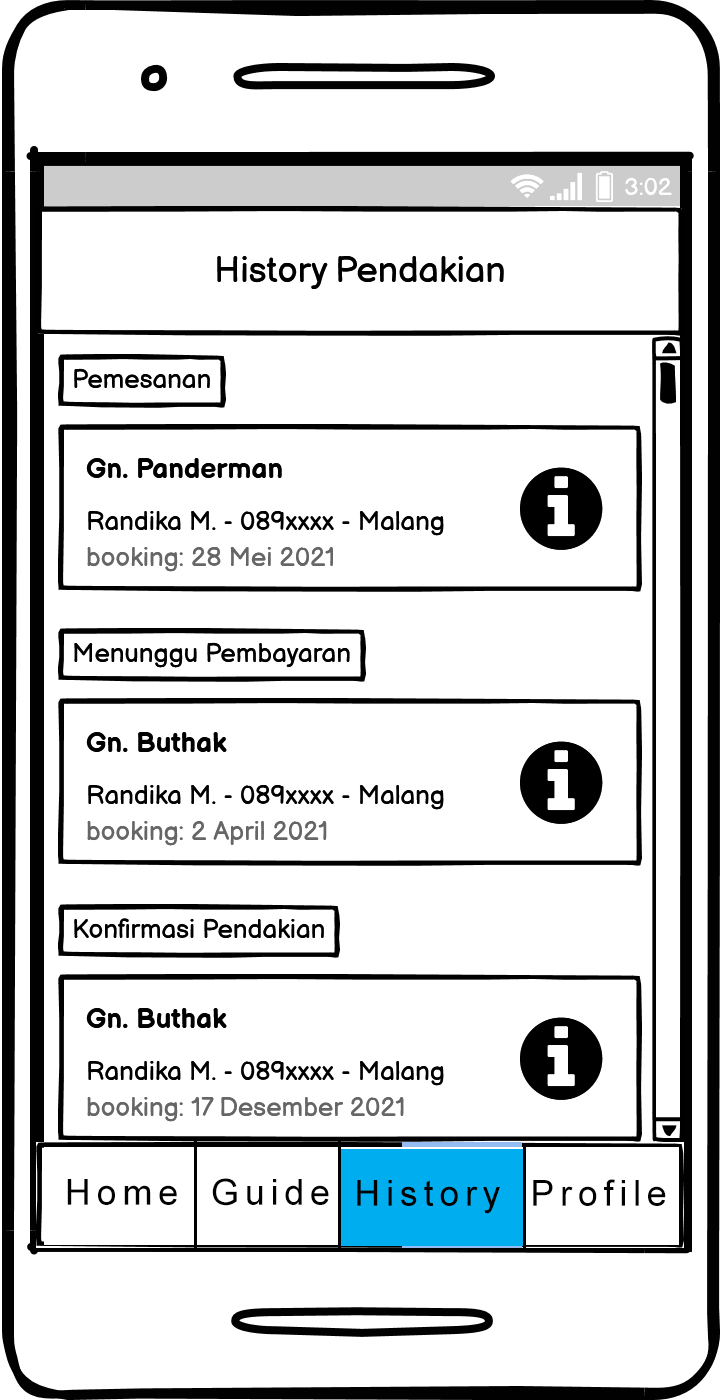
**Gambar 3.41** Halaman Kompas

Rancangan halaman Tracking Pendakian berisi tampilan peta dengan jalur pendakian yang dipilih sehingga pendaki tidak tersesat dalam jalur pendakian.



**Gambar 3.42** Halaman Peta Tracking Pendakian

Pada Gambar 3.43 merupakan halaman History Pendakian yang menampilkan riwayat dari pendakian yang dilakukan oleh pendaki.



**Gambar 3.43** Halaman Histori Pendakian



**Gambar 3.44** Halaman Pembayaran Pendakian

Pada Gambar 3.45 merupakan rancangan halaman dalam menampilkan QR Code pendakian sehingga dapat digunakan pendaki untuk proses *check in* dan *check out*.



**Gambar 3.45** Halaman qrcode Pendaki

Pada Gambar 3.46 merupakan halaman Profile untuk pendaki yang mana akan menampilkan data pendaki.



**Gambar 3.46** Halaman Profile

## **Rancangan Pengujian**

Dalam penelitian ini, peneliti menggunakan *Blackbox Testing* untuk tahap pengujian sistem. *Blackbox Testing* adalah pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, penguji dapat mendefinisikan kumpulan kondisi *input* dan melakukan pengecekan pada spesifikasi fungsional program. Dalam pengujian ini, seorang tester menguji sistem tanpa mengetahui struktur internal kode atau program. Penguji atau tester menyadari hal yang harus dilakukan oleh program namun tidak memiliki pengetahuan tentang cara untuk melakukannya. Pengujian sistem berdasarkan Blackbox Testing yaitu:

1. Fungsi salah atau hilang.
2. Kesalahan antarmuka.
3. Kesalahan dalam struktur data atau akses basis data eksternal.
4. Kesalahan kinerja.
5. Kesalahan terminasi.

Metode Blackbox Testing memperlakukan sistem yang di uji coba dengan tidak mengetahui kinerja internalnya, sehingga tester memandang sistem tersebut seperti layaknya “kotak hitam” yang tidak penting dilihat isinya, namun cukup dikenai proses testing dibagian luar. Dalam Blackbox Testing, sistem tersebut akan dieksekusi kemudian di tes apakah telah memenuhi kebutuhan pengguna yang didefinisikan pada saat awal tanpa harus membongkar isi program. Penguji akan mencari fungsi yang hilang atau salah, antarmuka, kinerja, inisialisasi program dan kesalahan *output*, atau kesalahan akses basis data eksternal. Setiap tes akan mengikuti skenario, memeriksa setiap fungsi, jika input yang valid diterima, yang tidak valid ditolak, dan jika *output* atau hasil yang diharapkan; atau dapat disebut metode “coba-coba”.

INPUT

Blackbox

OUTPUT

**Gambar 3.47** Blackbox Testing