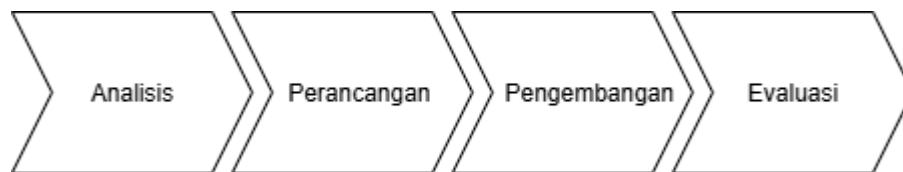


BAB III

METODE PENELITIAN

3.1 Metode Penelitian

Metode penelitian *Design and Development* (D&D) adalah pendekatan sistematis yang bertujuan untuk mengembangkan produk atau model. Pada penelitian ini akan mengikuti langkah-langkah atau “*Specific Project Phases*” pada kluster tipe pertama yakni *Product and Tool Research*, sebab fokus utama pada penelitian ini adalah pengembangan dan pengujian sistem celengan pintar berbasis IoT dengan teknologi *computer vision* untuk deteksi nominal uang, dengan tahapan operasional *Analysis, Design, Development, dan Evaluation* (Richey & Klein, 2007). Oleh karena itu, pada penelitian ini akan ditetapkan berdasarkan alur dari empat tahapan tersebut, yang dapat dilihat pada Gambar 3.1.



Gambar 3. 1 Tahapan Metode *Design and Development* (D&D)

Dalam hal ini, penelitian berfokus pada pengembangan sistem celengan pintar berbasis IoT dengan teknologi *computer vision*. D&D pada penelitian ini menggunakan pendekatan dengan empat tahap, yaitu Analisis, Perancangan, Pengembangan, dan Evaluasi. Ini sesuai dengan langkah-langkah yang diperlukan untuk mengembangkan sistem seperti celengan pintar yang melibatkan integrasi perangkat keras (*hardware*) dan perangkat lunak (*software*).

3.2 Analisis

Tahap analisis pada penelitian ini berfungsi sebagai dasar dalam pengembangan sistem celengan pintar berbasis IoT dan *computer vision*. Analisis dimulai dengan mengkaji latar belakang permasalahan, yaitu kebiasaan menabung secara tradisional yang masih memiliki keterbatasan dalam hal transparansi, keamanan, dan motivasi bagi penggunaanya, terutama anak-anak. Dari permasalahan tersebut, ditetapkan tujuan penelitian untuk merancang sebuah sistem celengan pintar yang tidak hanya mampu menyimpan uang secara fisik, tetapi juga

dapat mendeteksi nominal uang secara otomatis, mencatat transaksi tabungan, dan menampilkan hasil *monitoring* melalui antarmuka *website*.

Pada tahap ini juga dilakukan identifikasi kebutuhan sistem, baik dari sisi fungsional maupun nonfungsional. Kebutuhan fungsional meliputi kemampuan perangkat keras untuk menerima uang kertas maupun koin, deteksi nominal uang menggunakan model AI berbasis *computer vision*, pencatatan transaksi ke dalam *database*, serta penyajian informasi tabungan melalui *website monitoring*. Sementara itu, kebutuhan nonfungsional meliputi keandalan sistem agar mampu berjalan secara konsisten, keamanan data transaksi tabungan, efisiensi dalam pemrosesan gambar, serta kemudahan penggunaan baik dari sisi perangkat keras maupun perangkat lunak sehingga dapat digunakan oleh anak-anak maupun orang dewasa.

Selain mendefinisikan kebutuhan, tahap analisis juga mencakup perancangan rencana data, yaitu pengumpulan dan persiapan dataset citra uang Rupiah dalam berbagai kondisi pencahayaan dan sudut pandang yang akan digunakan untuk melatih model AI. Data yang diperoleh kemudian melalui proses anotasi, pengolahan, dan validasi agar kualitasnya terjamin. Mekanisme penjaminan mutu data juga dirancang dengan memastikan dataset yang dipakai mencakup variasi kondisi nyata, serta dilakukan pengujian untuk memverifikasi akurasi model deteksi uang. Dengan demikian, tahap analisis ini tidak hanya memetakan tujuan dan ruang lingkup penelitian, tetapi juga menyiapkan fondasi berupa kebutuhan sistem, rencana data, dan mekanisme kontrol mutu yang menjadi pijakan sebelum melangkah ke tahap desain, pengembangan, dan evaluasi sistem celengan pintar secara menyeluruh.

3.3 Perancangan

Setelah analisis kebutuhan selesai, selanjutnya adalah tahap perancangan. Perancangan pada penelitian ini akan membahas mengenai metode yang digunakan pada setiap sistemnya, serta akan menjelaskan mengenai rancangan pada sistem yang akan dibuat.

3.3.1 Metode Pengembangan

Dalam penelitian celengan pintar ini, kerangka pengembangan disesuaikan dengan karakteristik setiap komponen sistem. Proses pengembangan model deteksi nominal uang akan mengikuti *Artificial Intelligence Life Cycle* (AILC) agar tahapan seperti perumusan masalah, pengolahan data, pelatihan, hingga validasi model dapat berjalan sistematis. Untuk perangkat keras, akan digunakan metode *prototype*, karena pendekatan ini memungkinkan pembuatan rancangan awal dalam bentuk sederhana yang dapat diuji coba, dievaluasi, lalu disempurnakan hingga mendekati produk akhir. Sementara itu, perangkat lunak *website monitoring* akan dikembangkan dengan mengacu pada *Software Development Life Cycle* (SDLC) sehingga proses perencanaan, analisis kebutuhan, desain, implementasi, pengujian, hingga pemeliharaan dapat dilakukan secara lebih terstruktur.

1. *Artificial Intelligence Life Cycle* (AILC)

Pada pengembangan sistem untuk model deteksi nominal uang, akan mengikuti *tahapan AI Development Life Cycle* untuk menghasilkan model pendeteksi yang optimal. Pendekatan AILC lahir sebagai upaya untuk menutupi kelemahan metode konvensional yang umumnya tidak memperhatikan faktor risiko penggunaan, aspek etis, maupun tata kelola. Selain itu, metode tradisional tersebut sebenarnya tidak secara khusus dibuat untuk kebutuhan AI, melainkan hanya hasil adaptasi dari kerangka kerja umum (De Silva & Alahakoon, 2022). Tahapan AILC meliputi:

- *Problem Definition*

Tahap ini bertujuan menetapkan permasalahan utama yaitu bagaimana sistem dapat mendeteksi nominal uang kertas maupun koin mata uang rupiah secara otomatis dengan tingkat akurasi yang memadai. Tujuan yang ingin dicapai adalah menghasilkan model deteksi berbasis *computer vision* yang

mampu bekerja dalam kondisi pencahayaan bervariasi, serta dapat diimplementasikan pada perangkat dengan keterbatasan komputasi seperti Raspberry Pi. Definisi masalah juga mencakup kebutuhan sistem agar terintegrasi dengan modul IoT (ESP32) dan web *monitoring*, sehingga solusi yang dihasilkan tidak hanya akurat, tetapi juga fungsional dan aplikatif untuk mendukung literasi keuangan anak.

- *Data Acquisition And Preparation*

Melalui pengumpulan dan pelabelan dataset citra uang kertas maupun logam dengan variasi kondisi nyata. Untuk dataset akan digunakan uang kertas dan koin mata uang rupiah yang dimana akan dianotasi dan diklasifikasikan dalam 9000 dataset yang dibagi menjadi 9 kelas, yaitu nominal 500 dan 1000 untuk uang koin serta nominal 1rb, 2rb, 5rb, 10rb, 20rb, 50rb, dan 100rb untuk uang kertas. Dataset ini nantinya terbagi menjadi 3 subset yaitu data latih (*training*) sebesar 70%, data validasi (*validation*) sebesar 20%, dan data uji (*test*) sebesar 10%.

- *Model Development*

Pada tahap ini dilakukan pelatihan model YOLOv8 menggunakan dataset yang telah diproses sebelumnya. Proses pengembangan model tidak hanya sebatas melatih arsitektur dasar, tetapi juga menerapkan teknik augmentasi (seperti rotasi dan perubahan pencahayaan) untuk memperkaya variasi data sehingga model lebih *robust*. *Hyperparameter tuning* juga dilakukan, meliputi pengaturan *learning rate*, *batch size*, dan jumlah *epoch* untuk mendapatkan performa optimal.

- *Evaluation and Validation*

Evaluasi model dilakukan menggunakan *confusion matrix* untuk melihat distribusi kesalahan klasifikasi antar kelas nominal uang. Selain itu, dihitung pula nilai akurasi, presisi, *recall*, F1-Score, serta *mean Average Precision* (mAP) sebagai ukuran menyeluruh performa deteksi objek. Hasil evaluasi ini menjadi dasar dalam memutuskan apakah model sudah layak digunakan atau perlu dilakukan *retraining*.

- *Deployment*

Model YOLOv8 dijalankan secara efisien di perangkat Raspberry Pi. Tahap ini juga mencakup integrasi model dengan sistem IoT melalui ESP32, sehingga model tidak hanya mampu melakukan deteksi, tetapi juga mengirim hasil deteksi ke server, *website*, maupun Telegram secara *real-time*. Dengan demikian, sistem dapat bekerja langsung dalam skenario nyata.

- *Monitoring and Maintenance*

Setelah model terpasang, dilakukan pengujian langsung dengan berbagai kondisi nyata untuk memantau performanya. *Monitoring* berfokus pada kestabilan deteksi, respons waktu, serta konsistensi akurasi. Apabila ditemukan penurunan performa, maka dilakukan proses *retraining* dengan menambahkan dataset baru dari kondisi nyata yang sebelumnya belum terwakili. Tahap ini memastikan sistem dapat terus diperbarui dan menyesuaikan diri dengan variasi lingkungan, sehingga keberlanjutan performa tetap terjaga.

2. *Prototype*

Metode pengembangan yang dipilih untuk sistem perangkat keras adalah *prototype*, di mana rancangan awal sistem dibuat dalam bentuk model sederhana yang menyerupai produk akhir dengan tujuan untuk diuji coba dan dievaluasi sebelum benar-benar diproduksi secara penuh. *Prototype* dipandang sebagai representasi awal atau contoh nyata yang berfungsi sebagai acuan, sekaligus media pengujian ide serta verifikasi kebutuhan Chua & Storey, 2022). Dengan pendekatan ini, pengembang dapat memastikan desain yang dibuat sudah sesuai dengan tujuan sekaligus memiliki kualitas yang diharapkan. Tahapan pengembangan perangkat keras dengan metode *prototype* dalam penelitian ini meliputi:

- Identifikasi Kebutuhan Perangkat Keras

Menentukan komponen utama seperti ESP32 untuk pengendalian dan komunikasi IoT, Raspberry Pi untuk pemrosesan model YOLOv8, kamera digunakan untuk akuisisi citra uang, LCD sebagai media tampilan lokal, serta sensor/servo untuk mekanisme input koin dan uang kertas.

- Perancangan Awal Sistem (*Prototype I*)

Membuat rancangan awal berupa skema rangkaian elektronik, tata letak komponen, serta hubungan antar perangkat keras. Pada tahap ini dibuat bentuk sederhana dari celengan pintar agar alur kerja utama seperti deteksi uang dan pengiriman data dapat berjalan.

- Pembuatan dan Integrasi Komponen

Merakit komponen sesuai desain, menghubungkan ESP32 dengan Raspberry Pi, kamera, LCD, dan perangkat lainnya. Integrasi dilakukan secara bertahap untuk memastikan komunikasi antar perangkat berjalan baik.

- Pengujian Awal *Prototype*

Melakukan uji coba fungsi dasar, seperti kamera dalam menangkap citra, ESP32 dalam mengirimkan data ke server/*website*, serta tampilan saldo pada LCD. Hasil uji coba ini digunakan untuk mengidentifikasi kesalahan rangkaian atau kekurangan desain awal.

- Evaluasi dan Perbaikan

Mengevaluasi *prototype* berdasarkan hasil pengujian awal. Perbaikan dilakukan pada bagian yang belum optimal, misalnya posisi kamera untuk meminimalisir distorsi citra, stabilitas komunikasi WiFi, atau daya tahan komponen mekanis.

- Penyempurnaan *Prototype* (Iterasi Lanjutan)

Mengembangkan *prototype* menjadi versi yang lebih mendekati produk akhir dengan memperbaiki kualitas material, menambahkan fitur pendukung (misalnya notifikasi *real-time* melalui Telegram), serta mengoptimalkan konsumsi daya.

- Pengujian Akhir

Melakukan pengujian komprehensif mencakup uji akurasi deteksi nominal uang, serta fungsionalitas perangkat keras dalam penggunaan berulang. Dengan tahapan tersebut, metode *prototype* memungkinkan pengembangan perangkat keras celengan pintar dilakukan secara bertahap, di mana setiap iterasi berfungsi untuk memverifikasi ide, menguji rancangan, serta meningkatkan kualitas hingga tercapai desain yang optimal.

3. *Software Development Life Cycle (SDLC)*

Secara paralel, perangkat lunak *website monitoring* yang mendukung sistem ini dikembangkan berdasarkan *Software Development Life Cycle (SDLC)* agar proses pembangunan lebih terstruktur dan terkelola dengan baik (Sofyan, Puspitorini & Yulianto, 2016). Tahapan SDLC dalam konteks penelitian ini meliputi:

- *Planning*

Menentukan tujuan utama *website*, yaitu menyediakan tampilan *real-time* hasil deteksi nominal uang, rekap jumlah tabungan, histori transaksi, dan integrasi notifikasi dari sistem IoT.

- *Requirement Analysis*

Mengidentifikasi kebutuhan pengguna, baik dari sisi siswa sebagai pengguna celengan, maupun guru/orang tua sebagai pemantau tabungan, meliputi kebutuhan *database* transaksi, autentikasi pengguna, serta *dashboard* interaktif.

- *Design*

Mendesain arsitektur sistem berbasis *client-server*, perancangan basis data tabungan, antarmuka *website* yang *user-friendly*, serta alur komunikasi data dari Raspberry Pi/ESP32 menuju server web.

- *Implementation*

Mengembangkan *website* menggunakan *framework* Laravel, menghubungkannya dengan *database*, serta mengintegrasikannya dengan modul IoT untuk menampilkan hasil deteksi secara *real-time*.

- *Testing & Validation*

Melakukan uji fungsionalitas fitur (*login*, rekap saldo, histori transaksi, notifikasi), uji performa tampilan, serta uji fungsionalitas fitur.

- *Maintenance*

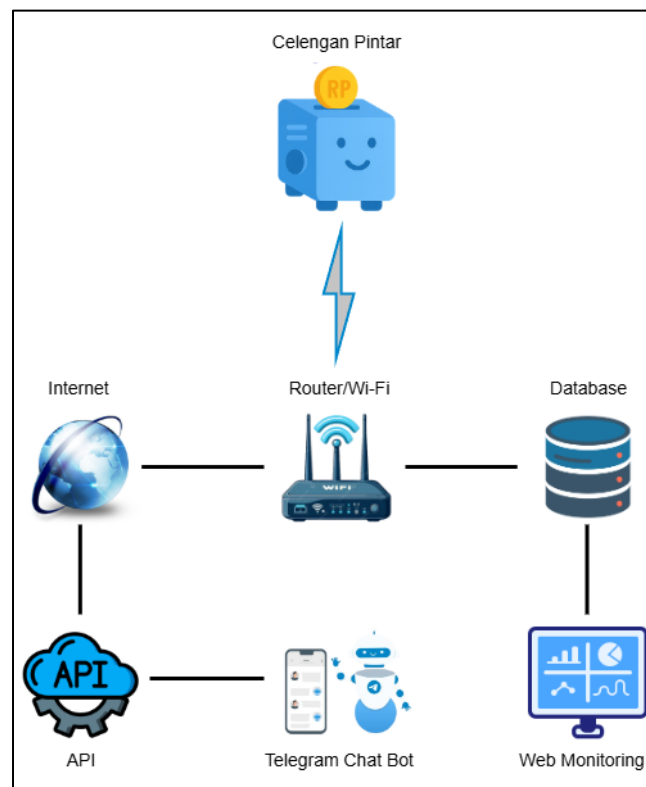
Menyediakan proses pemeliharaan berupa pembaruan sistem, perbaikan *bug*, serta pengembangan fitur tambahan seperti analisis tabungan mingguan atau integrasi dengan aplikasi *mobile* di masa depan.

3.3.2 Perancangan Sistem

Agar penelitian ini dapat dilakukan, dibutuhkan suatu perancangan untuk mendapatkan gambaran mengenai penelitian kedepannya. Maka dari itu, terdapat diagram arsitektur sistem, *activity* diagram, *block* diagram, *message flow* diagram, *use case* diagram, *wiring* diagram dan *class* diagram sistem untuk merepresentasikan komponen dari sistem yang dirancang.

1. Diagram Arsitektur Sistem

Diagram arsitektur sistem pada penelitian ini menggambarkan keterhubungan antar komponen utama celengan pintar berbasis IoT dan sistem *computer vision*. Perangkat celengan pintar terhubung melalui jaringan Wi-Fi untuk mengirimkan data ke *database* yang dapat diakses melalui web *monitoring*. Selain itu, integrasi dengan API dan Telegram *Chat Bot* memungkinkan proses notifikasi serta interaksi pengguna berjalan lebih mudah. Gambar 3.2 menunjukkan rancangan arsitektur sistem tersebut secara menyeluruh.



Gambar 3. 2 Diagram Arsitektur Sistem Celengan Pintar

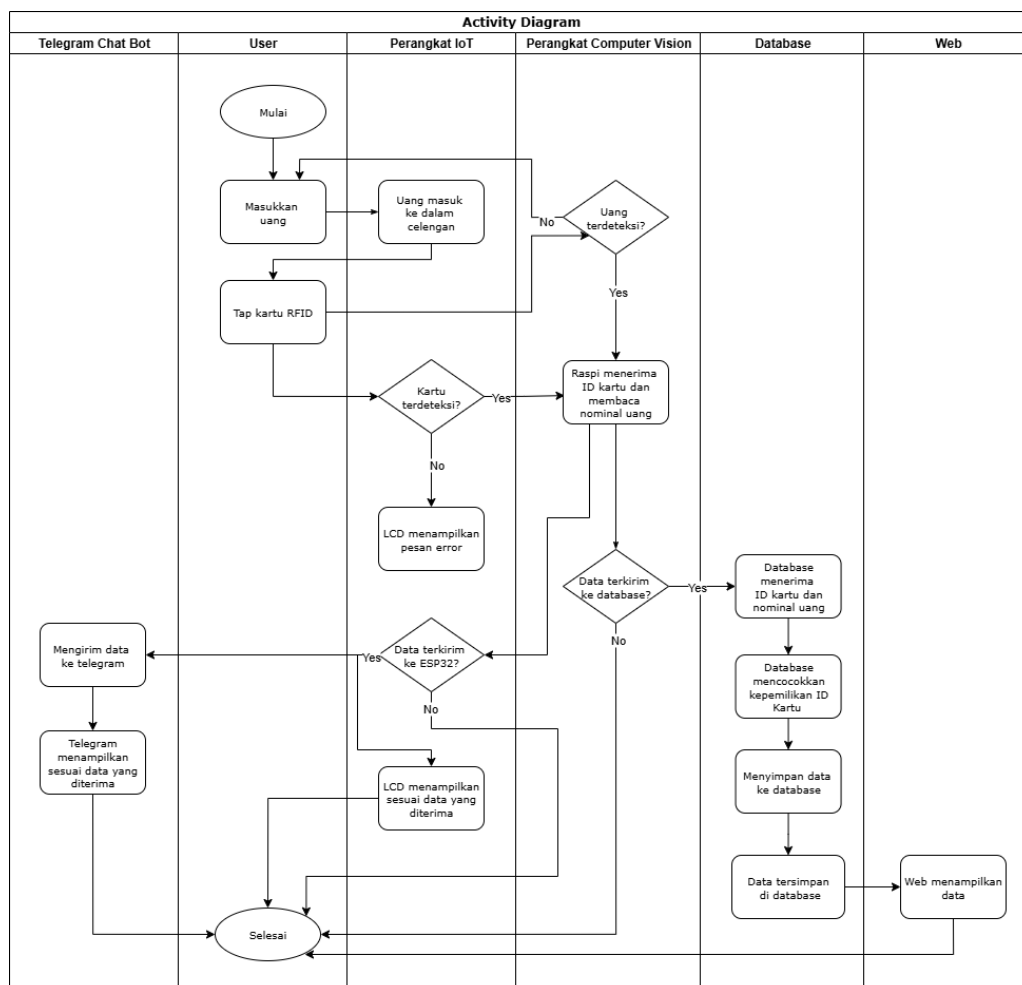
Pada Gambar 3.2, diagram arsitektur sistem celengan pintar dirancang untuk bekerja secara terintegrasi dengan menghubungkan perangkat keras dan perangkat lunak. Sistem ini menggunakan koneksi *router* atau Wi-Fi sebagai penghubung utama antara perangkat celengan pintar dengan berbagai komponen eksternal. Celengan pintar bertindak sebagai inti sistem yang menerima data dari sensor dan modul pengendali, kemudian mengolahnya untuk dikirimkan lebih lanjut.

Data transaksi yang dihasilkan oleh celengan pintar akan tersimpan pada *database*. *Database* ini berfungsi sebagai pusat penyimpanan informasi terkait nominal tabungan, identitas pengguna, serta riwayat transaksi. Informasi yang tersimpan kemudian dapat diakses melalui web *monitoring system*, yang menampilkan data secara *real-time* dalam bentuk grafik maupun laporan yang lebih mudah dipahami pengguna maupun admin.

Selain itu, celengan pintar juga terhubung dengan internet yang menjadi jalur komunikasi ke API. API ini berperan sebagai jembatan antara perangkat dengan layanan eksternal, seperti Telegram *Chat Bot*. Melalui Telegram Bot, pengguna dapat memperoleh notifikasi otomatis, status tabungan, serta informasi terbaru terkait aktivitas sistem. Dengan integrasi ini, sistem Celengan Pintar mampu memberikan layanan yang tidak hanya terbatas pada perangkat lokal, tetapi juga mendukung *monitoring* jarak jauh secara efisien.

2. *Activity Diagram* Sistem Celengan Pintar

Activity diagram ini menjelaskan alur interaksi pengguna dengan sistem mulai dari memasukkan uang dan autentikasi kartu *RFID*, hingga proses pendeteksian nominal oleh perangkat *computer vision*. Data yang diperoleh kemudian diverifikasi serta disimpan pada *database*, dan hasilnya ditampilkan melalui *web* maupun dikirimkan ke *Telegram bot*. Gambar 3.3 memperlihatkan alur proses tersebut secara ringkas dan terstruktur.



Gambar 3. 3 Activity Diagram Sistem

Pada Gambar 3.3, *activity diagram* menjelaskan alur kerja sistem Celengan Pintar yang terdiri dari interaksi antara pengguna, perangkat *IoT*, perangkat *computer vision*, *database*, hingga *web monitoring*. Proses diawali dengan aksi pengguna yang memasukkan uang ke dalam celengan serta menempelkan kartu *RFID* sebagai identitas. Kedua aktivitas ini menjadi langkah awal untuk memastikan bahwa sistem dapat mengenali siapa pemilik tabungan dan nominal uang yang dimasukkan. Setelah itu, perangkat *IoT* akan memproses masukan tersebut dan melakukan pengecekan apakah kartu *RFID* serta uang berhasil terdeteksi.

Jika sistem tidak dapat mengenali kartu maupun uang, maka *LCD* akan menampilkan pesan *error* sebagai bentuk notifikasi kepada pengguna bahwa transaksi gagal diproses. Sebaliknya, apabila kartu dan uang berhasil

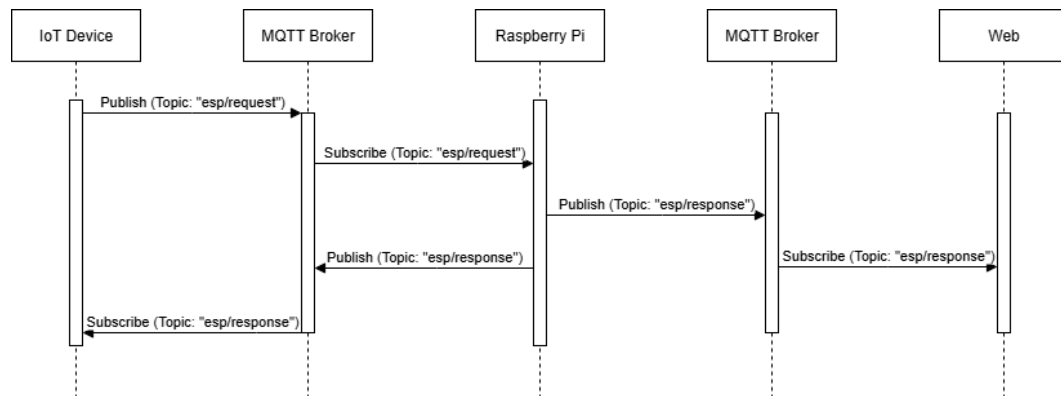
teridentifikasi, *Raspberry Pi* yang berperan sebagai perangkat *computer vision* akan menerima data ID kartu serta melakukan pembacaan nominal uang melalui pemrosesan citra. Proses ini mencakup pengambilan gambar, identifikasi jenis uang, hingga menghasilkan *output* berupa informasi nilai nominal yang siap diteruskan ke tahap berikutnya.

Selanjutnya, hasil deteksi tersebut dikirimkan menuju *database* untuk disimpan. Pada tahap ini, *database* tidak hanya berfungsi sebagai tempat penyimpanan, tetapi juga melakukan proses pencocokan *ID* kartu dengan data pemilik yang sudah ada sebelumnya. Dengan begitu, sistem dapat memastikan bahwa uang yang ditabung benar-benar tercatat pada akun yang sesuai. Setelah validasi berhasil dilakukan, *database* akan menyimpan seluruh informasi transaksi, meliputi identitas pengguna, nominal uang, serta waktu penyimpanan.

Data yang sudah tersimpan di dalam *database* kemudian dapat diakses oleh dua media utama. Pertama, *LCD* pada perangkat *IoT* akan menampilkan data yang telah tersimpan sehingga pengguna dapat langsung melihat informasi tabungan mereka secara *real-time* tanpa harus membuka aplikasi lain. Kedua, data juga dapat diakses melalui *web monitoring* yang terhubung dengan *database*. *Web monitoring* ini menampilkan data tabungan, riwayat transaksi, dan status sistem secara lebih detail, sehingga memudahkan pengguna maupun admin dalam melakukan pemantauan dari jarak jauh. Dengan adanya alur ini, sistem Celengan Pintar tidak hanya berfungsi sebagai wadah menabung, tetapi juga menghadirkan transparansi, keamanan, dan kemudahan akses informasi.

3. *Message Flow Diagram* Sistem Celengan Pintar

Pada penelitian ini, *message flow diagram* digunakan untuk menggambarkan alur pertukaran pesan antar komponen sistem melalui protokol MQTT. Diagram ini menjelaskan bagaimana *IoT device* mengirimkan data ke MQTT Broker untuk kemudian diteruskan kepada *Raspberry Pi* yang berperan dalam pemrosesan data. Setelah diproses, *Raspberry Pi* mengirimkan hasil kembali melalui broker dengan topik tanggapan yang dapat diterima kembali oleh perangkat *IoT* maupun aplikasi web. Gambar 3.4 merupakan *message flow diagram* yang memperlihatkan proses komunikasi tersebut.



Gambar 3. 4 *Message Flow Diagram* Sistem Celengan Pintar

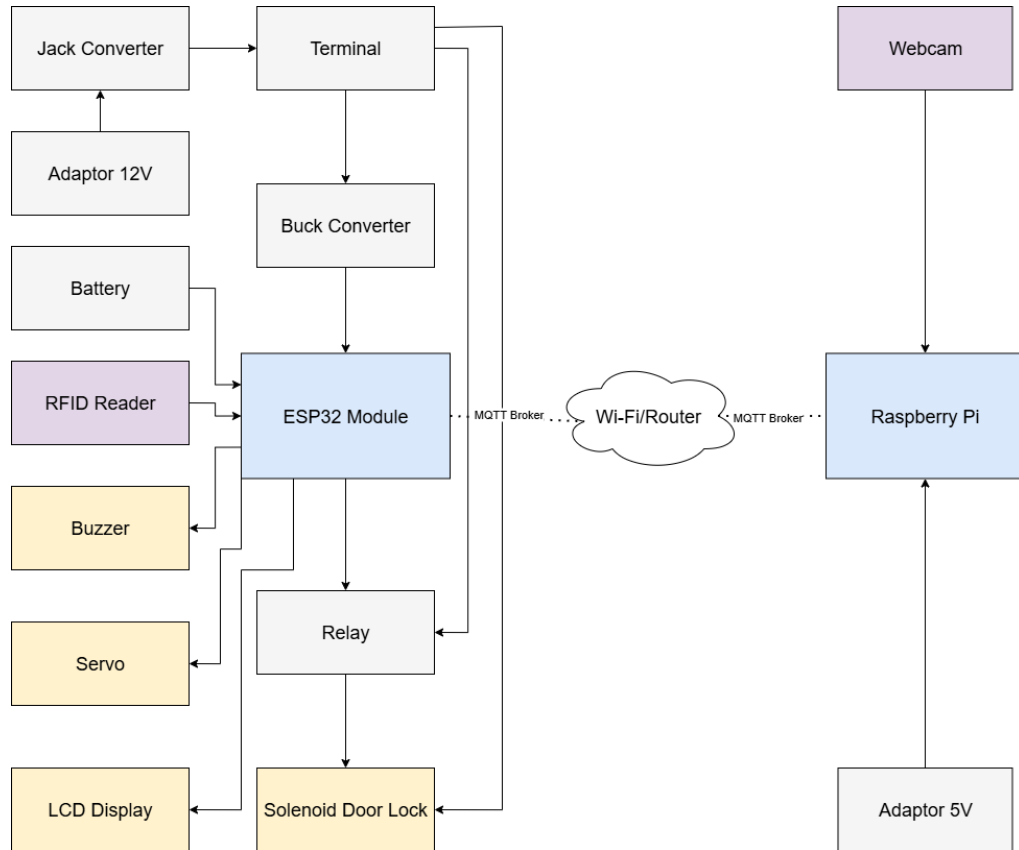
Message flow diagram pada Gambar 3.4 menggambarkan alur komunikasi data antar komponen sistem celengan pintar berbasis IoT yang memanfaatkan protokol MQTT sebagai penghubung utama. Alur komunikasi ini diawali dari *IoT Device* yang mengirimkan pesan dengan topik *esp/request* kepada MQTT Broker. Pesan tersebut kemudian diteruskan kepada Raspberry Pi yang berperan sebagai pengolah data sekaligus pusat integrasi dari keseluruhan sistem. Raspberry Pi akan memproses data yang diterima sesuai dengan logika program yang telah dirancang, kemudian menghasilkan keluaran atau tanggapan yang dipublikasikan kembali melalui MQTT Broker dengan topik *esp/response*.

Setelah dipublikasikan, pesan balasan ini dapat diakses kembali oleh *IoT Device* maupun sistem web yang berfungsi sebagai antarmuka pengguna. Pada tahap ini, web menerima data hasil pengolahan melalui MQTT Broker sehingga informasi yang ditampilkan kepada pengguna selalu mutakhir dan sesuai dengan kondisi sistem sebenarnya. Dengan mekanisme ini, setiap komponen memiliki peran yang jelas namun tetap saling bergantung, di mana *IoT Device* menjadi pengirim data awal, Raspberry Pi sebagai pemroses inti, MQTT Broker sebagai perantara komunikasi, dan web sebagai penyaji informasi kepada pengguna.

4. *Block Diagram* Sistem Celengan Pintar

Block diagram digunakan untuk memperlihatkan hubungan antar komponen utama dalam sistem celengan pintar. Diagram ini menjelaskan bagaimana setiap perangkat keras, seperti ESP32, Raspberry Pi, RFID Reader, dan modul lainnya, saling terhubung serta berkoordinasi melalui jaringan Wi-Fi

dengan bantuan protokol MQTT. Berikut detail penjelasan mengenai *block diagram* pada gambar 3.5.



Gambar 3. 5 *Block Diagram* Sistem Celengan Pintar

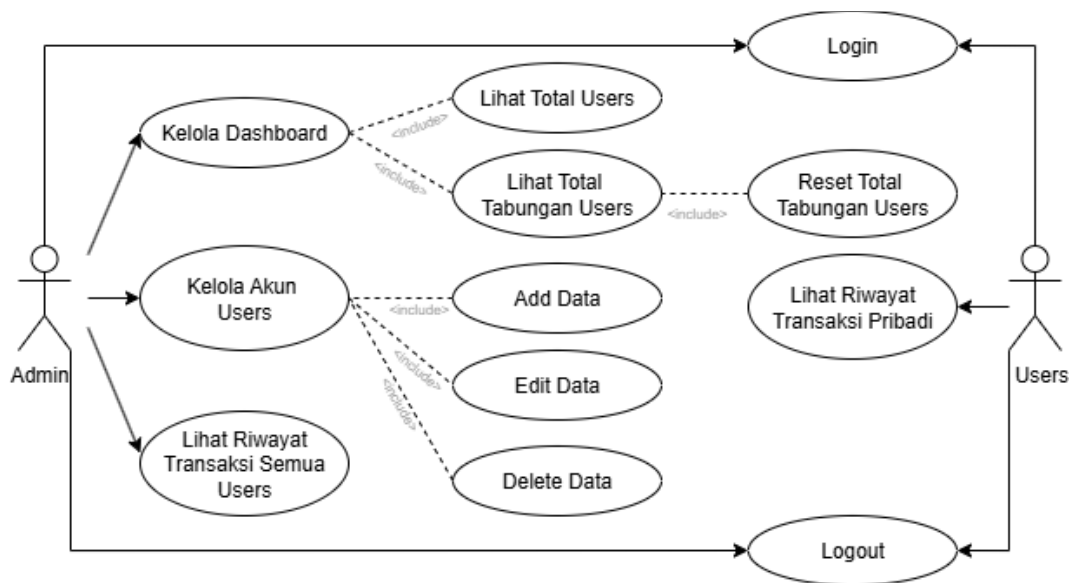
Pada Gambar 3.5, diagram ini menggambarkan komponen-komponen perangkat keras utama dan alur data dalam sistem Celengan Pintar. Inti dari sistem adalah Raspberry Pi yang menjalankan algoritme *computer vision* YOLOv8 untuk menganalisis dan mengidentifikasi nominal uang yang dimasukkan. Proses dimulai dengan *webcam* yang menangkap gambar uang. Gambar ini dikirim ke Raspberry Pi untuk deteksi objek dan pengenalan. Raspberry Pi kemudian mengirimkan data transaksi *database*.

Database berfungsi menyimpan riwayat transaksi dan memungkinkan pemantauan jarak jauh melalui *web server* dan *bot* Telegram. Selain komponen *computer vision*, sistem juga mencakup pembaca RFID untuk autentikasi pengguna, *solenoid door lock* untuk mengamankan celengan, dan layar LCD untuk menampilkan informasi. Daya untuk seluruh sistem disuplai melalui

power supply 12V DC, tetapi digunakan *step down* 5V DC untuk menghubungkan *power* ke *ESP32 module*. Integrasi komponen-komponen perangkat keras ini memungkinkan Celengan Pintar menyediakan solusi manajemen tabungan yang otomatis, aman, dan dapat diakses jarak jauh.

5. Use Case Diagram Sistem Celengan Pintar

Use case diagram pada penelitian ini menggambarkan interaksi antara *Admin* dan *Users* dengan sistem Celengan Pintar. *Admin* dapat mengelola dashboard, akun pengguna, serta memantau transaksi, sedangkan *Users* memiliki akses untuk login, melihat riwayat transaksi pribadi, dan logout. Diagram ini membantu memperjelas peran serta fungsi yang dijalankan oleh masing-masing aktor dalam sistem. Untuk lebih detailnya dapat dilihat pada gambar 3.6.



Gambar 3. 6 Use Case Diagram Sistem Celengan Pintar

Pada Gambar 3.6, ditunjukkan *Use Case Diagram* yang menggambarkan interaksi antara aktor (*Admin* dan *Users*) dengan sistem Celengan Pintar berbasis IoT. Diagram ini menjelaskan fungsi-fungsi utama yang dapat dilakukan oleh masing-masing aktor, baik admin ataupun *users* sesuai dengan hak aksesnya. *Admin* memiliki hak akses penuh terhadap sistem. Pertama, *Admin* dapat melakukan *login* ke dalam sistem untuk masuk ke *dashboard*. Pada menu *Kelola Dashboard*, *admin* bisa melihat informasi umum berupa jumlah total pengguna

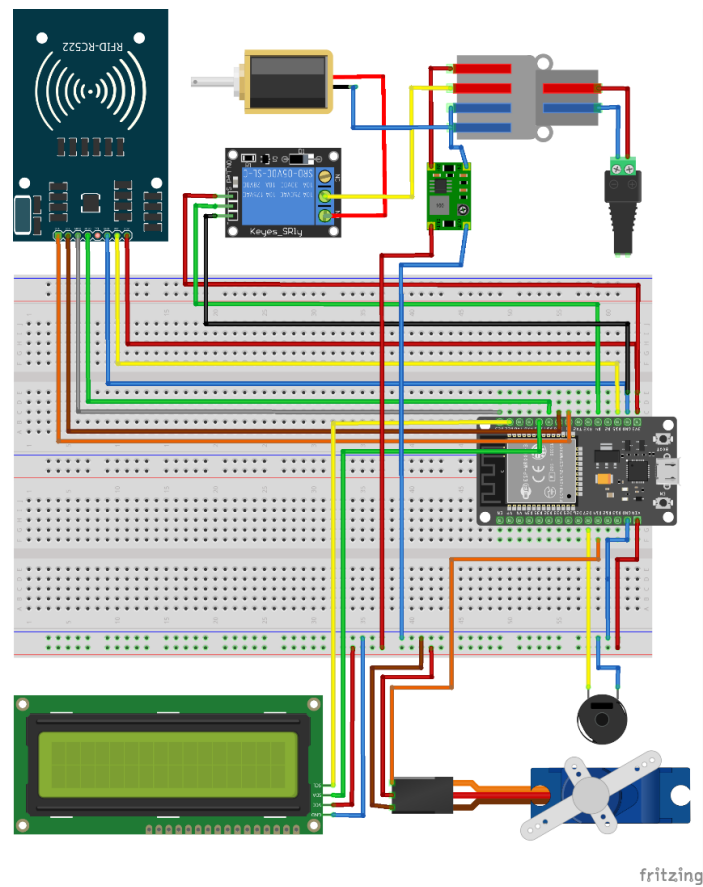
dan jumlah total tabungan yang tersimpan. Selain itu, *admin* juga diberikan hak untuk melakukan *reset* total tabungan *users* jika diperlukan, misalnya untuk kepentingan pengaturan ulang sistem atau perbaikan data. *Admin* juga dapat mengelola akun pengguna melalui fitur Kelola Akun *Users* yang mencakup penambahan data pengguna baru (*Add Data*), pengubahan data pengguna (*Edit Data*), dan penghapusan data pengguna (*Delete Data*). Dengan demikian, *admin* berperan sebagai pengendali penuh terhadap manajemen akun dalam sistem. Selain itu, *admin* juga memiliki wewenang untuk memantau seluruh aktivitas keuangan pengguna melalui fitur Lihat Riwayat Transaksi Semua *Users*, sehingga dapat memastikan data transaksi tersimpan dengan benar.

Sementara itu, aktor *Users* memiliki hak akses yang lebih terbatas. Setelah melakukan *login*, pengguna hanya dapat melihat riwayat transaksi pribadi yang berkaitan dengan akun mereka sendiri. Hal ini bertujuan untuk menjaga privasi masing-masing pengguna sekaligus memudahkan mereka dalam melakukan *monitoring* terhadap tabungan yang sudah disetorkan ke dalam celengan.

Baik *admin* maupun *user*, keduanya memiliki akses untuk melakukan *login* dan *logout* sebagai mekanisme autentikasi dan pengamanan sistem. Dengan demikian, *Use Case Diagram* ini memperlihatkan adanya pembagian peran yang jelas antara *admin* dan pengguna biasa. *Admin* berperan dalam manajemen data dan pengawasan keseluruhan sistem, sedangkan *user* hanya berfokus pada pemantauan riwayat transaksi pribadi.

6. *Wiring Diagram* Sistem Celengan Pintar

Pada penelitian ini, rangkaian *hardware* digunakan untuk menunjukkan hubungan antar komponen sistem Celengan Pintar berbasis IoT. Rangkaian ini menghubungkan modul RFID, motor, relay, LCD, buzzer, dan *servo* dengan mikrokontroler ESP32 melalui *breadboard*. Gambar 3.7 memperlihatkan susunan rangkaian tersebut secara lebih terstruktur.



Gambar 3. 7 *Wiring Diagram* Sistem Celengan Pintar

Pada Gambar 3.7, *wiring diagram* dari sistem Celengan Pintar berbasis IoT dengan teknologi *computer vision*. Rangkaian ini terdiri dari beberapa komponen utama, yaitu ESP32 sebagai mikrokontroler, modul RFID RC522, LCD I2C 16x2, servo motor, *solenoid door lock*, buzzer, dan relay. Setiap komponen dihubungkan sesuai dengan kebutuhan fungsi masing-masing agar sistem dapat berjalan dengan baik.

Pada bagian input, modul RFID RC522 terhubung ke ESP32 melalui komunikasi *SPI (Serial Peripheral Interface)* yang terdiri dari pin *MOSI*, *MISO*, *SCK*, dan *SDA*. Modul RFID ini berfungsi untuk membaca kartu RFID sebagai identitas pengguna. Selanjutnya, sistem juga dilengkapi dengan modul kamera *computer vision* (pada Raspberry Pi) yang bertugas untuk mendeteksi nominal uang, meskipun pada *wiring diagram* ini fokusnya masih pada rangkaian mikrokontroler ESP32.

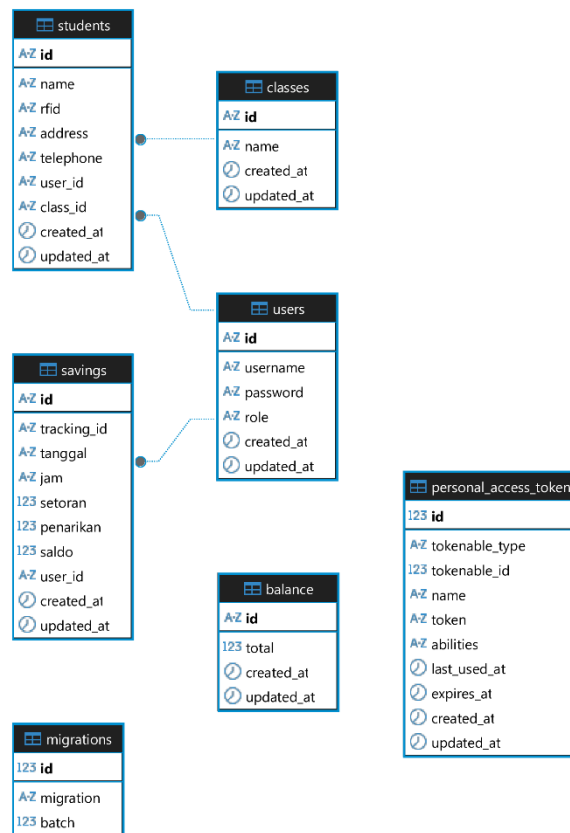
Untuk bagian output, LCD I2C 16x2 terhubung ke ESP32 melalui jalur *SDA* dan *SCL* sehingga dapat menampilkan informasi seperti ID kartu, nominal uang yang terdeteksi, maupun status penyimpanan data. Kemudian, terdapat *servo motor* yang dikontrol langsung oleh ESP32 dan digunakan sebagai mekanisme pengatur celah masuknya uang ke celengan. Selain itu, terdapat juga *buzzer* sebagai indikator suara ketika terjadi aksi tertentu, misalnya berhasil melakukan penyimpanan data atau terjadi *error*.

Bagian aktuator lain adalah *motor DC* yang berfungsi sebagai penggerak tambahan, dikendalikan menggunakan modul *relay* serta *driver motor* agar tegangan yang dibutuhkan tetap stabil. Motor ini mendapatkan sumber daya dari adaptor eksternal yang dihubungkan melalui terminal *power*, sedangkan ESP32 tetap diberi suplai dari jalur 5V dan GND yang terdistribusi di *breadboard*.

Secara keseluruhan, *wiring diagram* ini menggambarkan integrasi antara sensor input (RFID dan kamera), pengendali utama (ESP32), serta aktuator output (*servo*, *motor DC*, *buzzer*, dan LCD). Hubungan antar komponen dibuat melalui *breadboard* untuk mempermudah perakitan serta pengujian sebelum sistem dipasang secara permanen.

7. Entity Relationship Diagram Sistem Celengan Pintar

Pada penelitian ini, *Entity Relationship Diagram (ERD)* digunakan untuk menjelaskan rancangan basis data yang mengatur hubungan antar entitas dalam sistem Celengan Pintar. Diagram ini menampilkan keterkaitan tabel seperti *students*, *classes*, *users*, *savings*, *balance*, serta tabel pendukung lainnya agar struktur data lebih jelas dan terorganisir. Gambar 3.8 merupakan ERD yang menjadi acuan dalam pengelolaan *database* sistem.



Gambar 3. 8 *Entity Relationship Diagram* Sistem Celengan Pintar

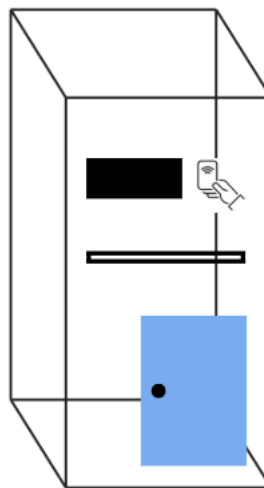
Pada Gambar 3.8, *Entity Relationship Diagram (ERD)* disusun untuk menggambarkan rancangan basis data yang digunakan dalam sistem Celengan Pintar berbasis IoT. Diagram ini berfungsi sebagai representasi visual yang memperlihatkan bagaimana setiap entitas saling berhubungan, sehingga mempermudah pemahaman mengenai struktur data yang dikelola oleh sistem. Dalam ERD ini, entitas utama yang ditampilkan antara lain *students*, *classes*, *users*, *savings*, dan *balance*, serta beberapa tabel tambahan seperti *personal_access_tokens* dan *migrations* yang berperan dalam mendukung kelancaran pengelolaan data.

Entitas *students* menyimpan informasi dasar mengenai siswa seperti nama, alamat, nomor telepon, dan identitas RFID yang berfungsi sebagai penanda unik saat melakukan interaksi dengan perangkat. Tabel ini dihubungkan dengan entitas *classes* untuk mengelompokkan siswa berdasarkan kelas, serta dengan

entitas *users* yang mengatur hak akses dan kredensial pengguna sistem. Sementara itu, entitas *savings* digunakan untuk mencatat setiap transaksi tabungan, baik berupa setoran maupun penarikan, yang selanjutnya berhubungan dengan entitas *balance* guna memperbarui total saldo yang dimiliki masing-masing pengguna.

8. Skema Perancangan Perangkat Keras

Untuk skema perancangan perangkat keras, serta komponen-komponen utama yang terdapat pada tampilan fisik celengan pintar, dapat dilihat pada Gambar 3.9 berikut.



Gambar 3. 9 Perancangan Sistem Perangkat Keras

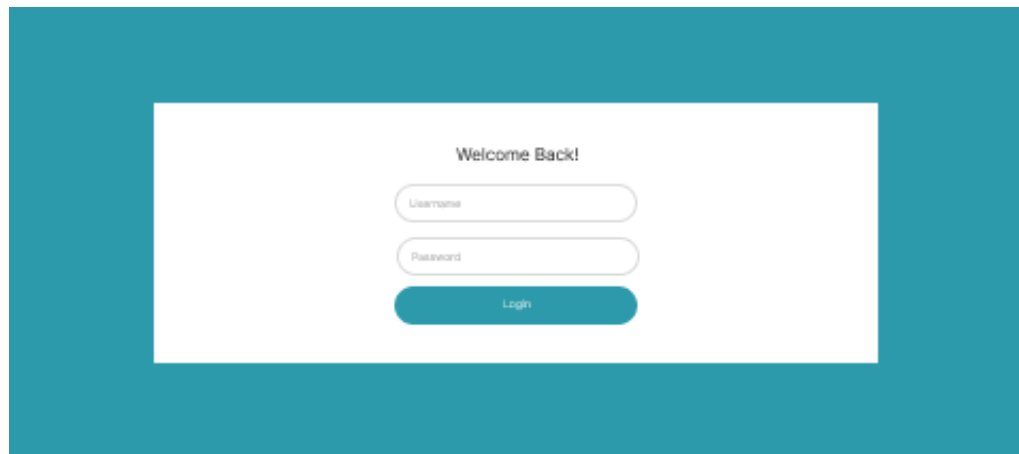
Pada Gambar 3.9, merupakan rancangan sistem perangkat keras yang akan dibuat, alat tersebut nanti akan dibuat dengan bahan yang kokoh dan terdiri dari beberapa komponen seperti LCD, RFID, pintu untuk membuka celengan dan lubang untuk memasukkan uang.

9. Skema Perancangan Perangkat Lunak

Skema perancangan perangkat lunak berbasis web akan dilakukan menggunakan *framework* Laravel dengan bahasa pemrograman Python. Sistem web ini dirancang sebagai antarmuka yang memungkinkan admin untuk mendaftarkan data siswa, melihat riwayat transaksi, serta untuk siswa dapat melihat riwayat transaksi dirinya sendiri secara *real-time*.

a. Halaman *Login*

Sebelum dapat menggunakan sistem, pengguna diarahkan terlebih dahulu ke halaman *login*. Tampilan halaman *login* ditunjukkan pada Gambar 3.10.

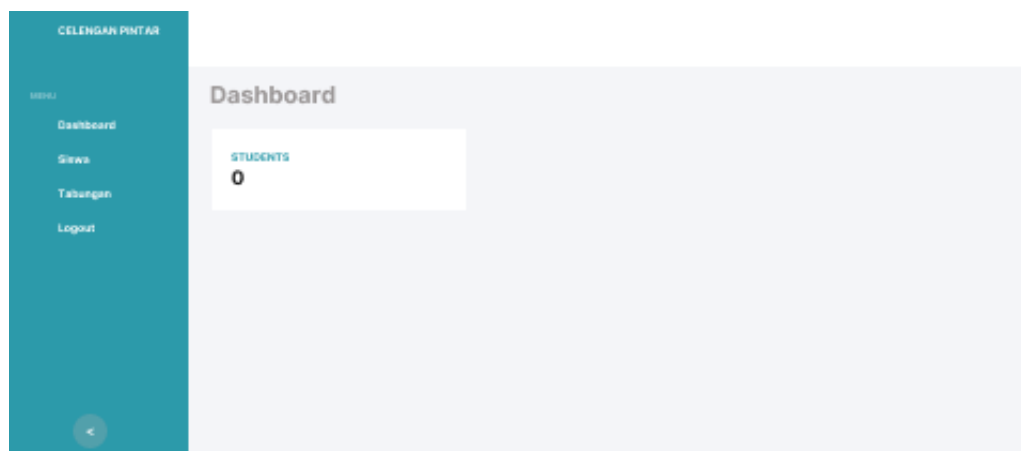


Gambar 3. 10 Perancangan Halaman *Login* pada *Website*

Pada Gambar 3.10, halaman ini berfungsi sebagai pintu masuk utama sistem sekaligus sebagai langkah keamanan. Pengguna harus memasukkan *username* dan *password* yang valid. Data login tersebut akan diperiksa ke dalam *database* untuk memastikan hanya akun yang terdaftar yang dapat mengakses sistem.

b. Halaman *Dashboard* Admin

Setelah login berhasil, pengguna akan diarahkan ke halaman *dashboard*. Tampilan halaman *dashboard* dapat dilihat pada Gambar 3.11.



Gambar 3. 11 Perancangan Halaman *Dashboard* Admin pada *Website*

Dashboard ini menampilkan ringkasan informasi penting terkait sistem,

seperti jumlah siswa yang terdaftar. Tampilan yang sederhana dan ringkas memudahkan admin untuk melihat kondisi sistem secara keseluruhan.

c. Halaman Daftar Siswa

Halaman daftar siswa berfungsi untuk menampilkan dan mengelola data siswa. Tampilan halaman ini ditunjukkan pada Gambar 3.12.



Gambar 3. 12 Perancangan Halaman Daftar Siswa pada *Website*

Informasi yang ditampilkan dalam tabel meliputi ID, *username*, nama, RFID, alamat, kelas, dan nomor telepon siswa. Setiap baris tabel dilengkapi dengan menu aksi berupa tombol edit dan hapus, sehingga data dapat diperbarui atau dihapus sesuai kebutuhan. Tersedia juga tombol “Tambah Siswa” untuk menambahkan data siswa baru

d. Halaman Input Data Siswa

Form input data siswa ini dapat diisi untuk mendaftarkan data baru, data baru berisi identitas siswa sebagai berikut yang ditampilkan pada Gambar 3.13.

Gambar 3. 13 Perancangan Halaman Input Data Siswa pada *Website*

Halaman Input Data Siswa pada *website* celengan pintar *form* ini digunakan untuk menambahkan siswa baru ke dalam sistem. Beberapa *field* yang harus diisi di antaranya *username*, *password*, nama, RFID, alamat, kelas, dan nomor telepon. Terdapat tombol Save untuk menyimpan data dan tombol *abort* untuk membatalkan proses input.

e. Halaman Daftar Tabungan

Halaman daftar tabungan khusus untuk tampilan pada *section* siswa ditampilkan pada Gambar 3.14.



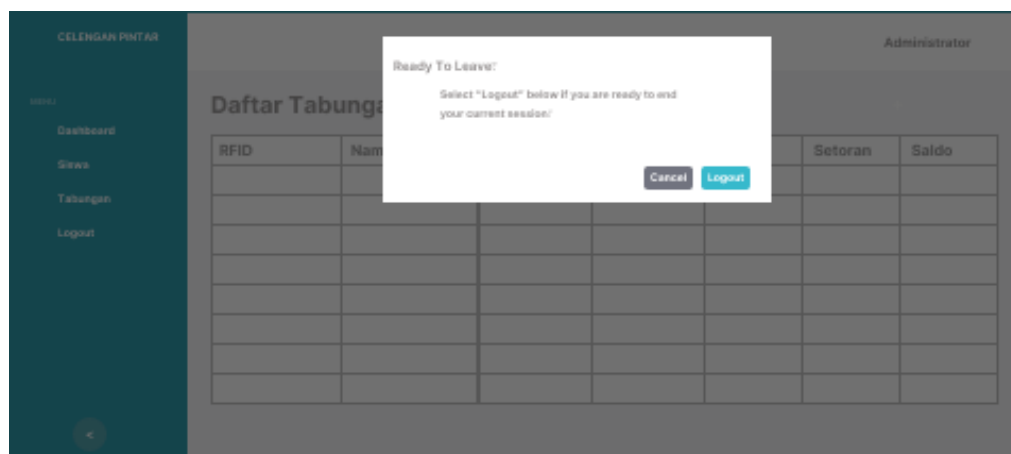
RFID	Nama	Kelas	Tanggal	Jam	Setoran	Saldo

Gambar 3. 14 Perancangan Halaman Daftar Tabungan Siswa pada *Website*

Berbeda dengan admin, halaman ini hanya menampilkan data transaksi tabungan milik siswa yang sedang login. Data yang ditampilkan berupa RFID, nama, kelas, tanggal dan jam transaksi, nominal setoran, serta saldo akhir. Dengan demikian, siswa dapat memantau perkembangan tabungan pribadinya secara mandiri.

f. Halaman Logout

Sebelum keluar dari sistem, admin akan diarahkan ke *popup* konfirmasi seperti yang terlihat pada Gambar 3.15.



Gambar 3. 15 Perancangan Halaman *Logout* pada *Website*

Gambar 3.15 Halaman *Logout* pada *website* celengan pintar *popup* ini memberikan pilihan kepada pengguna untuk benar-benar keluar atau membatalkan proses *logout*. Tombol *logout* berfungsi mengakhiri sesi, sedangkan tombol *cancel* digunakan untuk tetap berada di dalam sistem.

3.4 Pengembangan

Pada tahap pengembangan, rancangan sistem yang telah disusun mulai diwujudkan dalam bentuk implementasi nyata untuk melihat tingkat akurasi sekaligus efektivitasnya. Untuk mewujudkan hal tersebut, diperlukan beberapa perencanaan, mencakup perencanaan teknis baik perangkat keras maupun perangkat lunak yang akan digunakan dalam implementasi celengan pintar berbasis *Internet of Things (IoT)* dan *computer vision*. Perangkat keras berfungsi sebagai media fisik yang menjalankan fungsi input, proses, dan output, sedangkan perangkat lunak bertugas mengelola komunikasi, penyimpanan, serta analisis data termasuk pengolahan citra menggunakan *Artificial Intelligence*.

1. Rencana Perangkat Model yang akan Digunakan

Bagian penting dari penelitian ini adalah pengembangan model *computer vision* untuk mendeteksi nominal uang Rupiah. Untuk itu digunakan serangkaian perangkat yang mendukung pengolahan data, pelatihan model, dan implementasi AI. Rincian perangkat model ditampilkan pada Tabel 3.1.

Tabel 3. 1 Rencana Perangkat Model yang Akan Digunakan

No	Nama	Fungsi
1	Pyhton	Bahasa pemrograman utama untuk pengembangan dan pelatihan model <i>computer vision</i> .
2	Google Colaboratory	Lingkungan komputasi berbasis <i>cloud</i> untuk melatih model AI dengan dukungan GPU.
3	Roboflow Universe	Platform anotasi dataset dan <i>preprocessing</i> gambar untuk mempersiapkan data pelatihan AI.
4	OpenCV	Pustaka pemrosesan citra yang digunakan untuk membaca, mengolah, dan menganalisis gambar uang.
5	Jupyter Notebook	Media eksperimen interaktif dalam pengembangan model AI dan dokumentasi proses pelatihan.

2. Rencana Perangkat Keras yang Akan Digunakan

Komponen perangkat keras dipilih berdasarkan kebutuhan sistem untuk mendukung fungsi utama celengan pintar, yaitu identifikasi pengguna, deteksi nominal uang, pengendalian mekanisme celengan, dan pemberian umpan balik. Rincian perangkat keras yang digunakan ditampilkan pada Tabel 3.2.

Tabel 3. 2 Rencana Perangkat Keras yang Akan Digunakan

No	Nama	Fungsi
1	Raspberry Pi 5	<i>Single-board computer</i> yang berfungsi sebagai pusat komputasi untuk pemrosesan citra, menjalankan model AI, dan integrasi dengan database serta website.
2	ESP32	<i>Microcontroller</i> dengan konektivitas Wi-Fi dan Bluetooth, digunakan untuk mengendalikan

No	Nama	Fungsi
		perangkat keras (servo, relay, solenoid, buzzer, dan LCD) serta komunikasi data dengan Raspberry Pi melalui protokol MQTT.
3	RFID RC522	Modul pembaca RFID untuk autentikasi pengguna menggunakan kartu/tag.
4	LCD 12C16x2	Layar sederhana untuk menampilkan informasi seperti status sistem, nominal uang yang terdeteksi, dan notifikasi singkat.
5	Motor Servo	Menggerakkan mekanisme slot celengan agar uang dapat masuk setelah proses validasi.
6	<i>Solenoid Door Lock</i>	Kunci elektronik yang memastikan celengan tetap aman dan hanya dapat dibuka oleh pengguna yang terotorisasi.
7	Buzzer	Memberikan notifikasi suara sebagai indikator keberhasilan, peringatan, atau kesalahan.
8	Relay	Modul kendali arus untuk mengaktifkan perangkat bertegangan lebih tinggi seperti solenoid.
9	Kamera USB	Menangkap citra uang yang dimasukkan untuk proses deteksi nominal berbasis <i>computer vision</i> .

3. Rencana Perangkat Lunak yang Akan Digunakan

Selain perangkat keras, penelitian ini juga menggunakan perangkat lunak yang mendukung pengembangan sistem IoT serta aplikasi berbasis web. Perangkat lunak ini berfungsi untuk pengelolaan basis data, pengembangan antarmuka, serta integrasi antara perangkat keras dan server. Rincian perangkat lunak yang digunakan ditampilkan pada Tabel 3.3.

Tabel 3. 3 Rencana Perangkat Lunak yang Akan Digunakan

No	Nama	Fungsi
1	Laravel Framework	<i>Framework</i> berbasis PHP untuk membangun website monitoring celengan, termasuk tampilan data tabungan dan riwayat transaksi.
2	Visual Studio Code	Editor kode yang digunakan untuk menulis dan mengembangkan program, baik untuk ESP32, Raspberry Pi, maupun aplikasi web.
3	Laragon	Lingkungan pengembangan lokal untuk menjalankan <i>web server</i> Laravel secara efisien.
4	DBeaver	Alat manajemen basis data yang digunakan untuk mengelola dan memvisualisasikan data tabungan dalam database.
5	MySQL	Sistem manajemen basis data untuk menyimpan informasi transaksi tabungan, data pengguna, serta hasil deteksi nominal uang.
6	Mosquitto MQTT Broker	Protokol komunikasi ringan yang digunakan untuk pertukaran data antara ESP32, Raspberry Pi, dan <i>Website</i> .

3.5 Evaluasi

Setelah pengembangan selesai, sistem yang telah dibangun akan diuji secara menyeluruh. Pengujian ini mencakup berbagai kondisi seperti pencahayaan yang berbeda dan berbagai jenis halangan. Evaluasi akan dilakukan untuk mengukur akurasi deteksi nominal uang serta efektivitas teknologi *computer vision* dalam deteksi nominal uang. Hasil dari evaluasi pengujian akan digunakan untuk menilai keandalan dan efektivitas keseluruhan sistem, serta mengidentifikasi area yang perlu diperbaiki. Dengan mengikuti tahap-tahap ini, penelitian dapat mengembangkan dan mengevaluasi solusi yang komprehensif untuk masalah akurasi dalam deteksi nominal uang dalam proses deteksi.

Selanjutnya untuk pengujian pada penelitian ini ditunjukkan dengan adanya beberapa pengujian untuk semua hasil perancangan sistem. Pengujian keseluruhan sistem berfokus pada fungsionalitas sistem tanpa melihat internal atau kode program sistem tersebut. Tabel 3.4 menjelaskan format untuk nilai akurasi dari setiap kelas.

Tabel 3. 4 Format Tabel Nilai Akurasi dari Setiap Kelas Klasifikasi

Evaluasi	Kelas	100rb	50rb	20rb	10rb	5rb	2rb	1rb	1000	500
<i>Precision</i>										
<i>Recall</i>										
mAP50										
mAP50-95										

Sebelum melakukan pengujian model, terlebih dahulu dilakukan pendataan mengenai nilai akurasi dari setiap kelas yang terdapat dalam dataset. Setelah mendapatkan nilai akurasi, maka selanjutnya dapat dilakukan pengujian model yang ditampilkan pada Tabel 3.5.

Tabel 3. 5 Format Pengujian Model

Intensitas Cahaya	Uang	Hasil yang Diharapkan	Hasil Pengujian	Dokumentasi
25 lux	100rb	Sistem berhasil mendeteksi nominal Rp100.000		
	50rb	Sistem berhasil mendeteksi nominal Rp50.000		
	20rb	Sistem berhasil mendeteksi nominal Rp20.000		
	10rb	Sistem berhasil mendeteksi nominal Rp10.000		
	5rb	Sistem berhasil mendeteksi		

Intensitas Cahaya	Uang	Hasil yang Diharapkan	Hasil Pengujian	Dokumentasi
		nominal Rp5.000		
	2rb	Sistem berhasil mendeteksi nominal Rp2.000		
	1rb	Sistem berhasil mendeteksi nominal Rp1.000		
	1000	Sistem berhasil mendeteksi nominal Rp1.000		
	500	Sistem berhasil mendeteksi nominal Rp500		
40 lux	100rb	Sistem berhasil mendeteksi nominal Rp100.000		
	50rb	Sistem berhasil mendeteksi nominal Rp50.000		
	20rb	Sistem berhasil mendeteksi nominal Rp20.000		
	10rb	Sistem berhasil mendeteksi nominal Rp10.000		
	5rb	Sistem berhasil mendeteksi nominal Rp5.000		
	2rb	Sistem berhasil mendeteksi nominal Rp2.000		
	1rb	Sistem berhasil mendeteksi nominal Rp1.000		
	1000	Sistem berhasil mendeteksi nominal Rp1.000		
	500	Sistem berhasil mendeteksi nominal Rp500		

Intensitas Cahaya	Uang	Hasil yang Diharapkan	Hasil Pengujian	Dokumentasi
65 lux	100rb	Sistem berhasil mendeteksi nominal Rp100.000		
	50rb	Sistem berhasil mendeteksi nominal Rp50.000		
	20rb	Sistem berhasil mendeteksi nominal Rp20.000		
	10rb	Sistem berhasil mendeteksi nominal Rp10.000		
	5rb	Sistem berhasil mendeteksi nominal Rp5.000		
	2rb	Sistem berhasil mendeteksi nominal Rp2.000		
	1rb	Sistem berhasil mendeteksi nominal Rp1.000		
	1000	Sistem berhasil mendeteksi nominal Rp1.000		
	500	Sistem berhasil mendeteksi nominal Rp500		

Format pengujian diatas digunakan untuk menjelaskan mengenai pengujian model yang akan diuji menggunakan tiga lux yang berbeda, dengan masing-masing nominal uang yang nantinya akan diuji sebanyak 20 kali dan sebagai data tercatat yang nantinya akan disajikan dalam bentuk grafik agar dapat dilihat lebih jelas perbandingannya, yang dimana grafik tersebut akan ditampilkan pada hasil pengujian model.

Kemudian setelah dilakukan pengujian model, dilakukan juga pengujian untuk hasil perancangan perangkat keras dan juga perangkat lunak secara

keseluruhan, untuk tabel pengujian yang akan dilakukan dapat dilihat pada Tabel 3.6.

Tabel 3. 6 Format Pengujian Perangkat Keras

Fitur Pengujian	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Koneksi Wi-Fi	Menghubungkan ESP32 dengan jaringan Wi-Fi	ESP32 dapat terhubung ke jaringan W-Fi	
Kinerja LCD	Menampilkan informasi sistem pada layar LCD	Informasi tampil dengan jelas dan sesuai data	
Kinerja RFID	Menempelkan kartu/tag RFID pada sensor	Sistem membaca ID RFID dengan benar	
Kinerja servo	Memberikan perintah buka/tutup pada servo	Servo bergerak sesuai perintah (berputar/berhenti)	
Kinerja buzzer	Memberikan sinyal instruksi melalui buzzer	Buzzer berbunyi sesuai instruksi	
Kinerja relay	Mengaktifkan dan menonaktifkan relay	Relay dapat berfungsi sebagai saklar elektronik	
Kinerja solenoid doorlock	Memberikan perintah kunci/buka kunci pada solenoid	Solenoid dapat mengunci dan membuka sesuai instruksi	
Kinerja buck converter	Mengukur output dari buck converter	Output tegangan sesuai kebutuhan perangkat	

Fitur Pengujian	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Kinerja terminal	Menghubungkan kabel ke terminal untuk distribusi daya	Koneksi stabil tanpa gangguan	
Kinerja jack converter\	Menghubungkan adaptor melalui jack converter	Daya dapat masuk dan didistribusikan dengan baik	
Kinerja adaptor 5v	Menghubungkan adaptor 5V ke perangkat	Perangkat menyala dengan stabil	
Kinerja adaptor 12v	Menghubungkan adaptor 12V ke perangkat	Perangkat menyala dengan stabil	

Jika pengujian pada Tabel 3.6 sudah selesai dilakukan, maka tahap selanjutnya adalah melakukan pengujian pada perangkat lunak. Format pengujian *black box* perangkat lunak dapat dilihat pada Tabel 3.7.

Tabel 3. 7 Format Pengujian *Black Box* Perangkat Lunak

Fitur Pengujian	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
<i>Login</i> pengguna	Pengguna memasukkan username dan password yang valid	Sistem menerima input dan menampilkan halaman dashboard	
<i>Login</i> dengan kredensial salah	Pengguna memasukkan username atau	Sistem menolak login dan menampilkan	

Fitur Pengujian	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
	password yang salah	notifikasi error	
<i>Dashboard</i>	Setelah login berhasil, sistem menampilkan halaman utama dashboard	Dashboard tampil dengan informasi yang sesuai dengan data	
Reset total tabungan	Admin mengklik tombol reset balance	Data tabungan siswa direset sesuai perintah dan tersimpan di database	
Daftar Siswa	Admin membuka menu daftar siswa	Sistem menampilkan daftar siswa beserta data yang telah tersimpan	
Tambah data siswa	Admin menambahkan data siswa baru melalui form input	Data siswa baru tersimpan ke database dan langsung muncul di daftar siswa	
Edit data siswa	Admin memilih salah satu siswa kemudian mengubah data melalui form edit	Data siswa diperbarui di database dan tampilan daftar siswa menyesuaikan	

Fitur Pengujian	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
<i>Delete</i> data siswa	Admin memilih salah satu siswa dan menekan tombol delete	Data siswa terhapus dari database dan tidak lagi muncul di daftar siswa	
Daftar Tabungan	Admin membuka menu daftar tabungan	Sistem menampilkan daftar tabungan siswa sesuai dengan data yang tersimpan	
<i>Logout</i> Pengguna	Pengguna memilih menu logout	Sistem keluar dari akun dan kembali ke halaman login	

Melalui pengujian pada Tabel 3.7 ini, nantinya sistem tidak hanya dapat diakses dengan baik, tetapi juga mampu memberikan keluaran sesuai dengan masukan yang diterima. Selain itu, pada celengan pintar ini, digunakan juga sistem notifikasi telegram untuk mengirimkan notifikasi secara *real-time* mengenai aktivitas yang dilakukan pada celengan. Untuk lebih jelasnya mengenai pengujian pada telegram, terdapat format pengujian pada Tabel 3.8 berikut.

Tabel 3. 8 Format Pengujian Telegram

Fitur Pengujian	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Notifikasi admin membuka kunci	Admin tap kartu pada modul RFID	Celengan berhasil terbuka dan mengirimkan notifikasi bahwa akses admin	

Fitur Pengujian	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
		diterima dan <i>door lock</i> terbuka	
Notifikasi siswa menabung	Siswa tap kartu pada modul RFID dan memasukkan uang ke dalam celengan	Celengan mendeteksi ID dan nominal uang, lalu mengirimkan notifikasi berupa data ID, Nominal, Waktu, dan Total tabungan	
Notifikasi kartu tidak dikenal	Kartu tidak dikenal tap kartu pada modul RFID	Akses kartu ditolak oleh celengan, dan mengirimkan notifikasi ke telegram bahwa akses ditolak dan kartu tidak terdaftar	
Notifikasi admin reset tabungan	Admin klik fitur tombol <i>reset balance</i> pada <i>dashboard website</i>	Admin berhasil reset saldo tabungan dan mengirimkan notifikasi bahwa admin mereset saldo tabungan	

Setelah pengujian pada Tabel 3.8 ini, nantinya notifikasi telegram dapat digunakan sebagai akses terbuka untuk melihat aktivitas yang dilakukan pada

celengan, mulai dari akses yang dilakukan oleh admin ataupun siswa, sehingga segala aktivitas yang terjadi, dapat tercatat dan menjadi transparansi terhadap semua pengguna celengan.